CONTINUOUS IMPLEMENTATION THROUGH STANDARDIZED AND COMPLIANT INFRASTRUCTURE AS CODE

S. A. Udara Jayawardana

199379X

Thesis submitted in partial fulfillment of the requirements for the degree Master of Science in Computer Science & Engineering

Department of Computer Science & Engineering

University of Moratuwa

Sri Lanka

May 2021

DECLARATION

I declare that this is my own work and this dissertation does not incorporate without acknowledgment any material previously submitted for a degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Also, I hereby grant to the University of Moratuwa the non - exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic, or another medium. I retain the right to use this content in whole or part in future works.

Signature: ..... Date: .....

Name: S.A.U. Jayawardana

I certify that the declaration above by the candidate is true to the best of my knowledge and that this report is acceptable for evaluation for the In19-S3-CS5999 PG Diploma Project.

UOM Verified Signature

Signature: ..... Date: .....

Name: Prof. Gihan Dias

ACKNOWLEDGEMENT

I am extremely grateful to my supervisor, Prof.Gihan Dias, Department of Computer Science and Engineering, University of Moratuwa, for his constant guidance, which played a major part in successfully completing the research conducted and successful thesis. I would like to further extend my gratitude to my examiners Dr.Sunimal Rathnayake and Dr.Lochandaka Ranathunga for their invaluable insights and feedback.

I am fortunate to be a part of the DevOps and Site Reliability Engineering community, which took some valuable time out of their busy schedules to fill out the survey and share their current Infrastructure as Code related practices, which helped in identifying real-world difficulties and roadblocks related to IaC best practices.

Very special thanks to my family's unconditional support and continuous encouragement

ABSTRACT

With cloud computing becoming the new norm and organizations embracing cloud services benefits, the infrastructure provisioning methods of the old are quickly becoming obsolete. Infrastructure as Code concept introduced as an answer for this with provisioning infrastructure in an automated manner, with specifications defined on a machine-readable code. IaC made dynamic provisioning and modification on cloud resources possible, enabling organizations to utilize the full benefits of the cloud.

However, IaC without proper standardization and compliance could result in disastrous outcomes. In order to achieve this, the industry looked into the Software Engineering practices, due to IaC's similarities to coding. Though it may look similar, this proved to be less effective. Therefore, testing, compliance, and standardization methods, specifically tailored for IaC are required. A standardized and compliant IaC will make way to implement *Continuous Implementation*.

Immutable infrastructure is a major roadblock that can inhibit harnessing the benefits of cloud ecosystems. Though many organizations use Continuous Integration (CI) and Continuous Deployments (CD) for code deployments, the infrastructure & configurations mostly remain unchanged. However, infrastructure should follow the same principle of frequent updates, to get the best out of ever-changing cloud infrastructure.

This research focuses on introducing the concept of Continuous Implementation. Continuous Implementation pipelines will be evaluated with the traditional and currently widely-used infrastructure provisioning methods. A standardized IaC framework will be used to support fully automated infrastructure provisioning, modification, and configuration management, on imposing the organizational and security policies. Through the results obtained, a study was conducted on determining the importance of Continuous Implementation for cloud-based infrastructure.

TABLE OF CONTENTS

DECLARATION i

ABSTRACT ii

List of Figures v

List of Tables vi

List of Abbreviations vii

1. INTRODUCTION 1

1.1 Background 1

1.2 Research Problem 4

1.3 Motivation 7

1.4 The Purpose of the Research 9

1.5 The Proposed Solution & the Objectives 10

2. LITERATURE REVIEW 13

2.1 Overview 13

2.2 Prior Research carried under IaC 15

2.2.1 Research on the continuous implementation of infrastructure 15

2.2.2 Research on utilizing traditional frameworks for iac 16

2.2.3 Research on iac testing 18

2.2.4 Research on standardization & benchmarking cloud 21

2.3 Commercial and Open Source Tools 23

2.3.1 Acceptance and unit testing 23

2.3.2 Testing frameworks 25

2.3.3 Post-provision validation 29

2.3.4 Continuous integration 32

3. METHODOLOGY 33
3.1 Research Philosophy 33
3.2 Research Design 34
3.2.1 Infrastructure risks and violations 34
3.2.2 Identified requirement baseline 37
3.2.3 A Study on current IaC implementation & Standardization 40
3.2.4 Limitations 42
3.2.5 Architecture 44
3.3 Test Scenarios 47
3.3.1 Scenario 01 – Infrastructure drift 47
3.3.2 Scenario 02 – Code level violation 49
3.3.3 Scenario 03 – Organization compliance violation 51
3.3.4 Scenario 04 – Security violation 53
3.3.5 Scenario 05 – Custom quality gates 55
3.3.5 Scenario 06 – Successful implementation 57

4. Analysis and Findings 58
4.1 Survey Results Findings 58
4.2 Analysis 66

5. Conclusion 73

6. Appendix 76
6.1 Survey Questions 76

7. References 81

LIST OF FIGURES

1.Continuous Implementation – Proposed Solution 12
2.Interests in the IaC technologies in the last 10 years 14
3.Interests in the IaC technologies in by region 14
4.Hanappi Testing framework architecture 18
5.Ikeshita Testing framework architecture 20
6.Cloud Workbench architecture design 22
7.Architecture of AWS Lambda based post validation for S3 ACL violation 29
8.Cloud custodian architecture for multi-account AWS setup 30
9.Cloud custodian policy structuring on SCM 31
10.IaC standardization validation Model 40
11.Continuous Implementation Framework 46
12.Test 01: Infrastructure drift 48
13.Test 02: Gold modules were not used 50
14.Test 03: Mandatory tags were missing 52
15.Test 04: Security group with insecure rule 54
16.Test 05: A violation with different impact based on the environment 56
17.Test 06: Successful implementation 57
18. Survey – User designation 59
19. Survey – User experience on cloud computing 59
20. Survey – User experience on industry domains 59
21. Survey – Company's country of origin 60
22. Survey – Cloud service providers 60
23. Survey – Infrastructure provisioning in Cloud 61
24. Survey – IaC: Ease of use 61
25. Survey – IaC: Efficiency 62
26. Survey – IaC: Challenges 1 62
27. Survey – IaC: Challenges 2 63

28. Survey – Cloud resource update frequency 63
29. Survey – Cloud resource updating methods 64
30. Survey – IaC testing 65
31. Survey - IaC standardization tool 65

LIST OF TABLES

1.Technology changes in the Cloud Age	1
2.Design and operational changes in the Cloud Age	2
3.Commercial and Open-Source CI tools	32
4.CI solutions offered by CSPs	32
5. IaC defects analysis	66 - 71

LIST OF ABBREVIATIONS

Abbreviation	Description
IAC	Infrastructure as Code
DSL	Domain Specific Language
CI	Continuous Integration
CD	Continuous Deployment
SCM	Source Control Management
CSP	Cloud Service Providers
API	Application Programming Interface
K8s	Kubernetes
OASIS	Organization for the Advancement of Structured Information Standards
TOSCA	Topology and Orchestration Specification for Cloud Applications
GDPR	General Data Protection Regulation
HIPAA	Health Insurance Portability and Accountability Act
PCI DSS	Payment Card Industry Data Security Standard
SOC2	Service Organization Control 2
AWS	Amazon Web Services
SRE	Site Reliability Engineer
WAF	Web Application Firewall