ABSTRACT

Authorization is an important security concern in cloud computing environments. It aims at regulating an access of the users to system resources. A large number of resources associated with REST APIs typical in cloud makes an implementation of security requirements challenging and error-prone. To alleviate this problem, in this paper we propose an implementation of security cloud monitor. We rely on model-driven approach to represent the functional and security requirements. Models are then used to generate cloud monitors. The cloud monitors contain contracts used to automatically verify the implementation. We use Django web framework to implement cloud monitor and OpenStack to validate our implementation

SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

LIST OF FIGURES

Figure No.	Figure Title	Page No.
4.1.1	System Architecture	8
4.1.2	Data Flow Diagram	9
4.1.3	Use Case Diagram	12
4.1.4	Class Diagram	14
4.1.5	Sequence Diagram	15
4.1.6	Activity Diagram	16
6.1	User registration form	50
6.2	User login form	51
6.3	Admin approves User	52
6.4	User uploads files	52
6.5	User app is approved by cloud	53
6.6	User edits the file	53
6.7	User can delete the file	54
6.8	User can download the file	54
6.9	User upload files on uploaded file	55

ST. MARTIN'S ENGINEERING COLLEGE TECHNOLOGY FOR PROSPERITY UGC AUTONOMOUS

LIST OF ACRONYMS AND DEFINITIONS

S.No.	Acronym	Definition
1.	REST	Representational State Transfer
2.	API	Application Programming Interface
3.	URI	Uniform Resource Identifier
4.	OCL	Object Constraint Language
5.	DBC	Design By Contract
6.	IAAS	Infrastructure As A Service
7.	XACML	Extended Access Control Markup Language
8.	RBAC	Role Based Access Control
9.	UML	Unified Modelling Language

ST.MARTIN'S ENGINEERING COLLEGE
TECHNOLOGY FOR PROSPERITY
SMEC
UGC AUTONOMOUS

CONTENTS

ACKNOWLEDGEMENT i

ABSTRACT ii

LIST OF FIGURES iii

LIST OF ACRONYMS AND DEFINITIONS iv

CHAPTER 1 INTRODUCTION 01

1.1 Objective 02

1.2 Overview 03

CHAPTER 2 LITERATURE SURVEY 04

CHAPTER 3 SYSTEM ANALYSIS AND DESIGN 07

3.1 Existing System 07

3.2 Proposed System 07

CHAPTER 4 SYSTEM REQUIREMENTS & SPECIFICATIONS 08

4.1 Design 08

4.1.1 System Architecture 08

4.1.2 Data Flow Diagram 09

4.1.3 Use Case Diagram 12

4.1.4 Class Diagram 14

4.1.5 Sequence Diagram 15

4.1.6 Activity Diagram 16

4.2 Modules 17

4.2.1 Modules Description 17

4.3 System Requirements 18

4.3.1 Hardware Requirements 18

4.3.2 Software Requirements 18

4.4 Testing 19

4.4.1 Unit testing 19

4.4.2 Integration testing 19

4.4.3 Functional testing 19

4.4.4 System testing 20

4.4.5 White box testing 20

4.4.6 Black box testing 20

4.4.7 Unit testing 20

4.4.8 Integration testing 21

4.4.9 Acceptance testing 21

CHAPTER 5 SOURCE CODE 22

CHAPTER 6 EXPERIMENTAL RESULTS 49

CHAPTER 7 CONCLUSION & FUTURE ENHANCEMENT 55

7.1 CONCLUSION 55

7.2 FUTURE ENHANCEMENT 55

CHAPTER 8 REFERENCES 56