International Journal of Pure and Applied Mathematics
Volume 117 No. 15 2017, 325-339
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
Special Issue

Security Validation on Business Process Using Two Way Deterministic Finite Automata

[1]M. Thirumaran, [2]R. Padmanaban, [3]P. Anitha and [4]N.Sivakumar

[1]CSE, Pondicherry Engineering College,

Puducherry.

thirumaran@pec.edu

[2]CSE, Pondicherry Engineering College,

Puducherry.

srpmails@pec.edu

[3]CSE, Pondicherry Engineering College,

Puducherry.

anithabtech12@pec.edu

[4]CSE, Pondicherry Engineering College,

Puducherry.

sivakumar11@pec.edu

Abstract

In this generation of automation, mostly all the customer related services and processes are made online through Web Services and Web Applications designed by several organizations. But, nowadays certain organizations are unsecured due to thenew vulnerabilities that are raised in the web applications because of the flaw in the security logic implemented in the business process of the web service. The security holes that are prevailing in the web applications, paves a way for many flaws in the service that results in credit card misuse, personal identification theft, etc., These attacks tends to financial and reputational damage for many enterprises. Theattackers even compromise the browsing machines where the visited web sites get altered. The main reason for this type of security flaw is the lack of input validation in the service logic of the business process. This imperfect security clearance leads all the major vulnerabilities that occur in the web applications. A finite control is demanded for validating the input as per the security requirements. Hence, this paper describes a security validation model which uses "Two- Way Deterministic Finite Automata" for security assessment and "Finite State Machine" for managing the logical flow of the system by analyzing the state transition of the web application based on the crossing sequence techniques.

KeyWords: Web application, Web Services, Finite State Machine, Two-way Deterministic Finite Automata, Input validation, Security Assessment, Flow Control, and Crossing Sequence.

1. Introduction

In today's world, the percentage of people using internet connection is veryhigh. Evolution of web technology gives web developers the ability to create innovative and interactive web services. Since there is advancement in the cybercrimes done by experienced hackers, business organizations struggle a lot for maintaining the security of their data online. Thus, the task of securing the web applications is mandatory now which is done by enhancing the security checkpoints and the techniques at the developing phase itself. In order to safeguard the content alteration by the attackers, the web applications and web sites are to be monitored and verified at each and every stage of development.

Considering business environment, a cascaded and composed set of process is required to fulfill the complete business process. In web application, the execution order of the web pages differs according to the user convenience i.e. same operation is done for many times, same cycle of operations are repeated, navigating the web pages forward and backward. Each web application contains numerous logical web applications with different controls and functionalities. While functioning, crossing sequence for the transition is measured. The crossing sequence is based on the inputs that are encountered in the web applications along with the security requirements.

The crossing sequence of the transition of a web service needs to be maintained for all the web applications. Based on this, the satisfaction of Business Logic associated with the security requirements is verified. Using the differences in the crossing sequence, security inconvenience faced by the users can be identified easily. We can also find whether the core functionality of the web application is successfully completed or not by the variations in the crossing sequence.

In Two-Way Deterministic Finite Automata, security is maintained in all possible sequence of process using crossing sequence methodology. Business Process Execution Language (BPEL) is used for specifying the business logic and security logic actions. Whenever business logic is found by the service, the control moves towards right to the successive service. If business logic is found to have security logic embedded in its service, then the control moves towards left direction. Until the next security element is encountered, the same process will be continued till the transition sequence is completed. Thus, for any set of input on a service incorporated with the security logics, security validation will be done before execution of the service to avoid the flaw in the business process.

2. Related Works

The works related to the focus of the research are discussed in this section. M. Thirumaran et all evaluated the Business Process Enforcement in Long term

Composed Services (LCS) by Policy Mapping mechanism using Finite State Machine Model and also obtained the order of execution using the state transition [1]. M. Thirumaran et all measured the reliability of the web services from the functional work flow process and by extending the Finite State Machine for the composed set of services, quality parameters have been measured [2]. M. Thirumaran et all determines the dependency relation exists on the business rules, function and parameters by studying the logic flow of the through Finite State Machine exploitation and measured the quality of the integrated service through the property evaluator [3].

Anneliese A. Andrews et all combined test generation based on the Finite State Machine with certain constraints to reduce the state space explosion using system level testing technique and executed with an example of web-based course student information system [4]. In paper [5], Finite State Machine is used to implement fast and efficient access control mechanism in which state of FSMsuppliesaccess point through that the transaction between state to state happensby assigning the access controlvalue true or false. M. Thirumaran et allextracted a specific logic instead of extracting the overall function in Petri nets and evaluated the changes that have been implemented using certain change factors and modeled an efficient change management [6].

M. Thirumaran et all proposed a methodology for the changes that happens in web services using Finite State Machine which primarily uses business logic and generates dynamic source code and evaluated the architecture using Software Architecture Analysis Method (SAAM) [7].

Joe M. Tekli et all aimed at SOAP Performance and Enhancement by covering almost every phase of SOAP processing, serialization, de-serialization, security evaluation, etc., using java RMI technologies [8].

In paper [9], G. Naga VenkataKiran et all simulated the business logic set usingFinite State Machine (FSM which includes the business policy constraints with its associated mapping function to ensure the changes that are made during run time and also facilitates Business Analyst to dynamically control and manage the business logic of the targeting web services.

Dianxiang Xu et all presented an automated security testing approach using formal threat models and also facilitates the traceability of design level function by mapping the threat models from the individual actions and specified design level security threats [10].

M. Thirumaran et all constructed an Finite State Machine by the Business Logic model in order to state the flow of logic and also allows the enterprises to make use of the automated working of the system without the intervention of the developer at any stage and this in turn reduce the threshold of development and operation of the web services in the IT enterprises in modern service industry[11].

Tom Kirkham et all illustrates the threats that occurs in online and how the threats aimed directly at individuals and also highlights some important tips to get rid of those online issues [12].

M. Thirumaran et all accomplished web service change management in efficient manner so that it increases business growth rate and they done business analysis distinctively on their own instead of depending on the programmers and the model predicted change factors such as order of execution, schema validation, time and space complexity [13].

M. Thirumaran et all proposed a novel dependency analysis approach for the evaluation of the changes that are specified by the business analysts in an effective manner by making use of FSM based model for exhibiting the dynamic nature of the web services [14].

Hongxin Hu et all discovered a policy-based segmentation technique in order to identify policy anomalies and to derive anomaly resolutions along with an intuitive visualization representation of analysis results in an accurate manner and also explored that this mechanism can be applied to other existing access control policy languages [15].

M. Thirumaran et all presented a framework for managing and implementing the changes that are based on Finite State Machine where it is used for creating the set of business logic which also includes the policy constraints with its allied mapping function to guarantee the changes made during run time [16]. Marco Leogrande et all used packet filters in the state automata and guarantees the optimal number of checks on the packets even in multiple filters composition cases [17].

Bipin Upadhyaya proposed an approach that identifies and aggregates QoE attributes for the web services that shows significant precision and recall on identification and grouping of those attributes that are applicable to any other domain [18].

Xiaolin Zhao et all used Software behavior modeling in Finite State Machine that mainly uses data values and interaction traces between software components for their processing and the performance is evaluated [19]. Felipe Gomes Cabral et all proposed a model for discrete event systems for diagnosing online which makes use of Finite Automata [20].

Yongbao Liu et all represented the fault state in the finite state automata in order to describe the process of fault detection in the web services [21]. Matthew L.

Aldridge et all conducted and researched to explore the finite state machine's performance on a Palm mobile computing device platforms [22].

3. Proposed System

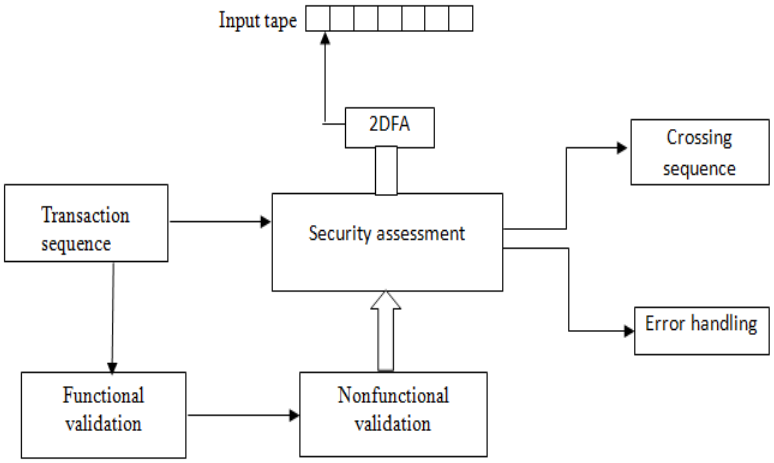

Figure 3.1: Proposed Architecture of Security Assessment using Two Way Deterministic Finite Automata

The above Fig. 3.1 shows the proposed architecture of security assessment using Two way Deterministic Finite Automata. We use Two way Deterministic Finite Automata (2DFA) to revisit the character that was already processed. In 2DFA, based on the current character or action, finite number of states with transition are provided between them. Each transition is characterized with a value representing whether the machine will move to its corresponding location (moving left or right or staying at the same position) based on the security requirements. This transition sequence is provided as the input to the system for security assessment and then they are further moved to the 2DFA input tape and accordingly the crossing sequence will be produced. The transition sequence which is the input provided to the system is the flow of function that is carried out in the system. According to the transition, functional and non-functional validation is done. In functional validation, Logical design gets verified whereas in the remaining gets verified using non-functional validation. After this validation, the input once again enters the security assessment phase. In this security assessment phase, the input gets entered into the input tape and the transition gets divided into BPEL actions such as BPEL without security specifications and BPEL with security specifications (Security Language).

When the input tape encounters the BPEL actions without the security specifications, then the control moves right and if the tape encounters BPEL actions along with the security specifications, then the control moves towards left. Two way Deterministic FiniteAutomata generate crossing sequence for each execution process. In our proposed system, when BPEL language is found, then the control moves towards right until it discovers security language.When

it finds any security specifications, then the control moves towards left until it discovers normal BPEL actions. The process continues till the transition sequence finishes.So, there is no chance of missing any security validation in any set of transition sequence even in multiple set of transition sequence. To detect and communicate with the error that occurs in case, Error Handling is used.

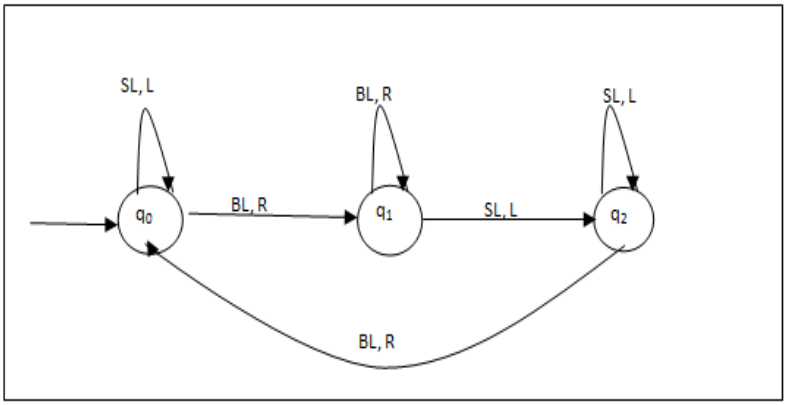

Figure 3.2: FSM Diagram for Two way Deterministic Finite Automata

The above Fig 3.2 shows the Finite State Machine for the Two way Deterministic Finite Automata. It consumes input in the SL, BL and transits to the corresponding direction according to the input. BL refers to the BHEL Logic without security specification and SL refers to the Security Logic i.e. BPEL actions along with the security mechanism.On consuming the input (SL, L), the initial state q0 goes Left and on consuming the input (BL, R), q0 state goes to the next state by moving right side and goes on till the transition sequence ends.

Table 3.1: State Transition Table

State	**Input**	
	0	**1**
q_0	$q_{1,R}$	$q_{0,L}$
q_1	$q_{1,R}$	$q_{2,L}$
q_2	$q_{0,R}$	$q_{2,L}$

The transition of state according to the input is shown in the Table 3.1 where input 0 represents that it does not contain security specification and input 1 represents that it contains the security specification. Also clearly represents that each input without security specifications are directed towards right ($q_{n,}$ R) and the input with security specifications are directed towards left (q_n, L).Fig 3.3 indicates the Design of FSM for the online shopping portalusing Two way Deterministic Finite Automata. It contains states form q_0 to q_{13} and the transition is made between them for viewing, ordering and purchasing products in secured

manner.Some process does not need security elements whereas some operations need security bound with that. In this, it is clearly shown that the login process needs security essentials and therefore for validation it goes Left. For browsing the items in the online portal, integrity needs to be achieved and so it moves towards left. For security assessment such as Third part verification, SSL connection, OTP verification, etc., the control moves towards left. And for all the other process, the control moves towards right.Likewise, direction of each and every transition state belongs to the presence of security specifications.

The common attack that arises in the web application under business process is SQL injection[23]. This attack happens by injection illegal SQL queries into the JavaScript code and gain access to the web application. The problem that happens while OTP (One Time Password) generation such as OTP failure, consumption of time to generate the secret code make the online system vague and this in turn changes the mind set of people over security of the system due to the inconvenience. So that, crossing sequence needs to be maintained. In our proposed system, for each users and for each time when they make use of the application, the system will be reliable and trustworthy.

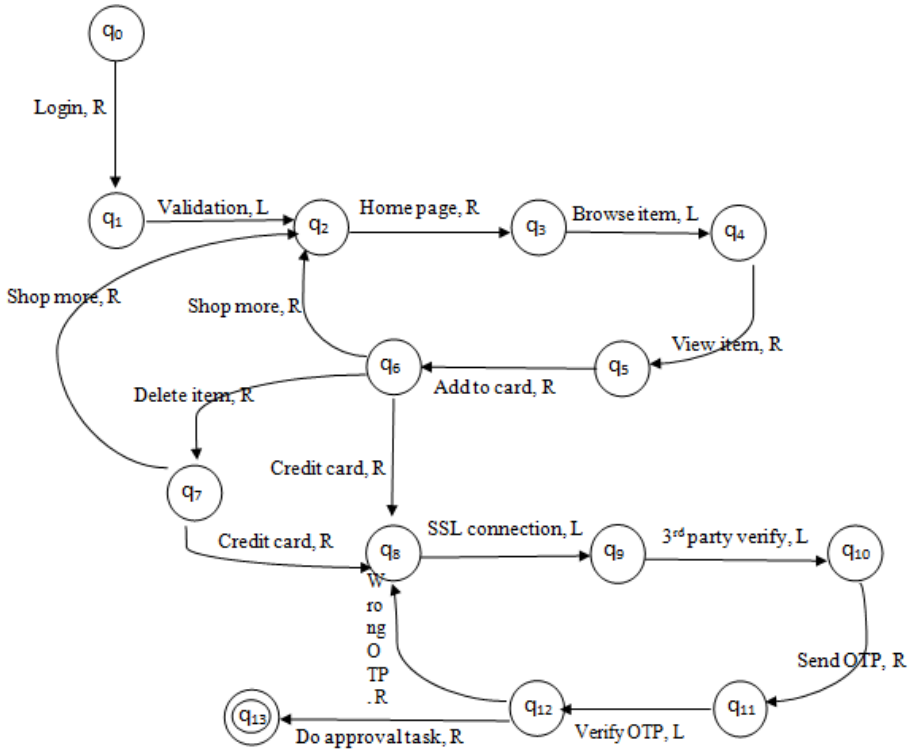

Figure 3.3: FSM Diagram for Online Shopping using Two Way Deterministic Finite Automata

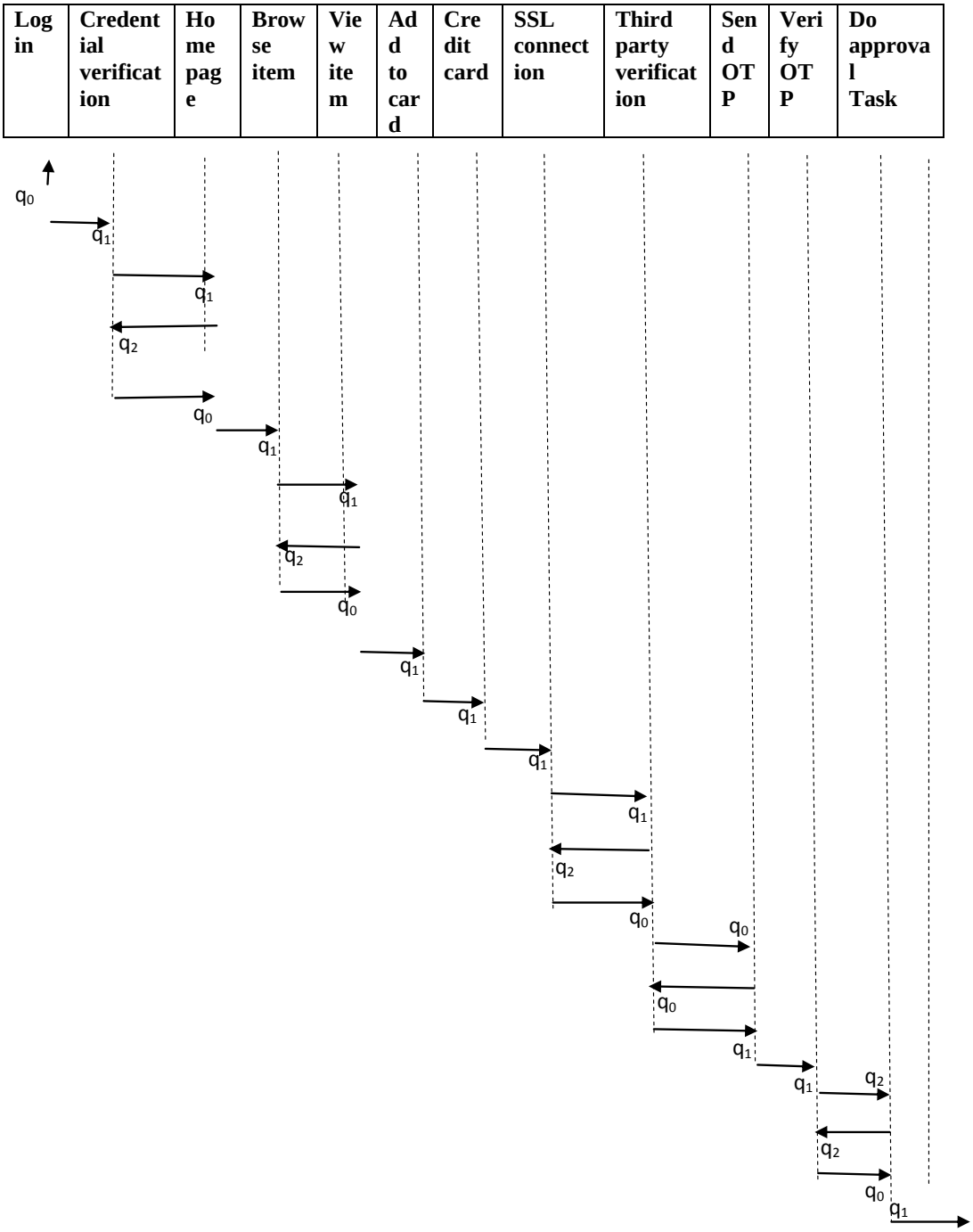

Figure 3.4: Crossing Sequence of Two way Deterministic Finite Automata

The above Fig. 3.4 shows the Crossing sequence of Two way Deterministic Finite Automata. It shows the flow of function and for what input the control moves either in left or right direction.From the crossing sequence, we can easily find whether it reached end state or not. So that, missing of security validation never happens. From this we can solve the consumers inconvenience problem over security and performanceof the system.

4. Experimental Result

Table 4.1: Input States and the Corresponding Crossing Sequence for Online Shopping using 2DFA

S no	Input sequence	Total no of state	No of node passes BL input	No of node passes SL input	No of complete crossing sequence	No of deviated crossing sequence	No of crossing sequence not reached end state	No of crossing sequence reached end state
1	Login → credential verification → home page→ browse-item→ view-item → add to card → credit card → SSL connection establish→ third party verification→verify credentials → generate OTP → send OTP → check OTP → do approval task	13	9	5	12	0	0	12
2	Login → credentials verification →home page→ browse-item → view-item → add to card → credit card →SSL connection establish → third party verification→verify credentials→ generate OTP→ send OTP → check OTP → delay to submit OTP → session expired	12	7	5	11	0	1	10
3	Login →credentials verification→ home page→ browse-item→ view-item → add to card → credit card → SSL connection establish→ third	10	6	4	8	1	4	8

	party verification→ verify-credentials→ wrong input							
4	Login →credentials verification→ home page → browse-item→ view-item → add to card → Shop more → home page → browse-item→ view-item → add to card → credit card → SSL connection establish→ third party verification→verify credentials → generate OTP → send OTP → check OTP → do approval task	12	9	5	17	1	0	17
5	Login →credentials verification→ home page → browse-item→ view-item → add to card → Shop more → home page → browse-item→ view-item → add to card → Delete item → credit card → SSL connection establish→ third party verification→verify credentials → generate OTP → send OTP → check OTP → do approval task	13	10	5	18	2	0	18
6	Login → credentials verification→ home page → browse-item→ view-item → add to card → Delete item → Shop more → home page → browse-item→ view-item → logout	9	8	2	12	2	6	12
7	Login → credentials verification→ home page → browse-item→ view-item → add to card → Delete item → logout	8	5	2	6	1	6	7

8	Login →credentials verification→ home page → browse-item→ view-item → add to card → Shop more → home page → browse-item→ view-item → add to card → Delete item →log out	8	6	2	12	2	6	7
9	Login → credentials verification →home page → browse-item → view-item → add to card → credit card →SSL connection establish → third party verification→verify credentials→ generate OTP→ send OTP → check OTP → delay to submit OTP → wrong OTP → logout	12	7	5	11	1	1	11
10	Login → credentials verification → wrong input	2	1	1	1	0	11	1

The experimental results of the Two way Deterministic Finite Automata for the Online shopping is listed with the series of input sequence and their corresponding Crossing sequence in Table 4.1. From this Crossing sequence, we can find easily how many of it reached the end state according to their input. Also whether security is compromised or not can be found easily and this makes our system reliable and trustworthy.

5. Evaluation of Security Validation Through Crossing Sequence

We implemented the security validation on business process for different web services and calculated the crossingsequence occurring in each service. The different domain and their respective sequences are tabulated below:

Table 5.1: Evaluation of Security Validation through Crossing Sequence on Various Enterprises

S.NO	BUSINESS DOMAIN	CROSSING SEQUENCE DETECTED		
		NO.OF SEQUENCE	COMPLETED SEQ	FAILED ON SECURITY REQUIREMENTS
1	Shopping	55	46	9
2	Banking	45	35	10
3	Insurance	40	35	5
4	Reservation	50	44	6

6. Conclusion

In this paper, we have described the actual need of the security validation in the business process of Web Applications. We usedTwo-way deterministic Finite Automata and FSM for security validation in all type of transition sequence.The services encountered in the application are checked in forward sequence if there is no security embedded in the services. Else, the transition is moved backward and forward according to the security logic of the process. TheFSM keeps track of this logicalflow of the process sand provides the necessary transition sequence.

The security validation for services embedded with security logic is carried out for the generated transition sequence using Crossing sequence methodology which checks whether the necessary security services are encountered and fulfilled by the customer.

This also points out the security inconvenience in the service incase of incomplete transactions and services. This sequence varies for different customers according to their needs. Thus,security validation for business process is provided using this model wherein each stage of the process is secured and is validated without any exception as a whole.

References

[1] Thirumaran M., Jannani M., A Finite State Machine based Evaluation of Business Policy Enforcement in Long Term Composed Services, International Journal of Innovative Research in Computer and Communication Engineering, (2014).

[2] Thirumaran M., Dhavachelvan P., Abarna S., Lakshmi P., Finite State Machine Based Evaluation Model for Web Service Reliability Analysis, International Journal of Web & Semantic Technology (IJWesT), (2011).

[3] Thirumaran M., Dhavachelvan P., Aranganayagi G., Evaluating Service Business Logic using Finite State Machine for Dynamic Service Integration, International Journal of Computer Applications, (2011).

[4] Anneliese A.A., Jeff O.,Roger T.A., Testing Web Applications by Modeling with FSMs, Springer, (2005).

[5] Thirumaran M., Dhavachelvan P., Aishwarya D., Shanmugapriya R., Finite State Machine Based Access Control Mechanism for Web Service Work Flow Management, International conference on electronic engineering and computer science, (2013).

[6] Thirumaran M., Dhavachelvan P., Aishwarya D., Rajkumarid K., Conventional Usage of Finite State Machine over Petri net in

Web Service Change Management Framework, International Conference on Agricultural and Natural Resource Engineering, (2013).

[7] Thirumaran M., Dhavachelvan P., Naga V.G., Finite State Machine Based Business Logic Model for Web Services Change Management, Advances in Computing and Communication (ICACC), (2012).

[8] Joe M.T., Ernesto Damiani, Richard Chbeir, Gabriele Gianini, SOAP Processing Performance and Enhancing, IEEE Transactions on Services Computing, (2012).

[9] Naga VenkataKiran G., Thirumaran M., Dhavachelvan P., A Policy Driven Business Logic Change Management For Enterprise Web Services, springer, (2012).

[10] Dianxiang Xu, Manghui Tu, Michael Sanford, Lijo Thomas, Daniel Woodraska, Weifeng Xu, Automated Security Test Generation with Formal Threat Models, IEEE Transactions on Dependable and Secure Computing, (2012).

[11] Thirumaran M., Dhavachelvan Ponnurangam, Naga VenkataKiran G., Divya A., A Novel Framework for Enterprise Web Services Change Management, International Conference on Advances in Computing, Communications and Informatics, (2012).

[12] Tom Kirkham, Sandra Winfield, Serge Ravet and Sampo Kellomäki, The Personal Data Store Approach to Personal Data Security, IEEE Security and Privacy, (2013).

[13] Thirumaran M., Dhavachelvan P., Aishwarya D., Kiran Kumar ready, Evaluation of Change Factors for Web Service Change Management, International Conference on Communication, Computing & Security, (2012).

[14] Thirumaran M., Dhavachelvan P., Naga venkataKiran G., A Collaborative Framework for Evaluation of Run-Time Changes in Enterprise Web Services, International Journal of Web & Semantic Technology, (2012).

[15] Hongxin Hu, Gail-Joon Ahn, Ketan Kulkarni, Discovery and Resolution of Anomalies in Web Access Control Policies, IEEE Transactions on Dependable and Secure Computing, (2013).

[16] NagaVenkataKiran G., Thirumaran M., Dhavachelvan P., Business Logic Model for Web Services Change Management, International Journal of Computer Applications, (2010).

[17] Marco Leogrande, FulvioRisso, Luigi Ciminiera, Modeling Complex Packet Filters with Finite State Automata, IEEE/ACM Transactions on Networking, (2015).

[18] Matthew L.A., Michael W.B., Performance of a Finite-State Machine Implementation of Iterative Cluster Labeling on Desktop and Mobile Computing Platforms, IEEE Transactions on Knowledge and Data Engineering, (2009).

[19] Vishnu, S., Vignesh, S., Surendar, A.”Design and implementation of ZETA micro-inverter for solar PV application”(2017), International Journal of Mechanical and Production Engineering Research and Development, 7 (4), pp. 215-222.

[20] Lakshmi, K., Surendar, A. ”Verification of axiprotocol using system Verilog”, (2017), International Journal of Mechanical Engineering and Technology, 8 (5), pp. 588-595.

[21] Surendar, A., Kavitha, M. “Secure patient data transmission in sensor networks”, (2017), Journal of Pharmaceutical Sciences and Research, 9 (2), pp. 230-232.

[22] Surendar, A.”FPGA based parallel computation techniques for bioinformatics applications”,(2017) International Journal of Research in Pharmaceutical Sciences, 8 (2), pp. 124-128.

[23] Bipin Upadhyaya, Ying Zou, Iman Keivanloo, Joanna Ng, Quality of Experience: User’s Perception about Web Services, IEEE Transactions on Secure Computing, (2015).

[24] Xiaolin Zhao, JingfengXue, Changzhen Hu, Rui Ma, Shanshan Zhang, Research on Software Behavior Modeling Based On Extended Finite State Automata, Conference on Communication security, (2014).

[25] Felipe Gomes Cabral, Marcos Vicente Moreira, OumarDiene, João Carlos Basilio, A Petri Net Diagnoser for Discrete Event Systems Modeled by Finite State Automata, IEEE Transactions on Automatic Control, (2015).

[26] Yongbao Liu, LiangliMa, Shuhong Huang, A Fault Detection and Isolation Model Based on Conditional Finite State Machine for Gas Turbine, International Conference on natural computation, (2009).

[27] RAJESH, M. "A SYSTEMATIC REVIEW OF CLOUD SECURITY CHALLENGES IN HIGHER EDUCATION." The Online Journal of Distance Education and e- Learning 5.4 (2017): 1.

[28] Rajesh, M., and J. M. Gnanasekar. "Consistently neighbor detection for MANET." Communication and Electronics Systems (ICCES), International Conference on. IEEE, 2016.