Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 7.056

IJCSMC, Vol. 9, Issue. 11, November 2020, pg.77 – 97

A SYSTEMATIC BROAD REVIEW ON SOFTWARE SECURITY TESTING IN DIFFERENT WAYS

Suman
Research Scholar, Department of Information Technology
Babasaheb Bhimrao Ambedkar University (A Central University) Lucknow
DOI: 10.47760/ijcsmc.2020.v09i11.008

ABSTRACT: ***In this modern era, mostly all wireless software applications are hacked and injected because of several harmful activities. Henceforth, different frameworks were developed to measure the security level in software systems. Especially, the penetration testing (PT) model is mostly worn by all applications to make the attack vulnerability assessment. To secure the system, initially the behaviour of attacks should be analysed and that is trained in specific agent or software module. Based on the trained activities the specific verification system can predict the present malicious activities. Furthermore, the verification agent or system should be automated to find the harmful activities in any time. Also, the PT is utilized in both hardware and software gadgets, it is processed based on the support of software functions. In other case, the Machine Learning (ML) and Reinforcement Learning (RL) model is functioned to process the attack behaviour in software systems. Moreover, in this review the usage of PT is elaborated in dissimilar ways for different applications. Subsequently, the performance of PT framework is differed based on its each software or applications.***
Keywords: Software, Penetration Testing, attack, vulnerability assessment, network, wireless medium

1. INTRODUCTION

Nowadays, the wireless interconnected digital application is growing very fast; also the wireless technology is significant in all fields such as, economics, government, etc [1]. Beside these each individuals also need this wireless less environment every day [2]. Thus the wireless technology is become a part of life [16]. Moreover, to collapse the wireless technology and to improve the hacking technology [40], several attacks are created and launched in network system [4]. In addition, there are several types of attacks some attacks

are stole the password and secret pin number of connected device to hack the system and some attacks monitor [43] and control the entire connected system [17]. To check the hardware components like computer or any other Internet of Things (IoT) penetration examine procedure is used [18]. Testing in software turn the interest of network operators, because the continuous delay in any application [19] is raise because of software error or error in code implementation. In recent the software test based on penetration method ear high accuracy rate of attack prediction and characterization [8]. Thus a numbers of threat based testing models are introduced to detect the unexpected abnormal activities during the execution of code [20].

The key focus of this PT model is to specify the threat during code execution [10]. Moreover, several units are present in the software model each unit are linked with one another and integrated with whole system [11]. The performance of the system is evaluated based on the unit intersection in the entire system [12]. In this modern decade, the tool PT is designed with many facilities like scanners, attack vulnerability assessment, security evaluation [21]. Thus, the designed facilities improve the system performance [14]. Moreover, the Pt also functioned based on two roles that are human interface based PT and automated PT [22]. For the Human interface PT, the each attack behaviour should be trained before the process from that the PT model can detect the attack [23] while it happens in software framework [24]. But for the automated system the detection process is in the manner of automated, one of the agent is fixed in the system to monitor the harmful activities [25].

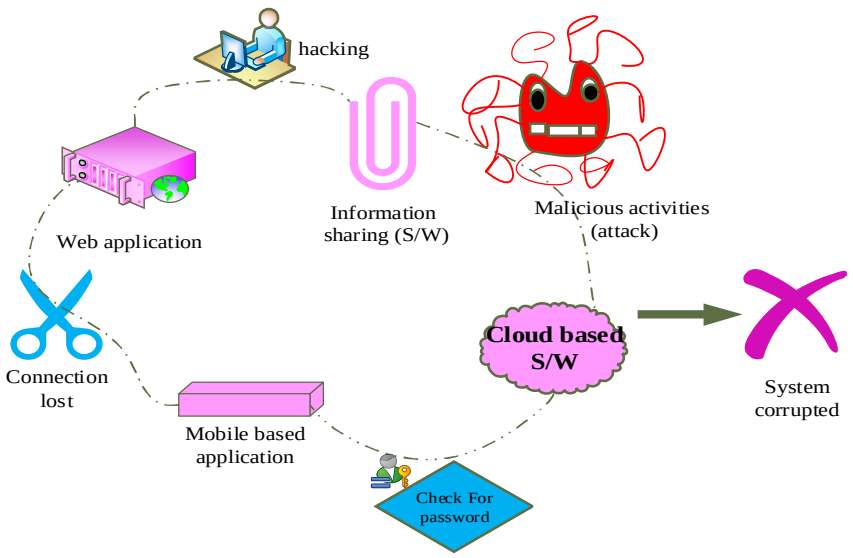

Fig.1 Process of Software's and network system with different interruption

PT is the best approach that is mostly used in software testing but the PT is the both hardware and software model [26]. The process of network application with the presence of some interruption is elaborated in fig.1. Thus the process of PT is functioned manually, and its examine series is very long [28]. To enhance the efficiency of software testing [29], automated tools and methods should introduced [30]. In addition, several automated PT systems are introduced but most of tools only scan the attack vulnerabilities [31], also it is scalable in nature. So it could not have the capability to function in dynamic fields [32]. Henceforth, this detailed review explained advantages and process of each PT model.

2. PT BASED SECURITY EVALUATION

NuthanMunaiah *et al* [46] proposed a completion approach to collect and monitor the behaviour of harmful activities. The developed model is functioned based on penetration testing. Moreover, crucial security in wireless management is software security; in many cases the control measure of software frame model is questionable. Beside these, a function adversary in the network medium is severe threats in software system. In addition, the success rate of the projected model is evaluated using some specific dataset. The attack prediction in network is crucial task, Zukri *et al* [47] proposed a novel penetration examine system to describe the behaviour of attack vulnerabilities. To monitor the behaviour of attack vulnerabilities one of the agents is injected in the system. Moreover, the process of agent penetration system is done by reinforcement learning mechanism. In addition, the cryptography process also functioned to secure the secret information.

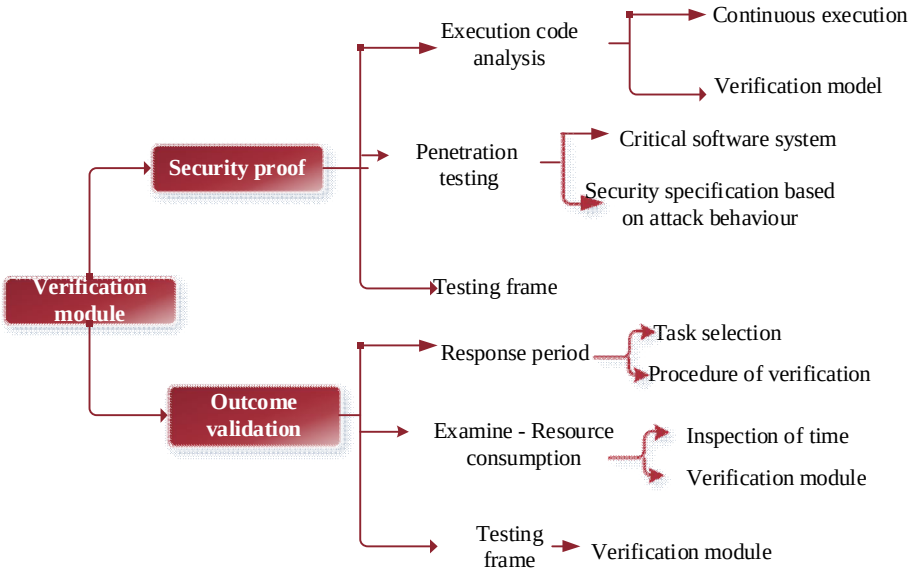

Fig.2 Verification procedure to check the security

To maximize the production rate in digital flied, software quality testing is the major concern. In addition, less production rate and low quality rate in digital application is attained because of software damage. The verification procedure of software testing is detailed in fig.2. The quality of software is estimated in the manner of series frame work because estimating the software for it harmful activities are the continuous process. Regarding this, Subhash Chandra Jat *et al* [48] proposed agreement based on service level to enhance the process of penetration test. Henceforth, in this proposed model the dissimilar way of penetration test is surveyed to improve the process of penetration test. This kind of examining offers a compressed and controlled method to identify the security problems.

2.1 ML with PT for evaluating attack vulnerability

The advances of AI techniques turn the researcher's interest towards to improve the specific application based on AI techniques [33]. On the other hand, the penetration methods are varied in complexity and computational measures [34].

The general process to protect the network from abnormal activities is termed as penetration test. In present several different penetration models are discovered but those are high resource consuming and standard less. In the same way different ML models are developed however, each model met specific threat. So, Mohamed C *et al* [49] projected an intelligent based automation with industrial frameworks to enable the test details after the completion of software inspection. Thus the PT plays a chief role in cyber security area to audit the statistics and protect the software from the harmful activities. Protecting the digital sector and modern application is the crucial task in network environment. In other definition, an exercise of security is penetration test; it supports to run the software as fast as earlier. Fabio Massimo Zennaro& Laszlo Erdodi [50] have analysed several risk module to develop an efficient penetration technique to estimate the software quality. Consequently, reinforcement Q-learning strategy is projected to find the hacking activities in the network medium. PhilippZech *et al* [51] announced a new approach to test and qualify the software based on log regression model. This proposed model is functioned based on the interface level to test the software quality. Furthermore, the risk profile is updated based on the analysis of security risk in an entire system. Here, the logic regression means, a pack of rules that is applied to predict the harmful behaviour.

2.2 AI base PT to make the system assessment

System PT is a significant frame to maintain the security in an information system. Yu Ye *et al* [52] described the difficulties during the process of PT, the high risk is found during the prediction of artificial storm. The key challenge rise in web application is rising loses in the form of security flaws. Here, J. K. Alhassan *et al* [53] elaborated a novel fuzzy model is created in assessment testing approach to protect the stored information. Several neural networks are applicable to solve the problem in software tools to estimate the security level in each application. Artem Tetskyi *et al* [54] developed choice based neural system to solve the security problems.

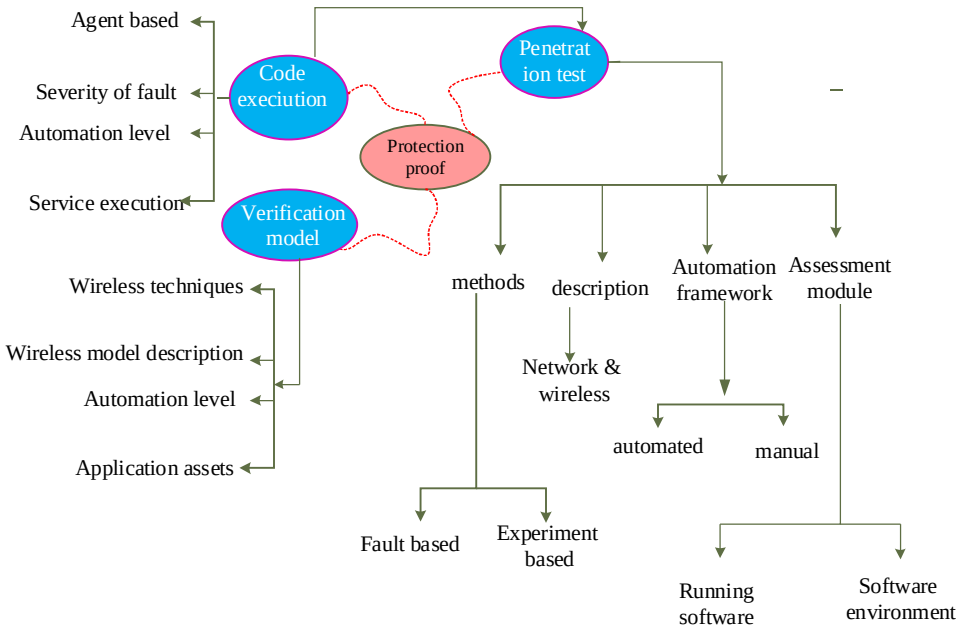

Fig.3 Assessment of protection model

The assumption of fault generally includes location of fault node location of data and target devices [35]. The assessment of protection model is shown in fig.3. Thus the attack models are functioned based on the each specific program. In another module, the PT is applied in WSN to estimate the system security. Moreover, this test contains assessment report based on the measure of system vulnerability. Thus the PT is processed in the manner of both wired and wireless model. George D. O'Mahony *et al* [65] developed radios framework based on software defined model, this model has provided whole security for entire system to improve the system performance. It evaluation is shown in fig.4.

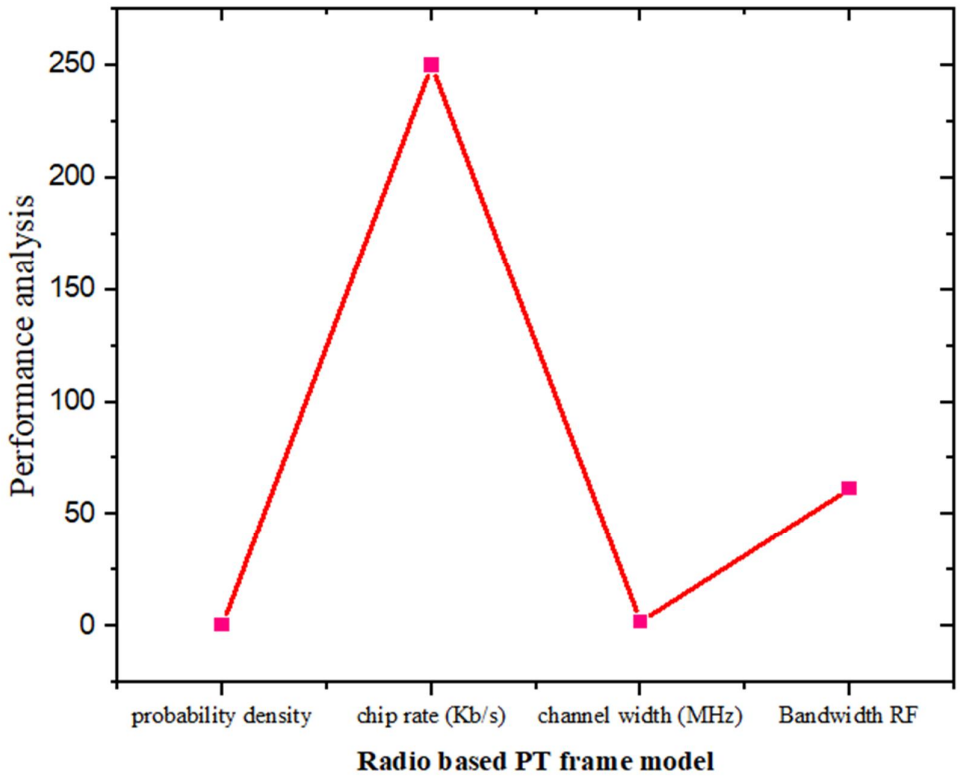

Fig.4 Performance of radio based model

3. ATTACK VULNERABILITIES ASSESSMENT IN SOFTWARE

The sudden growth of computer application maximizes the complex rate to process the operation [36]. Because of its complexity and several software models it is vulnerable to attack. The presence of attack in software model causes several harmful activities like hacking, id stole, etc. For that YuganshKhera *et al* [55] projected assessment of attack vulnerabilities, thus the develop model updated the status of security measures frequently. Seungsoo Lee *et al* [56] described Software defined PT to validate the attack vulnerabilities in software environment. In the result, the successive rate of software defined model is estimated by analysing the test report. Thus the projected model records the 26 types of attack behaviour. Moreover, here the proposed PT is names as DELTA. The wireless connected device is the key paradigm in digital world; several smart devices are fixed with devices and various home applications to monitor and to achieve its task. Because of wireless environment the harmful activities occur in network medium is invisible, so that it is vulnerable to attack [70]. Moreover, to prevent all kinds of attack scheduling the types of harmful activities is the key concern.

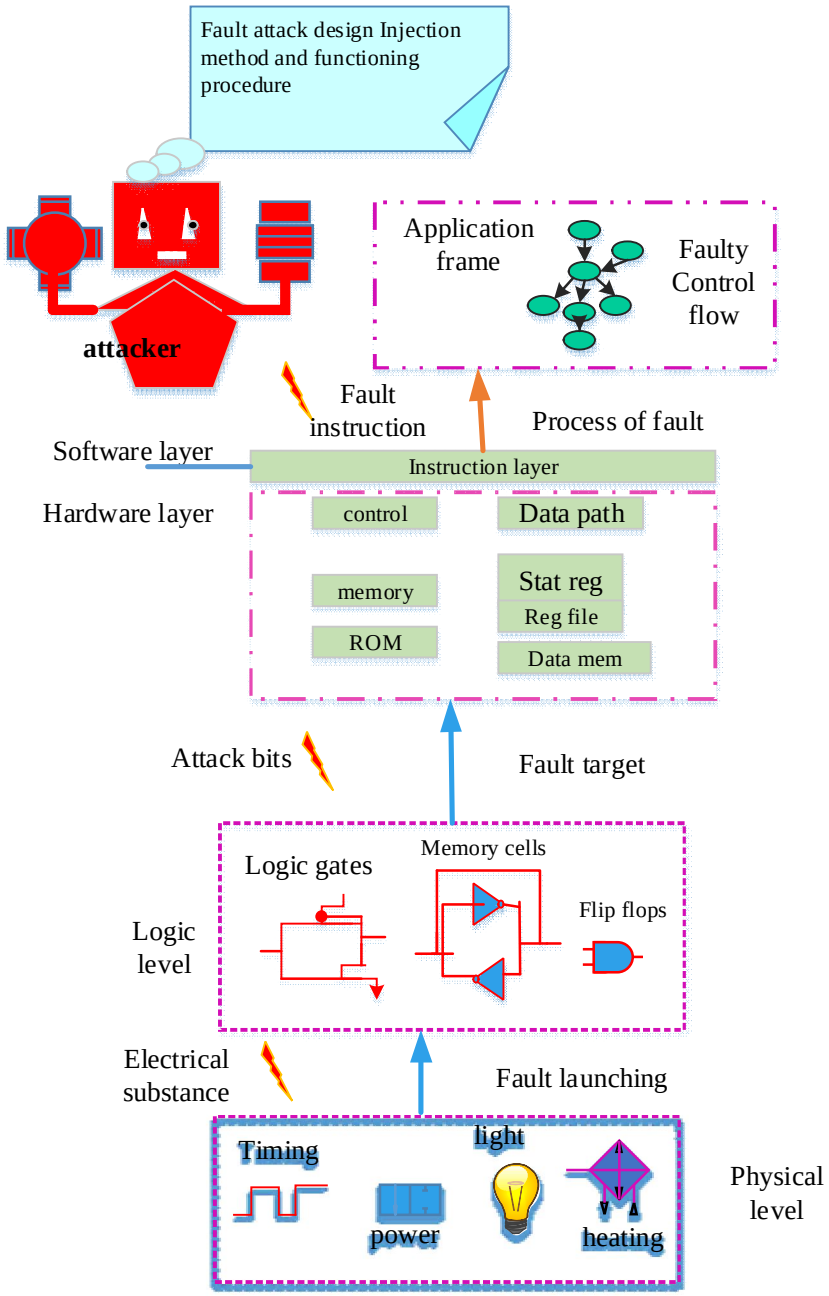

Fig.5 Attack module in software system

The process of attack module in software system is elaborated in fig.5. Thus Bako Ali *et al* [57] made the assessment for attack vulnerabilities in home smart applications. In addition, from this assessment the most of attack vulnerabilities are highlighted and recorded to make the aware about these kinds of malicious activities [69]. With the rapid growth of smart grid creation, the physical system of cyber is designed different multi dimensional system that is heterogeneous in nature. The interface among the power campaigns might maximize the

depended to the distributed cyber and power system. Rong Fu *et al* [58] developed an assessment for security measures to validate the level of attack vulnerabilities. Finally, the validation results analysed the throughput variation of different methods. On the other hand an important system in network is low power consumed network connection. The wireless smart applications are functioned based on the support of software system. For that Rashmi Sahay *et al* [59] announced the rank based model to arrange the sensor nodes in wireless environment. Also, the harmful activities like denial of Service (DoS) in distributed manner cause huge damage to the software system. So to evaluate the powerful rate of attack a graphical system is updated, at the final several graphs are obtained to reflect the attack effectiveness. The chief reason of security threat is expansion of cloud applications [67]. Also estimating the security range of cloud application is not simple task. So, numeral kind of monitoring system is developed to observe the behaviour of software system while running [68]. Especially, PT is the finest power module to test the current running software with good accuracy. So to provide the good assessment, the framework of penetration model should be enhanced. To predict the types of harmful activities the tool PT should have the knowledge about behaviour of attack. So, Valentina Casola *et al* [60] developed a knowledge based system in PT to identify the harmful activities with high accuracy. Dissimilar, methods are processed to strengthen the security in web application including mobile. The most frequently utilized framework to make the detail assessment of system security is PT. Thus Michele Peroli *et al* [61] proposed model based system as security validating model; this approach is automated often to achieve the task. Also, several case studies are evaluated to check the success rate of the developed model. In other case, Tomas Zitta *et al* [62] explained that PT as intrusion prediction system in all software applications. Moreover, when the PT is processed as defeat prediction model then it is flexible in nature.

3.1 Software agent

The subset of ML is reinforcement learning [37], the quality assessment of software is validated with some specific kind of software agent [38]. In reinforcement learning, humans supports also need in some cases, these issues reduce the uniqueness of the system [39].

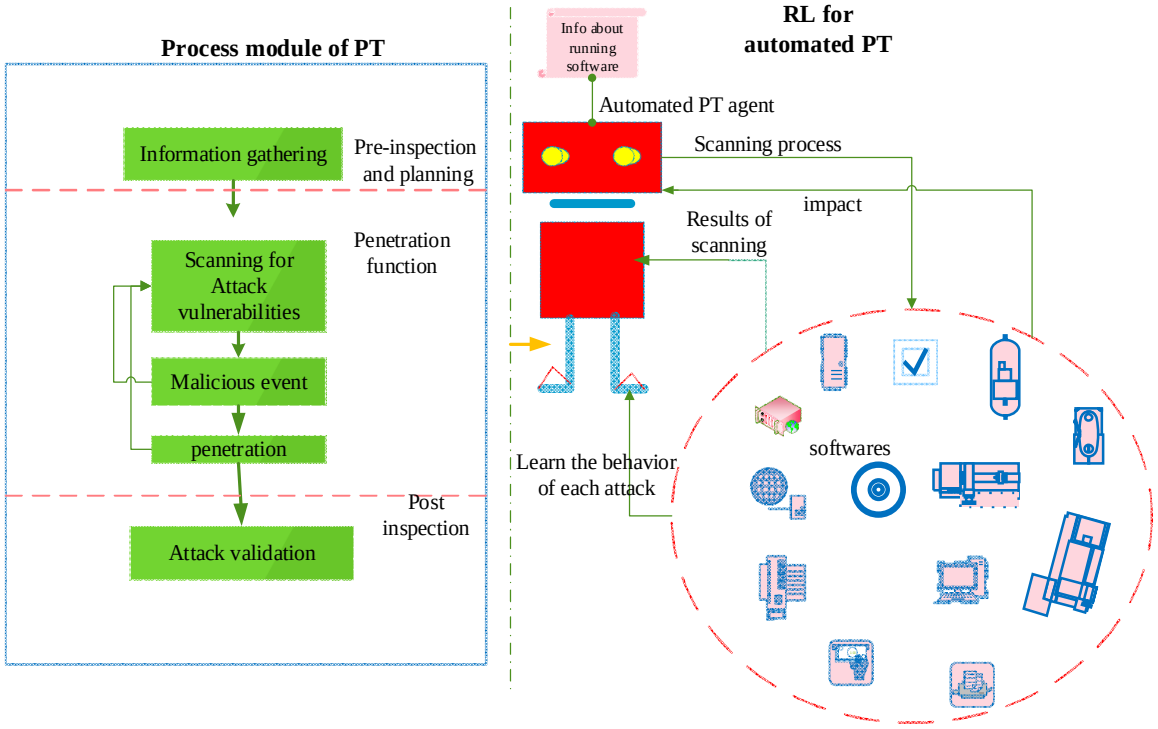

Fig.6 Agent based testing model

So **Sujita Chaudhary*et al*** [63] developed automated based post breach PT approach to assess the security level and attack efficiency. To process this function, primarily, one of the software agents is train to the system. The agent based security examine model is shown in fig.6. **Ge Chu and Alexei Lisitsa [64]** introduced belief based agent PT systems; this agent initiated its function based on its target applications. Before the initiation of function process it stores all target information, so when any information is get injected then it automatically detects that. Thus it attained massive accuracy for software testing. The practice of software security is processed to reduce the attack vulnerabilities. Thus Gary McGraw *et al* [44] conducted the software security framework process to diminish the vulnerabilities of attack. The most vulnerabilities models are utilized to find the error in execution code, because the error in execution node might increase the attack vulnerability rate. Hala Assal and Sonia Chiasson [42] describe several steps to improve security measure in each application. Here, the analyzement is initiated with design stage several themes are designed to provide the security for different applications. Malicious activities have two frames that are fault attack and attack design. To attack the target system, the adversaries inject the harmful program into several ways like encryption module, monitoring module etc. BilgidayYuce *et al* [41] noticed that each harmful activity functioned through each instruction cycle. Also the software model for PT was designed based assumption of hardware execution.

4. Performance Testing

The report of this detailed review is described in this performance testing section. The method PT has been processed from past decade to still now, sever model in PT is introduced to check the security level for each software and hardware components. Moreover, The PT is utilized for assessment frame work, different studies described and process the PT procedure in different way to check the attack occurrences possibility of each system. The performance of some specific model is shown in table.1 and table.2.

Table.1 Performance of some specific reviewed articles

Author	approach	Merits	Demerits
NuthanMunaiah *et al* [46]	MITRE ATT&CK frame	• Penetrating method • Functioning by controlling the software process • Record the behaviour of attack	• It utilized need more resources • It takes too time
Zukri *et al* [47]	Training agent and crypto model	• Monitoring mechanism is utilized to monitor the system performance • Password control • Also encryption function is used to hide the information from the attack	• In some case the encryption algorithm is break by malicious activities
Subhash Chandra Jat *et al* [48]	Agreement model	• It process the PT model in different way to detect the harmful activities • Examine model	• But the attacks like DoS escape from its monitoring

Mohamed C *et al* [49]	Intelligent based automation	• It worn to audit the data • Predict the attack like cyber • Very less processing time	• Complex to design
Fabio Massimo Zennaro & Laszlo Erdodi [50]	Reinforcement Q-learning	• It inspects several risk model • Find hacking activities	
PhilippZech *et al* [51]	Log regression model	• It developed interface model • It updates the risk profile	• The rules are differed based on the attack efficiency
J. K. Alhassan *et al* [53]	Fuzzy approach	• It created assessment profile of attack vulnerability • Software estimation	
ArtemTetskyi *et al* [54]	Choice based neural system	• Initially, it evaluates the characteristics of each attack	• This model attained very less accuracy in attack prediction
SeungsooLee *et al* [56]	Software defined PT	• It produced the report of software test • It records 26 types of attack	

Bako Ali *et al* [57]	Assessment frame work	• It checked the vulnerability of homemade applications • Make the awareness about malicious activities	• This model is failed to predict the attack in other applications.
Rong Fu *et al* [58]	Security assessment	• It measure the level of attack vulnerability • The variation of throughput is estimated	
RashmiSahay *et al* [59]	Rank based model	• The efficiency of each attack is represent in graphical structure	• Because of graphical representation it needs more resources
SujitaChaudhary *et al* [63]	Machine learning Training agent	• Cyber-attack prediction • It check the vulnerability of system frequently • Automated PT • Check for sensitive files	

Table.2 Performance Statistics of specific Software testing model

Author	techniques	model	applications	version	Severity analysis	design	Tool
Amankwah *et al* [1]	automated framework	Vulnerability checking	• SQL injection • Site scripting	2.5	8	medium	OWASP ZAP
Corteggiani *et al* [3]	embedded frame work	Inspection debugger	• C++	1.3	8 crashes	complex	KLEE
Thales Teixeira *et al* [5]	V2P	Performance investigation	Wi-Fi	802.11		complex	OMNeT++ 50s
Aparicio Carranza *et al [6]*	Investigation of clustered connected device	Verification model	• Raspberry pi • python	4,7,1	3 to 4 node	-	Kali linux
Schwartz *et al* [7]	Automated PT	RL	C++	3.5.0	1	hard	python
Casola*et al* [9]	Cloud PT	Agreement level	-	-	-	hard	OpenVas tool
Mahmoodi *et al* [12]	Virtual prototype	Model-driven scheme	C/C++ H/W or S/W	-	-	complex	Sniffer tool
Luswata *et al* [13]	External PT	Monitoring model		3.0	9 to 10	-	Smod
Lescisin *et al* [15]	Dynamic analysis	Software based	• java • HTML • SQL	71293		complex	Memory safety tool

Anomaly Detection System (ADS) is capable to process in multi software environment. Here the key module in the projected method is heterogeneous model. In addition it supports to detect the malicious behaviour in hardware components also based on PT method [27]. Its outcome is described in fig 7.

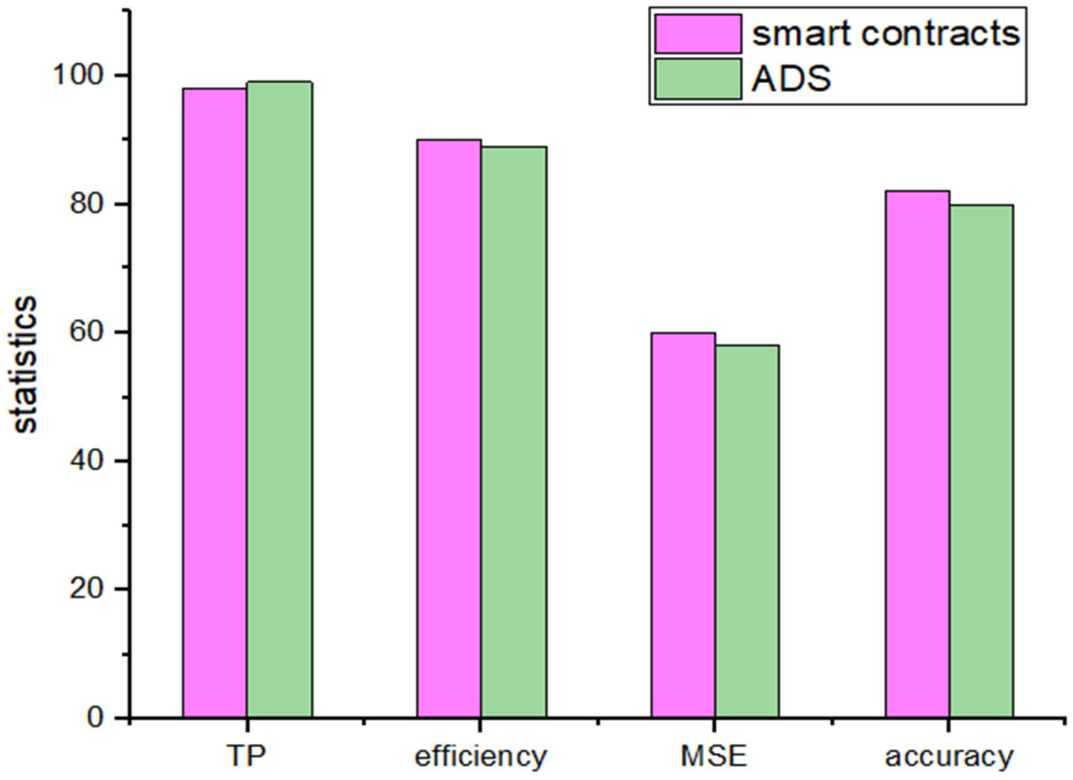

Fig.7 performance of ADS and smart contracts approach

After the scanning process, vulnerabilities and detected attacks are list out. In addition, the tool OpenVAS is operated manually to find the attack in connected devices. Consequently based on the report attack specification is processed. On the other hand, block chain the trending technology in software environment to store the large records but it lack in security. Several cryptographic techniques were introduced to stop the security threat but still it is not up to desired level. So that Reza M. Parizi *et al [66]* proposed automatic smart contracts model to check the attack vulnerability often. Moreover, in this recent era the method smar contracts for block chain is growing very fast. Thus it contains several model in that specific model smart contracts.

The validation of each method is unique here, Valentina Casola *et al* [9] estimated the number of present user in the network layer for 1 min. Moreover, cloud PT model is introduced to measure the security assessment and attack vulnerabilities. In cloud computing environment, one of the virtual machines is act as scanning tool. Thus the scanning process is performed by several test exploit that is shown in fig.8.

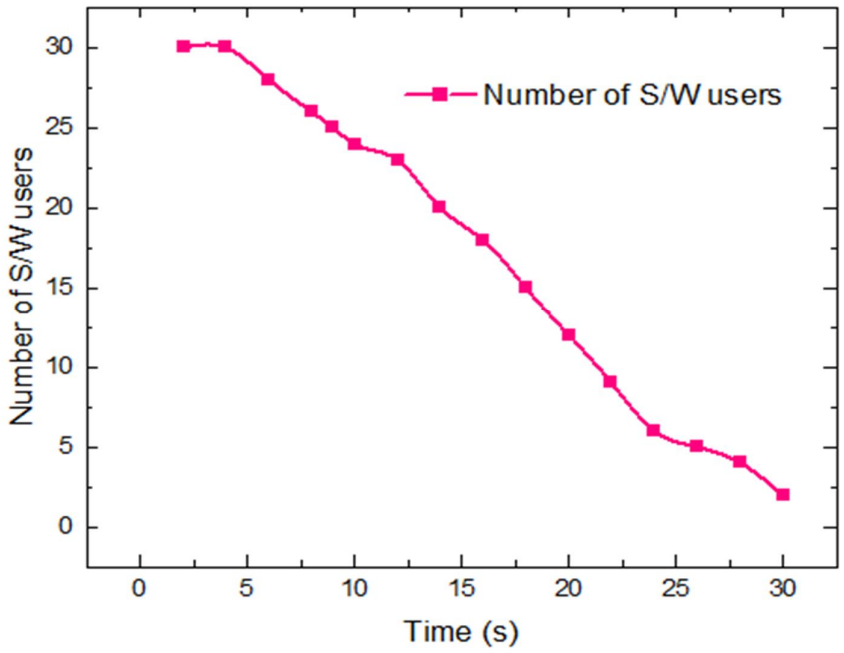

Fig.8 Statistics of Cloud PT model

The wireless paradigm is automated for data transmission, but it lacks in security because of harmful activities. So Thales Teixeira de Almeida *et al* [5] did the security validation in wireless frame model to reduce the security risk. In addition, the application adopted for that research is Vehicular framework, i.e Vehicular to peer (V2P). It function process is detailed in fig.9. To make the security check in software's, PT is the key methods that involves different functions [5]. Here, AI is used for automated PT model. Thus, the attained grades are exposed in fig.9.

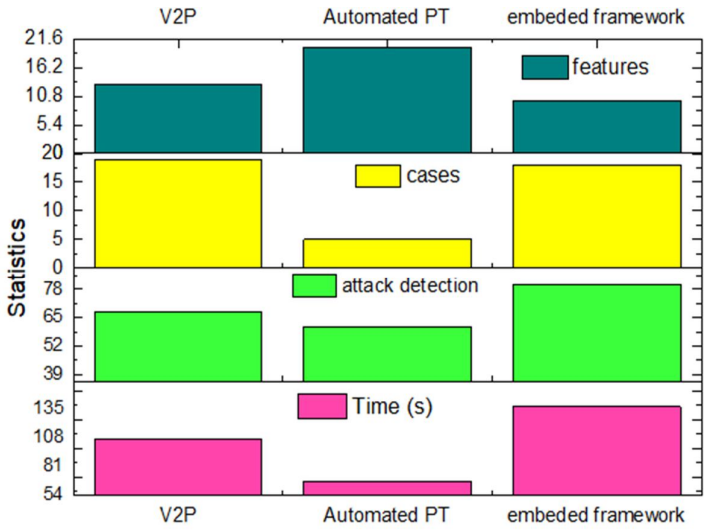

Fig.9 Performance of V2P, Automated PT and Embedded system

Embedded models are efficiently process in several appliances but still the connected gadgets meet several issues because of less security. As the reason of less security the embedded model is vulnerable to get attack [3]. Thus the performance of embedded approach is elaborated in fig,9. In some cases, dynamic model are better than static model to evaluate that Michael Lescisin and Qusay H. Mahmoud [15] described the research based on dynamic tools.

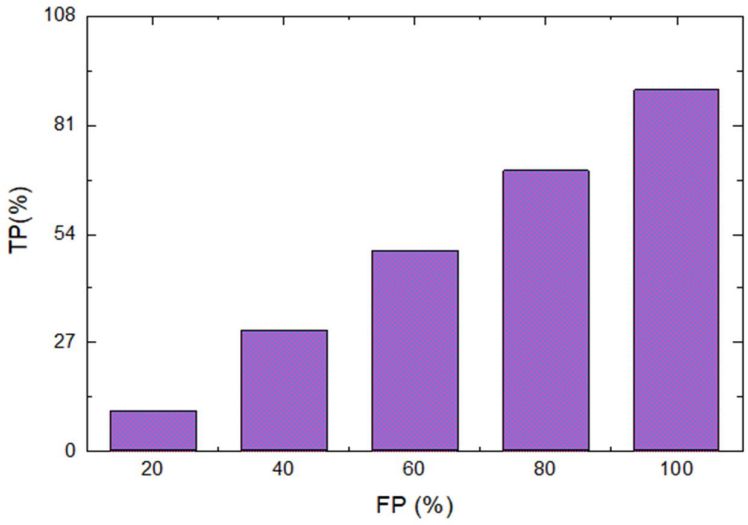

Fig.10 Performance of dynamic tools

The dynamic tools are run with the support of software models such as java, SQL and HTML. Finally, the benefits and limitations are listed systematically. Moreover, the leakage in memory model is happened because of data overload, so to avoid the leakage data traffic should be reduced. Thus the success rate of dynamic tool analysis is explained in fig.10. Thus from the validation, it is proved that software testing is differed and unique for each method.

5. Conclusion

From the broad review several approaches are studied and described based on security testing. In the most events, some literatures described that to detect and prevent the faulty activities in network or software application, pre knowledge about the behaviour of attack is more important. Moreover, the reviewed approaches verified that without the knowledge of malicious behaviour detection of attack is impossible. So that in many cases the security test framework is failed. In addition, the scheme PT is mostly worn in all applications to check the malicious vulnerability assessment. Also the PT model is processed in different way for different purpose to meet the target; its key target is to report the malicious events and vulnerability measures of connected system. On the other hand agent based model is introduced with ML and RL approaches to improve the performance of PT model. But still the high accuracy of PT model for attack detection is 80%. So to enhance the exactness measure of attack identification in future, a hybrid deep learning model with hybrid optimization approach should be developed. Moreover, it will reduce the design complexity rate.

REFERENCES

[1] Amankwah, Richard, et al. "An automated framework for evaluating open-source web scanner vulnerability severity." Service Oriented Computing and Applications (2020): 1-11.
[2] Sun, Yuyi, et al. "High-Confidence Gateway Planning and Performance Evaluation of a Hybrid LoRa Network." IEEE Internet of Things Journal (2020).
[3] Corteggiani, Nassim, Giovanni Camurati, and AurélienFrancillon. "Inception: System-wide security testing of real-world embedded systems software." 27th {USENIX} Security Symposium ({USENIX} Security 18). 2018.
[4] Maraj, Arianit, ErmirRogova, and GencJakupi. "Testing of network security systems through DoS, SQL injection, reverse TCP and social engineering attacks." International Journal of Grid and Utility Computing 11.1 (2020): 115-133.
[5] de Almeida, Thales Teixeira, et al. "Wi-Fi Direct Performance Evaluation for V2P Communications." Journal of Sensor and Actuator Networks 9.2 (2020): 28.
[6] Aparicio Carranza, Mahendra Ganesh, Harrison Carranza, and CasimerDeCusatis. "Performance Evaluation of a Raspberry Pi Bramble Cluster for Penetration Testing."

[7] Schwartz, Jonathon, and Hanna Kurniawati. "Autonomous penetration testing using reinforcement learning." arXiv preprint arXiv:1905.05965 (2019).
[8] McKinnel, Dean Richard, et al. "A systematic literature review and meta-analysis on artificial intelligence in penetration testing and vulnerability assessment." Computers & Electrical Engineering 75 (2019): 175-188.
[9] Casola, Valentina, et al. "Towards automated penetration testing for cloud applications." 2018 IEEE 27th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE). IEEE, 2018.
[10] Seng, Lim Kah, NorafidaIthnin, and Syed ZainudeenMohd Said. "The approaches to quantify web application security scanners quality: a review." International Journal of Advanced Computer Research 8.38 (2018): 285-312.
[11] Ribeiro, Victor Vidigal, Daniela Soares Cruzes, and GuilhermeHortaTravassos. "A Perception of the Practice of Software Security and Performance Verification." 2018 25th Australasian Software Engineering Conference (ASWEC). IEEE, 2018.
[12] Mahmoodi, Yasamin, et al. "Attack surface modeling and assessment for penetration testing of IoT system designs." 2018 21st Euromicro Conference on Digital System Design (DSD). IEEE, 2018.
[13] Luswata, John, et al. "Analysis of SCADA security using penetration testing: A case study on modbus TCP protocol." 2018 29th Biennial Symposium on Communications (BSC). IEEE, 2018.
[14] István, Paráda. "Basic of Cybersecurity Penetration Test." Hadmérnök 13.3 (2018): 435-442.
[15] Lescisin, Michael, and Qusay H. Mahmoud. "Evaluation of Dynamic Analysis Tools for Software Security." International Journal of Systems and Software Security and Protection (IJSSSP) 9.3 (2018): 34-59.
[16] Farhan, AR Shehab, and GM MostafaMostafa. "A methodology for enhancing software security during development processes." 2018 21st Saudi Computer Society National Computer Conference (NCC). IEEE, 2018.
[17] Muñoz, Fernando Román, Esteban Alejandro Armas Vega, and Luis Javier GarcíaVillalba. "Analyzing the traffic of penetration testing tools with an IDS." The Journal of Supercomputing 74.12 (2018): 6454-6469.
[18] Munaiah, Nuthan. "Assisted discovery of software vulnerabilities." Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings. 2018.
[19] Pozdniakov, K., et al. "Smart Computer Security Audit: Reinforcement Learning with a Deep Neural Network Approximator." (2020).
[20] Chou, Te-Shun, and John Jones. "Developing and Evaluating an Experimental Learning Environment for Cyber Security Education." Proceedings of the 19th Annual SIG Conference on Information Technology Education. 2018.
[21] Backman, Lars. "Why is security still an issue?: A study comparing developers' software security awareness to existing vulnerabilities in software applications." (2018).
[22] Kalogranis, Christos. AntiVirus software evasion: an evaluation of the AV Evasion tools. MS thesis. Πανεπιστήμιο Πειραιώς, 2018.
[23] Abu-Dabaseh, Farah, and EsraaAlshammari. "Automated Penetration Testing: An Overview." Computer Science & Information Technology (2018).
[24] Shmaryahu, Dorin, et al. "Simulated penetration testing as contingent planning." Twenty-Eighth International Conference on Automated Planning and Scheduling. 2018.

[25] Venson, Elaine, et al. "Costing secure software development: A systematic mapping study." Proceedings of the 14th International Conference on Availability, Reliability and Security. 2019.
[26] Speicher, Patrick, et al. "Towards automated network mitigation analysis." Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing. 2019.
[27] Khreich, Wael, et al. "Combining heterogeneous anomaly detectors for improved software security." Journal of Systems and Software 137 (2018): 415-429.
[28] Svilicic, Boris, et al. "Assessing ship cyber risks: a framework and case study of ECDIS security." WMU Journal of Maritime Affairs 18.3 (2019): 509-520.
[29] Elsayed, Marwa, and Mohammad Zulkernine. "Offering security diagnosis as a service for cloud SaaS applications." Journal of information security and applications 44 (2019): 32-48.
[30] Gehrmann, Christian, and Martin Gunnarsson. "A digital twin based industrial automation and control system security architecture." IEEE Transactions on Industrial Informatics 16.1 (2019): 669-680.
[31] Bhardwaj, Akashdeep, and Sam Goundar. "A framework to define the relationship between cyber security and cloud performance." Computer Fraud & Security 2019.2 (2019): 12-19.
[32] Anisetti, Marco, et al. "Test-based security certification of composite services." ACM Transactions on the Web (TWEB) 13.1 (2018): 1-43.
[33] Bernsmed, Karin, Martin GilieJaatun, and Per HåkonMeland. "Safety Critical Software and Security-How Low Can You Go?." 2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC). IEEE, 2018.
[34] Pekaric, Irdin, Clemens Sauerwein, and Michael Felderer. "Applying Security Testing Techniques to Automotive Engineering." Proceedings of the 14th International Conference on Availability, Reliability and Security. 2019.
[35] Oyetoyan, Tosin Daniel, et al. "Myths and facts about static application security testing tools: an action research at Telenor digital." International Conference on Agile Software Development. Springer, Cham, 2018.
[36] Liu, Qiang, et al. "A survey on security threats and defensive techniques of machine learning: A data driven view." IEEE access 6 (2018): 12103-12117.
[37] Zeebaree, Subhi RM, et al. "Security Approaches For Integrated Enterprise Systems Performance: A Review." International Journal of Scientific & Technology Research (IJSTR) 8.12 (2019): 2485-2489.
[38] Bitton, Ron, et al. "Deriving a cost-effective digital twin of an ICS to facilitate security evaluation." European Symposium on Research in Computer Security. Springer, Cham, 2018.
[39] Wysopal, Christopher J., Christopher J. Eng, and Matthew P. Moynahan. "Assessment and analysis of software security flaws." U.S. Patent No. 10,275,600. 30 Apr. 2019.
[40] Alhawi, Omar MK, Alex Akinbi, and Ali Dehghantanha. "Evaluation and application of two fuzzing approaches for security testing of IoT applications." Handbook of Big Data and IoT Security. Springer, Cham, 2019. 301-327.
[41] Yuce, Bilgiday, Patrick Schaumont, and Marc Witteman. "Fault attacks on secure embedded software: Threats, design, and evaluation." Journal of Hardware and Systems Security 2.2 (2018): 111-130.

[42] Alenezi, Mamdouh, and SadiqAlmuairfi. "Security risks in the software development lifecycle." International Journal of Recent Technology and Engineering (IJRTE) 8 (2019): 13.
[43] Chen, Chung-Kuan, et al. "Penetration testing in the iot age." Computer 51.4 (2018): 82-85.
[44] Williams, Laurie, Gary McGraw, and Sammy Migues. "Engineering security vulnerability prevention, detection, and response." IEEE Software 35.5 (2018): 76-80.
[45] Gorodissky, Boaz, Adi Ashkenazy, and Ronen Segal. "Penetration Testing of a Networked System." U.S. Patent Application No. 15/874,429..
[46] Munaiah, Nuthan, et al. "Characterizing Attacker Behavior in a Cybersecurity Penetration Testing Competition." 2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). IEEE, 2019.
[47] Zukri, NurulHidayah Ahmad, et al. "Agent-Based Encryption for Password Management Application." Charting the Sustainable Future of ASEAN in Science and Technology. Springer, Singapore, 2020. 529-541.
[48] Jat, Subhash Chandra, C. S. Lamba, and Vijay Singh Rathore. "Software quality improvement through penetration testing." Emerging Trends in Expert Applications and Security. Springer, Singapore, 2019. 239-244.
[49] Ghanem, Mohamed C., and Thomas M. Chen. "Reinforcement learning for efficient network penetration testing." Information 11.1 (2020): 6.
[50] Zennaro, Fabio Massimo, and Laszlo Erdodi. "Modeling Penetration Testing with Reinforcement Learning Using Capture-the-Flag Challenges and Tabular Q-Learning." arXiv preprint arXiv:2005.12632 (2020).
[51] Zech, Philipp, Michael Felderer, and Ruth Breu. "Knowledge-based security testing of web applications by logic programming." International Journal on Software Tools for Technology Transfer 21.2 (2019): 221-246
[52] Ye, Yu, et al. "High-risk Problem of Penetration Testing of Power Grid Rainstorm Disaster Artificial Intelligence Prediction System and Its Countermeasures." 2019 IEEE 3rd Conference on Energy Internet and Energy System Integration (EI2). IEEE, 2019.
[53] Alhassan, J. K., et al. "A fuzzy classifier-based penetration testing for web applications." International Conference on Information Theoretic Security. Springer, Cham, 2018.
[54] Tetskyi, Artem, VyacheslavKharchenko, and DmytroUzun. "Neural networks based choice of tools for penetration testing of web applications." 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT). IEEE, 2018.
[55] Khera, Yugansh, Deepansh Kumar, and NidhiGarg."Analysis and Impact of Vulnerability Assessment and Penetration Testing." 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon).IEEE, 2019.
[56] Lee, Seungsoo, et al. "A comprehensive security assessment framework for software-defined networks." Computers & Security 91 (2020): 101720.
[57] Ali, Bako, and Ali Ismail Awad. "Cyber and physical security vulnerability assessment for IoT-based smart homes." sensors 18.3 (2018): 817.
[58] Fu, Rong, et al. "Security assessment for cyber physical distribution power system under intrusion attacks." IEEE Access 7 (2018): 75615-75628.
[59] Sahay, Rashmi, G. Geethakumari, and KoushikModugu. "Attack graph—Based vulnerability assessment of rank property in RPL-6LOWPAN in IoT." 2018 IEEE 4th World Forum on Internet of Things (WF-IoT). IEEE, 2018.

[60] Abu-Dabaseh, Farah, and EsraaAlshammari. "Automated Penetration Testing: An Overview." Computer Science & Information Technology (2018).
[61] Peroli, Michele, et al. "MobSTer: A model□based security testing framework for web applications." Software Testing, Verification and Reliability 28.8 (2018): e1685.
[62] Zitta, Tomas, et al. "Penetration testing of intrusion detection and prevention system in low-performance embedded IoT device." 2018 18th International Conference on Mechatronics-Mechatronika (ME).IEEE, 2018.
[63] Chaudhary, Sujita, Austin O'Brien, and ShengjieXu."Automated post-breach penetration testing through reinforcement learning." 2020 IEEE Conference on Communications and Network Security (CNS).IEEE, 2020.
[64] Chu, Ge, and Alexei Lisitsa. "Poster: Agent-based (BDI) modeling for automation of penetration testing." 2018 16th Annual Conference on Privacy, Security and Trust (PST).IEEE, 2018.
[65] O'Mahony, George D., Philip J. Harris, and Colin C. Murphy. "Analyzing using Software Defined Radios as Wireless Sensor Network Inspection and Testing Devices: An Internet of Things Penetration Testing Perspective." 2020 Global Internet of Things Summit (GIoTS). IEEE, 2020.
[66] Parizi, Reza M., et al. "Empirical vulnerability analysis of automated smart contracts security testing on blockchains." arXiv preprint arXiv:1809.02702 (2018).
[67] Yang, Aimin, et al. "Security control redundancy allocation technology and security keys based on Internet of Things." Ieee Access 6 (2018): 50187-50196.
[68] Plappert, Matthias, et al. "Multi-goal reinforcement learning: Challenging robotics environments and request for research." arXiv preprint arXiv:1802.09464 (2018).
[69] Neal, Christopher, et al. "Reinforcement Learning Based Penetration Testing of a Microgrid Control Algorithm." arXiv preprint arXiv:2008.13212 (2020).
[70] Pozdniakov, K., et al. "Smart Computer Security Audit: Reinforcement Learning with a Deep Neural Network Approximator." (2020).

