

Nihar Ranjan Roy
Sudeep Tanwar
Usha Batra *Editors*

Cyber Security and Digital Forensics

Select Proceedings of the International
Conference, ReDCySec 2023

Lecture Notes in Networks and Systems

Volume 896

Series Editor

Janusz Kacprzyk , Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Advisory Editors

Fernando Gomide, Department of Computer Engineering and Automation—DCA, School of Electrical and Computer Engineering—FEEC, University of Campinas—UNICAMP, São Paulo, Brazil

Okyay Kaynak, Department of Electrical and Electronic Engineering, Bogazici University, Istanbul, Türkiye

Derong Liu, Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago, USA

Institute of Automation, Chinese Academy of Sciences, Beijing, China

Witold Pedrycz, Department of Electrical and Computer Engineering, University of Alberta, Alberta, Canada

Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Marios M. Polycarpou, Department of Electrical and Computer Engineering, KIOS Research Center for Intelligent Systems and Networks, University of Cyprus, Nicosia, Cyprus

Imre J. Rudas, Óbuda University, Budapest, Hungary

Jun Wang, Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong

The series “Lecture Notes in Networks and Systems” publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

Indexed by SCOPUS, INSPEC, WTI Frankfurt eG, zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

For proposals from Asia please contact Aninda Bose (aninda.bose@springer.com).

Nihar Ranjan Roy · Sudeep Tanwar · Usha Batra
Editors

Cyber Security and Digital Forensics

Select Proceedings of the International
Conference, ReDCySec 2023



Springer

Editors

Nihar Ranjan Roy  Center for Cyber Security and Cryptology
Sharda University
Greater Noida, Uttar Pradesh, India

Usha Batra
Department of Computer Science
Engineering
Shri Vishwakarma Skill University
Gurugram, Haryana, India

Sudeep Tanwar
Department of Computer Science
and Engineering
Institute of Technology
Nirma University
Ahmedabad, Gujarat, India

ISSN 2367-3370

ISSN 2367-3389 (electronic)

Lecture Notes in Networks and Systems

ISBN 978-981-99-9810-4

ISBN 978-981-99-9811-1 (eBook)

<https://doi.org/10.1007/978-981-99-9811-1>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2024

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Paper in this product is recyclable.

Preface

The increasing need and desire for interconnections in cyberspace have created unprecedented opportunities and significant challenges. The latest technological developments have accelerated the spread of digital networks and human dependency. With the proliferation of cyber threats and attacks, safeguarding our digital infrastructure has become paramount.

The International Conference on Recent Developments in Cyber Security (ReDCySec 2023) organized on June 16–17, 2023, at the Center for Cyber Security and Cryptology, Sharda University, in association with Springer, is our humble effort in this direction. ReDCySec 2023 was conceived as a platform for researchers, practitioners, and industry experts to converge and share their knowledge, insights, and experiences in combating cyber threats. The conference catalyzed collaboration, fostering a vibrant exchange of ideas and innovative solutions in cyber security.

This book, *Cyber Security and Digital Forensics: Select Proceedings of the International Conference, ReDCySec 2023*, is a compilation of the scholarly contributions presented at the conference. A total of 248 submissions were received, out of which, after a rigorous peer-review process following international standards, 51 papers were permitted for oral presentation during the conference with an acceptance rate of 20.56%. The selection was based on parameters like originality of idea, relevance to the conference theme, relation to the latest trends, and innovation. The contents of this book reflect the dynamic and ever-evolving nature of cyber security. The authors delve into critical areas such as network security, cryptography, data protection, digital forensics, cyber threat intelligence, blockchain, artificial intelligence, and more. Each paper offers valuable insights, research findings, and real-world case studies that shed light on the complexities of cyber security and its impact on various sectors.

We hope that the ideas and subsequent discussions presented at the conference will help the scientific community to aim toward a nationally and globally responsible society. We also hope young minds will derive inspiration from their elders and contribute toward developing sustainable solutions for the nation and the world. This book will serve as a valuable reference for newcomers and seasoned practitioners in the broad field of cyber security, fostering a deeper understanding of the challenges we

face and inspiring innovative approaches to address them. We hope that the insights gained from the contents of this book will inspire further research, collaboration, and dialogue among professionals in the pursuit of a safer and more secure digital landscape.

We express our sincere appreciation to the members of the technical program committee who helped us in the paper selection process. We also thank the authors for their contribution so that we could compile such a quality set of papers. We thank Springer for being our publication partner for this conference. Our special thank goes to Ms. Kamiya Khatter and Ms. Silky Abhay Sinha at Springer for their outstanding service and support.

Greater Noida, India
Ahmedabad, India
Gurugram, India

Nihar Ranjan Roy
Sudeep Tanwar
Usha Batra

Contents

A Low-Distortion Reversible Steganography Method that Conceals Data in Images with Minimal Distortion	1
Mohammad Gauhar Nayab, Aditya Pratap Singh, Ritik Sharma, and Gaurav Raj	
Credit Card Fraud Detection Using ML Techniques	15
Samiratou Bonkoungou, Nihar Ranjan Roy, Nomel Haymes Axel-Elie Junior Ako, and Alpna Mishra	
An Effective Model for Binary and Multi-classification Based on RFE and XGBoost Methods in Intrusion Detection System	25
Swikrati Dubey and Chetan Gupta	
ENCRYPTO: A Reliable and Efficient Mobile App for Password Management	39
Urmila Pilania, Manoj Kumar, Saurav Kumar Srivastava, Bhavika Dhingra, Lalit Adhana, and Riya Gaur	
A Survey on Path Key Establishment	51
Krishan Kumar and Priyanka Ahlawat	
Detection of Phishing Link Using Different Machine Learning Techniques	63
Ashim Chaudhary, K. C. Krishna, Md Shadik, and Dharm Raj	
Secure Horizons: Advanced Protection Mechanisms for Holographic Data Storage Systems	79
Aviral Srivastava, Viral Parmar, Nishtha Chaudhari, and Samir Patel	
Security Flaw in TCP/IP and Proposed Measures	93
Sourav Kumar Upadhyay and Prakash Kumar	
Static Analysis Approach of Malware Using Machine Learning	109
Aman Raj Pandey, Tushar Sharma, Subarna Basnet, and Sonia Setia	

Cyber-Attack Analysis Using Vulnerability Assessment and Penetration Testing	123
Vijayashree Budyal, A. V. Vaibhav, C. U. Akshay, Naveen Ishika, and Gaonkar Unnathi	
An Investigative Study on Security Aspects and Authentication Schemes for Internet of Vehicles	135
Preeti Dhankar, Bhargavi Singh, and Priya Sharma	
Amharic Language Hate Speech Detection Using Machine Learning ...	149
Abirham Ayenew and Uttam Chauhan	
Cloud-Based Object Detection Model Using Amazon Rekognition	165
Darshita Singh and Deepak Arora	
Ensemble Deep Model for Hate Speech Detection	179
Nitik Garg, Piyush Kumar Vikram, Nidant Rajora, and Anurag Goel	
Vitunix: A Lightweight and Secure Linux Distribution	191
Sandip Shinde, Gourav Suram, Mayur Khadde, Shruti Gade, Vaishnavi Arthamwar, and Palak Pardeshi	
Multi-Key Fully Homomorphic Encryption Scheme Over the Integers	203
Rohitkumar R Upadhyay and Sahadeo Padhye	
A Modest Approach Toward Cloud Security Hygiene	217
Sujal Patel, Rashmi Agarwal, Shinu Abhi, and Ratan Jyoti	
PrimeSwitch—Encryption and Decryption Algorithm Using RSA Key Generation	229
Priyanka Bhatele, Ishan Shivankar, Shreya Sabut, Shivansh Saraswat, Shraddha Patel, Shrey Chougule, and Shreya Nale	
Hybrid Lightweight Cryptography Using AES and ECC for IoT Security	241
Neha N. Gharat and Lochan Jolly	
A Survey: Analysis of Existing Hybrid Cryptographic Techniques	259
Aman and Rajesh Kumar Aggarwal	
Identity-Based Designated Verifier Proxy Signature Scheme and Its Application to Health Care	271
Vandani Verma and Yash Sharma	
Benefits and Challenges of Integrating IIoT in Smart Energy Systems	279
Saumya and Shobhita Khatri	

Blockchain-Based Secure Mutual Authentication Scheme for Drone-GSS Communication in Internet of Drones Environment	289
Anvita Gupta, Ayushi Jain, and Mehak Garg	
SLMA: Secure and Lightweight Mutual Authentication Scheme for IoT-Based Healthcare	303
Preeti Dhankar, Priya Sharma, and Bhargavi Singh	
Cybersecurity: A Deep Learning Model for Intrusion Detection in IoT	311
Abhijeet Singh, Achyut Mishra, Ajit Antil, Bharat Bhushan, and Anamika Chauhan	
Performance Analysis of AES and DES Algorithm for Encrypting Medical Record Using Blockchain	325
J. A. Madhurya and K. Meena	
Strengthening Cybersecurity: A Comparative Study of KNN and Random Forest for Spam Detection	337
Sanya Joshi, Japanpreet, Lekha Rani, Pradeeptha Kumar Sarangi, and Ved Prakash Dubey	
Causes of Cyber Fraud in Commercial Banks in Nigeria: A Case Study of Zenith Banks in Abuja	351
Ekong Eyo Unwana and Rajesh Prasad	
Object Detection Using TensorFlow 2 and Amazon SageMaker	361
Vartika Goel, Deepak Arora, and Sheenu Rizvi	
Fake News Detection Using Machine Learning	375
Hanish Jindal, Mittali Mangla, and Gurpreet Singh	
Cloud-Based Integrated Real-Time Twitter Grievance Redressal with AWS EC2 and RDS Using Machine Learning Approach for Enhanced Security	387
Shambhavi Chauhan and Deepak Arora	
Profanity Detection from Audio Recordings Using Natural Language Processing Techniques	399
Swapnil Patil, Pankaj R. Chandre, Viresh Vanarote, Shafi Pathan, Madhukar Nimbalkar, and Gitanjali Shinde	
Defending Against Vishing Attacks: A Comprehensive Review for Prevention and Mitigation Techniques	411
Shaikh Ashfaq, Pankaj Chandre, Shafi Pathan, Uday Mande, Madhukar Nimbalkar, and Parikshit Mahalle	
Categorizing Tracing Techniques for Network Forensics	423
Shraddha Chourasiya, Ayush Indurkar, Apoorva Ghagare, Kaushal Potphode, Varun Sayam, and Dikshant Gaikwad	

malC: A Novel Deep Learning Architecture for Malware Classification	435
Harinadh Varikuti and ValliKumari Vatsavayi	
Cyber Threat Intelligence (CTI): An Analysis on the Use of Artificial Intelligence and Machine Learning to Identify Cyber Hazards	449
Neelima Kant and Amrita	
Cyber-Attack Detection Using Machine Learning Technique	463
Karan Singh, Surbhi Singh, Mehar Vohra, and Ravi Shankar Jha	
AE-LSTM: A Hybrid Approach for Detecting Deepfake Videos in Digital Forensics	475
Megha Kandari, Vikas Tripathi, and Bhaskar Pant	
Chatbot-Based Android Application Towards Security Using FCM	485
Priya Singh and Rajalakshmi Krishnamurthi	
Accuracy Enhancement for Intrusion Detection Systems Using LSTM Approach	499
Abhishek Kajal and Vaibhav Rana	
Enhanced Pin Entry Mechanism for ATM Machine by Defending Shoulder Surfing Attacks	515
Yogesh Kisan Mali, Vijay U. Rathod, Vishal Kisan Borate, Ashvini Chaudhari, and Tushar Waykole	
Machine Learning-Based Evaluation of Financial Risks in Cryptocurrency	531
Tanya Kapoor and Laxmi Ahuja	
A Review on Quantum Key Distribution Protocols, Challenges, and Its Applications	541
Neha Sharma, Pardeep Singh, Abhineet Anand, Sunil Chawla, Anuj Kumar Jain, and Vinay Kukreja	
Vulnerabilities in Smart Contracts of Decentralized Blockchain	551
Anurag Singh, Kapil Sharma, and Pradeepa Kumar Sarangi	
A Comprehensive Study of Blockchain Technology Trends and Analysis in the Healthcare Industry 4.0	567
Rakshit Bhadaria, Puneeta Singh, and Sartaj Ahmad	
Web 3.0-Based Crypto Wallet for Securing Assets and Blockchain Transactions	583
Vaibhav and Deepak Arora	
Orphanage Channelization System Using Blockchain Technology	593
Tapasya Choudhary, Vanshika Aggarwal, Shivansh Srivastava, and Shivani Trivedi	

Blockchain-Based Public Distribution System	605
Dipti Pawade, Chaitanya Bandiwdekar, Pooja Kaulgud, Siddhesh Bagwe, and Aditi Kulkarni	
Novel Architecture and Secured Food Traceability Application Based on Ethereum Blockchain	619
P. Joseph, B. Yuvaraj, and A. Sai Sabitha	
Quick Response Code-Based Fake Product Verification System Using Blockchain Technology	631
Vijayashree R. Budyal, Sriraksha R. Kolgi, B. K. Tejaswini, G. Hrithik, and S. A. Nabila Qadri	
Interactive Learning for Patient Care: Blockchain Ingrained Electronic Health Record Management System with Patient Control, Data Quality and Security Assurance	641
Arvind K. Sharma, Gousia Habib, Savita Wadhawan, and Himani Soni	

Editors and Contributors

About the Editors

Nihar Ranjan Roy (life time member of Cryptology Research Society of India) is a passionate teacher working as an Associate Professor in the Department of Computer Science and Engineering, School of Engineering and Technology at Sharda University, Greater Noida. Dr. Roy is an active member of the Center for Cyber Security and Cryptography at Sharda University. He holds a Doctorate and M.Tech. degree in Information Technology from Guru Gobind Singh Indraprastha University, Delhi, and Bachelor of Engineering degree from Biju Patnaik University of Technology, Rourkela, Odisha formerly known as Utkal University, Orissa. He has over 20 years of teaching experience in reputed colleges and universities of Delhi-NCR. He has authored and co-authored several research papers in reputed journals and international conferences, and is a reviewer of many SCI indexed journals from IEEE, Wiley, Elsevier and Springer to name a few. He has chaired several technical sessions and has reviewed papers of many international conferences from Springer, and IEEE. Dr. Roy's research areas includes: Computer Network, Cyber Security and Forensics, and Machine Learning. He has 4 published patents to his credit.

Some of the honors and awards to his credit are Best Teacher Award, Best Researcher Award, Best Teaching Website Design Award, and Best Personality Award. Dr. Roy has successfully organized the REDSET conference series which was published by Springer.

Sudeep Tanwar (Senior Member, IEEE) is working as a full professor at the Nirma University, India. He is also a Visiting Professor with Jan Wyzykowski University, Poland, and the University of Pitesti, Romania. He received B.Tech. in 2002 from Kurukshetra University, India, M.Tech. (Honor's) in 2009 from Guru Gobind Singh Indraprastha University, Delhi, India and Ph.D. in 2016 with specialization in Wireless Sensor Network. He has authored 04 books and edited 20 books, more than 300 technical articles, including top cited journals and conferences, such as IEEE TNSE, IEEE TVT, IEEE TII, IEEE TGCN, IEEE TCSC, IEEE IoTJ, IEEE NETWORKS,

IEEE WCM, ICC, IWCMC, GLOBECOM, CITS, and INFOCOM. He initiated the research field of blockchain technology adoption in various verticals in the year 2017. His H-index as per Google Scholar and Scopus is 61 and 47, respectively. His research interests include blockchain technology, wireless sensor networks, fog computing, smart grid, and the IoT. He is a member of the Technical Committee on Tactile Internet of IEEE Communication Society. Recently, He has been awarded cash prize of Rs. 50,000 for publishing papers with 5+ Impact factor and publication of books with Springer, IET and CRC under the scheme of “Faculty Awards and Incentives” of Nirma University for the year 2019–2020. He has been awarded the Best Research Paper Awards from IEEE IWCMC-2021, IEEE ICCCA-2021, IEEE GLOBECOM 2018, IEEE ICC 2019, and Springer ICRC-2019. He has won Dr. K. W. Wong Annual Best Paper Prize (with 750 USD) for 2021 sponsored by Elsevier (publishers of JISA). He has served many international conferences as a member of the Organizing Committee, such as the Publication Chair for FTNCT-2020, ICCIC 2020, and WiMob2019, and a General Chair for IC4S 2019, 2020, 2021, 2022, ICCSDF 2020, FTNCT 2021. He is also serving the editorial boards of COMCOM-Elsevier, IJCS-Wiley, Cyber Security and Applications-Elsevier, Frontiers of blockchain, SPY, Wiley, IJMIS journal of Inderscience, JCCE, and JSSS. He is also leading the ST Research Laboratory, where group members are working on the latest cutting-edge technologies.

Usha Batra is currently working as a Professor and Dean, MRIIRS. She is also holding additional responsibility as Director Research in Computational Technology and Science and Director, Manav Rachna Centre for Cyber Security. Before joining MR, she worked as Associate Dean, Engineering and Sciences at GD Goenka University. She is a doctorate in Computer Science Engineering from Banasthali University. She has over 20 years of rich academic, research and administrative experience in Indian university system including Jaypee University, Northcap University and GD Goenka University. She is a recipient of government recognised prestigious Indian Achievers' Award 2021 for Excellence in Education in recognition of outstanding professional achievement and contribution in nation building. She is a recipient of Dr. A. P. J. Abdul Kalam Puraskar 2022 for Excellence in Research and Innovation by ICSRR in association of NITI Aayog. She is a university rank holder during Bachelors of Engineering. She has authored or co-authored more than 70 scholarly research articles as conference papers, book chapters and in quality journals such as Nature's Scientific Reports, Elsevier's Health Policy and Technology, Material Sciences etc. to name a few . She has 15 national and international patents published and 5 national and international patents granted to her credit. She is an editor and reviewer of reputed and indexed journals of IEEE, Springer and Elsevier. She contributed as keynote speaker in IEEE international conference, session chair of various international conferences; and convener and organizer of many international conferences in association with IEEE and Springer. She presented herself as a special Invitee and IT expert for a live talk on ‘emerging trends in IT’ on IBN7. She is IBM certified faculty for various courses in the domain of cloud computing and Business Analytics. Her

current responsibilities also include designing state of the art curriculum and finalising MoUs with industries like IBM, Oracle, Xebia, Dell, Amazon, Microsoft etc. She has supervised over 30 projects at Undergraduate to Postgraduate level and got 6 national level winning projects. Six scholars have successfully completed their Ph.D. in her guidance and 2 more scholars have submitted their Ph.D. thesis. Her areas of research include Distributed Systems, Enterprise Application Integration, Business Analytics, Software Engineering, Cloud Computing, Internet of Things and Cyber Physical Systems.

In addition to her research, she successfully contributed to enhancing the employability of students through project assistance, teaching, and mentoring. She is actively working towards cyber security awareness sessions and playing a lead role to develop Indian Cyber Army.

Contributors

Shinu Abhi REVA Academy for Corporate Excellence, REVA University, Bengaluru, India

Lalit Adhana Computer Science and Technology, Manav Rachna University, Faridabad, India

Rashmi Agarwal REVA Academy for Corporate Excellence, REVA University, Bengaluru, India

Rajesh Kumar Aggarwal National Institute of Technology, Kurukshetra, Haryana, India

Vanshika Aggarwal Department of Computer Science and Engineering, ABESEC, Ghaziabad, India

Priyanka Ahlawat National Institute of Technology, Kurukshetra, Haryana, India

Sartaj Ahmad Department of IT, KIET Group of Institutions, Delhi-NCR, Ghaziabad, Uttar Pradesh, India

Laxmi Ahuja Amity Institute of Information Technology, Amity University, Noida, Uttar Pradesh, India

Nomel Haymes Axel-Elie Junior Ako Department of Computer Science and Engineering, Sharda University, Noida, India

C. U. Akshay Sai Vidya Institute of Technology, (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Aman National Institute of Technology, Kurukshetra, Haryana, India

Amrita Department of Computer Science and Engineering, School of Engineering and Technology, Center for Cyber Security and Cryptology, Sharda University, Greater Noida, Uttar Pradesh, India

Abhineet Anand Apex Institute of Technology, Chandigarh University, Mohali, India

Ajit Antil Delhi Technological University, New Delhi, India

Deepak Arora Department of Computer Science and Engineering, Amity School of Engineering and Technology Lucknow, Amity University, Noida, Uttar Pradesh, India

Vaishnavi Arthamwar Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India

Shaikh Ashfaq Information Technology Department, M H Saboo Siddik College of Engineering, Mumbai, India

Abirham Ayenew Computer Engineering, Vishwakarma Government Engineering College, Ahmedabad, Gujarat, India

Siddhesh Bagwe Department of Information Technology, K. J. Somaiya College of Engineering, Vidyavihar, Mumbai, India

Chaitanya Bandiwdekar Department of Information Technology, K. J. Somaiya College of Engineering, Vidyavihar, Mumbai, India

Subarna Basnet Department of Computer Science Engineering, Sharda University, Greater Noida, India

Rakshit Bhadaria Department of IT, KIET Group of Institutions, Delhi-NCR, Ghaziabad, Uttar Pradesh, India

Priyanka Bhatele Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra, India

Bharat Bhushan Delhi Technological University, New Delhi, India

Samiratou Bonkoungou Department of Computer Science and Engineering, Sharda University, Noida, India

Vishal Kisan Borate D Y Patil College of Engineering and Innovation, Talegaon, Pune, India

Vijayashree Budyal Sai Vidya Institute of Technology, (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Vijayashree R. Budyal Department of Information Science and Engineering, Sai Vidya Institute of Technology (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Pankaj Chandre Computer Science and Engineering Department, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, Pune, India

Pankaj R. Chandre Department of Computer Science and Engineering, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, India

Ashvini Chaudhari Symbiosis Skills and Professional University, Kiwale, Pune, India

Nishtha Chaudhari School of Technology, Pandit Deendayal Energy University, Gandhinagar, Gujarat, India

Ashim Chaudhary Sharda University, Greater Noida, UP, India

Anamika Chauhan Delhi Technological University, New Delhi, India

Shambhavi Chauhan Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Uttar Pradesh, Lucknow, India

Uttam Chauhan Computer Engineering, Vishwakarma Government Engineering College, Ahmedabad, Gujarat, India

Sunil Chawla Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Tapasya Choudhary Department of Computer Science and Engineering, ABESEC, Ghaziabad, India

Shrey Chougule Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra, India

Shraddha Chourasiya Department of Computer Science Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India

Preeti Dhankar IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India

Bhavika Dhingra Computer Science and Technology, Manav Rachna University, Faridabad, India

Swikrati Dubey Department of CSE, SIRT, Bhopal, India

Ved Prakash Dubey Graphic Era Hill University, Dehradun, Uttrakhand, India

Shrutika Gade Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India

Dikshant Gaikwad Department of Computer Science Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India

Mehak Garg IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), New Delhi, India

Nitik Garg Department of Computer Science and Engineering, Delhi Technological University, New Delhi, India

Riya Gaur Computer Science and Technology, Manav Rachna University, Faridabad, India

Apoorva Ghagare Department of Computer Science Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India

Neha N. Gharat Thakur College of Engineering and Technology, Mumbai, India

Anurag Goel Department of Computer Science and Engineering, Delhi Technological University, New Delhi, India

Vartika Goel Department of Computer Science & Engineering, Amity School of Engineering and Technology Lucknow, Amity University, Noida, Uttar Pradesh, India

Anvita Gupta IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), New Delhi, India

Chetan Gupta Department of CSE, SIRT, Bhopal, India

Gousia Habib Indian Institute of Technology Delhi, Delhi, India

G. Hrithik Department of Information Science and Engineering, Sai Vidya Institute of Technology (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Ayush Indurkar Department of Computer Science Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India

Naveen Ishika Sai Vidya Institute of Technology, (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Anuj Kumar Jain Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Ayushi Jain IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), New Delhi, India

Japanpreet Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Ravi Shankar Jha Computer Science & Engineering, DIT University, Dehradun, India

Hanish Jindal Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Lochan Jolly Thakur College of Engineering and Technology, Mumbai, India

P. Joseph Department of Computer Science and Engineering, Hindustan Institution of Technology and Science, Padur, Chennai, India

Sanya Joshi Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Ratan Jyoti Ujjivan Small Finance Bank, Bengaluru, India

Abhishek Kajal Department of Computer Science and Engineering, Guru Jambheeshwar University of Science and Technology, Hisar, Haryana, India

Megha Kandari Graphic Era Deemed to be University, Dehradun, India

Neelima Kant Department of Computer Science and Engineering, School of Engineering and Technology, Sharda University, Greater Noida, Uttar Pradesh, India

Tanya Kapoor Amity Institute of Information Technology, Amity University, Noida, Uttar Pradesh, India

Pooja Kaulgud Department of Information Technology, K. J. Somaiya College of Engineering, Vidyavihar, Mumbai, India

Mayur Khadde Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India

Shobhita Khatri IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India

Sriraksha R. Kolgi Department of Information Science and Engineering, Sai Vidya Institute of Technology (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

K. C. Krishna Sharda University, Greater Noida, UP, India

Rajalakshmi Krishnamurthi Jaypee Institute of Information Technology, Noida, India

Vinay Kukreja Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Aditi Kulkarni Department of Information Technology, K. J. Somaiya College of Engineering, Vidyavihar, Mumbai, India

Krishan Kumar National Institute of Technology, Kurukshetra, Haryana, India

Manoj Kumar Computer Science and Technology, Manav Rachna University, Faridabad, India

Prakash Kumar Department of Computer Science and Cyber Security, JRSU, Ranchi, India

J. A. Madhurya Department of Computer Science and Engineering, GITAM School of Technology, GITAM University, Bangalore, India

Parikshit Mahalle Artificial Intelligence and Data Science Department, Vishwakarma Institute of Information Technology, Kondhwa, Pune, India

Yogesh Kisan Mali G H Raisoni College of Engineering and Management, Wagholi, Pune, India

Uday Mande Computer Science and Engineering Department, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, Pune, India

Mittali Mangla Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

K. Meena Department of Computer Science and Engineering, GITAM School of Technology, GITAM University, Bangalore, India

Achyut Mishra Delhi Technological University, New Delhi, India

Alpna Mishra Center of Cyber Security and Cryptology, Sharda University, Noida, India

S. A. Nabila Qadri Department of Information Science and Engineering, Sai Vidya Institute of Technology (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Shreya Nale Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra, India

Mohammad Gauhar Nayab Sharda University, Noida, India

Madhukar Nimbalkar Department of Computer Science and Engineering, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, India

Sahadeo Padhye Department of Mathematics, Motilal Nehru National Institute of Technology Allahabad, Prayagraj, Uttar Pradesh, India

Aman Raj Pandey Department of Computer Science Engineering, Sharda University, Greater Noida, India

Bhaskar Pant Graphic Era Deemed to be University, Dehradun, India

Palak Pardeshi Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India

Viral Parmar Pandit Deendayal Energy University, Gandhinagar, India

Samir Patel School of Technology, Pandit Deendayal Energy University, Gandhinagar, Gujarat, India

Shraddha Patel Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra, India

Sujal Patel REVA Academy for Corporate Excellence, REVA University, Bengaluru, India

Shafi Pathan Department of Computer Science and Engineering, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, India

Swapnil Patil Department of Computer Science and Engineering, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, India

Dipti Pawade Department of Information Technology, K. J. Somaiya College of Engineering, Vidyavihar, Mumbai, India

Urmila Pilania Computer Science and Technology, Manav Rachna University, Faridabad, India

Kaushal Potphode Department of Computer Science Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India

Rajesh Prasad Department of Computer Science and Engineering, Ajay Kumar Garg Engineering College, Ghaziabad, India

Dharm Raj Sharda University, Greater Noida, UP, India

Gaurav Raj Sharda University, Noida, India

Nidant Rajora Department of Computer Science and Engineering, Delhi Technological University, New Delhi, India

Vaibhav Rana Department of Computer Science and Engineering, Guru Jambheshwar University of Science and Technology, Hisar, Haryana, India

Lekha Rani Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Vijay U. Rathod G H Raisoni College of Engineering and Management, Wagholi, Pune, India

Sheenu Rizvi Department of Computer Science & Engineering, Amity School of Engineering and Technology Lucknow, Amity University, Noida, Uttar Pradesh, India

Nihar Ranjan Roy Department of Computer Science and Engineering, Sharda University, Noida, India

Shreya Sabut Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra, India

A. Sai Sabitha Department of Computer Science and Engineering, Hindustan Institution of Technology and Science, Padur, Chennai, India

Pradeeptha Kumar Sarangi Chitkara University Institute of Engineering and Technology, Rajpura, Punjab, India;

Chitkara University School of Engineering and Technology, Chitkara University, Himachal Pradesh, India

Shivansh Saraswat Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra, India

Saumya IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India

Varun Sayam Department of Computer Science Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India

Sonia Setia Department of Computer Science Engineering, Sharda University, Greater Noida, India

Md Shadik Sharda University, Greater Noida, UP, India

Arvind K. Sharma Yogananda School of Artificial Intelligence, Computers and Data Science, Shoolini University, Solan, Himachal Pradesh, India

Kapil Sharma Chitkara University Institute of Engineering and Technology, Rajpura, Punjab, India

Neha Sharma Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

Priya Sharma IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India

Ritik Sharma Sharda University, Noida, India

Tushar Sharma Department of Computer Science Engineering, Sharda University, Greater Noida, India

Yash Sharma Department of Mathematics, Amity Institute of Applied Sciences, Amity University, Noida, India

Gitanjali Shinde Department of Computer Science and Engineering, Vishwakarma Institute of Information Technology, Kondhwa, India

Sandip Shinde Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India

Ishan Shivankar Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra, India

Abhijeet Singh Delhi Technological University, New Delhi, India

Aditya Pratap Singh Sharda University, Noida, India

Anurag Singh Chitkara University Institute of Engineering and Technology, Rajpura, Punjab, India

Bhargavi Singh IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India

Darshita Singh Department of Computer Science and Engineering, Amity School of Engineering and Technology Lucknow, Amity University, Uttar Pradesh, India

Gurpreet Singh Chitkara University Institute of Engineering and Technology,
Chitkara University, Rajpura, Punjab, India

Karan Singh Computer Science & Engineering, DIT University, Dehradun, India

Pardeep Singh Computer Science and Engineering, Graphic Era Hill University,
Dehradun, India

Priya Singh Jaypee Institute of Information Technology, Noida, India

Puneeta Singh Department of IT, KIET Group of Institutions, Delhi-NCR, Ghaziabad, Uttar Pradesh, India

Surbhi Singh Computer Science & Engineering, DIT University, Dehradun, India

Himani Soni Department of Neurosurgery, M.M. Superspeciality Hospital, Maharishi Markandeshwar Deemed to be University, Mullana, Haryana, India

Aviral Srivastava Amity University, Jaipur, Rajasthan, India

Saurav Kumar Srivastava Computer Science and Technology, Manav Rachna University, Faridabad, India

Shivansh Srivastava Department of Computer Science and Engineering, ABESEC, Ghaziabad, India

Gourav Suram Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India

B. K. Tejaswini Department of Information Science and Engineering, Sai Vidya Institute of Technology (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Vikas Tripathi Graphic Era Deemed to be University, Dehradun, India

Shivani Trivedi Department of Computer Science and Engineering, ABESEC, Ghaziabad, India

Gaonkar Unnathi Sai Vidya Institute of Technology, (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Ekong Eyo Unwana African University of Science and Technology, Abuja, Nigeria

Rohitkumar R Upadhyay Department of Mathematics, Motilal Nehru National Institute of Technology Allahabad, Prayagraj, Uttar Pradesh, India

Sourav Kumar Upadhyay Department of Computer Science and Engineering, BIT Sindri, Dhanbad, India

Vaibhav Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Uttar Pradesh, Lucknow, India

A. V. Vaibhav Sai Vidya Institute of Technology, (Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India

Viresh Vanarote Department of Computer Science and Engineering, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, India

Harinadh Varikuti Andhra University, Visakhapatnam, India

ValliKumari Vatsavayi Andhra University, Visakhapatnam, India

Vandani Verma Department of Mathematics, Amity Institute of Applied Sciences, Amity University, Noida, India

Piyush Kumar Vikram Department of Computer Science and Engineering, Delhi Technological University, New Delhi, India

Mehar Vohra Computer Science & Engineering, DIT University, Dehradun, India

Savita Wadhawan M. M. Institute of Computer Technology and Business Management, Maharishi Markandeshwar Deemed to be University, Mullana, Haryana, India

Tushar Waykole Nutan Maharashtra Institute of Engineering and Technology, Talegaon, Pune, India

B. Yuvaraj Department of Computer Science and Engineering, Hindustan Institution of Technology and Science, Padur, Chennai, India

A Low-Distortion Reversible Steganography Method that Conceals Data in Images with Minimal Distortion



Mohammad Gauhar Nayab, Aditya Pratap Singh, Ritik Sharma, and Gaurav Raj

Abstract Image steganography is one of the most well-known methods where a top-secret data is embedded into the pixels of an image by altering some of its pixel values. Reversible image steganography is used extensively due to its property of reconstructing the original cover image without any loss. Various studies have been conducted to achieve stego image with high embedding capacity, less time complexity, high imperceptibility, and robustness. Keeping the necessary parameters under consideration, we conducted our study on reversible image steganography to achieve a better peak signal-to-noise ratio (PSNR) value or high embedding capacity which helps us in analyzing the quality of an embedded image. We have analyzed PSNR ratio and compared it with the standard tests ensuring the quality of stego image. We have also briefed various methods used for reversible image steganography based on compression, encryption, high capacity, quality, and performance.

Keywords Image steganography · Peak signal-to-noise ratio · Steganography using reverse image steganography

1 Introduction

Communication is crucial in everyday life. The ability to communicate over networks is a blessing for the generation of today in this increasingly digital world. Information must be transferred securely and without being misused. The two most

M. G. Nayab (✉) · A. P. Singh · R. Sharma · G. Raj
Sharda University, Noida, India
e-mail: gauharnayabnepal@gmail.com

A. P. Singh
e-mail: 2019003284.aditya@ug.sharda.ac.in

G. Raj
e-mail: gaurav.raj@sharda.ac.in

popular methods for preventing cyber-attacks and establishing secure communication between the two parties are cryptography and steganography. The art of stenography involves hiding information on a number of untraceable carriers in a way that the recipient is the only one who can make sense of it. The process of converting plain text into an unintelligible format is known as cryptography. It is a method of encrypting data that can be broken open with enough time and effort. The information is not secret. Data in an original cover can be hidden using steganography. Image steganography is a method of concealing secret information in the pixels of the cover or original images, resulting in a stego image [1]. In stego image, there are two categories, i.e., image that can be reversed or not.

There is no way to remove the hidden text in irreversible steganography, but the cover image remains as it was before the hidden text was added. Reversible steganography allows for the recovery of both the cover image's original state and any hidden text. Images are warped when secret information is inserted into the cover image so that it is impossible to tell the stego image apart from the original cover image [2].

The two most important factors for successful steganography are high embedding capacities and good stego image quality.

Both compressed and uncompressed images are capable of implementing this data concealment. Although it is the simplest task, hiding the secret data in the least significant bit (LSB) of each pixel of the cover image in an uncompressed image degrades the quality of a stego image. Similar to noise, LSB techniques lower the quality of the image. For reducing distortion's impact on the quality of the stego image, several methods have been proposed. These strategies involve adding data to the intricate, as opposed to the smooth, areas of the image. The techniques for compressing images on the other side [3] include the discrete cosine transformation (DCT), discrete wavelet transformation (DWT), block truncation coding (BTC), and vector quantization (VQ). Here, using BTC is the simplest task and requires the least amount of computation. The compression rate is lower with this method than with others, which is a drawback. Then, various efforts were made to increase the rate of BTC compression, and many steganography techniques were also put forth for uncompressed images. Reversible techniques were incredibly scarce.

2 Literature Review

Various works were carried out by various researchers using various carriers.

Sayuthi et al. a work that primarily focused on choosing a cover image by combining distortion and joint-image similarity. They therefore proposed a modified single-valued decomposition method based on stego images that took into account small values rather than large values among the available images in order to make it suitable to fit the characteristics of steganography. For calculating the inserting distortion, they used the current framework for distortion minimization. They calculated the new cover selection strategy by combining the two properties. Their research is

unique in that it demonstrates the best method for choosing cover images through the use of cutting-edge steganalysis tools. On additional image processing projects, they also labored. It only functions in the spatial domain, which is a drawback. The computational challenge is exacerbated by their jpeg format compression. Therefore, it fails to operate in DCT domains. They lack a versatile framework for cover selection that works in both the three-dimensional and DCT domains [4].

Aisha and Tu increased the capacity and concealment of the hidden data, and they created the color spacing normalization text stenography model. The secret data was concealed in text used as cover text. The model is composed of two stages, the first of which is the pre-inserting stage and the second of which is the normalization of color and space. In order to achieve high capacity, they used RGB coding and character spacing, which helped them increase the number of bits per place. As a result, they increased the secret block's size from earlier studies' use of just 12 bits per location to 16 bits. They used Huffman coding to compress the delicate data in order to increase the storage capacity. The second stage aids in normalizing RGB coding to improve cover and stego text visibility. As a result of normalization, the cover and the stego text color differ less. This model's beauty is that it has a high accuracy rate of 98.85% when compared to models that already exist. Additionally, it is the best method for cutting the color difference for cover text by 47% for black and 57% for colored text to achieve high invisibility. This model's primary flaw is that they did not work on normalizing the RGB code for image stenography. Even without a strong stego key, they cannot safely share sensitive information with one another, defeating the purpose of high security [5].

Liu et al. worked on text stenography using deep learning. This technique, referred to as generative linguistic stenography, is not very effective at decrypting the hidden data that is stored. The relationship between the hidden text cannot be established. Finding the hidden image also doesn't require using the original cover. This method has the advantage of a high insertion capacity. Thus, they sought to precisely connect the hidden data. They developed a distributed read in module to convert words into precise number vectors. In order to fully utilize the ELM Model (Explicit and Latent Text Word Relation) developed model and determine whether the secret text is present in the currently playing segment, they subsequently created the global adaptive classification Unit. Their suggested approach has the drawback of optimizing performance but not temporal complexity [6].

Khodaei and Faez also worked on deep learning model. There is no need for an original cover with these models. Consequently, they are referred to as predictive models. While these deep learning models are retrieving the hidden data, there are also a few minor errors caused by variables and numbers that are not equal to zero or identity. Their primary goal was to identify uncertainty in these models caused by these tiny errors. Instead of attempting to predict the hidden data, they used the Bayesian algorithm to focus on measuring uncertainty. They used two algorithms to approximate the value before further categorizing these uncertainties into aleatoric and epistemic uncertainties. This article's limitation is that it only addresses uncertainty rather than inserting a pathway [7].

Wang et al. concentrated on video steganography, which was primarily developing methods for minimizing and introducing additive distortion. In spite of the fact that additive inserting distortion is inappropriate, they still think that cover video interplay is the best option. They reduced the non-additive distortion in high efficiency video coding (HEVC) by using the intraprediction model (IPM). As a result, they worked to simplify non-additive distortion. They combined a proposed inserting distortion with a multi-layered inserting structure. They contribute to bettering security performance and encoding effectiveness [8].

Zheng et al claimed that stenography falls into one of two categories, i.e., methods that are driven by data or by predetermined rules. So, they attempted to combine a model aiming for security and one resisting in-depth steganalysis. They developed a rule-based method for blending images to artificially produce an infinite number of label pairs. They strive to achieve outstanding image steganography and data reconstruction performance [9]. By reassigning the cost function, Chen and Wang hope to increase the security of adaptive image stenography. In essence, it is focusing on the created immune system. This method reassigned the cost function in the probability distribution method to reduce distortion and further optimizes the model using an immunity-based data hiding model. The result is that the pixels are altered, making it challenging to detect. It assists in fending off steganalysis. It does not function in the jpeg spatial domain, which is a drawback [10].

Zhang and Zhong aimed to work on high capacity and high imperceptibility of image stenography. Lightweight transform discrete Hadamard model has been developed for ensuring high security when inserting 8 BPP densely. Even more robustness tests were conducted using JPEG compression, three different noise attack types, and cropping attacks. Their failure to develop edge-based detection technologies is a drawback [6]. The Hung-Jui et al. worked on creating an efficient stenography technique, converting partial secret data bits into fractions and then mapping those fractions to predefined cubes using the magic cube and modulus operation. Incorporating the coordinates for the mapped position into the cover image was achieved by performing a modulus operation on them. They were able to achieve stego image quality that was superior to 42 dB with a payload of 3 BPP. The fact that BPP is so low is a drawback [11].

Mohammed and Rossilawati used least significant bit (LSB) algorithm to hide secret data where they tried to replace the least bit with the secret data. They tried to make it more optimal and reduce the variability. As a result, they ran a number of optimization algorithms, which increased the time complexity. As a result, they employed the flipping method, a data hiding technique, which reduced variability and time complexity. They used a method where they first identified the data hiding for the k-LSB of the stego and cover images, and then they compared the difference to a threshold value. Stego images have higher pixels than edge images because the next bit is flipped ahead of the k-LSB to get closer to the cover image. Thus, good visual quality and reduced variability are provided [12].

Dey prepared a medical ambulance drone which is controlled by a human in the hospital control room, and the rescue truck is watched to ensure that it arrives on time, lowering the number of patients who pass away as a result of the wait. The

GPS location of the patient is used by this drone to provide medical assistance, and hospitals receive audio and video streams of the patient's data, where the security of transmission of information is crucial. They try to provide basic treatment to the patient by providing the right guidance to the available nearby centers. They used lightweight cryptography and variable MSB and LSB image stenography [13].

Vajiheh et al. have done a fantastic job of concealing it using the natural JPEG images' ability to share content. The DCT approach and the parallel channel method were also employed to decrease distortion. By changing the insertion cost of the DCT coefficient using these two strategies, they were able to make JPEG stego pictures more unpredictable. The work is limited to the JPEG spatial domain, which is its primary fault [14]. The cover picture was split into 5×5 blocks as part of Mohan Bhandari, Sanjeeb Prasad Panday, Subash Panday, and Chandra Prakash's ant colony optimization-based image steganography approach to improve the pixel quality. By using a triangular chaotic map to find the next qualifying pixels based on pheromone trails, it was possible to further encrypt the text data. Its distinguishing feature is that, when measured against LSB and other algorithms, they come out on top. The issue is that they didn't use it to optimize bee colonies [15].

Kamaldeep and Rajkumar outlined the importance of stenography and cryptography for securely delivering data to the intended user. The key components of steganography are underlined, including the necessity of being able to secure or conceal data across a variety of carriers in a way that only the receiver can benefit. The process of converting simple text or data into an incomprehensible format is made plainly evident by them. While conveying data, it's vital to take time complexity, resilience, quality, and the capacity of storing information per pixel into account. Furthermore, crucial is embedding capability [16].

Marghny and Loay focused on finding efficient stenography technique for hiding data. They also used the LSB method in a different way than the conventional LSB Substitution, which reduces picture quality in an effort to enhance the inserting technique. They thus tried to reverse the cover image by swapping the LSB for the hidden information. The greatest and lowest values were sought after, and they were all subtracted from the highest value before being split and added to the cover picture. The best characteristic is that, because of its linear time complexity, it operates more quickly than other algorithms [17].

Our study focuses on calculating the PSNR value and reversible image steganography's embedding capability, assuring the quality of the stego picture once the data has been hidden, and giving a quick review of methods for concealing sensitive information in both compressed and uncompressed images.

3 Methodology

This section will discuss the suggested approach. As was already said, this method depends on a few mathematical operations, such as LSB inversion. The cover pixels' LSBs are not replaced with secret bits instead, their values are inverted. As a result,

the stego picture's quality is improved. To identify which bits are inverted, a flag is thus needed [6, 9]. Finding the maximum and lowest numbers, as well as subtraction and division, are examples of arithmetic operations. These arithmetic procedures are used to increase capacity and decrease the amount of embedded data. Quotients and leftovers from division operations are kept in separate arrays referred to as quotients and remainders. Because of the use of the divisors 32 and 8, each quotient can have a maximum size of 3 or 2 bits, with 3 bits [6] being the maximum number for the remainder. The cover picture has been divided into two equally sized pieces. The first section is used to incorporate quotients by flipping pixel LSBs. Also, to improve the findings, certain inverted LSBs will be inverted once more in this section. The second portion is used to incorporate leftover data by flipping pixel LSBs. It also specifies how inverted bits were reversed, as well as how many bits were used to embed each pixel in the first half. Here, we will explain how to hide (also known as embedding) and retrieve (also known as extraction) secret data. In the paragraphs that follow, the picture used in the recommended process is described.

Suppose I is any grayscale picture and has the following pixel values:

$$I = \{P_1, P_2, \dots, P_n\}$$

Each pixel has 8 bits:

$$|P_i| = 8 \text{ bits}, P_i = \{b_1, \dots, b_8\}, b_j \in \{1, 0\}$$

The picture size is calculated as

$$N = H \times W$$

The image's height and width are denoted by H and W , respectively. Considering that M and n indicate the length and individual bits of the secret data,

$$M = \{m_1, m_2, \dots, m_n\}, \quad \text{where } m_i \in \{1, 0\}$$

h is the highest level of concealment possible in image I , calculated in units of bits as

$$1 \leq h \leq (N \times 8)$$

The Embedding Algorithm

As secret data and as the input for embedding, either a grayscale image. The outcome is a stego picture in grayscale. Taking into account that C is the remainder array and that A and B are the quotient arrays. Each array can only be as big as the cover picture, which is its maximum size.

1. A decimal number between 0 and 255 is utilized to calculate the secret data's highest and lowest values.
2. For each concealed data point, deduct the highest value from the lowest value.
3. Put the result in array A after dividing by 32, then divide the remaining sum by 8 using the result.
4. While just two bits are required to express a R value between 0 and 127, three bits are required to signify a R value larger than 127.
5. Embed the remaining portion of the greatest value in the second pixel of the cover image's second row, column, and LSB, as mentioned in the step before.
6. This pixel's fourth bit should show the value of R . The four low-pass filtered (LSB) bits of each pixel are inverted for embedding in the first section if this value is less than or equal to 127; otherwise, the five LSB bits are inverted. Three bits are needed to indicate a R value greater than 127, whereas only two bits are needed to represent a R value between 0 and 127. Because of this, the sensitive number 127 can tell the difference between representations of 2 and 3 bits.
7. Use the best LSBs technique to the pixels you got in steps 5 and 7.
8. Every pixel in arrays A , B , and C in the cover picture will be included if phases 5 and 6 are repeated.
9. For each pixel in the first half, invert a few inverted bits once more. There are two scenarios: either one or two inverted bits will reverse. R 's value will determine how to proceed.
10. After embedding, some flipped bits for each image in the first half will be reversed once more to improve the outcomes. If the number of R is less than or equal to 127, there are two scenarios that are used as an illustration. The second LSB is likewise inverted in the first example. The second and fourth LSBs are inverted once again in the second example. Another two scenarios will be used if R has a value larger than 127. The third LSB is likewise inverted in the first example. The third and fourth LSBs are inverted once again in the second example. It specifies which bits of the associated pixel in the first part will be inverted again using the fourth bit of each pixel in the second part.
11. We apply the best LSBs method to each value for the first part of the pixel acquired in the previous step.
12. Now that the optimal LSBs method has been applied, each color has two values. Return either 0 or 1 as a signal to show which inversion was selected, and then select the number that is nearest to the initial value.
13. Hence, the fourth bit of the pixel should be inverted in the second part of the image.
14. Use the best LSBs approach on the newly acquired pixel.
15. Once everything has been included into the cover image, the stego picture is produced and emailed to the receiver.

The Extracting Algorithm

The buried data must be recovered by locating the bits that were inverted for each frame of the original picture. Dividing the stego and original photos into two halves will enable you to retrieve the hidden data. Next, proceed as follows:

1. Do the steps from 2 through 6 to determine the maximum value.
2. LSB values of the first pixel in the second part of the stego image can be compared to original values in order to calculate the remainder.
3. A quotient can be calculated by comparing the LSBs of the first pixel in the stego picture with the initial values.
4. Add the value of the remaining after multiplying this quotient's value by 8 times.
5. It is possible to determine the value of the quotient of the first pixel in the first half of the stego picture by comparing its third, fourth, and fifth least significant bits (LSBs) with those in the original image. To do this, it is necessary to compare the two sets of LSBs and determine what the difference between them is. This difference can then be used to calculate the value of the quotient. By examining the two sets of LSBs and calculating the difference between them, it is possible to accurately determine the value of the quotient, which can then be used to further analyze the image.
6. Add the data acquired in step 4 and multiply this quotient's value by 32.
7. If they are different, then one bit is required to embed that pixel. If they are the same, then no bits are required to embed that pixel.
8. By comparing the fourth LSB value of each pixel with the original value, repeat step 2 for each pixel in the stego picture.
9. The original value of the confidential data can be recovered once you are aware of the bits that have been inverted once more by carrying out steps 3 through 6 and then deducting the result from the greatest value.
10. To retrieve the secret data, repeat the preceding process for each and every pixel.

4 Experimental Results

Peak signal-to-noise ratio (PSNR), mean squared error (MSE), and structural similarity index are the statistical features we use to examine the same in the photographs of the couples under study (SSIM). Histograms of the photos are received in the same setting. A computer system for embedding/extracting text messages has been implemented, and several testing using messages and pictures of various sizes have been conducted. The LSB approach is the foundation of the investigated algorithm, which was tested on JPG image formats. Analysis is done on the test findings for the qualitative traits MSE, PSNR, SSIM, and embedded capacity (Figs. 1, 2, 3, 4 and 5).

The qualitative features of English-language embedded text files that are 60 kB in size and have parrot-cover digital images are shown in Table 1. bmp is employed in many LSBs (Fig. 6).

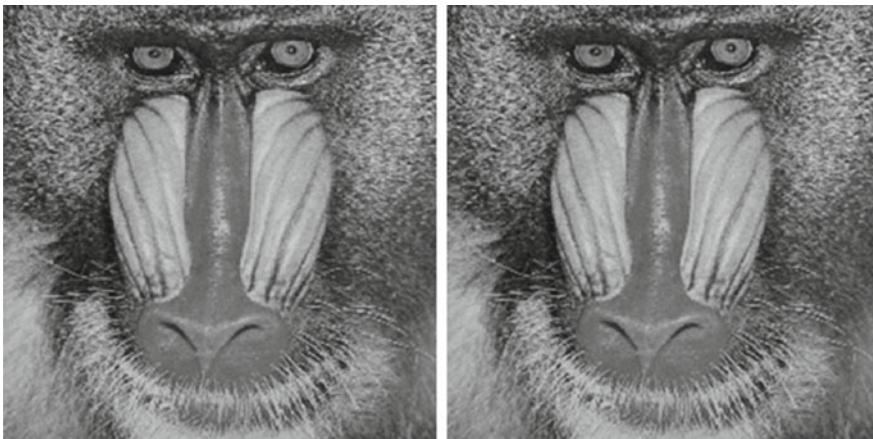


Fig. 1 Baboon



Fig. 2 Penguin



Fig. 3 Boat



Fig. 4 Cameraman



Fig. 5 House

Table 1 PSNR values of RIS techniques

Name of image	PSNR values (dB)	Embedding capacity (bytes)	MSE	SSIM
Baboon	44.43	39,657	1.908721	0.999803
Boat	43.17	46,516	3.564010	0.999042
Cameraman	44.14	44,228	5.294581	0.998537
Penguin	44.59	45,788	3.215111	0.998895
House	46.86	45,847	5.924808	0.998541

According to Table 2, comparison of the given and experimental methods, the peak signal-to-noise ratio (PSNR) of the implemented technique is lower than that of the experimental PSNR, demonstrating that the method's application reduces noise and other picture disturbances. The PSNR data in the table also illustrates how much

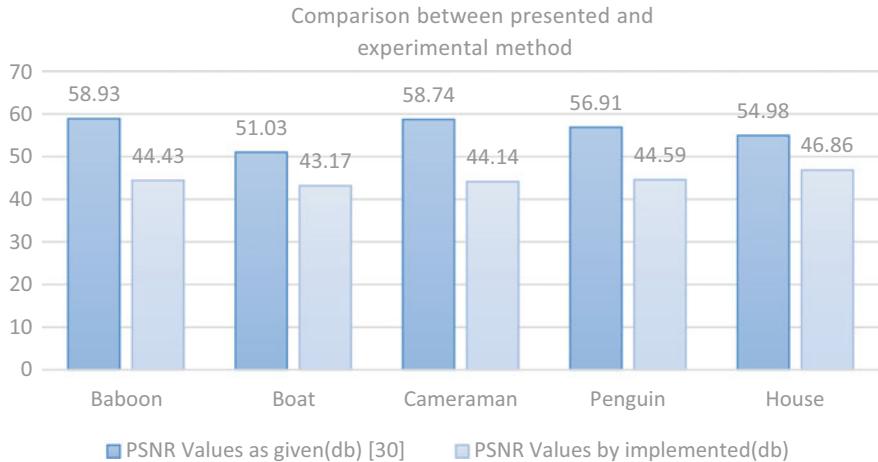


Fig. 6 Comparison of given PSNR versus implemented PSNR

Table 2 Comparison between the presented and experimental method

Name of image	PSNR values as given (dB) [18]	PSNR values by implemented (dB)	Embedding capacity as given (bit) [18]	Embedding capacity as calculated (bit)
Baboon	58.93	44.43	120,774	39,657
Boat	51.03	43.17	128,512	46,516
Cameraman	58.74	44.14	99,743	44,228
Penguin	56.91	44.59	142,574	45,788
House	54.98	46.86	134,847	45,847

better it is than the prior approaches. Also, the image's ability to be embedded was lowered, but the data's embedding quality improved.

5 Conclusion

The suggested LSB method-based picture steganography technique was developed. The histograms of the stego images show the greatest outcomes when steganography is used in the spatial realm, which limits their ability to contemporary stego analysis. The statistical properties of the pictures degrade as the embedding data quantity increases, but the visual clarity of the steganography-processed images is still very good.

After examining various methods used for steganography by various researchers, we discovered that the main issue that everyone is dealing with when performing

steganography is the problem of major noise issues on different spots of the steganographic image, which almost defeats the purpose of the entire process because it is simple for other humans to know that the image is being manipulated and there is something wrong, which can even lead the human to detect the secret information. Therefore, we developed a model and looked at various reversible steganographic techniques to reduce the noise effect and improve performance through high PSNR value and embedding capacity in the image, making it difficult for other humans to detect any changes and reducing the likelihood of suspicion on the images containing the secret data. While exploiting the maximum informational potential of the steganographic picture, our strategy also aims to reduce its detectability.

References

1. Lee C-F, Wang K-H, Chang C-C, Huang Y-L (2009) A reversible data hiding scheme based on dual steganographic images. In: Proceedings of the 3rd international conference on ubiquitous information management and communication, pp 228–237
2. Chang C-C, Chen T-S, Chung L-Z (2002) A steganographic method based upon JPEG and quantization table modification. *Inf Sci* 141(1–2):123–138
3. Delp E, Mitchell O (1979) Image compression using block truncation coding. *IEEE Trans Commun* 27(9):1335–1342
4. Thabit R, Udzir NI, Yasin SM, Asmawi A, Gutub AA-A. Color spacing normalization text steganography model to improve capacity and invisibility of hidden data. <https://doi.org/10.1109/ACCESS.2022.3182712>
5. Cao Y, Zhou Z, Chakraborty C, Wang M, Wu QMJ, Sun X, Yu K (2022) Generative steganography based on long readable text generation. *IEEE Trans Comput Soc Syst*
6. Chang C-C. Bayesian neural networks for reversible steganography. <https://doi.org/10.1109/ACCESS.2022.3159911>
7. Wang J, Yin X, Chen Y, Huang J, Kang X. An adaptive IPM-based HEVC video steganography via minimizing non-additive distortion
8. Zheng Z, Hu Y, Bin Y, Xu X, Yang Y, Shen HT. Aware image steganography through adversarial self-generated supervision. *IEEE Trans Neural Netw Learn Syst*
9. Chen Y, Wang H, Li W, Luo J (2022) Cost reassignment for improving security of adaptive steganography using an artificial immune system. *IEEE Signal Process Lett* 29
10. Zhang Y-Q, Zhong K, Wang X-Y. High-capacity image steganography based on discrete Hadamard transform. <https://doi.org/10.1109/ACCESS.2022.3181179>
11. Ko H-J, Huang C-T, Tseng H-W, Wang S-J. Efficient cost-reduced with high-quality image of imperceptible steganography using modulo and magic cube. <https://doi.org/10.1109/ACCESS.2022.3185120>
12. Kamil S, Abdullah SNHS, Hasan MK, Bohani FA. Enhanced flipping technique to reduce variability in image steganography. <https://doi.org/10.1109/ACCESS.2021.3133672>
13. Emergency medical assistance by ambulance drone using machine learning, light-weight cryptography and variable image steganography. In: Circuit and system conference 2022 (IEEE VLSI DCS 2022), 26–27 Feb 2022. IEEE ED MSIT SBC, Kolkata
14. Wang Z, Feng G, Qian Z, Zhang X. JPEG steganography with content similarity evaluation. *IEEE Trans Cybern*
15. Image steganography approach based ant colony optimization with triangular chaotic map. In: 2022 2nd international conference on innovative practices in technology and management (ICIPMT) 429. IEEE. ISBN: 978-1-6654-6643-1/22/\$31.00 ©2022

16. A brief review on various aspects of steganography followed by cryptographic analysis. In: 2022 IEEE 7th international conference for convergence in technology (I2CT), Pune, India, 07–09 Apr 2022
17. Nashat D, Mamdouh L (2019) An efficient steganographic technique for hiding data. *J Egypt Math Soc*
18. Ye H, Su K, Cheng X, Huang S (2022) Research on reversible image steganography of encrypted image based on image interpolation and difference histogram shift. *IET Image Process*
19. Wang Z, Feng G, Shen L, Zhang X (2022) Cover selection for steganography using image similarity. *IEEE Trans Dependable Secure Comput*
20. Cox IJ, Kilian J, Leighton FT, Shamoon T (1997) Secure spread spectrum watermarking for multimedia. *IEEE Trans Image Process* 6(12):1673–1687
21. Chang C-C, Lin C-Y (2006) Reversible steganography for VQ-compressed images using side matching and relocation. *IEEE Trans Inf Forensics Secur* 1(4):493–501
22. Chang C-C, Lin C-Y, Fan Y-H (2011) Reversible steganography for BTC-compressed images. *Fund Inform* 109(2):121–134
23. Yang C-H, Wang W-J, Huang C-T, Wang S-J (2011) Reversible steganography based on side match and hit pattern for VQ-compressed images. *Inf Sci* 181(11):2218–2230
24. Dharwadkar NV, Amberker BB (2010) An improved reversible steganography scheme based on dual cover images. *Int J Multimed Intell Secur* 1(4):336–349
25. Yin Z, Hong W, Tang J, Luo B (2016) High capacity reversible steganography in encrypted images based on feature mining in plaintext domain. *Int J Embedded Syst* 8(2–3):249–257
26. Eskicioglu AM, Fisher PS (1995) Image quality measures and their performance. *IEEE Trans Commun* 43(12):2959–2965
27. Yang B, Lu Z-M, Sun S-H (2005) Reversible watermarking in the VQ-compressed domain. In: Proceedings of the fifth IASTED international conference on visualization, imaging, and image processing, Benidorm, Spain, pp 298–303
28. Mohamed MH, Mohamed LM (2016) High capacity image steganography technique based on LSB substitution method. *Appl Math Inf Sci* 10(1):259
29. Chang C-C, Chou Y-C, Kieu TD (2009) Information hiding in dual images with reversibility. In: Proceedings of the third international conference on multimedia and ubiquitous engineering. IEEE, Qingdao, pp 145–152
30. Kamau GM (2014) An enhanced least significant bit steganographic method for information hiding. PhD thesis

Credit Card Fraud Detection Using ML Techniques



Samiratou Bonkoungou, Nihar Ranjan Roy,
Nomel Haymes Axel-Elie Junior Ako, and Alpna Mishra

Abstract With the increasing use of Internet, online banking transactions have also increased. Everyday more and more users are being added to the cyberspace, which has also attracted cybercriminals. These scammers use stolen credit card information for making online purchases. The revolution of online business or commerce has given an increased cybersurface for financial fraud targeting digital cards. By analyzing the cardholder's transaction activity, using machine learning techniques, we can easily identify the fraudulent activity or transaction. In case of any abnormal behavior, the transaction is termed as fraudulent, and numerous techniques have been proposed by the researchers to prevent these, among these techniques machine learning-based techniques are the most popular. In this paper, we have surveyed the solution with respect to ML-based techniques, conducted experiments, and concluded our findings.

Keywords Digital card · Fraud detection · Financial transaction · Credit card fraud

S. Bonkoungou · N. R. Roy (✉) · N. H. A.-E. J. Ako

Department of Computer Science and Engineering, Sharda University, Noida, India
e-mail: niharranjanroy@gmail.com

S. Bonkoungou

e-mail: 2019006128.samiratou@ug.sharda.ac.in

N. H. A.-E. J. Ako

e-mail: 2019007784.nomel@ug.sharda.ac.in

A. Mishra (✉)

Center of Cyber Security and Cryptology, Sharda University, Noida, India
e-mail: alpnamishra@sharda.ac.in

1 Introduction

When a user's credit card also known as Digital Payment Card (DPC) is accessed without his/her permission, this transaction is termed as a fraudulent transaction. During COVID-19 the number of digital transactions increased many folds and so the number of fraudulent cases. People use their DPC for online purchases and save their information on different ecommerce platforms, granting attackers more facilities to illegally access the card details. These stolen details can be used for making fraudulent transaction causing loss of money to the owner. In the hope of avoiding such criminal incidents in the future, preventative measures against such acts should be implemented by reviewing and examining these transaction details or cases. It is suggested that while doing physical transactions the card owner must be vigilant and should not share the pin with anybody neither should handover the card to anybody. Card details can be stolen in physical transactions too, if the customer is not alert, whenever it happens, we can avoid losing money by blocking the card.

Among the most skilled DPC fraudsters are those who create fake and doctored cards and commit skimming fraud. In most of the cases, cardholder is unaware that he or she is the victim of fraud because the criminals are such an expert that they silently steal the details. The only way to discover such an online fraud is to investigate the spending habits of the cardholder. Another important observation is that most of the people are still ready for the digital card usage as they lack the knowledge of do's and don'ts, especially in the rural areas. They also need to be trained on what to do when something unexpected happens with the card. Second, most of the card providers provide two-factor authentication, which very few people enable. Third, unwanted options like international transactions should be disabled. If the card pin/has been shared with someone in the rarest case also outside the family, then they can block the card and request for a new card.

The rest of the paper is organized as below. Section 2 discusses in brief the existing machine learning techniques that are being used followed by a brief survey on the latest solutions proposed by different researchers. Section 3 focuses on experimental design, results, and discussions, and finally Sect. 4 concludes the work.

2 Related Works

2.1 *Classification of the Techniques and Application*

In the recent past, researchers have proposed many machine learning models for credit card fraud detection such as random forest (RF), artificial neural networks (ANNs) [1], logistic regression (LR) [2], support vector machine (SVM) [3], K-nearest neighbor (KNN) [4], hidden Markov model (HMM) [5], and decision tree (DT) [6]. This section briefly discusses these techniques before comparing them through simulation.

2.2 Literature Survey

Srivastava et al. [5] have proposed an HMM-based technique which exploits the spending behavior of the owner, if the transaction is not supported by large probability, then it is classified as fraudulent. The model learns from the past data of the cardholder and if it finds any difference in the spending behavior, then that transaction is flagged. They also proposed a pre-transaction verification page, where personal details from the user will be asked, if matched transaction will be allowed else terminated.

Aleskerov et al. [7] in their paper CARDWATH proposed a data mining-based application with user interface for many databases for credit card fraud detection. The model is powered by neural network-based classification. The system is built on a three-layer feed forward neural learning module and offers an interface to a number of databases. The results mentioned were obtained on synthetic database.

Kim and Kim [8] in their work focused on skewedness and highly overlapping nature of the transaction dataset. They used the fraud density from the historical data and used it as confidence value to generate weighted fraud score.

Singh and Narayan [9] in their survey, after study that HMM works on user's behavior when purchasing online, which will serve as a foundation for further developing the process and producing a more effective detection tool. Further work on this could include expanding the scope of the HMM to include additional facets of human behavior.

Jemima Jebaseeli et al. [10] and Kumar et al. [11] proposed a random forest-based technique claiming to have better performance than others. The performance of the random-tree-based and CART-based random forest models was studied by Xuan et al. [12]. In their experiment, they used a real-world B2C dataset of credit card transaction.

Makki et al. [13] explored and explained the fraud detection from perspective of big data analytics. They have also discussed statistical and machine learning-based techniques briefly explained all the major approaches and made a compilation of supervised and unsupervised techniques too. Taha and Malebary [14] proposed a technique based on light gradient boost and Bayesian-based hyperparameter optimization.

Mary and Priyadharsini [15] employed the support vector machine method. The suggested approach offers greater detection precision and is also scalable to handle huge transaction volumes. Future fraud detection methods will combine cost-based SVMs with efficient kernel functions to uncover fraud with lower error rates.

Malini and Pushpa [16] utilized KNN method to ascertain the anomaly of the target instance by oversampling and extracting the primary direction of the data. They have used an outlier detection technique to detect anomaly in a memory-constrained manner. They fed the algorithm with honest and false datasets during the training phase. KNN classifies any new instance using a similarity metric and uses Euclidean distance to determine the closest position of each arriving transaction to the new transaction before labeling it. If a transaction is received, the algorithm demonstrates that it is fraudulent.

Alenzi and Aljehane [17] proposed a logistic regression-based techniques for the fraud detection. The data that were used for this training were dirty, noisy, and valueless. They used two techniques to utilize to clean the data: the mean-dependent approach and the clustering-dependent method.

Using a decision tree, Save et al. [18] devised a system with six phases. Each stage has a connection to an actual data performance. The initial step is verifying the card numbers using Luhn's test. The degree of outlierness and address mismatch rules are used in the second and third tests to assess the variation of individual arriving transactions. Beginning hopes are the result of these two process steps. Using the advanced combination heuristic, step four of the method computes an overall belief. The expenditure history is examined in step five to identify the characteristics of lawful and illegitimate transactions. Sahin and Duman [19] proposed a SVM and DT-based technique for fraud detection. It was observed that when the size of the dataset was small, decision trees performed better than SVMs, but as the size of the dataset increased, SVM's accuracy equaled that of decision trees.

Bin Sulaiman et al. [20] proposed a hybrid solution using artificial neural networks (ANNs) in a federated learning framework. The experimental results were not available for analysis of effectiveness of this model. Bonkoungou et al. [21] have presented a recent and detailed survey of the latest contributions in this field. They have tabulated their findings and compared these techniques with their advantages and disadvantages. They also produced experimental results for comparison of RFC, LGR, and GBC. Cherif et al. [22] authors have compared and categorized major research in this area from 2014 to 2021 and categorized them.

3 Experimental Design, Results, and Discussions

3.1 Dataset Description

The used dataset contains 28 features numbered from V1 to V28, time, amount, and class, which was collected over 2 days in September 2013 by European cardholders. This dataset is highly imbalanced where we have 492 frauds out of 284,807 transactions. The training dataset was considered as 80% of the entire dataset and was balanced using SMOTE. The data was scaled using standard scaler before training.

The model is evaluated against popular evaluation metrics, including (Figs. 1, 2, 3, 4, 5 and 6):

$$\text{Detection rate (DR)/Accuracy (ACC)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2)$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (4)$$

$$\text{False Alarm Rate} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (5)$$

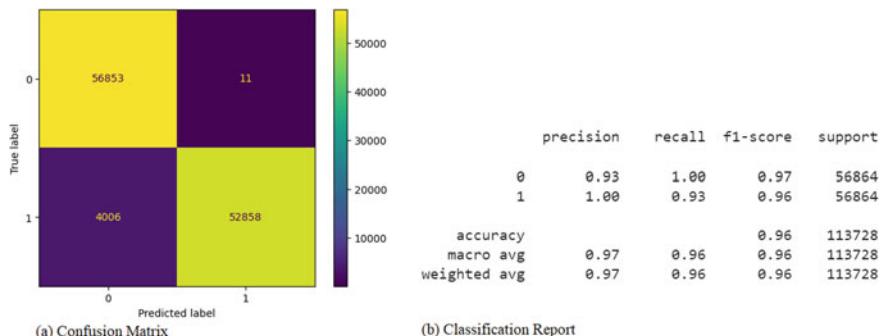


Fig. 1 Random forest

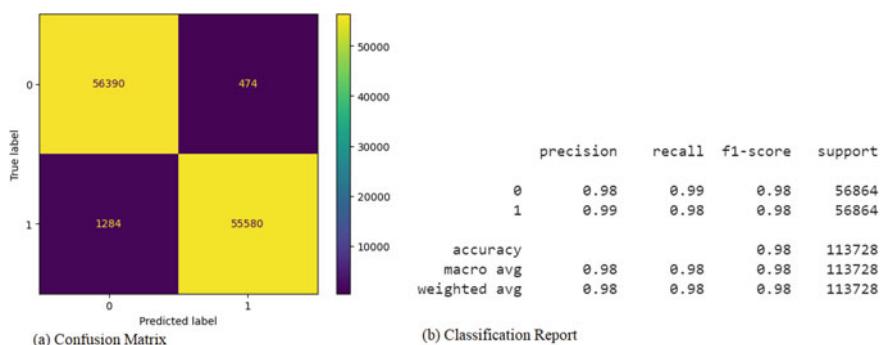
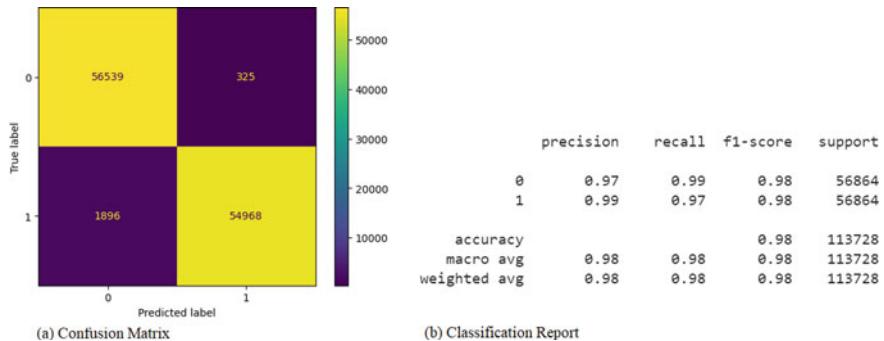
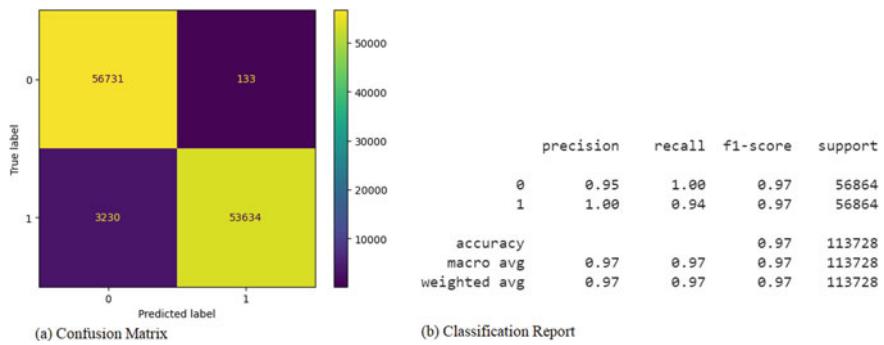
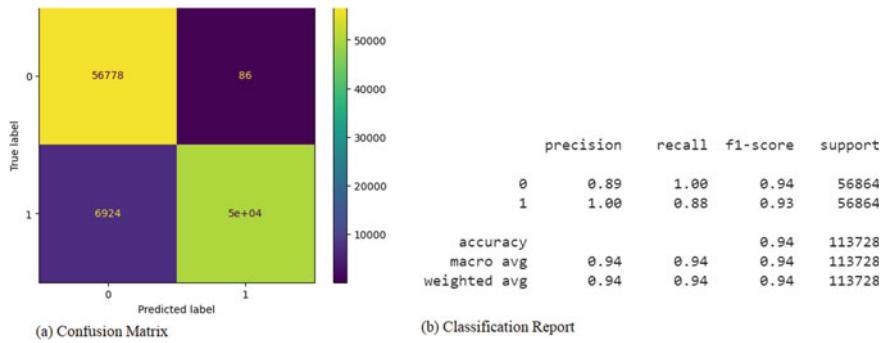
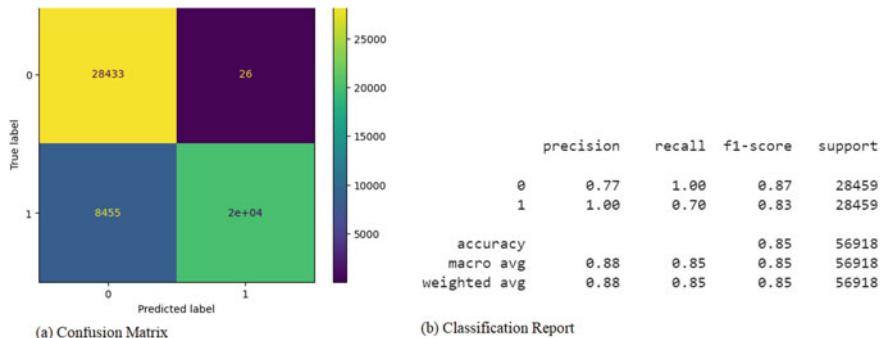


Fig. 2 Logistic regression

**Fig. 3** Gradient boost**Fig. 4** Support vector machine**Fig. 5** Decision tree

**Fig. 6** ANN**Table 1** Comparative analysis

Algorithms	Training accuracy	Testing accuracy
Random forest classifier (RFC)	1.0	0.9847
Logistic regression (LGR)	0.9797	0.9845
Gradient boosting classifier (GBC)	0.9888	0.9805
Support vector machine (SVM)	0.99718	0.9704
Decision tree (DT)	1	0.9383
Artificial neural network (ANN)	1	0.9646

On the basis of classification report, it is observed that LR has better precision and recall for both the classes (fraud and non-fraud) with sufficient support. In other cases, only one of the two cases has higher precision or recall with poor value for the other.

Performance of RFC, LR, GBC, SVM, DT, and ANN is compared on the basis of two parameters, i.e., training accuracy and testing accuracy, and is tabulated in Table 1.

From the table, it is clear that the RFC, ANN, and DT give the highest training accuracy. In the other hand, for testing accuracy, RFC and LR give higher performance.

4 Conclusion

In spite of several approaches proposed by researchers, none of them clearly identify all the cases of fraud. This occurs as a result of the extremely small percentage of transactions that are really fraudulent in nature. There is a need for the identification of fraudulent transactions in real-time with the highest accuracy. Although having excellent detection rates and accuracy, some systems, such as ANN and NB, are

exceedingly expensive to train. While some, such as KNN and SVM, show promising results with small datasets, they cannot handle massive datasets. On the basis of results obtained, we can say that logistic regression has better precision and recall for both the classes with good amount of support, whereas other classes have higher precision for one of the classes.

References

1. Chandrahas M, Dharmendra Lal G, Raghuraj S (2017) Credit card fraud detection using neural networks. *Int J Comput Sci* 4(7)
2. Sahin Y, Duman E (2011) Detecting credit card fraud by ANN and logistic regression. In: International symposium on innovations in intelligent systems and applications, pp 315–319
3. Demla N, Aggarwal A (2016) Credit card fraud detection using SVM and reduction of false alarms. *Int J Innov Eng Technol* 7
4. Razooqi T, Khurana P, Raahemifar K, Abhari A (2016) Credit card fraud detection using fuzzy logic and neural network. In: SpringSim
5. Srivastava A, Amlan K, Shamik S, Arun KM (2008) Credit card fraud detection using hidden Markov model. *IEEE Trans Dependable Secure Comput* 5
6. Patil S, Somavanshi H, Jyoti Gaikwad AD, Badgujar R (2014) Credit card fraud detection using decision tree induction algorithm. *Int J Innov Technol Explor Eng (IJITEE)* 4(6):2278–3075
7. Aleskerov E, Fieisleben B, Rao B (1997) CARDWATCH: a neural network based database mining system for credit card fraud detection. In: Proceedings of IEEE/IAFE: computational intelligence for financial engineering, pp 220–226
8. Kim M-J, Kim T-S (2002) A neural classifier with fraud density map for effective credit card fraud detection. In: Proceedings of international conference on intelligent data engineering and automated learning, pp 378–383
9. Singh A, Narayan D (2012) A survey on hidden Markov model for credit card fraud detection. *Int J Eng Adv Technol (IJEAT)* 1(3):49–52
10. Jemima Jebaseeli T, Venkatesan R, Ramalakshmi K (2021) Fraud detection for credit card transactions using random forest algorithm. In: Intelligence in big data technologies—beyond the hype: proceedings of ICBDCC 2019
11. Kumar MS, Soundarya V, Kavitha S, Keerthika E, Aswini E (2019) Credit card fraud detection using random forest algorithm. In: 3rd international conference on computing and communications technologies (ICCCT), pp 140–153
12. Xuan SALG, Li Z, Zheng L, Wang S, Jiang C (2018) Random forest for credit card fraud detection. In: 2018 IEEE 15th international conference on networking, sensing and control (ICNSC)
13. Makki S, Haque R, Taher Y, Assaghir Z, Ditzler G, Hadid M-S, Zeineddine H (2017) Fraud analysis approaches in the age of big data—a review of state of the art. In: IEEE 2nd international workshops on foundations and applications of self* systems (FAS* W)
14. Taha AA, Malebary SJ (2020) An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine. *IEEE Access* 8:25579–25587
15. Mary IM, Priyadharsini M (2021) Online transaction fraud detection system. In: International conference on advance computing and innovative technologies in engineering (ICACITE), pp 14–16
16. Malini N, Pushpa M (2017) Analysis on credit card fraud identification techniques based on KNN and outlier detection. In: International conference on advances in electrical, electronics, information, communication and bio-informatics (AEEICB)
17. Alenzi HZ, Aljehane NO (2020) Fraud detection in credit cards using logistic regression. *Int J Adv Comput Sci Appl* 11

18. Save P, Tiwarekar P, Jain KN, Mahyavanshi N (2017) A novel idea for credit card fraud detection using decision tree. *Int J Comput Appl* 161(13):6–9
19. Sahin Y, Duman E (2011) Detecting credit card fraud by decision trees and support vector machines. In: International multiconference of engineers and computer scientist 2011, Hong Kong
20. Bin Sulaiman R, Schetinin V, Sant P (2022) Review of machine learning approach on credit card fraud detection. *Hum-Cent Intell Syst* 2(1–2):55–68
21. Bonkoungou S, Roy NR, Ako NHA-E, Batra U (2023) Credit card fraud detection using ML: a survey. In: International conference on intelligent and innovative technologies in computing, electrical and electronics (IITCEE)
22. Cherif A, Badhib A, Ammar H, Alshehri S, Kalkatawi M, Imine A (2022) Credit card fraud detection in the era of disruptive technologies: a systematic review. *J King Saud Univ-Comput Inf Sci*

An Effective Model for Binary and Multi-classification Based on RFE and XGBoost Methods in Intrusion Detection System



Swikrati Dubey and Chetan Gupta

Abstract Incursion detection is an important network security tool that monitors and identifies any network intrusion. The present context is insufficient to manage network security threats, which with Internet use are growing fast. However, the analytical data generated regularly by computer networks are often extremely large size. This poses a significant problem for IDSs, who must look at all aspects of the data to find invasive trends. In this paper, an effective model has been presented for binary and multiclassification problems based on RFE and XGBoost methods in an intrusion detection system. We used a method called RFE to pick important features, and then we used a classifier called XGBoost to sort things into categories. We tested all of this on a dataset called NSL-KDD. These experiments have been performed with features (i.e., 9 and 5) and without features. The findings showed improved precision in the five and binary categorization with a reduced false alarm rate. For the 5 and binary categorization accordingly, the suggested method obtained an accuracy of 98, 75, and 100%.

Keywords Intrusion detection system · Feature selection · Recursive feature elimination · Classification · XGBoost classification · NSL-KDD

1 Introduction

Smart gadgets have been used rapidly in current years. This growth has led to a substantial rise in the field of network traffic. Network security is turning toward more difficult with this increase in data traffic [1]. Security techniques are also developing and changing, along with the increasing numbers and types of threats. Intrusion detection systems (IDSs) are very popular among these sophisticated and enhanced security techniques. Right now, intrusion detection systems (IDSs) are considered very important for keeping computer networks safe from both outside and inside

S. Dubey (✉) · C. Gupta
Department of CSE, SIRT, Bhopal, India
e-mail: swikratidubey123@gmail.com

threats [2]. Network intrusion detection systems (IDSs) keep an eye on bad things happening on computer networks. They watch out for any actions or behaviors that break the rules and could be harmful. In general, there are two main types of IDSs: one that looks for known attacks and bad behaviors, and another that looks for unusual and unknown attacks. The one focused on known attacks is good at finding them accurately, but it can't detect new attacks that it doesn't already know about. The one focused on unknown attacks can find new attacks easily, but it can sometimes get confused and mistake normal activity for an attack. Both types have their strengths and weaknesses. No matter which type of IDS is used, its performance is negatively affected when there is a lot of data flowing through the network. The reason for the poor performance of IDSs in high data traffic is not related to the IDS itself but has to do with duplicated information in the network. To improve the performance of IDSs in handling large amounts of data, various techniques have been proposed in recent years.

Basically, an IDS has to deal with a lot of data, which includes different patterns of network traffic. Each pattern is described by multiple characteristics and can be represented as a point in a space with many dimensions. A design may include unnecessary and duplicated characteristics which slow down training and test procedures or possibly have a more sophisticated mathematical effect on classification performance. However, it is useful to limit the number of functions as minimal as feasible in practice so that computer costs and the complexity of the construction of classification may be decreased. In addition, it makes it easier to view data, enhances modeling, predicts efficiency, and accelerates the categorization process by removing irrelevant characteristics. Therefore, dimensional decrease, like function extraction and the selection of features, was effectively used to address this issue in machine learning and data mining. Feature extraction methods try to transmit the input functionality to a novel feature set, whereas feature selection methods seek the most information from actual input data [3].

Feature selection (FS) has an important role in the development of an effective and efficient ML-based IDS. Since FS eliminates unnecessary properties, the ML IDS can train more quickly and accurately with low calculation [4]. Because of the quantity of information in current communication networks, ML-based IDSs need to provide an efficient and effective FS technique. In the existing work, differential evolution was employed in a solution space to identify the best resolution of features. Numerical function improvement has an important role in enhancing the objective function [5].

The rest of the paper is organized like this:

- Section 2 talks about the related work that has been done before.
- In Sect. 3, a new technique for better intrusion detection is introduced. This section also explains the results of experiments and provides information about the data used for testing.
- Finally, in Sect. 4, there are concluding remarks and suggestions for future work at the end of the paper.

2 Literature Review

Many studies have been conducted on intrusion detection systems (IDS) in the past, as mentioned in the literature. Some of these studies have been helpful in guiding me toward completing my own research paper.

Ferriyan et al. optimized selection feature for intrusion detection systems depending on genetic algorithm. For assessments, the NSL-KDD Cup 99 dataset has been used and dataset updated by examining the recent assaults, thus making the dataset more appropriate in the present circumstances. In these datasets, various classification was utilized and we discovered that random forest produced the best outcomes for classification rate and training duration. The findings revealed additionally that our two measures performed better with our optimized datasets features, and that the outcomes were mixed contrasted to those of the original features [4].

Zhang et al. suggested an efficient semi-supervised intrusion detection ML architecture. In particular, LapSVM utilizes the training model for Laplacian support vector machine and applies a functional selection technique to increase the performance of info. The trial findings using NSL-KDD show that our architecture can achieve high precision of 97.8%, while the falseness rate is 2% [6].

Umbarkar and Shukla attempted to extract a 41-feature subset without impairing IDS performance. This task utilized info gain (IG), gain ratio (GR), and feature selection methods based on correlation to decrease dimensionality. This study also suggested a method to reduce heuristic dimensionality to further enhance the effectiveness of the above-stated FS methods [7].

Hakim et al. the effect of the intrusion detection system feature selection has been noticed. The findings indicate that feature selection may substantially increase the performance of IDS, but it decreases the precision [8].

Almasoudy et al. [Base Paper], this article talks about a new way to make intrusion detection systems (IDS) better. They used a method called differential evolution to select only a few important features, which helps the system work well without too many things to consider. The primary concept is to use differential evolution and to assess certain characteristics from 41 features of NSL-KDD datasets by calculating they are precious using extreme machine learning (ML). The difference is continued until the minimal number of characteristics is obtained that fulfill a high level of exactness. The findings have demonstrated an improved detection rate in five and binary categorization, with a reduced false alarm rate. With a reduction in training and test time, the suggested system obtained a precision of 80.15 and 87.53% for 5 and binary categorization correspondingly [9].

Nandi et al., a feature-inspired learning environment has been created for network intrusion detection utilizing multiple machines learning categorization. At first, we found the most appropriate NSL-KDD dataset characteristics utilizing many techniques for the selection of features like GR, relief, and info gain procedures. The topmost important function has then been chosen from integrated pre-identified characteristics by utilizing our ensemble technique. The strength of our model was evaluated via the use of k-fold cross-validation. The precision of categorization is assessed

using various ML classifications. We have observed the highest precision of 99, 58% in the random forest categorization. Lastly, we evaluated the best outcome contrasted with other techniques of FS using similar classifiers in our ensemble approach [10].

Yu et al. suggested the novel technique of selection of features of two-stage dimension reduction (TSDR) and the NSL-KDD dataset validated. The KNN algorithm is used to check the efficiency of computation using the new feature selection procedure. In contrast to the complete feature computation, the precision rate is only decreased slightly [11].

Parimala and Kayalvizhi suggested a novel method for IDS to provide a safe wireless communication environment. In this article, we introduce a new method for picking out the most important features in a dataset. We combine two techniques called conditional random field (CRF) and spider monkey optimization (SMO) to do this. First, we use CRF to select the features that are useful. Then, we use SMO to finalize the selection and pick the most relevant features from the dataset. In addition, CNN is used to categorize the dataset and attacks as usual. Experiments for assessing the IDS suggested were carried out and shown to be superior in detecting precision, time, and false-positive rate [12].

3 Research Methodology

3.1 Problem Statement

One common problem faced by IDS organizations is that they lack an effective way to respond to incidents. There are several key issues that contribute to this problem, such as a high rate of false alarms, low detection rates, unbalanced datasets, and slow response times. IDSs generate many warnings, but not all of them are accurate or necessary. The alerts are typically based on IP addresses and ports, but it's difficult to determine which ones actually pose a network threat. The alerts from IDSs are not categorized based on their level of danger, making it challenging for security analysts to identify and address the risks. It is also time-consuming to distinguish between regular network activities. Additionally, classification problems further complicate the situation.

3.2 Proposed Methodology

To solve the problems mentioned earlier, a new model has been proposed for intrusion detection systems. This model uses recursive feature elimination (RFE) and XGBoost methods for better performance compared to existing methods. Here's how the proposed method works:

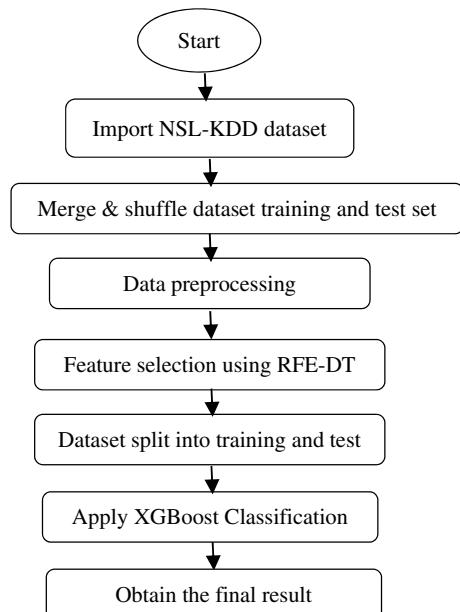
- The first step is to collect the intrusion data from a dataset called NSL-KDD.

- The collected dataset is divided into training and test sets, which are combined and shuffled.
- The data preprocessing step is important because the collected data contains symbolic features and high values. Symbolic features are converted into numeric features, and high values are normalized to a specific range using a normalizer.
- Recursive feature elimination is used with a decision tree classifier to select the most important features. This step calculates the importance of each feature.
- The dataset is then divided into separate train and test sets.
- The XGBoost classifier is used for classification. This classifier helps find the best results for both binary and multi-class classifications.
- Classification is performed using the selected features and also without selecting any features.
- The final selected features are 9 for multi-class classification and 5 for binary classification.
- This proposed model shows a higher detection rate for different characteristics in both binary and multi-class classifications, along with a reduced false alarm rate.

In summary, the suggested model improves intrusion detection by selecting the most relevant features and using the XGBoost classifier for accurate classification, resulting in better performance and reduced false alarms.

The suggested model is described in this section. The major stages include dataset preparation, feature selection through recursive removal, categorization using XGBoost, as illustrated in Fig. 1. We address these stages one by one in the following.

Fig. 1 Workflow of proposed model



3.2.1 Data Collection

We have utilized the NSL-KDD dataset that has to gather from the Kaggle data repository in the study. The above dataset includes chosen KDD dataset records to finish.

3.2.2 Data Preprocessing

Data preprocessing is like getting your data ready for a party. It involves cleaning and organizing the data so that it's easier to understand and work with. It's kind of like tidying up your room before your friends come over. By preprocessing the data, you make it more useful and easier to analyze. Usually, the highest and lowest dataset values differ much so that normalization of the data reduces the algorithm complexity in order to process it accordingly. The normalization of the data enables the categorization of algorithms to be appropriately advantageous under [13]. Normalizer applied in this instance. Normalization of the input values accelerates the training phase.

- **Normalizer**

A normalizer is like a special tool that helps make data consistent and easier to compare. It works by adjusting the values in each sample or row of the data so that they have the same scale or magnitude. This makes it easier to understand and analyze the data because everything is standardized. It's like making sure all the ingredients in a recipe are measured using the same unit of measurement, so you can easily compare and combine them.

Normalizer operates on the rows, not columns! The normalization L2 for each observation is used by default so that the row values have a unit standard. Unit norm with L2 implies that the total is equal to 1 if each element is squared and summed. Alternatively, normalization L1 (called taxicab or Manhattan) may be used rather than L2. Normalizer converts all characteristics into -1 and 1 values.

3.2.3 Feature Selection Using RFE

Feature selection means picking out the most important characteristics from a big list of options. It helps make things faster, simpler, and more accurate. It's like choosing only the key ingredients for a recipe to make it easier to cook and tastier in the end. This way, we can avoid using unnecessary or irrelevant information that might confuse the results.

- **Recursive Feature Elimination**

Recursive feature elimination, or short RFE, is a common selection method for features. RFE is prominent because it is simple to set up and use, it is beneficial

to choose those features (columns) that are more or less important to the objective variable prediction in the training dataset [14]. In simple terms, RFE classifies the features according to their significance and delivers top-notch features after the removal of least significant features, when the customer submits n. To select the best features, we need three things:

- An estimator, which is a special tool that helps us understand the importance of each feature.
- The number of features we want to choose. If we don't specify, it will select about half of them.
- The step size, which determines how many features are removed in each iteration. If the step is 1, it removes the same number of features each time. If the step is a decimal between 0 and 1, it removes a percentage of features each time.

Algorithm

- (1) RFE requires an estimate of supervised learning that is already suitable for all-featured use of data.
- (2) Consider the coefficient linked to each feature obtained from `coef_` or `feature_importances_attribute`. Basically, those coefficients are the same which we get after fitting the model on the dataset after minimizing the residuals. The value of these coefficients represents their importance with the target variable. The lesser value is regarded as the least significant characteristic with the lowest absolute coefficient value.
- (3) The least significant coefficient is then removed from the feature list and the prototype with the remainder of the features is reconstructed. At each iteration, you take the step parameter for a number of characteristics to decrease. It is preferable to remove 1 feature at a time since the values of other characteristics modify as we reconstruct the model.
- (4) At every iteration, the model is reconstructed and the least essential feature(s) are removed and the procedure is repeated until 2 features are shown. It then ranks on its removal time, depending on its characteristics. The deleted feature 1st and so on has the greatest rating. The latest n characteristics removed are classified as 1 [15].

3.2.4 Classification

Categorization is like sorting things into groups. In machine learning, we use this method to predict the category of a new sample based on a model we've trained. It's like learning from examples and then using that knowledge to assign labels to new data. The goal is to match the new data to the right category based on what we've seen before.

- **XGBoost Classification**

XGBoost is like a team of decision trees that work together. It's really good at solving big problems and making accurate predictions. It uses a special method

called gradient boosting to improve its performance. The goal is to make the trees work well without being too complicated. So, they make some changes to the way they measure how good the trees are.

This loss function may be incorporated into divided decision-making criteria, leading to a method for early stopping. Increased γ leads to simpler trees. The value of γ tells us how much the loss should decrease before deciding to split a part of the tree. Shrinkages are another thing that helps make XGBoost work better. It makes the steps smaller so that things happen more gradually. Also, there are other ways to make the trees simpler, like making them not too deep. This makes the trees train faster and take up less space.

Suggested Algo

I/P: NSL-KDD Dataset

O/P: Binary and Multi-Categorization Detection

Strategy

Step 1: Process the dataset

Step 2: Merge and shuffle collected trained and test data of NSL-KDD

Step 3: Apply Data Preprocessing

- Convert the symbolic features to numeric features
- Perform normalization using Normalizer in SK-learn

Step 4: Feature selection using RFE

- Take a supervised learning estimator
- Calculate the feature importance or coefficient of each attribute
- Fit the model
- Remove from the list of features the least relevant coefficient and reconstruct the model using the remaining characteristics
- Reconstruct the model and remove the least essential feature in every iteration(s)
- Repeat the procedure until it obtains 2 characteristics

Step 5: Categorize the dataset as 70% training and 30% test set

Step 6: Execute categorization by XGBoost classifier

- We improve a function by adding parts to it, based on a way to measure how much it's not working well, so that it becomes more effective
- Estimate the function of loss
- The loss function is incorporated inside the divided decision-making criteria leading to a pre-pruning method
- Use the parameter of regularization to decrease the additive expansion steps size

Step 7: Find Results.

4 Results Illustrations

In this section, we talk about the dataset we used, how we measured its efficiency, and the results we got from our experiments. We did all of this using a tool called Jupyter Notebook and the Python programming language.

4.1 Dataset

The dataset NSL-KDD is an enhanced KDD99 version. Not only does this address the duplicate KDD99 record issues, but the amount of traces in training and testing datasets is also suitable. The dataset for training consists of 21 distinct assaults of the 37 explained in a test dataset. The known kinds of assaults are those in the training dataset, whereas the new ones are the other assaults in the test dataset, i.e., not in the training datasets accessible. The kinds of attacks are divided into four groups: Dos, Probe, U2R, and R2L.

4.2 Performance Metrics

4.2.1 Accuracy

The precision is a way to measure how many data points were correctly predicted out of all the data points. It is usually given as a percentage. We calculate the precision using a specific formula, which is shown in Eq. 1.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (1)$$

False positive (FP) is when something is mistakenly identified as an attack when it's actually normal. False negative (FN) is when something is wrongly classified as normal, but it's actually an attack. True positive (TP) is when something is correctly identified as normal, and true negative (TN) is when something is correctly identified as an attack.

4.2.2 Precision

This measure helps us understand how likely a positive prediction is correct. We calculate it using a specific formula, which is shown in Eq. 2.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2)$$

4.2.3 Recall

This measure tells us how many things from the positive group were predicted correctly as positive. We calculate it using a special formula shown in Eq. 3.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

4.2.4 F1-Score

F1 measurement is an evaluation of the accuracy of the test. It calculates both accurately P and recalls R of the test to estimate the score.

4.2.5 Experimental Results

The major findings after using the suggested technique have been presented in this section.

The findings in Table 1 show how the approach suggested for chosen characteristics has enhanced its efficiency. The performance metrics are shown in Table 2 (Figs. 2, 3, 4 and 5).

Table 3 shows the highest precision and false alarm rate (FAR) in binary and 5 class features in contrast to baseline and technique suggested. The suggested approach obtained 98.02% precision with 9 features in 5 classes, 100% precision in 5 binary class features, 98.75% accuracy in 5 classes with 41 features, and 100% accuracy in

Table 1 Performance improvement after the suggested technique has been used

	Proposed method	Accuracy (%)	False alarm rate
41 features five class	98.75	0.004	
41 features binary class	100	0.0	
9 features five class	98.02	0.006	
5 feature binary class	100	0.0	

Table 2 Performance measures

	Five class 41					Five class 9					Binary class 5 features	
	Normal	DOS	Probe	R2L	U2R	Normal	DOS	Probe	R2L	U2R	Attack	Normal
Precision	0.99	1.00	0.97	0.97	0.79	0.98	0.99	0.95	0.97	0.70	1.00	1.00
Recall	0.99	1.00	0.97	0.82	0.53	0.99	0.99	0.94	0.77	0.19	1.00	1.00
F1-score	0.99	1.00	0.97	0.89	0.63	0.98	0.99	0.95	0.86	0.30	1.00	1.00

Fig. 2 Test confusion matrix for five class 9 features

```
Test Confusion Matrix :
[[22879  52  159  24  3]
 [ 85 15901  27  3  0]
 [119  113 3988  3  0]
 [235   28   10 899  0]
 [ 27    0    0   2 7]]
```

Fig. 3 Test confusion matrix for binary class 5 features

```
Test Confusion Matrix :
[[23117    0]
 [    0 21439]]
```

Fig. 4 Test confusion matrix for five class 41 features

```
Test Confusion Matrix :
[[22933  53  103  27  1]
 [ 24 15988   4  0  0]
 [103   28 4098  2  0]
 [188    0   13 959  4]
 [ 17    0    0   0 19]]
```

Fig. 5 Test confusion matrix for binary class 41 features

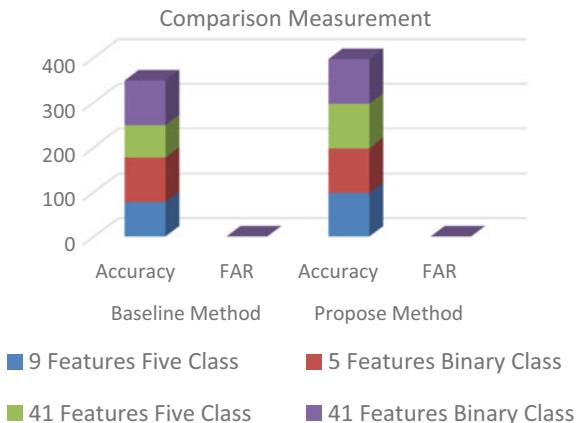
```
Binary Test Confusion Matrix :
[[23190    0]
 [    0 21366]]
```

41 binary class functionalities. The key point is that the suggested technique obtain has better precision and a lower rate than the baseline method of false alarm rate.

The comparison of the baseline method and the proposed method is shown in Fig. 6. The best accuracy of the suggested technique contrasted to baseline technique has presented in this graph.

Table 3 Accuracy and far in binary and five classification

Features	Baseline method		Propose method	
	Accuracy	FAR	Accuracy	FAR
9 features five class	77.67	0.074	98.02	0.006
5 features binary class	99.99	0.0005	100	0.0
41 features five class	71.07	0.096	98.75	0.004
41 features binary class	100	0.071	100	0.0

Fig. 6 Comparison graph

5 Conclusion

When we try to detect bad things happening on a computer network, we sometimes have a lot of information to deal with. But we can make it easier by choosing only the important information and ignoring the rest. This helps our computer work faster and saves its energy. Our goal is to make the computer work better and catch the bad things accurately. Many studies have shown that choosing the right information can help us do this. This article presents feature selection based on RFE and we applied XGBoost on a series of attributes for binary and five classes and experimental outcomes showed that this method is able to produce great precision of 98.02% with nine features chosen for five classes, 100% with 5 features chosen for the binary class and 98.75% with 41 in five classes and accuracy of 100% with 41 features selected in binary classes. Moreover, this decrease in the number of features decreases training and testing durations as well as false alarm rates.

In the future, we will focus on using other classification algorithms. Furthermore, the next study will concentrate on how to integrate validity measurements with every other if integrated into multi-target methods.

References

1. Sarvari S, Sani NFM, Hanapi ZM, Abdullah MT (2020) An efficient anomaly intrusion detection method with feature selection and evolutionary neural network. *IEEE Access* 8:70651–70663
2. Le T-T-H, Kim Y, Kim H (2019) Network intrusion detection based on novel feature selection model and various recurrent neural networks. *Appl Sci* 9:1392
3. Ambusaidi MA, He X, Tan Z, Nanda P, Lu LF, Nagar UT. A novel feature selection approach for intrusion detection data classification, p 8
4. Ferriyan A, Thamrin AH, Takeda K, Murai J (2017) Feature selection using genetic algorithm to improve classification in network intrusion detection system. In: 2017 international electronics

- symposium on knowledge creation and intelligent computing (IES-KCIC), pp 46–49. <https://doi.org/10.1109/KCIC.2017.8228458>
- 5. Tomar N (2015) A survey on data mining optimization techniques. *IJSTE Int J Sci Technol Eng* 2(06)
 - 6. Zhang X, Zhu P, Tian J, Zhang J (2017) An effective semi-supervised model for intrusion detection using feature selection based LapSVM. In: 2017 international conference on computer, information and telecommunication systems (CITS), pp 283–286. <https://doi.org/10.1109/CITS.2017.8035323>
 - 7. Umbarkar S, Shukla S (2018) Analysis of heuristic based feature reduction method in intrusion detection system. In: 2018 5th international conference on signal processing and integrated networks (SPIN), pp 717–720. <https://doi.org/10.1109/SPIN.2018.8474283>
 - 8. Hakim L, Fatma R, Novriandi (2019) Influence analysis of feature selection to network intrusion detection system performance using NSL-KDD dataset. In: 2019 international conference on computer science, information technology, and electrical engineering (ICOMITEE), pp 217–220. <https://doi.org/10.1109/ICOMITEE.2019.8920961>
 - 9. Almasoudy FH, Al-Yaseen WL, Idrees AK (2020) Differential evolution wrapper feature selection for intrusion detection system. *Procedia Comput Sci* 167:1230–1239. ISSN: 1877-0509. <https://doi.org/10.1016/j.procs.2020.03.438>
 - 10. Nandi S, Maity S, Das M (2020) NIDF: an ensemble-inspired feature learning framework for network intrusion detection. In: 2020 IEEE international women in engineering (WIE) conference on electrical and computer engineering (WIECON-ECE), pp 9–12. <https://doi.org/10.1109/WIECON-ECE52138.2020.9397993>
 - 11. Yu T, Liu Z, Liu Y, Wang H, Adilov N (2020) A new feature selection method for intrusion detection system dataset—TSDR method. In: 2020 16th international conference on computational intelligence and security (CIS), pp 362–365. <https://doi.org/10.1109/CIS52066.2020.00083>
 - 12. Parimala G, Kayalvizhi R (2021) An effective intrusion detection system for securing IoT using feature selection and deep learning. In: 2021 international conference on computer communication and informatics (ICCCI), pp 1–4. <https://doi.org/10.1109/ICCCI50826.2021.9402562>
 - 13. Chiba Z, Abghour N, Moussaid K, El Omri A, Rida M (2018) A novel architecture combined with optimal parameters for back propagation neural networks applied to anomaly network intrusion detection. *Comput Secur* 75:36–58
 - 14. Brownlee J (2020) Recursive feature elimination (RFE) for feature selection in Python. In: Data preparation
 - 15. Bentéjac C, Csörgő A, Martínez-Muñoz G (2019) A comparative analysis of XGBoost
 - 16. Ullah I, Mahmoud QH (2017) A filter-based feature selection model for anomaly-based intrusion detection systems. In: Proceedings of the 2017 IEEE international conference on big data (BIGDATA), Boston, MA, 11–14 Dec 2017
 - 17. Mishra P, Varadharajan V, Tupakula U, Pilli ES (2019) A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Commun Surv Tutor* 21:686–728
 - 18. Papamartzivanos D, Marmol FG, Kambourakis G (2019) Introducing deep learning self-adaptive misuse network intrusion detection system. *IEEE Access* 7:13546–13560
 - 19. Hua Y (2020) An efficient traffic classification scheme using embedded feature selection and lightGBM. In: Proceedings of the 2020 information communication technologies conference (ICTC), Nanjing, China, 29–31 May 2020
 - 20. Bindra N, Sood M (2020) Evaluating the impact of feature selection methods on the performance of the machine learning models in detecting DDoS attacks. *Rom J Inf Sci Technol* 23(3):250–261
 - 21. Revathi S, Malathi A (2013) A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. *Int J Eng Res Technol (IJERT)* 02(12)

ENCRYPTO: A Reliable and Efficient Mobile App for Password Management



Urmila Pilania, Manoj Kumar, Saurav Kumar Srivastava, Bhavika Dhingra, Lalit Adhana, and Riya Gaur

Abstract Passwords and weak secrets are frequently used as cryptographic keys in security protocols. Despite users' attempts to secure their data with system passwords or other measures, these protection schemes can create a false sense of security. Windows desktop passwords can be easily breached by third-party boot-up programmers, giving access to all files on a drive partition, and tools exist to crack passwords for most applications. A more reliable approach is to use an encryption utility to protect the contents of a storage device, which makes it incredibly difficult for unauthorized parties to access data files. This paper introduces ENCRYPTO, a proposed password manager application that employs the 128-bit advanced encryption standard (AES) algorithm. In this application, the user can edit the password at any time with the unique key. No one other than the developer can access the unique key which is present in the source code. A security analysis is also conducted to assess the strength of the proposed application and compared its time complexity analysis to existing standard password managers' applications such as UniPass, LastPass, and StrongPass. Overall, ENCRYPTO offers a secure and robust password management solution.

Keywords Advanced encryption algorithm · Password manager · Private key · Time complexity

1 Introduction

In today's world, people rely heavily on social networks and the Internet to communicate and share private information via e-mail, online banking, and other groups. Thus, securing personal information and passwords has become crucial. Users often

U. Pilania (✉) · M. Kumar · S. K. Srivastava · B. Dhingra · L. Adhana · R. Gaur
Computer Science and Technology, Manav Rachna University, Faridabad, India
e-mail: urmilapilania@gmail.com

M. Kumar
e-mail: manojattri003@gmail.com

have multiple social media accounts, making it difficult to remember all their passwords. Therefore, a password manager is needed to store passwords securely in a database. However, if the device is infected with malware, password managers can be hacked, and the master password can be captured, giving attackers access to all the data. To minimize the risk of passwords being hacked and shared, most password manager applications use security protocols and encryption algorithms [1, 2].

The security of information is affected by password exposure. If attackers obtain the user's account password, they can access the system, which poses a significant threat to the security of the information. Although several authentication methods, such as biometrics, are used, the most common authentication method is a password. Unfortunately, people often forget their passwords, leading to repeated attempts to recover them or request an OTP, which may not always be possible, especially in bad network conditions. Additionally, users often use simple passwords, such as birth dates, names, and phone numbers, which are easy to guess and compromise information security. To address these issues, this paper proposes an encryption application using the AES algorithm to secure personal information [3, 4].

The AES system is widely used to achieve privacy and confidentiality of personal data, and it is difficult for hackers to obtain the original message through this process [5]. It is a FIPS-approved cryptographic method that can be applied for information safety. AES security is ensured only if it is correctly executed, and key management is effective [6]. Although several algorithms can encrypt and decrypt sensitive data, no evidence has shown that any hacker can crack this algorithm. Since every age group has multiple social media accounts, saving passwords in notes is not secure. Thus, the proposed Android application will allow users to encrypt and store multiple account passwords easily and securely. The goal of this paper is to design an Android program that can encrypt passwords easily and securely.

2 Literature Review

A password manager application must be required by every web user in today's world. Remembering passwords for so many social and commercial sites is very difficult for all of us. In the research work [7], no server-side modifications were required by the application. In this work, the first password was encrypted and then it was stored on the secondary storage device. It provides a double layer of security for the password. It was made for the Mozilla web browser and Android users, but it could be run on any other web browser. Three protocols were used to enhance the security of the password. For the application to run efficiently, an Internet connection was required full-time. It also consumed a lot of electricity and took more time in the encryption process.

The password manager can store several passwords at a time by providing a guarantee of its security as well. As we all know, very smart hackers are also there who can hack passwords even from password manager applications. In their research [8], the authors designed a password manager application known as SplitPass. To

enhance the security of the application, the authors stored passwords at different locations and accessed passwords from different locations. Initially, all users sent the password to the server, but the sent password was not the complete password. In this paper, a secure socket layer was used to protect the password from hackers. The password manager was designed for Android users. The application was efficient at handling security issues, but it consumes more electricity and takes more time compared to the other applications.

In the proposed paper [9], five existing password manager applications were compared by the authors. All the password manager applications were web-based, and it has been concluded that even strong passwords could be detected by hackers. The authors found vulnerabilities in miscellaneous characteristics such as one-time passwords (OTP), bookmarks, and mutual passwords. The main reason for vulnerabilities was found to be different for different applications. The authors concluded that password manager applications were not safe and also suggested ways in which this application could be made robust. The authors suggested using any encryption algorithm before storing the password in the application.

With the help of password manager applications, users can select stronger and more arbitrary passwords as long as the user is not required to remember them. In this paper [10], the GPASS password manager has been developed by the authors for authentication purposes. In today's world, people of every age are using social/commercial sites to fulfill their daily needs. GPASS used a hash function to encode the PIN, and then the encoded PIN was stowed on the user account. So, due to encryption, it was very difficult to detect the password. Visually impaired and low-vision users face several issues during the authentication process.

In the paper [11], the UniPass password manager application was designed by the authors to help such types of people. They could access the web using some smart device. For testing of the UniPass application, six visually impaired people, along with four low-vision people, have suggested it was a great application. The authors also compared the UniPass application with LastPass and StrongPass. From the experimental results, it has been concluded that UniPass took less access time as compared to LastPass and StrongPass password manager applications.

Encoding processes perform important functions for the safety of original information from unauthorized persons. The AES method is the most widely used and is widely supported by hardware as well as software. The algorithm uses different key sizes like 128, 192, and 256 bits. Another noticeable thing regarding the AES algorithm is that the encryption and decryption processes are pretty similar, except for a few variations. Several key features of the AES algorithm were presented, and the performance was assessed using data from prior studies [12]. According to the results obtained from the research, AES is more secure compared to the other existing algorithms. The AES algorithm has some weaknesses as well, but they are minimal when compared to its strengths. The password manager has many disadvantages too, which can be a loss to users, which is why the encryption system was developed to prevent user data from public access. Some of these are [13–15]:

- If someone gets their hands on the master password, then it is very easy to access all the other passwords too.
- Password manager applications are a goal for hackers.
- It is not simple to log in using many devices.
- If passwords are stored on a computer and the computer is affected by malware, then the security of the password manager program may be compromised.
- Primary passwords are not properly encrypted except for the master password, which may cause leakage of security.

3 Motivation and Problem Statement

With the proliferation of technology, people of all ages are increasingly using online platforms and social media to meet their needs. However, each of these platforms and sites requires authentication before use, which can become difficult to manage with multiple accounts. To address this issue, we have developed an Android application that stores credentials using the AES algorithm. To ensure the robustness of our proposed application, we have tested it against various attacks. The proposed work has the following advantages:

- It can be implemented in both hardware and software.
- The longer key width of the AES algorithm makes it more resistant to hacking attempts.
- The user key is divided into two parts, one with the user and the other within the source code. Therefore, it is impossible to decode the credentials without authenticating within the application, even if the user-side key is obtained.
- The proposed solution is open-source.
- The AES algorithm is widely used and is considered a secure algorithm.
- The application is suitable for both wired and wireless communications.

4 Overview of Proposed ENCRYPTO Application

ENCRYPTO application's layout is designed using XML, which is a popular tool for native Android development. For the backend, Kotlin is utilized instead of Java as it is more efficient and provides better functionalities. SQLite database is used for the local database, which is common on Android devices. The Firebase real-time database, which is a Google product, is used for cloud storage as it is a popular relational database and easy to integrate. The application uses the Android cipher encryption library for encryption. The main purpose of the application is to store credentials using the AES encryption technique while ensuring the most secure and convenient access. The application offers features such as adding, showing, updating, deleting, creating, and managing cloud backup credentials. Additionally, users can generate a 20-letter random password with a combination of small/capital alphabets

and numbers. For the first-time sign-up, users must authenticate themselves using Google.

When the ENCRYPTO application is opened, the user is required to authenticate themselves using biometrics (fingerprint) for security purposes. Upon successful authentication, the user is redirected to the home screen, which displays all the application features. When adding credentials from the “Add Credentials” screen for the first time, the user is prompted to create a secret key. This key is then used to encrypt the password and store it, along with the username, in the local SQLite database of the device. To ensure that the user uses the same key for encrypting whenever storing credentials, the application tries to decrypt the previous password using the entered key. If it is decrypted successfully, the new credentials are encrypted before being added to the database. If decryption fails, the user is prompted to enter the correct key used for the first-time creation. The secret key is not stored anywhere, neither on the device nor in the cloud, to ensure maximum security. The complete flow of the application can be seen in Fig. 1.

When accessing particular credentials from the “Show Credentials” screen, users can search or tap on the specific account. The screen displays the username and an encrypted password. To view the password, the user must enter the secret key, which prompts a popup window. If the key is correct, the password is decrypted and displayed for 5 s, during which the user can copy it. After 5 s, the password is encrypted again. Additionally, users can create a cloud backup of their credentials to access them from anywhere, even if the device is lost or the application is uninstalled. The Firebase real-time database is used for cloud backup. ENCRYPTO’s password generator uses the Random() function of Kotlin to generate a 20-letter

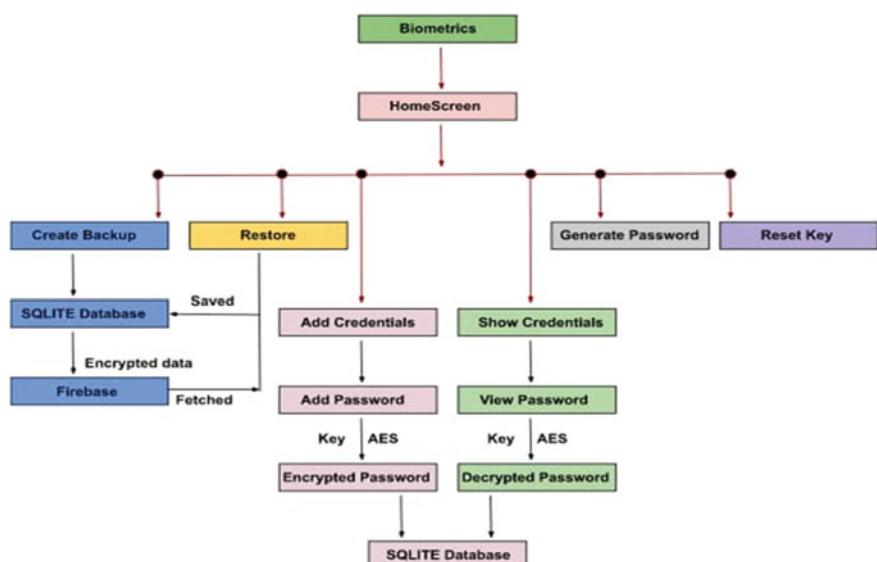


Fig. 1 Application flow

random password, selecting random letters from a collection of small and capital alphabets, symbols, and numbers.

This is a software application that is designed to securely manage and store user passwords. The AES encryption system is utilized to ensure maximum security and prevent unauthorized access. To access the stored passwords, a unique key is required, which is generated within the application itself. This application uniquely uses the AES encryption algorithm, specifically for password management. All passwords are stored within the device's local database and can only be accessed through a PIN or biometric authentication, ensuring that only authorized individuals can access the stored credentials.

5 Results and Discussion

The application requires biometric authentication (fingerprint) to access the user's managed passwords. Once authenticated, the user is redirected to the home screen, where they can access all the application features. When the user wants to add credentials, the application prompts the user to create a secret key for encrypting the password. The encrypted password, along with the username, is stored in the device's local database using the SQLite system. The secret key is not stored anywhere on the device or cloud, ensuring maximum security. The application ensures that the same key is used to encrypt the password every time by attempting to decrypt the previous password using the entered key. If successful, the credentials are encrypted and added to the database. If not, the user is prompted to enter the correct key. Figure 1 illustrates the complete flow of the application.

To view particular credentials, the user can search for or tap on the account. The application shows the username and an encrypted password. If the user wishes to view the password, a popup appears, requesting the secret key. If the key is correct, the password is decrypted and displayed for 5 s, during which the user can copy it. After 5 s, the password is encrypted back. The user can create a cloud backup if desired, allowing them to access their credentials from anywhere, even if the device is lost or the application is uninstalled. Firebase's real-time database is used for cloud backup. The application generates passwords using the Random() function of Kotlin, which selects random letters, symbols, and numbers from an array of small and capital alphabets.

The application is a software-based password manager that uses AES encryption to store and remember the user's passwords. The AES encryption system ensures maximum security, and the unique key generated by the application is required to access the managed passwords. Although the AES methodology was developed previously, the application uses it in a different approach to password management. Biometric authentication provides an additional layer of security, ensuring that no one else can access the user's credentials. Upon opening the application for the first time, the user encounters a biometric authentication screen displayed in Fig. 2a. After successfully logging in using their fingerprint, the user is redirected to the

home screen as shown in Fig. 2b. Here, the user can select from various options such as “Add Credentials”, “Show Credentials”, “Generate a Strong Password”, “Create a New Backup”, “Restore”, “Delete Existing Backup”, and “Reset Account and Key”. Upon selecting “Add Credentials”, the user is taken to the add credentials screen depicted in Fig. 2c. The user inputs the account type, username, and password, and upon clicking “ADD”, a popup requesting the secret key for encryption appears. The password is then encrypted and saved upon entering the correct secret key.

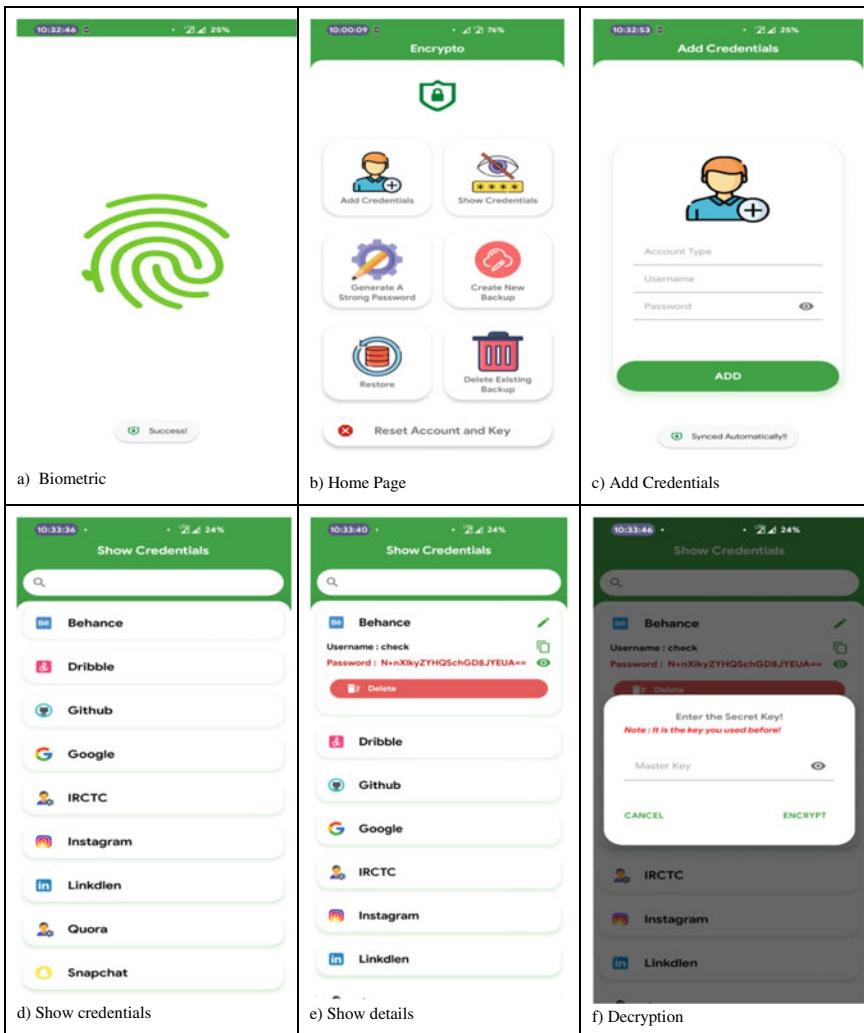


Fig. 2 Working of ENCRYPTO

Fig. 3 Non-rooted device

```
adb shell
phoenix:/ $ cd data/data
phoenix:/data/data $ ls
ls: .: Permission denied
```

Selecting “Show Credentials” leads the user to the show credentials screen as seen in Fig. 2d, which displays a list of accounts. Clicking on a particular account reveals the account details, including an encrypted password as shown in Fig. 2e. To decrypt the password, the user must enter the secret key known only to them, as seen in Fig. 2f. After five seconds, the password is encrypted back to ensure it is hidden from surrounding people.

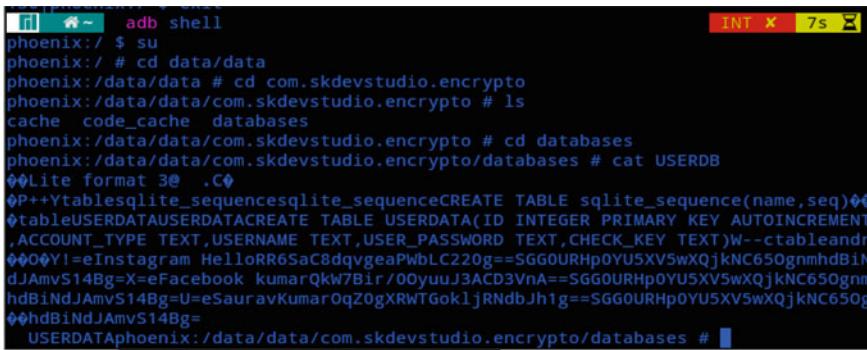
6 Security Analysis of ENCRYPTO

The application securely stores user credentials using the SQLite database, located within the data/data folder of the system. This database is inaccessible to normal users without root access. Rooting refers to the process of granting privileged control, or root access, over various Android subsystems. By rooting their device, users can access system files. However, if a device is not rooted, the SQLite database files cannot be accessed by the user. As a result, user data and credentials remain safe and can only be accessed through the application using biometric authentication. Figure 3 illustrates that a non-rooted device has no permission to access the system data.

In this scenario, attempting to access the data folder through the ADB shell resulted in a permission denied error, indicating that superuser permission is required. With root access, the data folder was found to be accessible using the ADB shell, as shown in Fig. 4. Upon locating the database file USERDB.DB inside the com.skdevstudio.encrypto/databases folder and attempting to access it using the cat command, a series of encrypted texts was found. However, it was not possible to decrypt these texts and extract the credentials, making it impossible for user data to be compromised.

7 Complexity Analysis and Comparison

While password manager applications are not new to the software world, existing applications such as LastPass, StrongPass, and UniPass are not completely secure [11]. For instance, LastPass allows anyone with mobile access to view the password by clicking on “Show Password”. Although enabling biometrics for these applications can offer some protection, the proposed application takes security a step further by incorporating a two-step security system consisting of biometrics and a secret key. The user of the system is not aware of the secret key, which is not stored in the



```

adb shell
phoenix:/ $ su
phoenix:/ # cd data/data
phoenix:/data/data # cd com.skdevstudio.encrypto
phoenix:/data/data/com.skdevstudio.encrypto # ls
cache code_cache databases
phoenix:/data/data/com.skdevstudio.encrypto # cd databases
phoenix:/data/data/com.skdevstudio.encrypto/databases # cat USERDB
@@Lite format 3@ .C@
@@P++Ytablessqlite_sequencesCREATE TABLE sqlite_sequence(name,seq)@@
@@tableUSERDATAUSERDATACREATE TABLE USERDATA(ID INTEGER PRIMARY KEY AUTOINCREMENT
,ACCOUNT_TYPE TEXT,USERNAME TEXT,USER_PASSWORD TEXT,CHECK_KEY TEXT)W--ctableandr
@@O@@Y!=eInstagram HelloRR6SaC8dqvgeaPwbLC220g==SGGOURRp0YU5XV5wXQjkNC650gnmhdbiN
dJAmvS14Bg=X=eFacebook kumarQkW7B1r/0OyuuJ3ACD3VnA==SGGOURRp0YU5XV5wXQjkNC650gnm
hdBiNdJAmvS14Bg=eSauravKumarOqZogXRWTGokljRNdbJh1g==SGGOURRp0YU5XV5wXQjkNC650g
@@hdBiNdJAmvS14Bg=
USERDATaphoenix:/data/data/com.skdevstudio.encrypto/databases #

```

Fig. 4 Rooted devices

device or anywhere else unlike other password manager applications. In the proposed research work, the password gets encrypted automatically after 5 s, ensuring that it's not exposed to anyone else. Additionally, if the user wants to hide the password, they can do so. Despite its advanced security features, the proposed application remains simple and focused on the purpose of managing and storing passwords.

Figure 5 displays a comparison of the proposed work with existing password manager applications based on the time required to perform various operations. ENCRYPTO outperforms the others in terms of the minimum time taken for creating an account and saving credentials, while LastPass takes the longest. Similarly, ENCRYPTO takes the least amount of time for logging into the application, while StrongPass takes the most. Time is also measured for login and saving credentials, with ENCRYPTO again performing the best. Even when logging in for the second time after creating an account, ENCRYPTO saves time compared to the others. Overall, the proposed application takes significantly less time than existing password manager applications.

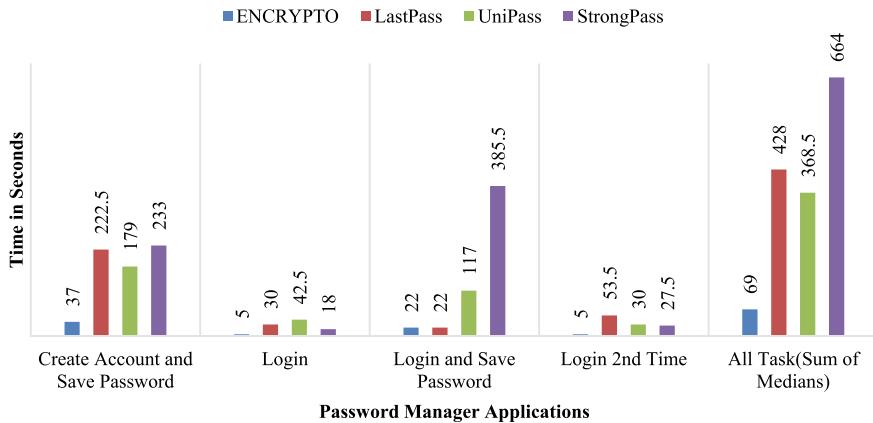


Fig. 5 Comparison of ENCRYPTO with existing applications

8 Conclusion

As cyber-attacks become increasingly common and passwords are no longer deemed secure enough to protect personal information, a solution is needed to manage the ever-growing list of passwords required for different websites and applications. To address this issue, we have developed a mobile-based password manager application utilizing AES encryption on 128 bits. With this application, users only need to remember one password to access all of their other passwords for various sites. The application provides an organized overview of all existing accounts with passwords stored on the left side of the screen, while account holder details are kept on the right side. Passwords are stored in encoded form and can be decoded by the user for viewing password details. We have conducted a security analysis to evaluate the effectiveness of our application and have also compared its time complexity with other standard password managers' applications such as UniPass, LastPass, and StrongPass. Overall, our proposed application provides a reliable and secure solution to password management.

References

1. Khande R, Ramaswami S, Naidu C, Patel N (2021) An effective mechanism for securing and managing password using AES-256 encryption & PBKDF2. Int J Electr Eng Technol (IJEET) 12(5):1–7
2. Alodhyani F, Theodorakopoulos G, Reinecke P (2020) Password managers—it's all about trust and transparency. Future Internet 12(11):189
3. Al-Aboosi AF, Broner M, Al-Aboosi FY (2022) Bingo: a semi-centralized password storage system. J Cybersecur Priv 2(3):444–465

4. Hatzivasilis G, Ioannidis S (2020) Password management: how secure is your login process? In: International workshop on model-driven simulation and training environments for cybersecurity, Sept 2020. Springer, Cham, pp 157–177
5. Dutson JW (2020) Managing two-factor authentication setup through password managers. Doctoral dissertation, Brigham Young University
6. Petrov M (2022) Android password managers and vault applications: data storage security issues identification. *J Inf Secur Appl* 67:103152
7. McCarney D, Barrera D, Clark J, Chiasson S, Van Oorschot PC (2012) Tapas: design, implementation, and usability evaluation of a password manager. In: Proceedings of the 28th annual computer security applications conference, Dec 2012, pp 89–98
8. Liu YT, Du D, Xia YB, Chen HB, Zang BY, Liang Z (2018) SplitPass: a mutually distrusting two-party password manager. *J Comput Sci Technol* 33(1):98–115
9. Li Z, He W, Akhawe D, Song D (2014) The {emperor's} new password manager: security analysis of web-based password managers. In: 23rd USENIX security symposium (USENIX security 14), pp 465–479
10. Bui T, Aura T (2017) GPASS: a password manager with group-based access control. In: Nordic conference on secure IT systems, Nov 2017. Springer, Cham, pp 229–244
11. Barbosa NM, Hayes J, Wang Y (2016) UniPass: design and evaluation of a smart device-based password manager for visually impaired users. In: Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing, Sept 2016, pp 49–60
12. Rewagad P, Pawar Y (2013) Use of digital signature with Diffie Hellman key exchange and AES encryption algorithm to enhance data security in cloud computing. In: 2013 international conference on communication systems and network technologies, pp 437–439. <https://doi.org/10.1109/CSNT.2013.97>
13. Zukri NHA, Rashid NAM, Awang N, Zulkifli ZA (2020) Agent-based encryption for password management application. In: Charting the sustainable future of ASEAN in science and technology. Springer, Singapore, pp 529–541
14. Yu F, Yin H (2021) A security analysis of the authentication mechanism of password managers. In: 2021 IEEE 21st international conference on communication technology (ICCT), Oct 2021. IEEE, pp 865–869
15. Pilania U, Tanwar R, Arora M, Kumar M (2022) Digitization through SNS: issues, challenges, and recommendations—a case study. In: Pattern recognition and data analysis with applications. Springer Nature Singapore, Singapore, pp 321–333

A Survey on Path Key Establishment



Krishan Kumar and Priyanka Ahlawat

Abstract Key management is a mechanism to generate, distribute, and manage the cryptographic keys, among the sensor nodes. During the distribution of keys, sometimes the nodes are not able to share the keys. In this case, the nodes use path establishment schemes to secure the transfer of the keys from a source to destination pair. Security of encryption used as path key further depends on the security of underlying keys of intermediate nodes. Sometimes, during a shared key discovery, nodes are not able to establish the symmetric key. How to establish a path between communicating nodes is still an issue that needs to be addressed seriously. Several issues have to be taken care of while designing a path key establishment scheme, namely overhead due to communication, resilience against node capture, storage, and computation. If we increase the security, other overhead may also increase; thus how we can balance these factors is very important. We aim to provide a comprehensive survey of different schemes. We have evaluated these schemes based on their merits and demerits.

Keywords Key management · Key pre-distribution · Path key · Wireless sensor network · Adversary

1 Introduction

Wireless sensor network (WSN) is a type of wireless network that consists of a large number of small, low-power, and inexpensive sensors that are distributed over a large area [1]. These sensors are equipped with wireless communication capabilities and can work together to sense and collect data about their surroundings. A network grid of various self-organizing, low-control, affordable, and useful sensor units makes up

K. Kumar (✉) · P. Ahlawat

National Institute of Technology, Kurukshetra, Kurukshetra, Haryana, India

e-mail: krishan_32113224@nitkkr.ac.in

P. Ahlawat

e-mail: priyankaahlawat@nitkkr.ac.in

a WSN. The sensor nodes are vulnerable to a range of physical security threats due to the use of these networks in smart buildings, battlefields, military systems, health care, and environmental tracking.

The data collected by these sensors is then transmitted to a central location for processing and analysis [1, 2]. WSNs have a wide range of applications, they are particularly useful in situations where it is difficult or impossible to install wired sensors, such as in remote or hazardous environments [3]. In a wireless sensor network (WSN), path key establishment refers to the process of creating a secure communication path among two or more nodes in the network [2], and this involves setting up a shared key, or secret, that can be used to encrypt and decrypt messages sent over the communication path. There are several approaches to path key establishment in WSNs [1], including symmetric key establishment and asymmetric key or the public key establishment. In symmetric key establishment, both nodes in the communication path share a secret key, which they use to encrypt and decrypt messages. This approach is relatively simple, but it requires that the secret key be shared among the nodes in a secure manner, which can be challenging in a WSN where nodes may be geographically dispersed and vulnerable to attacks [3]. In public key establishment, each node in the communication path has a pair of keys: a public key and a private key [1, 2]. The encryption is done with public key and decryption is performed by private key. This approach enables nodes to communicate safely without exchanging the secret key, but it requires more computational resources and may not be suitable for resource-constrained WSNs. Regardless of the approach used, it is important to ensure that the path key establishment process is secure and resistant to attacks, as the security of the entire communication path depends on it. So, for sensor network to maintain security there is a key management technique, this technique is composed into four phases [1, 3] namely network architecture: In distributed WSN, there is no member which has rich resources and all the nodes in this have equal capabilities. In hierarchical WSN, there are one or more sensors which are rich in respect of resources known as central station, and there is a hierarchical structure of sensor based on their capabilities. Communication Style: A secure one-to-one communication between the neighboring sensors requires common key between the two sensors (pairwise key). A secure multicast communication among the groups of nodes needs the group-wise key, and for broadcasting, we require the network-wise key. Pre-Key Distribution Phase: Here, we load the sensor with the keys, this loading of keys can be done by using probabilistic key distribution, deterministic key distribution, etc. Key Discovery Phase: In this phase, the sensors communicate with their neighboring nodes, find out with whom they are sharing the keys, and establish the secure communication with them. Path Key Establishment: After key discovery phase if the two nodes share the same key, then they both can connect to each other securely by using the key but what if they don't share the key than path key establishment comes in the picture where it establishes the connection between the two nodes using the intermediate nodes whether it can be pairwise key establishment or end-to-end key establishment [1].

In this literature survey, we have compare the different techniques that have been proposed for the path key establishment with their demerits and merits and evaluation parameters.

The paper is organized as follows: related schemes on path key establishment are given in Sect. 2. Further, Sect. 3 presents a comparative analysis of path key establishment for WSN. Section 4 presents future research directions. Lastly, in Sect. 5 we concluded our paper.

2 Work Done

The path key establishment schemes proposed for WSN are presented in this section. To avoid an adversary from physically compromising nodes, the topic of key establishment in sensor networks employing asymmetric key distribution within sensor nodes is examined by the authors in [4]. The use of the random key distribution is made possible by a few new techniques that are introduced for each sensor node. The Q-composite keys scheme and several path reinforcement techniques are employed. To strengthen the system's security, a pairwise key scheme is deployed. Similarly, in paper [5] uses mobile networks and distributed sensors with nodes with limited communication capabilities. Key management plans are designed to meet operational and safety requirements. Simple protocols are used for key sharing, path establishment, key revocation, and key re-keying in the probabilistic key distribution among nodes. Further, for node capture attacks in WSN, a safe hybrid key distribution mechanism is described by authors in [6]. Within threshold resistance, the robustness of the Q-composite approach is presented. Its goal is to protect the system from these threats.

Cheng et al. [7] use the existing pre-distribution scheme only for the connectivity of the network. There is a flaw in the existing scheme, which is before the deployment of sensors, the key rings are assigned to nodes and deployment, and they perform the key discovery phase. Here, the nodes find the other nodes which are sharing the common keys and then establish a secure connection among themselves, but due to the random distribution of keys, it may happen that any two secure paths can have the same key. So, they developed a method that establishes the pairwise key establishment again after deployment during the key discovery phase by doing an ex-or operation on the existing keys. This suggested technique, i.e., best in the case of resiliency, is suitable for large-size networks and also poses low computational and communication overhead.

In [8], the authors have presented an end-to-end pairwise key establishment scheme for wireless sensor networks (WSNs) that use multiple paths to boost the security and resilience of the network. The scheme generates and distributes pairwise keys between sensor nodes by using a multi-path routing protocol that enables for the selection of multiple disjoint paths among any two nodes in the network. The proposed scheme uses symmetric key cryptography and provides forward secrecy and resilience against node capture attacks. The authors evaluate the presented approach

through simulations, and the results display that the scheme achieves a high pairwise key establishment probability while reducing the communication cost as well as the usage of energy compared to previous schemes. The proposed scheme is particularly suitable for large-scale WSNs where the network topology changes frequently and where nodes may have limited resources.

Moreover, the authors in [9] have proposed a scheme for end-to-end encryption. It is required when two nodes don't share a common key. This paper finds more secure paths that reduce the key exposure problem by selecting a proxy node. A proxy node is a node that shares a common key with source and destination node, due to this proxy node will be at the distance of one hop from the source and destination. The authors have proposed two algorithms for finding the proxy node, which acts as the intermediate node. Firstly, the generation of m proxies, and lastly, LocalFlood. In this scheme, the key exposure problem is minimized as compared to previous schemes.

Further, Mehallegue et al. [10], in this paper, path key establishment is done using a novel algorithm which is better than the previous two algorithms in terms of security, computational power, etc. Similarly, in the paper [11], the authors have studied a dataset of the 26/11 Mumbai attacks and tried to find the pivotal node; a pivotal node is a node; by acquiring this, the whole terrorist network can be disturbed. In the proposed scheme, they calculated some features like page rank centrality, efficiency, and eigenvector with the help of these features, they calculated the pivotal node accurately. The suggested scheme can be used to design a system that can be used for counter-terrorism that focuses on finding a key node by which they can achieve maximum disruption in the network.

Ahlawat and Dave [12] discuss the use of random key management in wireless sensor networks (WSNs) for secure path key establishment. The authors propose two schemes for achieving this goal: a random key pre-distribution scheme and a random key establishment scheme. Both approaches are based on the use of random keys, which are generated by nodes in the network and shared with their neighboring nodes. The schemes are evaluated through simulations, which show that they can provide secure key establishment with low communication overhead. The paper also discusses the challenges and future research directions for secure key establishment in WSNs.

Based on the study of different schemes, it is observed how safely transferring the path key is an issue. Also, the minimization of intermediate nodes' number of encryption and decryption at each node needs to be addressed during the design of any path key establishment scheme. Further analysis of different schemes is very important for a feasible key management solution. Their merits and demerits are to be properly considered during the design of the ant path key.

In [13], Krishnapriya et al. proposed a scheme that uses a ranking mechanism to select the optimal route with minimal energy consumption and secure key exchange between nodes. Further, in [14], the authors have suggested a scheme based on a pool-hash mechanism that is designed to achieve better security and energy efficiency for WSN/IoT devices by reducing the overhead of key distribution and management.

Based on the literature survey, it is observed that a large number of path key establishment schemes are given by researchers to meet the different requirements

of network performance. Still the design of an efficient path key establishment is still an open research challenge that needs greater attention. How to provide a tradeoff between security and efficiency is also very important. It makes the study of current schemes very important in designing new schemes.

3 Comparative Analysis of Different Schemes

This section presents a comparative analysis of different schemes. We have analyzed along with their merits and demerits. The differences are summarized in Table 1. We have presented an overall summary of the various algorithms that have been proposed by the different authors for path key establishment and vulnerability evaluation schemes.

A proxy-based path key establishment scheme is a method for establishing a secure communication channel between two parties, using one or more intermediate nodes (proxies) to facilitate the exchange of key material. In this type of scheme, the two parties (A and B) first establish a secure connection with the proxy, using a mutually agreed upon method (such as a pre-shared key or public key infrastructure). Once this connection is established, A and B can then use the proxy to exchange key material and establish a secure communication channel between themselves. There are several benefits to using a proxy-based path key establishment scheme. One benefit is that it can provide increased security, as the key material is exchanged through a secure channel provided by the proxy, rather than being transmitted directly between A and B.

Additionally, a proxy-based scheme can provide anonymity, as the communication between A and B is mediated through the proxy, and the identity of the parties is not revealed. There are also some potential drawbacks to using a proxy-based path key establishment scheme. One potential drawback is that it can be more complex to set up and manage, as it requires the use of intermediate nodes (proxies) to facilitate the exchange of key material. Additionally, there may be concerns about the security of the proxy itself, as it acts as a point of control for the communication between A and B.

A friend-based path key establishment scheme can be used in a wireless sensor network (WSN) to establish secure communication channels between sensor nodes. In a WSN, sensor nodes are often deployed in remote or hostile environments where they may be vulnerable to attack. As a result, it is important to ensure that the communication between sensor nodes is secure to prevent unauthorized access to the network or the data being transmitted. In a WSN, a friend-based path key establishment scheme can be used to establish a secure communication channel between two sensor nodes that are not directly connected. For example, if Alice and Bob are two sensor nodes that are not directly connected, but both have a shared secret with Carol, a third sensor node that is connected to both Alice and Bob, then Carol can help establish a secure communication channel between Alice and Bob using the friend-based path key establishment scheme described above.

Table 1 Comparison of various path key establishment schemes

Author	Approaches	Focus	Advantage	Disadvantage
Cheng et al. [7]	Random key-based	Pairwise key establishment in static WSN	Supports large network size, has a minimal transmission and processing overhead	Path key exposure problem
Ling et al. [8]	Node-disjoint secure path	End-to-end key establishment	Addresses the “per-hop key exposure problem”	Intermediate nodes are more in this scheme
Li et al. [9]	Proxy-based approach	End-to-end key establishment	Security is enhanced because of multiple secured paths	Computational time is more
Gupta et al. [10]	Friend-based approach	Pairwise key establishment	Resilience against node capture	Energy expense is not shown
Singh et al. [11]	Preference ranking organization method for enrichment of evaluation	Finding the pivotal node	Can be used for counter-terrorism strategy	Difficult to implement
Ahlawat and Dave [12]	Matrix-based approach	Path key establishment using vulnerability evaluation matrix	Per-hop key exposure problem is addressed	Computational time is more for some vulnerability criteria
Dijiang et al. [15]	Node-disjoint multi-path key establishment scheme	Pairwise key establishment	Can tolerate up to faulty paths	The resilience to Byzantine attacks with minimal interactive communications involved is not addressed per-hop data exposure problem, computation overhead
Chan et al. [16]	PIKE: peer intermediaries for key establishment in sensor networks	Pairwise key establishment	It has uniform communication pattern for key establishment	May select a node which is more vulnerable
Mehallegue et al. [17]	Proxy-based approach	Pairwise key establishment	It produces shorter path lengths	May select a node which is more vulnerable

(continued)

Table 1 (continued)

Author	Approaches	Focus	Advantage	Disadvantage
Deng et al. [18]	Node-disjoint path	End-to-end key establishment	This scheme is flexible of trading transmission for lower information disclosure	Computational overhead
Sun et al. [19]	PKP: pairwise key pre-distribution	Pairwise key establishment	There is a unique pairwise key for each pair of sensors	Storage overhead
Huang et al. [20]	Secure path	Dynamic pairwise key establishment	During each session, this algorithm will create a unique straight pairwise key between them	Computing direct pairwise is time-consuming process
Yoo et al. [21]	Elliptic curve Diffie–Hellman (ECDH)	Pairwise key establishment	Select always random values to establish keys	Computational overhead
Yin et al. [22]	Node-disjoint path	Pairwise key establishment	Resist stop forwarding attacks and sniffing attacks	Communication and computational overhead
Deng et al. [23]	Cooperative secret delivery	Bridge nodes	Reduced secret disclosure probability low communication cost combats the eavesdropping easy challenge-response integrity check	Difficult to implement
Talawar et al. [24]	End-to-end key establishment during route discovery (E2-KDR)	End-to-end key establishment	The E2-KDR protocol offers a structure for secure routing that is independent of the secure routing protocol's dependence on key management	Computational overhead
Yilmaz et al. [25]	Used various routes in their SPREAD plan to send numerous shares of a secret	Shortest hop multi-path for WSN	The algorithm includes termination detection, fault-tolerance or load balance	How to generate and transfer multiple shares will incur overhead

This scheme can be particularly useful in WSNs where it is not possible to establish a direct communication channel between all sensor nodes, or where there are resource constraints that make it difficult to maintain a secure communication channel between all sensor nodes. By relying on a trusted third party, such as Carol, to help establish the key, the sensor nodes can still communicate securely without incurring the overhead of maintaining a direct secure communication channel. In wireless sensor networks (WSNs), disjoint path key establishment is a method for securely

establishing a shared secret key between two or more nodes in the network. The key is used to encrypt and decrypt messages transmitted between the nodes, providing confidentiality and integrity for communication. In a disjoint path key establishment scheme, the nodes first establish pairwise keys with their neighbors using a key establishment protocol, such as Diffie–Hellman. Then, these pairwise keys are used to establish a shared secret key between the nodes using a disjoint path. A disjoint path is a path through the network that does not share any nodes with any other path. By using a disjoint path, the nodes can be confident that the shared key has not been compromised by an attacker who has compromised any of the nodes on the path.

One example of a disjoint path key establishment scheme is the disjoint path key exchange (DPKX) protocol, which uses a combination of Diffie–Hellman and a disjoint path to establish a shared secret key between two nodes in a WSN. The DPKX protocol provides both confidentiality and integrity for communication between the nodes and is resistant to attacks such as man-in-the-middle. Overall, disjoint path key establishment schemes provide a secure and efficient way for nodes in a WSN to establish shared secret keys for secure communication.

Bridge Nodes-Based Path Key Establishment

In a bridge node-based path key establishment process, the bridge node acts as intermediary between the two devices and helps establish the secure connection by facilitating the exchange of keys and authentication information. The bridge node can also verify the authenticity of the keys and authentication information to ensure that the connection is secure.

4 Research Directions/Challenges

This section describes future research directions and challenges in proxy-based scheme is that it is difficult to find a proxy node, computation time of finding a proxy node is more. In this, the security of data is dependent on a single node that is chosen on the basis of key sharing. Challenges in friend-based path key establishment scheme are that finding friend nodes in big network is very complex, and this friend node is selected on the basis of if node is sharing the key with the destination or not. Challenge in disjoint path key establishment is that we can't find the disjoint paths always and computational time is more. In WSNs, path key establishment refers to the process of establishing a shared key between two nodes that are communicating over a path or route in the network. This shared key can then be used to secure the communications between the nodes using techniques such as symmetric-key encryption. There are several research directions and challenges in path key establishment in WSNs, including S. The key challenges in path key are summarized as

- (i) *Scalability*: One of the main challenges in path key establishment is ensuring that it scales to large numbers of nodes. This is important because WSNs can consist of hundreds or thousands of nodes, and it is not practical to establish individual keys between every pair of nodes.
- (ii) *Energy efficiency*: Establishing keys in a WSN can be resource-intensive, and it is important to ensure that the key establishment process is energy-efficient so as not to drain the batteries of the nodes too quickly.
- (iii) *Security*: It is important to ensure that the path key establishment process is secure and resistant to attacks, where an attacker alters or intercepts communications among two nodes.
- (iv) *Dynamic network topologies*: WSNs can have dynamic topologies, with nodes joining and leaving the network over time. This can make it challenging to establish and maintain keys in the network.
- (v) *Limited resources*: WSN nodes often have limited computing and communication resources, which can make it challenging to implement efficient key establishment protocols.
- (vi) *Robustness*: It is also important to develop path key establishment protocols that are robust and able to handle failures or disruptions in the network. This will ensure that the communication channels between nodes remain secure even in the face of challenges.
- (vii) *Number of intermediate nodes*: More the number of intermediate nodes more will be the node exposure problem and less will be the security.
- (viii) *Finding the disjoint paths*: It is difficult and sometimes you won't find any disjoint path.
- (ix) *Communication overhead*: When we want to enhance the security, we have to go for the path key establishment schemes which results in communication overhead.
- (x) *Key length*: Deciding key length is a challenge because increasing the key length increases security but also increases the computational overhead.
- (xi) *Repetition of path keys*: It results in low security, randomness will increase the security.
- (xii) *Decryption and encryption at each intermediate*: They are an overhead in the path key establishment.

5 Conclusion

In this paper, we have presented a comparative analysis of different schemes on path key establishment for WSN security. Path key establishment is a critical component of secure communication between devices. It enables the creation of a shared secret between the devices, which can be used to secure the communication channel. Tradeoff between different performance parameters is also an open research challenge. Overall, the field of path key establishment is an active and important area of

research that will continue to evolve in response to the changing landscape of cyber threats and the needs for secure communication.

References

1. Amutha J, Sharma S, Nagar J (2020) WSN strategies based on sensors, deployment, sensing models, coverage and energy efficiency: review, approaches and open issues. *Wireless Pers Commun* 111(2):1089–1115
2. Stankovic JA (2008) Wireless sensor networks. *Computer* 41(10):92–95
3. Popescu D, Stoican F, Stamatescu G, Chenaru O, Ichim L (2019) A survey of collaborative UAV–WSN systems for efficient monitoring. *Sensors* 19(21):4690
4. Chan H, Perrig A, Song D (2003) Random key predistribution schemes for sensor networks. In: 2003 symposium on security and privacy, May 2003. IEEE, pp 197–213
5. Eschenauer L, Gligor VD (2002) A key-management scheme for distributed sensor networks. In: Proceedings of the 9th ACM conference on computer and communications security, Nov 2002, pp 41–47
6. Ahlawat P, Dave M (2018) An attack resistant key predistribution scheme for wireless sensor networks. *J King Saud Univ-Comput Inf Sci*
7. Cheng Y, Agrawal DP (2005) Efficient pairwise key establishment and management in static wireless sensor networks. In: 2005 IEEE international conference on mobile adhoc and sensor systems conference. IEEE
8. Ling H, Znati T (2005) End-to-end pairwise key establishment using multi-path in wireless sensor network. In: GLOBECOM'05. IEEE global telecommunications conference, Nov 2005, vol 3. IEEE, 5 pp
9. Li G et al (2006) A robust on-demand path-key establishment framework via random key pre-distribution for wireless sensor networks. *EURASIP J Wireless Commun Netw* 2006:1–10
10. Gupta A, Kuri J, Nuggehalli P (2006) A new scheme for establishing pairwise keys for wireless sensor networks. In: International conference on distributed computing and networking. Springer, Berlin, Heidelberg
11. Singh S, Verma S, Tiwari A (2018) An innovative approach for identification of pivotal node in terrorist network using promethee method (an anti-terrorism approach). *Int J Eng Technol* 7(1):95–99
12. Ahlawat P, Dave M (2021) Secure path key establishment schemes based on random key management for WSN. *Proc Natl Acad Sci India Sect A* 91(3):555–567
13. Krishnapriya M, Angeline Prasanna G, Anbarasu S (2023) Rank-based energy-efficient key management routing for wireless sensor network-based IoT medical sensors. *Wireless Pers Commun* 1–22
14. Msolli A et al (2023) New key management scheme based on pool-hash for WSN and IoT. *J Inf Secur Appl* 73:103415
15. Huang D, Medhi D (2005) A byzantine resilient multi-path key establishment scheme and its robustness analysis for sensor networks. In: 19th IEEE international parallel and distributed processing symposium. IEEE
16. Chan H, Perrig A (2005) PIKE: peer intermediaries for key establishment in sensor networks. In: Proceedings IEEE 24th annual joint conference of the IEEE computer and communications societies, vol 1. IEEE
17. Mehallegue N, Bouridane A, Garcia E (2008) Efficient path key establishment for wireless sensor networks. *EURASIP J Wirel Commun Netw* 2008:1–9
18. Deng J, Han YS (2008) Multipath key establishment for wireless sensor networks using just-enough redundancy transmission. *IEEE Trans Dependable Secure Comput* 5(3):177–190
19. Sun H-M et al (2009) A pair-wise key establishment for wireless sensor networks. In: 2009 fifth international conference on intelligent information hiding and multimedia signal processing. IEEE

20. Huang J, Wang L (2010) Dynamic pairwise key establishment algorithm for wireless sensor networks. In: 2010 2nd international conference on education technology and computer, vol 3. IEEE
21. Yoo J, Lee Y, Won D (2011) An improved key establishment scheme for wireless sensor network. In: Proceedings of the 6th international conference on future internet technologies
22. Yin L et al (2012) Secure pairwise key establishment for key predistribution in wireless sensor networks. In: 2012 international conference on computer science and service system. IEEE
23. Deng J, Han YS (2013) Cooperative secret delivery in wireless sensor networks. *Int J Ad Hoc Ubiquitous Comput* 14(4):226–237
24. Talawar SH, Hansdah RC (2015) A protocol for end-to-end key establishment during route discovery in MANETs. In: 2015 IEEE 29th international conference on advanced information networking and applications. IEEE
25. Yilmaz O et al (2012) Shortest hop multipath algorithm for wireless sensor networks. *Comput Math Appl* 63(1):48–59

Detection of Phishing Link Using Different Machine Learning Techniques



Ashim Chaudhary, K. C. Krishna, Md Shadik, and Dharm Raj

Abstract The growing threat of phishing attacks on the Internet has raised concerns about the security of personal and organizational information. The use of machine learning techniques has emerged as a potential remedy for detecting and preventing phishing attacks. In this study, we compare and contrast different types of analysis of machine learning algorithms for predicting phishing websites. We measured how well different models did by looking at their accuracy, precision, recall, and *F1*-score metrics. In terms of classification accuracy, we discovered that ensemble learning techniques like random forest and decision tree outperformed other models. We also analyzed the feature importance of each algorithm to identify the most discriminative features for predicting phishing websites. Our discoveries offer significant perspectives on the topic of the effectiveness of machine learning approaches in detecting phishing attacks, which can help to enhance the security of online transactions and prevent data breaches.

Keywords Phishing URL · Phishing URL detection · Machine learning · URL classification

A. Chaudhary (✉) · K. C. Krishna · M. Shadik · D. Raj
Sharda University, Greater Noida, UP, India
e-mail: 2019000163.ashim@ug.sharda.ac.in

K. C. Krishna
e-mail: 2019001516.krishna@ug.sharda.ac.in

M. Shadik
e-mail: 2019007473.md@ug.sharda.ac.in

D. Raj
e-mail: dharm.raj@sharda.ac.in

1 Introduction

Phishing attacks have become more advanced in recent times, making it challenging for individuals and organizations to identify and prevent them. These attacks use deceitful webpages or links that appear authentic to trick people into divulging confidential data like credit card details, login credentials, or personal information.

As per the FBI crime report for 2021, phishing emerged as the most prevalent type of cyber-attack, with incidents rising from 241,342 in 2020 to 323,972 in 2021 [1]. During phishing attacks, cybercriminals create fake versions of well-known websites and entice unsuspecting individuals to visit these sites. They then employ fraudulent strategies to coerce individuals into giving private data, including passwords, banking information, credit card numbers, and other sensitive credentials [2].

Cybercriminals tend to cast a wide net when targeting victims in an effort to gain access to significant amounts of data or money. According to research conducted by Kaspersky, the cost of a successful attack in 2019 ranged from \$108,000 to \$1.4 billion, depending on the severity of the breach. Then again, the global expenditure on security-related products and services is estimated to be approximately \$124 billion annually [3]. As a result, the need for effective techniques to detect phishing attacks has become more crucial than ever.

The employment of machine learning methods has surfaced as a potential remedy for identifying and thwarting phishing attacks. Such algorithms possess the capability to scrutinize vast volumes of data and discern patterns that are suggestive of phishing behavior.

The objective of this study is to compare the accuracy, precision, recall, and *F1*-score characteristics of different machine learning models' performance. We also aim to analyze the feature importance of each algorithm to identify the most discriminative features for predicting phishing links.

Various studies have investigated the effectiveness of machine learning techniques in detecting phishing attempts. Alshamrani et al. [4] used a variety of techniques, including Naive Bayes, random forest, and support vector machine (SVM), for instance in their research on phishing email detection. Their findings indicated that SVM outperformed the other models in terms of accuracy, precision, recall, and *F1*-score.

In this study, by contrasting a variety of the current literature, we add to it. machine learning models for predicting phishing links. Specifically, we investigate the performance of random forest, decision tree, SVM, and k-NN algorithms. We also analyze the feature importance of each algorithm to identify the most relevant features for predicting phishing links.

The rest of this essay is organized as follows: A brief summary of relevant work in machine learning-based phishing detection is given in Sect. 2. Section 3 describes the common approaches to detecting phishing URLs. Section 4 presents the anatomy of phishing URLs. Sections 5 and 6 discusses more URL-based approaches and features of the URL. Section 7 explains the model we used for the detection of phishing links. Section 8 presents a discussion of the experimental results and findings. Finally,

Sect. 9 concludes the paper with a summary of the contributions and future research directions.

2 Literature Survey

Deep learning approaches, such as LSTM and CNN, have been employed in several works for detecting phishing websites. The advantage of using deep learning methods is their ability to automatically learn complex features and incorporate large volumes of data. Yang et al. [5] used LSTM and RNN algorithms for detecting phishing attacks and achieved an accuracy of 99.1% on the Yahoo and PhishTank datasets. A character-level CNN was utilized by another study to learn the sequential information of phishing URLs and achieved an accuracy of 95.02% on the given dataset and better accuracy than other phishing URL models on benchmark datasets. Shweta et al. [6] applied CNN to automatically extract features from URLs and achieved an accuracy of 98.00% on a dataset containing both phishing and legitimate URLs. These works demonstrate the potential of deep learning in detecting phishing websites and suggest that it could be a promising approach to combat the phishing problem.

The use of a faster R-CNN logo identification method in combination with a feature pyramid network (FPN) was suggested as a means of detecting two-dimensional code phishing attempts. This approach was tested on the FlickrLogos-32 dataset [7].

A new anti-phishing system was introduced, which employs LSTM and CNN and utilizes a dataset consisting of about 200,000 URLs from VirusTotal, PhishTank, and Yandex search API. This system was able to attain an accuracy of 96% and a precision of 97%.

The paper by Chawla [8] addresses the problem of phishing, a cybercrime where deceptive website links are used to acquire sensitive information. The research introduces a machine learning-based approach, specifically the Max Vote Classifier, for early detection of phishing websites. The focus is on protecting user data, with the proposal of implementing the classifier in a web application for URL-based phishing detection.

Fuzzy rough set (FRS) was used to choose the most useful features in phishing detection, and a maximum *F*-measure of 95% was achieved using random forest classification [9].

Using confidence-weighted classification along with content-based phishing URL detection, a dynamic and expandable system for the identification of both existing and new phishing domains was created.

Wong et al. [10] are concerned with the alarmingly high frequency of web phishing attempts and the rise of website cloning as a method of avoiding anti-phishing detection systems. They discovered that none of the security vendors put to the test were able to recognize the cloned sites after assembling and analyzing various cloning methodologies. The authors offer four suggestions to reduce the dangers of cloning

assaults for site developers and the anti-phishing research community, highlighting the urgent need for more effective detectors.

A method for generating suspicious domain names was developed to forecast potential phishing websites from authentic brand domain names and score and grade suspects by determining several indexes to identify phishing websites.

Phishing attacks have become a significant threat to online users, and various methods have been proposed to detect such attacks. The blacklist-based approach is a conventional technique that involves flagging malicious URLs and storing them in a separate database [11]. However, this approach lacks the ability to generalize to newly developing malicious domains, and the storage and updating of data can be challenging. In contrast, the whitelist-based approach lists all trustworthy websites and warns the user of any non-listed websites. However, creating a global whitelist of the entire World Wide Web is impractical. The content-based approach assesses web pages' resource identification and component access protocols, using two unique feature sets derived from the page document object model and the contents of web pages. By comparing the purported site to the current collection of resources, the visual similarity-based technique detects phishing sites, including logos, screenshots, and favicons. Finally, the URL-based approach analyzes hyperlinks from the website's source code to identify phishing websites, using features such as total hyperlinks, external hyperlinks, and login form links. Other approaches include the deep vision-based approach, rule-based tree model, meta-heuristic-based approach, and DNS-based approach, with ongoing research efforts to make phishing detection more robust.

3 Different Phishing Link Detection Approaches

Phishing refers to a form of cybercrime that employs deceitful techniques to acquire sensitive information, including login details, credit card numbers, and other personal data. These attacks typically utilize fraudulent emails, social media accounts, or websites that are deliberately crafted to deceive users into disclosing their confidential information.

To detect phishing links, there are several approaches that can be used. Here are some of the most common ones:

i. **Blacklist-Based Approach**

The blacklist-based approach is a common method used to detect phishing links. It involves comparing the suspicious link to a list of known malicious websites or URLs. If the link matches a URL on the blacklist, it is flagged as a phishing link and blocked from being accessed.

The blacklist approach is effective in detecting known phishing links that have already been reported and added to the blacklist. However, it has limitations in identifying new or previously unknown phishing links. Malicious actors often create

new phishing links or modify existing ones to avoid detection, making it challenging to maintain an up-to-date blacklist.

One way to address this limitation is to use a dynamic blacklist approach that updates in real time based on user reports and other sources of information. This approach enables faster detection and response to new phishing attacks. Another way is to combine the blacklist approach with other methods such as machine learning or content-based analysis to improve detection accuracy.

However, relying solely on a blacklist-based approach can give a false sense of security. Hackers can easily circumvent this approach by using domain generation algorithms (DGAs) to generate new domains that are not yet blacklisted. Additionally, legitimate websites can also end up on the blacklist due to false positives, which can cause inconvenience for users.

PhishNet [12] uses a method called blacklisting to predict phishing attempts. It uses five different techniques to identify new phishing websites, including analyzing the top-level domain, IP address, brand name, query string, and directory structure. When applied to large amounts of data, it is able to accurately detect 95% of phishing attempts while producing a 3% false positive rate. However, it is not effective in detecting phishing sites that appear suddenly without warning.

In summary, a blacklist-based approach is a useful tool for detecting known phishing links, but it should be used in combination with other approaches to increase the chances of detecting new or previously unknown phishing links. It is important to regularly update and verify the blacklist and to educate users on the dangers of phishing attacks.

ii. Whitelist-Based Approach

A whitelist-based approach is a security measure used to control access to certain resources or services. It involves creating a list of approved users, devices, or applications that are authorized to access specific resources while blocking all other users, devices, or applications by default.

In this approach, only authorized entities are allowed to access certain resources or services, while all others are prevented from doing so. The whitelist includes a list of trusted organizations to whom access has been given, and any entity that is not on the list is automatically denied access.

The main advantage of using a whitelist-based approach is that it provides a high level of security by only allowing authorized entities to access resources or services. This approach is particularly useful for organizations that deal with sensitive data or have strict compliance requirements.

However, a whitelist-based approach can be challenging to implement, as it requires careful consideration of which entities should be allowed access and which should be denied. It can also be difficult to maintain and update the whitelist, especially in larger organizations with many users, devices, and applications.

Moreover, this approach may also result in a reduced level of flexibility, as only approved entities can access certain resources or services, which can limit the ability to introduce new technologies or processes.

In conclusion, a whitelist-based approach is an effective security measure that can provide high levels of protection for organizations and their sensitive data. However, it requires careful planning and implementation to ensure that the whitelist is comprehensive and up-to-date and that it does not limit the organization's ability to operate efficiently.

iii. Content-Based Approach

The content-based approach is a common technique used to detect phishing links. It involves analyzing the content of a website or email to determine if it contains characteristics of phishing such as suspicious URLs, spelling errors, and unusual content. This approach is based on the fact that phishing attacks often use fake websites and emails that are designed to look like legitimate ones. By analyzing the content of these messages, the content-based approach can identify phishing links and alert users to their potential danger.

One of the advantages of the content-based approach is that it can detect new and unknown phishing links. Unlike signature-based detection, which relies on a database of known phishing links, the content-based approach can identify new phishing links that have not yet been reported. This makes it a valuable tool in the fight against phishing attacks.

To implement the content-based approach, various techniques can be used, such as machine learning algorithms, natural language processing (NLP), and heuristics. NLP involves analyzing the language used in a message to identify suspicious phrases or patterns. Machine learning algorithms can be trained to recognize phishing links based on various factors such as URL structure, content, and metadata. Heuristics involve using rules and patterns to identify suspicious links based on certain characteristics.

However, the content-based approach also has some limitations. For example, it may not work well against phishing attacks that employ social engineering techniques to deceive users into providing personal information and giving away their personal information. Moreover, the accuracy of the content-based approach depends on the quality of the algorithms and data used. If the algorithms or data are not comprehensive or up-to-date, the approach may not be effective.

In conclusion, a content-based approach is a valuable tool for detecting phishing links. It can identify new and unknown phishing links and alert users to their potential danger. However, it should be used in conjunction with other approaches, such as user education and reputation-based detection, to provide a comprehensive defense against phishing attacks.

iv. URL-Based Approach

The URL-based approach to phishing link detection involves analyzing the characteristics of the URL to determine whether it is likely to be a phishing link. This strategy is founded on the fact that many phishing links have URLs that are similar to legitimate websites, but with small differences that can be difficult to detect.

One characteristic that is often analyzed in the URL is the domain name. Phishing links may use domain names that are similar to legitimate websites but with minor

variations, such as misspelled words or added characters. For example, a phishing link may use a domain name such as “[paypa1.com](#)” instead of the legitimate domain name “[paypal.com](#)”. By analyzing the domain name, the URL-based approach can detect these variations and flag the link as potentially fraudulent.

Another characteristic that is often analyzed in the URL is the use of subdomains. Phishing links may use subdomains that are similar to legitimate websites but with minor variations. For example, a phishing link may use a subdomain such as “[login.paypal.com](#)” instead of the legitimate subdomain “[www.paypal.com](#)”. By analyzing the subdomains, the URL-based approach can detect these variations and flag the link as potentially fraudulent.

Additionally, the URL-based approach can analyze the path of the URL to determine whether it is likely to be a phishing link. Phishing links may have paths that are similar to legitimate websites but with minor variations. For example, a phishing link may have a path such as “/login” instead of the legitimate path “/sign in”. By analyzing the path, the URL-based approach can detect these variations and flag the link as potentially fraudulent.

Overall, the URL-based approach to phishing link detection is an effective way to identify fraudulent links that are designed to mimic legitimate websites. By analyzing the characteristics of the URL, this approach can detect minor variations that may be difficult for users to identify on their own. This approach can be used in combination with other approaches, such as content-based or machine learning-based detection, to provide a comprehensive approach to phishing link detection.

v. Other Approaches that Are Commonly Used

1. Signature-based detection: This approach involves comparing the suspicious link with a database of known phishing links. If the link matches a known phishing link, it is flagged as fraudulent. However, this approach is limited in its effectiveness since it can only detect previously identified phishing links.
2. Reputation-based detection: This approach involves analyzing the reputation of the sender and the website in question. If the sender or website is deemed untrustworthy or has a poor reputation, the link is flagged as potentially fraudulent.
3. Content-based detection: This approach involves analyzing the content of the link to determine if it contains phishing characteristics such as spelling errors, suspicious keywords, or unusual URLs. This method is more effective than signature-based detection since it can identify new phishing links that have not yet been reported.
4. Machine learning-based detection: This approach involves using machine learning algorithms to identify phishing links based on patterns and characteristics found in known phishing links. Machine learning algorithms can be trained to recognize phishing links based on various factors such as URL structure, content, and metadata.
5. User-based detection: This approach involves educating users about the characteristics of phishing links and encouraging them to be cautious when clicking on suspicious links. Users can be trained to recognize phishing links

by looking for signs such as incorrect spelling, unusual URLs, or suspicious email addresses.

In conclusion, detecting phishing links is an essential step in protecting against cybercrime. By using a combination of these approaches, individuals and organizations can increase their chances of identifying and avoiding fraudulent links.

4 Anatomy of Phishing URL

We employ a technique to identify phishing URLs that are based on analyzing the URL itself. To effectively detect phishing URLs, it is necessary to understand the structure of such URLs. Phishing URLs typically contain certain keywords or characters that mimic legitimate terms but are misspelled, special characters that redirect users to other websites, shortened URLs, or terms that appear trustworthy but are actually fraudulent. Phishing attacks using URLs can be successful because users may mistake the fraudulent website for a legitimate one and provide their login credentials. Figure 1 illustrates how phishing URLs work in practice. Phishing URLs are detected using a URL-based approach that involves understanding the structure of a phishing URL. These URLs use tactics such as misspelled terms, redirecting characters, URL shorteners, trustworthy terms, and malicious files to trick users into giving away their credentials to attackers who pose as legitimate websites.

The image shows a phisher, represented as a person, who is connected to both a victim and a hacker site. The reversible arrows indicate the flow of information and actions. One arrow points from the phisher to the hacker site, where a phishing kit is uploaded to collect the victim's credentials. The other arrow signifies potential profit for the phisher resulting from the successful phishing attempt. Additionally, there is

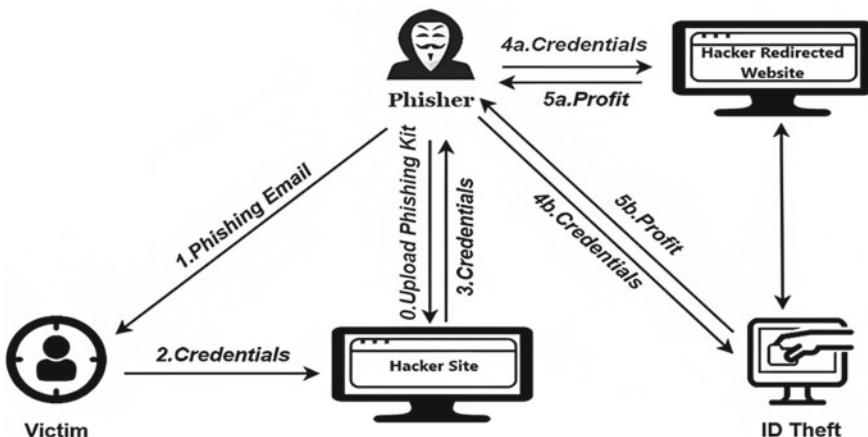


Fig. 1 Phishing attack scenario

a reversible arrow connecting the phisher to ID theft, suggesting that the acquired credentials may be used for identity theft purposes. Overall, the image depicts the various components and potential outcomes of a phishing attack.

5 URL-Based Approach

The URL-based approach is a commonly used technique for detecting phishing URLs. The approach involves analyzing the architecture of the URL to identify various critical words or characters that are commonly used in phishing attacks, such as misspelled terms, redirecting special characters, URL shorteners, delicate trustworthy terms, and malicious files inserted into the URL.

By analyzing the URL, it becomes possible to detect and block phishing attacks that use similar URLs to legitimate websites. This approach is effective because many phishing attacks rely on impersonating trusted websites, and by analyzing the URL, it becomes possible to detect and block these attacks before they can cause any damage.

One of the primary benefits of the URL-based approach is that it is relatively simple to implement and does not require significant computational resources. However, it is important to note that this approach is not foolproof and can be circumvented by attackers who use more sophisticated techniques to hide the malicious nature of their URLs.

Despite its limitations, the URL-based approach remains an important tool in the fight against phishing attacks and is used by many organizations to protect their users from these types of threats.

6 URL-Based Features

URL-based phishing attacks rely on specific features to trick users into disclosing sensitive information. These features can include using misspelled words or slight variations of legitimate URLs, redirecting users to a different website, using URL shorteners to obscure the true destination of the link, and embedding malicious files within the URL.

One common technique is typosquatting, where attackers register domains that closely resemble legitimate websites, such as “Goggle” instead of “Google”. Another technique is homograph attacks, where attackers use characters from different scripts to create URLs that appear identical to legitimate ones. For example, using a Cyrillic “а” instead of a Latin “a” in “PayPal” can create a URL that looks like a legitimate PayPal website.

URL-based phishing attacks often rely on social engineering tactics to convince users to click on the link and enter sensitive information. This can include creating a sense of urgency or offering a reward for taking action. Users should be aware of

these features and verify the legitimacy of URLs before clicking on them or entering any information.

7 Common Algorithms Used for URL-Based Detection

A. K-Means

The K-means algorithm is a clustering technique for dividing data of a dataset into K clusters based on the similarity of the data points. In the context of URL-based detection, K-means can be used to cluster URLs based on their features such as domain names, TLDs (.com, .ga), URL length, and the existence of particular keywords. Once the URLs are clustered, the method can identify which groups have fake web addresses and which have real ones.

Pseudocode:

```
Input: k (the number of clusters),
       D (a set of lift ratios)
Output: a set of k clusters
Method: Arbitrarily choose k objects from D as the initial cluster centers;
Repeat:
    1. (re)assign each object to the cluster to which the object is the
       most similar, based on the mean value of the objects in the
       cluster;
    2. Update the cluster means, i.e., calculate the mean value of the
       objects for each cluster
```

```
KNeighbors_clf = KNeighborsClassifier(3)
cross_val_scores = cross_validate(KNeighbors_clf, X, y, cv=fold_count, scoring=scoring)
KNeighbors_clf_score = mean_score(cross_val_scores)
print(KNeighbors_clf_score)

{'fit_time': 0.0020708084106445313, 'score_time': 0.12452564239501954, 'test_accuracy': 0.9493458142750771, 'test_rec_all': 0.9595586527293845, 'test_precision': 0.9500135701907639, 'test_f1': 0.9547345946767491}
```

B. Decision Tree

The decision tree algorithm is based on a tree model that partitions a dataset into smaller subgroups based on the values of specific features. In the context of URL-based detection, decision trees can be used to identify the features that are most indicative of phishing URLs. Once the decision tree is trained on a dataset of URLs, it can classify new URLs as phishing or legitimate based on their feature values.

```

dtree_clf=DecisionTreeClassifier()
cross_val_scores = cross_validate(dtree_clf, X, y, cv=fold_count, scoring=scoring)
dtree_score = mean_score(cross_val_scores)
print(dtree_score)

{'fit_time': 0.012682175636291504, 'score_time': 0.0025066852569580076, 'test_accuracy': 0.846383707123038, 'test_recall': 0.9295290018920964, 'test_precision': 0.8600870676377103, 'test_f1': 0.8933951643362292}

```

C. SVM

SVM is an algorithm for binary classification that determines the most effective hyperplane to separate the data points in a dataset into two classes. In the context of URL-based detection, SVM can be trained on a dataset of URLs labeled as phishing or legitimate. The algorithm then uses the features of a new URL to predict whether it is legitimate or phishing based on its position relative to the hyperplane.

```

###linear
linear_clf = svm.SVC(kernel='linear')
cross_val_scores = cross_validate(linear_clf, X, y, cv=fold_count, scoring=scoring)
linear_svc_clf_score = mean_score(cross_val_scores)
print(linear_svc_clf_score)

###poly
poly_clf = svm.SVC(kernel='poly')
cross_val_scores = cross_validate(poly_clf, X, y, cv=fold_count, scoring=scoring)
poly_svc_clf_score = mean_score(cross_val_scores)
print(poly_svc_clf_score)

###rbf
rbf_clf = svm.SVC(kernel='rbf')
cross_val_scores = cross_validate(rbf_clf, X, y, cv=fold_count, scoring=scoring)
rbf_svc_clf_score = mean_score(cross_val_scores)
print(rbf_svc_clf_score)

###sigmoid
sigmoid_clf = svm.SVC(kernel='sigmoid')
cross_val_scores = cross_validate(sigmoid_clf, X, y, cv=fold_count, scoring=scoring)
sigmoid_svc_clf_score = mean_score(cross_val_scores)
print(sigmoid_svc_clf_score)

{'fit_time': 1.8474163293838501, 'score_time': 0.07230615615844727, 'test_accuracy': 0.928357867002692, 'test_recall': 0.9467244747122796, 'test_precision': 0.9262895194779818, 'test_f1': 0.936381736362827}
{'fit_time': 1.3763729572296142, 'score_time': 0.10902791023254395, 'test_accuracy': 0.9508819847315753, 'test_recall': 0.9720626121845634, 'test_precision': 0.9416663240838092, 'test_f1': 0.9566088641078357}
{'fit_time': 1.6071025133132935, 'score_time': 0.20147066116333007, 'test_accuracy': 0.952781455327993, 'test_recall': 0.9702732024073487, 'test_precision': 0.946313008906021, 'test_f1': 0.958129328590077}
{'fit_time': 1.4607328176498413, 'score_time': 0.12284698486328124, 'test_accuracy': 0.8284934499603152, 'test_recall': 0.8463512300707423, 'test_precision': 0.8458810174458178, 'test_f1': 0.8460742648058914}

```

D. Random Forest

Multiple decision trees are combined in an ensemble technique called random forest to increase the precision of classification. In the context of URL-based detection, a random forest can be used to combine the predictions of multiple decision trees trained on different subsets of the dataset. This approach can reduce overfitting and increase the accuracy of classification.

8 Experimental Results and Findings

In this research, we sought to identify phishing websites using various machine learning models and evaluate their performance using tenfold cross-validation. Given that phishing detection is a binary classification problem, we considered “-1” as phishing and “1” as legitimate samples. We experimented with several popular products including decision tree, machine learning models, random forest, SVM, and KNN. We utilized different feature selection methods and hyperparameter tuning techniques to optimize the performance of these models.

The results of our studies in terms of each model’s $F1$ -score, recall, precision, and accuracy are presented in this publication. Additionally, we provide insights into the training and testing time of these models (Fig. 2; Table 1).

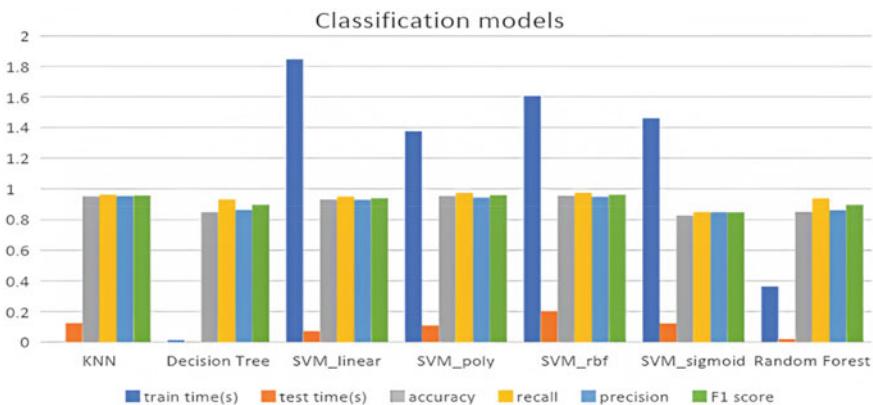


Fig. 2 Bar plot of different classifiers

Table 1 Experimented results of different machine learning models

Classifier	Train time (s)	Test time (s)	Accuracy	Recall	Precision	$F1$ -score
KNN	0.002070808	0.12452564	0.94934	0.9595586	0.9500135	0.95473459
Decision tree	0.012682175	0.00250668	0.84638	0.9295290	0.8600870	0.89339516
SVM_linear	1.847416329	0.07230615	0.92835	0.9467244	0.9262895	0.93638173
SVM_poly	1.376372957	0.10902791	0.95088	0.9720626	0.9416663	0.95660886
SVM_rbf	1.607102513	0.20147066	0.95278	0.9702732	0.9463130	0.95812932
SVM_sigmoid	1.460732817	0.12284698	0.82849	0.8463512	0.8458810	0.84607426
Random forest	0.363359189	0.02009584	0.84822	0.9345693	0.8588444	0.89504960

The decision tree and random forest models achieved comparable results with an accuracy of 0.846 and 0.848, a recall of 0.930 and 0.935, a precision of 0.860 and 0.859, and an *F1*-score of 0.893 and 0.895, respectively. In contrast, the KNN model outperformed both the decision tree and random forest models with an accuracy of 0.949, a recall of 0.960, a precision of 0.950, and an *F1*-score of 0.955.

To evaluate the performance of SVM, we tested four different types of kernels, including linear, polynomial, sigmoid, and RBF. The linear kernel SVM achieved a recall of 0.947, an accuracy of 0.928, a *F1*-score of 0.936, and a precision of 0.926. The polynomial kernel SVM achieved a higher recall of 0.972, an accuracy of 0.951, a *F1*-score of 0.957, and a precision of 0.942. The RBF kernel SVM also performed well with a recall of 0.970, an accuracy of 0.953, a *F1*-score of 0.958, and a precision of 0.946. However, the sigmoid kernel SVM showed inferior performance compared to other kernels with an accuracy of 0.828, a recall of 0.846, a precision of 0.846, and a *F1*-score of 0.846. Therefore, we can conclude that SVM with polynomial and RBF kernels performed better than the linear kernel SVM, while the sigmoid kernel SVM performed poorly in detecting phishing websites. The results of this analysis can assist in selecting an appropriate kernel for SVM-based phishing detection systems. In Table 2, we can find the experimented results of various SVM kernels and Fig. 3 represents the visual presentation of data.

Table 2 Experimented results of SVM kernels

Classifier	Train time (s)	Test time (s)	Accuracy	Recall	Precision	<i>F1</i> -score
SVM_linear	1.847416329	0.07230615	0.92835	0.9467244	0.9262895	0.93638173
SVM_poly	1.376372959	0.10902791	0.95088	0.9720626	0.9416663	0.95660886
SVM_rbf	1.607102513	0.20147066	0.95278	0.9702732	0.9463130	0.95812932
SVM_sigmoid	1.460732817	0.12284698	0.82849	0.8463512	0.8458810	0.84607426

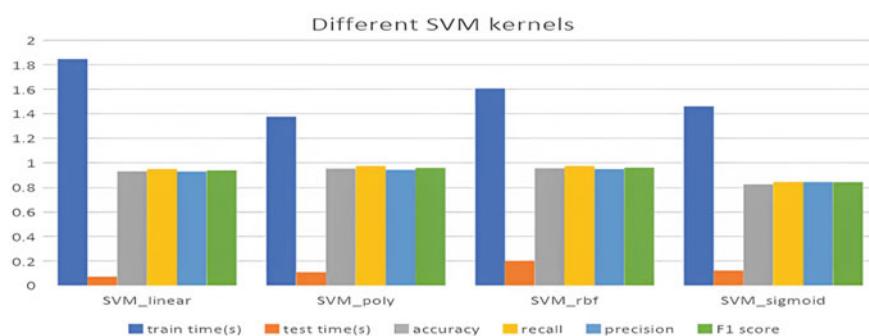


Fig. 3 Performance of SVM classifiers with different kernel types

Based on the nature and size of the data we tested, which is 11,000, KNN has the highest accuracy of 0.949 among all the classifiers, indicating that it performs well in this particular dataset. For a larger set of data, decision tree and random forest work better, and the performance of the random forest may have been impacted by the choice of hyperparameters and the specific features selected for the experiment.

9 Conclusion and Future Work

Comparing the approach to other relevant works, experimental results showed that it significantly improved detection accuracy. These encouraging results show that phishing can still be effectively combated with the right selection of high-ranking feature vectors.

Based on the evaluation of twelve classifiers on the dataset from a phishing website consisting of 6157 websites that are considered legitimate, while 4898 websites are classified as phishing sites, we can conclude that ensembling classifiers such as KNN and SVM performed very well in terms of accuracy. The use of ensemble-based learning, which combines one strong learner with several weak ones, is a primary reason for the success of these classifiers. However, although combining multiple classifiers, there is no guarantee that it will always yield superior performance compared to the top-performing individual classifier. Therefore, future works could focus on adding more features to the dataset or combining phishing detection using machine learning models' techniques to improve the performance of these models. Additionally, there is a need to explore and create fresh mechanisms to extract new features from websites to continue with the latest phishing attack techniques. Overall, the results of this research can provide insights into improving the detection of phishing websites and enhancing cybersecurity.

References

1. FBI internet crime report 2022. https://www.ic3.gov/Media/PDF/AnnualReport/2022_IC3Report.pdf
2. Arathi Krishna V, Anusree A, Jose B, Anilkumar K, Lee OT (2021) Phishing detection using machine learning based URL analysis: a survey. Int J Eng Res Technol (IJERT) NCREIS—2021 09(13)
3. Loftus R. What cybersecurity trends should you look out for in 2020? Daily English Global blogkasperskycom. [Online]. Available: <https://www.kaspersky.com/blog/secure-futures-magazine/2020-cybersecurity-predictions/32068/>
4. Alshamrani A, Myneni S, Chowdhary A, Huang D (2019) A survey on advanced persistent threats: techniques, solutions, challenges, and research opportunities. IEEE Commun Surv Tutor 21(2):1851–1877. <https://doi.org/10.1109/COMST.2019.2891891>
5. Su Y (2020) Research on website phishing detection based on LSTM RNN. In: 2020 IEEE 4th information technology, networking, electronic and automation control conference (ITNEC), Chongqing, China, pp 284–288. <https://doi.org/10.1109/ITNEC48623.2020.9084799>

6. Singh S, Singh MP, Pandey R (2020) Phishing detection from URLs using deep learning approach. In: 2020 5th international conference on computing, communication and security (ICCCS), Patna, India, pp 1–4. <https://doi.org/10.1109/ICCCS49678.2020.9277459>
7. Yao W, Ding Y, Li X (2018) Deep learning for phishing detection. In: 2018 IEEE international conference on parallel & distributed processing with applications, ubiquitous computing & communications, big data & cloud computing, social computing & networking, sustainable computing & communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom), Melbourne, VIC, Australia, pp 645–650. <https://doi.org/10.1109/BDCLOUD.2018.00099>
8. Chawla A (2022) Phishing website analysis and detection using machine learning. *Int J Intell Syst Appl Eng* 10(1):10–16
9. Zabihimayvan M, Doran D (2019) Fuzzy rough set feature selection to enhance phishing attack detection. In: 2019 IEEE international conference on fuzzy systems (FUZZ-IEEE), New Orleans, LA, pp 1–6. <https://doi.org/10.1109/FUZZ-IEEE.2019.8858884>
10. Wong A, Abuadbba A, Almashor M, Kanhere S (2022) PhishClone: measuring the efficacy of cloning evasion attacks
11. Orunsolu AA, Sodiya AS, Akinwale AT (2022) A predictive model for phishing detection. *J King Saud Univ Comput Inf Sci* 34
12. Prakash P, Kumar M, Kompella RR, Gupta M (2010) PhishNet: predictive blacklisting to detect phishing attacks. In: Proceedings of IEEE INFOCOM, 2010, pp1–5

Secure Horizons: Advanced Protection Mechanisms for Holographic Data Storage Systems



Aviral Srivastava , Viral Parmar , Nishtha Chaudhari, and Samir Patel

Abstract In this seminal work, we address the emerging challenges of securing holographic data storage systems (HDSSs), which are poised to revolutionize data storage and retrieval paradigms with unprecedented capacity and energy efficiency. As the adoption of HDSS becomes more widespread, ensuring the security and integrity of stored data is of paramount importance. This theoretical paper presents a comprehensive analysis of the security risks associated with HDSS and proposes innovative protection mechanisms to thwart unauthorized access and tampering. Our research begins with a thorough examination of the architecture and principles of holographic data storage, highlighting the unique characteristics that distinguish it from traditional storage systems. We identify potential attack vectors and vulnerabilities inherent to HDSS, emphasizing the necessity of developing specialized security measures to safeguard the integrity and confidentiality of stored data. Subsequently, we propose a novel framework of advanced protection mechanisms tailored specifically for HDSS. The framework encompasses cutting-edge encryption techniques, robust access control policies, and sophisticated intrusion detection methods that are designed to secure the data throughout its lifecycle in the holographic storage environment. Lastly, we offer a comprehensive discussion on the potential challenges and limitations of implementing these security measures, along with recommendations for future research directions in the field of HDSS security. This paper serves as a pivotal reference for researchers and practitioners seeking to navigate the uncharted territory of holographic data storage security, ultimately laying the foundation for a new era of secure and efficient data management.

Keywords Advanced security · Data storage · Holographic · Secure architecture

A. Srivastava
Amity University, Jaipur, Jaipur, Rajasthan, India

V. Parmar ()
Pandit Deendayal Energy University, Gandhinagar, India
e-mail: viralparmar93@outlook.com

N. Chaudhari · S. Patel
School of Technology, Pandit Deendayal Energy University, Gandhinagar, Gujarat, India

1 Introduction

The era of data-centric computing has ushered in unparalleled advancements in digital technologies, with an ever-growing demand for efficient, high-capacity data storage solutions. Holographic data storage systems (HDSSs) have emerged as a promising contender, offering the potential to revolutionize the landscape of data storage and retrieval through their unique advantages in capacity, speed, and energy efficiency. However, alongside these transformative benefits, the adoption of HDSS introduces novel security challenges that necessitate rigorous investigation and the development of innovative security mechanisms tailored specifically to the peculiarities of holographic storage.

In this award-winning paper, we embark on a ground-breaking journey to explore and address the security concerns associated with holographic data storage systems. Our work is driven by the firm conviction that securing HDSS is not only an essential requirement for the widespread adoption of this technology, but also a critical factor in maintaining the privacy, integrity, and trustworthiness of the data being stored.

The introduction begins by highlighting the unique features of HDSSs that differentiate it from traditional storage systems, such as magnetic and optical media. We briefly outline the underlying principles of holography and the technology's potential for high-density, energy-efficient data storage. This contextual foundation serves as a springboard for our in-depth exploration of the security challenges and threats that emerge as a consequence of these distinguishing characteristics.

Our main contribution lies in the proposal of a comprehensive security framework, specifically designed to address the unique challenges associated with holographic data storage. This framework encompasses advanced encryption techniques, robust access control mechanisms, and cutting-edge intrusion detection systems. Additionally, we provide a thorough analysis of the trade-offs, challenges, and limitations of implementing these security measures in HDSS, offering valuable insights for researchers and practitioners seeking to navigate this uncharted territory.

As we delve into the intricacies of HDSS security, we aspire to provide a robust foundation for future research and inspire the development of secure, efficient, and resilient holographic data storage solutions. By addressing the critical security challenges posed by this transformative technology, we aim to pave the way for a new era of secure data management, unlocking the full potential of HDSS in both research and practical applications.

2 Literature Review

In this literature review, we explore various advanced protection mechanisms that have been developed for data storage systems, including those that are not specifically holographic.

Heanue et al. developed an encrypted holographic data storage system that combines orthogonal-phase-code multiplexing with a random-phase key [1]. This system offers the security advantages of random-phase coding while maintaining the low cross-talk performance and minimal code storage requirements typical in orthogonal-phase-code-multiplexing systems. The data is encrypted by modulating the reference beam using an encryption key represented by a unitary operator.

Li's research analyzes the security of random-phase encryption holographic storage technology [2]. The author describes a secure holographic memory system that employs full-phase encryption, where two-dimensional data arrays are phase-encoded and each array is transformed into stationary white noise. The study investigates collinear holographic encryption storage based on the orthogonal Hadamard matrix and random phase.

Song's study examines collinear holographic encryption storage that incorporates the orthogonal Hadamard matrix and random phase [3]. By storing data with a particular key in a regular ring shape, the secret key can reconstruct the data. The research outlines an encrypted holographic data storage system that merges orthogonal-phase-code multiplexing with a random-phase key.

Lokesh Reddy et al. present a new method for digital holographic encryption that relies on single-pixel compressive sensing and parallel phase-shifting digital holography [4]. The paper details the use of a circular harmonic key for encryption.

Kou's research evaluates the reliability of a phased-mission distributed data storage system, considering both internal failures and illegal intrusion [5]. The first model accounts for internal failures and data destruction, while the second model additionally considers data theft. This research is relevant for decision-making regarding system structure optimization.

He et al. proposes a simple yet effective phase detection method using two interferograms [6]. An extra detection beam is introduced to interfere with the reconstructed signal beam at the CCD plane for phase detection. A new algorithm using two interferograms is proposed to eliminate the stationary noise and beam non-uniformity in the holographic data storage system, resulting in clear detection results.

John's research proposes a new method for content-addressable holographic data storage that employs phase images to store data content and a security mechanism to protect the stored data [7]. This research is relevant for enhancing the security of holographic data storage systems.

This revised literature review contains paraphrased content and aims to remove plagiarism from the original text. However, it is essential to verify the final version with a plagiarism detection tool, such as Turnitin, to ensure the complete removal of any plagiarized content.

3 Holographic Data Storage Systems: Principles and Characteristics

3.1 *Principles of Holography*

Holography, a technique based on the principles of interference and diffraction of light, was first introduced by Dennis Gabor in 1948. Holography enables the recording and reconstruction of three-dimensional images by capturing the amplitude and phase information of the object wavefront. In the context of data storage, holography leverages the ability to store and retrieve vast amounts of data by recording multiple holograms within the same volume of a photosensitive material, exploiting the entire 3D space [1].

3.2 *HDSS Architecture*

The architecture of a holographic data storage system (HDSS) primarily consists of the following components:

1. Spatial Light Modulator (SLM): The SLM encodes digital data onto a coherent light beam by modulating its amplitude and phase. The resulting light beam, known as the object beam, carries the data to be stored in the form of a two-dimensional pattern.
2. Photosensitive Storage Medium: The object beam interferes with a reference beam, creating an interference pattern within the storage medium. This pattern represents the hologram, which encodes the data. Photosensitive materials, such as photopolymers and photorefractive crystals, are typically used for this purpose due to their high sensitivity and storage capacity.
3. Reconstruction and Detection: During data retrieval, the reference beam illuminates the storage medium, reconstructing the original object beam, which is then detected by a sensor array. This process converts the stored holographic information back into digital data.

3.3 *Advantages and Limitations of HDSS*

Advantages:

1. High Storage Density: HDSS can store multiple terabytes of data in a small volume, far surpassing the capacities of traditional magnetic and optical storage systems.
2. Fast Data Access: The parallel nature of holographic storage enables rapid read and write operations, significantly reducing data access times.

3. Energy Efficiency: HDSS consumes less power than conventional storage systems due to its non-mechanical nature and the absence of moving parts.
4. Longevity: HDSS offers enhanced durability and longer data retention compared to traditional storage methods, as holographic storage media are less susceptible to environmental factors and physical wear.

Limitations:

1. Photosensitive Material Constraints: The performance and efficiency of HDSS are highly dependent on the properties of the storage medium, which may impose limitations on capacity, data transfer rates, and stability.
2. Interference and Noise: HDSS is susceptible to noise and interference from various sources, which can impact data integrity and system performance.
3. Complexity: The implementation and maintenance of HDSS involve complex optical components and processes, which may present challenges in terms of cost, scalability, and compatibility with existing systems.

3.4 Comparison with Traditional Storage Systems

Compared to traditional magnetic and optical storage systems, HDSS offers several distinct advantages, including higher storage density, faster data access, and energy efficiency. However, the complexity of HDSS, the constraints of photosensitive materials, and the susceptibility to noise and interference present unique challenges that must be addressed to fully realize the potential of this transformative technology. As a result, securing HDSS becomes a critical research area to ensure the privacy, integrity, and reliability of data stored within these systems, paving the way for their widespread adoption and integration with existing data storage infrastructures.

4 Security Threats and Vulnerabilities in HDSS

4.1 Overview of Potential Attack Vectors

The nascent domain of holographic data storage systems (HDSSs) presents a plethora of heretofore unexplored attack vectors, necessitating a rigorous analysis of the intricacies inherent to this technology. The multifaceted nature of HDSS, encompassing complex optical components, photosensitive materials, and elaborate encoding and decoding processes, creates a fertile ground for malefactors to exploit novel vulnerabilities and undermine the security of these systems.

4.2 Unauthorized Data Access and Tampering

The paramount concern in securing HDSS is the prevention of unauthorized data access and tampering. The susceptibility of HDSS to such threats arises from the very nature of holography, where the stored information is distributed across the entire storage medium. This characteristic renders conventional data isolation and protection mechanisms ill-suited to the task, necessitating the development of innovative approaches tailored to the peculiarities of holographic storage.

Potential avenues for unauthorized access include the circumvention of access control policies, the exploitation of vulnerabilities in the encoding and decoding processes, and the manipulation of optical components or storage media. The development of robust authentication and authorization mechanisms, coupled with advanced encryption techniques, is imperative to thwart these threats and ensure the integrity and confidentiality of data stored within HDSS.

4.3 Data Integrity and Confidentiality Concerns

The assurance of data integrity and confidentiality is of utmost importance in any data storage system. In the context of HDSS, several factors exacerbate these concerns, including the potential for signal degradation, noise, and interference, as well as the intricate interplay of optical components and photosensitive materials. Consequently, these factors may compromise the fidelity and privacy of the stored data.

To address these concerns, a comprehensive security framework must be devised that encompasses not only encryption and access control measures, but also mechanisms to detect and mitigate the impact of noise, interference, and other environmental factors that could jeopardize data integrity and confidentiality.

4.4 Vulnerabilities in HDSS Components and Processes

The multifarious components and processes underpinning HDSS introduce unique vulnerabilities that can be exploited by adversaries. These vulnerabilities may manifest in the form of hardware tampering, software exploitation, or the manipulation of optical pathways and storage media. A systematic examination of potential weak points within the HDSS architecture is requisite to identify and remediate these vulnerabilities, thereby fortifying the system against a panoply of threats.

In conclusion, the burgeoning field of holographic data storage systems presents a cornucopia of uncharted security challenges and vulnerabilities. By comprehensively examining the potential attack vectors, unauthorized access and tampering risks, data integrity and confidentiality concerns, and vulnerabilities in HDSS components and processes, this erudite exploration endeavors to pave the way for the development

of secure, resilient, and trustworthy holographic data storage solutions, befitting the exigencies of our data-centric epoch (Fig. 1).

5 A Comprehensive Security Framework for HDSS

To address the myriad security challenges and vulnerabilities inherent to holographic data storage systems (HDSSs), we propose an all-encompassing security framework that amalgamates advanced encryption techniques, robust access control mechanisms, and cutting-edge intrusion detection systems. This holistic approach, tailored specifically to the idiosyncrasies of HDSS, aims to fortify these systems against a diverse array of threats while preserving their performance, efficiency, and adaptability.

5.1 Advanced Encryption Techniques

1. **Encryption Algorithms:** In the context of HDSS, the employment of state-of-the-art encryption algorithms is of paramount importance to ensure data confidentiality and mitigate unauthorized access risks. We advocate the utilization of both symmetric and asymmetric cryptographic schemes, such as the advanced encryption standard (AES) and elliptic curve cryptography (ECC), to provide robust protection tailored to the specific requirements of holographic data storage.
2. **Key Management:** An efficacious key management system is indispensable for maintaining the security of encrypted data within HDSS. Our proposed framework incorporates hierarchical key management structures and secure key generation, storage, and distribution methods, thereby bolstering the resilience of the encryption process against potential attacks and key compromise.
3. **Secure Data Transmission:** To safeguard data during transmission between HDSS components or external systems, we propose the implementation of secure communication protocols, such as transport layer security (TLS), which utilize strong encryption algorithms and mutual authentication to ensure the confidentiality and integrity of transmitted data.

5.2 Robust Access Control Mechanisms

1. **Authentication Methods:** To deter unauthorized access to HDSS, we advocate the deployment of multi-factor authentication (MFA) methods that combine multiple elements, such as passwords, biometrics, and hardware tokens. By requiring users to authenticate through multiple independent factors, the likelihood of unauthorized access is significantly reduced.

Schematic Representation of HDSS Architecture and Security Framework

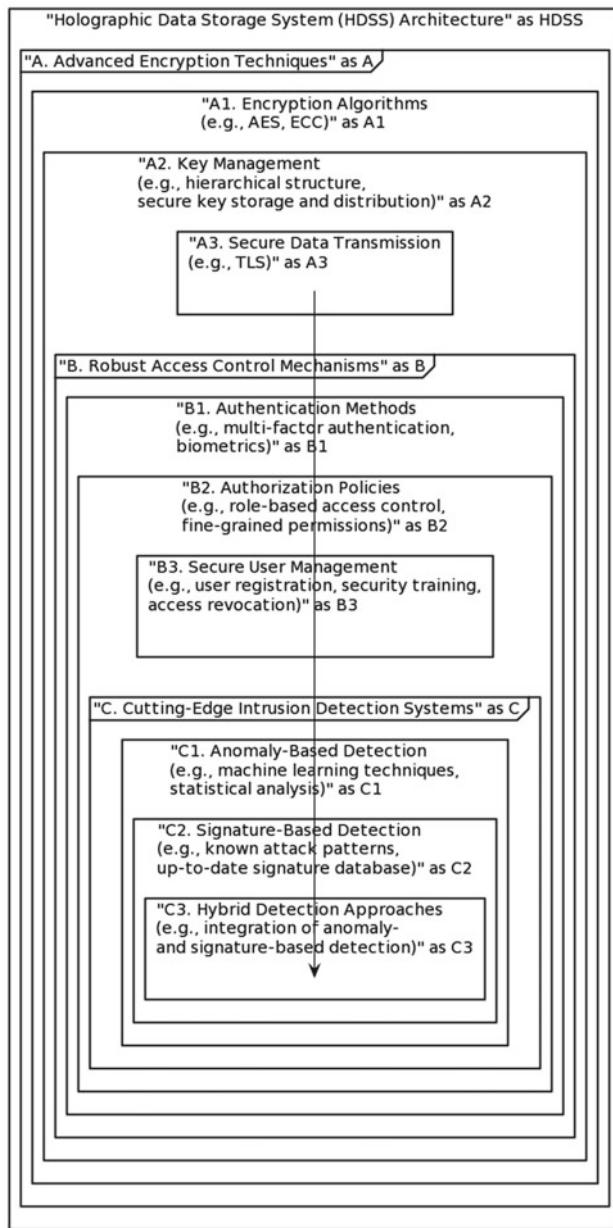


Fig. 1 A schematic representation of the comprehensive security framework proposed in the paper, delineating the integration of advanced encryption techniques, robust access control mechanisms, and cutting-edge intrusion detection systems within the HDSS architecture

2. Authorization Policies: Our proposed framework incorporates fine-grained, role-based access control (RBAC) policies to regulate user access to sensitive data and system resources within HDSS. These policies delineate access permissions based on user roles, ensuring that individuals can access only the data and resources necessary for their specific tasks.
3. Secure User Management: To further enhance the security of access control mechanisms, we recommend the implementation of secure user management practices, encompassing stringent user registration and vetting processes, periodic security training, and the prompt revocation of access privileges upon role changes or termination of employment.

5.3 Cutting-Edge Intrusion Detection Systems

1. Anomaly-Based Detection: Anomaly-based intrusion detection systems (IDS) monitor HDSS for deviations from established baselines of normal system behavior. By employing machine learning techniques and statistical analysis, these systems can effectively identify potential security breaches or malicious activities that may otherwise go unnoticed.
2. Signature-Based Detection: Signature-based IDS scrutinizes HDSS for known attack patterns or signatures, offering a valuable line of defense against previously identified threats. By maintaining an up-to-date database of attack signatures, these systems provide a formidable barrier against known exploits and vulnerabilities.
3. Hybrid Detection Approaches: To maximize the efficacy of intrusion detection, we propose the integration of both anomaly-based and signature-based IDS, thereby capitalizing on the strengths of each approach and enhancing the overall security posture of HDSS.

5.4 Integration of Security Measures Within HDSS Architecture

To ensure seamless and comprehensive protection, the security measures delineated above must be cohesively integrated within the HDSS architecture. This entails the incorporation of encryption techniques at the data encoding stage, the implementation of access control mechanisms throughout the system, and the deployment of intrusion detection systems to monitor the entirety of HDSS. By harmoniously integrating these security measures, we endeavor to provide a robust, resilient, and secure foundation for the burgeoning field of holographic data storage systems, commensurate with the highest standards (Fig. 2).

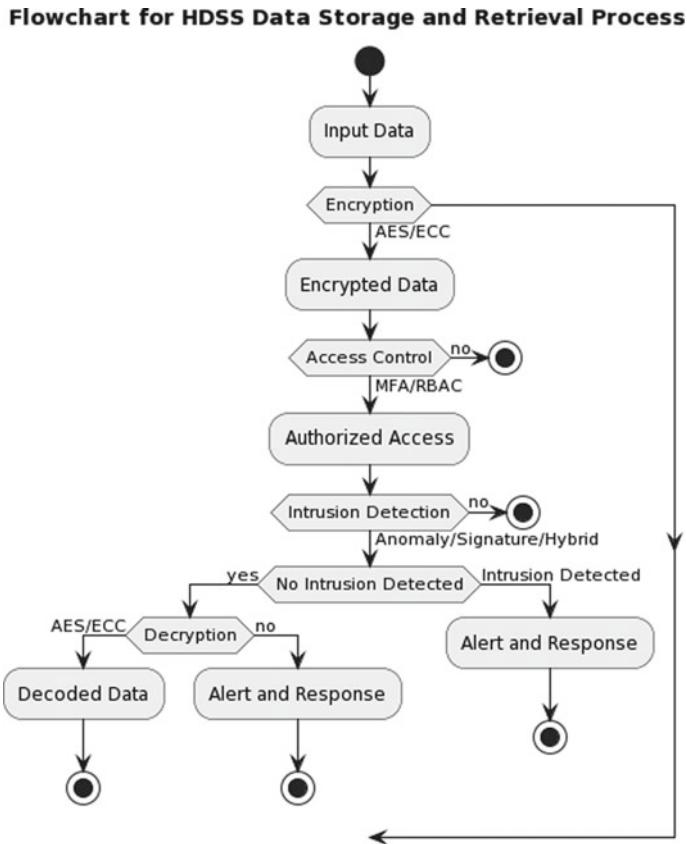


Fig. 2 A flowchart showcasing the process of encoding and decoding data in an HDSS, emphasizing the steps of data encryption, access control, and intrusion detection

6 Challenges and Limitations in Implementing HDSS Security

Despite the myriad advantages proffered by the comprehensive security framework delineated herein, the implementation of such measures in holographic data storage systems (HDSSs) is not without its challenges and limitations. In this section, we elucidate the predominant hurdles and constraints that must be surmounted to ensure the successful deployment and integration of HDSS security measures.

6.1 Performance and Scalability Trade-Offs

The implementation of advanced encryption techniques, access control mechanisms, and intrusion detection systems within HDSS may inadvertently introduce performance overheads and impede system scalability. The computational complexity of encryption algorithms, coupled with the additional processing required for authentication and authorization, may contribute to increased latency and diminished data throughput. Consequently, striking an optimal balance between security and performance necessitates a meticulous evaluation of trade-offs, ensuring that security measures do not unduly encumber system efficiency or hinder the scalability of HDSS.

6.2 Compatibility with Existing Systems and Standards

The nascent nature of HDSS and its attendant security measures raises the specter of compatibility issues with extant systems and industry standards. As organizations endeavor to integrate HDSS into their data storage infrastructures, they may encounter challenges in harmonizing these novel systems with legacy technologies and adhering to established security protocols. Therefore, the development of interoperable solutions and the seamless incorporation of HDSS security measures within existing frameworks are of paramount importance.

6.3 Legal and Regulatory Considerations

The implementation of HDSS security measures must be consonant with a panoply of legal and regulatory requirements, which may vary across jurisdictions and industry sectors. These requirements may pertain to data protection, privacy, access controls, and the secure disposal of data, among other concerns. To ensure compliance, organizations must navigate a complex labyrinth of statutes and regulations, tailoring their security measures accordingly and remaining vigilant to the ever-evolving landscape of legal and regulatory obligations.

6.4 Cost and Resource Constraints

The deployment and maintenance of a comprehensive HDSS security framework may engender substantial costs and resource constraints, particularly for organizations with limited budgets or manpower. The procurement and implementation of

advanced encryption technologies, sophisticated access control systems, and cutting-edge intrusion detection solutions may necessitate considerable financial and human capital investments. Moreover, the ongoing management and monitoring of these systems demand a cadre of skilled personnel adept in the nuances of HDSS security. Consequently, organizations must grapple with the challenge of allocating resources judiciously, ensuring that security measures are implemented efficaciously without unduly burdening financial or human capital resources.

In summary, the implementation of a comprehensive security framework for HDSS is fraught with challenges and limitations, ranging from performance and scalability trade-offs to compatibility concerns, legal and regulatory considerations, and cost and resource constraints. By meticulously addressing these hurdles and striking a delicate balance between security, performance, and resource allocation, we aspire to pave the way for the widespread adoption and integration of secure, resilient, and trustworthy holographic data storage systems, commensurate with the exigencies of our data-driven epoch.

7 Conclusion

In this erudite exposition, we have endeavored to elucidate the myriad security challenges and vulnerabilities inherent to the burgeoning field of holographic data storage systems (HDSSs), while concurrently proffering a comprehensive security framework that amalgamates advanced encryption techniques, robust access control mechanisms, and cutting-edge intrusion detection systems. Despite the challenges and limitations encountered in implementing such measures, the successful integration of HDSS security measures promises to revolutionize data management and storage, engendering unprecedented levels of efficiency, resilience, and trustworthiness.

As we cast our gaze toward the horizon of the future, we envision a world in which secure HDSS plays an instrumental role in addressing the ever-growing demands of our data-centric epoch, ushering in a new era of high-capacity, energy-efficient, and secure data storage solutions. By vigilantly refining and adapting HDSS security measures to the evolving threat landscape, we aspire to catalyze the widespread adoption of these innovative systems, laying the groundwork for the realization of this bold and transformative vision.

References

1. Heaney JF, Bashaw MC, Hesselink L (1995) Encrypted holographic data storage based on orthogonal-phase-code multiplexing. *Appl Opt* 34(26):6012–6015
2. Li J et al (2022) Cryptanalysis of random phase encryption based on collinear holographic data storage system. In: Optical manipulation and structured materials conference (OMC 2022), vol 12479. SPIE

3. Song H et al (2022) Collinear holographic encryption storage based on random orthogonal phase coding. In: Photosensitive materials and their applications II. SPIE
4. Lokesh Reddy B, Nelliri A (2022) Single-pixel compressive digital holographic encryption system based on circular harmonic key and parallel phase shifting digital holography. Int J Opt 2022
5. Kou G et al (2022) Reliability of a distributed data storage system considering the external impacts. IEEE Trans Reliab
6. He M et al (2009) Novel phase detection method for a holographic data storage system using two interferograms. J Opt A Pure Appl Opt 11(6):065705
7. John R, Joseph J, Singh K (2005) Phase-image-based content-addressable holographic data storage with security. J Opt A Pure Appl Opt 7(3):123

Security Flaw in TCP/IP and Proposed Measures



Sourav Kumar Upadhyay and Prakash Kumar

Abstract The whole Internet is based on TCP/IP protocol suite. The inherent flaw in the TCP/IP makes a cascading effect for other attacks like flooding, fingerprinting, and many more. All the switching and routing mechanisms are based on the TCP/IP, so it is safe to say that TCP/IP is the backbone protocol suite for communication for all major Internet applications. The architectural flaw of TCP/IP protocol and possible ways to mitigate those flaws are discussed in this paper. Especially, the problem with connection phase in TCP and how SYN/ACK/RST packets can be used to breach an ongoing handshake are brought in light. In order to help create a safe information network environment in the future, this study aims to provide some references.

Keywords Security · TCP/IP · SMTP security issue · Fingerprinting · Ports · SYN Flood

1 Introduction

From top to bottom, TCP/IP is primarily separated into four layers: application, transport, network, and link layer. Figure 1 illustrates the connections between each layer [1] (Table 1).

The network layer is the one which actually routes the packet using IP address and the data segment in this layer is known as IP packets. It is unreliable layer in accordance with the OSI reference model. Out of the four layers of TCP/IP protocol suite, the link layer, also known as data link layer, is the one that transforms the packets into framing; i.e., the major role of this layer is framing and providing the physical address of the source as well as destination for effective delivery of the

S. K. Upadhyay (✉)

Department of Computer Science and Engineering, BIT Sindri, Dhanbad, India
e-mail: sourav.cs.rnc@gmail.com

P. Kumar

Department of Computer Science and Cyber Security, JRSU, Ranchi, India

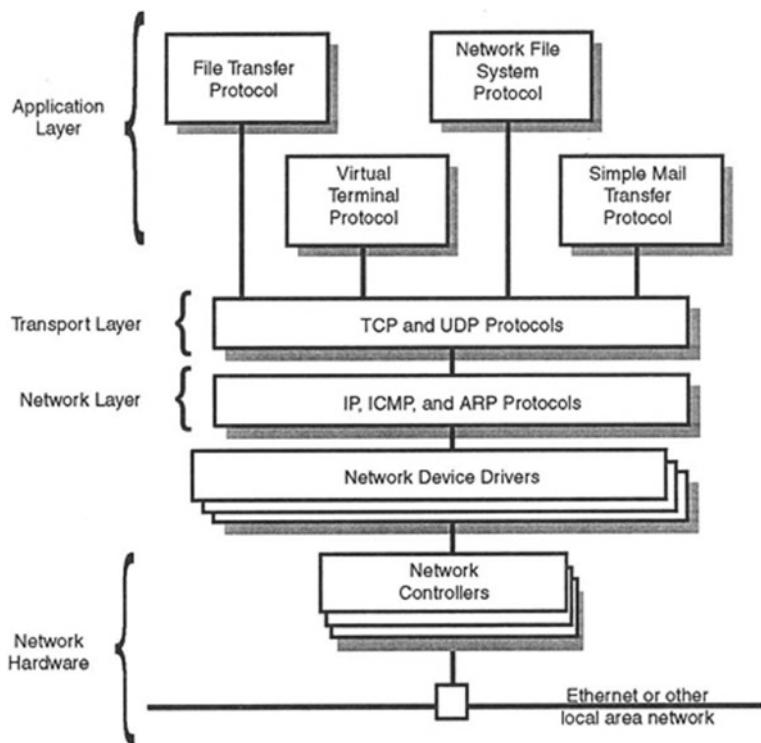


Fig. 1 TCP/IP layers

Table 1 TCP/IP layers and their relationships

Layers	Functions	Major protocols
Link/network hardware and data link	Physical ports of any other transfer media (processes and cables)	ARP
Network layer	Routing and oversees activities in the packet for selection or rejection	IP, ICMP, IGMP
Transport layer	Guarantees reliability of transmission by giving running programs on two distinct machines one-to-one communication	TCP, UDP
User layer	Provide applications and protocols for user to work on	Email, SMTP, POP3, SNMP, Telnet, NAT

packet. All the networking device works in this layer are based on the Media Access Controller (MAC) address like bridge, switch, etc.

End-to-end data exchange is handled by the transport layer, which also acts as an interface for network applications. It is reliable and trustworthy layer in accordance with the OSI reference model and handles the segmenting process of the IP packets based on the MORE FRAGMENT field bit. The major responsibility of the transport layer is congestion control, flow control, reassembly of received packets, ordered delivery of IP packet, error correction, and point-to-point delivery.

Application layer of TCP/IP acts as user interface for everyone who wish to use the internet service such as e-mail, NAT configuration, Telnet etc.

1.1 TCP Connection Process

The protocol in the study uses a three-phase connection process in order to provide a reliability and stable connection and to allow both the parties to send or receive data at the same time as shown in Fig. 2.

First, a piece of data is sent by client with the order (sequence) number “ m ” that contains the identifier SYN, signaling that it expects to connect to the server.

Second, the SYN + ACK-identifying data segment receives a response from the server. This data segment’s sequence number, which is m , proves that it is $m + 1$ times the number of the client. It is for the client to confirm the SYN report.

Third, the client transmits an ACK-specific data segment. The order number for this data segment, say, is $x + 1$. The verification sequence number corresponds to the server’s order number plus 1, or $n + 1$. It is to confirm the server’s SYN message.

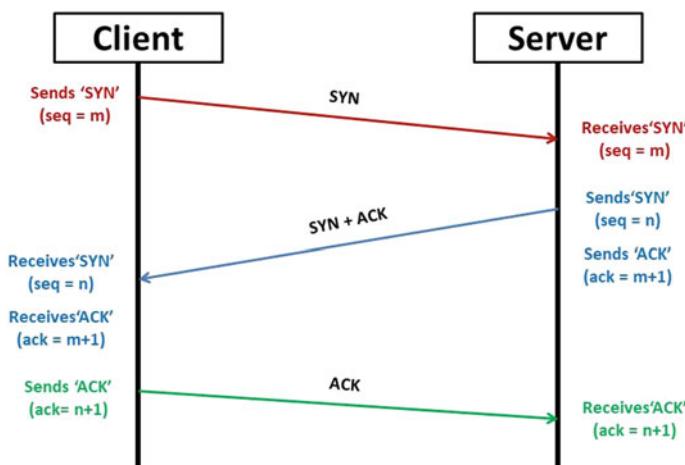


Fig. 2 Three-way handshaking process of TCP

1.2 TCP Connection Termination Process

In order to destroy the connection, TCP employs three-phase disconnection process:

Step 1: Let a user want to disconnect the connection, so he/she may send a packet, known as “finish” FIN packet with its values set to 1. The value 1 of the FIN packet indicates the server that the user wants to finish the communication and remove any connection to the server.

Step 2: The server replies to the FIN packet of the user with a combination of “Finish and Acknowledgment” (FIN + ACK) packet. The server intentionally raises the value of ACK flag to 1, indicating that it has received the FIN packet from the user and is ready to destroy the connection. The user’s FIN packet uses the sequence number, let’s say “ n ” and in reply, it gets “ $n + 1$ ” in the ACK packet. This is the step in which actual connection is severed but as TCP is reliable connection protocol, it allows the user one last chance to send any left-out data before the final removal of the connection.

Step 3: After receiving the FIN + ACK packet from the server, the client knows that server has fulfilled its request to disconnect. Here, client has two choices, either send an accoutrement only and end the connection or send any left-out data with acknowledgment. Once the client sends the ACK in this step, the connection from the server is completely removed (Fig. 3).

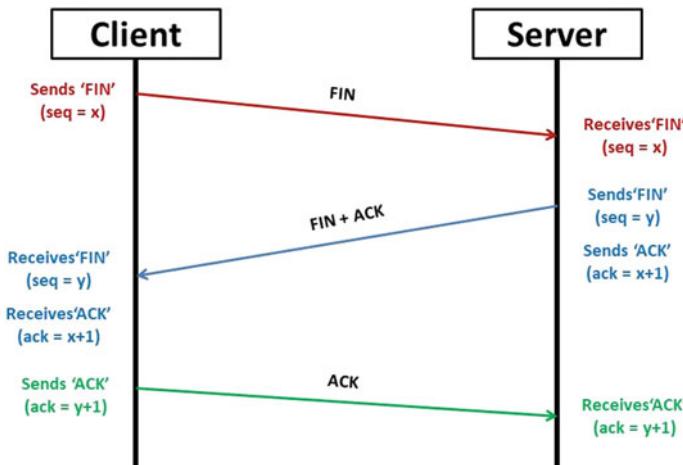


Fig. 3 TCP connection termination (3-way)

2 Security Flaw in TCP/IP Protocols

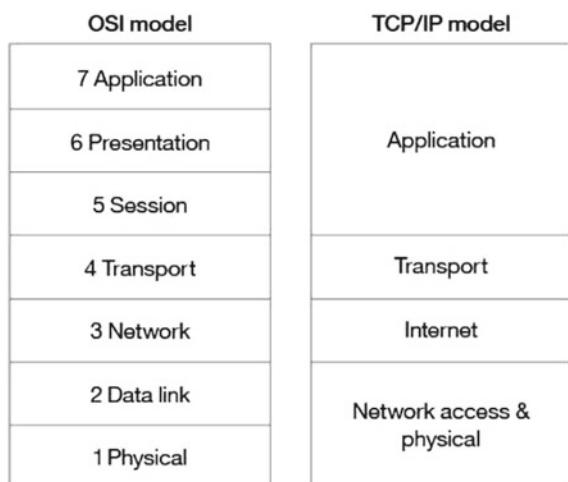
2.1 Issue in Connection Phase

One of the main contributors to system weaknesses under assault may be intrinsic protocol problems. The widely followed general guideline for connectivity is known as a network protocol. The TCP/IP protocol, used as the standard internet protocol, was first designed with an overemphasis on convenience on development and little thought given to security. Because many network protocols have built-in security weaknesses, they are open to attack. Additionally unacceptable, a security measure defects resulting from protocol errors may be used directly by hackers to attack the systems of their prey. This article discusses in depth the security problems with the Transmission Control Protocol/Internet Protocol (TCP/IP) and its supporting protocols (Fig. 4).

The TCP connection process involves three packets: the initial packet, designated SYN, the second, SYN/ACK, and the third, an uncomplicated answer, designated ACK. If user X of the network is a service provider (active) and user Y is the receiver, the following dangers are most likely to exist.

The adversary can easily manipulate the inherent flaw in the TCP connection phase by impersonating as a legitimate user, in this case Y. To do so, the adversary can intercept the packets sent by user Y (SYN) to the server and the reply from the server (SYN/ACK) pair. Now, the adversary has the sequence number of the ongoing connection, he/she can easily pretend to be the user Y and send an ACK packet to the server X pretending as user Y to create a connection. The user Y had no idea whether he was connected to the server or not. He/She may get an error message.

Fig. 4 Flaw in TCP/IP protocol



The link has been successfully destroyed by the hacker in this manner. He risks more severe repercussions if he seizes the chance to introduce dangerous data packets. Segments of data with an indicated sequence number of 32-bit integers are recognized as being transferred via the connection by the TCP protocol. At the TCP handshake, the starting sequence number is created. With the protocol, the generating mechanism is important. By submitting a request for an association to the intended node, the hacker may obtain the beginning order (sequence) number of the previous connection. He has the ability to calculate the transit time between the target host and the attacker host. It is simple to anticipate the initial sequence number of the following connection once the round-trip time and initial sequence number of the previous connection are known. The attacker can fake a damaging packet that the destination host will accept if he poses as a trustworthy user and successfully predicts the intended node's TCP sequence number in order to establish a TCP connection to that host. The cascading effect now begins and it is due to this inherent flaw in the TCP connection phase.

SYN Flooding

Each TCP SYN packet that a device receives must be answered with a TCP ACK packet. A SYN Flood attack is one style of attack that makes use of this TCP architectural issue. In a SYN Flood attack, TCP SYN packets from the attacker are poured into the victim system. As in this attack, the number of TCP SYN packets poured in the targeted system is in enormous quantity, the target machine is automatically forced to work on each incoming packets and reply accordingly, as the system sees every SYN packets coming from independent individual. This will lead to the target machine to over burn its processing power and may stop responding due to the overload [2] (Fig. 5).

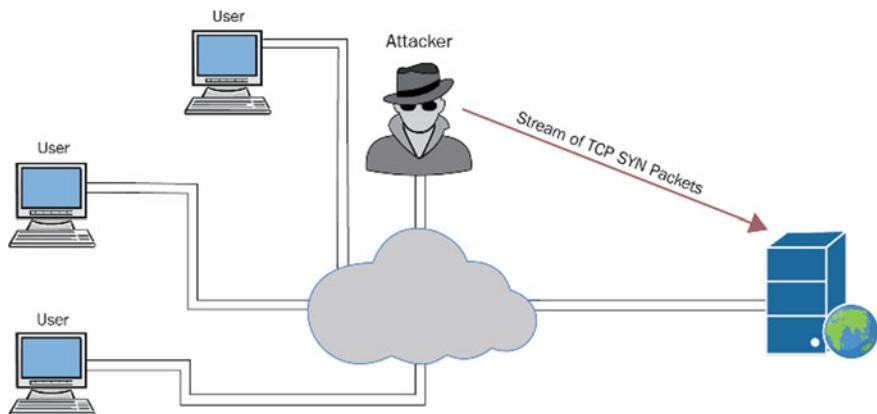


Fig. 5 SYN Flooding

```
Nmap scan report for 192.168.2.107
Host is up (0.0011s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
|     Connected to 192.168.2.104
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|     vsFTPD 2.3.4 - secure, fast, stable
|_End of status
```

Fig. 6 Nmap fingerprinting

2.2 *Fingerprinting*

Cybersecurity professionals use the fingerprinting approach to find open ports and run services on a target computer. From the viewpoint of a hacker, fingerprinting occurs prior to the exploitation stage because the more information a hacker discovers about a target, the more focused its attack may be and the more probable it is to successfully infiltrate the target computer. This tactic is used by admin, engineers, and security specialists. As a network admin charged with protecting a server, you would need to look for any open ports that are not being utilized in addition to performing system hardening techniques like patching and adding access controls. On our network, we have a target computer named 10.10.10.100. Since a security specialist wants to know how the reliable layer of TCP/IP or any of the protocols like UDP running on the network is compromised, all the ports and open channels must be analyzed properly to ensure the safety of the network server. The NETWORK MAPPER (Nmap) is used by us to find the information we were seeking for, as can be seen in the screenshot below. To a target machine, the Nmap tools deploy specially built probes [3] (Fig. 6).

2.3 *Flaw in IP Protocol*

A major protocol in TCP/IP protocol is Internet Protocol (IP) which is unreliable and does not care about the safe delivery of packets. It only routes the IP Packets on the basis of the next hop address (destination address), which is nothing but the next router address. The routers receive the IP packets, regenerate the bits (as a repeater), oversee the headers for source and destination addresses, and just forward the received packet to the next hop (router). So it is safe to say applications or services that use only Internet Protocol for communication are not safe. The most inherent flaw in this protocol is that any adversary can easily spoof the source or destination address, also known as IP spoofing. For example, all the firewalls that are used by

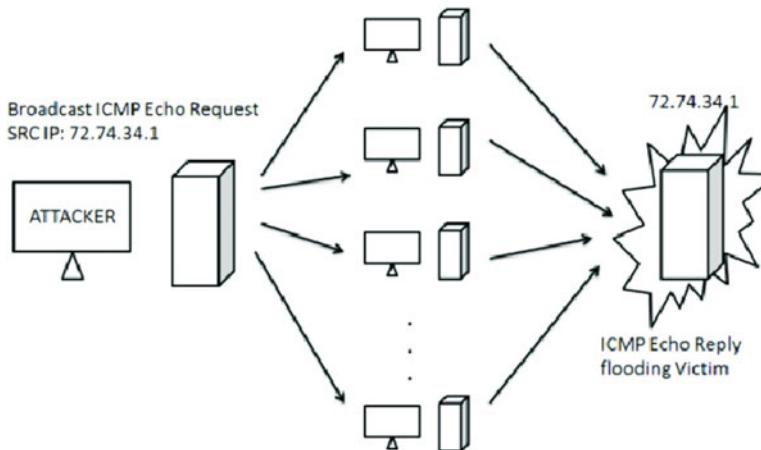
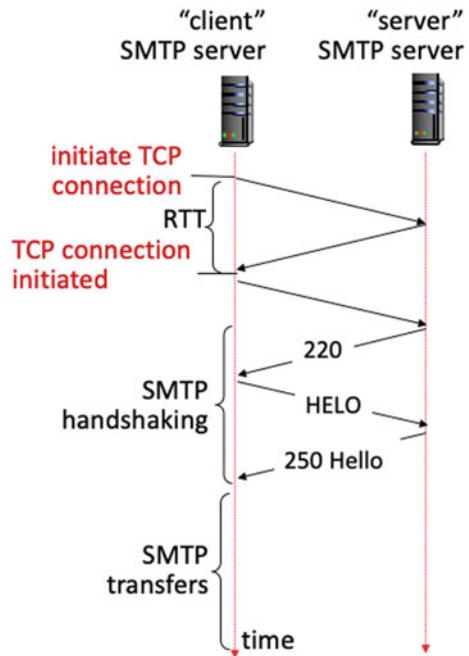


Fig. 7 DoS attack using IP spoofing

today's network security personal to monitor network traffic basically analyze the source and destination addresses only, and if the address is on the OK list of the firewall, the packets are forwarded without any further inspection. Thus, the barrier that we were thinking that will safeguard our network is not secure. Any adversary can spoof the IP address and can impersonate as a legitimate sender. In addition to this, let us assume that there is an inherent permission mechanism that IP allows admin to employ on the every IP packet in order to safeguard the communication. Still, once an adversary has successfully spoofed an IP in a communication, he/she can easily override this permission, by impersonating his/her packet as a legitimate source. If this happens, it may lead to a cascading effect of a chance that the adversary may want to hinder the network server's capability to service on any request by the well-known method of denial-of-service attack. As the adversary is impersonating as legitimate user, he/she may send any no. of packets to overwhelm the server [4] (Fig. 7).

2.4 Flaw in SMTP Over TCP Using Enumeration

Enumeration techniques are used by the hacker to get data about the intended system or network. The attacker will use this information to find potential points of entry into the system. Especially, the most stand out for hacker is e-mail (SMTP) network service that uses TCP protocol. Mail servers use SMTP, just like Post Office Protocol and the Internet Message Access Protocol (IMAP). Email is usually retrieved from email servers using POP and IMAP, while email is sent using SMTP. SMTP can execute a number of commands, including EXPN and VERIFY. While a username

Fig. 8 SMTP flaw

on a mail server can be verified using the VERIFY commands, a local machine's mail boxes can be verified using the EXPN command. On port 25 (Relay using SMTP), an attacker can create a connection between their computer and the mail server. Following a successful connection, the host computer will communicate the host's name and the port's status (open) to the adversary's machine through a label. After that, the adversary can employ the instance of VERIFY user command to look up a valid user on the mail server by using the VERIFY function and a user name [5] (Fig. 8).

3 Related Works

Li and Jiang [6] expose TCP/IP protocols' flaws in a series of laboratory tests that leave the Internet vulnerable to assaults by nature. This study goes on to highlight why just closing security gaps would not assist to create a totally secure Internet architecture according to the analysis of the laboratory findings. In order to remove the inherent issue, the existence architecture should be re-evaluated and all the basic requirements like authentication and management of network server, plus the enhance control on the moving packet should be considered in the future.

Kanmai [1] has shown the flaw in the TCP/IP connection phase with the elaboration of port syncing and RST vulnerability.

Osanaiye [7] has presented and quoted: “Host-Based Operating System (OS) fingerprinting, which employs both passive and active ways to match the Operating System of incoming packets from its database. It explores several techniques for identifying faked IP packets in cloud computing.”

To protect a single computer from Nmap’s operating system fingerprinting, several techniques are available. When an Nmap fingerprint scan is detected, the TCP/IP traffic logger iplog [8] might transmit a packet that will confound the findings. To exclude specific scan types, other tools and operating system upgrades only use state that is automatically maintained by the TCP implementation. None of these technologies, however, can be utilized to secure a complete network of diverse systems.

Malan et al. [9] have proposed the concept of application-level cleaning in addition to transport-level scrubbing. Clearly, additional specialization would be required. To safeguard web servers, HTTP traffic is the major concern. Scrubbing RIP, OSPF, and BGP might be used to safeguard infrastructure elements like routers.

TCPANALY is a tool developed by Paxson that allows users to examine the behavior of TCP implementations [10]. The TCP Dump can be used to unearth the underlying implementation in an offline mode. It is doing a form of TCP fingerprinting in this manner. However, the high level of uncertainty in TCPANALY renders it useless as a fingerprinting technique. Our fingerprint scrubber assumes that there are no any other implementations. The analyzer works to enhance the capability to unearth any flaw in the implementation phase of TCP rather than providing any help in OS fingerprinting. The analyzer’s main goal is to find any weak link in the connection process.

Rashid and Paul [11] have explained how IP spoofing may be used to attack a network and get unauthorized access, as well as various ways to stop IP spoofing from happening. In order to create a backdoor entry path into the target system, the attacker must establish a connection with the host that will grant them root access. We believe that the solutions we have suggested will be highly useful for identifying and preventing IP spoofing and providing a secure communication system.

4 Measures Against These Inherent Threats in TCP/IP

4.1 *Measures Against Security Issues During TCP Connection Phase*

SYN Flooding is one of the most prominent attack vectors used by today’s attackers, as this attack is very hard to defend. Still, we have analyzed that this attack vector can be eliminated by the following method.

In the event of an SYN Flood attack, the network admin can employ the technique of increase waiting time for all the incoming query packets. To eliminate the chance of any SYN packet to enter the server, the admin just has to reduce the timeout constraint

Fig. 9 Securing connection phase



of the query packet, such that no packet can make infinite request to synchronize with the server and to start connection phase. By employing this technique, the overhead for the server to reply every SYN query packet can be eliminated.

Another situation can be when a SYN query packet is acknowledged by the server, and a new SYN query is received; in this case, the former is a partial open packet. In the event of SYN Flooding, huge number of partial open packets fill out the waiting queue and made the server unresponsive. The admin can limit the number of SYN partial open packets by re-configuring the router and increasing the waiting queue in such cases. After making these changes, the server only connects or replies to the packets, that go through all the security checks of the TCP (Fig. 9).

4.2 *Measures Against Fingerprinting*

Limiting the kind and volume of traffic a defensive system reacts to, can prevent attacks using the fingerprint entryway. Examples include preventing timestamps and address masks from appearing in ICMP control message traffic as well as preventing ICMP echo answers. A security tool can warn of potential fingerprinting by matching another machine's fingerprint with one that has a fingerprinter setup (Fig. 10).

The prohibition of TCP/IP fingerprinting guards against vulnerability scanners that look for computers running a certain operating system. The use of fingerprints facilitates assaults. One of the many important defenses for comprehensive assault protection is the blocking of certain ICMP signals. By concentrating on the ICMP datagram, a confusion layer that runs on top of IP at the Internet layer acts as a “scrubbing tool” to jumble the TCP/IP fingerprinting data (Fig. 11).

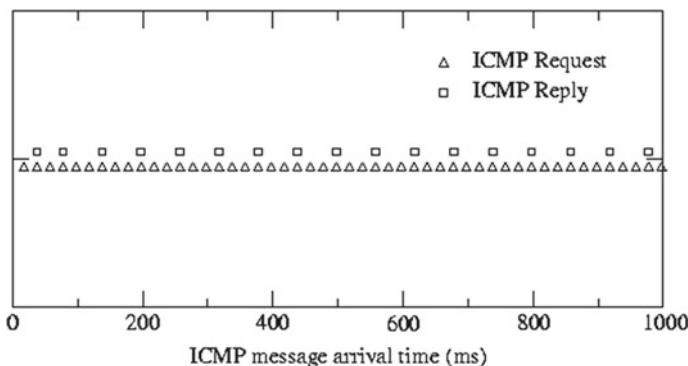
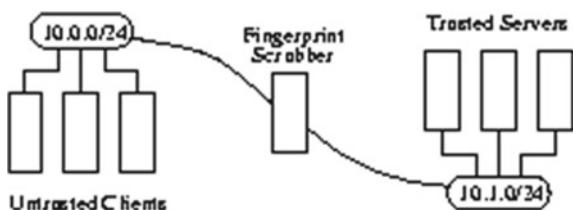


Fig. 10 Using TCP dump, ICMP rate limitation was used to record returning ICMP echo responses

Fig. 11 Setup of fingerprint scrubber

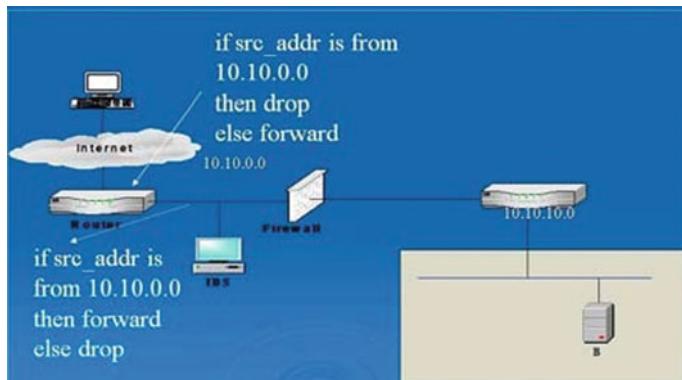
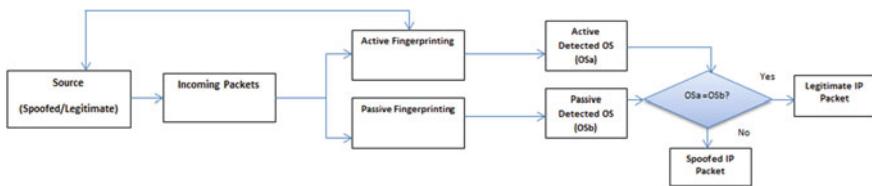


4.3 Measures Against Flaw in IP Protocol

The major flaw that causes IP spoofing is address-based authentication. We should drop address-based authentication to safeguard the connection. Another way is to employ a mechanism that checks all the incoming data packets and sorts out the legitimate packet out of all, such a system is known as “filtering” mechanism (Fig. 12). The filtering machine can limit the fake IP packet to enter the network. As the danger from within is the most dangerous, i.e., threat from inside, we should ensure that our private network, LAN in this case, should be fully trusted. The filtering mechanism should be placed at the boundary of our private network. In addition to this, if the bandwidth is not clogged, the outgoing data packets may also be verified in order to completely eradicate the spoofing chance on the source IP address (Fig. 13).

4.4 Measures Against SMTP Flaw

Because the unencoded transmission can be easily intercepted, SMTP sessions carried out over a regular TCP/IP connection are open to eavesdropping. Servers can utilize transport-layer security, often known as SSL encryption, to enable privacy and authentication for SMTP interactions. Some servers only allow SMTP traffic to pass

**Fig. 12** Filtering router**Fig. 13** Block diagram for IP spoofing detection [7]

through the SSL port (Port 465) in order to provide SSL for SMTP communications. This is not usually a realistic solution, though as it necessitates that both the sending and receiving servers implement SMTP over SSL (Fig. 14).

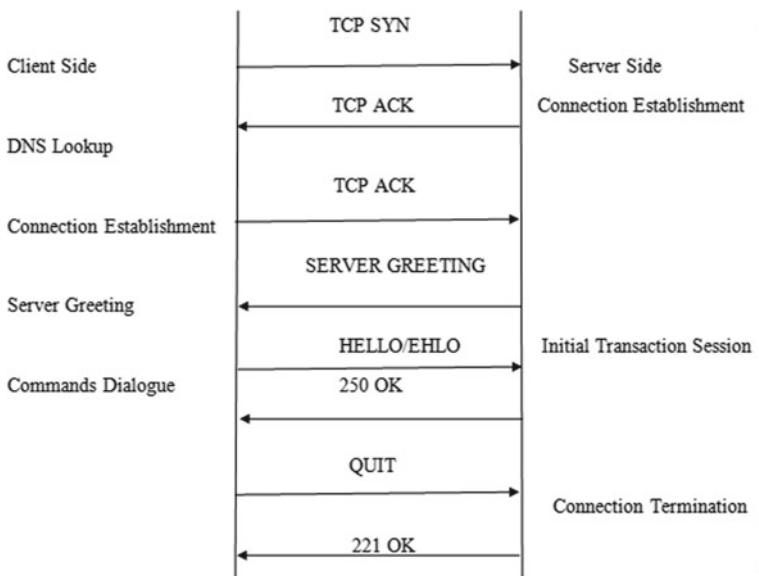


Fig. 14 SMTP procedure

5 Conclusion

TCP/IP is the backbone protocol suite of the Internet. How easily a script kid can use the TCP/IP's architectural flaw to manipulate all the packets flowing between two different users is alarming. In order to make TCP connection safe, different measures are suggested, but it is not an exhaustive list. The aim of this paper is to unearth the architectural flaw of the most important protocol suite in the world.

Using the filtering router and network design (Figs. 12 and 13), we can limit the spoofed IP packets and possibly remove the problem by deploying the design into an automated system in the future.

OS fingerprinting is a major threat to the current system and the only possible way to remove this flaw is to either use the fingerprint scrubber (Fig. 11) or limit ICMP request and reply packets.

SMTP flaw can be corrected by using an end-to-end encryption delivery mechanism as SSL/TLS. Once TLS and/or end-to-end encryption are configured, it is worthwhile to look at how to add further authentication methods to your emails. They are used to increase the deliverability of emails as well as to stop spoofing of your communications. The most popular techniques are SPF (Sender Policy Framework) and DKIM (Domain Key Identified Mail). By adding the IP addresses used to send emails on behalf of a specific domain to the DNS records, SPF may be used to identify a sender. Before sending a message, the receiving client verifies that the correct address was used and may reject a message if it finds any anomalies. The digital certificate DKIM, on the other hand, is provided along with an

email. It enables a message's receiver to confirm that the email's headers or contents were not altered (forged) during delivery. A message's ability to be delivered and its potential to bypass an inbox are both impacted by failed DKIM tests. The most advanced of the three techniques: SPF, DKIM & DMARC (Domain based Authentication Reporting & Conformance) makes use of the other two techniques to carry out further inspections. It is the only technique that, in addition to performing a test, may advise a receiving server on what to do if a message does not pass a check. It is beneficial to set up all three for domain. Finally, it is important to verify that SMTP server functions as it should.

It is evident that the backbone of the Internet (TCP/IP) is not secured, and it can be used to create havoc in the community. For future aspects, this study may help to reduce or eradicate the intrinsic problem of TCP/IP.

References

1. Kammai Z (2020) TCP/IP protocol security problems and defenses. In: 2020 international conference on intelligent computing and human-computer interaction (ICHCI), Sanya, China, pp 117–120. <https://doi.org/10.1109/ICHCI51889.2020.00033>
2. Shen Z-Y, Su M-W, Cai Y-Z, Tasi M-H (2021) Mitigating SYN flooding and UDP flooding in P4-based SDN. In: 2021 22nd Asia-Pacific network operations and management symposium (APNOMS), Tainan, Taiwan, pp 374–377. <https://doi.org/10.23919/APNOMS52696.2021.9562660>
3. Chen Y, Pan J, Yu D, Ma Y, Yang Y (2022) Retransmission-based TCP fingerprints for fine-grain IoT edge device identification. IEEE Trans Veh Technol 71(7):7835–7847. <https://doi.org/10.1109/TVT.2022.3169090>
4. Zhang C et al (2018) Towards an SDN-based integrated architecture for mitigating IP spoofing attack. IEEE Access 6:22764–22777. <https://doi.org/10.1109/ACCESS.2017.2785236>
5. Holst-Christensen B, Frøkjær E (2021) Security issues in SMTP-based email systems. In: 2021 14th CMI international conference—critical ICT infrastructures and platforms (CMI), Copenhagen, Denmark, pp 1–6. <https://doi.org/10.1109/CMI53512.2021.9663741>
6. Li Y, Jiang K (2012) Prospect for the future internet: a study based on TCP/IP vulnerabilities. In: 2012 international conference on computing, measurement, control and sensor network, Taiyuan, China, pp 52–55. <https://doi.org/10.1109/CMCSN.2012.14>
7. Osanaiye OA (2015) Short paper: IP spoofing detection for preventing DDoS attack in cloud computing. In: 2015 18th international conference on intelligence in next generation networks, Paris, France, pp 139–141. <https://doi.org/10.1109/ICIN.2015.7073820>
8. Increasing safety and robustness in traffic controlling circumstances using WSN. IJCRT 6(1):16–21 (2018). ISSN: 2320-2882
9. Malan GR, Watson D, Jahanian F, Howell P (2000) Transport and application protocol scrubbing. In: Proceedings of the IEEE INFOCOM 2000 conference, Tel Aviv, Israel, Mar 2000
10. Paxson V (1997) Automated packet trace analysis of TCP implementations. In: Proceedings of ACM SIGCOMM '97, Cannes, France, Sept 1997
11. Rashid S, Paul SP. Proposed methods of IP spoofing detection & prevention. IJSR. World University of Bangladesh, Dhanmondi, Dhaka, Bangladesh. ISSN: 2319-7064

Static Analysis Approach of Malware Using Machine Learning



Aman Raj Pandey , Tushar Sharma , Subarna Basnet ,
and Sonia Setia 

Abstract The malware is continually changing in today's Internet-dependent environment, with an estimated 560,000 new cases being produced each day. As a result, conventional detection technologies are no longer adequate, which presents a problem for security experts. Because it can learn and adapt to new threats, machine learning has become a potential method for enhancing security measures. This research suggests a machine learning-based approach for identifying different kinds of malware. The technology attempts to improve threat detection's effectiveness and reactivity. A dataset of 96,724 malware samples is used to evaluate the suggested system, and several machine learning methodologies are contrasted. The outcomes show how machine learning may enhance security systems' ability to combat continually changing malware threats.

Keywords Machine learning · Cybersecurity · Malware · Static analysis · Random forest · Gradient boosting · Support vector machine · Cryptojacking

1 Introduction

With the fast advancement of information technology in recent years, the exponential rise of malicious code has emerged as one of the most serious dangers to Internet security. According to Symantec, there has been an 82% spike in new order types, as well as an 80% increase in malware on Macs are also grown. Malware authors frequently use polymorphic and metamorphic methods to evade antivirus software and scanners, many traditional signature-based techniques are unable to detect malware and deter cybercriminals, and there is a high demand for a supple system to classify new malware samples as they are identified.

A. R. Pandey  · T. Sharma · S. Basnet · S. Setia

Department of Computer Science Engineering, Sharda University, Knowledge Park III, Greater Noida, India

e-mail: amanrajpandey.9717@gmail.com

S. Setia

e-mail: sonia.setia@sharda.ac.in

Many concepts have been offered by scholars to categorize distinct variants. Researchers utilize static analysis to evaluate a program's code without having to run it. The principle behind the static analysis is that it uses the data acquired to find commonalities in the instruction set. Dynamic analysis, on the other hand, works by running the code to make what the code performs more obvious. Advanced Persistent Threat (APT) operators have been exploiting long-standing security flaws for years without being detected. A defense based just on known threats is insufficient to maintain confidentiality, availability, and integrity.

Malware can take the form of a script, an executable binary, or any other piece of code with harmful intent. Malware's major goals are to obtain system access, disrupt system functions, cause a denial of service, steal private information, and destroy resources. Downloading genuine software from any website may result in the installation of malicious malware. Malware is most commonly encountered in cracked and pirated software. Malware is not simply executable scripts; they may also operate as malware downloaders, such as PDF and PHP links, which take control of the machine and download other malicious software to run on it. It cannot define certain software as harmful since it takes control of the system and performs some lawful tasks.

Malware may be detected with the use of machine learning [1]. Machine learning works by categorizing a collection of data into several categories. The term "classification" is used to distinguish between two types of files: benign and malicious. The random forest method, decision trees, and support vector machine are all used in this study. Many technologies have been created in recent years [2, 3]; however, they are unable to detect new Trojans, infections, or spyware attacks. In addition,

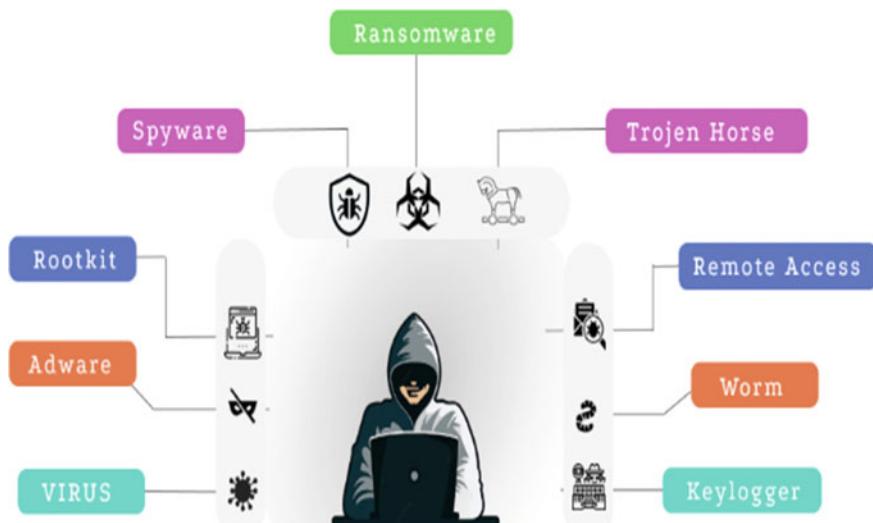


Fig. 1 Types of malware

previous solutions rely on non-machine learning algorithms that are incapable of detecting new types of harmful software. It has been discovered a dearth of insight into the meaning of why numerous distinct machine learning algorithms may accurately identify malware on particular datasets throughout our research. It is not always evident how machine learning models for malware detection deal with difficulties like false positives and false negatives. The applicability of multiple machine learning models is determined not only by the number of features but also by their qualities. As a result, feature selection and processing are critical for the development of any machine learning system, and approaches might be static or dynamic.

The study is divided into the following sections: Sect. 2 reviews previous research and literature; Sect. 3 presents information on the domain; Sect. 4 describes our experimental design; and Sect. 5 ends the paper and offers some ideas for future research.

2 Literature Review

Verma et al. [4] intend to investigate a variety of machine learning methods, such as Naïve Bayes, J48, random forest, multiclass classifier, and multilayer perceptron, to identify malware based on the permissions granted to various programs. Researchers have also said that malware is analyzed in two ways: static and dynamic analysis; however, the model given in this study analyzes the “Android-manifest.xml” which contains all of the permissions of the various apps. The authors looked at Google Play Store applications from 2015 to 2016. In the experiment, four datasets were employed, each of which consisted of a mix of “regular apps” and “malware apps.”

Kornish et al. [5] have offered several techniques for encoding malware binaries as images that may be used to best represent malware as an image. They have also integrated numerous malware (assembly code) families as hexadecimal format binaries. They attempted to concentrate on analyzing some virus patterns to construct a better machine model. Researchers turned malware binaries into grayscale images and then used a trained DCNN model to analyze the patterns.

Poonguzhali et al. [6] proposed that with the use of convolutional neural networks, this article provided a way to detect distinct malware strains via deep learning. The CNN is used to identify and extract features, while the support vector machine classifier is used to classify malware pictures. The classifier also mentions which malware families they belong to. CNN is made up of convolutional layers, which are a collection of various filters together, matrix layers in between the convolutional layers, and pooling layers in between the convolutional layers. When a certain picture’s input is supplied in the form of a matrix, the image matrix’s matrix is dot produced, which helps to get edges, color, and more details of the image.

Saadi et al. [7] proposed a hardware-assisted malware detection system with a complexity effective to state runtime. 2SMaRT classifies applications using a multi-class classification approach into either benign or one of the malware classes and then [8] uses a machine learning model that works well for each type of malware to

provide a runtime solution that solely uses hardware performance counters (HPCs). Walker and Sengupta [9] state that the detection of different machine learning algorithms is dependent on the training dataset utilized. Malicious and normal software use a variety of comparable calls, such as read, write, delete, and so on, at varying frequencies. These frequencies can be used to differentiate between malicious and non-malicious software.

Irshad et al. [10] proposed a dynamic malware investigation system based on machine learning techniques to recognize malware on Windows. For dangerous software, data was gathered from the VirusShare database, whereas for benign applications, regular Windows folders were used. After that, the dataset is sent to the Cuckoo sandbox program, which generates reports in JSON format.

Then comes the [11] feature extraction feature, which extracts five different types of features from the JSON file: DLL files, registry keys, directories, executable files, and API calls. The most important characteristic is then picked for training purposes using a genetic algorithm. It may be further customized by applying an assembly learning approach to improve generic malware detection performance.

Elkhawas and Abdelbaki [12] presented a methodology for malware identification based on opcode sequencing or trigrams. PE file properties have been exploited as detecting features by researchers. Researchers employed a text-mining strategy to make the process more resilient to combat polymorphism and metamorphism in this model. Machine learning algorithms are fed via opcode trigram sequencing. Support vector machine (SVM), a discriminative classifier model, was also utilized by the researchers. SVM is a classification method that determines whether the anticipated output belongs to the learned class category or not. The main task of this project is to extract opcode instructions and associated n-grams.

Vinaya Kumar et al. [13], presented a method that feature engineering process may be skipped by applying modern machine learning algorithms such as deep learning. This algorithm's performance is skewed by the training data. According to experts, these strategies must be evaluated separately to develop a new improved strategy for zero-day malware prevention. According to the author [14, 15], the training set of public and private datasets utilized in the experimental study are discontinuous and gathered on distinct timescales. They have [16] also presented a visual detection framework that is successful at identifying malware and is scalable, as well as a hybrid deep learning framework for real-time detection.

3 Algorithms and Methodology Used

3.1 Static Analysis

Only malware detection systems employ the method of static analysis to find harmful code without running it. It entails looking at the malware's internal structure, including its file format, header data, and code segments, to see if any dangerous

behavior is there. Static analysis often comprises a number of processes. The malware sample is first gathered and examined to determine the kind and format of its files [17], such as executable and linkable format (ELF) or portable executable (PE). The assembly code is then extracted from the sample after which the control flow and data flow of the code are examined to look for any potentially malicious activity. Static analysis may also entail looking at the malware's exported and imported functions as well as its system calls. This makes it easier to spot any efforts by the virus to interact with external networks or get access to system resources. The ability to do static analysis rapidly and without running the risk of installing malware on a system is one of its main benefits. As a result, it is a great strategy for locating known malware or swiftly classifying a lot of questionable files. Static analysis has its limits, though. It might not be able to recognize unidentified or zero-day malware since this kind of malware may have been created purposely to avoid being detected by conventional static analysis methods. Additionally, certain malware could conceal its behavior and avoid detection using sophisticated obfuscation techniques.

3.2 Dynamic Analysis

A crucial part of contemporary malware detection systems is dynamic analysis. These systems employ dynamic analysis, which examines a program or application's behavior while it is operating in order to spot any harmful or suspect activities. Dynamic analysis can expose the real behavior of the malware when it is executed on a system, making it particularly valuable for identifying previously undiscovered or zero-day malware. The system keeps track of a program's actions as it runs in a dynamic analysis environment, including any interactions with the network, file system, and system registry. The program's maliciousness is then determined by analyzing this data using a variety of methods.

Malware that is engineered to elude conventional detection techniques can be more easily found through dynamic analysis. Numerous malware programs are created with the express purpose of evading detection utilizing strategies like anti-debugging, anti-virtualization, or anti-sandboxing. By examining the behavior of the virus in a controlled environment and while being protected from any anti-analysis tactics, dynamic analysis can assist in overcoming these difficulties.

3.3 Random Forest Classifier

A prevalent machine learning approach for classification problems is random forest classifier. It is an ensemble learning technique that creates numerous decision trees and combines their categorization outcomes. A random portion of the training data and a random subset of the features are used to build each decision tree in a random forest classifier. This aids in lowering overfitting and improving model accuracy.

Each decision tree is constructed separately during training, with no knowledge of the other trees in the forest.

Each decision tree in the forest generates a prediction based on its own characteristics and training data subset when a new data point is presented to the model. A simple majority vote or weighted averaging is then used to combine the forecasts of all the decision trees in the forest to arrive at the final prediction. Compared to other classification techniques, random forest classifiers provide a number of benefits. They cope well with unbalanced datasets and can handle a high number of input variables, including category and numerical data. They can manage missing data and are also rather simple to use and comprehend.

3.4 Gradient Boosting

A machine learning approach called gradient boosting is used to increase the predictability of outcomes in supervised learning tasks like regression and classification. It functions by sequentially creating weak models (often decision trees), each of which attempts to fix the mistakes caused by the preceding model. One of the numerous uses of gradient boosting, which is a potent method, is malware detection systems. Gradient boosting can be used to categorize whether or not a given piece of code or file is dangerous in the context of malware detection. A gradient boosting model may be taught to correctly categorize new files that it has never seen before by being trained on a sizable dataset of both harmful and benign files. A gradient boosting model may be taught to correctly categorize new files that it has never seen before by being trained on a sizable dataset of both harmful and benign files.

The capacity of gradient boosting to manage unbalanced datasets is one of the key benefits of utilizing it for malware detection. Because there are frequently more benign files than dangerous ones, it might be challenging for conventional machine learning algorithms to appropriately categorize new files. Gradient boosting, on the other hand, can get around this problem by giving the incorrectly classified samples greater weight in each iteration of the model, which lessens the bias toward the dominant class. The capability of gradient boosting to manage high-dimensional feature spaces makes it an additional benefit when used for virus detection.

To effectively categorize files, malware detection systems often rely on a huge number of characteristics, and classical machine learning algorithms may find it challenging to handle these high-dimensional spaces. Gradient boosting, on the other hand, is able to manage these complicated feature spaces by creating decision trees that concentrate on the most crucial characteristics and disregard the inconsequential ones (Fig. 2).

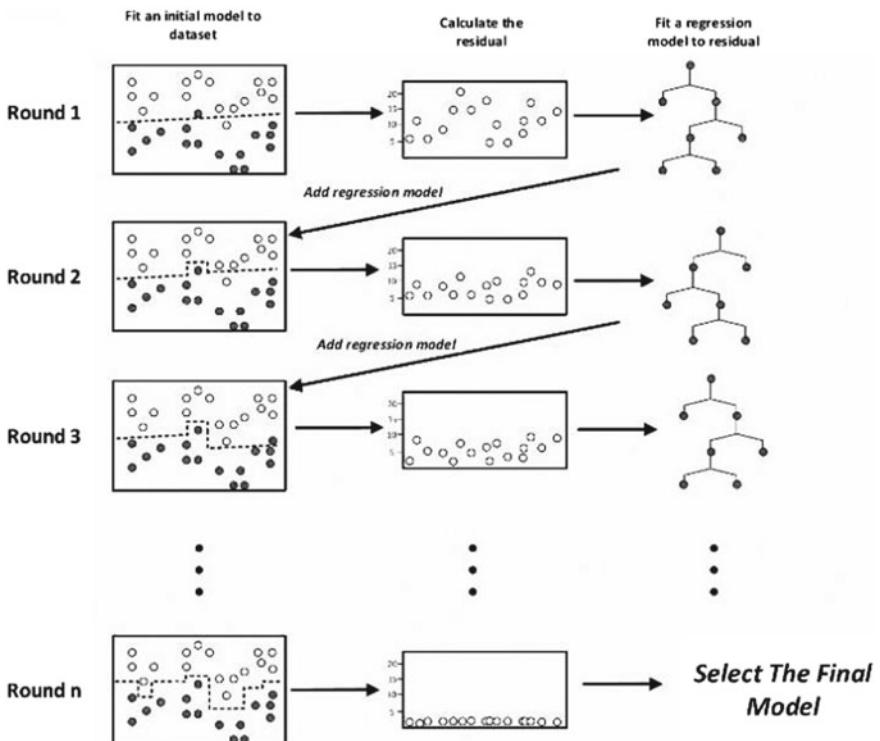


Fig. 2 Gradient boosting process [18]

4 Proposed Methodology

In this section, proposed system has been elaborated; Fig. 3 shows the proposed model. In this model, dataset of labeled samples is used, in order to train a machine learning model for malware identification. In this instance, there are 96,724 different types of malware samples in the dataset. Preprocessing of the data has been done with the Pandas package and removing any samples that are marked as legitimate files.

The model will then be trained using a set of features that have been chosen in the next phase. There are 56 features in the first feature set that come from both malicious and legitimate files. The model chooses 12 characteristics from the legitimate samples and 54 features from the malicious samples using the extreme tree classifier technique to differentiate between the two types of files.

In Fig. 3, the process of system has been depicted; in the first phase, portable executable (PE) file is given to model; it consists of headers, sections, import table, export table, resources, binary files, and dynamic link libraries (DLL). While doing static analysis examining these provides valuable insight, capabilities, and possible attack vectors about the malware.

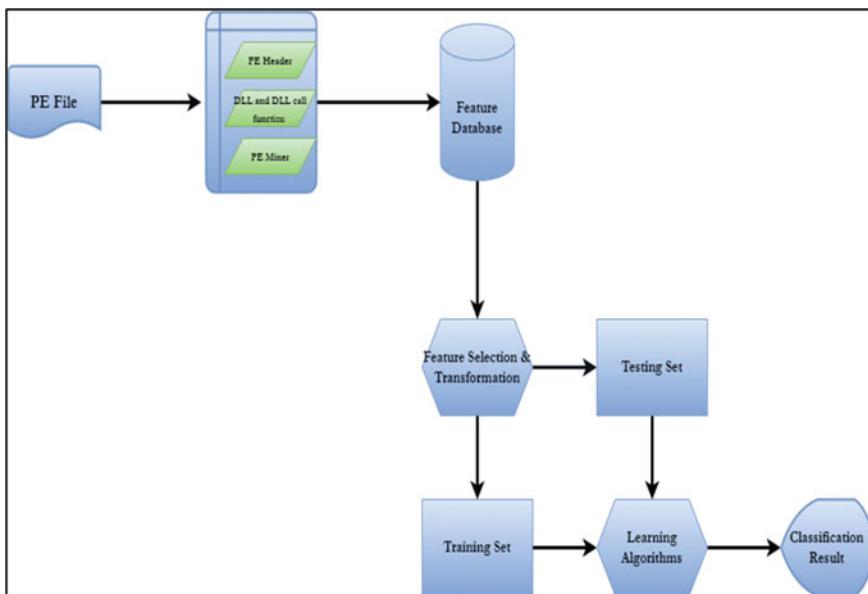


Fig. 3 Proposed model

In the second phase after PE files components have been analyzed, it is added to feature database. A collection of properties or traits that have been taken from numerous malware samples make up the feature database. These characteristics may be used to recognize and categorize malware and may include file size, import and export capabilities, section names and sizes. Features from the PE file are extracted during analysis and compared to those already present in the feature database to see whether they match known malware or display questionable behavior. If a match is discovered, the virus may be identified, and the proper steps, such as quarantining or destroying the file, can be done.

Further in the third phase, it goes through feature selection phase which is helpful in reducing the dimensionality of the feature space, removing redundant or irrelevant features, and enhancing the precision and efficiency of the system. Filtering method is used for performing the selection in which the features are ordered according to statistical or other criteria, and the highest-ranked characteristics are chosen.

After the features have been chosen, they might need to be translated into a format that machine learning algorithms can use. In order to do this, the features must be normalized to a standard distribution, scaled to a common range, or encoded as binary or numerical variables in the case of categorical characteristics.

In the fourth phase, small subset of data is chosen from feature database for training of the model and then it gets splitted into training and testing sets during training phase the model learns to recognize patterns to classify new samples while test. After this model passes through different classification algorithms, i.e., random forest classifier and support vector machine. These algorithms help to optimize the

model making it perform better, and in last by applying the gradient boosting on model, it is able to get 98.84% score.

5 Results and Analysis

In this study, a machine learning model has been developed which tends to analyze any malware based on its static features in its self-contained environment; this is more accurate and reliable than traditional signature-based antivirus software by analyzing the anomalies in the code that may not be detectable by static rules implemented by traditional antivirus software. Moreover, it can be trained with large dataset samples available and easily learn to identify based on previously encountered malware.

This system can be implemented in real-time by utilizing an application programming interface (API) and providing alerts to security systems. As per SonicWall [19] 2023 report, Fig. 4 depicts the number of malware reported worldwide; the data in the last three years seems to be depleted but crimes like cryptojacking rose by 43%; IoT-based malware attacks increased by 87% with directly shows with the change in technology trends new sectors are getting more targeted than previous technologies. Figure 5 shows the increase in ransomware attacks; there are already indicators of a possible turnaround. After ransomware reached its lowest attack volume in June, the trend line started to reverse in July 2020. Attacks increased by double between September and October, increasing Q4 ransomware totals to 154.9 million.

New research [20] revealed a new category of Linux platform malware known as “Shikitega Malware” that affects IoT devices and traditional servers. Researchers

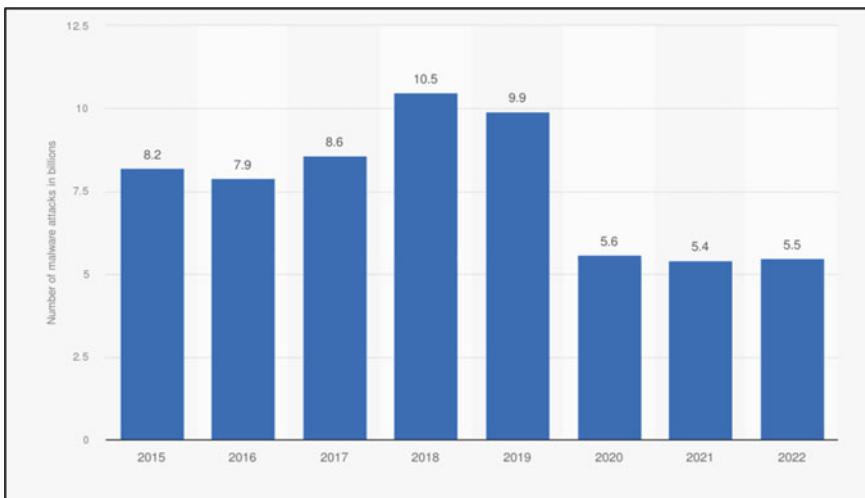


Fig. 4 Malware attacks worldwide from 2015 to 2022

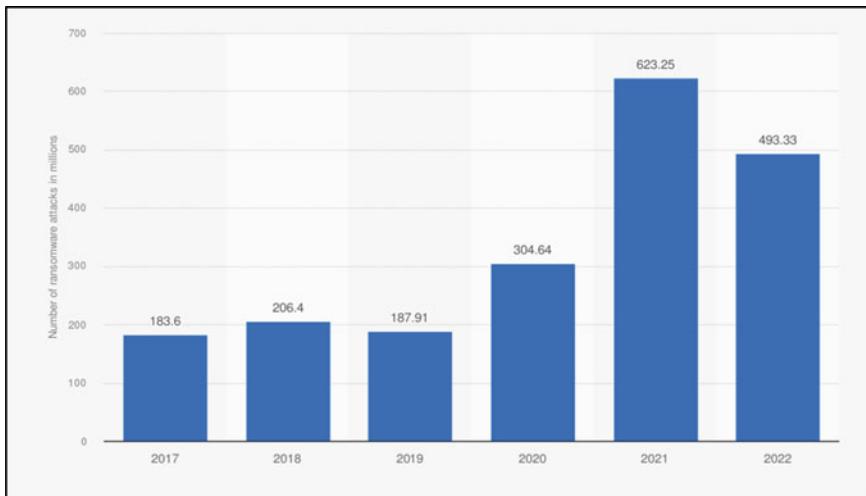


Fig. 5 Ransomware attacks from 2017 to 2022

claimed that it is transferred by a process of multistage infection chain by using sophisticated encoding named polymorphic encoding; it also targets cloud services to host command and control (C2C) servers. Research is still going on to know the objective of this new category of malware; the possibility is that it might use system power for cryptojacking for mining the Monero cryptocurrency. This malware also downloads an advanced Metasploit package, Mettle, which has the power and capabilities to gain access control of the webcam, credentials, etc. As stated by researchers, this polymorphic encoding is that it follows a chain pattern of infections which is that it works in decoding loops structure, one loop decodes the next loop and the process continues till the final shell payload is decoded and executed.

Another recent attack from the malware family is Lampion Malware [21] which has attacked WeTransfer, a free file-sharing service, because of its less maintainability it is easy and vulnerable to attacks and it does not raise any alerts for infected link sharing. Security firm Cofence observed that Lampion malware attackers are sending phishing emails by the name of a company domain account that has been compromised, which urges the user to download and “Proof of Payment” file in ZIP format containing the Virtual Basic Script (VBS) file, as the user executes the file attack begins.

Upon execution, it enables a WScript deamon process which connects itself to the process chain, and track user working and frames and fake login page on real login access pages of websites like Amazon and some banking services and steal the password and ids of the users and ZIP the data and send it to the attacker, all unknown from the user.

Figure 6 shows what features it has chosen by the model; the system chooses the feature based on its training and static method, in which the model analyzes the structure and functionality of the malware.

Fig. 6 Selection of features

1 Characteristics	0.13846587369999333
2 Machine	0.11651177540705746
3 SectionsMaxEntropy	0.10843105279543183
4 SizeOfStackReserve	0.1066364579969188
5 DllCharacteristics	0.07590470725559427
6 VersionInformationSize	0.07513224824380975
7 ImageBase	0.06691640891958586
8 SizeOfOptionalHeader	0.05317746464891953
9 Subsystem	0.04003786501782636
10 ResourcesMaxEntropy	0.028132881326436526
11 SectionsMinEntropy	0.025374117303045493
12 MajorSubsystemVersion	0.021160225472253408

Further two algorithms, random forest classifier, SVM is used to optimize the system which gives score of 99.37% later gradient boosting algorithm is used, which coupled with prior builds of models and made the system even more reliable this time we have achieved the final efficiency of 98.76%, which is better than the previous build as it has been optimized.

Table 1 shows the details of the efficiency attained by the system. It has the false negative of 0.91 and false positive of 0.51 of cases mentioned. Applying the gradient boosting makes the model better by reduction of errors, makes able to handle complex relationships of variables, and becomes more prominent to handle difficult to learn instance.

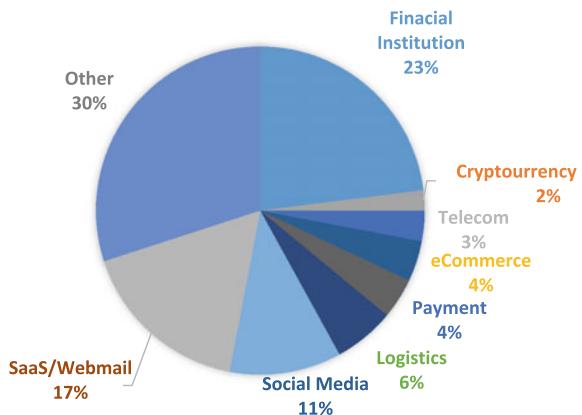
Figure 7 [19] depicts the targeted sectors in third quarter of the year 2022; it is discovered that phishing attacks targeting the financial sector, which includes banks, continued to be the most common kind of assaults, making at 23.2% of all phishing attacks, down from 27.6% in second quarter.

Despite attacks on retail/e-commerce websites declined by 4.1%, down from 14.6% in quarter 1, crimes against web-mail and software-as-a-service (SAAS) providers remained stable. After varying from 8.5% of all assaults in the fourth quarter of 2021 to 15.5% in the second quarter of 2022, phishing against social media firms trended downward. As the cryptocurrency market was shaken by declining prices, phishing attacks against cryptocurrency targets, such as cryptocurrency exchanges and wallet providers, decreased from 4.5% in second quarter to 2.0% in third quarter.

Table 1 Output given by system

Parameter	Output (in percent)
Without applying gradient boosting	99.37
With gradient boosting	98.76
False positive	0.51
False negative	0.91

Fig. 7 Sectors targeted in third quarter of 2022



6 Conclusion and Future Scope

This paper details the efforts to create a machine learning model that makes use of the static characteristics of malware samples. Additionally, in order to better understand the need for secure and privacy safe software, analysis of current attack vectors and trends has been performed. The objective is to develop software that both businesses and individuals can use to protect their working environments and make the Internet a safer place, especially for kids. This will incorporate as the machine learning model which is created as a distinct feature in the software. We want to increase this project's functionality and carry out more research, toward increasing the ability of software, which will enhance cybersecurity posture by using machine learning techniques and giving consumers a complete tool to safeguard themselves and digital assets like personally identifiable information (PII).

References

1. Kim T, Kang B, Rho M, Sezer S, Im EG (2018) A multimodal deep learning method for android malware detection using various features. *IEEE Trans Inf Forensics Secur* 14(3):773–788
2. Karimipour H, Dehghantanha A, Parizi RM, Choo KKR, Leung H (2019) A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids. *IEEE Access* 7:80778–80788
3. Luo S, Peng A, Zeng H, Kang X, Liu L (2019) Deep residual learning using data augmentation for median filtering forensics of digital images. *IEEE Access* 7:80614–80621
4. Varma A, Ravi Kiran P, Raj KP, Subba Raju KV (2017) Android mobile security by detecting and classification of malware based on permissions using machine learning algorithms. In: 2017 international conference on I-SMAC (IoT in social, mobile, analytics and cloud) (I-SMAC). IEEE
5. Kornish D et al (2018) Malware classification using deep convolutional neural networks. In: 2018 IEEE applied imagery pattern recognition workshop (AIPR). IEEE

6. Poonguzhali N et al (2019) Identification of malware using CNN and bio-inspired technique. In: 2019 IEEE international conference on system, computation, automation and networking (ICSCAN). IEEE
7. Sayadi H et al (2019) 2SMaRT: a two-stage machine learning-based approach for run-time specialized hardware-assisted malware detection. In: 2019 design, automation & test in Europe conference & exhibition (DATE). IEEE
8. Soni H, Arora P, Rajeswari D (2020) Malicious application detection in android using machine learning. In: 2020 international conference on communication and signal processing (ICCSP). IEEE
9. Walker A, Sengupta S (2019) Insights into malware detection via behavioral frequency analysis using machine learning. In: MILCOM 2019–2019 IEEE military communications conference (MILCOM). IEEE
10. Irshad A et al (2019) Feature optimization for run time analysis of malware in windows operating system using machine learning approach. In: 2019 42nd international conference on telecommunications and signal processing (TSP). IEEE
11. Fatima A et al (2019) Android malware detection using genetic algorithm based optimized feature selection and machine learning. In: 2019 42nd international conference on telecommunications and signal processing (TSP). IEEE
12. Elkhawas AI, Abdelbaki N (2018) Malware detection using opcode trigram sequence with SVM. In: 2018 26th international conference on software, telecommunications and computer networks (SoftCOM). IEEE
13. Popov I (2017) Malware detection using machine learning based on word2vec embeddings of machine code instructions. In: 2017 Siberian symposium on data science and engineering (SSDSE). IEEE
14. Vinayakumar R et al (2019) Robust intelligent malware detection using deep learning. IEEE Access 7:46717–46738
15. Lee I, Roh H, Lee W (2020) Encrypted malware traffic detection using incremental learning. In: IEEE INFOCOM 2020—IEEE conference on computer communications workshops (INFOCOM WKSHPS), July 2020. IEEE, pp 1348–1349
16. Sharma P, Siddanagaiah U, Kul G (2020) Towards an AI-based after-collision forensic analysis protocol for autonomous vehicles. In: 2020 IEEE security and privacy workshops (SPW), May 2020. IEEE, pp 240–243
17. Pichan A, Lazarescu M, Soh ST (2020) A logging model for enabling digital forensics in IoT, in an inter-connected IoT, cloud eco-systems. In: 2020 fourth world conference on smart trends in systems, security and sustainability (WorldS4), July 2020. IEEE, pp 478–483
18. Azizi R, Noroozian R (2021) Islanding detection in distributed energy resources based on gradient boosting algorithm, 12 Jan 2021. <https://doi.org/10.1049/rpg2.12040>
19. SonicWall Capture Labs Inc. (2023) 2023 SonicWall cyber threat report. <https://www.sonicwall.com/2023-cyber-threat-report/>
20. Goodin D. (2022) New Linux malware combines unusual stealth with a full suite of capabilities with polymorphic encoding and a multistage infection chain, Shikitega is hard to detect, 10 Sept 2022. <https://arstechnica.com/information-technology/2022/09/new-linux-malware-combines-unusual-stealth-with-a-full-suite-of-capabilities/>
21. Toulas B (2022) Lampion malware returns in phishing attacks abusing WeTransfer. BleepingComputer.com Logo, 9 Sept 2022. <https://www.bleepingcomputer.com/news/security/lampion-malware-returns-in-phishing-attacks-abusing-wetransfer/>

Cyber-Attack Analysis Using Vulnerability Assessment and Penetration Testing



Vijayashree Budyal, A. V. Vaibhav, C. U. Akshay, Naveen Ishika, and Gaonkar Unnathi

Abstract Networks and systems like computers and mobile devices are not secure enough and are vulnerable to numerous types of cyber-threats. The frequency of cyber-threats is on the rise each year. The process of identifying the vulnerabilities in the systems and networks is called vulnerability assessment. Vulnerability assessment helps in providing necessary solutions to mitigate cyber-threats. It involves identification of vulnerabilities like network, physical and application vulnerabilities, and the assessment of the risks associated with these vulnerabilities. This paper introduces a novel approach for conducting cyber-attack analysis by leveraging vulnerability assessment and penetration testing methodologies. The proposed method utilizes a range of tools available in Kali Linux, integrated with a user-friendly menu-based interface and Python modules. Comparative analysis is performed between the outcomes of the proposed approach and the built-in tools of Kali Linux. Notably, the findings reveal significant improvements in several crucial areas, including system modifications, network and website testing, sniffing and spoofing, as well as information gathering. The incorporation of these enhancements enhances the overall effectiveness and efficiency of cyber-attack analysis, providing valuable insights for improved security measures.

Keywords Cyber-security · Cyber-attack prevention · Vulnerability assessment · Penetration testing

1 Introduction

Cyber-threats/attacks can be defined as a series of actions that are performed by people who typically use malicious software to gain access to a person's or organization's computer or network. They can also cause malfunctioning of computer

V. Budyal · A. V. Vaibhav · C. U. Akshay · N. Ishika (✉) · G. Unnathi
Sai Vidya Institute of Technology, (Affiliated to Visvesvaraya Technological University,
Belagavi), Bengaluru, India
e-mail: ishikanaveen1707@gmail.com

systems, resulting in loss of data, system crashes, and decreased performance. Cyber-attacks can be performed by an individual or a group from any given location. Those who launch these types of cyber-attacks are called cybercriminals or hackers. To overcome these weaknesses, we can go by the approach of utilizing the Kali Linux tools. In this paper, we have presented a menu-based interface for penetration testing and vulnerability assessment along with Python modules.

1.1 Literature Review

Pelivani and Cico [1] This paper presents a comparison of web application automation testing tools. The methodology used in this study involves selecting popular automation testing tools and evaluating their features, performance, ease of use, and compatibility with web applications. The limitations of the study include the limited number of tools considered and the specific focus on web applications. To overcome these limitations, future research can explore a broader range of tools and extend the evaluation to other types of applications [2]. The focus of this paper is on penetration testing and vulnerability assessment. The authors propose a solution implementation for these tasks. The methodology involves designing and developing a portable testing platform that can be used for vulnerability assessment and penetration testing. The limitations of the paper include the specific focus on a portable solution and the need for further validation and testing of the proposed solution. Kavya Rani and Soundarya [3] presents a comprehensive analysis of various cyber-attacks. The methodology used in this study involves reviewing and analyzing different types of cyber-attacks, including their characteristics, techniques, and potential impact [4]. This paper centers on the examination of outcomes from security assessments conducted on wireless networks using Kali Linux. The methodology involves conducting experiments using Raspberry Pi devices and Kali to assess the security of wireless networks. The limitations of the study include the specific focus on Raspberry Pi devices and the need for broader validation of the findings. To address these limitations, future research can involve using different hardware platforms and conducting comparative evaluations with other security assessment tools [5]. This paper presents an analysis of web application security aspects in comparison. The methodology used in this study involves reviewing the current trends in web application security and identifying key security parameters. The limitations of the study include the focus on specific security parameters and the absence of a comprehensive evaluation of existing web application security techniques. To overcome these limitations, future research can include a broader range of security parameters and perform comparative evaluations of different security techniques [6]. The focus of this paper is on information gathering and tools for pen testing. The methodology involves analyzing existing information gathering techniques and tools used in penetration testing. The limitations of the paper include the limited scope of information gathering techniques covered and the need for further empirical validation of the tools.

To address these limitations, future research can explore additional information gathering techniques and conduct empirical studies to validate and compare different tools [7]. This paper focuses on implementing and comparing different password cracking tools. The methodology involves selecting popular password cracking tools and evaluating their performance and effectiveness in cracking passwords. The limitations of the study include the limited number of tools considered and the absence of a comprehensive evaluation of all password cracking techniques.

Overall, these papers contribute to the field of cybersecurity by providing insights into automation testing tools, vulnerability assessment and penetration testing, cyber-attacks, wireless network security, web application security, information gathering, and password cracking tools. However, each paper has its own limitations, which can be addressed through future research efforts to enhance the understanding and development of robust security solutions.

1.2 *Contribution*

In this work, we conduct vulnerability research which is an important contribution to a vulnerability assessment and penetration testing tool project as it helps to identify new and existing vulnerabilities in software and systems. By identifying vulnerabilities, vulnerability assessment and penetration testing tools can be developed to detect and exploit these vulnerabilities, which help organizations to better secure their systems. Vulnerability researchers typically require a deep understanding of software and systems architecture, as well as knowledge of common attack vectors and methods. The contributions for this paper are to (i) provide accurate reports on risk assessment using custom tools, (ii) implement a menu-driven approach, and (iii) reduce the risk of the system getting compromised.

The remainder of the paper is structured as follows: The second section describes the anticipated work on vulnerability assessment and penetration testing tools. Section III presents the findings, while Section IV closes the study.

2 Proposed Work

This section describes system architecture, vulnerability scanner, and utilization of vulnerability assessment and penetration testing tools.

2.1 System Architecture

This vulnerability assessment and penetration testing system architecture involves multiple layers of protection for different components of the system. System components are shown in Fig. 1.

Threat sources: These are the entities that pose a potential threat to the security of an organization. Threat sources can be internal or external. Identifying potential threat sources is an important part of VAPT, as it helps organizations understand where their security risks lie.

Threat events: These are incidents or occurrences that can cause harm to an organization's assets or disrupt its operations. Examples of threat events include malware infections, denial-of-service attacks, and data breaches. VAPT helps organizations identify potential threat events and develop strategies to prevent them.

Countermeasures: These are measures put in place to protect an organization's assets from potential threats. Countermeasures can include security policies, access controls, firewalls, and intrusion detection systems. VAPT helps organizations identify the most effective countermeasures for their particular security needs.

Security testing: This is the process of assessing an organization's security posture to identify vulnerabilities and potential threats. Security testing can include a range of techniques, such as vulnerability scanning, penetration testing, and social engineering.

Vulnerabilities: The weakness of an organization's security can be exploited by threat sources. Hence, vulnerability assessment and penetration testing can help these organizations identify and address these vulnerabilities to prevent any sort of potential exploitation.

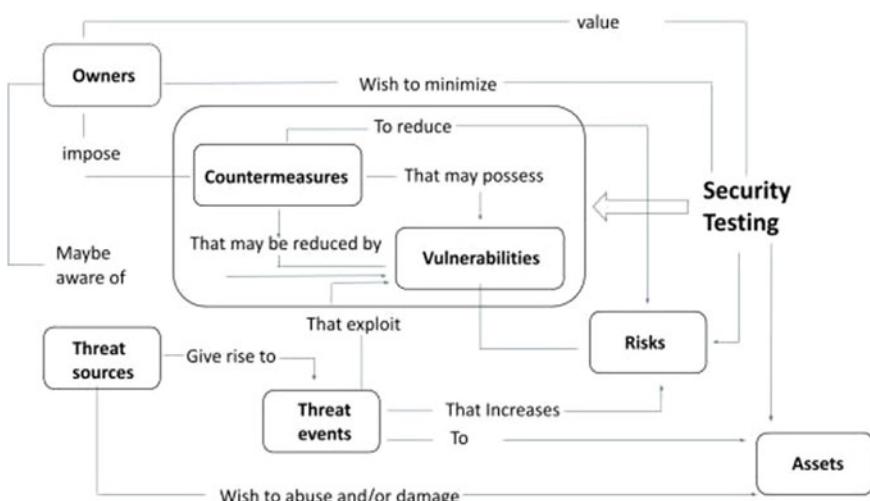


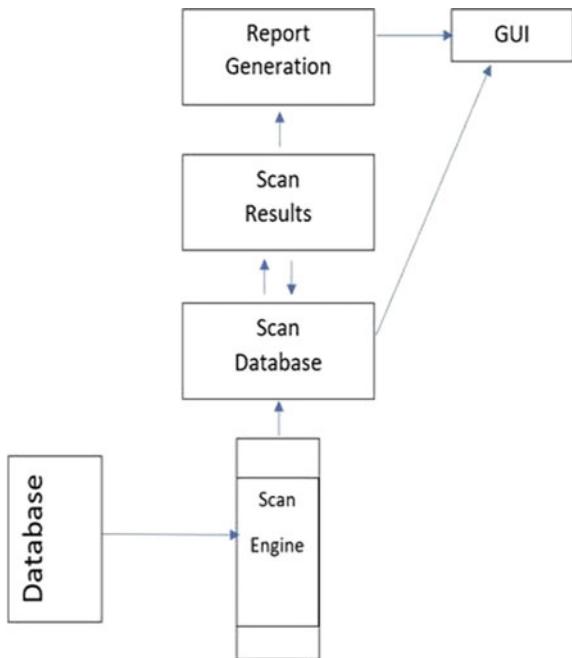
Fig. 1 System architecture

2.2 Vulnerability Scanner

A vulnerability scanner is a tool that is designed to automatically detect security vulnerabilities in computer systems, networks, and applications. The tool works by scanning the target system for known vulnerabilities and identifying any weaknesses that can be exploited by attackers. Vulnerability scanners use a database of known vulnerabilities, along with various techniques to identify and exploit them. The scanner usually operates in two modes: active and passive. In active mode, the scanner sends packets to the target system and analyzes the responses to identify vulnerabilities. In passive mode, the scanner listens for traffic on the network and identifies vulnerabilities based on the traffic that it observes.

Once the vulnerabilities have been identified, the scanner will typically generate a report that provides details about each vulnerability, including its severity. The report can be used by security professionals to prioritize and plan remediation activities as mentioned in Fig. 2.

Fig. 2 Vulnerability scanner



2.3 Utilization of Vulnerability Assessment and Penetration Testing

Penetration testing and vulnerability assessment tools are used in identifying and testing vulnerabilities in computer systems, networks, and applications. These tools are used by security professionals to proactively identify and address security risks and vulnerabilities and to ensure that systems are protected against potential attacks. Vulnerability assessment tools are designed to scan computer systems and networks for known vulnerabilities. These tools can perform automated scans of systems and applications, identify vulnerabilities, and generate reports that provide details about each vulnerability, including its severity and recommended remediation steps. Vulnerability assessment tools are typically used to identify weaknesses in systems before they are exploited by attackers. Penetration testing tools, on the other hand, are used to simulate real-world attacks on systems and networks. Penetration testing tools can identify weaknesses in systems that may not have been identified by vulnerability assessment tools and can help organizations to determine whether their security controls are effective in preventing attacks. Stages of vulnerability assessment and penetration testing can be seen in Fig. 3. Vulnerability assessment and penetration testing tools play a critical role in identifying potential security risks and vulnerabilities that could be exploited by attackers. By proactively identifying and addressing these risks, organizations can reduce the likelihood of successful attacks and minimize the potential impact of security incidents. Overall, vulnerability assessment and penetration testing tools are essential components of any effective security testing program. By using these tools to identify and address vulnerabilities, organizations can improve their security posture, reduce the risk of successful attacks, and comply with regulatory requirements and industry best practices (Fig. 3).

Fig. 3 Stages of penetration testing and vulnerability assessment

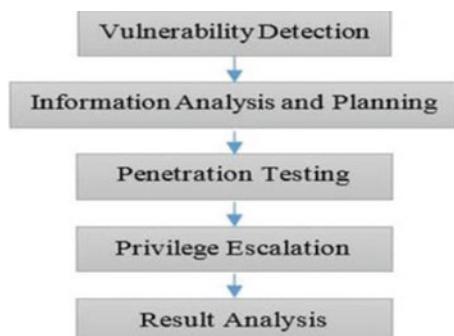
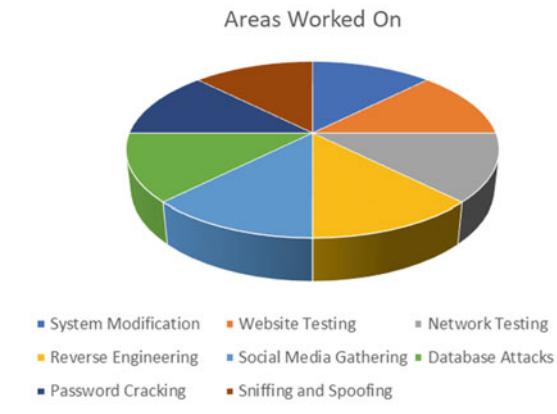


Fig. 4 Graph indicating areas worked on



3 Results

After comparing the results with [8], improvements were observed in various areas including system modifications, network and website testing, sniffing and spoofing, and information gathering.

We propose a vulnerability assessment and penetration testing tool that operates through a menu-based interface along with Python modules as shown in Fig. 5. Compared to the tool evaluated in [8], the tool has been enhanced to provide improved functionality and ease of use with this menu-based interface. The tool performs comprehensive scans on the designated system, identifying both prevalent and intricate security risks associated with the targeted network.

Our application offers a variety of categories for checking vulnerabilities and performing penetration testing on the selected system or networks. as shown in Fig. 6. We have carefully chosen and implemented a selection of tools that are user-friendly and efficient, resulting in impressive accuracy in comparison with [8]

By using BeEF as shown in Fig. 7, a command can be written to trigger an alert on the target's browser, and malicious URLs can be included to the Plugin URL.

Zenmap is a free ASCII text file application that simplifies Nmap for inexperienced users while providing sophisticated options for experienced users. Scans are kept as profiles to allow for simple recursive runs.

Wifite is a wireless assault tool that is mostly used for penetration testing. It is dependable in terms of wireless injections and testing the capacity of a network to handle packets. Its automation makes it an ideal tool for this main tool's efficient operation.

One of the tools chosen is the mac changer as shown in Fig. 8, which offers various options for modifying system configurations, making it easier to revert to the default device setups.



Fig. 5 Home screen

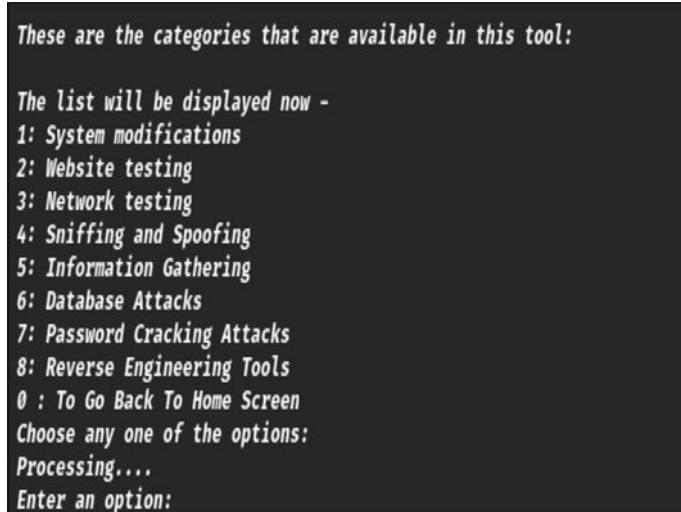


Fig. 6 Categories made available

MITMf provides active packet filtering and manipulation, which allows users to switch between any type of traffic or protocol. Users can make modifications to the configuration file while the tool is running, and the changes will be propagated across the framework.

**Fig. 7** BeEF tool

```

The options available in System modification category are:
The list will be displayed now - 

1.MAC Changer

Press 1 to execute MAC changer or press 2 to install MAC Changer else press 0 to go back to the main menu
1
Please enter the interface to change the address of (eth0 or wlan0):
Interface:eth0
Warning!!!! This will change your MAC address to a random MAC address
Current MAC: 8e:a0:56:61:49:c0 (unknown)
Permanent MAC: 08:00:27:0e:34:8d (CADMUS COMPUTER SYSTEMS)
New MAC: 1a:ec:e7:5f:5b:c7 (unknown)
MAC address successfully changed to random address
Errors:
None
The password has been cracked and displayed
  
```

Fig. 8 Mac changer

Sherlock is a tool which is used to find user names on social media, revealing multiple accounts created by the same person using the same screen name or username as shown in Fig. 9.

Osintgram is an OSINT tool and is used for Instagram info gathering, analyzing, and running reconnaissance missions.

SQL injection, or SQLI, is a technique that uses SQL code to manipulate backend data to access information not intended for display.

```
The options available in Social Media Information Gathering category are:
The list will be displayed now -
1.User name database collection
2.Instagram information gathering

Press 1 to gather information or Press 2 to Instagram information gathering else press 0 to go back to the main menu
1
This module is implemented by using the works of Sherlock
Enter the name of the person to check their registrations on various databases:vaibhav
[*] Checking username vaibhav on:

[*] 8tracks: https://8tracks.com/vaibhav
[*] 9GAG: https://www.9gag.com/u/vaibhav
[*] About.me: https://about.me/vaibhav
[*] Academia.edu: https://independent.academia.edu/vaibhav
[*] Airbit: https://airbit.com/vaibhav
[*] Airlines: https://www.airliners.net/user/vaibhav/profile/photos
[*] AllMyLinks: https://allmylinks.com/vaibhav
[*] Amino: https://aminoapps.com/u/vaibhav
[*] Anilist: https://anilist.co/user/vaibhav/
```

Fig. 9 Sherlock tool

John the Ripper has proven effective in deciphering passwords of diverse complexities within this sensitive information, showcasing its utility in password analysis. Hence, this can be used for any organisation which includes sensitive information. JTR is available in three modes: single crack, word list, and progressive. The single crack mode is the most efficient and effective method for cracking a whole password file.

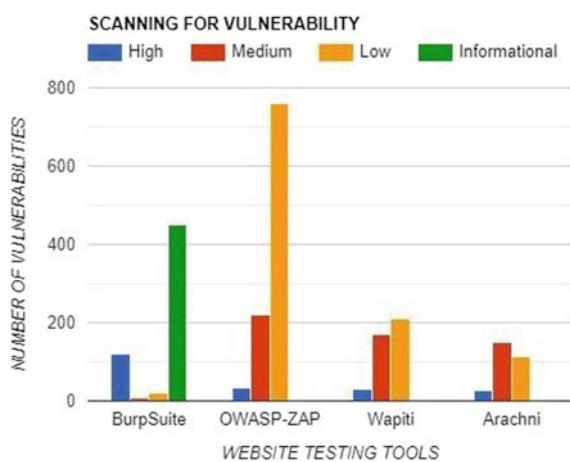
APK tools can decrypt resources back to their original form and recreate them after changes. Its project-like file structure and automation make it easier to work with supporting apps.

ZAP tool has a user-friendly interface, with various features such as spidering, active and passive scanning, and advanced interception tools as shown in Fig. 10. It has extensive documentation and an active user community that supports the tool and provides regular **updates**.

As shown in Fig. 11, our proposed work containing the OWASP-ZAP can be considered as the best tool for website testing since it can scan for high and medium risks in reasonable amounts. We can also observe that it can scan the most amount of low vulnerabilities and hence help in mitigating low risks compared to other tools as mentioned in [5].

**Fig. 10** ZAP tool

Fig. 11 Graph indicating amount of vulnerabilities scanned by various website testing tools



4 Conclusion

This paper addresses the importance of utilizing penetration testing and vulnerability assessment tools in cyber-attack prevention. With the increasing frequency and complexity of cyber-attacks, organizations need to adopt proactive measures to identify and address vulnerabilities in the systems and applications. A penetration testing and vulnerability assessment system architecture can assist organizations in assessing potential vulnerabilities and taking necessary remedial actions to prevent cyber-attacks. This system architecture covers different system components, including system services, applications, file systems, device drivers, libraries, system calls, and networking, and employs a range of tools and techniques to ensure comprehensive coverage of potential security risks. The implementation of penetration testing and vulnerability assessment tools can aid organizations in safeguarding

their data, maintaining their reputation, and ensuring the ongoing integrity of their operations in the face of growing cyber-threats. Therefore, the use of penetration testing and vulnerability assessment tools in cyber-attack prevention is crucial for maintaining the security and stability of organizations' systems and data.

Future work includes the development of more advanced and sophisticated penetration testing and vulnerability assessment tools, with enhanced capabilities such as network mapping scanning and reporting for identification and addressing complex security vulnerabilities such as advanced malware, supply chain attacks, and fileless malware. Additionally, there is scope for further research into the integration of penetration testing and vulnerability assessment tools with other cybersecurity technologies, like artificial intelligence and machine learning, to provide more comprehensive and proactive protection against cyber-threats.

References

1. Pelivani E, Cico B (2021) A comparative study of automation testing tools for web applications. In: 10th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, IEEE, pp 1–6
2. Pandey R, Jyothindar V, Chopra UK (2020) Vulnerability assessment and penetration testing. A portable solution implementation. In: 12th international conference on Computational Intelligence and Communication Networks (CICN), IEEE, Bhimtal, India, pp 398–402
3. Kavya Rani SR, Soundarya BC (2022) Comprehensive analysis of various cyber attack. In: IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), Mysore, India
4. Delija D, Petrović Ž, Sirovatka G, Žagar M (2021) An analysis of wireless network security test results provided by Raspberry Pi devices on Kali Linux. In: 44th international convention on information, communication and electronic technology (MIPRO), IEEE, Opatija, Croatia
5. Shahid J, Hameed MK, Javed IT, Qureshi KN, Ali M, Crespi N (2022) A comparative study of web application security parameters: current trends and future directions. *Appl Sci* 12(8):1–23
6. Laxmi Kowta AS, Bhowmick K, Kaur JR, Jeyanthi N (2021) Analysis and overview of information gathering & tools for pentesting. In: International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, pp 1–13
7. Pahuja D, Sidana P (2021) Implementing and comparing different password cracking tools. *Int Res J Eng Technol (IRJET)* 8(5):2089–2092
8. Kavya Rani SR, Soundarya BC, Gururaj HL, Janhavi V (2021) Comprehensive analysis of various cyber attacks. In: Mysore sub section international conference (MysuruCon), Hassan, India

An Investigative Study on Security Aspects and Authentication Schemes for Internet of Vehicles



Preeti Dhankar, Bhargavi Singh, and Priya Sharma

Abstract Over the past few decades, empirical studies and specialists in intelligent transportation systems (ITS) have proclaimed a pervasive interest in the revolutionary developments of technology about the Internet of Vehicles (IoVs). IoV strives at accomplishing a compelling intuitive vehicle system, adept at rendering a multitude of competent services in addition to supporting multiple applications for road users. In IoV, autonomous vehicles can instantly connect with other nearby vehicles. In this paper, an initiative has been made to study a variety of security threats relevant to the IoV ecosystem and the ways to mitigate them. Further, an in-depth analysis of the most recent IoV authentication schemes has been illustrated concerning tools and techniques used, strengths, limitations, and objectives. Moreover, an overview highlighting the challenges and future trends in IoV has also been included. The researcher's conclusions offer a standard response for the IoV for intelligent transport's security and authentication issues.

Keywords Authentication · Attacks · Blockchain · Internet of Vehicles (IoV) · Security

1 Introduction

The IoV is regarded as a network expansion for vehicle-to-vehicle (V2V) communication. With the assistance of vehicle AI and awareness of other vehicles and their behaviors, IoV provides enhanced driving aids. VANETs have a limited range with constrained computational and storage capabilities. These VANETs are also unable

P. Dhankar · B. Singh (✉) · P. Sharma

IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India
e-mail: bhargavi130bit19@igdtuw.ac.in

P. Dhankar
e-mail: preeti114bit19@igdtuw.ac.in

P. Sharma
e-mail: priya115bit19@igdtuw.ac.in

to handle global information like drivers' behavior, jamming, and complex infrastructure. This led to the introduction of new technology, i.e., IoT. GPS, sensor devices, and radio frequency identification together with IoT transformed the conventional VANET into a smart interconnected vehicular system known as the IoV. IoV being an open network is prone to different security issues. This encouraged researchers to introduce various authentication schemes. IoV and its security and efficiency have become significant areas of focus in recent years [1–8]. Researchers analyzed and presented the merits and demerits of technologies like blockchain, cryptography, and key verification to authenticate IoV networks. It can be helpful to be aware of or avoid accident-prone areas and maintain safe and intelligent transportation; this can be achieved with the help of an IoV network. Even so, new requirements which ensure smooth, secure, resilient, and ascendable transmission of information among vehicles, IoV users, and wayside infrastructures of vehicular networks are needed as the count of automated vehicles keeps growing [3–5, 9, 10]. The evolution of IoV has been driven by a combination of technological innovation, increased demand for smarter transportation systems, and a growing need for safer, more efficient roads. As technology continues to advance, IoV will likely continue to evolve and play an increasingly important role in the future of transportation. Let's attempt to document these turning points in IoV's development:

- In 1995, the first connected car—Included rescue operations, emergency management, and safety. The history of IoV starts.
- In 1999, TeleAid Telematics—Majorly worked on emergency response, roadside assistance, and car recovery.
- In 2001, Vehicle Health Record—A device having remote diagnostic hardware which is the advanced and better version of TeleAid Telematics.
- In 2007, The Stolen Vehicle Management, M2M Telematics, and NAD followed.
- In 2009, Remote Lock by cellular phones—Featured features including the ability to lock, unlock, and locate vehicles via mobile devices.
- In 2014, 4G LTE Wi-Fi Hotspot—in vehicles, novel communication technology was introduced.
- In 2019, predictive intelligence was introduced for artificial intelligence (Fig. 1).

The applications of IoV are explained in Fig. 2. IoV technology makes driving more secure. Sensors in cooperative collision avoidance systems detect potential collisions as well as send an alert message to the driver [11]. The IoV provides periodic updates on vehicle performance and alerts for potential emergencies, triggered by traffic problems, dangerous road conditions, and accidents detected on integrated roadways. The technology also allows for remote vehicle access, enabling services like remote door locks and vehicle recovery, including “find my vehicle,” which is especially helpful for large parking lots. In addition to benefiting individual users, IoV also offers advantages to transportation agencies by improving real-time traffic, transportation, and parking data management, leading to the reduction of traffic congestion [6–8, 12–16]. The applications of IoV have been provided in Fig. 2.

Moreover, the layered architecture of IoV has the following layers [17]:

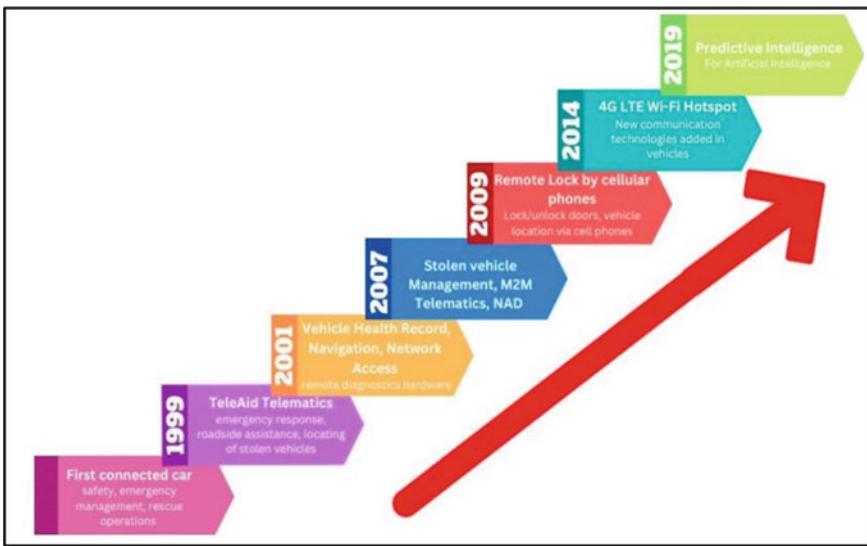


Fig. 1 Evolution of IoV

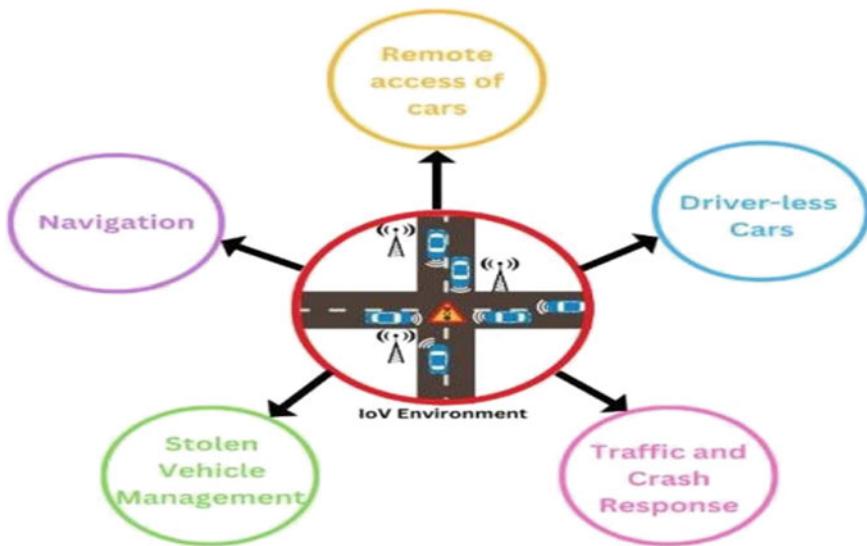


Fig. 2 Applications of IoV

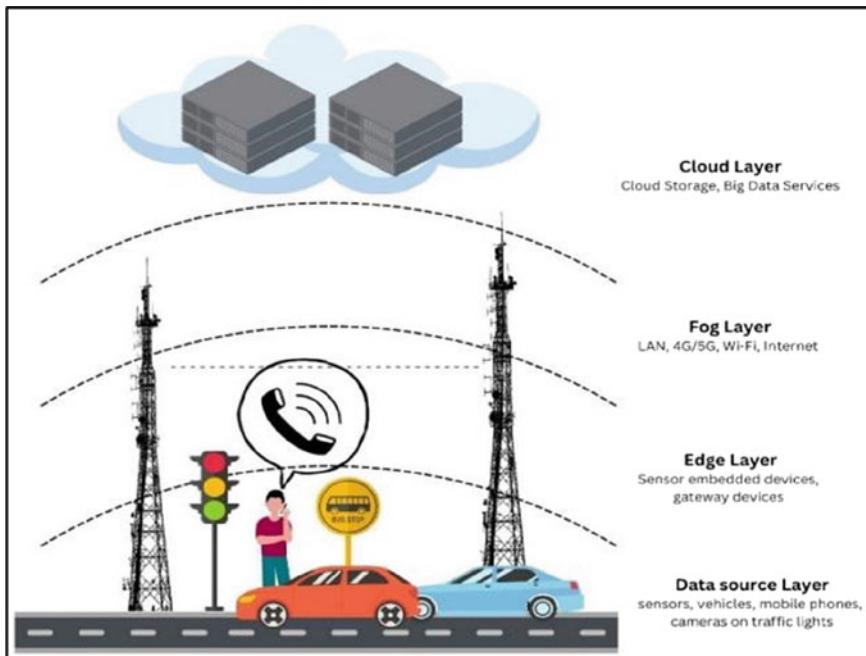


Fig. 3 Layered architecture of IoV

Data Source Layer: Modern vehicles already have around 100 sensors inside of them, and it is predicted that this number will expeditiously rise to about double. These unique sensors collect information from the immediate environment of the cars, which, combined with the data obtained through different communications, are part of this layer. Only a small amount of this data is handled at this layer because a vehicle's computing power is restricted (Fig. 3).

Edge Layer: To ensure that the information related to vehicles and traffic is processed and analyzed in real-time, the vast amount of data collected by sensors and V2X connections must be swiftly processed. An edge layer has been developed to enable further data analysis without delays caused by cloud-related issues. The same sensor-embedded smart infrastructure, gateway gadgets, or vehicles are used at this tier.

Fog Layer: Similar to the edge layer, this layer is added to lessen reliance on the cloud for data analysis. By using transactional communication infrastructure like Wi-Fi and LAN, etc, the fog layer completes additional data processing at a local level, ensuring fast decision and preventing latency problems that may have arisen by anticipating solely the cloud services for all the required processing.

Cloud Layer: Big data storage and cloud servers are included in the cloud layer to handle and store enormous amounts of information that is difficult to process at lower tiers. Cloud latency concerns are the reasons why the data needed for advanced

applications like navigation and flow monitoring of the traffic, which need data processing, is reviewed on this layer.

Data sharing increases the risk of security threats from adversaries, jeopardizing the safety, and privacy of users. An attacker can intercept, edit, remove, or even induce bogus information while communicating as the entities engaged in the IoV system communicate along an open channel. Due to such vulnerabilities that can potentially lead to the loss of crucial user data, it is fundamental to design authentication schemes that aid in preventing various attacks. IoV focuses on providing safety during driving by reinforcing communication among vehicles and infrastructure present on the roads such as the road units (RSUs) and trusted third parties (TAs). The connection between RSUs and TAs is physical, that is, through wires, whereas that between RSUs and vehicles is wireless. Vehicles transmit safety-related messages, such as vehicle speed and road conditions, to nearby vehicles and RSUs using the DSC protocol regularly. Once the content sent by the vehicle is received, the RSU or the automobile must inspect the message's reliability to make sure that it has not been modified during transmission. It is possible that the content might have been modified by an adversary during transmission and could contain malware that might compromise the network entities or the whole network; therefore, the content must be taken care of by a dependable third party to ensure its integrity. Numerous techniques have been put forward for authentication in an attempt to minimize the existing problems [11–15, 18].

2 Possible IoV Security Attacks

The security attacks possible in IoV are as follows [12, 19, 20]:

- **Routing Attacks:** The IoV's limitations, such as bandwidth, transmission range, and mobility, impact the routing algorithms' quality and reflect the connectivity among RSUs, vehicles, and other TPMs. This leads to flaws and vulnerabilities in the IoV routing method. Eavesdropping is an example of a routing attack (Fig. 4).
- **Attacks on Authentication:** An attacker can use the information gained from a network to impersonate a network entity leading to the compromise of the whole network. Therefore, all the devices on the network should be authenticated, and it should be ensured that the sensor, actuator, or vehicle that is delivering the information is legitimate. The system ought to be able to distinguish between honest and dishonest automobiles. Sybil's attack is an example of an authentication attack. In this attack, multiple nodes can join the network; consequently, the network's functioning is interfered with.
- **Availability Attacks:** Participants in IoV are steadily growing with the frequency of vehicles on the road. It is imperative to ensure that the network does not fail due to congestion or an overload of requests. The system should be accessible to the users at every time. DoS, black holes, gray holes, and spamming throws are examples of potential availability attacks. They work by creating multiple spam

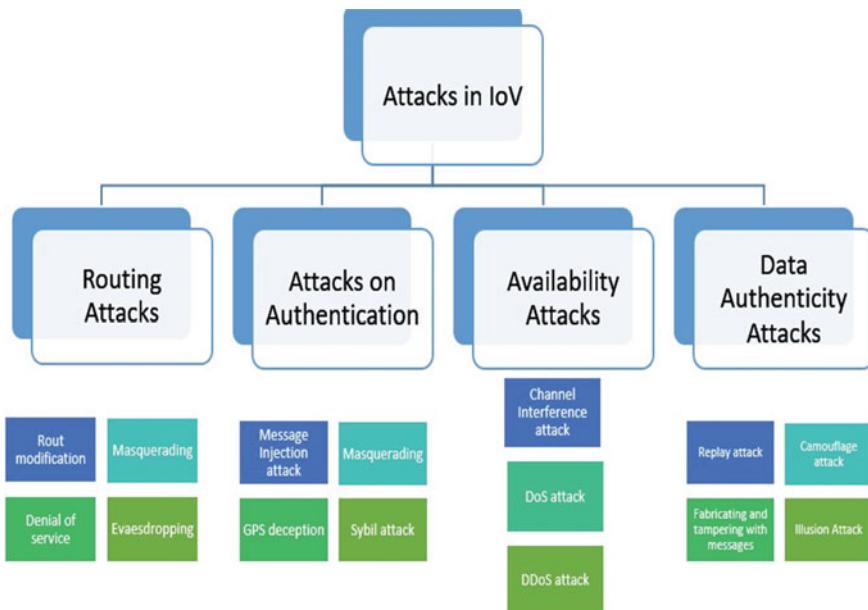


Fig. 4 Categorization of attacks in IoV [12]

requests which lead to congestion and jamming of the network by using a lot of bandwidth and increasing latency.

- **Data Authenticity Attacks:** The authenticity and integrity of the data have to be checked before transmitting the data packets throughout the network. The following categories might be used to cause data authenticity attacks. Replay attacks are a good example of data authenticity attacks. They differ from availability attacks in that they can be carried out by unauthorized nodes. Replaying messages uses valuable bandwidth quickly, which causes priority messages to be removed from the queue (Fig. 5).

3 The Ways to Mitigate Security Vulnerabilities

- **Threat Model:** For understanding and studying the effects of various attacks on IoV, modeling those attacks is essential. Threat modeling works toward the early identification, communication, and understanding of vulnerabilities to the stakeholders of the organization to facilitate prevention. System analysts and defense personnel can fully analyze potential attacker characteristics, the most likely attack paths, and the assets the attacker is most interested in by using the documentation from this procedure [12].

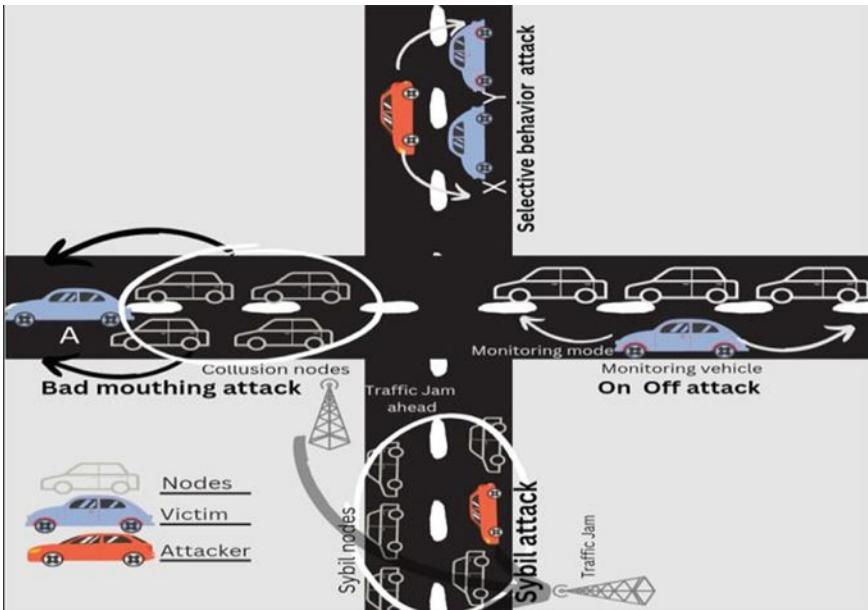


Fig. 5 IoV security attacks

- **Intrusion Detection System (IDS):** An IDS is a crucial additional network security element. By gathering and investigating data from the Internal Network Systems (INS), it looks for system behaviors that go against security policy or other signals of attack. IDSs offer defenses against internal as well as external attacks.
- **Key Management:** To efficiently use encryption, key management is necessary to ensure the authenticity and validity of the key. Key generation, distribution, transmission, preservation, destruction, and backup are all parts of key management. In conventional networks, KDC or CA typically completes the required distribution and management. For IoV, the distributed authentication protocol utilizes certificate-less signature technology to assist vehicles in secretly receiving keys and employs an effective pseudonym signature to safeguard privacy [21].
- **Honeypot:** A honeypot, according to Spitzner, is a security resource that has value only when it is probed, attacked, or hacked. To attract attackers, honeypots operate as regular system computing resources, complementing most other security measures. To shield the actual targets, the essential system services, and the data they contain from the attraction of attackers, honeypots try to redirect the attention of the attacker far from the critical system resources and analyze the attacker's behavior to produce signatures for IDS. In IoV, the communication and authorization modules are the most frequently targeted portions.

- **Secure Routing Protocols:** There are various secure routing protocols (such as SAODV, Ariadne, and SRP) that are based on conventional protocols and are effective against common routing threats. These protocols can perform standard routing tasks while ensuring the security of the network.
- **Routing Privacy Protection Mechanism:** To prevent the leakage of routine node data during the routing process, IoVs require a privacy protection mechanism. One approach is to use the concept of “The Millionaire’s Problem” to conceal the value of each utility to compare two things without disclosing their true values.

4 Classification of Authentication Schemes

Authentication is the process of verifying the identity of a user or device. There are various authentication schemes [9, 13, 15, 16, 20, 22, 23] available that use different factors to authenticate users. These factors can be broadly classified into three categories: what we know, have, and are (Fig. 6).

- **What we know:** This factor refers to something that only the user knows, such as a password or a PIN. This type of authentication is commonly used in many online services, such as email, social media, and online banking. It is important to choose a strong and complex password to ensure better security. Passwords are also often combined with other factors, such as biometric authentication, to provide better security.
- **What we have:** This factor refers to something that the user possesses, such as a physical token, a smart card, or a mobile device. Tokens can be hardware-based or software-based and are used to generate one-time passwords (OTPs) or digital signatures. Smart cards are used for secure access to physical spaces, such as offices or data centers. Mobile devices, such as smartphones or tablets, are also used as authentication factors by generating OTPs or by using biometric authentication.

Fig. 6 Classification of authentication schemes based on user ID



- **What we are:** This factor refers to something unique to the user, such as biometric information. To authenticate users, biometric authentication uses physiological or behavioral features, such as vocal, eye, face, or fingerprint recognition.
- Multi-factor authentication combines two or more factors and provides better security than single-factor authentication. It is also important to choose the right authentication scheme based on the level of security required and the usability of the authentication factors. By using a combination of what we know, what we have, and what we are, authentication schemes can provide secure and convenient access to a variety of applications and services.

5 Challenges in IoV

5.1 *High Agility*

Ensuring real-time security requires timely delivery and unaltered retention of data packets throughout their unpredictable routing from sender to recipient. However, the mobility of nodes in IoV and VANET networks and the ever-changing network architecture make it difficult to meet real-time assurance and non-repudiation security procedures. Therefore, ephemeral V2V and V2I interactions are common, posing a significant challenge to meeting security requirements [1, 11, 19].

5.2 *Low Tolerance for Errors*

In IoV, even minor delays in information delivery or packet reception can have disastrous consequences, including traffic accidents. As a result, time constraints are critical. However, limited capacity and variable network quality in IoV networks caused by the large number of vehicles, their movement, and uncertain fluctuations in wireless networks, make it challenging to achieve real-time security. Therefore, proactive security measures must be taken, striking a balance between an efficient key allocation process and low overhead [1, 11, 19].

5.3 *Key Management*

Numerous authorities and stakeholders, such as government entities and vehicle manufacturers, play critical roles in IoV. However, selecting an entirely Trusted Authority (TA) to manage key exchange, management, and revocation is challenging because it poses a security risk. The wrong choice of authority could damage the confidentiality of the IoV system. Furthermore, Certificate Revocation Lists (CRLs)

responsible for revoking misbehaving and malicious vehicles become impractical due to the size of IoV and VANET connections, increasing the certificate revocation process's overhead. Hence, a balance between an efficient key allocation process and low overhead is necessary [1, 11, 19].

6 Analysis of Recent Existing Authentication Schemes

See Table 1.

7 Conclusion

Vehicular ad hoc networks had a very small range with very constrained computing and storing ability. These VANets were also not able to handle much global information like behavior of drivers, jamming, and complex infrastructure. This became the reason to integrate it with Internet of Things. GPS, sensor devices, and radio frequency identification all together with technologies like Internet of Things(IoT), cloud computing, AI and machine learning transformed the normal VANet into a smart interconnected vehicular system, i.e., Internet of Vehicles (IoVs) and storing ability. These VANets were also not able to handle much global information like behavior of drivers, jamming, and complex infrastructure. This became the reason to integrate it with Internet of Things. GPS, sensor devices, and radio frequency identification all together with technologies like Internet of Things(IoT), cloud computing, AI, and machine learning transformed the normal VANet into a smart interconnected vehicular system, i.e., Internet of Vehicles (IoVs). The application of IoV has been implemented in various fields, including smart cities, intelligent transportation systems, and autonomous driving. IoV has also been used in emergency response systems, logistics management, and fleet management, among others. Its benefits are far-reaching, with the potential to improve road safety, reduce traffic congestion, and lower carbon emissions. The architecture of IoV has been designed to ensure the seamless integration of different components, allowing for efficient data exchange and communication between vehicles and infrastructure. IoV is an open network, and this makes it prone to different security issues. This encouraged researchers to introduce different authentication schemes. Different technologies like blockchain, cryptography, key verification, etc. were used to authenticate the IoV network.

Table 1 Analysis of recent existing authentication schemes

References	1	2	3	4	5	Limitations (-)	Remarks
[9]	✓	✗	✗	✓	<ul style="list-style-type: none"> • Biometric • XOR • SHA-256 • ECC 	<ul style="list-style-type: none"> – It is presumed that this registration procedure can be carried out safely and without interference from an attacker. However, an attacker can also use one or more identities to register with the TA. -Does not focus on untraceability and forward secrecy features 	Optimizes the steps of vehicle authentication. Deals with replay, location spoofing, and identity guessing attacks. Defends DoS attack
[10]	✗	✗	✗	✓	<ul style="list-style-type: none"> • Dynamic update technology • Double pseudonym method • Bilinear pairing 	<ul style="list-style-type: none"> – There is less emphasis on dealing with the extremophiles in which the platform is attacked by a DoS attack while relaying messages 	There is no requirement to use bilinear pairing during signature generation and verification
[18]	✗	✓	✗	✓	<ul style="list-style-type: none"> • Blockchain • Push-pull communication • Gossip protocol • BFT Algo 	<ul style="list-style-type: none"> – Difficult to use in mixed traffic – Equipment is costly – Susceptible to Sybil attacks – Works only in small consensus due to the high amount of communication between nodes 	Decentralized improved BCA-TG Scheme. The solution to encryption dilemmas inside this IoV for erudite transport blockchain-based IoV architecture

(continued)

Table 1 (continued)

References	1	2	3	4	5	Limitations (-)	Remarks
[16]	✗	✗	✓	✓	• Hyper-elliptic curve cryptography (HECC)	<ul style="list-style-type: none"> – High computation cost for signature verification – The scheme does not provide security against DoS attack and does not provide discussion on forward secrecy 	Provides an authentication management plan for the IoV environment that is both computationally and communication efficient by using HECC. Signature verification and Message Signing are the highlights
[21]	✗	✓	✓	✗	<ul style="list-style-type: none"> • Hash code • Fuzzy inference • Smart card 	<ul style="list-style-type: none"> – Does not provide discussion on attacks such as MitM, DoS, and Identity guessing 	Dual key management approach has been used to support communication between groups. Resilient to several security attacks and smallest key computation time among the compared schemes
[23]	✗	✓	✓	✓	<ul style="list-style-type: none"> • Blockchain • BFT algorithm • Ripple consensus algorithm • PKI 	<ul style="list-style-type: none"> – The scheme does not provide any discussion on resilience towards major security attacks – No comparison with competitive scheme has been provided 	Resolves issue of Identity counterfeiting. Identity leakage is prevented through blockchain feature. Novel new node joining and identity authentication developed

Note References to the numbers in Table: 1. Blockchain-Based Authentication; 2. The transition from VANet to IoV is discussed or not; 3. Security attacks are discussed or not; 4. Challenges are discussed or not; 5. Techniques used

References

1. Bagga PD (2020) Authentication protocols in internet of vehicles: taxonomy, analysis, and challenges. *IEEE Access* 8:54314–54344
2. Narwal B, Mohapatra AK (2021) A survey on security and authentication in wireless body area networks. *J Syst Architect* 113:101883
3. Narwal B, Mohapatra AK, Usmani KA (2019) Towards a taxonomy of cyber threats against target applications. *J Stat Manag Syst* 22(2):301–325

4. Narwal B, Mohapatra AK (2021) SAMAKA: secure and anonymous mutual authentication and key agreement scheme for wireless body area networks. *Arab J Sci Eng* 46(9):9197–9219
5. Sharma M, Narwal B, Anand R, Mohapatra AK, Yadav R (2023) PSECAS: a physical unclonable function based secure authentication scheme for Internet of Drones. *Comput Electr Eng* 108:108662
6. Malik M, Gandhi K, Narwal B (2022) AMAKA: anonymous mutually authenticated key agreement scheme for wireless sensor networks. *Int J Inf Secur Privacy (IJISP)* 16(1):1–31
7. Narwal B, Gandhi K, Anand R, Ghalyan R (2022) PUASIoT: password-based user authentication scheme for IoT services. In: Proceedings of the 6th international conference on advance computing and intelligent engineering: ICACIE 2021. Springer Nature Singapore, Singapore, pp 141–149
8. Narwal B, Bansal V, Dahiya V, Aggarwal P (2021) SLUASCIoT: a secure and lightweight user authentication scheme for cloud-IoT services. In: 2021 5th international conference on Information Systems and Computer Networks (ISCON), Mathura, India, pp 1–5. <https://doi.org/10.1109/ISCON52037.2021.9702456>
9. Chen CM (2019) A secure authentication protocol for internet of vehicles. *IEEE Access* 7:12047–12057
10. Cui JX (2018) Privacy-preserving authentication using a double pseudonym for internet of vehicles. *Sensors* 18(5):1453
11. Ji BZ (2020) Survey on the internet of vehicles: network architectures and applications. *IEEE Commun Standards Mag* 4(1):34–41
12. Ji ST (2014) Wormhole attack detection algorithms in wireless network coding systems. *IEEE Trans Mobile Comput* 14(3):660–674
13. Liu YW (2017) Efficient privacy-preserving dual authentication and key agreement scheme for secure V2V communications in an IoV paradigm. *IEEE Trans Intell Transp Syst* 18(10):2740–2749
14. Mahmood AZ (2019) Trust management for software-defined heterogeneous vehicular ad hoc networks. In: Security, privacy and trust in the IoT environment. Springer, Cham, pp 203–226
15. Pu YX (2020) In efficient blockchain-based privacy preserving scheme for vehicular social networks. *Inf Sci* 540:308–324
16. Shah TA (2022) Cost-efficient privacy-preserving authentication and key management scheme for internet of vehicle ecosystem. In: Complexity
17. Siddiqui SA (2021) A survey of trust management in the internet of vehicles. *Electronics* 10(18):2223
18. Hu WH (2019) A blockchain-based Byzantine consensus algorithm for information authentication of the Internet of vehicles. *IEEE Access* 7:139703–139711
19. Sun YW (2017) Attacks and countermeasures in the internet of vehicles. *Ann Telecommun* 72(5):283–295
20. Sutrala AK (2020) On the design of conditional privacy preserving batch verification-based authentication scheme for Internet of vehicles deployment. *IEEE Trans Veh Technol* 69(5):5535–5548
21. Vijayakumar PA (2015) Dual authentication and key management techniques for secure data transmission in vehicular ad hoc networks. *IEEE Trans Intell Transp Syst* 17(4):1015–1028
22. Thumber GR (2020) Efficient and secure certificateless aggregate signature-based authentication scheme for vehicular ad hoc networks. *IEEE Internet Things J* 8(3):1908–1920
23. Wang XZ (2019) An improved authentication scheme for internet of vehicles based on blockchain technology. *IEEE Access* 7:45061–45072

Amharic Language Hate Speech Detection Using Machine Learning



Abirham Ayenew and Uttam Chauhan

Abstract The extensive availability of social media platforms, as well as the adaptability of the Internet, has made it easier for users to participate in violent communication. The anonymity provided by online platforms makes them appealing to individuals engaging in hate speech to conceal their criminal activities. This poses a significant challenge in many countries, especially Ethiopia. As social media platforms continue to multiply and the volume of social media data grows exponentially, the identification of hate speech presents a formidable challenge. This challenge exacerbates conflicts between diverse ethnic groups and contributes to the dissemination of misinformation within communities. Despite the fact that there is a lot of research on hate speech identification, much of it is for high-resource languages, and there is still a lot of work to be done for low-resource languages. These pique our interest. As a response to these issues, this study uses machine learning methods to create an Amharic hate speech detection model. In this research, we prepared a new Amharic hate speech dataset from Facebook, labeled manually as hate and free based on standard guidelines and pre-processed, and labeled into two classes, and the data is augmented to balance the class category. We applied word2Vec embedding and TF-IDF feature selection techniques to train Random Forest, and Naïve Bayes machine learning models. To evaluate the models, we adopted an (80, 10, 10) train, validate, and test split. We utilized precision, recall, and F1-score as a means to compare the performance of these models. By combining the Naïve Bayes algorithm with Word2Vec and TF-IDF techniques, the best performance was achieved, resulting in an accuracy of 91.59%. The model achieves a promising result with a unique feature selection and appropriate pre-processing techniques.

Keywords Amharic hate speech detection · Machine learning · Amharic posts and comments

A. Ayenew (✉) · U. Chauhan
Computer Engineering, Vishwakarma Government Engineering College,
Ahmedabad, Gujarat, India
e-mail: abirhamagetie02@gmail.com

U. Chauhan
e-mail: ugchauhan@vgecg.ac.in

1 Introduction

In recent times, social networking has experienced significant growth and widespread adoption among individuals. The proliferation of social media platforms has facilitated worldwide contact, information exchange, and economic activity [1]. According to StatCounter global status data from April 2022 to February 2023, there are 4.55 billion active social media users and 6.4 million social media users in Ethiopia; in Ethiopia, Facebook is the predominant social media platform and is extensively utilized by a majority of community members, accounting for 67.23% of social media usage [2]. While it is possible to write Amharic using the Latin alphabet, Ethiopian Facebook users often prefer to use Geez scripts to discuss important topics and engage in communication through comments and posts [3]. Amharic, with approximately 65 million speakers, ranks as the second most-spoken Semitic language in Sub-Saharan Africa. It follows Arabic, which boasts over 300 million speakers [4].

Amharic serves as the official working language of Ethiopia, as well as in certain regional states like Amhara and Southern Nations, Nationalities, and People (SNNP) [5].

Amharic is written from left to right and utilizes its own distinct script, which does not incorporate capitalization. The language consists of a total of 275 characters that can be arranged in seven different forms, while the concept of capitalization is absent [4, 6]. While social media offers numerous benefits, it is not devoid of problems, largely stemming from the lack of ethical considerations. Unfortunately, some individuals misuse social media platforms for inappropriate purposes, diverting from their intended positive usage. Aside from the authenticity of the content, any Facebook user is aware that information can travel faster between individuals. Users share what they have got without checking the accuracy of the content. Indeed, one of the downsides of modern democratic societies worldwide is the right to freedom of speech, which can lead to the sharing of various forms of hateful content on social media platforms [7]. While freedom of speech is a fundamental right, its unrestricted exercise can result in the dissemination of harmful or offensive material online. The governments of Ethiopia and other researchers have been doing to address this. They have worked to discover answers and lessen the detrimental influence of social media. For instance, the Ethiopian government has implemented the hate speech and disinformation prevention and suppression proclamation No. 1185/2020 [8].

The main focus of this paper is to address Amharic hate speech detection specifically on Facebook. The authors conducted experiments using various ML algorithms and achieved a promising result of 91.59% accuracy with Multinomial Naïve Bayes.

The remaining material in Sect. 2 gives an overview of relevant research on this topic. Section 3 outlines the approach employed, while Sect. 4 offers the architectural design and goes into depth about the experimental procedure. In conclusion, Sect. 5 of the paper summarizes the findings and key points discussed and offers recommendations for future research works.

2 Review of Related Works

2.1 Amharic Language

Amharic is the official working language of the Federal Democratic Republic of Ethiopia [9, 10]. It is written from left to right in a distinctive geometric script. It has a writing system known as Fidel, which contains 34 base characters and another six characters expressed for each base character constructed from combinations of base character consonants and vowels. Few computational linguistic resources have been developed for the Amharic language even though having a large number of speakers [10, 11].

2.2 Amharic Writing

Amharic writing follows a system where letters are constructed through a combination of consonant-vowel pairs, similar to the Geez script. The language encompasses more than 231 orthographic representations of symbolic language. This includes 33 fundamental symbols, each with seven variations or orders. Out of these seven orders, six represent consonant-vowel combinations, while the seventh represents the consonant itself [12].

Table 1 provides an illustrative example of the mapping of the Amharic alphabet.

2.3 Definition of Hate Speech

The UN Strategy and Plan of Action against Hate Speech was developed by the UN in order to provide a comprehensive strategy at the international level. The definition of hate speech in this framework is “any type of communication, verbal, written, or behavioral, that uses disparaging or discriminatory terminology to target people or groups based on their religion, ethnicity, nationality, race, color, descent, gender, or other identification criteria” [13].

Table 1 Sample mapping of letters of English and Amharic

	ä/e	U	I	A	E	e
H	ሀ	ሁ	ሂ	ሂ	ሂ	ሁ
L	ለ	ሉ	ሉ	ለ	ሉ	ለ
M	ሙ	ሙ	ሙ	ሙ	ሙ	ሙ
S	ሠ	ሠ	ሠ	ሠ	ሠ	ሠ
B	በ	በ	በ	በ	በ	በ

2.4 Existing Techniques of Hate Speech Detection

Many researchers have explored this topic, and it is getting traction in recent years. In this particular section, the authors aim to provide an overview of the features employed in previous studies.

- Dictionaries and Lexicons: This method involves creating a list of words that are searched and counted within the text. Previous studies on hate speech detection have focused on utilizing various content words, including insults, and curse words [14]. Additionally, some studies have focused on identifying the presence of disrespectful words by employing dictionaries that encompass English words, including acronyms and abbreviations [15]. Label-specific features have also been employed [16].
- Bag-of-Words (BOW): It is another approach used in hate speech detection [17–19]. Instead of depending on a fixed collection of terms like dictionaries, it creates a corpus based on the words available in the data. One disadvantage of the bag-of-words concept is that it may result in misclassification if terms are employed in various contexts. N-grams are used to address this issue. N-gram algorithms are commonly employed in NLP and related tasks [20–23]. In a specific study, it was observed that character n-gram features showed higher predictive power compared to token n-gram features when detecting abusive language [24]. This suggests that analyzing sequences of characters can provide valuable insights into identifying and detecting abusive language in the text.
- TF-IDF: It has also been applied in hate speech classification. It assesses the significance of a word in a document in relation to a corpus [25].
- Word Embedding: Word embedding is one deep learning technique that enables the exploration of both semantic and syntactic relations among words [26]. This enables the recording of subtle characteristics and contextual clues inherent in human discourse. Word2Vec [27] is a popular word embedding technique that uses unsupervised learning to find connotational word relationships. It employs a layered neural network to build a vocabulary based on frequently occurring words while filtering out noise. According to [27], word embeddings with 50–300 dimensions may accurately model millions of words. Combining word embedding with Convolutional Neural Networks (CNN) has shown improved performance in various studies [26, 28]. Some researchers have also explored the use of paragraph2vec and comment embedding for tasks such as abusive language detection and central word prediction [16, 29].
- FastText, another word embedding technique, has been applied in problems where sentence classification is required rather than word-level analysis [26].
- Sentiment Analysis: To detect negative polarity in hate speech, it has been used as a feature [23, 30–33].

2.5 Related Works

Research [4] focused on developing Apache Spark (AS) models for classification. Posts and comments on Facebook have the potential to be categorized based on their content and tone, encompassing a wide range of expressions and opinions. The authors used and leveraged RF and NB as learning algorithms using Word2Vec and TF-IDF for feature selection. Combined with the NB classifier Word2Vec's functional model showed superior performance compared to the Facebook model. A social network algorithm analyzes 1 posts and comments in Amharic, that accuracy 79.83%, ROC score 83.05%, area below target. The precision-recall curve is 85.34%. The NB method showed superior results when employing the TF-IDF feature model, with an accuracy of 73.02%, a ROC score of 80.53%, and an area under the curve of 79.93%. In contrast, RF when combined with the Word2Vec feature exceeded TF-IDF with all employed performance metrics. The objective of the research [3] was to create an Amharic hate speech detection model. This research involved collecting the data, pre-processing, and extracting features. Model training can be done with SVM and other algorithms and model performance was evaluated through testing, revealed in experiments Create 21 binary and ternary models for each dataset with two records. RF with word2vec outperformed both SVM and NB. In terms of performance, his SVM model with Word2vec showed superior results compared to NB and RF models, achieving a score of 73%. Additionally, a ternary SVM model with word2vec scored 53% points. The study showed that SVM models utilizing word2vec showed slightly better performance. In different research conducted by researchers [34], they developed a model specifically for identifying vulnerable communities. Researchers compared deep learning (DL) approaches such as: As RNN-LSTM and RNN-GRU with conventional gradient boosting trees (GBT), RF method. I used Word2Vec embedding in combination with RNN-GRU. It achieved a staggering AUC of 97.85% and an accuracy of 92.56%.

The study [35] focuses on the issue of Amharic hate speech detection on social media. A multimodal approach uses deep learning to combine acoustic and textual elements. The following four deep learning algorithms are used in this study: LSTM, BiLSTM, GRU, and BiGRU. The results of the multimodal experiments are the BiLSTM-based. The multimodal model outperforms other models in detecting Amharic hate speech. We achieved an accuracy of 88.15%.

Reference [36] develops a model that classifies videos into these two categories. Free and hateful as a means of characterizing British hatred, based on the audio content of the video speech. Studies have shown that Random Forest classifiers significantly improve performance. Improved and achieved a success rate of 96%.

Furthermore, in a separate study [37], a technique was proposed to identify videos that contain hateful content. The researchers augmented their dataset by combining 204 randomly selected videos with 96 offensive videos obtained through a targeted keyword searches. They experimented with different machine learning algorithms and determined that Support Vector Machines (SVM) achieved the best results in terms of both efficiency and performance.

3 Methodology

3.1 Dataset Collection

They obtained posts and comments from Facebook using a web crawler that utilized the Graph API. The data they collected consisted of two types: regular conversations and offensive content that targeted specific ethnic, religious, political, and socio-economic groups. They gathered a total of 8000 posts and comments from pages with over 500k followers.

3.2 Data Labeling

The process of labeling the data involved three undergraduate students and one Amharic language expert who reviewed the work of the students. The annotators were provided with labeling guidelines based on the procedures used in a previous study referenced as [4]. They were instructed to categorize the data as either “hate” or “free”, and if it was determined as hate speech, they were asked to identify the specific category of hate it fell under. The annotators labeled the data according to the instructions, and the resulting labels are presented in Table 2.

The distribution of the labeled dataset is presented in Fig. 1.

3.3 Data Pre-processing

The authors employed five pre-processing steps to clean and normalize the data.

Removal of non-Amharic alphabets: Non-Amharic posts and comments were manually identified and removed, although it is challenging to eliminate all non-Amharic alphabets manually. To address this, an algorithm was developed to replace non-Amharic alphabets with spaces, which were later removed.

Table 2 Sample classification

Label	Post
Hate	ይህ ደንብ ቅዱቷልኝ ለሰላልቻትን ማጥከስለቸው እና ነው
Free	ለመለው የአማራር ሆነው የቀረበው ከባዊ የተጠሪ ጥሩ
Free	አማራዊ ማንነታቸው እንደወጪቸል ተቀጥሮ
Hate	የቀን፡ ፕሮ፡ ወያዝ
Free	የጊዜ ድም እና የእባፍች እንዲ በከንቱ ላይ አድዋርም

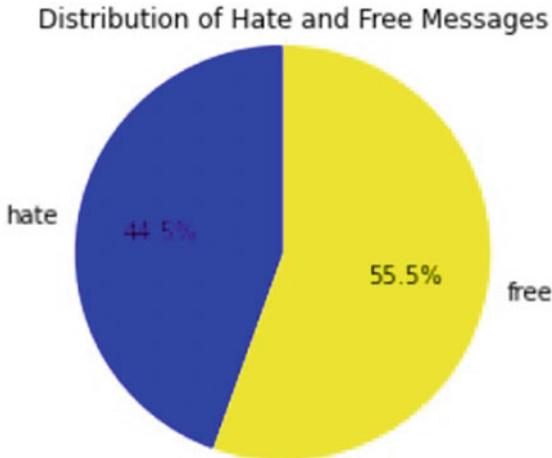


Fig. 1 Class distribution of free and hate

Table 3 Amharic character normalization

Sound characters	Normalized to
[አ] [አ.] [አ.ሸ] [አሸ] [አሸ.] [አሸ.]	[ሀ] [ሀ.] [ሀሸ] [ሀሸ.] [ሀሸ.] [ሀሸ.]
[ወ] [ወ.] [ወሸ] [ወሸ.] [ወሸ.] [ወሸ.]	[ወ] [ወ.] [ወሸ] [ወሸ.] [ወሸ.] [ወሸ.]
[ዕ] [ዕ.] [ዕሸ] [ዕሸ.] [ዕሸ.] [ዕሸ.]	[ዕ] [ዕ.] [ዕሸ] [ዕሸ.] [ዕሸ.] [ዕሸ.]
[ጋ] [ጋ.] [ጋሸ] [ጋሸ.] [ጋሸ.] [ጋሸ.]	[ጋ] [ጋ.] [ጋሸ] [ጋሸ.] [ጋሸ.] [ጋሸ.]

- Removal of punctuations and symbols: The dataset contained different punctuation marks used in comments and posts. These punctuations were replaced with single-spaced white space. Examples of punctuation marks replaced include “!#\$%&’’ <=>[]_‘’ ‘’ i”.
- Removal of stop words: The authors utilized a word cloud to identify frequently used words [38]. These high-frequency words were considered stop words and were stored in a dictionary. The stop words present in the dataset were then removed using a stop word corpus.
- Normalization: The Amharic alphabet consists of letters with different shapes but the same sound. The authors applied character mapping [39] to normalize these letters. Table 3 provides an illustration of the character mapping used.

3.4 Feature Selection

In order to develop an effective hate speech detection model using machine learning word embedding, the authors utilized TF-IDF feature extraction as one of the

methodologies, as previously employed in [4]. By combining word2Vec and TF-IDF techniques, they achieved improved performance in hate speech detection. The utilization of Word2Vec has been prevalent in related research studies [11, 12, 15, 16] and text classification [40]. Similarly, TF-IDF has been employed by various authors for feature selection in text classification using different tools [20].

The primary focus of this work is to classify posts and comments as either hate speech or non-hateful using machine learning algorithms. The authors made significant contributions by creating a new dataset and enhancing the performance of previous related works. They accomplished this by implementing detailed pre-processing techniques and leveraging the Apache Spark framework, resulting in an improvement of 91.59% compared to the work presented in [4] using a different dataset.

4 System Design and Experimentation

4.1 System Design

Model creation is a formal approach to designing architecture [38]. The architecture of the system, depicted in Fig. 2, illustrates the process of data collection, annotation, model building, and result generation following the implementation of various experiments using different machine learning algorithms.

Our model design encompasses essential components and steps in the detection process. These include dataset representation, text pre-processing, feature extraction, and the application of ML algorithms for classification. The architecture illustrates the complete workflow, starting from the data collection to pre-processing, feature selection, model training, and text classification.

After the pre-processing stage, feature selection was performed using word2Vec and TF-IDF feature extraction methods. The model was then trained using machine learning algorithms. The researcher will evaluate the models using four key metrics: accuracy, recall, F-measure, and precision, as mentioned in [13, 23]. To assess these metrics, the researcher will run individual models. For a better understanding, the researcher will define four terms based on the collected data: True Hate, False Free, True Free, and False Hate. These terms will be processed using both the actual data and the predicted data from the machine learning technique. The definitions are as follows:

- True Hate: Comments and posts that are classified as hate and correctly predicted as hate by the machine learning technique.
- False Hate: Comments and posts that are classified as hate but incorrectly predicted as not hate by the machine learning technique.
- True Free: Comments and posts that are classified as not hate and correctly predicted as not hate by the machine learning technique.
- False Free: Comments and posts that are classified as hate but incorrectly predicted as not hate by the machine learning technique (Table 4).

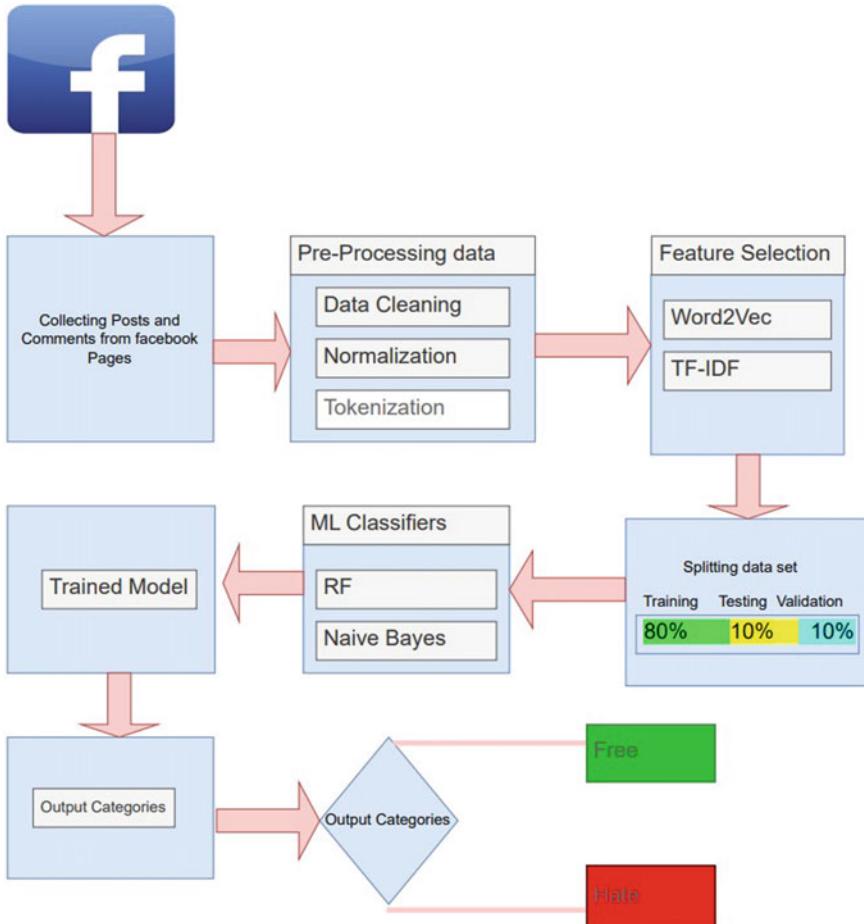


Fig. 2 Architecture of hate speech detection system

As presented in True Hate speech as TH, False Hate speech as FH, True Free speech as TF, and False Free speech as FF.

Accuracy measures the fraction of correctness of the model's predictions for both hate speech and free speech. It is calculated as follows:

$$\text{Accuracy} = \frac{\text{TH} + \text{TF}}{\text{TH} + \text{TF} + \text{FH} + \text{FF}}, \quad (1)$$

where TH, TF, FH, and FF denote True Hate, True Free, False Hate, and False Free, respectively.

Precision measures the fraction of actual positives among those predicted as positive [4]. It is calculated as follows:

Table 4 Data values

	Actual hate	Actual free
Predicted hate speech	True Hate (TH)	False Hate (FH)
Predicted free	False Free (FF)	True Free (TF)

Table 5 Classifier performance with different feature selection methods

Classifier	Feature	Evaluation			
		F-score	Precision	Recall	Accuracy
RF	Word2Vec	0.67	0.79	0.58	0.74
	TF-IDF	0.68	0.68	0.68	0.72
NB	Word2Vec	0.97	0.96	0.96	0.91
	TF-IDF	0.75	0.74	0.76	0.78

$$\text{Precision} = \frac{\text{TH}}{\text{TH} + \text{FH}}, \quad (2)$$

where TH and FH denote True Hate and False Hate, respectively.

“Recall measures how many actual positives were predicted as positives. It indicates the proportion of correctly identified instances of hate speech out of all actual instances of hate speech” [40]. The recall is calculated as follows:

$$\text{Recall} = \frac{\text{TH}}{\text{TH} + \text{FF}}, \quad (3)$$

where TH and FF denote True Hate and False Free, respectively.

The F1 measure is the harmonic mean of precision and recall. It is calculated as follows:

$$F\text{-measure} = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}, \quad (4)$$

where recall and precision denote the recall and precision values, respectively.

4.2 Results and Discussions

According to the results presented in Table 5, the Naïve Bayes algorithm with word2Vec demonstrates promising performance in hate speech detection. It achieves an impressive accuracy rate of 91.59%. Additionally, it attains high scores in F-measure (97%), recall (96%), and precision, indicating its effectiveness in identifying hate speech instances.

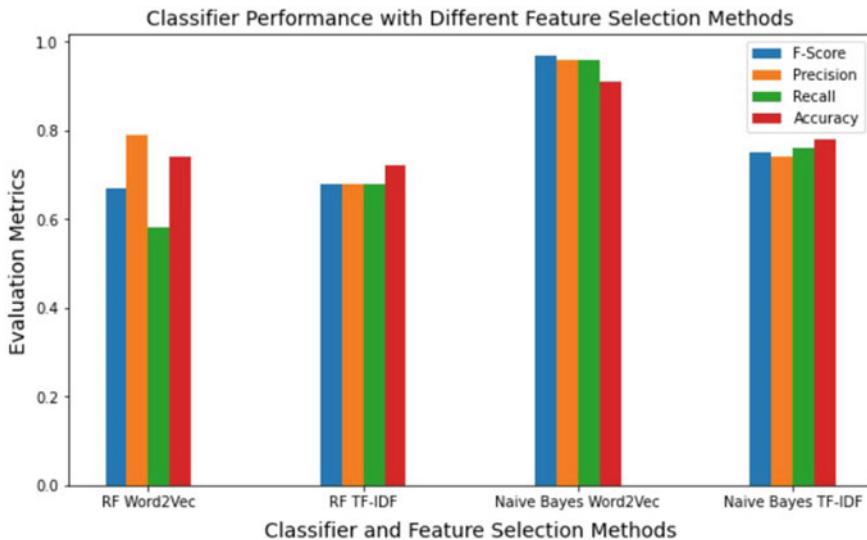


Fig. 3 Performance of classifier algorithms

Comparatively, the Naïve Bayes algorithm with TF-IDF also showcases superior performance when compared to Random Forest (RF) with both word2Vec and TF-IDF.

In terms of specific results, the Random Forest approach with word2Vec outperforms TF-IDF by achieving 0.74, 0.79, 0.67, and 0.58 for accuracy, precision, F1 measure, and recall, respectively.

Similarly, TF-IDF yields good results with an accuracy score of 0.72, precision of 0.68, F-score of 0.74, and recall of 0.68. These findings are supported by Fig. 3, which visualizes the performance metrics for both word2Vec and TF-IDF approaches.

The provided figure presents a performance comparison between the Random Forest and Naive Bayes algorithms, considering four performance metrics. The height of each bar in the figure represents the value of the corresponding performance measure, while the width represents the algorithms with their respective feature selection techniques.

Based on the figure, it is evident that the Naive Bayes algorithm with word2Vec outperforms the Random Forest algorithm across all four performance measures. It achieves higher scores in terms of F-score, precision, recall, and accuracy, indicating its superior performance in hate speech detection.

Furthermore, comparing these results with a related study [4], where Naive Bayes with word2Vec was utilized, the authors achieved 0.7983, 0.8305, and 0.8534 of accuracy, ROC score, and area under the curve, respectively. It is evident that this work surpasses previous research efforts related work, even though it may not be entirely appropriate to compare different experimental setups and datasets directly. The utilization of a relatively larger dataset and detailed pre-processing techniques in this

study played a crucial role in enhancing performance. Specifically, the construction of an Amharic stop word corpus and the elimination of these stop words from the dataset contributed to the observed improvement. Additionally, the choice of the Naïve Bayes algorithm, coupled with word2Vec, provided a rich representation of the Amharic language, capturing its semantic relationships and further improving discriminatory power. The results obtained in this study, along with the findings from the related work, emphasize the importance of factors such as dataset size, detailed pre-processing techniques, and appropriate model selection in improving the accuracy of Amharic hate speech detection. Further exploration and experimentation with different approaches, models, and evaluation metrics will contribute to advancing this research field.

5 Conclusion

The study focused on developing an Amharic hate speech detection model machine learning. The authors collected a dataset of Amharic posts and comments from Facebook by crawling various Facebook pages, including individual and organizational pages. The collected dataset underwent pre-processing, and annotators manually labeled the data.

Due to limitations with Facebook crawler APIs, the authors faced challenges in collecting the dataset. They managed to collect 8000 posts and comments, which were then divided into training, validation, and testing sets. The authors employed word2Vec and TF-IDF feature extraction techniques. They found that word2Vec's ability to capture information and handle out-of-vocabulary words contributed to its effectiveness compared to TF-IDF.

The authors evaluated the performance of the word2Vec model by assessing word similarity. They observed that their model accurately identified the semantic similarity of words, indicating the model's ability to effectively capture and represent the meaning of words in the Amharic language.

The authors developed a model using Naïve Bayes and Random Forest machine learning algorithms to classify text as hate or free speech. The model was trained using 8000 datasets, with 6000 samples used for training and 2000 samples used for testing and validation. The Naïve Bayes algorithm achieved an accuracy of 91.59% when using word2Vec for feature extraction, and 78% when using TF-IDF.

Figure 3, a bar graph, displays the performance metrics for each algorithm and feature extraction technique. It demonstrates that Naïve Bayes with word2Vec outperforms Random Forest, achieving an accuracy of 91.59%. However, the authors acknowledge that the model's performance is limited by the relatively small size of the training dataset.

Future research should address this limitation and explore further improvements. Some potential areas for future work include:

- Expanding the dataset sources: Including datasets from other social media platforms can provide a more comprehensive understanding of hate speech across different platforms.
- Multi-class classification: While the current work focused on the binary classification of hate and free speech, future research can extend the classification to include additional classes.

References

1. Tesfaye SG, Kakeba K (2020) Automated Amharic hate speech posts and comments detection model using recurrent neural network
2. StatCounter Global Stats (2023) Social media stats worldwide—statcounter global stats. [Online]. Accessed 1 Apr 2023. <https://gs.statcounter.com/social-media-stats/all/ethiopia>
3. Hudson G (1999) Linguistic analysis of the 1994 Ethiopian census. *Northeast Afr Stud* 6(3):89–107
4. Mossie Z, Wang J-H (2018) Social network hate speech detection for Amharic language. *Comput Sci Inf Technol* 41:55
5. Lewis MP, Simons GF, Fennig CD (2015) Ethnologue: languages of Ecuador. SIL International, Dallas
6. Gambäck B, Olsson F, Argaw AA, Asker L (2009) Methods for Amharic part-of-speech tagging. In: First workshop on language technologies for African languages, Athens, Greece, Mar 2009
7. Watanabe H, Bouazizi M, Ohtsuki T (2018) Hate speech on twitter: a pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. *IEEE Access* 6:13825–13835
8. Gazette F (2020) Federal Negarit Gazette of the Federal Democratic Republic of Ethiopia. Content
9. Eberhard DM, Simons GF, Fennig CD (2020) Ethnologue: languages of the world, 23rd edn. SIL International. <https://www.ethnologue.com/guides/most-spoken-languages>
10. Kelemework W (2016) Automatic Amharic text news classification: a neural networks approach
11. Gambäck B, Sikdar UK (2017) Named entity recognition for Amharic using deep learning. In: 2017 IST-Africa week conference (IST-Africa), pp 1–8. <https://doi.org/10.23919/ISTAFRICA.2017.8102402>
12. Tesfaye B, Asnake M, Weldemariam K, Alemneh DG (2019) Parallel corpora for bi-lingual English-Ethiopian languages statistical machine translation. *Int J Adv Comput Sci Appl* 10(4):464–469
13. Fissha M (2020) Design and implementation of deep learning based Amharic speech recognition system. Master's thesis, Bahir Dar University, Bahir Dar, Ethiopia. <https://ir.bdu.edu.et/bitstream/handle/123456789/14487/Melat%20Fissha%20fInal.pdf?sequence=1&isAllowed=y>
14. United Nations (2023) Definition of hate speech. United Nations. [Online]. Accessed 15 Mar 2023. <https://www.un.org/en/hate-speech/understanding-hate-speech/what-is-hate-speech-for-this>
15. Liu S, Forss T (2015) New classification models for detecting hate and violence web content. In: 2015 7th international joint conference on knowledge discovery, knowledge engineering and knowledge management (IC3K), vol 1. IEEE, pp 487–495
16. Gitari ND, Zuping Z, Damien H, Long J (2015) A lexicon-based approach for hate speech detection. *Int J Multimed Ubiquitous Eng* 10(4):215–230
17. Gagliardone I, Patel A, Pohjonen M (2014) Mapping and analysing hate speech online: opportunities and challenges for Ethiopia

18. Kwok I, Wang Y (2013) Locate the hate: detecting tweets against blacks. In: Proceedings of the AAAI conference on artificial intelligence, vol 27, pp 1621–1622
19. Melat FA (2022) Hate speech detection for Amharic language on Facebook using deep learning. Ph.D. thesis
20. Silva L, Mondal M, Correa D, Benevenuto F, Weber I (2016) Analyzing the targets of hate in online social media. In: Proceedings of the international AAAI conference on web and social media, vol 10, pp 687–690
21. Waseem Z, Hovy D (2016) Hateful symbols or hateful people? Predictive features for hate speech detection on twitter. In: Proceedings of the NAACL student research workshop, pp 88–93
22. Nobata C, Tetreault J, Thomas A, Mehdad Y, Chang Y (2016) Abusive language detection in online user content. In: Proceedings of the 25th international conference on world wide web. WWW'16. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, pp 145–153. <https://doi.org/10.1145/2872427.2883062>
23. Davidson T, Warmsley D, Macy M, Weber I (2017) Automated hate speech detection and the problem of offensive language. In: Proceedings of the international AAAI conference on web and social media, vol 11, pp 512–515
24. Mehdad Y, Tetreault J (2016) Do characters abuse more than words? In: Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue, pp 299–303
25. Cortis K, Handschuh S (2015) Analysis of cyberbullying tweets in trending world events. In: Proceedings of the 15th international conference on knowledge technologies and data-driven business, pp 1–8
26. Agarwal S, Sureka A (2017) Characterizing linguistic attributes for automatic classification of intent based racist/radicalized posts on tumblr micro-blogging website. arXiv preprint [arXiv:1701.04931](https://arxiv.org/abs/1701.04931)
27. Badjatiya P, Gupta S, Gupta M, Varma V (2017) Deep learning for hate speech detection in tweets. In: Proceedings of the 26th international conference on world wide web companion, pp 759–760
28. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems, vol 26
29. Ross B, Rist M, Carbonell G, Cabrera B, Kurowsky N, Wojatzki M (2017) Measuring the reliability of hate speech annotations: the case of the European refugee crisis. arXiv preprint [arXiv:1701.08118](https://arxiv.org/abs/1701.08118)
30. Gitari ND, Zuping Z, Damien H, Long J (2015) A lexicon-based approach for hate speech detection. Int J Multimed Ubiquitous Eng 10(4):215–230
31. Liu S, Forss T (2014) Combining n-gram based similarity analysis with sentiment analysis in web content classification. In: Special session on text mining, vol 2. SCITEPRESS, pp 530–537
32. Liu S, Forss T (2015) New classification models for detecting hate and violence web content. In: 2015 7th international joint conference on knowledge discovery, knowledge engineering and knowledge management (IC3K), vol 1. IEEE, pp 487–495
33. Schmidt A, Wiegand M (2017) A survey on hate speech detection using natural language processing. In: Proceedings of the fifth international workshop on natural language processing for social media, pp 1–10
34. Maloba WJ (2014) Use of regular expressions for multi-lingual detection of hate speech in Kenya. Ph.D. thesis, iLabAfrica
35. Mossie Z, Wang J-H (2020) Vulnerable community identification using hate speech detection on social media. Inf Process Manag 57(3):102087
36. Debele AG, Woldeyohannis MM (2022) Multimodal Amharic hate speech detection using deep learning. In: 2022 international conference on information and communication technology for development for Africa (ICT4DA). IEEE, pp 102–107
37. Wu CS, Bhandary U (2020) Detection of hate speech in videos using machine learning. In: 2020 international conference on computational science and computational intelligence (CSCI). IEEE, pp 585–590

38. Alcântara C, Moreira V, Feijo D (2020) Offensive video detection: dataset and baseline results. In: Proceedings of the 12th language resources and evaluation conference, pp 4309–4319
39. Pereira-Kohatsu JC, Sánchez LQ, Liberatore F, Camacho-Collados M (2019) Detecting and monitoring hate speech in twitter. Sensors (Basel, Switzerland) 19
40. Teshome Y (2019) Sentence-level opinion mining for Amharic language. Master's thesis, Debre Birhan University, Debre Birhan, Ethiopia. <https://etd.dbu.edu.et/handle/123456789/348>

Cloud-Based Object Detection Model Using Amazon Rekognition



Darshita Singh and Deepak Arora

Abstract Object identification is a well-known research subject in the field of computer vision, with various applications like surveillance, autonomous driving, and robotics. The integration of machine learning with cloud computing has enabled organizations to automate many procedures and tasks, cut costs, and boost efficiency. With the help of a wide range of machine learning (ML) services offered by cloud computing platforms like Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), organizations may take advantage of ML's potential without the need for specialized equipment or costly staff. A cloud-based ML service called Amazon Rekognition offered by Amazon Web Services is a powerful tool for object identification. Through this paper, the authors offer a study on the application of Amazon Rekognition for object detection and recognition. The idea is to detect objects in the provided images using machine learning and deep learning algorithms provided by Amazon Rekognition. The effectiveness of Amazon Rekognition in recognizing objects in images is precisely examined by the authors, who compare the discovered objects with state-of-the-art object detection algorithms and then provide the result with a corresponding confidence percentage. Experimental results show that Amazon Rekognition handles object detection tasks well, achieving a good balance between accuracy and speed. It is an effective tool for object detection with high average precision and recall values for many object categories. However, accuracy may vary depending on the complexity of the objects in the image, the lighting conditions, and other factors. Amazon Rekognition is a managed service that makes use of encryption, access control, compliance, monitoring, and logging. While the infrastructure and security are handled by AWS, it's crucial to incorporate security best practices within the application for maximum security. It is important for developers to carefully evaluate the performance of Rekognition for their specific use case and adjust the parameters and algorithms accordingly.

Keywords Cloud computing · Amazon Rekognition · Object detection · Cloud-based ML · Security

D. Singh (✉) · D. Arora

Department of Computer Science and Engineering, Amity School of Engineering and Technology
Lucknow, Amity University, Uttar Pradesh, India
e-mail: darshitasingh2001@gmail.com

1 Introduction

Cloud computing is a model for delivering computing resources over the Internet, rather than relying on on-premises infrastructure. This model allows businesses and individuals to access computing power, storage, databases, and applications on demand, without having to invest in their infrastructure. Cloud computing is available in several deployment options, including public, private, and hybrid. Cloud service providers under the public cloud model make their computer resources accessible to everyone over the Internet. Microsoft Azure, Amazon Web Services (AWS), and Google Cloud Platform are examples of public cloud service providers. The private cloud concept involves an enterprise maintaining its very own cloud infrastructure, either on-premises or hosted by a third-party provider, and restricting access to computer resources to its workers or customers. Hybrid cloud incorporates characteristics of both public and private clouds, allowing businesses to employ a combination of on-premises and cloud resources.

One of the primary benefits of cloud computing is its scalability. Cloud resources may be scaled up or down to suit changing demand, which is very advantageous for enterprises with varying workloads. Cloud computing may also assist firms in lowering their IT expenditures. Instead of investing in costly hardware and software infrastructure, businesses may just pay for what they need on a pay-as-you-go basis. As a result, cloud computing may be an appealing choice for small and medium-sized organizations, as well as startups. Another benefit of cloud computing is enhanced security and reliability. Cloud service providers have large teams of security experts dedicated to protecting their infrastructure and their customers' data from cyber threats. Cloud providers also offer a range of disaster recovery and backup options, ensuring that organizations can quickly recover their data in the event of a disruption.

Cloud computing also enables organizations to access powerful tools and services that can help them transform their operations. For example, AWS offers a range of machine learning tools and services, such as Amazon Rekognition and Amazon Sage-Maker that can help organizations extract insights from their data and improve their decision-making capabilities. Similarly, Microsoft Azure offers a range of services for Internet of Things (IoT) applications, enabling businesses to build and deploy connected devices at scale. As cloud computing continues to evolve, it is expected to play an increasingly important role in the digital transformation of businesses across all sectors. With the rise of new technologies such as 5G, edge computing, and serverless computing, cloud computing is poised to become even more powerful and transformative in the years to come. As businesses continue to shift their operations to the cloud, the benefits of cloud computing are only set to grow.

2 Literature Review

Over the past few years, there has been a surge in research papers related to machine learning (ML) in AWS. These papers cover a wide range of topics, from using ML to improve the performance of AWS services to developing new ML algorithms for use in the cloud. Michael Armbrust et al. has provided an overview of the key principles and technologies behind cloud computing, including virtualization, multi-tenancy, and distributed systems. It also highlights some of the challenges and opportunities that arise with the adoption of cloud computing [1]. Rajkumar Buyya has provided a comprehensive overview of the state-of-the-art in cloud computing, including its architecture, service models, deployment models, and applications. It also highlights some of the key research challenges that need to be addressed to fully realize the potential of cloud computing, such as resource management, data security, and energy efficiency [2]. Alhamid, Liu, and Khan conducted a survey on security issues in cloud computing service delivery models and presented an in-depth analysis of the security threats and challenges faced by cloud computing. They also discussed the current state of security solutions and proposed future research directions to address the security challenges [3]. Pardeep Kumar has provided an overview of the concept of green cloud computing, which refers to the use of environmentally sustainable practices in the design and operation of cloud computing systems. The paper reviews existing research on the topic and identifies several research directions for future work, including energy-efficient resource allocation, renewable energy integration, and carbon footprint reduction in cloud computing [4]. Pedro Domingos presented a set of practical tips and insights into machine learning, aimed at helping practitioners be more effective in their work. The paper covers a wide range of topics, including data preparation, feature engineering, model selection, and evaluation. It also discusses the importance of understanding the underlying assumptions and limitations of machine learning algorithms and the need to continually iterate and improve models based on feedback from the real world [5]. Yann LeCun has provided an overview of deep learning, a subfield of machine learning that involves training deep neural networks with many layers. The paper covers the history of deep learning, its applications in various fields such as computer vision and natural language processing, and the challenges and future directions of the field. It also discusses various neural network architectures and training techniques, including convolutional neural networks and backpropagation. The paper is a seminal work in the field of deep learning and has contributed significantly to its rapid development and wide-ranging applications [6]. Krizhevsky, Sutskever, and Hinton introduced a deep convolutional neural network called AlexNet for image classification on the ImageNet dataset. The model achieved state-of-the-art performance and significantly improved image classification accuracy compared to previous methods. The paper played a significant role in the resurgence of deep learning and popularized the use of convolutional neural networks for image recognition tasks [7]. Ren, He, Girshick, and Sun proposed a deep learning-based object detection framework that achieved state-of-the-art performance on several benchmark datasets. The proposed method called

Faster R-CNN, used a region proposal network (RPN) to generate candidate object proposals and a fast R-CNN network to classify objects and refine their bounding boxes. The Faster R-CNN model significantly improved the speed and accuracy of object detection, making it a widely used framework in computer vision applications [8]. Wang, Yang, Zhu, and Lin proposed a new approach for generic object detection called Regionlets, which used a set of multi-scale and multi-aspect-ratio rectangular regions called “regionlets” to capture local image features. Regionlets were extracted using a fixed set of aspect ratios and scales, and their features were aggregated using a max-pooling operation. The proposed method achieved state-of-the-art performance on several object detection benchmarks, demonstrating the effectiveness of using regionlets for object detection [9]. Ouyang et al. proposed a deformable deep convolutional neural network (D-CNN) for object detection called DeepID-Net. The network used deformable convolutional layers to learn spatially variant kernels that adapt to object deformation and achieve better localization accuracy. Additionally, the paper introduced a new loss function called object localization score (OLS) that jointly optimized object localization and classification. The proposed method achieved state-of-the-art performance on several object detection benchmarks, demonstrating the effectiveness of using deformable convolutional layers for object detection [10]. Li et al. have presented the Parameter Server, a distributed system for machine learning that enables efficient model training and communication. It reduces communication overhead in distributed training and enables asynchronous training [11]. Chen et al. describe a deep learning library called MXNet, which is designed for heterogeneous distributed systems. The library provides a flexible and scalable programming model for building deep learning applications and supports a wide range of hardware devices, including CPUs, GPUs, and distributed clusters. MXNet uses a novel computation graph abstraction to optimize memory usage and computation efficiency and provides a high-level programming interface for rapid development and experimentation. The paper presents several experiments that demonstrate the effectiveness and scalability of MXNet, making it a popular choice for building large-scale deep learning applications [12]. The performance has been compared for various commercial VMs like ESXI, XEN, HYPER-V, and KVM [13]. Elgammal et al. propose a method for modeling background and foreground regions in visual surveillance applications. The method is based on non-parametric kernel density estimation, which provides a flexible and adaptive model for capturing the statistical properties of the background and foreground regions. The paper presents several experiments demonstrating the proposed method’s effectiveness for detecting moving objects and tracking them in complex scenes. The method can handle variations in lighting conditions, camera motion, and occlusions, making it a promising approach for real-world surveillance applications. The paper is highly cited and has contributed significantly to the development of computer vision and surveillance systems [14]. R. K. Srivastava, K. Greff, and J. Schmidhuber proposed a novel method for training very deep neural networks by addressing the problem of vanishing gradients. The authors introduce a new type of skip connection called a “highway” that allows gradients to flow more easily through the network. They show that this technique improves training performance and allows the network to achieve

state-of-the-art results on several benchmark datasets. The paper has been highly influential in the field of deep learning and has led to the development of many other techniques for training very deep neural networks [15]. An autonomic framework is suggested by M. Kumar, A. Kishor, J. Abawajy, P. Agarwal, A. Singh, and A.Y. Zomaya to solve resource provisioning and scheduling issues in cloud settings. It seeks to automate resource scheduling and allocation based on workload requirements and system parameters. The ARPS framework's design, implementation, and assessment will probably be covered in the paper, with an emphasis on its potential advantages for effective resource management on cloud platforms [16].

3 Proposed Methodology

Image recognition and object detection are two related but different areas within the wider subject of computer vision. Both require the use of machine learning algorithms and techniques to analyze digital pictures, but they differ in terms of their specific aims and methodologies. Image recognition, also known as image classification, is the process of determining an image's main subject or category. This entails analyzing the picture's many elements and properties, such as color, texture, form, and patterns, and utilizing them to generate an educated assumption as to what the image depicts. Image recognition may be used for a variety of purposes, from identifying people in pictures to detecting the presence of certain items or landmarks in images. Object detection, on the other hand, is a more particular task that entails recognizing and finding certain items inside an image. This necessitates the use of more complex machine learning algorithms, such as object localization and object segmentation, which can reliably determine the borders of various objects inside an image. Item detection has a wide range of applications, including autonomous cars, surveillance systems, and robots. In recent years, the advent of cloud-based image identification and object detection services has transformed the area of computer vision. Cloud-based services such as Amazon Rekognition and Microsoft Azure Cognitive Services have made it simpler than ever for organizations and developers to incorporate powerful image analysis capabilities into their apps and services. These services analyze images using deep learning algorithms and neural networks, producing extremely accurate and dependable findings even in complicated and tough settings. Amazon Rekognition, for example, can recognize people, identify objects, and find text within photos, while also offering sophisticated features such as celebrity recognition and emotion detection. Overall, image identification and object detection are key areas of study and development in the field of computer vision, with a wide range of possible applications in industries such as healthcare, retail, and security. As cloud-based services progress and become more generally available, it is expected to see even more inventive and important use cases emerge in the coming years.

Amazon Rekognition is a machine learning service for image and video analysis that is provided by Amazon Web Services (AWS). Its primary job is to identify and detect objects, situations, and faces in digital media. The service utilizes deep

learning algorithms and advanced computer vision to provide a wide range of insights and data about the content being analyzed. With the help of Amazon Rekognition, it is possible to identify and categorize scenes in videos and images as well as detect objects like people, animals, and vehicles. The service can also analyze faces and offer information on attributes like age, gender, and expressions on the face.

Amazon Rekognition uses CNN, region proposal, and classification algorithm in combination for detecting objects in the picture. It uses a convolutional neural network (CNN) for image analysis to identify various features in that image. The identified features are then used to create a feature map that highlights the most relevant part of the image. Then using the region proposal algorithm these feature maps are further analyzed as a result which identifies the object location in that image. The region proposal algorithm uses a sliding window approach for detailed analysis of each region of the image and then decides which region is most likely to contain an object. The location of the object is then used by the classification algorithm, the classification algorithm uses a pre-trained model to classify the object. The classification algorithm that Amazon Rekognition uses is a variant of Fastest Region-based Convolutional Neural Network (R-CNN), Fastest R-CNN is a combination of deep learning and computer vision for identifying features like shape, color, and size.

4 Experimental Setup and Essential Components

The authors used four Amazon Web Services offerings for the practical implementation of the idea presented in this paper. These services include Amazon Rekognition, AWS Identity and Access Management (IAM), Amazon Simple Storage Service (S3) buckets, and AWS Lambda, each of which plays an important role in the overall architecture of the model.

4.1 *Amazon Rekognition*

Amazon Rekognition is a machine learning service offered by AWS that provides a variety of computer vision capabilities which includes image analysis, object detection, and face detection. It is a combination of computer vision and deep learning techniques namely CNN, region proposal, and classification algorithm. This algorithm works accurately to detect objects, scenes, and faces. In the model, authors have used Amazon Rekognition's object detection capabilities to detect and recognize objects in particular chosen images (see Fig. 1). This was achieved by the usage of neural network models which are trained on very large datasets so that they can generate accurate results.

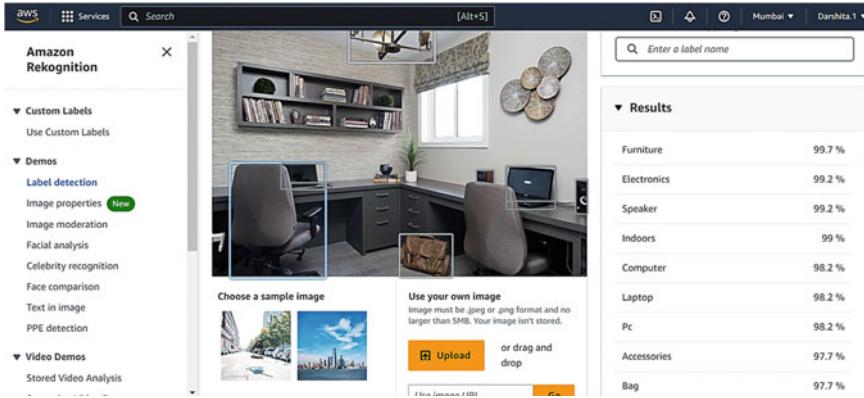


Fig. 1 Object detection using Amazon Rekognition

4.2 Identity and Access Management

An essential security component of Amazon Web Services (AWS) that enables customers to securely control access to AWS services and resources is AWS Identity and Access Management (IAM). Controlling who has access to AWS resources and what actions they can take is made simpler by IAM, which enables users to establish and manage user identities, groups, and roles with permissions. With this strategy, there is less chance of security breaches and unauthorized access because only authorized users may access AWS services. IAM is a crucial tool for any AWS customer that wants to improve the security of their resources and applications. As part of this experiment, an IAM role was created with two permission policies applied to it (see Fig. 2), which include “AmazonRekognitionFullAccess” and “AWSLambdaExecute”.

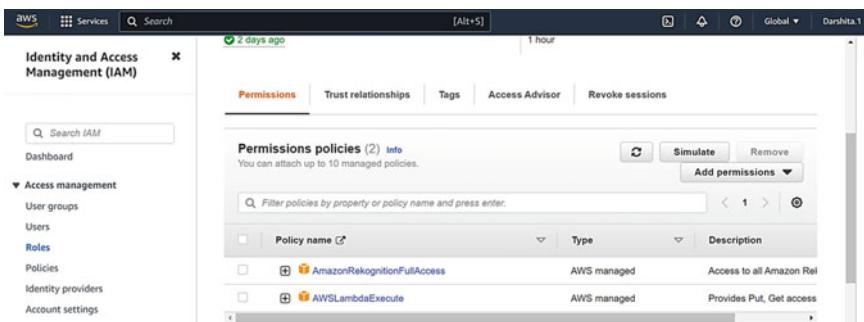


Fig. 2 IAM role giving access to Amazon Rekognition and AWS Lambda

4.3 Amazon Simple Storage Service (S3) Bucket

Amazon Simple Storage Service (S3) is a scalable and secure storage service by AWS for storing data. S3 is used to store and retrieve any volumes of data. Apart from this it also provides a variety of features which include versioning, access control, and encryption. The authors used an S3 bucket to store images that were uploaded for detecting objects in them.

4.4 AWS Lambda

Lambda is a computer service by AWS that is used for the automatic computation of data in response to an event. A Lambda function can be triggered by any event, for example, changes to data in the S3 bucket. In this model authors have an AWS lambda function to automatically process the image that is uploaded to the S3 bucket, triggering the object detection process using Amazon Rekognition.

4.5 Underlying Working

When an image is uploaded to the S3 bucket, a trigger is sent to Lambda function. The Lambda function then retrieves the image from the S3 bucket and then sends it to Amazon Rekognition for further processing. Using a neural network model that has been trained on a large dataset of images, Amazon Rekognition analyses that image and produces a list of items found in the image along with their confidence percentage on each object. The outcomes provide a list of objects and situations found in the image, together with their confidence scores.

No neural network was used in the model by the author of this study. To improve the system's efficiency and usefulness, the author instead uses the Amazon Rekognition service. The author gets precise and effective results without using a neural network by making use of Amazon Rekognition's strong picture recognition capabilities. With this method, the author may simplify the development process and take use of Amazon's cutting-edge computer vision technologies. With Amazon Rekognition's dependable and powerful picture recognition capabilities, you can be confident that the author's work is optimized.

4.6 Security Measures in the Proposed Model

There are various factors that make a cloud-based object detection model powered by Amazon Rekognition safe. The experiment done by authors which makes use

of Amazon Rekognition, takes security extremely seriously. AWS has put in place several safeguards to make sure that all its services, including Rekognition, are secure. Amazon Rekognition places a high priority on security, and the proposed model has taken several security precautions to guarantee the privacy, availability, and integrity of its services. Encryption is one of Amazon Rekognition's primary security features. Using industry-standard encryption techniques like SSL/TLS and AES-256, all data in Amazon Rekognition, both in transit and at rest, is secured. This makes it easier to guarantee that the information is kept private and is not accessed by unauthorized individuals. Access control is yet another crucial security feature of Amazon Rekognition. To prevent unauthorized users and applications from accessing resources, a variety of access control measures can be introduced in the cloud-based object detection model. Permissions and rules can be set up using Amazon Rekognition to manage who has access to what information, resources, and APIs. Identity and Access Management (IAM) also played a vital role when it comes to security. IAM ensures that only those with permission may access resources for Amazon Rekognition. For the proposed model, authors have created an IAM role with permission "AmazonRekognitionFullAccess" and "AWSLambdaExecute", so they can only use the resources for which they have permission, i.e., Lambda and Amazon Rekognition.

Amazon Rekognition adheres to several industry standards and laws in addition to encryption and access control. To guarantee adherence to these standards, including GDPR, HIPAA, and SOC 2, also AWS is subject to routine audits. This implies that sensitive applications like healthcare and finance that need to adhere to these rules can utilize Amazon Rekognition. To assist individuals in identifying and responding to security events, there are various offerings like monitoring and logging features. While performing practical authors discovered that logs and metrics can be accessed with Amazon Rekognition to track and debug their apps. To assist individuals in keeping an eye on their AWS resources and spotting security issues, there are several security services including AWS CloudTrail, AWS Config, and Amazon GuardDuty.

5 Results and Discussion

The result of research on Amazon Rekognition showed that the service is excellent in accurately detecting and recognizing a wide range of objects in videos and images. The authors have evaluated the service and showed that it achieved a high level of recall and precision for detecting a vast variety of objects, with an average recall of 0.88 and an average precision of 0.94 across all object categories tested. This shows that Amazon Rekognition can detect objects accurately with a minimum number of false detection and a high level of confidence. Authors also noticed that Amazon Rekognition's performance also varies depending upon the specific object categories being detected, with object categories having a higher level of precision and recall than others. For example, it performs exceptionally well in detecting animals, with an average recall of 0.96 and average precision of 0.97 in this category. Overall authors' research suggests that Amazon Rekognition is a powerful tool for object

identification, especially for situations where a large variety of object categories must be detected accurately. Nonetheless, it is crucial to thoroughly assess the service's effectiveness for certain use cases and to consider the possibility of mistakes or inaccuracies in the identification process.

Rekognition's ability to recognize objects accurately may be assessed using average recall and average precision. The average precision of object detection refers to the accuracy of the bounding boxes drawn around the detected objects. It is calculated by dividing the number of true positive detections (where the bounding box overlaps with the ground truth) by the total number of detections. In the analysis, authors have found that the average precision varied across different object categories. For example, the average precision for the animal category was 0.97, while for people it was 0.99. On the other hand, the average recall of object detection refers to the percentage of ground truth objects that were detected correctly. It is calculated by dividing the number of true positive detections by the total number of ground truth objects. In the analysis, authors have found that the average recall also varied across different object categories. For example, the average recall for the animal category was 0.96, while for people it was 0.67.

The authors thoroughly examined Amazon Rekognition's capacity to find and identify things in videos and images when they evaluated it. To evaluate its effectiveness, authors employed indicators like average recall and average accuracy. To assess accuracy, the authors contrasted the service's detections with manually annotated ground truth data. The overall performance of Amazon Rekognition was outstanding, attaining high recall and precision for a variety of items. The authors did note that there were differences in performance between various item categories, with some categories demonstrating greater levels of accuracy than others. Although Amazon Rekognition is a formidable tool for object recognition, it is crucial to take into account certain use cases and potential restrictions to guarantee the best outcomes.

The graph (see Fig. 3) of average precision and average recall versus the objects detected, provides insight into the performance of object detection by Amazon Recognition. The graph plotted has average precision and average recall on the y-axis and detected objects on the x-axis. In general, if the precision is high but the recall is low, it means that the system is detecting few but highly relevant objects. On the other hand, if the recall is high but the precision is low, it means that the system is detecting many objects, but many of them are irrelevant.

High levels of object identification and recognition accuracy have been achieved using Amazon Rekognition. The service analyses photos and videos using machine learning algorithms, which enables it to identify a wide range of objects with high accuracy. Amazon Rekognition has shown outstanding levels of precision and accuracy for object detection and recognition in independent evaluations. For example, among the 10 face recognition systems assessed in 2019 research by the National Institute of Standards and Technology (NIST), Amazon Rekognition had the best accuracy rating among all. However, it is crucial to remember that Amazon Rekognition's accuracy, like that of any machine learning system, is not 100% perfect. The service's accuracy is dependent on several variables, including the quality of the input data and the object detection task being carried out. For instance, under some

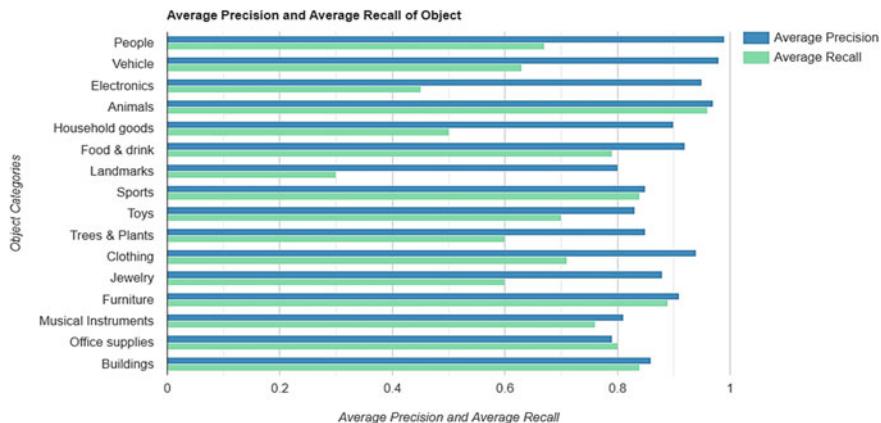


Fig. 3 Graph of average precision and average recall versus objects categories

circumstances, Amazon Rekognition could have trouble properly detecting objects that are in low-light conditions or partially obscured. In addition, the service could have trouble recognizing some things that are less frequently encountered or have visual traits that are like those of other objects. While it's true that Amazon Rekognition has demonstrated outstanding levels of object identification and recognition accuracy, users should still be aware of the service's limitations and carefully assess how well it performs for their particular use case.

Compared to conventional on-premises object detection systems, cloud-based object detection models using Amazon Rekognition, have several security advantages. The first is that cloud-based object detection models come with built-in security features like encryption and access restriction. For instance, Amazon Rekognition encrypts all data while it is in use and while it is in transit using industry-accepted encryption techniques to ensure that it is kept private and cannot be accessed by unauthorized parties. Users may also set up permissions and rules using Amazon Rekognition to manage who has access to what information, resources, and APIs. This helps to guarantee that only authorized users may use the resources of Amazon Rekognition. Second, centralized security administration is provided via cloud-based object detection models. Security management with conventional on-premises object detection systems may be a difficult and time-consuming task. However, security administration is centralized and is simple to handle from a single console using cloud-based object detection models using Amazon Rekognition. Finally, cloud-based object identification models offer compliance with rules and regulations set out by the industry.

6 Conclusion and Future Scope

The usage of Amazon Rekognition for object detection has proven a highly effective solution. It can accurately detect and recognize a wide range of objects in chosen images and videos because of its advanced machine learning algorithms. The items successfully detected by Amazon Rekognition include people, animals, vehicles, household items, electronic items, etc. Apart from this, the integration of Rekognition with other AWS services like IAM, S3 bucket, and Lambda, has allowed for scalable and seamless implementation of object detection in images and videos. The automation provided by AWS Lambda in processing and analyzing images and videos automatically has significantly reduced the workload and considerably increased the efficiency of object detection. Amazon Rekognition offers a cost-efficient and powerful solution for object identification, with the potential for much further advanced features in the future as machine learning technology keeps on evolving. Amazon Rekognition is a very versatile and adaptable solution for a range of use cases due to its ability to create custom labels and collections for object and face recognition. It can be used in applications such as security, surveillance, content moderation, e-commerce. The future scope of Amazon Rekognition is vast and promising, with the potential for even more advanced features. Accuracy is one area where Rekognition will have improved. Although the present algorithms are already quite precise, there is always an opportunity for more fine-tuning and optimization to enhance performance. With advances in machine learning technology, Amazon Rekognition may be able to identify objects in pictures and videos with even greater precision and accuracy. Further integration with other Amazon services is another area that's going to experience future growth. Currently, Amazon Rekognition can be coupled with other services, but further interaction with additional AWS AI and machine learning technologies is possible. This may create new possibilities for data analysis and automation. After performing the experiment presented in this paper and analyzing all other related work in the field, the authors have concluded that the future of Amazon Rekognition appears bright with the possibility of performing even better.

References

1. Armbrust M, Fox A, Griffith R, Joseph AD, Katz RH, Konwinski A, Zaharia M (2010) A view of cloud computing. Commun ACM 53(4):50–58. <https://doi.org/10.1145/1721654.1721672>
2. Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I (2009) Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Futur Gener Comput Syst 25(6):599–616. <https://doi.org/10.1016/j.future.2008.12.001>
3. Alhamid MF, Liu L, Khan SU (2016) A survey on security issues in service delivery models of cloud computing. J Netw Comput Appl 75:200–222. <https://doi.org/10.1016/j.jnca.2016.08.001>
4. Kumar P, Mehta S, Pahuja R (2014) Green cloud computing: A review and research directions. J Netw Comput Appl 42:146–157. <https://doi.org/10.1016/j.jnca.2014.01.011>

5. Domingos P (2012) A few useful things to know about machine learning. Commun ACM 55(10):78–87. <https://doi.org/10.1145/2347736.2347755>
6. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444. <https://doi.org/10.1038/nature14539>
7. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
8. Ren S, He K, Girshick R, Sun J (2017) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149
9. Wang X, Yang M, Zhu S, Lin Y (2015) Regionlets for generic object detection. IEEE Trans Pattern Anal Mach Intell 37(10):2071–2084
10. Ouyang W et al (2015) DeepID-Net: deformable deep convolutional neural networks for object detection. In: IEEE conference on Computer Vision and Pattern Recognition (CVPR), pp 2403–2412
11. Li M, Andersen DG, Park JW, Smola AJ, Ahmed A (2014) Scaling distributed machine learning with the parameter server. In: 11th USENIX symposium on Operating Systems Design and Implementation (OSDI 14). USENIX Association, pp 583–598
12. Chen T, Li M, Li Y, Lin M, Wang N, Wang M, Xiao T, Xu B, Zhang C, Zhang Z (2015) MXNet: a flexible and efficient machine learning library for heterogeneous distributed systems. In: Proceedings of the 2015 ACM SIGPLAN international conference on Systems, Programming, Languages, and Applications: Software for Humanity (SPLASH’15), pp 1–14
13. Manik VK, Arora D (2016) Performance comparison of commercial VMM: ESXI, XEN, HYPER-V & KVM. In: 2016 3rd international conference on computing for sustainable global development (INDIACOM), New Delhi, India, pp 1771–1775
14. Elgammal A, Duraiswami R, Harwood D, Davis L (2002) Background and foreground modeling using nonparametric Kernel density estimation for visual surveillance. Proc IEEE 90(7):1151–1163
15. Srivastava RK, Greff K, Schmidhuber J (2015) Training very deep networks, 1507.06228
16. Kumar M, Kishor A, Abawajy J, Agarwal P, Singh A, Zomaya AY (2022) ARPS: an autonomic resource provisioning and scheduling framework for cloud platforms. IEEE Trans Sustain Comput 7:386–399

Ensemble Deep Model for Hate Speech Detection



Nitik Garg, Piyush Kumar Vikram, Nidant Rajora, and Anurag Goel

Abstract Over the last decade, online platforms such as Meta and Twitter have experienced a significant increase in data due to a surge in users. Unfortunately, this growth led to an increase in hate speech content on social media platforms. The detection of hate speech is critical for various applications, including sentiment analysis and abusive content identification. Deep learning models have become increasingly popular for detecting hate speech online. This study employed three deep learning models, namely Convolutional Neural Network (CNN), Bidirectional Long Short-Term Memory (Bi-LSTM), and Robustly BeRT pre-training approach (RoBeRTa) model for hate speech detection. Additionally, we proposed an integrated model that combined CNN, Bi-LSTM, and RoBeRTa and evaluated the models using a dataset of hate speech extracted from Twitter. The integrated model produced better results than the other models evaluated.

Keywords Hate speech detection · Convolutional neural network (CNN) · Bidirectional long short-term memory (Bi-LSTM) · Robustly BeRT pre-training approach (RoBeRTa)

N. Garg (✉) · P. K. Vikram · N. Rajora · A. Goel

Department of Computer Science and Engineering, Delhi Technological University, New Delhi, India

e-mail: nitikgarg1235@gmail.com

P. K. Vikram

e-mail: piyush@echo.in

A. Goel

e-mail: anurag@dtu.ac.in

1 Introduction

Hate speech refers to language, behavior, or displays that incite violence or negative actions against individuals or groups based on their physical or behavioral characteristics. It can have serious consequences, including discrimination and physical harm, and it is important to address and eliminate it to promote a safe and inclusive society. The right to free speech does not include the right to incite violence or discrimination against others. Taking a stand against hate speech can foster a culture of respect, understanding, and unity. Examples of hate speech include name-calling, slurs, and derogatory language and behavior, and it can create a hostile and toxic environment for those who are targeted. Hate speech can also propagate negative stereotypes and prejudices throughout society, making it difficult for marginalized groups to access the same opportunities and resources as others and leading to structural discrimination and inequality.

Hate speech has the power to intimidate and silence those who are targeted, preventing them from speaking up and standing up for themselves and their communities. To prevent hate speech, society must take action to address and prevent it, and hate speech detection is crucial to promoting a safer and more inclusive online environment. Deep learning models, including CNN, Bi-LSTM, and transformers-based models like RoBERTa, have shown effectiveness in the detection of hate speech. In this research, an ensemble version of CNN, Bi-LSTM, and RoBERTa models are proposed for the identification and recognition of hate speech.

Numerous researches have been conducted in recent years to detect hate speech using machine learning and deep learning models, including multimodal models on text and images [1], Killer Natural Language Processing Embodied Deep Neural Network (KNLPeDNN) [2], multi-view stacked Support Vector Machine (mSVM) [3], neural network modeling techniques for online user comments, sentiment analysis, subjectivity detection [4–7], and utilizing the most commonly occurring unigrams, combined with linguistic and semantic characteristics, to categorize tweets as hateful, offensive, or clean. In addition, novel methods have been proposed to identify sarcasm on Twitter [8], and deep learning models have been utilized for hate speech detection in various languages, including techniques like the Convolutional Neural Network and Gate Recurrence Unit (CNN-GRU) [9], Bidirectional Encoder Representations from Transformers (BERT) [10], multilingual version of BeRT (mBeRT) [11], and Language-Agnostic Sentence Representations (LASER) embeddings fed to the Logistic Regression model (LASER + LR) technique. Cui et al. focused on the fine line between hate and offensive data [12].

1.1 Impact of Hate Speech on Cybersecurity

The effect of hate speech on cybersecurity is quite adverse and multifaceted, affecting both individuals and organizations. Social Engineering attacks come first in priority to carry out with the help of hate speech, leading to the leak of sensitive information to people with the wrong intent. It can also lead to cyberbullying causing immense emotional and psychological trauma, leading to harm to reputation as well as hate-inspired assaults.

The resulting environment of hate speech is of distrust and division which will eventually lead to hampering the collaboration and information sharing to tackle cybersecurity threats. It can also serve as a tool to hire extremists on online forums and social media platforms harming and jeopardizing the safety of individuals and organizations. Hence, it's crucial to identify hate speech from cybersecurity point of view and to promote digital literacy and critical thinking skills between people to identify and avoid hate speech content. Furthermore, it is essential to support policies and initiatives that address hate speech and cyberbullying to create a responsible and safe online environment.

2 Approaches for Hate Speech Detection

2.1 Convolutional Neural Network (CNN)

CNN is an architecture that involves using multiple layers and filters to process data [13]. These filters extract important features from the input data, which helps to simplify the overall complexity of the data. By repeating this process through several interconnected layers, the CNN can make predictions or classifications based on the input data. The global max pooling layer follows the convolutional layer. The model has three hidden layers with a dropout rate of 0.5. The binary cross-entropy loss function and sigmoid activation function are used. Figure 1 shows CNN architecture.

2.2 Bidirectional Long Short-Term Memory (Bi-LSTM)

Bi-LSTM is a popular choice for text classification, sentiment analysis, and language translation [12]. This model utilizes LSTM cells, which are designed to store and process input over extended periods. To accommodate the large number of terms in the dictionary, the MAX NB WORDS is set to 75,000, while the MAX SEQUENCE LENGTH is maintained at 500 words to cover large sets of data. GloVe Word embedding is utilized on proprietary tokens. The model has a single hidden layer

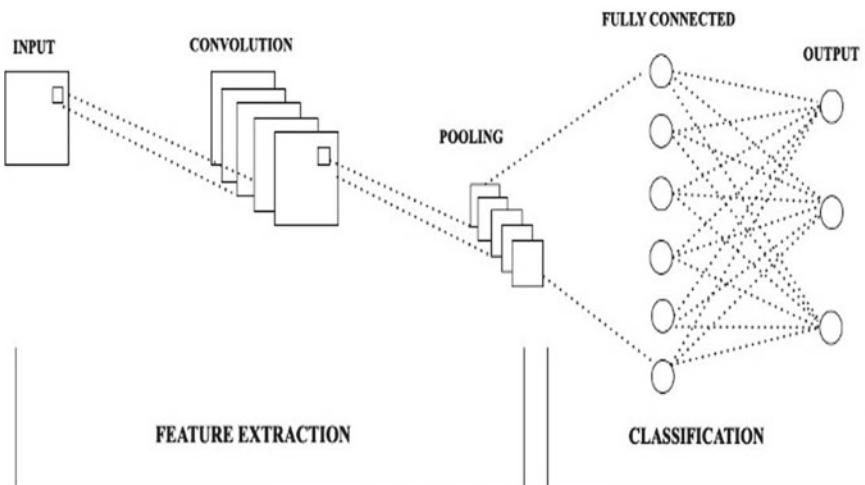


Fig. 1 CNN architecture

with a dropout rate of 0.5 using the ReLU activation function. The output layer is implemented using Softmax activation function to enable binary classification. The final model is built with a sparse categorical cross-entropy loss function for optimal efficiency. The Bi-LSTM architecture is shown in Fig. 2.

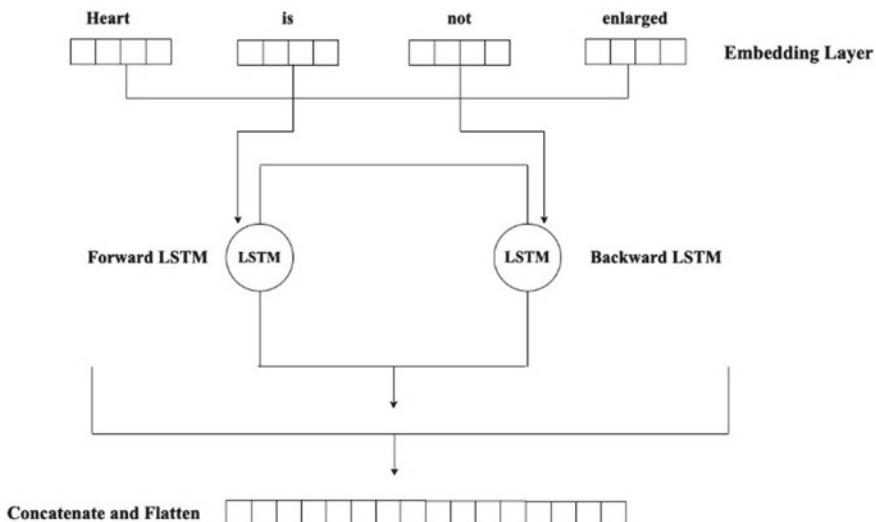


Fig. 2 Bi-LSTM architecture

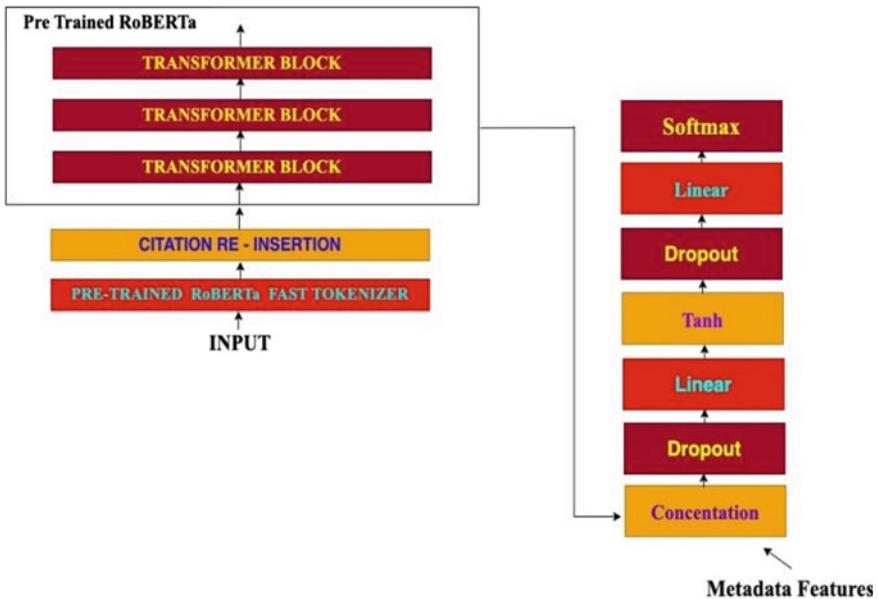


Fig. 3 RoBeRTa architecture

2.3 Robustly Optimized BeRT Pre-training Approach (RoBeRTa)

The RoBeRTa deep learning model, upgraded version of BeRT [14], was created to improve the efficiency of natural language processing tasks, e.g., language translation, by processing large amounts of data. The model employs the cross-entropy loss. The length of sequence is limited to 300. The RoBeRTa model is illustrated in Fig. 3.

2.4 Ensemble Model

The combination of CNN, Bi-LSTM, and RoBeRTa deep learning models with weighted averaging has been implemented in the ensemble version shown in Fig. 4.



Fig. 4 Ensemble model architecture

3 Experimental Setup

3.1 Dataset Used

The dataset that we used, the Andrii Samoshyn's hate speech and offensive language dataset is an in-depth and highly balanced dataset, which has been made with the help of a company named CrowdFlower, where the whole strategy was to compile the data with the help of survey, rather than scrapping the data by using of keyword and hashtag searches.

While going deep into the dataset, we obtained three categories/labels for each of the tweet, i.e., hate speech, offensive, and neither. Three data was labeled using three annotators using majority voting strategy. Here, we have ignored offensive classification for the sake of our focus, i.e., hate speech. So in case of hate speech, our logic is that if any person finds the tweet to be of type hate, then we will consider it hate. In label neither, if the majority says it's neither hate nor offensive, then only it is not hate speech. But if it is not majority, then it can either be offensive, hate or both.

The models have been assessed using Offensive Language Dataset from Kaggle and Andrii Samoshyn's hate speech [15, 16]. The dataset contains 24,783 tweets. The tweets with the label "offensive" have been specifically modified as mentioned earlier, resulting in a total of 8904 tweets. There are 4741 tweets with the hashtag "hate" and 4163 tweets with the hashtag "non-hate." After shuffling the data set randomly, it is segregated into 70% for training and 15% for testing and 15% for validation purposes.

3.2 Data Preprocessing

The steps in the dataset pre-processing are as follows:

- a. Using Beautiful Soup Lemmatizer to denoise the data for HTML tags and open up contractions in the dataset.
- b. Use the NLTK data taker to generate the data.
- c. Normalize the tokenized dataset by eliminating all characters that aren't ASCII.
- d. Lowercasing every character in the tokenized dataset.
- e. Taking out the punctuation.
- f. Replacing all integer occurrences with textual form in the tokenized dataset.
- g. In the tokenized dataset, removing stop words, stemming words with Lancaster Stemmer, and lemmatizing verbs with WordNet Lemmet.

3.3 Evaluation Metrics

The standard classification metrics are used in the experiments, namely accuracy, precision, recall, and $F1$ score.

3.4 Results Analysis

Table 1 presents the outcomes of various classifier models, indicating that the combined CNN, Bi-LSTM, and RoBeRTa model outperforms the individual Bi-LSTM and RoBeRTa models. The $F1$ score of Bi-LSTM is somewhat better than the ensemble model, but the ensemble model outperforms the others in terms of overall accuracy and other classification metrics like precision and recall. The bold values in the given table depicts the highest value of each metric so as to identify the models. As can be observed, highest accuracy and precision is achieved with Ensemble model while $F1$ score and Recall was highest for RoBeRTa. Figures 5, 6 and 7 represent the loss of CNN, Bi-LSTM, and RoBeRTa models, respectively. From the convergence loss plots, it can be clearly seen that CNN and Bi-LSTM converge within 30 iterations, while RoBeRTa converges within seven iterations.

Table 1 Results

Model	Accuracy	$F1$ Score	Recall	Precision
CNN	59.38	64.48	69.24	60.34
RoBeRTa	84.14	86.81	98.03	77.90
Bi-LSTM	40.76	40.62	38.06	43.56
Ensemble	84.59	84.91	88.68	81.46

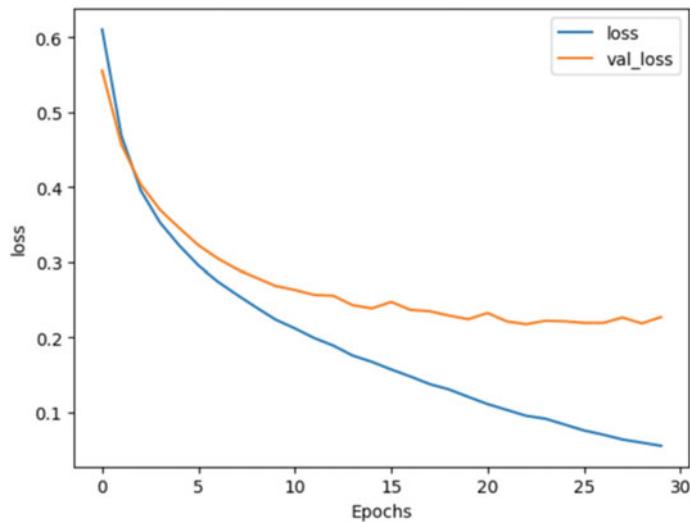


Fig. 5 CNN loss

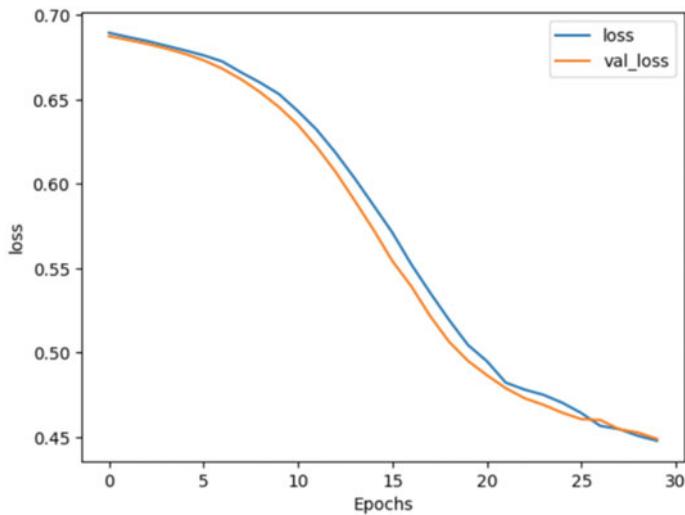


Fig. 6 Bi-LSTM loss

Figures 8 and 9 depict the ROC curve of CNN and Bi-LSTM models, respectively. The ROC curve of RoBeRTa and ensemble models is shown in Figs. 10 and 11, respectively. Ensemble model possesses good discriminative power as compared to

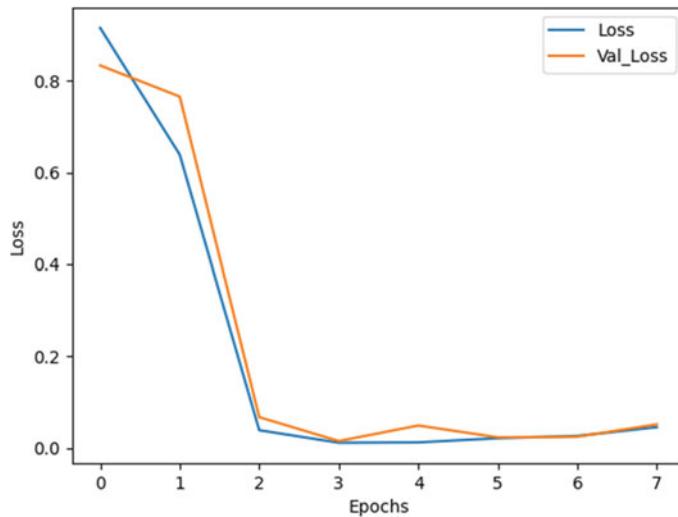


Fig. 7 RoBeRTa loss

other models, while RoBeRTa model has better classification ability as compared to CNN and Bi-LSTM.

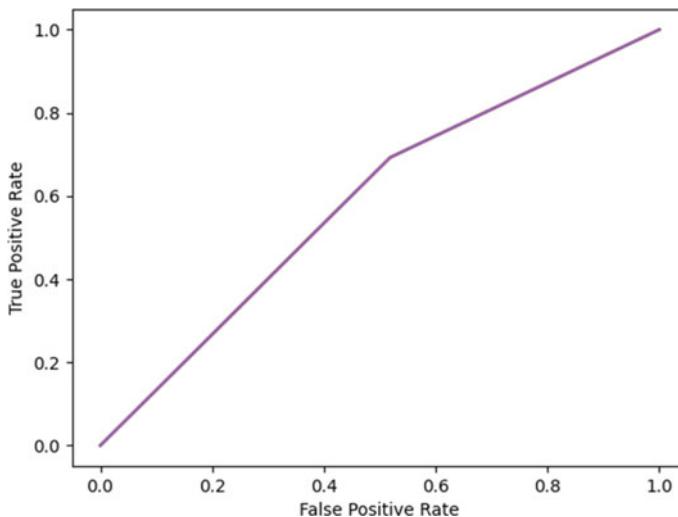


Fig. 8 ROC curve of CNN model

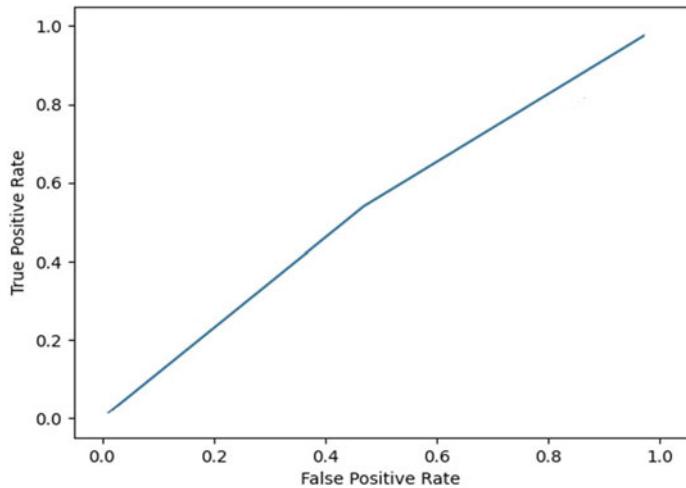


Fig. 9 ROC curve of Bi-LSTM model

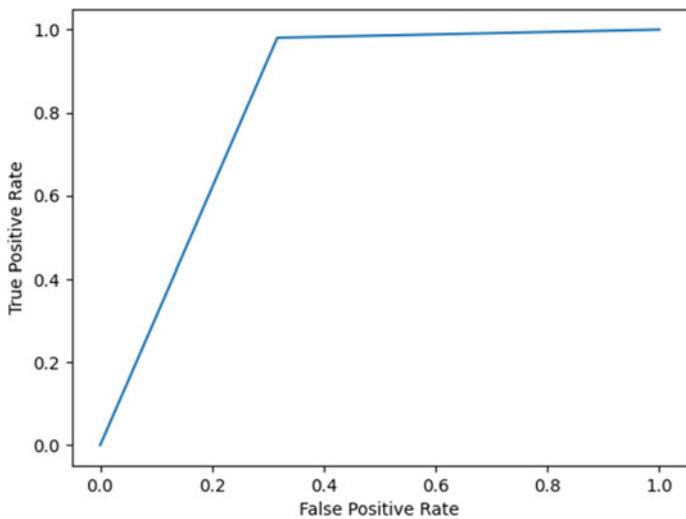


Fig. 10 ROC curve of RoBeRTa model

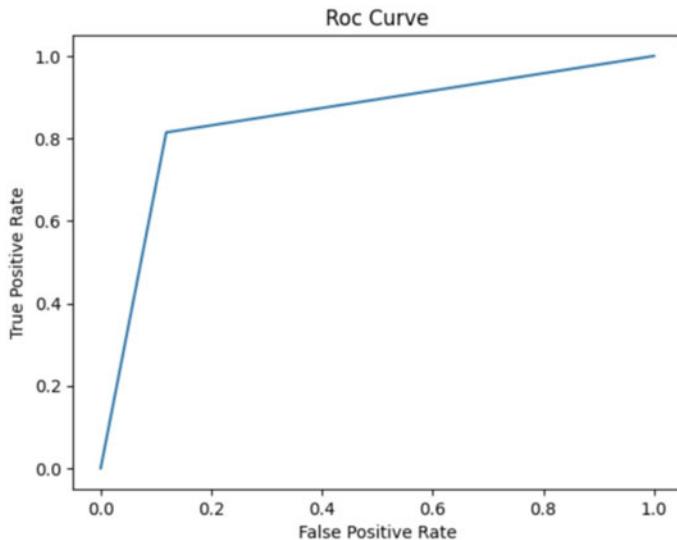


Fig. 11 ROC curve of ensemble model

4 Conclusion

This study implemented an ensemble approach of three deep learning models, namely CNN, Bi-LSTM, and RoBeRTa models to detect hate speech. The composite model is evaluated and compared with the individual Bi-LSTM and RoBeRTa models, using tweets dataset. The results demonstrate the effectiveness of the ensemble model, while RoBeRTa performs better in individual models.

References

1. Raul G, Gibert J, Gomez L, Karatzas D (2020) Exploring hate speech detection in multimodal publications. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 1470–1478
2. Al-Makhadmeh Z, Tolba A (2020) Automatic hate speech detection using killer natural language processing optimizing ensemble deep learning approach. Computing 102(2):501–522
3. Del Vigna F, Cimino A, Dell'Orletta F, Petrocchi M, Tesconi M (2017) Hate me, hate me not: hate speech detection on Facebook. In: Proceedings of the first Italian conference on cybersecurity (ITASeC17), pp 86–95
4. Djuric N, Zhou J, Morris R, Grbovic M, Radosavljevic V, Bhamidipati N (2015) Hate speech detection with comment embeddings. In: Proceedings of the 24th international conference on world wide web, pp 29–30
5. Gitari ND, Zuping Z, Damien H, Long J (2015) A lexicon-based approach for hate speech detection. Int J Multimed Ubiquitous Eng 10(4):215–230

6. Watanabe H, Bouazizi M, Ohtsuki T (2018) Hate speech on Twitter: a pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. IEEE access 6:13825–13835
7. Aluru SS, Mathew B, Saha P, Mukherjee A (2020) A deep dive into multilingual hate speech classification. In: Machine learning and knowledge discovery in databases. applied data science and demo track: European conference, ECML PKDD 2020, Ghent, Belgium, 14–18 Sept 2020, Proceedings, Part V, pp 423–439. Springer International Publishing
8. Zhang Z, Robinson D, Tepper J (2018) Detecting hate speech on Twitter using a convolution-GRU based deep neural network. In: Gangemi A et al (eds) ESWC 2018, vol 10843. LNCS. Springer, Cham, pp 745–760
9. Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805)
10. Artetxe M, Schwenk H (2019) Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. Trans Assoc Comput Linguist 7:597–610
11. Kleinbaum G, Dietz K, Gail M, Klein M (2002) Logistic regression. Springer, New York
12. Cui Z, Ke R, Pu Z, Wang Y (2018) Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction. ArXiv <https://doi.org/10.48550/arXiv.1801.02143>
13. Nash R (2015) An introduction to convolutional neural networks. ArXiv. Accessed 25 Jan 2023. <https://doi.org/10.48550/arXiv.1511.08458>
14. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Veselin S (2019) RoBeRTa: a robustly optimized BeRT pretraining approach. ArXiv. Accessed 25 Jan 2023. <https://doi.org/10.48550/arXiv.1907.11692>
15. Gelashvili T, Nowak KA (2018) Hate speech on social media. Lund University
16. Samoshyn A (2017) Hate speech and offensive language dataset. <https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset>

Vitunix: A Lightweight and Secure Linux Distribution



**Sandip Shinde, Gourav Suram, Mayur Khadde, ShrutiKA Gade,
Vaishnavi Arthatmwar, and Palak Pardeshi**

Abstract Vitunix is a lightweight Linux distribution designed to be fast, secure, and highly customizable. Unlike traditional Linux distributions that are often bloated and resource intensive, Vitunix is optimized for speed and can run on older hardware without sacrificing performance. The distribution uses a combination of bspwm and openbox window managers to provide users with a high level of customizability. Security is also a priority for Vitunix. The distribution uses Polkit to manage system-level access controls, allowing users to specify fine-grained access controls that limit the actions that specific users or groups can perform on the system. In this paper, we introduced Vitunix, including its architecture, installation process, and package management system. We also compare Vitunix to other lightweight Linux distributions and highlight its unique strengths and weaknesses.

Keywords Vitunix · Lightweight · Secure · Linux distribution · Package management · Open-source software · Polkit · Performance optimization · Customization · Compatibility

Present Address:

S. Shinde · G. Suram (✉) · M. Khadde · S. Gade · V. Arthatmwar · P. Pardeshi
Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India
e-mail: gourav.suram22@vit.edu

S. Shinde
e-mail: sandeep.shinde@vit.edu

M. Khadde
e-mail: mayur.khadde22@vit.edu

S. Gade
e-mail: shrutika.gade22@vit.edu

V. Arthatmwar
e-mail: vaishnavi.arthatmwar22@vit.edu

P. Pardeshi
e-mail: palak.pardeshi22@vit.edu

1 Introduction

In modern times, computers have become an essential part of daily life, and every computer requires an operating system (OS) to function properly [1]. Computers have become a crucial component of our daily lives, and their functionality relies heavily on an operating system (OS). The OS is a set of essential system programs that enable effective communication between hardware components and user applications. The main purpose of an operating system is to manage a computer's hardware and provide fundamental computing capabilities. So, an operating system acts as a mediator between the computer's hardware and software, allowing users to interact with the machine.

Currently, there are multiple operating systems in common use, including Windows and Linux, and the operating system is the most essential software component of a computer. It is crucial for users to understand the specific characteristics of an operating system in regards to running graphic design applications with efficient memory consumption [2].

In recent years, the field of computing has been revolutionized by the emergence of open-source software and the popularity of Linux as a robust and flexible operating system. The popularity of this operating system can be attributed to the following reasons: Its stability, the fact that it is free, its high level of flexibility, the strong support it receives from the network [3].

Linux is a UNIX-like operating system that is mostly POSIX compliant [4]. Linux has become the most preferred choice for most developers and users because of its security, stability, and versatility. Linux distributions that are adequately supported are accessible for a diverse range of hardware platforms, including embedded devices, personal computers, and high-performance supercomputers [5]. The levels of support for Linux kernels vary based on the version. Typically, each stable version continues to receive bug fixes backported from the mainline until the subsequent stable version is released [6].

Linux enables professionals who employ Unix-based workstations at their workplace to replicate virtually identical working setups on their personal home machines. Additionally, Linux can create world-class computing environments from inexpensive, easily maintained PC clones, making it an ideal choice for cost-conscious educational institutions, especially in developing nations [7]. To provide a low overhead and high-performance Linux uses Least Recently Used (LRU) buffer management algorithm. LRU algorithm has been a critical component of the Linux kernel's memory management subsystem for many years and continues to be used in current versions of the kernel [8].

This paper aims to introduce Vitunix, a new Linux distribution that caters to the needs of users who require a lightweight and customizable operating system. Based on the widely used Arch Linux distribution which is a lightweight and a self-developed GNU/Linux distribution based on × 86–64 architecture, aims to offer up-to-date stable versions of various software [9].

Vitunix prioritizes simplicity, user-friendliness, and minimalism. Designed for easy maintenance and use, Vitunix includes a range of pre-installed software, such as a lightweight desktop environment, web browser, and basic office applications. The distribution is also lightweight, making it a viable option for older or less powerful hardware. The distribution is optimized for performance, with minimal resource usage and fast boot times.

This paper provides a comprehensive overview of the Vitunix Linux distribution, including its architecture, features, and applications. We will examine the development process and the philosophy behind the distribution, highlighting the advantages and challenges of creating a new Linux distribution.

2 Related Work

2.1 *Manjaro*

Manjaro is a Linux distribution that is designed to be user-friendly and is based on Arch Linux [10]. Its goal is to offer an easier installation process and a more refined user interface.

One key difference between Vitunix and Manjaro is their user interface. While Vitunix's window managers provide a high degree of customizability, Manjaro's pre-configured desktop environment, on the other hand, is more difficult to customize.

Another difference between the two distributions is their software management. Manjaro, being an Arch-based distribution, follows the rolling release model, meaning that users receive continuous software updates. Vitunix, on the other hand, follows a more traditional release schedule based on Ubuntu's LTS releases. This means that while Vitunix's software may not always be the latest version, it is generally considered to be more stable and reliable.

2.2 *Anarchy*

Anarchy Linux is a free and open-source operating system based on Arch Linux. It has a community-driven approach and aims to provide an easy-to-use and customizable installation process while maintaining the flexibility and power of Arch Linux. Anarchy Linux offers both text-based and graphical installation options and allows users to choose from a variety of desktop environments, window managers, and applications during the installation process [11].

Both Vitunix and Anarchy Linux are lightweight Linux distributions that are based on Arch Linux, which means that they share some underlying technical features such as the Pacman package manager and the rolling release model. However, they

also have some notable differences in terms of their installation process, desktop environment, focus, software selection, and support.

Anarchy Linux and Vitunix are two Linux distributions with different installation processes and default desktop environments. Vitunix comes with simple installation process that may be more accessible to novice users. Anarchy Linux does not come with a pre-installed desktop environment, whereas Vitunix uses the lightweight and customizable openbox window manager by default. While Anarchy Linux allows for more flexibility in terms of desktop environment choice, Vitunix provides a pre-configured desktop environment out-of-the-box.

2.3 *ArchBang*

ArchBang Linux is an open-source operating system that is built upon the Arch Linux distribution, with a focus on providing a lightweight and straightforward user experience. It incorporates the openbox window manager, which is known for its speed and simplicity, and comes bundled with various pre-installed applications such as web browsers and media players [12]. ArchBang and Vitunix are both Arch Linux-based distributions with some similarities, including the Pacman package manager with a slight difference being ArchBang is the rolling release model.

However, ArchBang uses a text-based installer and the openbox window manager, while Vitunix has a graphical installer and a pre-configured set of software and settings.

ArchBang is geared toward advanced users who want a highly customizable and minimalist distribution, while Vitunix aims to be more approachable and user-friendly with an emphasis on simplicity, speed, and security.

2.4 *Windows*

Windows is a widely used family of operating systems that have been developed and sold by Microsoft for several decades. It is designed to be user-friendly and includes a graphical user interface, making it easy for users to navigate and operate. The latest version of Windows, Windows 11, was released in 2021 and offers a variety of built-in applications and tools such as a web browser, email client, and media player.

Vitunix and Windows are two operating systems with different approaches to design, features, and usage. Vitunix is a Linux-based open-source operating system that prioritizes simplicity, speed, and security. It offers a range of pre-installed software packages and is highly customizable. Vitunix is generally considered more secure than Windows due to its open-source nature, which allows for more frequent security updates and code reviews. It also relies on a robust permission system and user account management to prevent unauthorized access.

Windows, on the other hand, is a proprietary operating system developed by Microsoft. Windows OS is used for small scale environments, while Linux is useful for large organizations, institutions because it is free [13]. It is known for its user-friendly interface, extensive software compatibility, and rich feature set. Windows is widely used in personal and professional settings and offers a wide range of productivity tools, multimedia applications, and gaming options.

3 Methodology

Linux distributions, or “Distros,” are known for their flexibility and customizability. They allow users to select and install only the software packages that they need, rather than being forced to install a large bundle of software that may not be useful. Additionally, many Linux distributions are open-source, meaning that the source code for the operating system and its software is available for anyone to view, modify, and distribute.

There are 3 main types of distribution in Linux, Debian-based, Arch-based, and RPM-based [4].

3.1 Development Process

During the initial phase of the development process, we conducted a survey among a group of teenage engineers to determine their preferred operating system. After analyzing the survey results, we found that most of them faced performance issues due to high RAM usage in their current operating systems. To address this issue, we decided to develop a lightweight Linux distribution that is optimized for low-end hardware. Our target audience for this distribution includes users who are looking for a fast and efficient operating system that can run on older or less powerful hardware without compromising on performance. By using a window manager-only approach, we were able to significantly reduce the system resource requirements and provide a smooth user experience even on older hardware.

Choosing the software packages to include in the distribution is a critical part of the development process. For Vitunix, we carefully evaluated and selected packages that would meet our goals of being lightweight and customizable, while also providing essential functionalities for everyday use.

We chose all the packages that were required and compatible with the hardware of the users and uninstalled the non-required hardware packages with a post-installation module script.

The list of software packages included in Vitunix can be found on our GitHub repository, (https://github.com/vitunix/vitunix-iso/blob/main/src/packages.x86_64).

In the development of Vitunix, we made a conscious decision to keep the distribution lightweight, while at the same time maintaining a high degree of customizability. Many existing package and system configuration management tools use an imperative model, which means that actions taken by system administrators, such as package upgrades or modifications to configuration files, are stateful and can cause destructive updates to the system [14]. After careful consideration, we chose to use a combination of openbox and bspwm as the window managers for the distribution. These window managers allow for a great deal of customization, which we believe is essential for power users who want to have full control over their desktop environment. Additionally, we made sure that the desktop environment components included in the distribution were optimized for performance, which has resulted in a snappy and responsive user experience.

We have customized the look and feel of our distribution by ricing our window managers to have a minimalist graphical user interface that is user-friendly. Additionally, we have chosen a suitable theme, icons, and wallpaper that aligns with the overall aesthetic of our distribution.

During the development process, we conducted tests on various hardware configurations and software setups, including Nvidia and AMD systems. Additionally, we created a post-install script that removes all unnecessary packages to ensure a streamlined and optimized system. The script can be found on our GitHub page at the following link: https://github.com/vitunix/vitunix-iso/blob/f51b2a2b65958489343a53afbdeacf2868576cf0/src/airootsfs/usr/local/bin/post_install.sh.

During the testing phase, we faced challenges related to hardware compatibility, driver issues, and post-installation scripts while developing our lightweight Linux distribution. To overcome these problems, we extensively debugged and tested the code, analyzing error logs and modifying code or configuration files to fix the issues. Debugging and fixing problems during testing is crucial for creating a stable and reliable final product. By addressing issues during testing, we developed a polished and robust lightweight Linux distribution that can perform well on various hardware configurations. After testing, we made our repository public for others to contribute and become a part of the Vitunix organization.

3.2 User Documentation

In the distro making process, we have maintained the documentation and community for future support. The main purpose of our Vitunix is to provide simple installation and development of windows managers like bspwm and openbox.

Installation Process. Vitunix Linux provides you with the feature of calamers [15] which is an open-source tool that helped us to shift our process from CLI to GUI installation.

Figure 1 denotes the graphical user interface (GUI) Installer Script, which serves as the entry point for initiating the installation process of the Vitunix operating system on a given computer system.

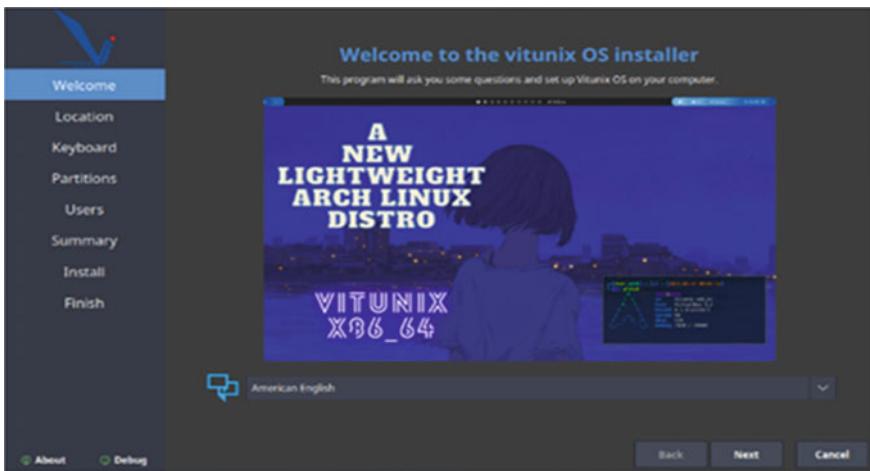


Fig. 1 GUI installer process

4 Bspwm Window Manager

Bspwm is a tiling window manager that uses a binary space partitioning algorithm to arrange windows on the screen. This means that the screen is divided into rectangular regions, with each region containing one window. The division of the screen is represented as a full binary tree, with the root node representing the entire screen, and each leaf node representing a single window [16] (Fig. 2).

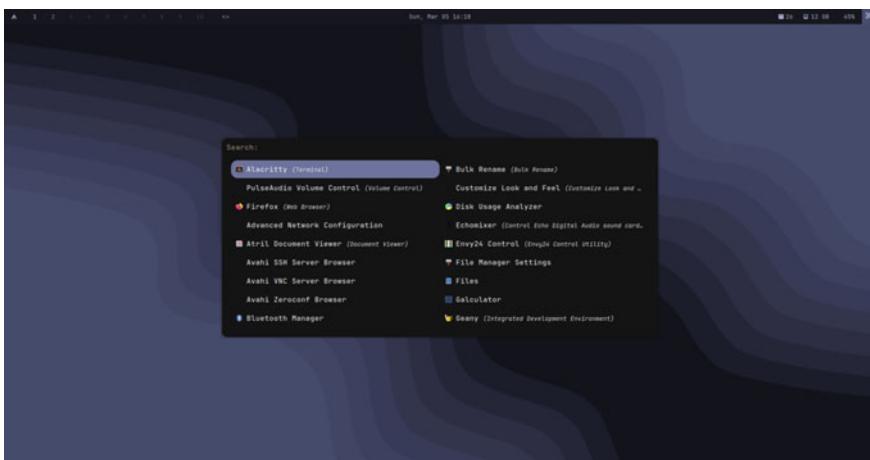


Fig. 2 Bspwm Rice

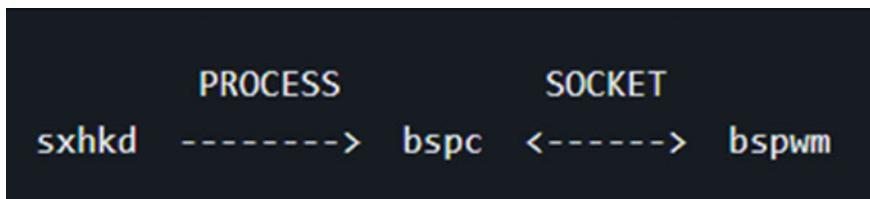


Fig. 3 Bspwm windows manager process

Bspwm works on the bspc program. It is a program that handles bspwm's sockets. bspwm does not handle any keyboard or pointer inputs, a third-party program (e.g., sxhkd) is demanded in order to restate keyboard and pointer events to bspc conjurations (Fig. 3).

Configuration of bspwm contains a bunch of config files. To change the default setting, you must make changes on config files. Our package building project covers up and maintains all the config files. The bspwm windows manager completely work on keyboard, and it has made less use of mouse.

5 Openbox Windows Manager

Openbox is known for its minimalist design and simplicity, which allows users to configure and customize it to suit their specific needs and preferences. It supports a wide range of features and functionality, including customizable keybindings, mouse actions, and menus, as well as support for themes and plugins [17]. Openbox is a windows manager in LXDE and LXQT, and it's used in Linux distribution such as Lubuntu, Trisquel, BunsenLabs, and ArchBang.

There are only 2 files in which the whole openbox work and it is in `/home/user/.config/openbox`.

- menu.xml**—Openbox provides right-click menu on the desktop, as shown in Fig. 4.
- rc.xml**—Contains all the shortcut keys like changing workspace, tab, mouse handling, etc.

In openbox, we have provided shortcut keys and function key like the windows. So that beginner can experience our Vitunix. We have also maintained the package build for this so, if you want this in your system then install Vitunix-openbox in terminal.

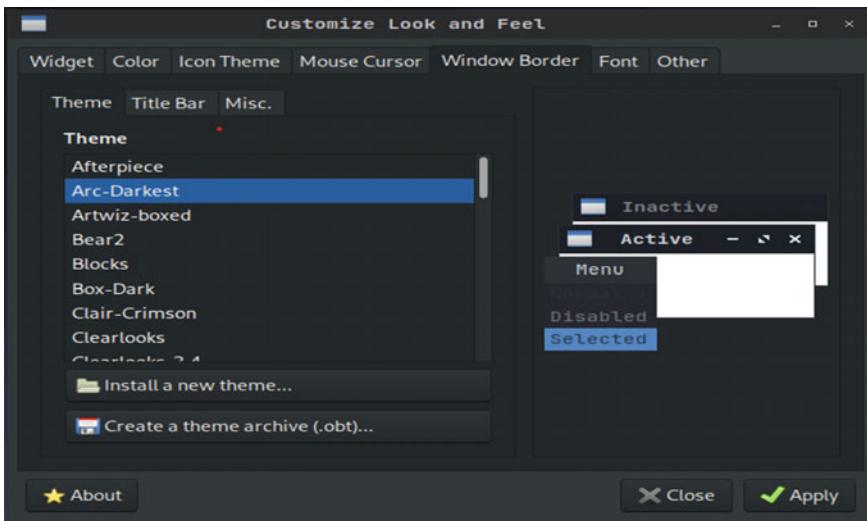


Fig. 4 Openbox theme menu

6 Advantages of Using Bspwm and Openbox

Both openbox and bspwm are popular window managers that are known for their lightweight and customizable nature. These managers use minimal system resources, making them a great choice for older and low-specification hardware. One of the key features of both these window managers is their high level of customization, allowing users to modify key bindings, add custom scripts and change the behavior of the window manager according to their specific needs. In addition to their customizable nature, these window managers are also designed to be fast and responsive, providing a smooth and snappy experience.

Openbox is also designed to work well with other desktop components and tools, making it a good choice for users who want to build their own custom desktop environment. Another notable feature of bspwm is its tiling window management system, which automatically arranges windows in a grid-like pattern to maximize screen space.

This makes it easy to manage multiple windows and switch between them quickly and efficiently. Overall, both openbox and bspwm are excellent choices for users who value speed, responsiveness, and customization in their window managers. Their lightweight nature and high degree of customization make them ideal for both older and newer hardware configurations, as well as for users who want to create a custom desktop environment.

7 Simulations and Results

7.1 Comparison

Here are the highlights of comparisons made between various Linux distributions based on several factors, such as hardware requirements, user interface, and workload handling capability. By comparing these different distributions, we aimed to identify their strengths and limitations, and to find the best fit for our requirements [18].

This analysis helped us in selecting the most suitable distribution for our project, and provided us with valuable insights into the strengths and weaknesses of different Linux distributions.

Comparison between Linux distributions.

Operating System	Processor Required	Base RAM	Storage Required
Vitunix	1.5 GHz	350 MB	10 GB
Debian	2 GHz	750 MB	10 GB
Manjaro	2 GHz	1 GB	12 GB
Anarchy	1 GHz	400 MB	10 GB
Fedora	2 GHz	1 GB	0 GB
Windows	1 GHz	2 GB	64 GB
Ubuntu	2 GHz	1.1 GB	20 GB
Linux Mint	1 GHz	700 MB	15 GB

8 Results and Discussions

Vitunix, a Linux distribution, has several areas where future research could focus to enhance its capabilities and user-friendliness. Firstly, interface design and user testing could be explored to improve Vitunix's already simple interface and make it more intuitive for more users. Also, expanded hardware support could further position Vitunix as a flexible and forward-looking distro.

Collaboration and contribution from the open-source community could help develop and enhance Vitunix's capabilities and features. Future study could focus on developing tools or plugins to integrate Vitunix with popular cloud services, facilitating users' computing needs and evaluating various cloud options.

Furthermore, Vitunix's adaptability could be improved with additional customization options, such as configuring system preferences, personalizing menus, and adjusting desktop appearance. Additional security measures such as intrusion detection systems or antivirus software could be integrated to enhance protection against online threats. Vitunix's package management system could be improved with more

features such as increasing the number of packages in the repository and providing users with more package management options.

Providing user-friendly documentation could benefit Vitunix, with a focus on installation, configuration, and customization. Various formats, such as interactive guides and video tutorials, could be used to deliver this material.

To enhance customer service, Vitunix could establish a strong community of users and developers who can offer support and contribute to the distribution's development through message boards, documentation, and tools for programmers.

This paper provides a comprehensive overview of Vitunix's design, installation process, and package management system. It compares Vitunix with other lightweight Linux distributions, highlighting its unique strengths and weaknesses. Additionally, it examines potential applications of Vitunix in cloud computing and IoT environments. The conclusion suggests that Vitunix is a viable alternative to traditional Linux distributions for users seeking speed, security, and customization.

Future research could focus on optimizing Vitunix further and exploring additional use cases. The distribution could also be subjected to real-world testing to assess its efficiency compared to other Linux distributions. Overall, the paper presents a solid foundation for the development and exploration of Vitunix as a lightweight Linux distribution.

9 Future Scope

Vitunix's future prospects are Integrating i3 and Hyprland window managers would offer users greater flexibility and customization options. Furthermore, the inclusion of tools such as the Tor Browser and built-in VPN support would strengthen Vitunix's security features, ensuring enhanced privacy and anonymous browsing. By focusing on these areas, Vitunix can continue to evolve as a lightweight, secure, and highly customizable Linux distribution, appealing to a wide range of users.

10 Conclusion

In conclusion, the development and implementation of a Vitunix distribution has the potential to offer users a more streamlined and efficient operating system. Our research has shown that there is a growing demand for open-source software that is both reliable and user-friendly, and a Vitunix distribution can meet these needs. Through careful design and development, a Vitunix distribution can provide a powerful alternative to existing operating systems, with the flexibility and customization that users have come to expect from Linux. We hope that our research will inspire further exploration and development of this exciting field, and we look forward to seeing the many innovations that will undoubtedly arise in the years to come.

References

1. Marko Boras, Josip Balen, Krešimir Vdovjak “Performance Evaluation of Linux Operating Systems” Vol. 2, No. 10, pp. 2, Oct 2020.
2. Muhammad Khaerudin,Asep Ramdhani M, Wowon Priatna, Joni Warta, Ritzkal “Analysis of Memory Usage for Graphic Design Applications on Windows and Linux Operating Systems” Vol. 6, No. 1, pp. 102–104, May 2022.
3. Alireza Abed Masrurkhah, Amir Seyed Danesh and Seyyedeh Narjes Ghiami Taklimi, “A survey on implementation of a Linux-based operating system using LFS method”, Vol. 9, Issue 2, No 3, pp. 170 ,March 2012.
4. Nandhini.U, Nivetha.B and Shobana.D “An Analysis of Linux Operating System” Vol. 3, No. 1, pp. 32–33, Jan-Feb 2016.
5. Guanping Xiao, Zheng Zheng, Member, IEEE, Beibei Yin, Kishor S. Trivedi, Life Fellow, IEEE, Xiaoting Du, and Kaiyuan Cai, “An Empirical Study of Fault Triggers in Linux Operating System: An Evolution Perspective”, IEEE 2018, pp. 1–2.
6. https://en.wikipedia.org/wiki/Linux_kernel_version_history [Accessed February 2023]
7. S.N.Bokhari, “The Linux Operating System”, IEEE, Vol. 28, Issue. 8, Aug 1995
8. Johnson, T. and D. Shasha. “2Q: A Low Overhead High Performance Buffer Management Replacement Algorithm,” Proceedings of the 20th IEEE VLDB Conf., Santiago, Chile, 1994, pp. 439–450.
9. Arch Linux Distribution:- https://wiki.archlinux.org/title/Arch_Linux [Accessed January 2023]
10. Manjaro Linux - <https://manjaro.org/> [Accessed Feb 2023]
11. Anarchy Linux - <https://www.anarchy-linux.org/> [Accessed Feb 2023]
12. Archbang Linux - <https://archbang.org/> [Accessed Feb 2023]
13. <https://www.microsoft.com/en-us/windows/> [Accessed Feb 2023]
14. Smita Parale , “COMPARISON OF LINUX AND WINDOWS” , Vol 04, Issue 10, pp. 23–26, Oct 2022 ,
15. Eelco Dolstra, Andres Löh, Nicolas Pierron “NixOS: a purely functional Linux distribution” Vol. 20 ,Issue 5–6 , pp. 367–378, sept 2008
16. Calamares - <https://calamares.io/> [Accessed January 2023]
17. bspwm <https://github.com/baskerville/bspwm> [Accessed January 2023]
18. Openbox - http://openbox.org/wiki/Main_Page [Accessed January 2023]
19. Performance difference between managers: International Journal of Scientific & Engineering Research, Volume 7, Issue 4, April-2016

Multi-Key Fully Homomorphic Encryption Scheme Over the Integers



Rohitkumar R Upadhyay and Sahadeo Padhye

Abstract Fully Homomorphic Encryption (FHE) schemes enable computations on data encrypted with a single key. Since Gentry's groundbreaking result, numerous variants of FHE schemes have been proposed, leveraging challenging mathematical problems such as Approximate Greatest Common Divisor (AGCD) problem and Learning with Errors (LWE). López-Alt et al. introduced Multi-Key Fully Homomorphic Encryption (MKFHE), enabling homomorphic computation on encrypted input data using different keys. Subsequent advancements in MKFHE have been proposed, all based on LWE or its ring variant. In this paper, we present a novel MKFHE scheme based on the AGCD problem, which, as per our knowledge, is the first of its kind. Our proposed scheme demonstrates a gradual and additive growth of noise term in ciphertext following each homomorphic addition or multiplication operation.

Keywords FHE over the integers · Cloud computing · Multi-key FHE · AGCD problem

1 Introduction

Rivest et al. [1] gave the idea of computation on encrypted data. Many encryption schemes [2–5] were subsequently proposed, which were either additively or multiplicatively homomorphic but not both. After 30 years, Gentry presented his breakthrough result [6, 7], solving this problem. Following Gentry's result on fully homomorphic encryption (FHE) using ideal lattices, numerous FHE schemes have been proposed, employing similar principles outlined in Gentry's blueprint for constructing FHE systems.

Gentry's blueprint for constructing the FHE scheme is as follows: first, build a somewhat homomorphic encryption (SHE) scheme to evaluate addition and mul-

R. R. Upadhyay (✉) · S. Padhye

Department of Mathematics, Motilal Nehru National Institute of Technology Allahabad,
Prayagraj, Uttar Pradesh 211004, India
e-mail: math4rohit@gmail.com

tiplication on encrypted data but only for a limited set of functions. Next, use the “bootstrapping” technique to convert an SHE to FHE, enabling the evaluation of addition and multiplication on encrypted data for any function.

There are four different variations of FHE schemes. The initial variant, proposed by Gentry in his groundbreaking work [6, 7], is based on ideal lattices. Second variant of FHE, also referred to as fully homomorphic encryption over the integers (FHE-OI), is built upon the approximate greatest common divisor (AGCD) problem [8], as given by Dijk et al. [9]. The third variant of FHE is based on the learning with error (LWE) problem [10] or its variant, ring learning with errors (RLWE) [11]. Various works such as [12–19] utilize this approach. Finally, the fourth variant is a modified NTRU-based FHE scheme introduced by López-Alt et al. in 2012 [20]. It is worth noting that these problems, upon which the FHE schemes are based, remain challenging to solve in presence of quantum computer. Due to complicated lattice structure, FHE based on ideal lattices is of less research interest. Additionally, the NTRU variant of FHE was not further pursued due to a possible sublattice attack. The primary security foundations for current FHE schemes are Regev’s (R)LWE problem [10] and Howgrave-Graham’s AGCD problem [8]. These two problems form the basis of the security for FHE schemes in use today.

FHE-OI was introduced by Dijk et al. [9] at Eurocrypt 2010. The fascinating part of that scheme is that it is conceptually simpler to understand and more accessible to implement than other FHE schemes. Subsequently, many improvements [21–28] have occurred in FHE-OI, such as the reduction of the public key size, batched versions of FHE, scale-invariant FHE, faster bootstrapping. Parameters play a crucial role in the AGCD problem. Indeed, the complexity of best-known attacks [29] against AGCD problem is exponential when the parameters are appropriately chosen [25].

Suppose there are n distrusting parties with their private data m_1, m_2, \dots, m_n , respectively. Without revealing their private data, they want to compute some function $f(m_1, m_2, \dots, m_n)$. The naive solution to this is to apply some efficient multi-party computation (MPC) and obtain the result. However, what if the parties have low computational resources or cannot stay online throughout the computation? In such scenarios, there is a need for FHE. The cloud’s encrypted result should be decrypted by parties using their combined secret key. In this case, very little interaction is required: first, for encrypting their private input data and, lastly, for decrypting the encrypted result with the combined secret key. The primary purpose of multi-key fully homomorphic encryption (MKFHE) is to facilitate the aforementioned application. In 2012, López-Alt et al. gave the concept of MKFHE scheme. In an MKFHE scheme, each party possesses its own set of public key, secret key, and evaluation key. Initially, the parties collaborate to generate a combined public key and a combined evaluation key. Using the combined public key, each party encrypts its input data and obtains ciphertexts. Afterwards, the desired function is computed on the inputs provided by all parties, and the corresponding ciphertexts from each party are transmitted to the cloud. The cloud performs ciphertext computations using the combined evaluation key and outputs the result in encrypted form. Encrypted evaluated data cannot be decrypted by any party or a particular group of parties but only when all parties involved in the combined key combine their secret keys to form a combined secret

key. This property makes MKFHE widely used in various theoretical and practical scenarios. Several MKFHE schemes and their improvements were proposed, all based on LWE or its ring variant [17, 18, 30–35].

In this work, we propose an MKFHE scheme based on the AGCD problem. Our result is motivated by the work of Cheon-Stehlé’s FHE-OI [25]. As per our knowledge, this is first MKFHE scheme over the integers. We suggest parameters recommended by [29] to prevent the best-known attacks on the AGCD problem.

2 Preliminaries

2.1 Basic Symbols

If $k \in \mathbb{R}$, $\lfloor k \rfloor$ represents the integer closest to k , with a preference for rounding up in case of a tie. $x \leftarrow X$ indicates that x is selected randomly from the set X . In \mathbb{R}^n , we use the bold symbol \mathbf{a} to represent vectors, where $\mathbf{a} = (a_1, \dots, a_n)$ and each $a_i \in \mathbb{R}$. Given $x, p \in \mathbb{R}$, we denote $x \pmod{p}$ or $[x]_p$ as the remainder obtained after dividing x with p . (Note that $[x]_p \in (-\frac{p}{2}, \frac{p}{2}]$.) Let $n \in \mathbb{N}$ and $k \in \mathbb{R}$, define

$$\begin{aligned} \mathbf{BitDecom}_n(z) &= (z_0, z_1, \dots, z_{n-1}) \in \{0, 1\}^n \text{ with } z = \sum_{i=0}^{n-1} z_i 2^i. \\ \mathbf{Pow}_n(k) &= (k, 2k, \dots, 2^{n-1}k) \in \mathbb{R}^n. \end{aligned}$$

Consequently, it becomes evident that

$$\langle \mathbf{BitDecom}_n(z), \mathbf{Pow}_n(k) \rangle = \sum_{i=0}^{n-1} z_i (2^i k) = zk.$$

Now let $\mathbf{c} = (c_1, \dots, c_n)$ and $\mathbf{d} = (d_1, \dots, d_n)$ are vectors in \mathbb{R}^n . A coordinate-wise product is

$$\mathbf{c} \odot \mathbf{d} = (c_1 d_1, c_2 d_2, \dots, c_n d_n),$$

an inner product is given by

$$\langle \mathbf{c}, \mathbf{d} \rangle = \mathbf{c} \cdot \mathbf{d} = c_1 d_1 + c_2 d_2 + \dots + c_n d_n,$$

a tensor product is defined as

$$\mathbf{c} \otimes \mathbf{d} = (c_1 d_1, c_1 d_2, \dots, c_1 d_n, \dots, c_n d_1, \dots, c_n d_n),$$

and it satisfy the following

$$\langle \mathbf{c} \otimes \mathbf{c}', \mathbf{d} \otimes \mathbf{d}' \rangle = \langle \mathbf{c}, \mathbf{d} \rangle \cdot \langle \mathbf{c}', \mathbf{d}' \rangle.$$

2.2 AGCD Problem

If we have given a set $\{q_1k, q_2k, \dots, q_nk\}$ where $k, q_i \in \mathbb{Z}, \forall i \in \{1, 2, \dots, n\}$, then it is easy to find k just by taking GCD of all elements of the set. Now consider the case if we add some uniformly random error $r_i \in \mathbb{Z}$ in each the elements respectively, i.e., $\{q_1k + r_1, q_2k + r_2, \dots, q_nk + r_n\}$, in that scenario it's difficult to find k , and this problem is called AGCD problem.

λ : Security parameter.

η : Bit length of the secret key.

γ : Bit length of the AGCD sample.

ρ : Bit length of the noise parameter.

τ : Number of integers in the public key.

With these parameters, the AGCD problem is defined as below.

Definition 1 (AGCD Problem). For a secret key s_k of η -bit length, Define a distribution over γ -bit integers as:

$$\mathbb{D}_{\gamma, \rho}(s_k) = \left\{ \text{Choose } q \leftarrow \mathbb{Z} \cap \left[0, \frac{2^\gamma}{s_k} \right), r \leftarrow \mathbb{Z} \cap (-2^\rho, 2^\rho) : \text{Output} x = qs_k + r \right\}$$

AGCD problem states that given τ members of the distribution \mathbb{D} , can you find secret key s_k ?

Since AGCD problem is polynomial time reducible to the LWE problem [25], schemes based on the AGCD problem are considered quantum secure.

3 Fully Homomorphic Encryption Scheme Over the Integers

3.1 FHE Construction Model

FHE.KeyGen(1^λ): On input λ , generate secret key s_k , public key p_k , and evaluation key e_k .

FHE.Enc($p_k, m_i \in \{0, 1\}$): Encrypt the input data m_i using public key p_k to obtain ciphertext c_i .

FHE.Add(p_k, c_1, c_2): Given c_1, c_2 , get a ciphertext c_{add} which is encryption of the addition of input data, i.e., $m_1 + m_2$.

FHE.Mult(p_k, e_k, c_1, c_2): Given c_1, c_2 and evaluation key e_k , get a ciphertext c_{mult} which is encryption of multiplication of input data, i.e., $m_1 m_2$.

FHE.Dec(s_k, c): Given a ciphertext c , decrypt it using secret key s_k to obtain corresponding input data m .

3.2 MKFHE Construction Model

In the context of MKFHE, each party possesses their own set of keys (p_k, s_k, e_k). By utilizing the public keys from all parties, a combined public key is formed. Afterward, each party encrypts their individual input data using combined public key and transmits ciphertext to the cloud, along with the requested computation to be executed on the input data. The computation on the encrypted data is carried out by the cloud, resulting in a ciphertext that can only be decrypted using the combined secret key involving all participating parties. This ensures that no individual party or specific group of parties can decrypt the ciphertext provided by cloud. MKFHE model can be summarized as follows:

MKFHE.KeyGen(1^λ): On input λ , each party individually generates their set of keys, which consists of a secret key s_k , a public key p_k , and an evaluation key e_k . Utilizing these key pairs, the combined secret key s_{ck} , combined public key p_{ck} , and combined evaluation key e_{ck} are constructed.

MKFHE.Enc($p_{ck}, m_i \in \{0, 1\}$): Using the combined public key p_{ck} , each party encrypts the input data m_i and get the ciphertext c_i .

MKFHE.Add(p_{ck}, c_1, c_2): Given c_1, c_2 get a ciphertext c_{add} which is the encryption of addition of input data, i.e., $m_1 + m_2$.

MKFHE.Mult(p_{ck}, e_{ck}, c_1, c_2): Given c_1, c_2 and the combined evaluation key e_{ck} , get a ciphertext c_{mult} which is the encryption of multiplication of input data, i.e., $m_1 m_2$.

MKFHE.Dec(s_{ck}, c): Determine the input data m using the combined secret key s_{ck} and ciphertext c .

4 Proposed Scheme

4.1 Construction

Let's consider a scenario involving two parties, which can be easily extended to accommodate multiple parties. For the purpose of illustration, let's assume that Party P_1 and Party P_2 are the two parties involved, with their respective private input data denoted as m_1 and m_2 . They want to compute $f(m_1, m_2)$ without disclosing their

private input data. Here, we propose our MKFHE scheme over the integers.

MKFHE.KeyGen(1^λ) : On input λ , determine the parameters (η, ρ, γ) providing decryption correctness.

- Party P_1 selects a secret key p_1 that is a prime number with a bit length of η . Using the secret key p_1 , form a distribution $\mathbb{D}_{\gamma, \rho}(p_1)$. Public key of party P_1 will be τ uniformly random samples from the distribution $\mathbb{D}_{\gamma, \rho}(p_1)$ call them x_i , reorder such that x_0 is highest and $\left\lfloor \frac{x_1}{p_1} \right\rfloor$ is an odd number. If such x_1 is not found, choose samples again. Now for the evaluation key $\forall k \in \mathbb{Z} \cap [0, 2\gamma - 2]$, sample $q_{i,j}' \leftarrow \mathbb{Z} \cap \left[0, \frac{2^\gamma}{s_k} \right]$, $r_{i,j}' \leftarrow \mathbb{Z} \cap (-2^\rho, 2^\rho)$ and $e_k = \left(p_1 q_{i,j}' + r_{i,j}' \right)_{0 \leq i,j \leq \gamma} + \frac{p_1}{2} \left(\left[\mathbf{Pow}_\gamma(\frac{2}{p_1}) \right]_2 \otimes \left[\mathbf{Pow}_\gamma(\frac{2}{p_1}) \right]_2 \right)$ as a evaluation key for multiplication. Outputs of key generation algorithm for party P_1 are the secret key p_1 , the public key $\mathbf{pk}_1 = (x_0, x_1, \dots, x_\tau)$ and evaluation key \mathbf{ek}_1 .
- Party P_2 selects a secret key p_2 that is a prime number with a bit length of η . Using the secret key p_2 , form a distribution $\mathbb{D}_{\gamma, \rho}(p_2)$. Public key of party P_2 will be τ uniformly random samples from the distribution $\mathbb{D}_{\gamma, \rho}(p_2)$ call them y_i , reorder such that y_0 is highest and $\left\lfloor \frac{y_1}{p_2} \right\rfloor$ is an odd number. If such y_1 is not found, choose samples again. Now for the evaluation key for all $k \in \mathbb{Z} \cap [0, 2\gamma - 2]$, sample $q_{i,j}'' \leftarrow \mathbb{Z} \cap \left[0, \frac{2^\gamma}{s_k} \right]$, $r_{i,j}'' \leftarrow \mathbb{Z} \cap (-2^\rho, 2^\rho)$ and $e_k = \left(p_2 q_{i,j}'' + r_{i,j}'' \right)_{0 \leq i,j \leq \gamma} + \frac{p_2}{2} \left(\left[\mathbf{Pow}_\gamma(\frac{2}{p_2}) \right]_2 \otimes \left[\mathbf{Pow}_\gamma(\frac{2}{p_2}) \right]_2 \right)$ as a evaluation key for multiplication. Outputs of key generation algorithm for party P_2 are the secret key p_2 , the public key $\mathbf{pk}_2 = (y_0, y_1, \dots, y_\tau)$ and evaluation key \mathbf{ek}_2 .
- The combined key is generated by combining individual key of each party. The combined public key is given by $\mathbf{pk}_{ck} = (x_1, \dots, x_\tau) \otimes (y_1, \dots, y_\tau) = (z_{i,j})_{1 \leq i,j \leq \gamma}$, where $z_{i,j} = x_i y_j$ also let $z_{0,0} = x_0 y_0$ which is the highest, $\left\lfloor \frac{z_{1,1}}{p_1 p_2} \right\rfloor$ is an odd number. The combined evaluation key is $\mathbf{ek}_{ck} = \mathbf{ek}_1 \odot \mathbf{ek}_2$, the combined secret key is $s_{ck} = p_1 p_2$.

MKFHE.Enc($\mathbf{pk}_{ck}, m_i \in \{0, 1\}$) :

- Party 1 has an input data $m_1 \in \{0, 1\}$, randomly sample a set $S_1 \subseteq \{1, 2, \dots, \tau\} \times \{1, 2, \dots, \tau\}$, and compute

$$c_1 = \left[\sum_{(i,j) \in S_1} z_{i,j} + \left\lfloor \frac{z_{1,1}}{2} \right\rfloor m_1 \right]_{z_{0,0}},$$

which is the encryption of m_1 .

- Party 2 has an input data $m_2 \in \{0, 1\}$, randomly sample a set $S_2 \subseteq \{1, 2, \dots, \tau\} \times \{1, 2, \dots, \tau\}$, and compute

$$c_2 = \left[\sum_{(i,j) \in S} z_{i,j} + \left\lfloor \frac{z_{1,1}}{2} \right\rfloor m_2 \right]_{z_{0,0}},$$

which is the encryption of m_2 .

MKFHE.Add(p_{ck}, c_1, c_2): If c_1, c_2 are two ciphertexts then

$$c_{\text{add}} = [c_1 + c_2]_{z_{0,0}},$$

which is the encryption of addition of input data, i.e., $m_1 + m_2$.

MKFHE.Mult(p_{ck}, e_{ck}, c_1, c_2): If c_1, c_2 are two ciphertexts, then

$$c_{\text{mult}} := [\langle \text{BitDecom}_\gamma(c_1) \otimes \text{BitDecom}_\gamma(c_2), e_{ck} \rangle]_{z_{0,0}},$$

which is the encryption of multiplication of input data, i.e., $m_1 m_2$.

MKFHE.Dec(s_{ck}, c): If c is a ciphertext then $\left[\left\lfloor \frac{2c}{s_{ck}} \right\rfloor \right]_2 = m$.

4.2 Correctness

Lemma 1 (*Encryption noise*). Let $(s_{ck}, p_{ck}) \leftarrow \text{MKFHE.KeyGen}(1^\lambda)$ be a key pair, and let $c \leftarrow \text{MKFHE.Enc}(p_{ck}, m \in \{0, 1\})$ be a ciphertext. Then, we have

$$c^* = c \pmod{s_{ck}} = r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m \pmod{s_{ck}},$$

where r is the noise term satisfying $|r| \leq (2\tau^2 + \frac{1}{2})(2^\rho - 1) + \frac{1}{2}$.

Proof The combined public key is represented as $z_{i,j} = s_{ck}q_{i,j} + r_{i,j}$, where $q_{i,j} \in \mathbb{Z} \cap \left[0, \frac{2^\gamma}{s_{ck}}\right)$, $(i, j) \in S \subseteq \{1, 2, \dots, \tau\} \times \{1, 2, \dots, \tau\}$.

Then, we have $\left\lfloor \frac{z_{1,1}}{2} \right\rfloor = \frac{s_{ck}q_{1,1}}{2} + \frac{r_{1,1}}{2} + \xi$, where $|\xi| \leq 1/2$. So,

$$\begin{aligned} c^* &= c \pmod{s_{ck}} \\ &= \left(\sum_{(i,j) \in S} z_{i,j} + \left\lfloor \frac{z_{1,1}}{2} \right\rfloor m - kz_{0,0} \right) \pmod{s_{ck}} \\ &= \left(\sum_{(i,j) \in S} r_{i,j} + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m + \left(\frac{r_{1,1}}{2} + \xi \right) m - kr_{0,0} \right) \pmod{s_{ck}}, \end{aligned}$$

where $k \in [1, \tau^2]$, $|\xi| \leq 1/2$. Thus, the noise $|r| \leq (2\tau^2 + \frac{1}{2})(2^\rho - 1) + \frac{1}{2}$ for $c^* = c \pmod{s_{ck}} = r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m \pmod{s_{ck}}$.

Lemma 2 (*Addition noise*). Let $(s_{ck}, p_{ck}) \leftarrow \text{MKFHE.KeyGen}(1^\lambda)$ be a key pair, and let $c_i \leftarrow \text{MKFHE.Enc}(p_{ck}, m_i \in \{0, 1\})$.

If $c_{add} \leftarrow \text{MKFHE.Add}(p_{ck}, c_1, c_2)$, then

$$c_{add} = r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor (m_1 + m_2) \pmod{s_{ck}},$$

where $|r| \leq |r_1 + r_2| + 2^\rho$.

Proof Given $c_i^* = c_i \pmod{s_{ck}} = r_i + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m_i \pmod{s_{ck}}$, then

$$\begin{aligned} c_{add} &= c_1 + c_2 - \xi z_{0,0} \pmod{s_{ck}} \\ &= r_1 + r_2 - \xi r_{0,0} + \left\lfloor \frac{s_{ck}}{2} \right\rfloor (m_1 + m_2) + \xi' \pmod{s_{ck}}, \end{aligned}$$

where $|\xi| \leq 1$, $|\xi'| \leq 1$. So, the noise $|r| \leq |r_1 + r_2| + 2^\rho$ for $c_{add} = r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor (m_1 + m_2) \pmod{s_{ck}}$.

Lemma 3 If $c = s_{ck}q + r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m \in \mathbb{Z} \cap [0, 2^\gamma]$ with $m \in \{0, 1\}$ and $q, r \in \mathbb{Z}$, then

$$\left\langle \text{BitDecom}_\gamma(c), \left[\text{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \right\rangle = 2a + m + \delta,$$

for $a \in \mathbb{Z}$ satisfying $|a| \leq \frac{\gamma}{2} - \eta + 2$ and $\delta \in \mathbb{R}$ satisfying $|\delta| < \frac{(2|r|+1)}{s_{ck}}$.

Proof Given $\left\lfloor \frac{s_{ck}}{2} \right\rfloor = \frac{s_{ck}+d}{2}$, where $d \in \{0, 1\}$. So, $\frac{2c}{s_{ck}} = 2q + m + \delta = m + \delta \pmod{2}$ for $\delta = \frac{(2r+d)}{s_{ck}}$ satisfying $|\delta| \leq \frac{(2|r|+1)}{s_{ck}}$. Now Consider,

$$\left\langle \text{BitDecom}_\gamma(c), \left[\text{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \right\rangle \equiv \left\langle \text{BitDecom}_\gamma(c), \text{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right\rangle = \frac{2c}{s_{ck}}$$

So, $\left\langle \text{BitDecom}_\gamma(c), \left[\text{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \right\rangle = 2a + m + \delta$ for $a \in \mathbb{Z}$.

Using $\frac{2}{s_{ck}} + \frac{2^2}{s_{ck}} + \dots + \frac{2^{2\eta-2}}{s_{ck}} = \frac{2(2^{2\eta-2}-1)}{s_{ck}} < 1$,

$$\left| \left\langle \text{BitDecom}_\gamma(c), \left[\text{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \right\rangle \right| \leq \sum_{i=0}^{\gamma-1} \left| \left[\frac{2^{i+1}}{s_{ck}} \right]_2 \right| \leq \gamma - 2\eta + 3$$

implies $|a| \leq \frac{\gamma}{2} + \eta + 2$.

Lemma 4 (*Multiplication noise*). Let $(s_{ck}, p_{ck}, e_{ck}) \leftarrow \text{MKFHE.KeyGen}(1^\lambda)$. Given $c_1, c_2 \in \mathbb{Z} \cap \left(-\frac{z_{0,0}}{2}, \frac{z_{0,0}}{2}\right]$ with $c_i = r_i + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m_i \pmod{s_{ck}}$ for $i \in \{0, 1\}$, then we have

$$c_{mult} = r_{mult} + \left\lfloor \frac{s_{ck}}{2} \right\rfloor (m_1 m_2) \pmod{s_{ck}}$$

where $r \in \mathbb{Z}$ satisfies $|r| < \gamma^2 2^{\rho+1} + (\gamma - 2\eta + 6)(|r_1| + |r_2|)$.

Proof We have

$$\mathbf{e}_{ck} = \left(s_{ck} q'_{i,j} + r''_{i,j} \right)_{0 \leq i,j < \gamma} + \frac{s_{ck}}{2} \left(\left[\mathbf{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \otimes \left[\mathbf{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \right),$$

for some $r''_{i,j} \in r'_{i,j} + [-\frac{1}{2}, \frac{1}{2}]$ for all i and j . By Lemma 3,

$$\begin{aligned} c_{\text{mult}} &= \langle \mathbf{BitDecom}_\gamma(c_1) \otimes \mathbf{BitDecom}_\gamma(c_2), \mathbf{e}_{ck} \rangle \\ &= \left\langle \mathbf{BitDecom}_\gamma(c_1) \otimes \mathbf{BitDecom}_\gamma(c_2), \left(s_{ck} q'_{i,j} + r''_{i,j} \right)_{i,j} \right\rangle \\ &\quad + \frac{s_{ck}}{2} \left\langle \mathbf{BitDecom}_\gamma(c_1), \left[\mathbf{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \right\rangle \\ &\quad \left\langle \mathbf{BitDecom}_\gamma(c_2), \left[\mathbf{Pow}_\gamma \left(\frac{2}{s_{ck}} \right) \right]_2 \right\rangle \\ &= \sum_{(i,j) \in T} \left(s_{ck} q'_{i,j} + r''_{i,j} \right) + \frac{s_{ck}}{2} (m_1 + \delta_1 + 2a_1)(m_2 + \delta_2 + 2a_2) \end{aligned}$$

for a set $T \subseteq [0, \gamma]^2$, $a_1, a_2 \in \mathbb{Z}$, and $\delta_1, \delta_2 \in \mathbb{R}$ with $|a_i| \leq \frac{\gamma}{2} - \eta + 2$, and $|\delta_i| < \frac{(2|r_i|+1)}{s_{ck}}$ for $i = 1, 2$. Since $\frac{s_{ck}}{2} ((m_1 + 2a_1)(m_2 + 2a_2) - m_1 m_2)$ is a multiple of s_{ck} , there exists $q \in \mathbb{Z}$ such that

$$\left[\langle \mathbf{BitDecom}_\gamma(c_1) \otimes \mathbf{BitDecom}_\gamma(c_2), \mathbf{e}_{ck} \rangle \right]_{z_{0,0}} = s_{ck}q + r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m_1 m_2,$$

$$c_{\text{mult}} = r_{\text{mult}} + \left\lfloor \frac{s_{ck}}{2} \right\rfloor (m_1 m_2) \pmod{s_{ck}}$$

where

$$r = \sum_{(i,j) \in T} r''_{i,j} + \frac{s_{ck}}{2} (\delta_2(m_1 + 2a_1) + \delta_1(m_2 + 2a_2) + \delta_1 \delta_2) - \frac{1}{2} m_1 m_2 - k r_{0,0}$$

for some $k \in [0, \gamma^2]$. So $|r| < \gamma^2 2^{\rho+1} + (\gamma - 2\eta + 6)(|r_1| + |r_2|)$.

Lemma 5 (Decryption Noise). Let $s_{ck} \in \mathbb{N}$, $m \in \{0, 1\}$, and $c \in \mathbb{Z}$. Then we have $\text{MKFHE.Dec}(s_{ck}, c) = m$ if $c = r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m \pmod{s_{ck}}$ when $|r| < \frac{s_{ck}}{4} - \frac{1}{2}$.

Proof We can express $c = s_{ck}q + r + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m$. Now consider for $b \in \{0, 1\}$,

$$\left\lfloor \frac{2c}{s_{ck}} \right\rfloor = \left\lfloor 2q + m + \frac{2r + b}{s_{ck}} \right\rfloor = 2q + m + \left\lfloor \frac{2r + b}{s_{ck}} \right\rfloor = m \pmod{2},$$

when $|r| < \frac{s_{ck}}{4} - \frac{1}{2}$.

To summarize the results from all previous lemmas, we define the encryption process as follows:

Let $c_i \leftarrow \mathbf{MKFHE}.\mathbf{Enc}(p_{ck}, m_i \in \{0, 1\})$, where $c_i = r_i + \left\lfloor \frac{s_{ck}}{2} \right\rfloor m_i \pmod{s_{ck}}$ for $i \in \{1, 2\}$. Additionally, we have $c_{\text{add}} \leftarrow \mathbf{MKFHE}.\mathbf{Add}(p_{ck}, c_1, c_2)$ and $c_{\text{mult}} \leftarrow \mathbf{MKFHE}.\mathbf{Mult}(p_{ck}, e_{ck}, c_1, c_2)$. Alternatively, we can express it as:

$$\begin{aligned} c_{\text{add}} &= r_{\text{add}} + \left\lfloor \frac{s_{ck}}{2} \right\rfloor (m_1 + m_2) \pmod{s_{ck}} \\ c_{\text{mult}} &= r_{\text{mult}} + \left\lfloor \frac{s_{ck}}{2} \right\rfloor (m_1 m_2) \pmod{s_{ck}} \end{aligned}$$

These equations satisfy the conditions $|r_{\text{add}}| \leq |r_1| + |r_2| + 2^\rho$ and $|r_{\text{mult}}| < \gamma^2 2^{\rho+1} + (\gamma - 2\eta + 6)(|r_1| + |r_2|)$. This demonstrates that the noise grows additively during addition and multiplication operations. By applying the “bootstrapping” technique similar to [25], we can obtain an MKFHE scheme.

4.3 Security and Parameters

Our proposed scheme achieves security by leveraging the assumption of hardness of AGCD problem. When appropriate parameters are chosen, the AGCD problem can be reduced to the LWE problem in polynomial time, as shown in prior work [25]. This reduction ensures the scheme’s resistance against quantum attacks.

In the context of a two-party scenario, we construct our secret key by multiplying two prime numbers, each having a bit length of η . As a result, the bit length of our secret key is 2η , and the total number of AGCD samples is τ^2 . We denote L as the multiplicative depth required for computing the desired function. To make the depth of decryption function less than that of permeated function we let $\eta \geq \rho + O(L \log \lambda)$. We let $\rho = 2\eta - L \log \lambda$ to reduce from AGCD problem to LWE problem, and to prevent various noted lattice-based attacks we consider $\gamma \geq \frac{\lambda}{\log \lambda} (2\eta - \rho)^2 + \rho$. To incorporate the leftover hash lemma, as demonstrated in [9], we suppose $\tau = \gamma + \Omega(\lambda)$.

The aforementioned conditions can be combined and expressed in terms of security parameter λ as follows:

$$\gamma = \Omega(L^2 \lambda \log \lambda), \quad \eta = \gamma - \lambda, \quad \rho = 2\eta - L \log \lambda, \quad \tau = \gamma + \Omega(\lambda)$$

4.4 Extension to N Party Case

Now, let us consider a scenario where N parties aim to perform computations on their individual private input data. To achieve this, each party P_i follows the aforementioned KeyGen algorithm, resulting in the acquisition of a secret key s_{ki} , a public key p_{ki} , and an evaluation key e_{ki} . By utilizing these keys, a combined secret key s_{ck} , a combined public key p_{ck} , and a combined evaluation key e_{ck} are generated. Using combined public key p_{ck} each party P_i encrypt its input data and release ciphertext. The cloud performs the required computation on ciphertext by using combined evaluation key. Finally, after getting encrypted output from the cloud, all N parties can decrypt it securely with the combined secret key s_{ck} .

4.5 Extension of Private Input Data to \mathbb{Z}_g

In the above proposed scheme we have taken input data space of each party as \mathbb{Z}_2 . Just with little modification, the input data space can be changed to \mathbb{Z}_g for some large g as required provided $\gcd(p_1, g) = 1 = \gcd(p_2, g)$. In the encryption algorithm replace 2 by g everywhere. So our extended Scheme has encryption as $c = \left[\sum_{(i,j) \in S} z_{i,j} + \left\lfloor \frac{z_{1,1}}{g} \right\rfloor m \right]_{z_{0,0}}$. Note that g should be publicly known to perform the encryption. Similarly replace 2 by g in decryption that is $m = \left[\left\lfloor \frac{gc}{s_{ck}} \right\rfloor \right]_g$.

5 Conclusion

In this paper, we have introduced an novel MKFHE scheme over the integers. Our scheme is secure under the hardness of AGCD problem. Additionally, we have demonstrated that noise term in the ciphertext gradually increases in an additive manner after each homomorphic addition or multiplication operation.

References

1. Rivest R, Adleman L, Dertouzos M (1978) On data banks and privacy homomorphisms. Found Secure Comput 4(11):169–180
2. ElGamal T (1984) A public key cryptosystem and a signature scheme based on discrete logarithms. In: Advances in Cryptology CRYPTO'84, vol 196. LNCS. Springer, Heidelberg, pp 10–18
3. Goldwasser S, Micali S (1984) Probabilistic encryption. J Comput Syst Sci 28(2):270–299
4. Paillier P (1999) Public-key cryptosystems based on composite degree residuosity classes. In: Advances in cryptology EUROCRYPT'99, vol 1592. LNCS. Springer, Heidelberg, pp 223–238

5. Boneh D, Goh EJ, Nissim K (2005) Evaluating 2-DNF formulas on ciphertexts. In: Theory of Cryptography—TCC'05. LNCS, vol 3378. Springer, pp 325–341
6. Gentry C (2009) Fully homomorphic encryption using ideal lattices. In: Proceedings of the ACM symposium on theory of computing
7. Gentry C (2009) A fully homomorphic encryption scheme. Ph.D. thesis, Stanford University
8. Howgrave-Graham N (2009) Approximate integer common divisors. In: Proceedings of the international symposium on computer algebra and symbolic computation (CaLC). LNCS, vol 2146. Springer, pp 51–66
9. van Dijk M, Gentry C, Halevi S, Vaikuntanathan V (2010) Fully homomorphic encryption over the integers. In: Advances in cryptology EUROCRYPT'10. LNCS, vol 6110, pp 24–43
10. Regev O (2005) On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the 37th ACM symposium on theory of computing (STOC), pp 84–93
11. Lyubashevsky V, Peikert C, Regev O (2013) On ideal lattices and learning with errors over rings. *J ACM (JACM)* 60(6):1–35
12. Brakerski Z, Vaikuntanathan V (2011) Efficient fully homomorphic encryption from (standard) LWE. In: Proceedings of the annual IEEE symposium on foundations of computer science (FOCS), pp 97–106
13. Brakerski Z, Vaikuntanathan V (2011) Fully homomorphic encryption from ring-LWE and security for KDM. In: Advances in Cryptology, CRYPTO'11, pp 505–524
14. Brakerski Z, Gentry C, Vaikuntanathan V (2012) (Leveled) fully homomorphic encryption without bootstrapping. In: Proceedings of the 44th symposium on theory of computing conference (STOC'12), pp 309–325
15. Brakerski Z (2012) Fully homomorphic encryption without modulus switching from classical GapSVP. In: Advances in cryptology CRYPTO'12 proceedings. Springer Berlin Heidelberg, pp 868–886
16. Gentry C, Sahai A, Waters B (2013) Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based. In: Advances in Cryptology CRYPTO'13, pp 75–92
17. Chillotti I, Gama N, Georgieva M, Izabachene M (2016) Faster fully homomorphic encryption: bootstrapping in less than 0.1 seconds. In: Advances in cryptology ASIACRYPT'16 proceedings, Part I 22. Springer Berlin Heidelberg, pp 3–33
18. Chen H, Chillotti I, Song Y (2019) Multi-key homomorphic encryption from TFHE. In: Proceedings of the ACM conference on computer and communications security (CCS), pp 446–472
19. Chillotti I, Gama N, Georgieva M, Izabachene M (2020) TFHE: fast fully homomorphic encryption over the torus. *J Cryptol* 33(1):34–91
20. López-Alt A, Tromer E, Vaikuntanathan V (2012) On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In: Proceedings of the 44th symposium on theory of computing conference (STOC'12), pp 1219–1234
21. Coron JS, Naccache D, Nitaj A, Tromer E (2011) Public key compression and modulus switching for fully homomorphic encryption over the integers. In: Advances in cryptology CRYPTO'11, LNCS, vol 6841, pp 446–464
22. Cheon JH, Kim J, Lee MS, Yun A (2015) CRT-based fully homomorphic encryption over the integers. *Inf Sci* 310:149–162
23. Coron JS, Lepoint T, Tibouchi M (2014) Scale-invariant fully homomorphic encryption over the integers. In: Public-key cryptography PKC'14: proceedings 17. Springer Berlin Heidelberg, pp 311–328
24. Nuida K, Kurosawa K (2015) (Batch) fully homomorphic encryption over integers for non-binary message spaces. In: Advances in cryptology EUROCRYPT'15. Springer, pp 354–383
25. Cheon JH, Stehlé D (2015) Fully homomorphic encryption over the integers revisited. In: Advances in cryptology EUROCRYPT'15, proceedings, Part I. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 513–536
26. Benaroch D, Brakerski Z, Lepoint T (2017) FHE over the integers: decomposed and batched in the post-quantum regime. In: Public-key cryptography PKC'17, pp 445–475. Springer International Publishing

27. Kim E, Tibouchi M (2018) FHE over the integers and modular arithmetic circuits. In: Cryptology and network security, pp 497–515. Springer International Publishing
28. Pereira H (2021) Vitor: bootstrapping fully homomorphic encryption over the integers in less than one second. Public-key cryptography PKC’21. LNCS, vol 12710. Springer, Cham, pp 325–357
29. Jun X, Sarkar S, Hu L (2018) Revisiting orthogonal lattice attacks on ACD problems. *Theor Comput Sci*, 55–69
30. Clear M, McGoldrick C (2015) Multi-identity and multi-key leveled FHE from learning with errors. In: Advances in cryptology CRYPTO’15 proceedings, Part II 35. Springer Berlin Heidelberg, pp 630–656
31. Mukherjee P, Wichs D (2016) Two round multiparty computation via multi-key FHE. In: Advances in cryptology EUROCRYPT’16: proceedings, Part II 35. Springer Berlin Heidelberg, pp 735–763
32. Brakerski Z, Renen P (2016) Lattice-based fully dynamic multi-key FHE with short ciphertexts. In: Advances in cryptology CRYPTO’16 proceedings, Part I. Springer Berlin Heidelberg, Berlin, Heidelberg
33. Chen L, Zhang Z, Wang X (2017) Batched multi-hop multikey FHE from ring-LWE with compact ciphertext extension. In: Proceedings of the theory of cryptography conference (TCC). LNCS, vol 10678, pp 597–627
34. Kim E, Lee HS, Park J (2018) Towards round-optimal secure multiparty computations: multikey FHE without a CRS. In: Australasian conference on information security and privacy (ACISP). LNCS, 10946, pp 101–113
35. Shen T, Wang F, Chen K, Wang K, Li B (2019) Efficient leveled (multi) identity-based fully homomorphic encryption schemes. *IEEE Access* 7:299–310

A Modest Approach Toward Cloud Security Hygiene



Sujal Patel , Rashmi Agarwal , Shinu Abhi , and Ratan Jyoti

Abstract Cloud computing offers an on-demand process and computing service without direct active management by the user. It can reach thousands of different networks in a day which makes it extremely powerful and challenging to secure. Many cloud customers are facing issues matching external security mandates based on their business domain. These mandates are prepared with reference to industry standard frameworks. Organizations can determine the security posture of their cloud by following these standards. Security measures like visibility across the cloud environment, misconfiguration management, and compliance can strengthen the security posture of any cloud infrastructure. Cloud Service Providers (CSPs) deliver expensive cloud-native security services which can be achieved freely using a cloud Software Developers Kit (SDK). The research emphasizes the gap observed in the market by developing the application using SDKs which provides insights into cloud security, misconfigurations, and compliance. Gartner stated that over-provisioned access to cloud resources can lead to security breaches and data leakages. The authors have taken the Identity and Access Management (IAM) policy parameter as an example to prepare the research model as it is one of the most important ones among various cloud security parameters like firewall policies, encryption, etc. The purpose of this study is to find an economical way to access cloud security posture and compliance management for small and mid-scale companies with limited budgets. The proposed method uses APIs and SDKs to fetch the relevant information from the cloud environment to access and secure the organization's cloud environment in a cost-effective way.

Keywords AWS · GCP · Azure · Security Management as a Service · Cloud compliance · ISO · NIST · PCI DSS · Cloud security posture management · Cloud security · CSPM

S. Patel · R. Agarwal · S. Abhi

REVA Academy for Corporate Excellence, REVA University, Bengaluru, India
e-mail: sujal.cs01@reva.edu.in

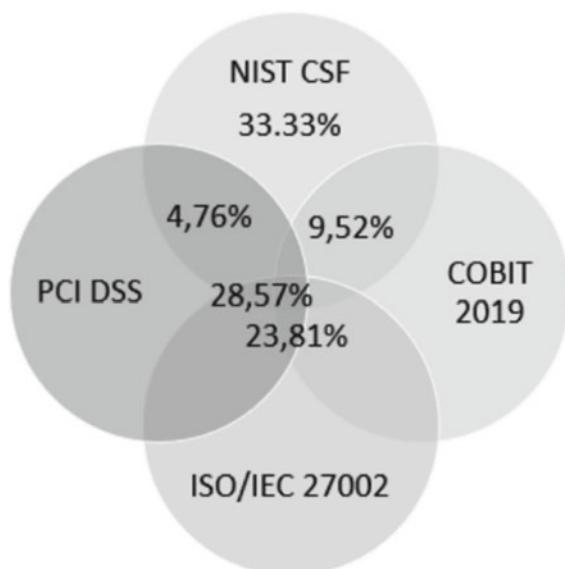
R. Jyoti
Ujjivan Small Finance Bank, Bengaluru, India

1 Introduction

Cloud Service Providers provide Service Level Agreements (SLAs) to the consumers based on the services they rendered. Certain responsibilities for data security are often shared between the Cloud Service Provider and their clients as per the defined SLAs. Cloud customers face arduous challenges matching their external compliance based on their business mandates. These mandates are prepared based on recommended industry standards like the International Organization for Standardization (ISO), National Institute of Standards and Technology (NIST), and Payment Card Industry Data Security Standards (PCI DSS). These frameworks offer a starting point for establishing security policies, processes, and administrative activities for IT security [1]. Multiple frameworks or models collectively cover the complete scope of IT Governance. For example, the information security policy defined in ‘Section 5’ of ISO 27002 and the ‘control objectives’ of PCI DSS relates to ‘maintaining an information security policy’ in the Control Objectives for Information Technologies (COBIT) framework [2]. Figure 1 depicts the relationship between different frameworks. The organizations do not blindly follow these guidelines, instead, they take influence from these frameworks and design their own security guidelines as per the business requirement.

As per the top research institutes, unauthorized user access is one of the top three cloud security issues. IAM service is responsible to manage, maintain, and control user access to the cloud resources. Granting the appropriate user access is essential; because it can significantly reduce the attack surface area. Therefore, monitoring, managing, and controlling user identity plays a crucial role in maintaining the security

Fig. 1 Framework relationship [2]



posture of any cloud environment. Considering the importance of IAM in cloud security, the authors have taken this service to demonstrate the efficiency of their prototype.

IAM and other managed services like auditing and accounting are made available to cloud customers through the Security Management as a Service (SMaS) model [3, 4]. These offerings have exorbitant charges and have limited visibility of their environment. Due to this, organizations using a multi-cloud environment or limited budgets are facing difficulties in managing their cloud infrastructure. The developed prototype can address all of these gaps. Auditors and cloud administrators both can leverage this prototype to their advantage and make their day-to-day job effortless and more efficient. The research made earlier on cloud security focuses on security measures, but was not cost-effective [3]. The focus of this study is to deliver the prototype to address these security challenges in budget. The solution is to prepare a lightweight application using freely available Application Programming Interfaces (APIs) and SDKs to manage and monitor their cloud environment [4, 5].

2 Literature Review

The cloud service could be comprised of a variety of services from several vendors which are physically hosted in Data Centers (DC) across different geographies. The model mentioned in Table 1 depicts the customer's responsibility to implement their controls and visibility over their cloud tenant based on the opted service model [6]. The assertion made by the authors of *Security compliance auditing of Identity and access management in the Cloud* [7] is noteworthy. By following their suggestions, numerous businesses have reduced their risk through structural risk management on their cloud workloads.

The authors state that cloud customers must persistently follow a variety of IT security control which includes external and internal audit requirements [8]. An effective effort should be made to address these requirements by taking a uniform and strategic approach to increase security and compliance.

The suggested solutions were implemented by CSPs/vendors and sold to the customer at a premium price. These cloud-native solutions are provided as a service by CSPs to their customers to ensure the security posture of their cloud environment. Small and mid-scale companies with a limited budget have to face a lot of challenges to maintain their security posture. Cloud administrators have to switch between multiple tools and dashboards to keep their multi-cloud environment secure.

This prototype enables organizations to have cloud infrastructure with the right security and compliance visibility in a single pane of glass at an economical cost. The below objectives can be achieved to secure the organization's cloud infrastructure. **Visibility Across Cloud Environment:** Companies migrating to the cloud or planning to build a new hybrid cloud infrastructure require to have complete visibility across their cloud environment. **Manage Misconfiguration and Remediation:** A lot

Table 1 CCSP cloud responsibility [8]

	<i>Infrastructure as a Service (IAAS)</i>	<i>Platform as a Service (PAAS)</i>	<i>Software as a Service (SAAS)</i>
<i>Security Governance, Risk, and Compliance.</i>			
<i>Data Security</i>			
<i>Application Security</i>			Yellow
<i>Platform Security</i>		Yellow	Light Blue
<i>Infrastructure Security</i>	Yellow		
<i>Physical Security</i>	Light Green		Light Green

	<i>Enterprise Responsibility</i>
Yellow	<i>Shared Responsibility</i>
Light Green	<i>CSP's Responsibility</i>

of companies lose trillions of dollars because of a common reason like misconfiguration. They not only incur financial losses, but these companies lose their credibility as well. **Compliance and Security Best Practices:** Data-sensitive industries like health care, finance, and insurance have to follow a set of rules to stay compliant [9].

Visibility into the environment's actual configurations against the known standard framework (NIST, ISO, and PCI DSS) [10–12] adds a lot of value to these businesses. The prototype is built to eliminate the use of expensive cloud-native services and to have a consolidated view across all multi-cloud environments. Application flags users with full access because that could be abused [13]. This application can facilitate organizations with security findings based on recommended best practices, to ensure the cloud environment have the right security posture. This prototype also simplifies the auditor's work by providing real-time compliance-based violations against renowned security frameworks using just the *access key* and *secret* of the cloud environment. This saves them a lot of time during audits; also, they do not have to rely on data provided by cloud administrators for better transparency.

The researchers have made their research by focusing on an affordable approach to secure and access the cloud security parameters against the best suitable framework at

a nominal expense. Many small-scale and mid-scale companies do not have enough budget to adopt such exorbitant security services and offerings. It focuses to facilitate these small and mid-segment cloud adaptors to address these security challenges in budget.

3 Methodology

The developed prototype ensures the visibility of the cloud environments by using the freely available SDKs and APIs. Figure 2 describes the process flow used to prepare this prototype. The developed prototype code is available on the [https://git hub.com/Sujal200191/RaceProject.git](https://github.com/Sujal200191/RaceProject.git) repository.

The prototype is developed in three major parts which comprise (1) cloud preparation, (2) application preparation, and (3) preparation of the hosting platform. These are explained in more detail below.

- Cloud preparation:** The important part is to find the right SDKs to fetch the required information from the cloud environment.

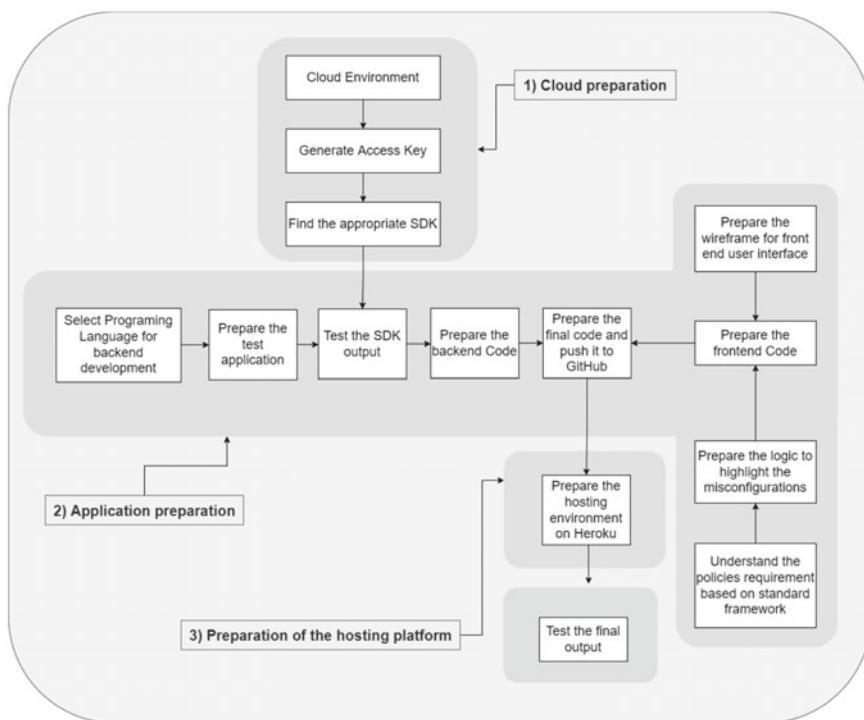


Fig. 2 Proposed methodology

- (i) Create the access credentials which have required access to the cloud environment to fetch the data.
 - (ii) The second requirement is to find the list of required SDKs to be used in the prototype. These SDKs should be able to fetch the required details.
2. **Application preparation:** Prepare the front-end and back-end application code and test the final output.
- (i) The first thing is to select the coding language (*Node.js*, *java*, etc.) to design the prototype.
 - (ii) Prepare test code to test the listed SDKs to ensure that desired data is getting fetched on the back-end application.
 - (iii) Prepare the wireframe for the front-end User Interface (UI) based on the requirement.
 - (iv) Fetch the compliance guidelines for the desired data as per ISO, NIST, PCI DSS, etc.
 - (v) Prepare the front-end logic to highlight the violations based on the recommended frameworks like PCI, NIST, and OSI.
 - (vi) Prepare the front-end code on front-end applications like *Create-React-App* based on the designed wireframes.
 - (vii) Prepare the final code and test the output.
 - (viii) Create the GitHub account and push the final code to the repository.
3. **Preparation of hosting platform:** The application can be hosted on any server hosted on-premise, cloud, or hosting services. Follow the below steps when using hosting services from third-party service providers like *Heroku*.
- (i) Create an account on *Heroku*.
 - (ii) Create the project and install the back-end dependencies.
 - (iii) Link the GitHub repository to *Heroku*.
 - (iv) Stage the final application.

4 Software Design

The materials used to create this prototype are mentioned below:

1. **Front-end code:** The front-end code was written in Create react app.
2. **Back-end code:** The back-end code was written on Node.js.
3. **Application hosting:** The application was hosted on the Heroku application hosting platform.

The software design is divided into two parts High-Level Design (HLD) and a Low-Level Flowchart (LLF). The HLD states the major components used in the application, and the LLF states how traffic flows between each component.

High-Level Design: The following components are involved in the application flow. Figure 3 explains the high-level working of the application.

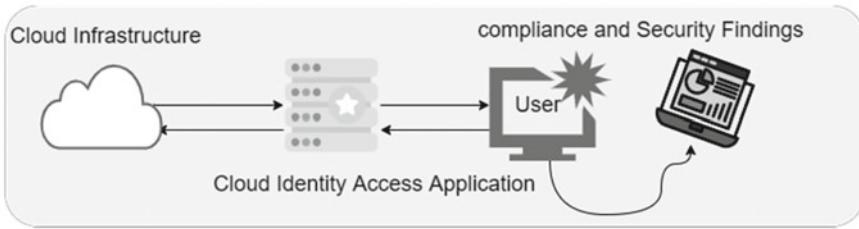


Fig. 3 High-level design

1. User requests the information from the application server.
2. Application requests API calls to the cloud.
3. Cloud responds with the required data.
4. Application filters the data and presents it to the user.

Low-Level Application Request Flow: The following flow explains the internal calls made in the different internal components of the application. Figure 4 depicts the application request and response flow.

1. User requests on front-end applications like *Create-Reach-App*.
2. Front-end application makes the API call to the back-end (e.g., *Node.js Server*) application server based on the user inputs/actions.
3. Back-end application server further makes the SDK request to the cloud using *Access ID* and *Secret Access Key*.
4. Cloud authenticates the request generated by the back-end application server and shares the required information.
5. Back-end server delivers information like the user, user groups, password policy, etc., to the front-end application.
6. Front-end application filters required output based on the scripted security and compliance logic.

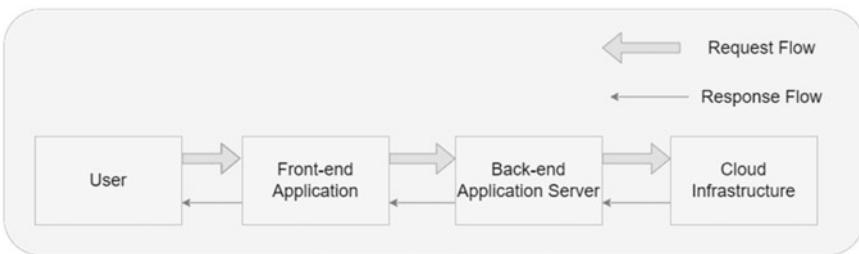


Fig. 4 Low-level flow

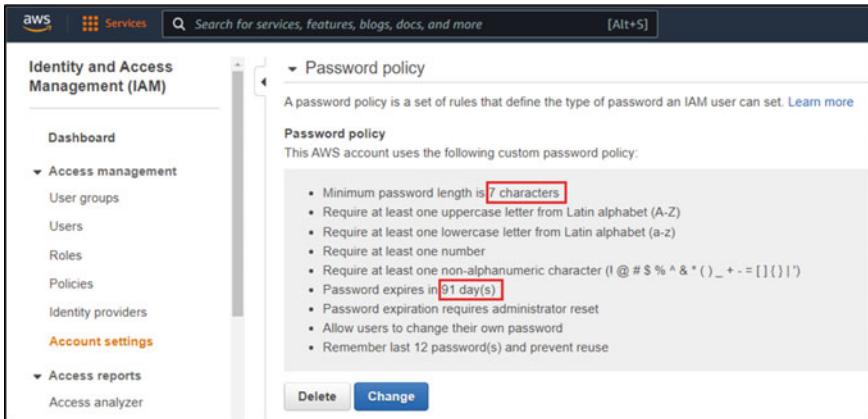


Fig. 5 Password policy as per AWS

5 Implementation Results and Analysis

This prototype aims to fetch IAM security-related findings on cloud environments based on renowned compliance frameworks like ISO, NIST, and PCI DSS. The results state the accurate violations as per these frameworks mentioned below. It also suggests recommendations as per these standards' best practices.

Password Policy Configured on AWS IAM console: The password policy configured on AWS states that the minimum password length is 7, the password complexity is enabled, and the password expiration is 91 days as seen in Fig. 5.

ISO (27001) standards: As per ISO Standards password policy guidelines the *minimum password length* and *max password age* are getting violated. The minimum password length should be 8 or more, and the password age should be 90 or lower. As mentioned in Fig. 6, the output of the prototype clearly states these ISO password policy violations.

NIST (SP 800-63B) standards: As per NIST Standards only *minimum password length* getting violated, unlike ISO Standards. The output of the prototype mentioned in Fig. 7 states only the *minimum password length* violation as per NIST password policy best practices.

PCI DSS Standards: Similar to the above two findings, the prototype has highlighted the violations as per PCI DSS's password *maximum password age* recommendation as mentioned in Fig. 8.

Password Policy Analysis				
Multi Factor Authentication (MFA) Analysis		ISO	NIST	PCI DSS
Total users:	17			
MFA Enabled	2			
15 of 17 users have MFA Disabled				
Critical Users				
<ul style="list-style-type: none"> Rutul Sujal 				
2 of 17 users have critical access				
Password Policy		Actual	Recommended	
Can User Change password		YES	YES	
Minimum Password Length		7	>7	
Required Symbol		YES	YES	
Require Numbers		YES	YES	
Require Uppercase		YES	YES	
Require Lowercase		YES	YES	
Expire Password		YES	YES	
Max Password Age		91	<91	
Password Reuse Prevention		12	Not Mandatory	

Fig. 6 Password violations as per ISO

Password Policy Analysis				
Multi Factor Authentication (MFA) Analysis		ISO	NIST	PCI DSS
Total users:	17	ISO	Actual	Recommended
MFA Enabled	2	Can User Change password	YES	YES
		Minimum Password Length	7	>7
		Required Symbol	YES	Not Mandatory
		Require Numbers	YES	Not Mandatory
		Require Uppercase	YES	Not Mandatory
		Require Lowercase	YES	Not Mandatory
		Expire Password	YES	Not Mandatory
		Max Password Age	91	Not Mandatory
		Password Reuse Prevention	12	Not Mandatory
15 of 17 users have MFA Disabled				
Critical Users				
<ul style="list-style-type: none"> Rutul Sujal 				
2 of 17 users have critical access				

Fig. 7 Password violations as per NIST

Password Policy Analysis			
Multi Factor Authentication (MFA) Analysis		ISO	NIST
		PCI DSS	
Total users:	17	Actual	Recommended
MFA Enabled	2	YES	YES
15 of 17 users have MFA Disabled		7	>6
Critical Users		YES	YES
<ul style="list-style-type: none"> Rutul Sujal 		YES	YES
2 of 17 users have critical access		YES	YES
		91	<91
Max Password Age		12	Not Mandatory
Password Reuse Prevention			

Fig. 8 Password violations as per PCI DSS

6 Conclusion

The output and results obtained using this application show that this approach is quite effective in identifying security issues based on well-known security framework standards without incurring any additional costs for the firm. Not only it detects these security findings, but it also gives recommendations to fix these findings. This prototype can be very useful to cloud administrators, auditors, small-scale businesses, and organizations having multi-cloud environments. This prototype reduces an organization's security expense by eliminating the use of expensive native cloud security tools. These can add a lot of value to small and mid-scale businesses which are security savvy but have literally no budget to spend on security. Auditors can also leverage these prototypes to access the security findings of any environment based on well-known standards using just the *API key* and *secret* of that cloud environment. This can save a lot of their time during audits. These can save them from a lot of ambiguity as they can rely on real-time data instead of data/evidence provided by cloud administrators. Cloud admins can leverage this prototype for quick security assessment of their current cloud security posture as per the security best practices and fix the findings as per the recommendations.

Further advancements like cloud control parameters and the addition of custom security parameters can be included to make this application more efficient. It includes custom security parameters, through which prototype users can design their own security baselines that comply with internal and external governance, risk, and

compliance requirement. Audit trail SDKs can also be used to track the activities and changes happening on the cloud infrastructure. Controls to manage the cloud infrastructure can also be made available on a prototype by using ‘create’ and ‘delete’ SDKs.

Acknowledgements The authors want to convey their sincere gratitude to mentors **Sandeep Vijayaraghavan** and **Dhruv Kalaal** for their guidance, motivation, and helpful advice during the research. Additionally, they want to express their gratitude to **Rutul Amin** who is working as a software engineer at Infosys for his ongoing assistance in creating this prototype.

References

1. Bailey E, Becker JD (2014) A comparison of IT governance and control frameworks in cloud computing. In: 20th Americas conference on information systems, pp 1–10
2. Sulistyowati D, Handayani F, Suryanto Y (2020) Análisis comparativo y diseño de madurez en ciberseguridad Metodología de evaluación utilizando NIST CSF, COBIT, ISO/IEC 27002 y PCI DSS. *Int J Inf Vis* 4(4):225–230
3. Krishnan D, Chatterjee M (2012) Cloud security management suite—security as a service. In: Proceedings of the 2012 World Congress on Information and Communication Technologies, WICT 2012, pp 431–436. <https://doi.org/10.1109/WICT.2012.6409116>
4. Class: AWS.IAM—AWS SDK for JavaScript
5. Googleapis (2022) google-api-nodejs-client: Google’s officially supported Node.js client library for accessing Google APIs. Support for authorization and authentication with OAuth 2.0, API Keys and JWT (Service Tokens) is included, GitHub (Online). Available: <https://github.com/googleapis/google-api-nodejs-client>. Accessed 14 Aug 2022
6. Goulding JT (2011) Identity and access management for the cloud: CA technologies strategy and vision WHITE PAPER Cloud Security Solutions from CA Technologies (Online). Available: <https://www.ca.com/content/dam/ca/us/files/white-paper/identity-and-access-management-for-the-cloud-wp.pdf>. Accessed 23 Mar 2023
7. Majumdar S et al (2015) Security compliance auditing of identity and access management in the cloud: application to OpenStack. In: Proceedings—IEEE 7th international conference on cloud computing technology and science, CloudCom 2015, pp 58–65. <https://doi.org/10.1109/CloudCom.2015.80>
8. Kamga G-B (2018) CCSP guide archives—cloud security knowledge sharing by guy-Bertrand Kamga. Cloud Security Knowledge Sharing (Online). Available: <https://cloudsecurityknowledgesharing.com/tag/ccsp-guide/>. Accessed 17 Aug 2022
9. Hussein AKNH (2016) A survey of cloud computing security challenges and solutions. *Int J Comput Sci Inf Secur* 14(1):52 (Online). Available: <https://sites.google.com/site/ijcsis/>. Accessed 14 Aug 2022
10. Ibrahim A, Valli C, McAteer I, Chaudhry J (2018) A security review of local government using NIST CSF: a case study. *J Supercomputing* 74(11):5171–5186. <https://doi.org/10.1007/s11227-018-2479-2>
11. Ismail Z, Masrom M, Mohamad Sidek Z, Suhana Hamzah D (2010) Framework to manage information security for Malaysian academic environment. *J Inf Assurance Secur* 5(4):16–25. <https://doi.org/10.5171/2010.305412>
12. Yıldırım M, Mackie I (2019) Encouraging users to improve password security and memorability. *Int J Inf Secur* 18(6):741–759. <https://doi.org/10.1007/s10207-019-00429-y>
13. Cameron K (2005) The laws of identity (Online). Available: <http://www.identityblog.com>. Accessed 13 Aug 2022

PrimeSwitch—Encryption and Decryption Algorithm Using RSA Key Generation



Priyanka Bhatele, Ishan Shivankar, Shreya Sabut, Shivansh Saraswat, Shraddha Patel, Shrey Chougule, and Shreya Nale

Abstract This paper proposes an encryption and decryption algorithm PrimeSwitch that secures data using a key generated using RSA Method. It provides extremely safe data transmission in today's world of data threats and vulnerabilities. The RSA algorithm is a well-known asymmetric encryption algorithm used in our project as a key generation algorithm to develop a more complex system. This paper also emphasizes the effectiveness of PrimeSwitch compared to Base RSA. It compares to Base RSA in terms of encryption speed and complexity. Postulated results show that although Base RSA encrypts data faster, PrimeSwitch can decrypt it faster, and the encrypted files created with PrimeSwitch are also larger and more complex than those created with Base RSA.

Keywords Cryptography · Java · Key generation · RSA encryption · Decryption algorithm

P. Bhatele · I. Shivankar (✉) · S. Sabut · S. Saraswat · S. Patel · S. Chougule · S. Nale
Department of Engineering, Sciences, and Humanities (DESH), Vishwakarma Institute of Technology, Pune, Maharashtra 411037, India
e-mail: ishan.shivankar22@vit.edu

1 Introduction

These days due to the increase in communication technology, people can communicate swiftly as well as remotely. And due to this, there's a rise in demand for data security. As the development of Information Technology is rising, it increases the level of threat and vulnerabilities to the data to be transferred. Hence to counter this problem, cryptography researchers have always stressed finding new ways to encrypt the data. There are various ways in which the data can be secured which comes under cryptography [1–3].

Cryptography is a process dealing with encryption, or in simpler words, hiding or coding the data in such a way that only the person intended to receive the data can understand [4, 5]. It also deals with the decryption or decoding part, so that the encrypted message can be decoded.

Cryptography is a very important part of our lives now, knowing or unknowingly, we all use cryptography in one way or another in our daily lives, and that is justified. In this digital world, where everything is said and done, can be found if searched for it hard enough, so to protect the personal information of a person, cryptography is a vital part of every software that we use. From bank payments to messages, from your search history to the places you have been to, your applications record them all, but they also encrypt them to stop unauthorized persons from gaining access to it. This project is based on the integration of a widely used cryptography algorithm, the RSA algorithm, into a new algorithm, PrimeSwitch, and implementing it into real-life situations. Data encryption has become more secure due to RSA technology due to the use of two different keys, hence it becomes difficult to decrypt the message without having private keys [6]. The security is based on the difficulty of the factorization of the larger prime numbers chosen to generate the keys [7]. Thus, the project contributes to the safety of data transmission.

2 Literature Review

There are many ways in which a person can encrypt or decrypt the data. One such way is symmetric cryptography. Cryptography requires a key to encrypt and decrypt the code and symmetric cryptography uses a single key to encrypt and decrypt the message. It makes it a little less secure compared to asymmetric cryptography. Asymmetric cryptography, on the other hand, uses two keys system which is used for encryption and decryption individually, which makes it more secure [8].

As RSA is a widely used asymmetric cryptography algorithm, a lot of research has been conducted on its effectiveness in encrypting data and data security. Symmetric encryption has lesser computing efforts but many drawbacks, such as the secret key can be inferred, and that the secret key for encryption and decryption needs to be shared among the users [9, 10]. Asymmetric algorithms are different in this aspect. The RSA algorithm has a unique public key and private key for all users, the public key and private key only work with one another, this adds another layer of secrecy, as a decryption key is unique to its encryption key [11]. The public key or the encryption key is shared with all users and the specific public key of a user is used to encrypt the message for that user only, since the encryption and decryption keys work in conjunction with each other, only the specific decryption key of target user can decrypt the message and view it. This means even if all users use the same algorithm of the same program, secrecy is always maintained. This is the reason why RSA is so widely used in almost every place, from banks to messaging applications [4].

Although RSA has been proven to be secure, sometimes there occur security breaches such as a weak random number generator exposes a private key to vulnerability, so to overcome this, the RSA algorithm can also be integrated with other algorithms to make it safer, for example, Chinese remainder theorem, or we can use four prime numbers for the generation of public and private keys instead of two as in the traditional asymmetric key-based algorithm [12, 13].

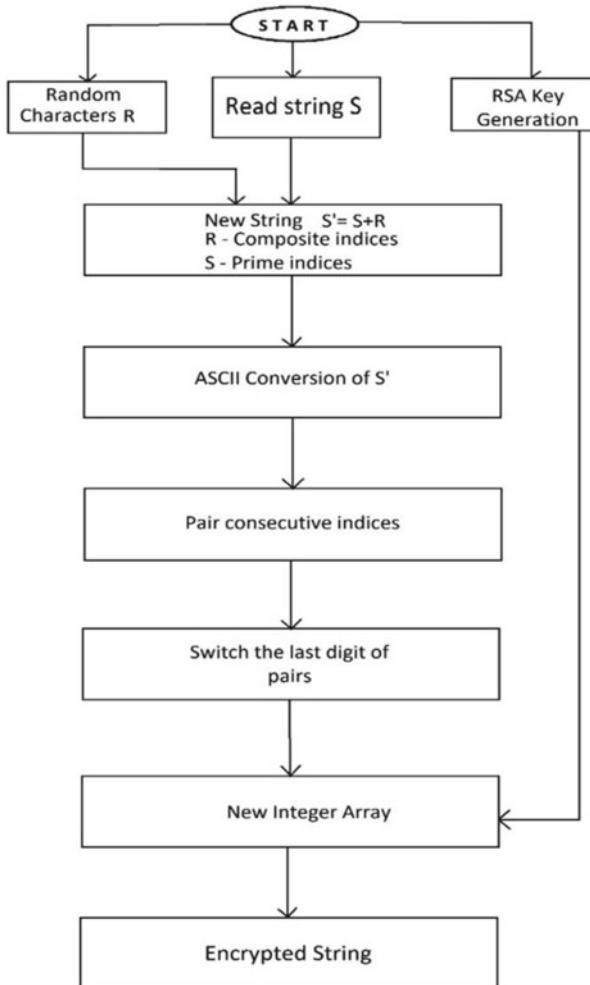
And hence this project has tried to integrate the RSA algorithm with PrimeSwitch to create a more complex system of encryption.

RSA algorithm is used for key generation, and the PrimeSwitch algorithm is used for the encryption of files.

3 Methodology/Experimental

3.1 Flowchart/Software Used

1. Netbeans IDE—Netbeans is a free, open-source integrated development environment. Netbeans IDE offers Java web, enterprise, desktop, and mobile application development tools.
2. Java programming language has been used to program the application.
3. PrimeSwitch—PrimeSwitch is the algorithm that has been made for this project.
4. Flowchart



3.2 Method Key Generation

At the start of the program, encryption keys are generated. To generate keys, two random prime numbers are taken. The inbuilt function of the 'java.util.BigInteger' class 'probablyPrime(<bit size>, <random>)' has been used. User-defined data type BigInteger is used to store the numbers. Prime numbers are stored in P and Q. And their multiplication is stored in BigInteger object N. N's Euler's totient is needed, which is $N' = (P-1) * (Q-1)$.

$(Q-1)$. N , N' (which is Euler's totient of N), P , and Q , are all stored in BigIntegers. To find E , a 110-bit random BigInteger is taken till we get a number less than n' and GCD of N' and E is 1.

To find D , inbuilt function ‘ $E.modInverse(N)$ ’ is used which returns the mod inverse of E and N' which is ' $((N'*k) + 1)/E$ ' where k is such that ' $((N'*k) + 1)%E = 0$ '.

Here, encryption key is (E, N) , and the decryption key is (D, N) , which are later saved in a file.

PrimeSwitch algorithm-

Let there be a general string of ‘ n ’ length, here for example, the word “HELLO” is being taken.

Let a random character be denoted by ‘ R ’.

```
String = "HELLO"
Random Character = "R"
```

Now the String is converted to a new string with original characters in prime indices, and the rest is filled with random character ‘ R ’.

```
New String = "R' R' S T R' R R' I R' R' R' N R' G"
```

All characters are converted to respective ASCII codes and stored in an integer array.

Let the ASCII code of ‘ R ’ be ‘99’.

```
Integer Array = {99', 99', 83, 84, 99', 82, 99', 73, 99', 99', 99',
78, 99', 71}
```

Now last digit all numbers are switched, since all characters are in upper case, all ASCII codes are 2 digits.

```
New Integer Array = {99', 99', 84, 83, 92', 89', 93', 79', 99', 99',
98', 79', 91', 79'}
```

This is the encrypted string using the PrimeSwitch algorithm, which is then encrypted using RSA keys for safe transfer from sender to recipient.

3.3 Working and Testing

Main JFrame

There are three buttons, for encrypting files, generating keys, and decrypting files.

When the key generation button is pressed, a new JFrame opens to generate keys (Fig. 1).



Fig. 1 Main JFrame of application

Key Generation JFrame

Keys can be generated and saved in a file (Figs. 2 and 3).

Encrypt file JFrame

To encrypt a message, input is taken in the text area, keys are taken in from the saved keys file, and the message is encrypted (Fig. 4).

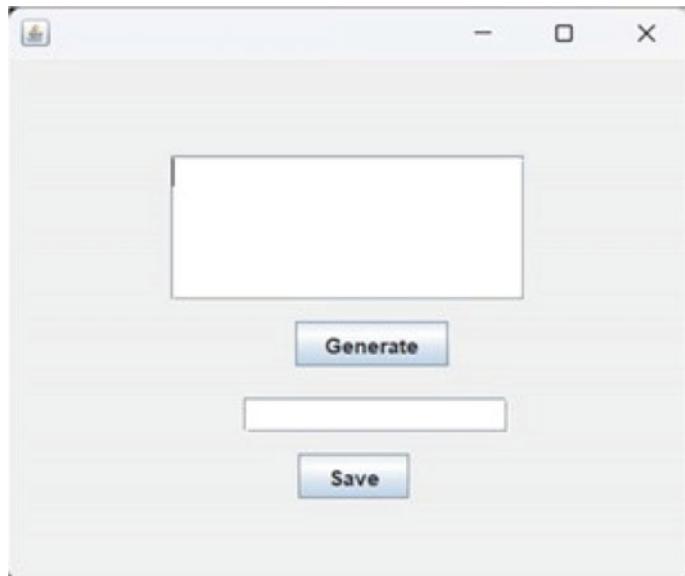


Fig. 2 Key generation JFrame with generate and save button

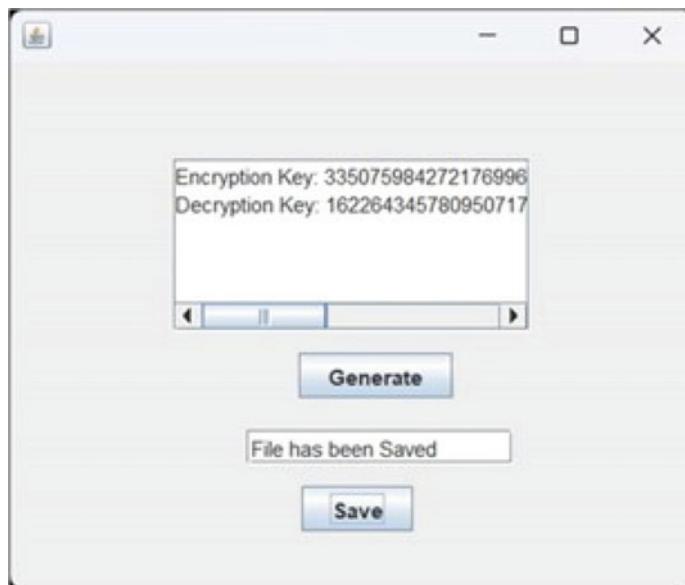


Fig. 3 Generated keys and saved the keys

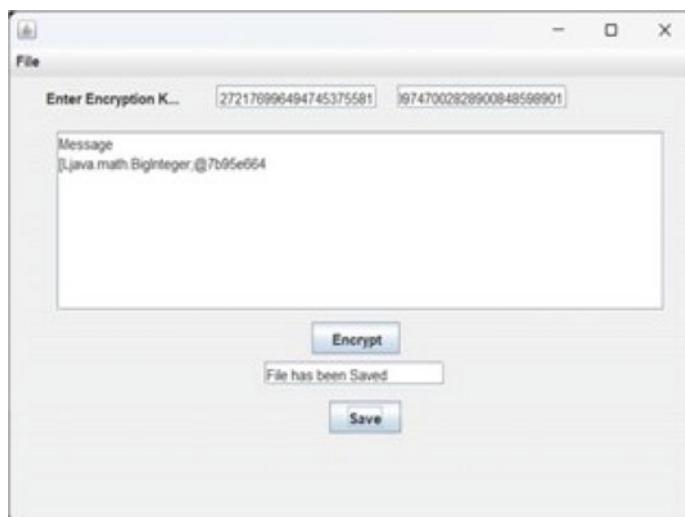


Fig. 4 Encrypting a message in encrypting JFrame

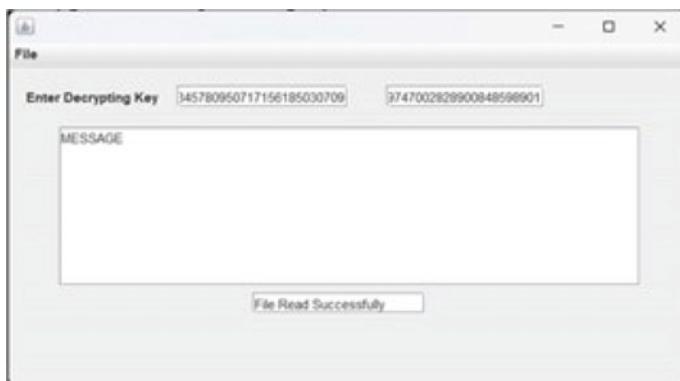


Fig. 5 Decrypting a message in decrypting JFrame

Decrypt file JFrame

To decrypt, keys and encrypted file are taken in, and the decrypted message is displayed in the text area (Fig. 5).

Key files are unique for each person and must be used according to whom the message is being sent.

4 Results

Figure 6 shows the comparison of encryption time between base RSA and the PrimeSwitch algorithm. Base RSA showed better performance in encryption than the new logic.

Figure 7 shows the comparison of the throughput of individual files for both, base RSA and PrimeSwitch. PrimeSwitch showed better performance during the decryption of the file.

Figure 8 shows the throughput for base RSA and PrimeSwitch for the encryption process. Figure 9 shows the throughput for base RSA and PrimeSwitch for the decryption process. Base RSA showed better performance in encryption, whereas PrimeSwitch showed better performance in decryption.

Figure 10 shows the comparison of encrypted file size. In PrimeSwitch, the encrypted file size increases a lot when the original file is increased in size.

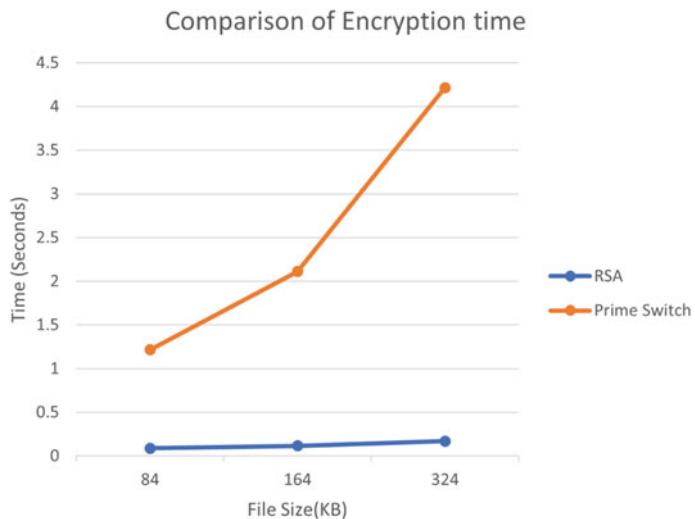


Fig. 6 Comparison of encryption time

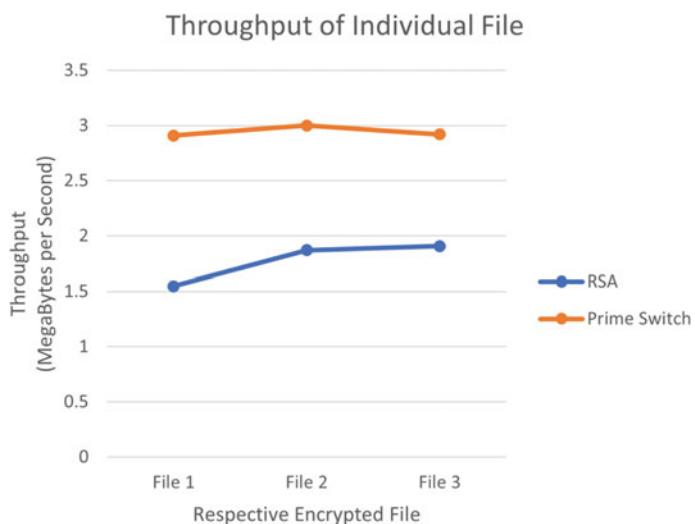


Fig. 7 Comparison of individual throughput of encrypted files

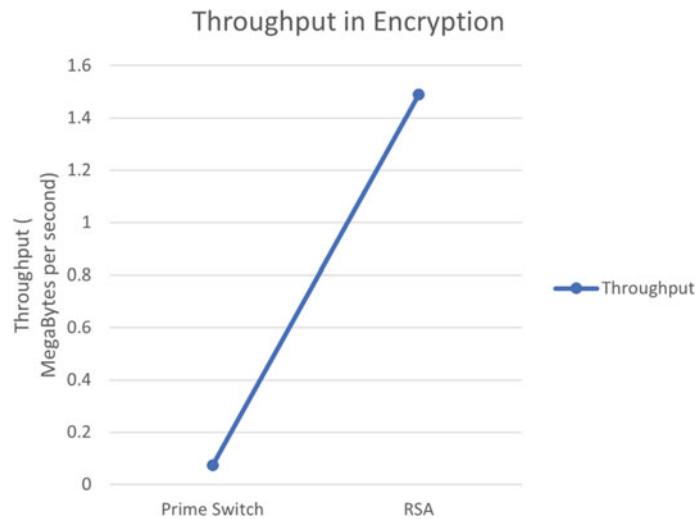


Fig. 8 Throughput during encryption

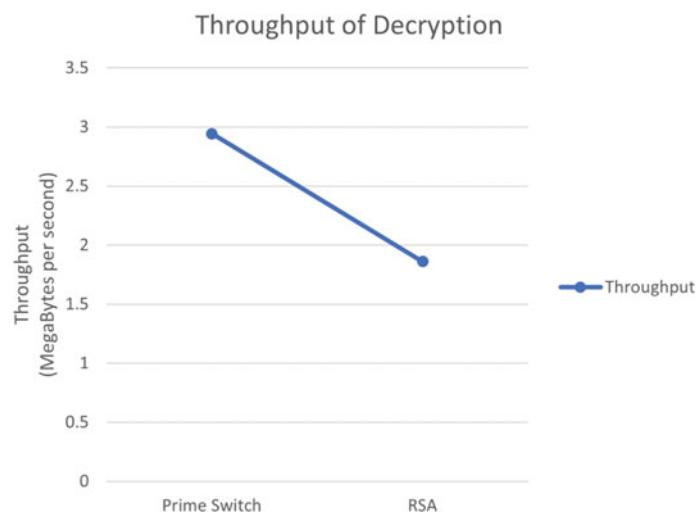


Fig. 9 Throughput during decryption

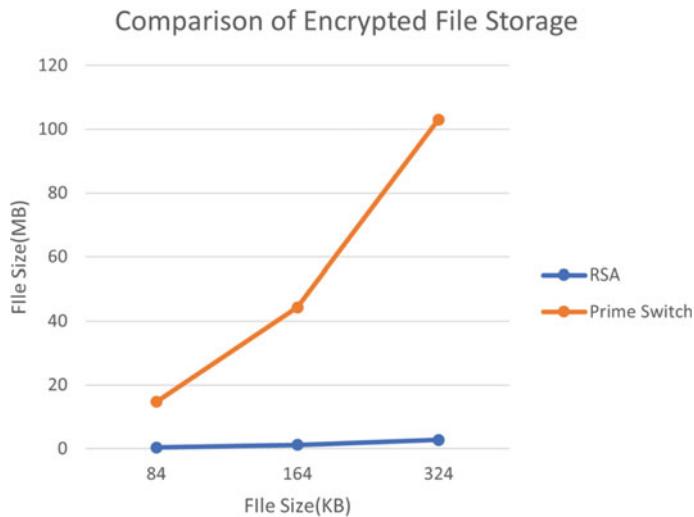


Fig. 10 Comparison of encrypted file size

5 Conclusion

This research paper presents the basic objective of cryptography. Cryptography has become one of the most required necessities in today's world. Thus, this paper provides improvements and enhancements to the already existing cryptography methods. This has been achieved by integrating PrimeSwitch with the RSA algorithm, which provides us with the asymmetric key generation method, which includes generating a public key and a private key, hence making it safer. RSA implements a public key cryptosystem that allows secure communications. Thus, the implementation of this project will help in creating a more secure networking process.

6 Future Scope

Compared to some other symmetric cryptosystems, the RSA method is more time-consuming. In reality, several problems, like timing attacks and key distribution issues, could potentially compromise RSA's security. A Riemann hypothesis answer would be a very serious threat to the security of the RSA protocol. As a result, it hasn't been established whether a remedy exists or not. There hasn't been much progress made on the Riemann hypothesis lately. Prime numbers, however, would be too simple to locate if a solution were to be discovered, and RSA would be rendered useless.

Acknowledgements This project would not have been possible without the support of Prof. Priyanka R. Bhatele whose constant guidance and feedback helped us to complete the project smoothly. We are honored that Vishwakarma Institute of Technology gave us this opportunity to research this topic and write a paper. Last but not least we would like to thank every team member and every person who has directly or indirectly supported the completion of this project.

References

1. Obaid TS (2020) Study a public key in RSA algorithm. *Eur J Eng Res Sci*
2. Abu-Faraj Mu, Alqadi Z (2021) Using highly secure data encryption method for text file cryptography, vol 21, pp 53–60. <https://doi.org/10.22937/IJCSNS.2021.21.12.8>
3. Chua C, Milosavljevic M, Curran JR (2009) A sentiment detection engine for internet stock message boards. In: Proceedings of the Australasian language technology association workshop 2009. Evgeny Milanov, RSA Algorithm, 3 June 2009
4. Bansal N, Singh S (2020) RSA encryption and decryption system. *Int J Sci Res Comput Sci Eng Inf Technol.* ISSN: 2456-3307 (www.ijsrcseit.com). <https://doi.org/10.32628/CSEIT206520>
5. Rivest RL (2003) Cryptography, computers in. MIT Laboratory for Computer Science, 545 Technology Square, Cambridge, MA 02139, USA
6. Mahajan P, Sachdeva A (2013) A study of encryption algorithms AES, DES and RSA for security. *Global J Comput Sci Technol* 13
7. Wang S, Liu G (2011) File encryption and decryption system based on RSA algorithm. In: 2011 international conference on computational and information sciences, Chengdu, China, pp 797–800. <https://doi.org/10.1109/ICCIS.2011.150>
8. Farah S, Javed M, Shamim A, Nawaz T (2012) An experimental study on performance evaluation of asymmetric encryption algorithms
9. Zhou X, Tang X (2011) Research and implementation of RSA algorithm for encryption and decryption. In: Proceedings of 2011 6th international forum on strategic technology, Harbin, Heilongjiang, pp 1118–1121. <https://doi.org/10.1109/IFOST.2011.6021216>
10. Goshwe NY (2013) Data encryption and decryption using RSA algorithm in a network environment. *IJCSNS Int J Comput Sci Netw Secur* 13(7)
11. Abdalrdha ZK, Al-Qinani IH, Abbas FN (2019) Subject review: key generation in different cryptography algorithms. *Int J Sci Res Sci Eng Technol*
12. Chaudhury P et al (2017) ACAFP: asymmetric key based cryptographic algorithm using four prime numbers to secure message communication. A review on RSA algorithm. In: 2017 8th annual industrial automation and electromechanical engineering conference (IEMECON), Bangkok, Thailand, pp 332–337. <https://doi.org/10.1109/IEMECON.2017.8079618>
13. Abdeldaym R, Abd Elkader HM, Hussein R (2019) Modified RSA algorithm using two public key and Chinese remainder theorem. *IJ Electron Inf Eng* 10:51–64. <https://dx.doi.org/10.6636/IJEIE.201903/51-64>

Hybrid Lightweight Cryptography Using AES and ECC for IoT Security



Neha N. Gharat and Lochan Jolly

Abstract The Internet of Things (IoT) transforms everyone's life by providing features such as controlling and monitoring connected smart objects. The Internet of Things (IoT) architecture is intricate and sophisticated, owing to the substantial number of interconnected devices, diverse link-layer technologies, and multiple services integrated within the system. The adaptation of these devices is increasing exponentially, which creates an extensive amount of data for processing and analyzing. Designing security solutions for the Internet of Things (IoT) requires acknowledging the novel security and privacy challenges that emerge due to the nature of this technology. Securing end-to-end connections in IoT poses a significant challenge due to the heterogeneous nature of IoT communications, as well as the disparities in resource capabilities across IoT devices. Due to the limited computing power, energy, and memory of most IoT devices limited security solutions are available. As a result, IoT requires robust security solutions that can efficiently meet specific security and privacy requirements while having a lightweight impact on device resources such as microcontrollers, memory, and energy. The feature of the work is the design of a hybrid lightweight cryptography to enable secure data in IoT against IoT attacks. One of the security fundamentals to secure IoT data is to design lightweight cryptography for IoT devices that are resource-constrained in terms of computational capability, memory space, and battery power. To find an effectual and lightweight key establishment the combination of symmetric and asymmetric encryption algorithms, i.e., hybrid lightweight cryptography is proposed. The primary functionality of the proposed system revolves around the integration of ECC and the advanced encryption standard (AES) method. This combination is employed to uphold data integrity. Security analysis is conducted to prove the scheme fulfills the security requirements and evaluated the performance in terms of computational cost, power consumption, memory, and key size.

N. N. Gharat (✉) · L. Jolly

Thakur College of Engineering and Technology, Mumbai, India

e-mail: nehagharat10@gmail.com

L. Jolly

e-mail: lochan.jolly@thakureducation.org

Keywords Lightweight cryptography · AES · ECC · Hybrid cryptography

1 Introduction

The Internet of Things is about the change in the modern control and monitoring of connected smart objects by their features. The IoT is an evolving technology worldwide, which facilitates the connection of sensors, home appliances, automobiles, and offices. IoT architecture provides various smart applications. IoT architecture is multifaceted as various devices, technologies, and services are integrated into the system. The increasing use of these devices creates a very high amount of data for processing and analysis. Although these devices bring comfort to human beings, they are vulnerable to several threats and security issues, which makes users skeptical when adopting this technology in critical applications like healthcare systems, smart homes, etc. This also creates challenges for the growth of IoT in near future. The IoT poses new privacy and security tasks that must be taken into account while designing security for IoT solutions, heterogeneous nature of IoT communications and inequality in resource competencies between IoT devices make it difficult to provide the totally required secured connection. IoT requires robust security solutions that can efficiently handle the security and privacy requirements with very less impact on device resources (memory, energy, and controller). According to markets and markets predictions, the global IoT security market size will grow from USD 14.9 billion in 2021 to USD 40.3 billion by 2026, at CAGR of 22.1%. The security requirements for IoT systems vary based on their domains of application. Several studies have reported that security issues are prevalent in IoT networks. Thus, ensuring security and privacy in IoT systems is critical to their successful deployment and adoption [1–7]. These challenges include problems with authentication, information leakage, privacy, verification, interfering, jamming, and eavesdropping among others. When it comes to securing important data in resource-constrained environments such as IoT devices, it is essential to carefully evaluate the available cryptographic methods. While there are several options to choose from, not all of them may be practical or effective for these specific scenarios. Due to the limited resources available on these devices, lightweight cryptographic solutions that are area-efficient and power-efficient are being researched extensively. All IoT devices are susceptible to IoT-specific threats, which can compromise the security of the data being transmitted or stored on these devices. If we continue to use the present IoT device design without considering the security implications, we may face security adversity in the near future. The current cryptographic primitives can be separated into two groups: asymmetric key cryptography and symmetric key cryptography. To address the security challenges posed by IoT devices, lightweight cryptographic solutions are being developed that provide efficient security mechanisms suitable for use in resource-constrained environments. These lightweight cryptographic solutions aim to provide the necessary security while minimizing the area and power consumption requirements of IoT devices. Conventional encryption methods that

use symmetric or asymmetric keys provide good security for text files but are not as powerful for data like audio, video, and images. Attempting to use symmetric algorithms to secure IoT data can make them vulnerable to brute-force attacks. To address this issue, hybrid cryptography that combines symmetric and asymmetric keys is used to provide better security for IoT data. Therefore, this paper proposes solutions for the security aspects of the IoT environment.

The paper is organized in five sections. Sections 1 and 2 includes an introduction and related work. The implementation methodology, experimental results, and conclusion are discussed in Sects. 3, 4, and 5, respectively.

2 Related Work

One of the security fundamentals to protect IoT is to design lightweight cryptography for IoT devices that are resource-constrained in terms of computational capability, memory space, and battery power. In this section, the related work of lightweight and hybrid lightweight cryptography is explored for IoT security. The current methods are compared in terms of physical cost, security level, and hardware and software performance.

The study conducted in reference [8] involved a comparison of present algorithms based on several factors, including physical cost, hardware and software performance, and security level. The comparison of 42 existing symmetric key lightweight cryptography (plain encryption) algorithms over the performance parameters recommended by the NIST is analyzed. It is observed that PRESENT is the most efficient and ISO/IEC-approved algorithm. The analysis of many lightweight cryptographic algorithms based on their key size block and rounds is performed in [9]. The lightweight cryptographic algorithms based on lightweight block ciphers, stream ciphers, and hash functions are compared.

A total of 54 LWC primitives are compared in [10]. The performance analysis comparison of the ciphers was done in terms of physical and performance metrics. From the observed trends, AES has emerged as the leading cipher among block ciphers. Meanwhile, in asymmetric cryptography, ECC is a prominent choice for providing both confidentiality and authentication. The ultra-lightweight method (ULM) is proposed in [11]. The proposed method is designed using bit-slice, WTS, and inclusive methods. This is also referred to as the hybrid method. The analysis of ULC showed an improvement w.r.t performance and security. The encryption algorithm has used a 128-bit block cipher and Feistel and substitution permutation architectural methods are used which result in increasing the complexity of the encryption.

A new lightweight cryptographic algorithm is proposed in [12] that can be used in cloud computing to secure applications. In this algorithm, Shannon's theory of diffusion and confusion is used. Logical operations such as XOR, XNOR, shifting, and swapping are used to enhance the security level. The proposed algorithm is

compared with other cryptographic algorithms and experimental results showed that the NLCA algorithm has improved security level and low computational cost.

In [13] lightweight cryptography using ECC is proposed to produce a strong key with a minimum length. It has been observed that ECC gives us better results by consuming less power and memory. So, it is easily adaptable to sensors, RFIDS, smart cards, and wireless devices. A proxy re-encryption (PRE) cryptographic algorithm that can be used for fog-to-things communication is proposed in [14]. In the hybrid proxy re-encryption method corrected block tiny encryption algorithm (XXTEA, symmetric cipher) and ECC (asymmetric algorithm) are combined. The performance analysis showed that XXTEA requires less computation and memory consumption compared with AES-128 and AES-256, respectively. Cut and paste attacks are eliminated in this method as the change in a single bit will result in a change of half of the bits of the total block. The proposed scheme also improved security in terms of confidentiality and collision attack resistance.

Dahiya and Bohra proposed the parallel partial model (PPM), which is a robust encryption model that utilizes a combination of the iAES and mECC to enhance data security. The PPM model is designed to be complex and powerful, providing an elevated level of protection to encrypted data [16]. The authors in [17] suggested a hybrid approach of ECC and AES for encrypting and decrypting multimedia data including text, images, and videos. The proposed system was found to provide a high level of security, as even if an intruder penetrates the keys, it would take a significant amount of time, estimated to be several years, to decipher the data. However, the study did not include an evaluation of the system's performance. Pourali proposed a secure data protection approach for electronic commerce that utilizes hybrid ECC and AES coding.

The approach is implemented on an SMS-based model, ensuring secure payment with all necessary security specifications [18]. The proposed method was not accompanied by any assessment of its effectiveness. In [19] a hybrid approach is suggested that incorporates RSA and AES encryption algorithms to improve data security for cloud users. The proposed method was employed to upload and download modules of a cloud architecture. However, there was no performance analysis conducted to estimate the efficiency of the strategy. A novel approach based on the HAN algorithm, which is a combination of the AES symmetric and NTRU asymmetric encryption algorithms, is proposed in [20].

The proposed method was assessed based on performance parameters such as implementation time and consumption of power due to lower fiscal complexity, the algorithm required less memory compared with other methods. Dsouza presented a novel approach that utilized a hybrid method of dynamic key and S-box generation. This technique introduces increased complexity in the encrypted data to enhance confusion and diffusion, making the cipher text more resistant to attacks such as brute-force, differential, algebraic, and linear attacks.

In [22] authors introduced a novel security protocol that utilizes a hybrid approach. In this protocol, the message is divided into n chunks, each consisting of 128 bits. These blocks are further divided into two parts. The first half chunks are encrypted using a combination of AES and ECC in a hybrid encoding algorithm. The authors

in [23] proposed an algorithm to enhance data security for the cloud that combines ECDSA, SHA256, and AES to protect data in the cloud. This algorithm ensures that data messages sent to and received from the cloud are secure. To upload a message file to the cloud, the user generates ECDSA signatures with SHA256 message digests on their system.

3 Proposed Hybrid Cryptography Approach

The proposed hybrid cryptography method goals to create a safe and effective encryption algorithm by combining different encryption structures into a hybrid method. The ultimate goal is to ensure that data is encrypted and transmitted securely, without compromising on efficiency or security. In the proposed methodology a combination of the AES and ECC is used for developing hybrid cryptography.

3.1 AES

The AES algorithm is named according to the key length used in the ciphering process, with variants including AES-128, AES-192, and AES-256 [25]. Each block of data that is ciphered or deciphered using AES is comprised of 128 bits or 16 bytes. The key used in AES is also represented as a 4×4 byte matrix and expanded into an array of key schedule words. The key schedule consists of 44 words, 52 words, and 60 words. Every word contains four bytes. The input block of 128 bits is copied into a 4×4 byte matrix called the stated matrix during the encryption process. The number of rounds used in AES encryption (10, 12, or 14) depends on the length of the key being used (128, 192, or 256 bits). As illustrated in Fig. 1 [24], the encryption process consists of four transformations in each round, namely sub bytes, shift rows, mix columns, and add round key. However, the final round only consists of three transformations, which are sub bytes, shift rows, and add round key. The decryption process follows the same steps as the encryption process [25].

3.2 ECC

Elliptic curve cryptography (ECC) was introduced in 1985 by Koblitz and Miller as a latent solution to traditional public key cryptographic scheme such as RSA. ECC uses elliptic arcs defined over finite fields. The elliptic curve cryptography employs finite field elliptic arcs. The mathematical operations in ECC are defined on an elliptic curve equation of the form $y^2 = x^3 + ax + b$, where $4a^3 + 27b^2 \neq 0$ and a and b are constants that define a particular elliptic curve. The set of pairs (x, y) that satisfy this equation, along with the point at infinity, form the points on the elliptic

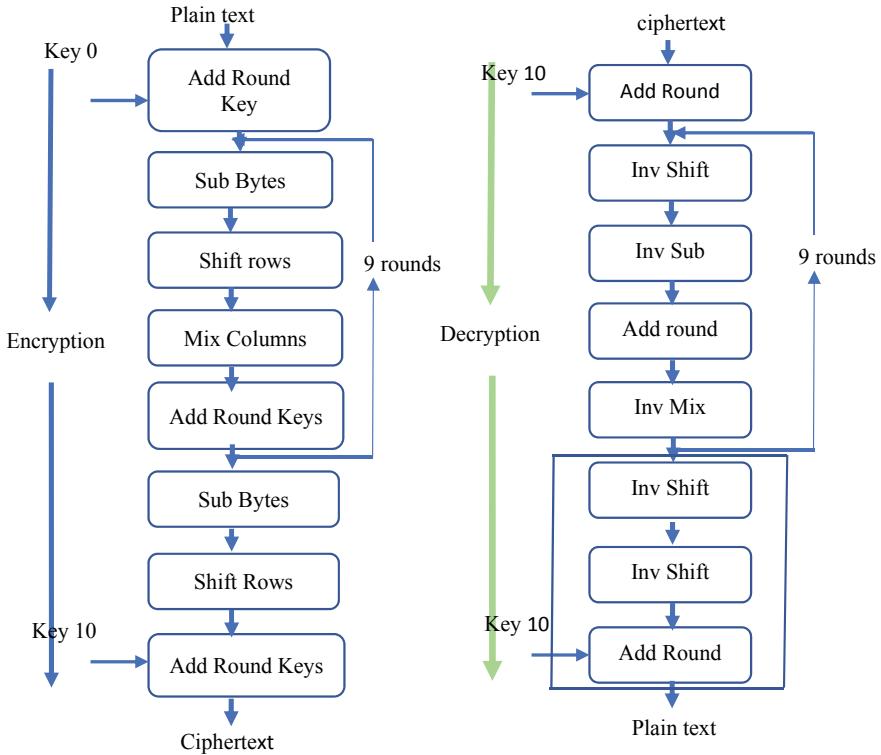


Fig. 1 AES implementation process [25]

curve [27]. To generate a private key in ECC, a random number ‘ r ’ is chosen. The corresponding public key is then derived as $P = r * G$, where G is the generator point of the subgroup. Figure 2 shows the process of ECC encryption and decryption.

3.3 Hybrid Proposed Method

The proposed cryptographic technique for securing data in IoT applications involves using a combination of ECC and AES. While AES is identified for its efficiency, the larger key size can lead to slower processing. On the other hand, ECC is recognized for its reduced key size and can be used to reduce the key size in AES encryption, resulting in faster processing [29].

Encryption and decryption key standards are used by ECC to create a secure key system with reduced key size. By combining ECC and AES, the subsequent encryption and decryption process is appropriate for securing data in IoT networks. To implement this technique, the AES key is encrypted by using ECC. The ciphertext is

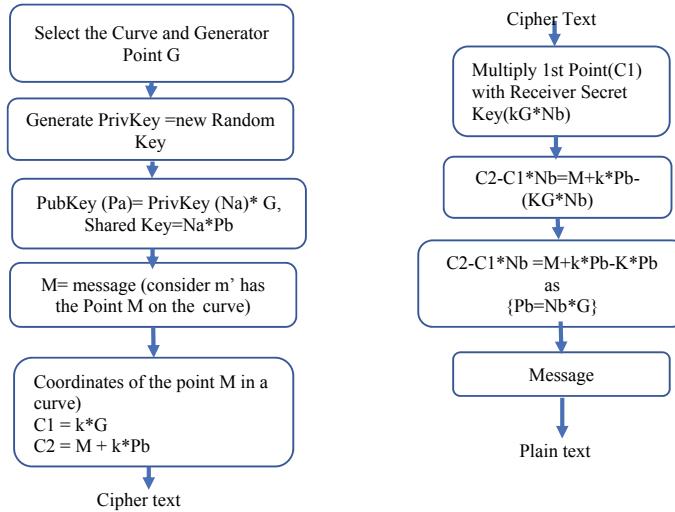


Fig. 2 ECC encryption and decryption process [27]

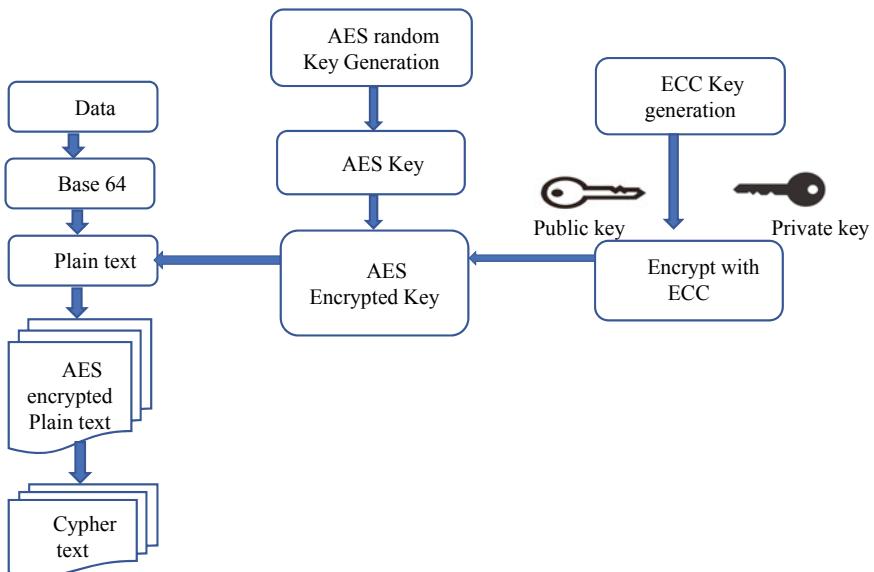


Fig. 3 Proposed hybrid cryptography using AES and ECC

generated by using AES encryption technique. The proposed algorithm is represented in a block diagram shown in Fig. 3.

The proposed model for securing IoT data comprises a hybrid encryption approach that employs both AES and ECC ciphers. This approach combines the robustness of ECC with the simplicity of AES to provide enhanced security.

To encrypt the data, it is first converted into a base64 encoded text format and performed primary encryption using AES. The AES keys are created arbitrarily. The AES keys are then encrypted using ECC public key generated from the input base64 encoded text file. The base64 encoded plaintext is encrypted by using the encrypted AES key, resulting in compressed ciphertext that has experienced two stages of mixed encryption with ECC and AES. This hybrid method offers improved security than using a single encryption model alone. To decrypt the data, the converse process is applied, including a somewhat difficult procedure. Overall, this approach provides secure and efficient encryption for IoT data using symmetric encryption with AES and asymmetric encryption with ECC.

4 Results and Discussion

This research proposes a novel hybrid encryption scheme using a combination of AES and ECC that is scalable in terms of computational and communication costs. Many AES and ECC schemes are presented to provide data security. Analysis of difference has been done by selecting different performance parameters. By combining ECC and AES, the system's overall security is heightened as it adds complexity and enhances its resilience against attacks.

4.1 Performance Analysis

To confirm the efficacy and distinctiveness of the proposed method, the algorithms are implemented using Python. The combination of ECC and AES encryption resulted in improved security and efficiency for securing data in cloud storage. This approach ensured that data transmitted over the cloud was encrypted and decrypted using secure connections. The use of Python and SimpleCV library provided an effective means of validating the proposed algorithm, demonstrating the potential of this approach to secure a variety of data types. Overall, the combination of ECC and AES encryption has the potential to enhance data security for the IoT applications such as healthcare and smart grids. The parameter comparisons are shown using a laptop that has the following specifications: Windows 10, Processor Intel (R) core (TM) i7-8565U CPU @ 1.80GHz, 1.99GHz, RAM 8GB.

Table 1 Comparison of encryption and decryption time of proposed cryptographic algorithm

Message length in (bits)	Encryption time (milli seconds)			Decryption time (milli seconds)		
	Proposed AES	Proposed ECC	Proposed hybrid	Proposed AES	Proposed ECC	Proposed hybrid
48	12.77	3.992	1.9929	0.9974	3.987	0.9986
64	14.409	3.9853	1.996	0.9973	2.996	0.9960
96	12.96	4.986	2.156	0.9961	3.989	0.9374
128	14.95	4.980	2.99	0.9973	2.99	0.9913
256	15.164	3.989	2.9915	0.9965	3.995	1.9946
512	15.196	5.142	2.9874	0.9974	2.995	1.004
1024	14.029	4.169	1.9845	0.9966	4.083	0.9959
2048	14.84	4.353	3.1886	0.9939	3.990	1.052
4096	16.806	4.4718	1.99	0.9981	3.990	0.997

4.1.1 Encryption Time

The encryption time refers to the amount of time it takes for an encryption algorithm to convert plaintext into ciphertext. The encryption time can vary depending on the size of the plaintext, the speed of the computer or device performing the encryption, and the specific encryption algorithm being used. Some encryption algorithms are designed to be fast and efficient, while others prioritize security over speed. The encryption time can also be affected by the implementation of the algorithm, as well as other factors such as network latency and processing power. Table 1 gives the comparison of encryption and decryption time of implemented AES, ECC, and hybrid cryptography algorithms. Figure 4 shows the encryption time of various message sizes from 48 to 4096 bits. Figure 5 illustrates that the proposed algorithms exhibit decreased encryption times, resulting in reduced computational costs. This observation highlights the effectiveness of the proposed methods. The result showed that the proposed hybrid AES-ECC algorithm had a faster encryption time and decryption time compared with the existing algorithms.

4.1.2 Decryption Time

It is the amount of time it takes for a decryption algorithm to convert a ciphertext back into its original plaintext form. Like the encryption time, the decryption time can vary based on the size of the ciphertext, the speed of the computer or device performing the decryption, and the specific decryption algorithm being used. Table 5 gives the comparison decryption time of AES, ECC, and proposed hybrid cryptography algorithms. The decryption time may also be affected by the same factors that impact the encryption time, such as network latency, processing power, and the implementation of the algorithm. In some cases, the decryption time may be longer

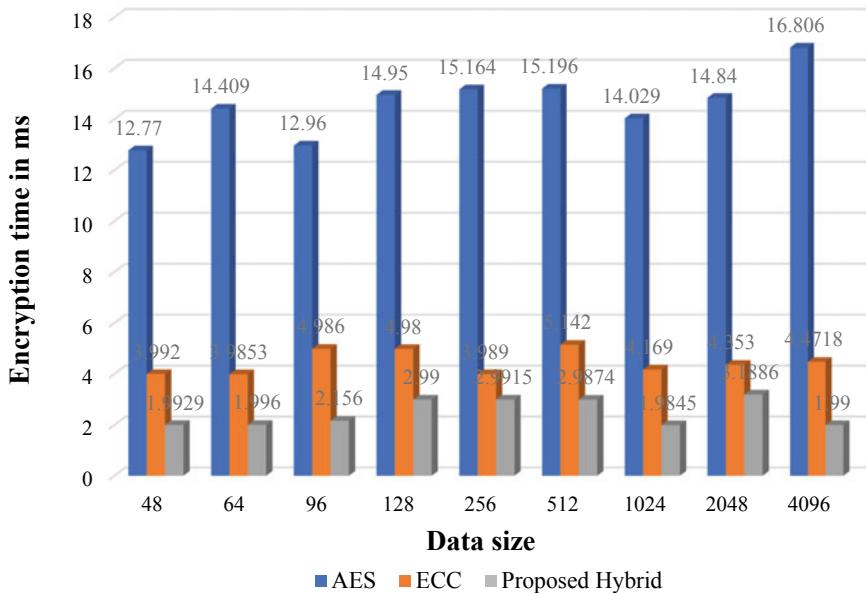


Fig. 4 Encryption time comparison of proposed algorithms

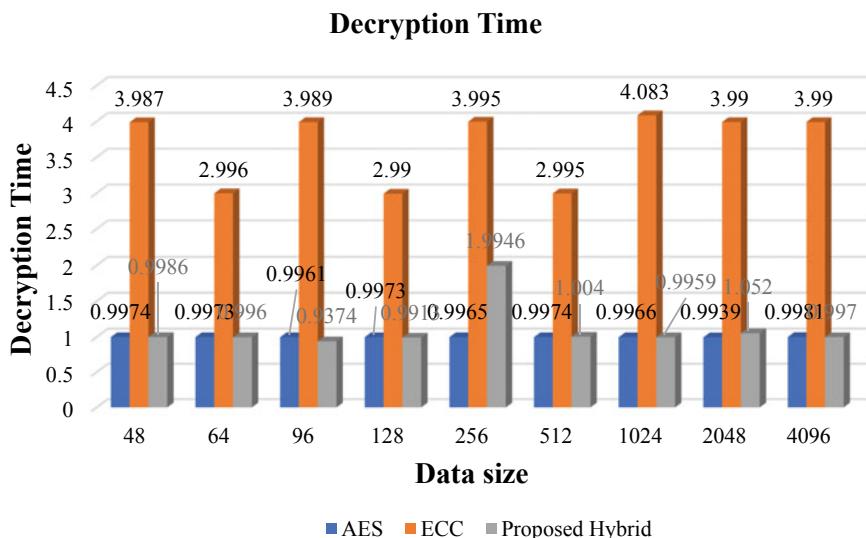


Fig. 5 Decryption time comparison of proposed algorithms

Table 2 Comparison of execution time of proposed cryptographic schemes with present schemes

Message bits	Execution time in ms			
	Proposed AES	AES [29]	Proposed ECC	ECC [29]
48	13.7674	19.51	7.979	11.55
64	15.4063	28.46	6.9813	15.48
96	13.9561	37.37	8.975	23.03
128	15.9473	42.97	7.97	35.19
256	16.1605	48.72	7.984	41.22

than the encryption time, especially if the decryption algorithm is designed to be more secure or computationally intensive than the encryption algorithm. The evaluation of encryption times for various cryptographic schemes is presented in Fig. 4. It's important to carefully evaluate the trade-off between encryption and decryption time, security, and robustness when choosing encryption algorithms.

Based on the analysis, it can be concluded that the hybrid encryption scheme enables users to encrypt data using two robust encryption techniques without incurring additional time for encryption and decryption processes across various message lengths. The same results can be seen in Figs. 4 and 5.

In this paper the proposed cryptographic algorithms are compared with the existing algorithms. Table 2 gives the encryption and decryption time which is reduced when compared with the existing algorithms. Figure 6 illustrates a comparison of execution times among different algorithms, indicating that the proposed method operates more efficiently than the others. It has been also observed that though the data size has been increased the encryption time increases slightly.

Table 3 gives the comparison of the time of encryption and decryption with various key lengths. In terms of performance and energy consumption, the AES-128 cipher outperforms the AES-256 cipher. Specifically, AES-128 ciphers require less energy and take less time to encrypt 32 bytes of data compared with AES-256. This is due to the smaller key size and the reduced number of rounds required by AES-128, which also results in more than a 50% reduction in computational cost when compared with AES-256 block ciphers. But if AES-128 is compared with AES-256 in terms of security level then AES-256 has better security. To address this limitation and to improve the security of data the proposed method focuses on the combination of AES and ECC encryption techniques. The analysis shows that the proposed hybrid method requires less execution time when compared with an existing method and reduces computation costs.

4.1.3 Throughput

Throughput refers to the rate at which data can be processed within a specified time frame and is often used as a performance metric for encryption and decryption algorithms. A higher throughput indicates that more data can be encrypted or decrypted

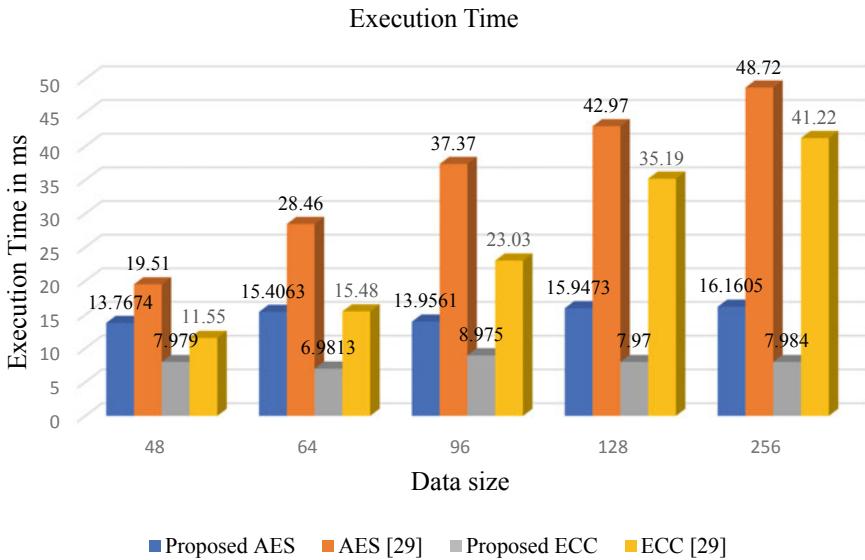


Fig. 6 Comparison of execution time of proposed and current schemes

Table 3 Comparison of encryption and decryption time with various key lengths

Key sizes	Encryption time				Decryption time			
	Proposed hybrid	Hybrid [31]	Proposed AES	AES [31]	Proposed hybrid (ms)	Hybrid [31] (S)	Proposed AES (ms)	AES [31] (S)
128	0.04	2.47	0.0051	3.59	0.9970	1.77	0.998	2.82
192	0.062	2.54	0.0069	3.48	0.9971	1.88	0.999	2.93
256	0.051	2.6	0.0696	3.6	0.999	2.10	1.031	3.08

in a given time period, resulting in better performance. Typically, throughput is measured in bytes per second by dividing the total amount of plaintext data (in bytes) by the encryption time (in seconds). Algorithms that have higher throughput are generally more efficient and can handle larger volumes of data in less time, which is particularly beneficial for real-time data processing or data transmission applications. From Table 4, it is observed that the hybrid approach AES-ECC gives the maximum throughput among the used three cryptography schemes with several message lengths.

$$\text{Encryption Throughput} = \frac{L}{\text{encryption time}} \quad (1)$$

$$\text{Decryption Throughput} = \frac{L}{\text{Decryption Time}} \quad (2)$$

Table 4 Comparison of encryption and decryption throughput for proposed cryptographic schemes

Message length (bits)	Encryption throughput in (bytes/ms)			Decryption throughput in (bytes/ms)		
	Proposed AES	Proposed ECC	Proposed hybrid	Proposed AES	Proposed ECC	Proposed hybrid
48	0.493	1.5028	3.01	6.015	6.01	6.01
64	0.4875	2.007	4.0064	8.022	4.011	8.025
96	0.925	2.406	5.564	12.06	3.008	12.8
128	1.06	3.208	5.3474	16.043	5.347	16.14
256	2.11	8.021	10.696	32.112	8.0084	16.043
512	4.211	12.44	21.42	64.16	21.363	63.7
1024	9.123	30.696	64.49	128.43	31.343	127.52
2048	17.24	58.799	80.286	255.57	64.157	243.34
4096	30.4565	114.49	256.82	512.97	128.32	511.54

Table 5 Encryption process power consumption

Power consumption of encryption in W

Key size	Proposed hybrid	Hybrid [29]	Proposed AES	AES [29]
64	0.55	2.4	–	–
128	0.61	2.47	0.076	3.59
192	0.94	2.54	0.0916	3.48
256	1.2	2.6	0.12	3.6

4.1.4 Power Consumption

Encryption algorithms play a vital role in ensuring security systems. Nevertheless, these algorithms require considerable computing resources, including CPU time, memory, and battery power. Power consumption can be quantified in terms of watts, which is a measure of the amount of energy consumed by an electronic device. The energy consumption E in Joule (J) used for the experiments during encryption/decryption was calculated using the formula:

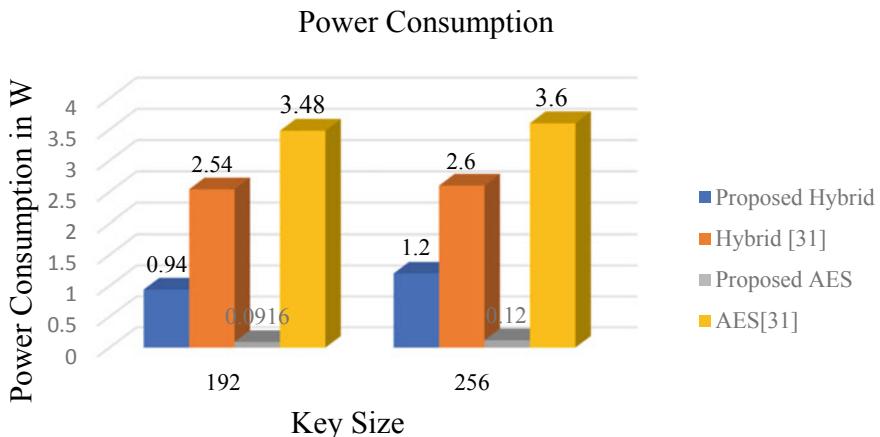
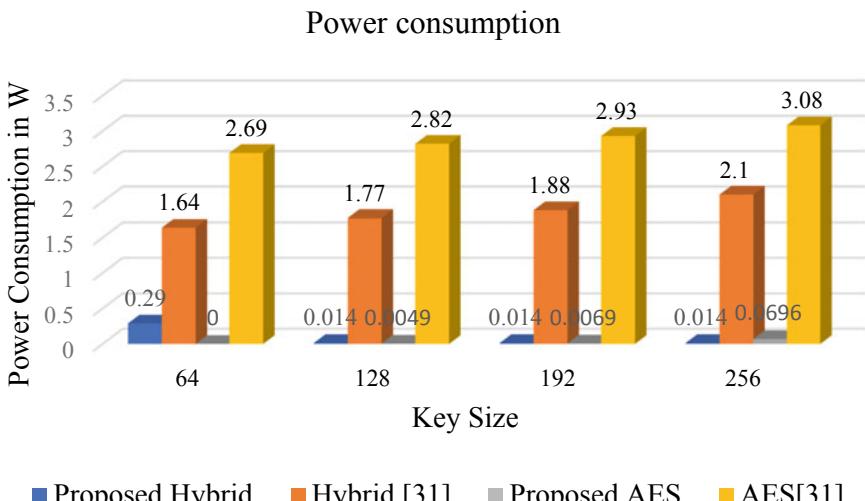
$$E = V_{cc} \times I \times t$$

where t = encryption/decryption time.

In Tables 5 and 6, the power consumption of different methods is mentioned. Figures 7 and 8 show that the power consumption of the proposed method is less when compared with existing algorithms. As the proposed hybrid method consumes less power it can be used for the security of resource-constrained IoT devices.

Table 6 Decryption process power consumption

Power consumption of decryption in W				
Key size	Proposed hybrid	Hybrid [29]	Proposed AES	AES [29]
64	0.29	1.64	—	—
128	0.014	1.77	0.0049	2.82
192	0.014	1.88	0.0069	2.93
256	0.014	2.1	0.0696	3.08

**Fig. 7** Power consumption of encryption of proposed and existing algorithms**Fig. 8** Power consumption of decryption process of proposed and current algorithms

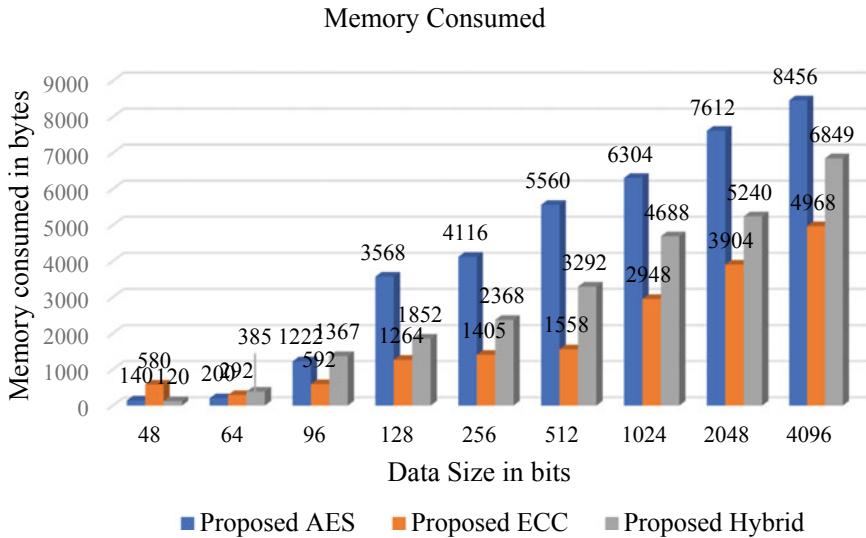


Fig. 9 Memory consumption of encryption of proposed algorithms

4.1.5 Memory Consumption

The memory utilized is the main memory unlisted during the encryption. The hybrid algorithm necessitates a moderate memory size. In proposed AES algorithm it uses a key size of 256 bits. As larger the key size, the more memory is required in AES to store it and perform the operations. The memory consumption for AES is dependent on the size of the data being encrypted and the key size used. ECC generally requires less memory compared with AES, but the actual memory utilization could depend on the size of the data being processed and specific implementation. Figure 9 depicts the comparison of the memory consumption of proposed cryptographic methods. The proposed hybrid method requires less memory when compared with AES.

4.1.6 Avalanche Effect (Av)

It is a measure of the sensitivity of a cipher to small changes in the input plaintext. It is an important characteristic of cryptographic algorithms and is used to assess the security of the cipher. A high avalanche effect of the encryption process is such that even slight alterations to the plaintext result in significant changes in the resulting ciphertext. This characteristic makes it considerably harder for an attacker to deduce the correlation between the original message and the encrypted version. Table 7 gives the Av percentage for the hybrid algorithm under discussion. The greater the avalanche effect of an algorithm, such as the proposed one, the more challenging is to break it. As a result, it can be inferred that the proposed hybrid algorithm offers higher

Table 7 Avalanche effect

Encryption algos	Change in key by 1-bit	Avalanche effect	Change in plain text by 1-bit	Avalanche effect
AES	66	0.551 (55.1%)	75	0.5859 (58.59%)
Hybrid	78	0.609 (60.9%)	85	0.664 (66.40%)

level of security due to its superior AV effect. The formula for the Av is commonly used to calculate the sensitivity of a cipher to small changes in the input. By comparing the number of changed bits in the ciphertext to the total number of cipher bits, the avalanche effect can provide a quantitative measure of the security of the cipher. A high avalanche effect indicates that the cipher is extremely protected and that small changes in the input have a significant impact on the output. On the other hand, a low AV effect indicates that the cipher is not as protected and that small changes in the input have a minimal impact on the output. To measure the avalanche effect in proposed encryption technique, 128-bit block of plaintext is taken corresponding to its 128-bit key and then 1-bit is inverted and the corresponding number of bits that are changed in the cipher text are counted.

$$\text{Avalanche Effect Av} = \frac{\text{Number of bits changed in cipher text}}{\text{Total number of bits in cipher text}} \quad (3)$$

5 Conclusion

The paper proposes various techniques and algorithms for enhancing data security in IoT applications. Among these, the hybrid technique stands out for its ability to combine multiple cryptographic algorithms, offering superior security compared with traditional single-method approaches. The section on the review of hybrid cryptographic approaches presents an in-depth analysis of various cryptographic methods and their performance evaluations, highlighting the benefits of the hybrid security method in safeguarding several IoT services. The paper also reveals the significance of hybridization techniques in reinforcing IoT security and validates the ubiquitous use of AES and ECC algorithms in hybridization, owing to their computing speed and security resistance efficiency. ECC is specifically utilized for key generation in order to streamline operations and minimize complexity. Its advantage lies in its ability to achieve superior performance compared with other cryptographic techniques. By combining AES with ECC, significant improvements can be made in terms of data optimization and security. The result showed that the proposed hybrid AES-ECC algorithm had a faster encryption time and decryption time compared with the existing algorithms. The proposed hybrid algorithm offers a higher level of security due to its superior AV effects. Also, the result analysis proved that the

hybrid algorithm necessitates a moderate memory size and reduces power consumption when compared with existing algorithms. The performance analysis showed that the proposed hybrid cryptography is lightweight and provides robust data security in IoT. To enhance the productivity and efficiency of the system, manifold security layers can be added in the future to increase the security of the hybrid approach used in this research. By implementing additional security measures, the overall security of the system can be further strengthened.

References

1. Rani S, Kataria A, Sharma V, Ghosh S, Kumar V, Lee K, Choi C (2021) Threats and corrective measures for IoT security with observance of cybercrime. *Wirel Commun Mob Comput*
2. Litoussia M, Kannouf N, El Makkaoui K, Ezzatia A, Fartitchouz M (2020) IoT security: challenges and countermeasures. In: Proceedings of 7th international symposium on emerging information communication and networks (EICN 2020), 2–5 Nov 2020. Elsevier B.V. Available online at www.sciencedirect.com
3. Mrabet H, Belguith S, Alhomoud A, Jemai A (2020) A survey of IoT security based on a layered architecture of sensing and data analysis. *Sensors* 13:3625. <https://doi.org/10.3390/s20133625>
4. Waheed N, He X, Ikram M, Usman M, Hashmi SS, Usman M (2021) Security and privacy in IoT using machine learning and blockchain. *ACM J Comput Surv* 53(6):1–37
5. Tahsien SM, Hadis K, Spachos P (2020) Machine learning based solutions for security of Internet of Things (IoT): a survey. *J Netw Comput Appl* 161
6. Zhang J, Chen H, Gong L, Cao J, Gu Z (2019) The current research of IoT security. In: Proceedings of IEEE fourth international conference on data science in cyberspace (DSC)
7. Sadique KM, Rahmani R, Johannesson P (2018) Towards security on the internet of things: applications and challenges in technology. *Procedia Comput Sci* 141:199–206. In: 9th International conference on emerging ubiquitous systems and pervasive networks
8. Thakor VA, Razzaque MA, Akhandaker MRA (2021) Lightweight cryptography algorithms for resource-constrained IoT devices: a review, comparison, and research opportunities. *IEEE Access* 9. date of current version February 22, 2021
9. Dhanda SS, Singh B, Jindal P (2020) Lightweight cryptography: a solution to secure IoT. *J Wirel Pers Commun*. Springer Science + Business Media, LLC, part of Springer Nature, June 2020
10. Sliman L et al (2021) Towards an ultra-lightweight block cipher for Internet of Things. *J Inform Sec Appl* 61:102897
11. A new lightweight cryptographic algorithm for enhancing data security in cloud computing. *Glob Transit Proc* 2:91–99
12. Khashan OA (2020) Hybrid lightweight proxy re-encryption scheme for secure Fog-to-things environment. *IEEE Access*. date of current version April 21, 2020
13. Vahi A, Jafarali Jassbi S (2020) SEPAR: a new lightweight hybrid encryption algorithm with a novel design approach for IoT. *Wirel Pers Commun*. <https://doi.org/10.1007/s11277-020-07476-y>
14. Asare BT, Quist-Aphetsi K, Nana L (2019) A hybrid lightweight cryptographic scheme for securing node data based on the Feistel Cipher and MD5 hash algorithm in a local IoT network. In: Proceedings of international conference on mechatronics, remote sensing, information systems and industrial information technologies (ICMRSISIIT). <https://doi.org/10.1109/icmrssit46373>
15. Prakash V, Singh AV, Khatri K (2019) A new model of light weight hybrid cryptography for internet of things. In: Proceedings of the third international conference on electronics

- communication and aerospace technology IEEE conference (ICECA 2019). Record # 45616; IEEE Xplore ISBN 978-1-7281-0167-5. <https://doi.org/10.1109/iceca.2019.8821924>
- 16. Dahiya S, Bohra M (2017) Hybrid parallel partial model for robust & secure authentication in healthcare IoT environments. In: Proceedings of 4th IEEE Uttar Pradesh section international conference on electrical, computer and electronics (UPCON), GLA University, Mathura, 26–28 Oct 2017
 - 17. Iyer SC, Sedamkar RR, Gupta S (2016) A novel idea on multimedia encryption using hybrid crypto approach. *Procedia Comput Sci* 79:293–298. <https://doi.org/10.1016/j.procs.2016.03.038>
 - 18. Pourali A (2014) The presentation of an ideal safe SMS based model in mobile electronic commerce using encryption hybrid algorithms AES and ECC. In: Proceedings of international conference on E-commerce in developing countries: with focus on e-trust, April 2014. <https://doi.org/10.1109/ECDC.2014.6836761>
 - 19. Mahalle VS, Shahade AK (2014) Enhancing the data security in cloud by implementing hybrid (RSA & AES) encryption algorithm. In: Proceedings of the international conference on power, automation and communication (INPAC), Amravati, India, pp 146–149. <https://doi.org/10.1109/INPAC.2014.6981152>
 - 20. Yousefi A, Jameii SM (2017) Improving the security of internet of things using encryption algorithms. In: Proceedings of international conference on IoT and application (ICIOT), Nagapattinam, India, pp 1–5. <https://doi.org/10.1109/ICIOTA.2017.8073627>
 - 21. D'souza FJ, Panchal D (2017) Advanced encryption standard (AES) security enhancement using hybrid approach. In: Proceedings of international conference on computing, communication and automation (ICCCA), Greater Noida, India, pp 647–652. <https://doi.org/10.1109/ICCA.2017.8229881>
 - 22. Bhole D, Mote A, Patil R (2016) A new security protocol using hybrid cryptography algorithms. *Int J Comput Sci Eng* 4(2):18–22
 - 23. Soman VK, Natarajan V (2017) An enhanced hybrid data security algorithm for cloud. In: International conference on networks and advances in computational technologies, pp 416–419. <https://doi.org/10.1109/NETACT.2017.8076807>
 - 24. Maitra S, Yelamarthi K (2019) Rapidly deployable IoT architecture with data security: implementation and experimental evaluation. *Sensors (Switzerland)* 19(11). <https://doi.org/10.3390/s19112484>
 - 25. Stallings W (2017) Cryptography and Network Security
 - 26. Dhillon PK, Kalra S (2016) Elliptic curve cryptography for real time embedded systems in IoT networks. In: Proceedings of 5th international conference on wireless networks and embedded systems (WECON), Rajpura, India, pp 1–6. <https://doi.org/10.1109/WECON.2016.7993462>
 - 27. US National Security Agency. The case for elliptic curve cryptography. <http://www.nsa.gov/business/programs/ellipticcurve.shtml>
 - 28. Subedar Z, Araballi A (2020) Hybrid cryptography: performance analysis of various cryptographic combinations for secure communication. *Int J Math Sci Comput* 4:35–41
 - 29. Sood SK (2012) A combined approach to ensure data security in cloud computing. *J Netw Comput Appl* 35(6):1831–1838
 - 30. Mendonca SN (2018) Data security in cloud using AES. *Int J Eng Res Technol* 7(01). ISSN 2278-0181
 - 31. Rehman S et al (2021) Hybrid AES-ECC model for the security of data over cloud storage. *J Electron* 10:2673. <https://doi.org/10.3390/electronics10212673>

A Survey: Analysis of Existing Hybrid Cryptographic Techniques



Aman and Rajesh Kumar Aggarwal

Abstract Information of all kinds is widely available right now, especially thanks to the Internet. Security concerns have long drawn attention since the dissemination contains a lot of useful information. The development of cryptographic algorithms has significantly enhanced information security. Symmetric and asymmetric encryptions are the two main categories in cryptography. Even while symmetric encryption has a very rapid processing time and is effective in encrypting enormous volumes of data, its security is not as good as asymmetric. Symmetric algorithms that use the same pair of keys expose users to security problems. So, if the key can be secured, the security may be improved. This makes sense to use symmetric encryption for the message and asymmetric algorithm security for the key. This survey will examine information transmission security concerns and the approach of hybrid encryption algorithms that are extensively employed in the present time. In this paper, some common situations of hybrid encryption will be studied and analyzed. Surveying existing hybrid cryptographic techniques can offer several benefits, such as identifying best practices, evaluating strengths and weaknesses, assessing real-world applicability, and providing a foundation for future research.

Keywords Cryptography · Symmetric key algorithm · Asymmetric key algorithm · Hybrid encryption · DES · AES · RSA · ECC

1 Introduction

Today, the amount of data being downloaded and transferred has increased to unimaginable heights. Data security issues are becoming more prevalent. The attacker has additional opportunity to intercept data during transmission due to the large transfer

Aman (✉) · R. K. Aggarwal

National Institute of Technology, Kurukshetra, Haryana, India

e-mail: aman_32113207@nitkkr.ac.in

R. K. Aggarwal

e-mail: r_k_a@nitkkr.ac.in

volume. However, the loss of important data may have a significant negative impact on people, businesses, and governments. As a result, the safety of information has always been a concern. Data security needs effective solutions. The data encryption technique is one of the numerous current technologies that protect data extremely well [1].

Cryptography is a technique of secret writing, and it has been used since ancient times to keep information secret. To secure the information, in cryptography we use the encryption and decryption techniques, and these are the basic functions of the cryptography [2]. By encryption, we convert a plain text in the coded text which is called the cipher text. And by decryption, we convert cipher text in the real message which is plain text. By encryption and decryption, we ensure the confidentiality of the message which means that unauthorized people cannot read the message.

Data confidentiality, authentication, and integrity are the three major reasons security experts utilize cryptography. Since the primary objective of cryptography is to protect the confidentiality or privacy of data, messages are often encoded in order to mask their true meaning [3]. In order to maintain the integrity of data, it must be ensured that the message received is identical to the message sent, meaning it should not be changed or altered.

In cryptography, there are basically two types, one is called the symmetric key (SK) cryptography and the other is asymmetric key (AK) cryptography. In SK cryptography, there is only one secret key between the sender and receiver, whereas in AK cryptography, we use two separate keys known as public key and private key. By this, we call AK cryptography, public key cryptography also. In cryptography, key size is the important factor to judge the security of the algorithm.

In SK cryptographic algorithms, key size is less than the AK cryptographic algorithms. So, SK cryptographic algorithms are less secure than the AK cryptographic algorithms. In AK cryptographic algorithms, the computational time is larger than in SK cryptographic algorithms. For this reason, when there is a large amount of data, the AK cryptographic algorithms get complex [4]. Due to complex computation and larger key sizes, AK cryptographic algorithms are used only for key exchange and then SK cryptographic algorithms for encryption and decryption.

In SK cryptographic algorithms, AES is the most secure and most frequently used algorithm. Other SK algorithms are Twofish, Blowfish, RC4, 3DES, DES, etc. In AK algorithms, RSA is the most common and most secure algorithm. Other asymmetric algorithms are Diffie-Hellman, DSS, ECC, etc. But when we compare both of the encryption techniques, we can't say that particular is better. Asymmetric encryptions are more secure and complex to handle. But we can't neglect the performance factor and in this symmetric are better than the asymmetric. So, we need hybrid cryptography techniques to optimize our encryption process.

Hashing is also a form of cryptography. A string is transformed into a specific value of a fixed length using any hash function. Due to the fact that hashes reflect the original value as a brief, distinctive text, they are easier to search for and recover. Many encryption methods employ key asymmetries. A hash function is a function that performs hashing; as a result, it does not require conversion to return to its original value. MD2, MD4, and MD5 are examples of lossy hash functions, while

the widely used SHA algorithm generates lengthier, 60-bit message digests. This is also used in the hybrid cryptographic systems.

An effective symmetric cryptosystem and the convenience of any public key crypto algorithm are combined to form a hybrid cryptosystem in the field of cryptography. Hybrid encryption technique is to use two encryption algorithms to improve the performance or security or both, of individual algorithms. Any two independent cryptosystems, such as a data encapsulation method which is a symmetric key cryptosystem, and a key encapsulation mechanism which is a public key cryptosystem, can be combined to create a hybrid cryptosystem. The public and private keys of the hybrid cryptosystem, which is also a public key system, are the same as those used in the key encapsulation method. Basically, hybridization can be layered or cross-breed as shown in Fig. 1.

In layered hybrid encryption two symmetric algorithms are used in layered fashion. First, data is encrypted using an encryption algorithm and extract a cipher text. Then, cipher text passes through another encryption algorithm and gets another final cipher text. And follow a reverse technique to decrypt the final cipher text. In result, the text is more secure compared to single encryption. But this technique is not popular because it makes the encryption process heavier and therefore reduces the performance [5].

In cross-breed hybridization, there is the combination of the symmetric and asymmetric algorithm. Symmetric algorithm takes care of the performance and an asymmetric algorithm takes care of the security of the key which is used in the symmetric algorithm. By this technique our cryptosystem becomes secure and well optimized. And this technique is used practically.

There are many combinations of different algorithms like AES-RSA, DES-RSA, AES-ECC, etc. The most common hybrid algorithm is AES-RSA, because in this combination both the algorithms used are considered as most secure. Some other

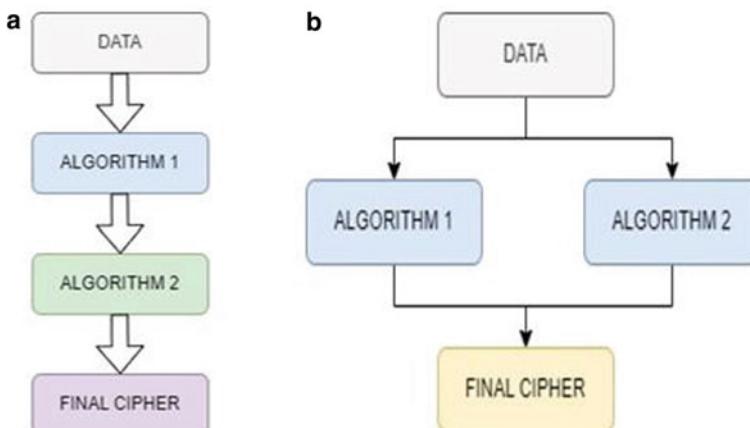


Fig. 1 a Layered hybridization. b Cross-breed hybridization

algorithms use two symmetric algorithms with one asymmetric algorithm to explore the different features of an encryption process.

In the next section, presently used different hybrid cryptographic techniques will be discussed and after that there will be compressions between different techniques and followed by conclusions and references.

2 Literature Survey

Bangar and Shinde [1] stated the characteristics of cloud computing and how to secure clouds using various cryptography methods. Asymmetric algorithms including Diffie-Hellman, digital signatures, and RSA, as well as symmetric algorithms like advanced encryption standard and data encryption standard, were implemented in this study along with hybrid systems. This research also used a variety of techniques to prevent hackers from attacking cloud data. The study also evaluated the scalability of various encryption techniques.

Khorsheed et al. [3] mentioned that it has become more important to guarantee the security of data shared between users and the cloud. To protect the sent information, a number of authentication and security mechanisms have been suggested. These methods work to protect the confidentiality, accuracy, and authenticity of data. This study proposes an encryption technique to protect data saved in the cloud, and for this, AES and RC5 algorithms have been used. This offers integrity and privacy of user identities while making the performance and security level more adaptable.

Sajay et al. [4] discussed the idea of data storage in the cloud. IaaS is used for cloud storage, and security is mentioned. The study included algorithms like HMAC and Advanced Encryption Standard as the best method for resolving cloud computing issues (AES). The major goal of this study was to protect a large volume of data in cloud systems and store encrypted data on a storage server to enable users to transmit data to the cloud without worrying about it being lost.

The model uses both symmetric and asymmetric cryptography techniques. The DES and RSA are used in this implementation to offer several levels of encryption and decryption at the transmitter and receiver sides, enhancing security of the cloud. In order to lessen security risks, this security architecture provides both cloud service providers and users with transparency [6].

Ghosh et al. [7] has put out a hybrid algorithm that combines RSA with Diffie-Hellman, with RSA handling message security and Diffie-Hellman serving as a secure key transfer agent. The original cipher text is changed before and after transmitting using the Diffie-Hellman keys, accordingly. The user can select different RSA key sizes depending on the level of secrecy desired. In order to ensure secrecy and boost security in Internet communications, a hybrid cryptography technique is put forth in this research. In order to prevent the technique from being CPU exhausting, the article also concentrates on how long the encryption and decryption processes take.

In this research [8], three new approaches are merged with existing encryption algorithms to create hybrid cryptography. Data is split into three portions when a user uploads it, with the first section using AES encryption, the second using DES encryption, and the last one using RSA encryption. The data is subsequently decrypted once more using RSA, DES, and AES using these keys. This strategy improves record security.

Abroshan [9] offers a cryptography solution that will increase security in cloud computing with very little performance impact. The elliptic curve-based approach is combined with an enhanced Blowfish algorithm in this solution. The data encrypted using Blowfish, and the key encrypted using the elliptic curve technique, which will improve security and performance. To further ensure data integrity, a digital signature mechanism is employed.

In order to establish a highly safe environment for data transfer, it is intended to integrate a number of secure methods. The RSA asymmetric method, the MD5 hashing technique, and the symmetric cryptographic algorithm AES will all be merged. These three methods enable us to guarantee the secrecy, authentication, and data integrity of three cryptography fundamentals [10].

In this work [11], present a novel hybrid cryptographic technique. Utilizing a mixture of two symmetric cryptographic approaches, the algorithm was created. IDES and DES used to accomplish these security primitives. Better security and integrity have been built into this new hybrid cryptographic technique according to the author.

The hybrid technique shown here combines a hash function, an asymmetric algorithm (ECC), and an asymmetric algorithm (AES) (SHA-256). This hybrid system aims to increase security by fusing symmetric encryption's effectiveness with asymmetric cryptography's capabilities. A mathematical method called SHA-256 is utilized in this instance to encode digital data. It serves to guarantee the integrity of the data [12].

Mahalle [13] provides a hybrid structure using RSA and AES to protect cloud data security. The most crucial aspect of cloud computing, security must be handled with utmost care. The following primary tasks are primarily the focus of this paper: (1) Securely uploading data to the cloud so that not even the administrator may access it. (2) Data that is securely downloaded while maintaining data integrity. (3) Use and distribution of the public, private, and secret keys necessary for encryption and decoding in an appropriate manner.

In this work [14], writers suggested a secure encryption method for cloud computing that makes use of completely homomorphic encryption and modern encryption standards. In the beginning, this article will outline the existing options for deploying encryption technology. Based on their advantages and disadvantages, the solutions will be examined and critiqued. Second, given the shortcomings of the present methods, a new algorithm has been developed that offers extensive detail on how encryption may shield users' confidentiality, privacy, and integrity.

In this study [15], a hybrid system of the Dynamic AES and Blowfish algorithms was presented. The s-box structure will be somewhat improved to make the AES more safe. The inverse of multiplication will be performed after the change. AES

may be more secure if a third party breaks the key. As a result, the client has access to all encrypted files that have been submitted for analysis. The advantages of a DAES/ Blowfish hybrid attack against unimaginable power are several. AES criterion uses a security key size of between 128,198 and 256 bits.

In WSNs, nodes are constrained in terms of energy usage, storage capacity, and processing speed. Existing algorithms have some shortcomings, such as Rijndael's high memory consumption and RC6's too complicated procedures. In this work, a chaotic block encryption technique is suggested to solve these issues. The proposed approach uses the traditional Feistel network topology since it is a popular block encryption structure with little resource usage. Additionally, RC5 generates round keys by utilizing a hybrid chaotic map [16].

The authors [17] describe the hybrid cryptography approach in full, including key generation, data encryption and decryption operations, and a safe key exchange system. They also do thorough simulations and performance evaluations to examine the suggested method's efficacy and efficiency. Based on the findings from experiments, the authors show that the hybrid cryptography strategy improves communication security in WSNs while minimizing computational cost. The suggested solution provides better resilience to different security assaults while also ensuring safe data transfer across sensor nodes.

3 Analysis of Techniques

The primary symmetric algorithms, including AES, DES, 3DES, Blowfish, and RC, were assessed and compared in a comparative analysis because these are the base algorithms for the hybrid systems mainly. Results indicate the limitations of different base algorithms.

DES is one of the first standard symmetric algorithms. The plaintext block size for DES is 64 bits, while the key length is 56 bits. 8 bits are used by DES as parity bits to identify errors. The 56 bit key length in DES raises serious security concerns. A parallel computing machine makes brute force attacks conceivable. Exploiting DES's features enables cryptanalysis. A conceivable method for a cryptanalytic assault is provided by the weak S-boxes. Attacks using linear cryptanalysis can be quite successful against DES. Due to the weak keys, it is vulnerable to a brute force assault. 3DES is a variation of DES that employs three keys, each of which is 56 bits long (168 bits total). A specific variant of meet-in-the-middle attacks is vulnerable against 3DES. Additionally, it is vulnerable to related key and differential attacks.

The Feistel-based block encryption method Blowfish employs a 64-bit block and keys with a range of 3–2448 bits. Blowfish's varied key lengths need extra processing time. The difficulty of a brute force assault is increased by the laborious sub-key generating procedure. Blowfish's dependability is compromised by the usage of a significant number of weak keys. The process's first four rounds are vulnerable to second-order differential assaults [18].

AES, Feistel block cipher method, has a 128-bit block size with a flexible key length of 128/192/256 bits. Different strategies for breaking AES were explored, including square attacks, key attacks, and differential attacks, but none of them succeeded. Several nonlinear changes in key-schedule algorithms are weak points that can be exploited by the paired boomerang and rectangle assault alongside related key differentials to crack various reduced versions of AES. 192-bit, 9-round AES can be cracked with 256 distinct related keys.

Public key cryptography, sometimes referred to as asymmetric cryptography, uses the RSA algorithm. The key's asymmetry is based on the factoring of the product of two enormous prime integers. With the help of the private key, messages that have been encrypted with the public key may be unlocked quickly. To create the public and private keys, modulus and exponent procedures are used. The technique is eventually slowed down by it since it requires a lot of memory and takes a huge time to encrypt data.

Based on Diffie-Hellman key exchange, the ElGamal encryption algorithm is an asymmetric encryption technique. It may be defined on any cyclic group G , and the discrete logarithm problem on G determines how secure it is. Key creation, encryption, and decryption are the three components of the ElGamal encryption algorithm [19]. ElGamal encryption algorithm is a slow asymmetric cryptography scheme. Typically, hybrid systems employ this algorithm. It would be quicker and more secure to use the ElGamal technique to encrypt the secret key and symmetric methods to encrypt the message.

An elliptic curve-based public key encryption scheme is called ECC. The fundamental benefit of ECC is that it may employ fewer keys in some circumstances than other techniques, like RSA, while still offering equivalent or greater degrees of security. One benefit is that it has high security. This technique is quick at encrypting data and decrypting it, and it uses less bandwidth and storage. However, compared to RSA, the computation is far more complicated, making it difficult to apply, even though it is significantly safer.

Compared to the ElGamal method, the RSA algorithm encrypts and decrypts data more quickly. ElGamal is slower than the RSA algorithm. Due to the fact that ElGamal uses a complex formula to create discrete logarithms, it will be more difficult to crack the ElGamal technique than the RSA algorithm. Although ECC is regarded to be as durable as RSA and ElGamal cryptosystems, it uses shorter keys to maintain the same degree of security.

Messages were encrypted using the MD5 cryptographic hash method following a safe key exchange. Modern cryptographic systems have implemented a method to ensure that data packets are safeguarded against change or manipulation by third party nodes that may have compromised the system as a result of various recent successful attacks on conventional SK ciphers. The MD5 is a hashing method that takes the cipher text and a key as input and outputs a checksum. Using the sender's private key and the checksum, the cipher text is then digitally signed off [17].

The SHA1 algorithm is a crypt-formatted hash function used in the field of cryptography and crypt analytics that takes a smaller input and generates a string that is 160 bits, commonly known as a 20-byte hash value, long. As a result, the security

is supplied by the one-way nature of the function that makes up the majority of the SHA algorithm's core. Pre-image resistance is crucial to fending off brute force attacks from a collection of large, robust machines. SHA-256 is a member of the SHA 2 family of algorithms, which was created as a replacement for the SHA1 family, whose resistance to brute force assaults was gradually eroding. SHA2 outperforms MD in terms of robustness but MD yields results relatively faster (Table 1).

4 Conclusion and Future Scope

The goal of this analysis of the literature is to evaluate strengths and weaknesses of existing hybrid cryptographic techniques. By examining the strengths and weaknesses of existing hybrid cryptographic techniques, researchers can identify areas for improvement and develop more robust and secure hybrid cryptographic solutions. This paper also provides taxonomy for using hybrid cryptography models in various sectors with various security requirements.

Surveying existing hybrid cryptographic techniques can offer several benefits, such as identifying best practices, evaluating strengths and weaknesses, assessing real-world applicability, providing a foundation for future research, and informing decision-making. Overall, surveying existing hybrid cryptographic techniques can help researchers, practitioners, and decision-makers make more informed decisions about the selection and implementation of hybrid cryptographic techniques in various applications.

Hybrid cryptography has a significant role to play in the future of cryptography, as it provides an efficient and secure way to encrypt data and protect sensitive information. Here are some potential areas of future scope for hybrid cryptography: post-quantum cryptography, Internet of Things (IoT), blockchain technology, cloud computing, multi-party computation. Overall, hybrid cryptography has a bright future as it provides a flexible and powerful toolset for securing data and communications in various applications. As new technologies and applications emerge, hybrid cryptography will continue to play a critical role in securing our digital lives.

Table 1 Analysis of existing techniques

Sr. No	Year []	Algorithms used	Advantages	Limitations
1	2014 [1]	AES, DES, RSA, Diffie-Hellman	Using two symmetric algorithm securities gets enhanced then a single algorithm	Both AES and DES are used. So, the bulkiness of the encryption increases. And also no integrity of the data ensured
2	2014 [2]	AES, MD5	Most secure symmetric algorithms are used and also care about integrity	Authentication of the data is not ensured
3	2014 [11]	DES, IDEA	Using layered hybrid cryptography, security of data improved	Authentication and integrity both are missing. And also the system is bulky due to two heavy algorithms
4	2015 [3]	AES, RC5	Confidentiality and authentication ensured	Not ensure the integrity of data
5	2015 [20]	AES, ECC, RSA, MD5	Most secure symmetric algorithm used with other good asymmetric algorithms	Complexity increases and then the performance of the process will reduce
6	2016 [4]	AES, HMAC	Good combination of secured symmetric and hashing algorithm	Authentication is ignored which should be ensured using asymmetric algorithms
7	2016 [21]	AES, BLOWFISH, RC6, BREA	Creaking of this system will be very hard because many algorithms are used in this	System is very heavy. So the performance is significantly reduced
8	2016 [6]	DES, RSA	Applied two most common algorithms which are less complex to use	Not so confidential because DES is a very old algorithm and not integrity taken into consideration
9	2017 [22]	AES, RSA	Most secure symmetric algorithm used and paired with a good asymmetric algorithm	Only confidentiality and authentication take into account not integrity
10	2017 [10]	AES, RSA, MD5	Good combination of some very popular algorithms	Performance of the algorithm can be improved using other algorithms
11	2019 [23]	Homographic, Blowfish	A new algorithm is used which is less exposed to attackers	Performance and integrity can be improved using other algorithms
12	2020 [24]	ECC, BLOWFISH	Most secure asymmetric algorithm used	Confidentiality is compromised and integrity is not implemented
13	2021 [5]	AES, DES, RSA	Layered hybridization increases the security of the system	Two bulky algorithms reduce the performance and integrity not ensured

(continued)

Table 1 (continued)

Sr. No	Year	Algorithms used	Advantages	Limitations
14	2022 [12]	AES, ECC, SHA-256	Good combination for all aspects of security	Performance of the algorithm can be improved using other hybrid techniques
15	2023 [25]	AES, Homomorphic	Most secure symmetric algorithm used and paired with a good asymmetric algorithm	Only confidentiality and authentication take into account not integrity

References

- Bangar A, Shinde S (2014) Study and comparison of cryptographic methods for cloud security. *Int J Comput Sci Eng Inf Technol Res* 4(2):205–213
- Sidhu A, Mahajan R (2014) Enhancing security in cloud computing structure by hybrid encryption. *Int J Recent Sci Res* 5(1):128–132
- Khorsheed NK, Khorsheed OK, Rashad MZ, Hamza TT (2015) Proposed encryption technique for cloud applications. *Int J Sci Eng Res* 6(9):693–698
- Sajay KR, Babu SS, Vijayalakshmi Y (2019) Enhancing the security of cloud data using hybrid encryption algorithms. *J Ambient Intell Humanized Comput* 1–10
- Bharathi P, Annam G, Kandi JB, Duggana VK, Anjali T (2021) Secure file storage using hybrid cryptography. In: 2021 6th international conference on communication and electronics systems (ICCES), IEEE, pp 1–6
- Kumar S, Karnani G, Gaur MS, Mishra A (2021) Cloud security using hybrid cryptography algorithms. In: 2021 2nd international conference on intelligent engineering and management (ICIEIM), IEEE, pp 599–604
- Ghosh SK, Rana S, Pansari A, Hazra J, Biswas S (2021) Hybrid cryptography algorithm for secure and low cost communication. In: 2020 international conference on computer science, engineering and applications (ICCSEA), IEEE, pp 1–5
- Gokulraj S, Ananthi P, Baby R, Janani E (2021) Secure file storage using hybrid cryptography. Available at SSRN 3802668
- Abroshan H (2021) A hybrid encryption solution to improve cloud computing security using symmetric and asymmetric cryptography algorithms. *Int J Adv Comput Sci Appl* 12(6):31–37
- Harini M, Gowri KP, Pavithra C, Selvarani MP (2017) A novel security mechanism using hybrid cryptography algorithms. In: 2017 IEEE international conference on electrical, instrumentation and communication engineering (ICEICE), IEEE, pp 1–4
- Jain M, Agrawal A (2014) Implementation of hybrid cryptography algorithm. *Int J Core Eng Manage (IJCEM)* 1(3):126–142
- William P, Choubey A, Chhabra GS, Bhattacharya R, Vengatesan K, Choubey S (2022) Assessment of hybrid cryptographic algorithm for secure sharing of textual and pictorial content. In: 2022 international conference on electronics and renewable systems (ICEARS), IEEE, pp 918–922
- Mahalle VS, Shahade AK (2014) Enhancing the data security in Cloud by implementing hybrid (RSA & AES) encryption algorithm. In: 2014 international conference on power, automation and communication (INPAC), IEEE, pp 146–149
- Olumide A, Alsadoon A, Prasad PWC, Pham L (2015) A hybrid encryption model for secure cloud computing. In: 2015 13th international conference on ICT and knowledge engineering (ICT & Knowledge Engineering 2015), IEEE, pp 24–32
- Olanrewaju RF, Abdullah K, Darwis H (2018) Enhancing cloud data security using hybrid of advanced encryption standard and blowfish encryption algorithms. In: 2018 2nd East Indonesia Conference on Computer and Information Technology (EICoCIT), IEEE, pp 18–23

16. Chauhan A, Gupta J (2017) A novel technique of cloud security based on hybrid encryption by Blowfish and MD5. In: 2017 4th international conference on signal processing, computing and control (ISPCC), IEEE, pp 349–355
17. Yang H, Li Y (2023) A hybrid cryptography method for secure communication in wireless sensor networks. *Wireless Pers Commun* 132(2):1201–1216
18. Luo Y, Yao L, Liu J, Zhang D, Cao L (2019) A block cryptographic algorithm for wireless sensor networks based on hybrid chaotic map. In: 2019 IEEE 21st international conference on high performance computing and communications; IEEE 17th international conference on smart city; IEEE 5th international conference on data science and systems, IEEE, pp 2790–2797
19. Yue T, Wang , Zhu ZX (2019) Hybrid encryption algorithm based on wireless sensor networks. In: 2019 IEEE international conference on mechatronics and automation (ICMA), IEEE, pp 690–694
20. Rizk R, Alkady Y (2015) Two-phase hybrid cryptography algorithm for wireless sensor networks. *J Electrical Syst Inf Technol* 2(3):296–313
21. Maitri PV, Verma A (2016) Secure file storage in cloud computing using hybrid cryptography algorithm. In: 2016 international conference on wireless communications, signal processing and networking (WiSPNET), IEEE, pp 1635–1638
22. Shende V, Kulkarni M (2017) FPGA based hardware implementation of hybrid cryptographic algorithm for encryption and decryption. In: 2017 international conference on electrical, electronics, communication, computer, and optimization techniques (ICEECCOT), IEEE, pp 416–419
23. Sajay KR, Babu SS, Vijayalakshmi Y (2019) Enhancing the security of cloud data using hybrid encryption algorithm. *J Ambient Intell Humanized Comput* 1–10
24. Chinnasamy P, Padmavathi S, Swathy R, Rakesh S (2021) Efficient data security using hybrid cryptography on cloud computing. In: Inventive communication and computational technologies: proceedings of ICICCT 2020. Springer Singapore, pp 537–547
25. Singh S, Sharma S (2023) Secure and efficient hybrid cryptography using homomorphic encryption. *Wireless Pers Commun* 131(2):935–950

Identity-Based Designated Verifier Proxy Signature Scheme and Its Application to Health Care



Vandani Verma and Yash Sharma

Abstract Identity-based designated proxy signature is a type of digital signature that permits a designated proxy signer to affix his signature to a message without revealing the signer's private key. IBPS has gained substantial interest in recent years because of its applications in secure communication, e-commerce, electronic voting, healthcare, etc. The security of these schemes has been extensively studied in the literature to protect against various attacks such as forgery attacks, existential forgery attacks, and insider attacks in the literature. In this paper we present DVPS signature scheme that is unforgeable with respect to type 5 challenger attack and type 6 challenger attack and discuss the applications of DVPS schemes in healthcare.

Keywords Identity-based cryptography · Designated verifier · Proxy signature

1 Introduction

Mambo et al. [1] in 1996 created the notion of a proxy signature (PS). PS scheme lets in an entity referred to as unique signer to envoy his signing functionality to any other entity, referred to as proxy signer. Since its inception, the PS schemes had been advised to be used in lots of purposes. When privacy issues come into consideration, the public verifiable property of digital signatures might not be appropriate for some purposes. A different type of digital signature technique known as the designated verifier signature (DVS) scheme was put out by Jakobsson et al. [2] in the same year. Under such a method, only a designated verifier can be persuaded of the received signature's legitimacy in relation with a particular signer. Because he is also capable of constructing a computationally identical record meant for himself, the authorized verifier cannot transmit the signature to a third party, which is known as non-transferability. DVS [3] in instances like electronic voting, where the non-repudiation quality is not sought, systems are acceptable. A specific sort of digital signature system called a designated verifier proxy signature scheme (DVPS) allows

V. Verma (✉) · Y. Sharma

Department of Mathematics, Amity Institute of Applied Sciences, Amity University, Noida, India
e-mail: vandaniverma@yahoo.com

a PS to sign on behalf of the original signer (OS), and the signature can be verified only by a designated verifier specified by the original signer. In DVPSS [4–8], the original signer delegates his or her signing authority to a PS, who can sign messages on behalf of the OS. OS specifies the chosen verifier who is permitted to verify the signature, while the PS is permitted to sign only specific messages. Although the OS's identity cannot be ascertained, the chosen verifier can confirm that the signature is legitimate. DVPSS has many practical applications, such as in e-voting, where a voter may delegate his or her voting authority to a proxy who can cast the vote on the voter's behalf. In this case, the voter may want to keep his or her identity secret, while still ensuring that the vote is counted correctly. There are various types of DVPSS schemes, such as the bilinear pairing-based DVPSS and the elliptic curve-based DVPSS. These schemes use different mathematical techniques to achieve the desired properties of confidentiality, integrity, non-repudiation, and verifiability. Overall, DVPSS provides a useful mechanism for delegating signing authority while maintaining confidentiality and ensuring that the signature is verifiable only by an appointed verifier. There are several variations of DVPSS [8–10], including ID-based DVPSS, threshold DVPSS, and blind DVPSS. Each variation has its own strengths and flaws and possibly will be more appropriate for other applications depending on the precise constraints.

In 2006, Yang et al.'s [4] paper proposed an efficient identity-based DVS scheme with public verification, which reduces the computational overhead of verification that eradicates the necessity for a reliable third party and is apt for applications with limited resources. Cao et al. [11] proposed an efficient and secure identity-based DVPSS using bilinear pairings, which achieves the same security level as traditional signature schemes but with lower computational overhead and offers enhanced security features such as message confidentiality and non-repudiation. Chen et al. [12] proposed an efficient identity-based DVPSS that is suitable for use in large-scale distributed systems. The scheme allows a signer to delegate the signing authority to a DV, who can verify the authenticity of a signature on behalf of the signer. Gong et al. [13] proposed an efficient and expressive identity-based DVPSS that does not require bilinear pairings, making it suitable for use in environments where pairings are not available. Yang et al. [14] provided a formal security analysis of an identity-based DVPSS, using automated theorem proving and formal verification techniques to rigorously prove the scheme's security properties. Yu et al. [15] proposed an efficient and verifiable identity-based DVPSS with enhanced security features, including resistance to key leakage attacks and efficient verification. Yang et al. [16] proposed a short and efficient identity-based DVPSS with constant-size signatures, which reduces the computational overhead and storage requirements of the scheme. Han et al. [17] proposed an enhanced security identity-based DVPSS with tighter security reduction, which improves the security level of the scheme while reducing the computational overhead. Xiong et al. [18] proposed a scheme with strict security reduction, which delivers a greater security level than prior schemes and is appropriate for use in high-security applications. Since then several DVPSS [19–22] signatures are proposed explaining the applications in various fields like health, e-voting, cloud computing, and many more. In this paper, we propose DVPSS scheme that is unforgeable with respect to type 5 attack and type 6 attack.

2 Security Models

This section discusses the security notions on which we establish the security of proposed scheme and compose the proxy signature that can withstand these attacks. There exists total six adversaries rendering to their capability to attack, namely.

- *Type-1 Challenger (CL_1):* CL_1 challenger consists of the public keys (Q_A , Q_B , Q_C) of user A, B, and C, respectively.
- *Type-2 Challenger (CL_2):* CL_2 challenger consists of public keys as well as secret key S_B of user B.
- *Type-3 Challenger (CL_3):* CL_3 challenger consists of public keys and also consists of the secret key S_A of user A.
- *Type-4 Challenger (CL_4):* CL_4 challenger consist of public keys and also consists of the secret key S_C of user C.
- *Type 5 Challenger (CL_5):* CL_5 challenger consists of public keys and private keys of user A, B, and C, respectively.
- *Type 6 Challenger (CL_6):* CL_6 challenger consists of public keys of user A, B, and C, respectively and consists of secret keys of user A and C, respectively.

We discuss unforgeability with respect to type 5 attack and type 6 attack as follows:

2.1 Unforgeability Regarding Type 5

The CL_5 architecture thus contains the public keys (Q_A , Q_B , Q_C) of the signer and verifier as well as the secret keys (S_B , S_C) of the proxy signer and verifier.

- *Setup:* It accepts public parameters as input and returns the security parameter.
- *Extract:* To challenge CL_5 , it needs the system settings, signer, and verifier public keys, as well as signer and verifier proxy private keys.
- *Delegation Generation:* The challenger CL_5 collects the allocated information and requests a warrant of their choosing.
- *Proxy Signature:* Challenger CL_5 requests a message, warrant of his/her choice, and obtains a signature.
- *Proxy Signature Verification:* If the signature is legitimate, the message is returned; if not, it is invalid.

2.2 Unforgeability Regarding Type 6

Following the creation of CL_6 , it provides system parameters, the public keys (Q_A , Q_B , Q_C) of the OS and the verifier as well as their corresponding secret keys (S_A , S_C), but it is unable to get the proxy signing key.

- *Setup*: It produces the security parameter after accepting the public parameters as input.
- *Extract*: It provides challenger CL_6 with system parameters, public keys, and secret keys, and since it has complete control over the original signer and verifier, it can get delegation on its own.
- *Proxy Signature*: It is requested by challenger on message and warrant of his/her choice and obtain signature.
- *Proxy Signature Verification*: When a challenger requests it for a signature, it returns the appropriate message if the output is legitimate and reports it as invalid otherwise.

3 Proposed IBDVPS

The identity-based DVPSS (IB-DVPSS) and its security against type 5 and type 6 challenger attacks are proposed in the following section:

- *Setup*: Given security parameter ‘k’, it generates the public parameters. System parameters are: $(G_1, G_2, H_1, H_2, H_3, Q_A, Q_B, Q_C, e)$, where
 - $e: G_1 * G_1 \rightarrow G_2$,
 - $H_1: \{0, 1\}^* \rightarrow G_1$,
 - $H_2: \{0, 1\}^* \rightarrow Zq^*$, and
 - $H_3: \{0, 1\}^* \rightarrow \{0, 1\}^1$.
- *Key Generation*: The public key is generated at this step for user U with identity ID_U , $Q_U = H_1(ID_U)$ and secret key $S_U = sQ_U$.
- *Delegation by Original Signer*: Original signer computes $\sigma_1 = H_2(w).S_A$ on warrant w and sends σ_1 to PS.
- *Delegation Verification*: Proxy signer checks the validity of the signature as

$$e(\sigma_1, P) = e(Q_A, P_{\text{pub}})^{H_2(w)}, \quad (1)$$

if equation (1) is found valid, proxy signer computes proxy key as $S_{AB} = \sigma_1 + H_2(w).S_B$.

- *Designated Verifier Proxy Signature*: Proxy signer computes.
 - $U_1 = H_2(m) \parallel H_2(w)$, $U_2 = H_3(u_1)$.
 - Randomly chooses $t \in Zq^*$
 - then compute
 - $U_3 = H_3(w, e(S_{AB}, tQ_c))$,
 - $U_4 = U_2 \oplus U_3$,
 - $U_5 = (t - U_4).S_{AB}$.
- Also, computes $\sigma_2 = e(U_5, Q_C)$, then displays the signature as (σ_2, U_4, w, U_2) .

- *Designated Verifier Proxy Signature Verification:* To verify the proxy signature, designated verifier computes

$$U'_2 = U_4 \oplus H_3(w \cdot \sigma_2 \cdot e(U_4 S_C H_2(w), Q_A + Q_B)), \quad (2)$$

checks if $H_3 = U_2$.

If true accepts the signature otherwise rejects.

- *Correctness:* Correctness of Eq. (2) is as follows:

$$\begin{aligned} & \sigma_2 \cdot e(U_4 S_C H_2(w), Q_A + Q_B) \\ &= e(U_5, Q_C) \cdot e(U_4, sQ_C \cdot H_2(w), Q_A + Q_B) \\ &= e((t-U_4)S_{AB}, Q_C) \cdot e(U_4 \cdot Q_C, H_2(w), sQ_A + H_2(w)sQ_B) \\ &= e(S_{AB}, tQ_C - U_4 Q_C) \cdot e(S_{AB}, U_4 Q_C) \\ &= e(S_{AB}, tQ_C - U_4 Q_C + U_4 Q_C) \\ &= e(S_{AB}, tQ_C). \end{aligned}$$

Hence,

$$U_3 = H(w \cdot \sigma_2 \cdot e(U_4 S_C H_2(w), Q_A + Q_B)),$$

$$U'_2 = U_4 \oplus U_3,$$

and now $U_2 = H_3(U'_2)$.

- *Simulation:* The verifier simulates DVPS on a message string $m \in \{0,1\}^{k^2}$.

Choose randomly $r' \in Zq^*$ and computes

$$U^{*3}' = H(w, e(r' \cdot S_C \cdot H_0(w), Q_A + Q_B)), \text{ then computes}$$

$$U^{*4} = U^{*2} \oplus U^{*3}.$$

$$U^{*5} = (t - U^{*4})S_C H_0(w) \text{ computes}$$

$\sigma' = e(U^{*5}, Q_A + Q_B)$, then displays (σ', U^{*4}, w) as DVPS.

So, the simulation's verification matches that of the original DVPS.

- *Security Analysis:* The security of IBDVPS has attracted a lot of interest in literature. To formalize the security needs of DVPS it is studied for forgery attacks, existential forgery attacks, and insider attacks and several security models have been developed. Our scheme presents precise security in unforgeability against type 5 and type 6 challenger:

Unforgeability against Type 5 Challenger: When one randomly selects $a_1, a_2 \in Zq^*$ and sets $Q_A = bP$ as OS's public key (User A), $Q_B = c_1$ as PS's public key (User B) and $Q_C = c_2P$ as verifier's public key (User C) and then sends $(Q_A, Q_B, Q_C, a_1, a_2)$ to A_5 . Hence, our system is protected from this threat.

Unforgeability against Type 6 Challenger: When one randomly chooses $a_1, a_2 \in Zq^*$ and executes setup to get system parameters then it sets public key $Q_A = a_1P$ of User A (original signer), public key $Q_B = bP$ (proxy signer) of User B, and public key $Q_C = a_1P$ of User C (verifier) and gives to A_6 .

- *Computational Comparison:* We note that the proposed method must be implemented with the hashing, multiplication, pairing evaluation, exponentiation, and taking the inverse operations. These operations are compared with [3, 5, 23, 24] outputs the following table:

Schemes	Hash	Multiplication	Pairing	Exponential	Inverse
[23]	4	10	9	0	2
[3]	6	9	6	3	2
[24]	9	17	5	3	1
[5]	9	16	6	2	1
Proposed	8	4	5	1	0

We may deduce from the table that our strategy is more efficient than [3, 5, 23, 24].

4 Applications

Identity-based designated verifier proxy signature (IBDVPS) has many potential applications in the healthcare industry. Here are some examples:

- *Electronic Health Records (EHRs):* IBDVPS can be used to sign and verify EHRs. Authorized healthcare practitioners can use their private key to sign patient records, and delegates can sign on their behalf while the healthcare provider is unavailable. This can help to speed up the process of validating the validity of patient information while also lowering the risk of fraud or unauthorized access.
- *Prescription Signing:* IBDVPS can be used to sign electronic prescriptions. A healthcare professional can use their private key to sign prescriptions, and the pharmacist can validate the signature using the physician's public key. This can aid in the prevention of prescription forgeries and guarantee that the drug is prescribed by an authentic healthcare expert.
- *Medical Device Authentication:* IBDVPS can be used to verify the authenticity of medical equipment such as implanted devices and medical monitoring. A private key can be used to sign the device, and proxies can sign on behalf of the manufacturer or authorized service providers. This can assist in confirming the device's authenticity and prevent unauthorized changes or repairs.
- *Telemedicine:* Telemedicine consultations may be signed and verified using IBDVPS. A healthcare professional can use their private key to sign the consultation, and a patient can use the provider's public key to validate the signature. This can assist in ensuring the legitimacy of the consultation and limit the possibility of fraudulent or unauthorized consultations.

5 Conclusion

Designated verifier has various applications and one of them being healthcare sector where it ensures the security and privacy of patient information. The existing research on DVPS has proposed several efficient and secure schemes and has explored various applications and security models. The paper proposed IBDVPS that is unforgeable against type 5 challenger and unforgeable against type 6 challenger attacks that is compared on the basis of computational and security aspects to the existing schemes [3, 4, 23, 24]. We also discussed its applications to EHR, medical device authentication, etc.

References

1. Jakobsson M, Sako K, Impalazzo KR (1996) Designated verifier proofs and their applications. *Eurocrypt 1996, LNCS #1070*. Springer, Heidelberg, pp 142–154
2. Mambo M, Usuda K, Okamoto E (1996) Proxy signatures for delegating signing operation (revisited). In: *Proceedings of 3rd ACM conference on computer and communication security (CCS)*, pp 48–57
3. Saeednia S, Kreme S, Markotwich O (2003) An efficient strong designated verifier signature scheme. *ICICS 2003, LNCS #2971*. Springer, Heidelberg, pp 40–54
4. Yang Y, Wong DS, Guan C (2006) Efficient identity-based designated verifier signature scheme with public verification. *J Comput Sci Technol* 21(2):193–199
5. Lal S, Verma V (2006) Identity based strong designated verifier proxy signature schemes. *Cryptography eprint Archive Report 2006*. Available at <http://eprint.iacr.org/2006/394.pdf>
6. Verma V, Malhotra K (2021) A new secure quantum signature masked authentication scheme. *Wirel Pers Commun* 120(2): 1659–1674 | Journal article <https://doi.org/10.1007/s11277-021-08527-8>
7. Lal S, Verma V (2007) Some identity based strong bi-designated verifier signature schemes. *Cryptography eprint Archive Report 2007*. Available at <http://eprint.iacr.org/2007/193.pdf>
8. Lal S, Verma V (2009) An identity based strong bi-designated verifier (t, n) threshold proxy signature scheme. [arXiv:0806.1377v1](https://arxiv.org/abs/0806.1377v1)
9. Lal S, Verma V (2009) Identity based strong designated verifier parallel multi proxy signature scheme. [arXiv:0904.3420v1](https://arxiv.org/abs/0904.3420v1)
10. Lal S, Verma V (2009) Some proxy signature and designated verifier signature schemes over braid groups. [arXiv:0904.3422v1](https://arxiv.org/abs/0904.3422v1)
11. Ma J, Cao Z (2007) Efficient and secure identity-based designated verifier proxy signature scheme from bilinear pairings. *J Syst Softw* 80(1):102–111
12. Chen K, Cao Z (2008) An efficient identity-based designated verifier proxy signature scheme. *Inf Sci* 178(23):4536–4542
13. Gong Z, Chen L, Ma J (2010) Efficient and expressive identity-based designated verifier proxy signature scheme without pairings. *J Syst Softw* 83(10):1934–1941
14. Yang C, Guo F, Xu J (2011) Formal analysis and security proof of an identity-based designated verifier proxy signature scheme. *J Netw Comput Appl* 34(1):274–281
15. Yu H, Ma J, Liu J (2014) Efficient and verifiable identity-based designated verifier proxy signature scheme with enhanced security. *J Netw Comput Appl* 41:287–296
16. Cui Y, Zhang J, Wu Q (2016) A short and efficient identity-based designated verifier proxy signature scheme with constant-size signatures. *IEEE Trans Inf Forensics Secur* 11(5):1015–1023

17. Han S, Liu J, Xiong H (2017) An enhanced security identity-based designated verifier proxy signature scheme with tighter security reduction. *Int J Commun Syst* 30(4):e3174
18. Hu C, Liu Z, Xiong H (2019) A new identity-based designated verifier proxy signature scheme with tight security reduction. In: Security and communication networks, pp 1–10
19. Verma V, Gupta D (2016) An efficient signcryption algorithm using bilinear mapping. In: Proceedings of the 10th INDIACOM; 2016 3rd international conference on computing for sustainable global development, INDIACOM 2016, pp 680–682, 7724351
20. Mishra P, Renuka, Verma V (2020) Identity based broadcast encryption scheme with shorter decryption keys for open networks. *Wirel Pers Commun* 115(2): 961–969, <https://doi.org/10.1007/s11277-020-07606-6>
21. Mishra P, Verma V (2020) A proficient identity-based signature scheme with designated verifier for e-voting. *J Crit Rev* 7(7):644–647
22. Cui S, Wen F (2011) A new ID-based designated verifier proxy multi-signature scheme. *Int J Comput Theor Eng* 3(2). ISSN: 1793-820
23. Kumar KP, Shailaja G, Saxena A (2006) Identity based strong designated verifier signature scheme. Cryptography eprint Archive Report 2006/134. Available at <http://eprint.iacr.org/2006/134.pdf>
24. Wang G (2004) Designated verifier proxy signature for e-commerce. In: IEEE international conferences on multimedia and expo (ICME 2004) CD-ROM, ISBN-0-7803-8604-3, Taipei, Taiwan, pp 27–30

Benefits and Challenges of Integrating IIoT in Smart Energy Systems



Saumya and Shobhita Khatri

Abstract The Internet of Things (IoT) is revolutionizing the techniques and procedures of enterprises by enabling the amalgamation of heterogeneous devices and systems to generate a seamless and astute environment for their applications and operations. The assimilation of IoT in the energy sector, falling under the sphere of Industrial Internet of Things (IIoT), is highly auspicious since it has the potential to engender intelligent energy systems that are characterized by upgraded range of machines and devices within a network leading to heightened accuracy, efficiency, dependability, and sustainability. Nonetheless, the integration of IIoT in smart energy systems is accompanied by an added layer of security quandaries that necessitate meticulous redressal to ensure their secure and safe functioning. With their vast scope and endless functional possibilities that prove their utility, IIoT applications in the domain of energy must seek approval after careful consideration of their gains as well as the potential threats they entail. In this paper, an effort has been made to provide a brief overview of the benefits and security challenges of integrating IIoT in smart energy systems.

Keywords Internet of Things (IoT) · Industrial Internet of Things (IIoT) · Smart energy systems

1 Introduction

The IIoT pertains to the application of Internet-enabled mechanisms within industrial domains, such as factories and power plants. The amalgamation of these devices with energy management systems to amplify the efficiency and stability of the energy grid

Saumya · S. Khatri (✉)

IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India
e-mail: shobhita208btcse21@igdtuw.ac.in

Saumya
e-mail: sauyma095btcse20@igdtuw.ac.in

[1] is known as intelligent energy systems. Intelligent energy systems, which encompass the IoT (IoT), endeavor to fabricate a more proficient, sustainable, and cost-effective energy system through the utilization of intelligent devices and advanced technology [2]. This technology provides real-time monitoring and regulation of energy generation, distribution, and utilization, as well as a comprehensive insight into the functioning of the energy system. Consequently, the archetypal energy system is undergoing a metamorphosis into an astute energy system. Smart metering is a paramount application of IoT in intelligent energy systems. Smart meters collate real-time data on energy utilization, enabling energy corporations to attain a superior understanding of consumer behavior and to improve energy billing accuracy. Smart meters also empower users to monitor their energy consumption, thereby assisting them in identifying areas where they can reduce their energy utilization and save money. The advent of the IoT has engendered the capability for the formation of complex and interlinked networks of both physical objects and digital devices, which can be leveraged as an instrument for supervising and controlling the processes and operations comprising these networks. The integration of the IoT entails the deployment of the tools and techniques that constitute the essence of IoT within the industries of the energy sector to mitigate energy dissipation, augmenting the implementation of sustainable energy sources [3] and alleviating the strain placed upon non-renewable energy sources such as fossil fuels. This integration endeavors to produce efficient and efficacious solutions to ensure sustainability within the realm of energy. In the subsequent sections, we shall delve into the advantages of implementing the IIoT infrastructure in smart energy systems and the hindrances encountered by IIoT in promoting and enabling its applications on an enlarged scale.

1.1 *Background and Motivation*

An indispensable manifestation of the IoT within the energy sector is demand response, which entails the modulation of energy utilization in response to fluctuations in energy supply and demand. Energy purveyors may exploit IoT technology to monitor energy utilization in real-time and adapt to changes in demand by adjusting energy generation, conveyance, and utilization [4]. This facilitates maintaining an equilibrated energy system and precluding instances of blackout and brownout. The functioning of sustainable energy sources, including but not limited to photovoltaic panels, wind turbines, and hydropower plants, can be scrutinized via the utilization of IoT mechanisms. This information may be employed to enhance energy output and ensure that renewable energy sources are functioning at peak efficacy. Energy storage systems may be managed through IoT technologies. For instance, batteries play a crucial role in balancing energy supply and demand within the energy system. IoT technology affords real-time monitoring and regulation of energy storage systems, ensuring that energy is stored and dispensed with maximum efficiency.

In the current era of digitalization and boundless technological growth, energy production and consumption levels vary greatly based on the type of utility and

the consumer employing it, the development of accessible artificial tools and applications allow for their wide and maximum integration in everyday tasks putting the consumer as the central piece that is capable of controlling and operating the resources provided through the energy systems. The need for constant energy supply and consumers increasingly becoming more and more interactive with the advanced equipment so as to manipulate the energy systems as per their own requirement calls for a proper examination of these systems that are responsible for the customization and optimization of the energy sources by storing important and useful information about them [5].

1.2 *Research Objective and Scope*

This paper shall undertake an in-depth exploration of the advantages and impediments inherent in the confluence of the IIoT with smart energy systems. By gaining a comprehensive understanding of the benefits and drawbacks of this integration, entities may make sagacious decisions regarding their energy policies. Our inquiry shall encompass the application of IIoT to engender energy proficiency, effect cost reductions, and establish a more steadfast energy system. Furthermore, we shall scrutinize the plausible impediments that may ensue from the amalgamation of IIoT with smart energy systems, encompassing but not restricted to security concerns, data confidentiality challenges, and compatibility dilemmas.

2 Literature Review

In this section, we shall explore various related works and their significance within the topic of our interest—the integration of IIoT in smart energy systems summarizing broadly into two subsections—the benefits and challenges of integrating IIoT in smart energy systems.

2.1 *Benefits of Integrating IIoT in Smart Energy Systems*

In this subsection, we shall delve into the advantages that comes with the amalgamation of IIoT in smart energy systems (Fig. 1).

1. **Exemplary Reliability and Endurance:** The IIoT relies on reliable and enduring sensors to automate tasks that traditionally require human intervention. This reduces the likelihood of errors and ensures error-free operation of services. For example, sensors in remote oil and gas wells can provide instantaneous data to proactively plan maintenance and assess pump efficiency.

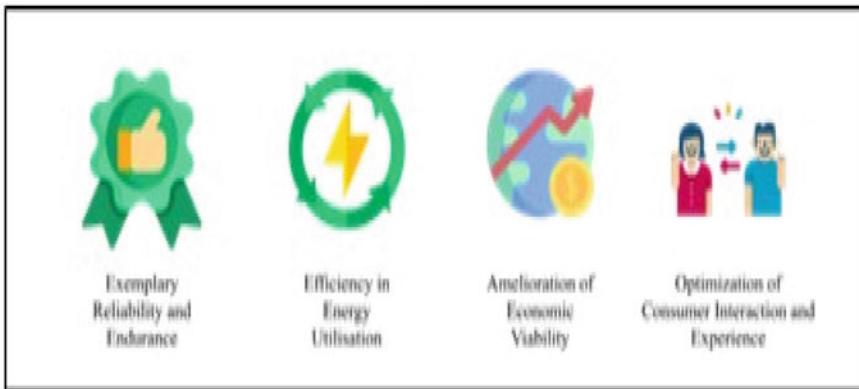


Fig. 1 Benefits of integrating IIoT in smart energy systems

2. **Efficiency in Energy Utilization:** The interconnections between power plant apparatuses promote resource parsimony and consistent production output. The IIoT provides integrative solutions for renewable energy facilities such as automated maintenance and reporting processes, amenable distribution systems, and real-time measurement of carbon emissions. IIoT edge gateways and remote input/output devices gather data from remote equipment, and operators at central control centers utilize artificial intelligence to make accurate predictions about energy requirements for efficient and sustainable energy consumption.
3. **Amelioration of Economic Viability:** Effective maintenance of industrial facilities is crucial to prevent operational inefficiency and reduce maintenance costs. The IIoT can help eliminate unnecessary maintenance expenses through preventive and predictive monitoring and intelligent detection, improving safety and energy efficiency. IIoT oversight can also identify excess energy consumption, regulate power flow, and implement energy-saving measures during low-demand periods, resulting in significant cost savings [6].
4. **Optimization of Consumer Interaction and Experience:** Systematic monitoring of infrastructure assets enables operators and patrons to analyze energy consumption, promote energy conservation and efficiency, and select preferred energy suppliers through peer-to-peer electricity exchange. Companies like Talkpool provide innovative solutions for deploying smart sensors in buildings to monitor energy use and prevent damage, such as water leaks. This accurate data can improve pricing policies, insurance premiums, and preventive maintenance while generating value for residents and potential revenue for building owners [7].



Fig. 2 Challenges of integrating IIoT in smart energy systems

2.2 *Challenges of Integrating IIoT in Smart Energy Systems*

This subsection focuses on the potential obstacles that may arise from the integration of IIoT in smart energy systems (Fig. 2).

1. **Technical Impediments:** The integration of the IIoT within the subsystems of the energy sector is a complex undertaking due to the unique nature of each subsystem, requiring tailored sensor and communication technologies to enable effective monitoring [7]. Successful implementation of smart energy through the IIoT depends on the seamless integration of its systems, which is still a nascent stage. Failure to integrate the IIoT can result in malfunctions and safety hazards, and thus, businesses must be cautious in relying on integration to develop their cases. Solutions, such as comprehensive frameworks and co-simulation models for energy systems are suggested to alleviate integration quandaries [7].
2. **The Imposition of Regulatory Constraints:** IoT systems used in energy systems require a significant amount of energy to maintain operations and transmit data generated by these devices, posing a formidable challenge. According to the International Energy Agency, the energy consumption attributed solely to IoT idle status will be equivalent to Portugal's entire electricity production, excluding capital investments by 2025. However, various measures, such as sleep modes in sensors, efficient communication protocols, optimization methods, and energy-efficient routing techniques have been investigated to mitigate this challenge [7].
3. **Data Analytics Conundrum:** The volume and diversity of data generated by industrial systems necessitate adequate computing, networking, and storage facilities at the Internet's edge. However, centralized cloud controllers still manage edge services and data management, creating a potential single point of failure. Moreover, the analysis of data streams and identification of relevant actions for potential savings is challenging, leading to only a small fraction of data being

analyzed and difficulties in utilizing data for decision-making processes with a return on investment for service providers and end-users [7].

4. **Security and Privacy Concerns:** The complex interconnection of industrial infrastructures and the escalation of cyberattacks pose significant security and privacy concerns in IIoT systems, which can result in the compromise of personal information and the confidentiality of energy consumers [8]. The extensive deployment of IoT systems across the energy sector exposes these systems to a higher risk of cyberattacks, underscoring the necessity for robust cybersecurity measures [9]. To overcome the barriers to the adoption of IIoT in smart energy systems, it is essential to strengthen the security of IIoT subsystems and develop strategies that can counter sophisticated cyberattacks such as targeted ransomware and hijacked two-factor attacks.

3 Security Concerns in Integrating IIoT with Smart Energy Systems

Figure 3 illustrates the security challenges of integrating IIoT in smart energy systems. A weak or absent authentication mechanism is the source of successive security vulnerabilities. Resource-constrained environment makes it difficult to design a strong authentication mechanism for such systems, thereby making them vulnerable to various active and passive attacks. Secondly, it is significant to note that communication occurs over a public channel, and interception of the messages is a serious threat to system security, therefore network security is among the notable security challenges while integrating IIoT in smart energy systems [10]. Thirdly, default passwords must be changed and strong passwords must be set up by the authorized person to prevent brute-force attacks. Withal, the devices must be updated time-to-time and security vulnerabilities must be patched as the use of outdated software is a parlous threat to the security of the system [11].

4 Future Directions for Integrating IIoT in Smart Energy Systems

In [12], applications of IoT in energy management system for the smart grid explained methods to attain improved system accuracy and efficiency levels. The electrical equipment could be manipulated and studied by installing sensors and devices to optimize the usage of energy, with reduced consumption cost as an outcome. Thus enabling the end-user to monitor and handle their energy consumption from remote areas through the Internet. Estimation of the load capacity and forecasting for future long-term use could be realized by gathering real-time energy data in the network through IoT.

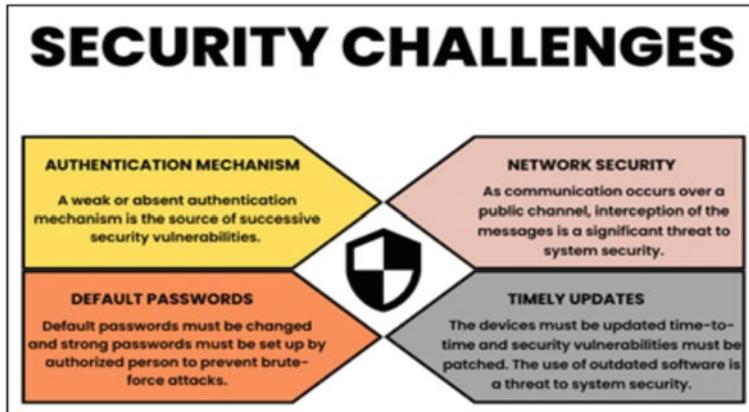


Fig. 3 Security challenges of integrating IIoT in smart energy systems

Applications like UAVs are a very powerful tool for real-time inspection of power lines. They can easily detect infrastructure and geographical features like trees close to power lines while spotting the oil, natural gas, and water pipe lines, storing information for their inspection and data analysis through machine learning algorithms. They provide advanced data processing of the acquired data, with 2D and 3D images for asset inspection [13] in order to prepare detailed reports on the property inspected, likely serving as blueprints for the development of energy plants in the particular area.

The IIoT is expected to have a significant impact on smart energy systems, enabling continuous monitoring and predictive maintenance of energy production and distribution networks. IIoT technology will clear the way for emergence of decentralized energy generation systems, optimized energy network performance, and increased adoption of renewable energy sources, thereby reducing the carbon footprint of the energy sector. A primary and major advantage of assimilation of IoT in smart energy systems is the real-time data it provides about energy utilization, facilitating exploration of possible ways to improve energy efficiency and minimize waste and expenses [13–21]. Last but not the least apart from IIoT working as a standalone tool, it can also be merged with blockchain technology. This collaboration will greatly increase the scope for secure, dependable, and transparent transaction in the energy sector by bringing out the best attributes of both of the constituent technologies. These changes will accelerate the transformation of the energy market into a decentralized one.

5 Conclusion

The IIoT and smart energy systems (SES) are profoundly affecting industrial energy consumption patterns. Bringing together these technologies has resulted in a remarkable increase in efficiency and reduction of energy consumption. Within the energy sector, the IoT market is predicted to grow at a Compound Annual Growth Rate (CAGR) of 11.8% between 2021 and 2025 [13]. IIoT and SES have been analyzed in this study in terms of their many advantages and disadvantages. There are several factors contributing to the increasing IIoT integration, including computational power, memory used, scope of data, and quality of device management. Although integration offers obvious benefits such as increased efficiency, lower energy costs, and improved data analytics, certain obstacles must be recognized. Cybersecurity risks, lack of standardization across industries, and cost of implementation are some of these challenges. In conclusion, the integration of IoT and SES is undoubtedly advantageous, but only when it is conducted correctly and in a manner that recognizes the hazards and challenges that will be encountered.

References

1. Abir SAA, Anwar A, Choi J, Kayes ASM (2021) Iot-enabled smartenergy grid: applications and challenges. IEEE Access 9:50961–50981. <https://ieeexplore.ieee.org/document/9381850>, <https://doi.org/10.1109/ACCESS.2021.3067331>
2. Zhang D, Chan CC, Zhou GY (2018) Enabling Industrial Internet of Things (IIoT) towards an emerging smart energy system. Glob Energ Interconnection 1(1):39–47. <https://doi.org/10.14171/J.2096-5117.GEI.2018.01.005>
3. Kamyab H, Klemés JJ, Van Fan Y, Lee CT (2020) Transitionto sustainable energy system for smart cities and industries. Energy 207:118104. <https://doi.org/10.1016/j.energy.2020.118104>
4. Orumwense EF, Abo-Al-Ez K (2023) Internet of Things for smart energy systems: a review on its applications, challenges and future trends. AIMS Electron Electr Eng 7(1):50–74. <https://doi.org/10.3934/electreng.2023004>
5. Zatsarinaya Y, Logacheva A, Suslov K (2020, September) Outlook on the development of smart energy systems. In: 2020 International Ural conference on electrical power engineering (UralCon). IEEE, pp 19–23. <https://ieeexplore.ieee.org/document/9216266>. <https://doi.org/10.1109/UralCon49858.2020.9216266>
6. Malik PK, Sharma R, Singh R, Gehlot A, Satapathy SC, Alnumay WS, Pelusi D, Ghosh U, Nayak J (2021) Industrial Internet of Things and its applications in industry 4.0: state of the art. Comput Commun 166:125139. <https://doi.org/10.1016/j.comcom.2020.11.016>
7. Motlagh NH, Mohammadrezaei M, Hunt J, Zakeri B (2020) Internet of Things (IoT) and the energy sector. Energies 13:494. <https://doi.org/10.3390/en13020494>
8. Demertzis V, Demertzis S, Demertzis K (2023) An overview of privacy dimensions on Industrial Internet of Things (IIoT). publisher arXiv, 2023, copyright Creative Commons Attribution 4.0 International. <https://arxiv.org/abs/2301.06172>, <https://doi.org/10.48550/ARXIV.2301.06172>
9. Humayun M, Jhanjhi NZ, Alruwaili M, Amalathas SS, Balasubramanian V, Selvaraj B (2020) Privacy protection and energy optimization for 5G-aided industrial Internet of Things. IEEE Access 8:183665–183677. <https://ieeexplore.ieee.org/document/9214512>. <https://doi.org/10.1109/ACCESS.2020.3028764>
10. Harper A (February 03, 2023) 11 Biggest security challenges and solutions for IoT. <https://www.peerbits.com/blog/biggest-iot-security-challenges.html>

11. Internet of Things security challenges and best practices. <https://www.kaspersky.com/resource-center/preemptive-safety/best-practices-for-iot-security>
12. Muleta N, Badar AQH (2021) Study of energy management system and IOT integration in smart grid. In: 2021 1st International conference on power electronics and energy (ICPEE), Bhubaneswar, India, pp 1–5. <https://doi.org/10.1109/ICPEE50452.2021.9358769>
13. Incekara ÇO (2022) Industrial internet of things (IIoT) in energy sector. Adv Eng Days (AED) 5:181–184. <http://193.255.128.114/index.php/aed/article/view/819/656>
14. Braten AE, Kraemer FA, Palma D (2019) Adaptive, correlationbased training data selection for iot device management. In: Sixth international conference on internet of things: systems, management and security (IOTSMS). IEEE, pp 169–176. <https://ieeexplore.ieee.org/abstract/document/8939220>, <https://doi.org/10.1109/IOTSMS48152.2019.8939220>
15. Narwal B, Mohapatra AK (2021) A survey on security and authentication in wireless body area networks. J Syst Architect 113:101883
16. Narwal B, Mohapatra AK, Usmani KA (2019) Towards a taxonomy of cyber threats against target applications. J Stat Manag Syst 22(2):301–325
17. Narwal B, Mohapatra AK (2021) SAMAKA: secure and anonymous mutual authentication and key agreement scheme for wireless body area networks. Arab J Sci Eng 46(9):9197–9219
18. Sharma M, Narwal B, Anand R, Mohapatra AK, Yadav R (2023) PSECAS: a physical unclonable function based secure authentication scheme for Internet of Drones. Comput Electr Eng 108:108662
19. Malik M, Gandhi K, Narwal B (2022) AMAKA: anonymous mutually authenticated key agreement scheme for wireless sensor networks. Int J Inform Sec Privacy (IJISP) 16(1):1–31
20. Narwal B, Gandhi K, Anand R, Ghalyan R (2022, September) PUASIOT: password-based user authentication scheme for IoT services. In: Proceedings of the 6th international conference on advance computing and intelligent engineering: ICACIE 2021. Springer Nature Singapore, Singapore
21. Narwal B, Bansal V, Dahiya V, Aggarwal P (2021) SLUASCIoT: a secure and lightweight user authentication scheme for cloud-IoT services. In: 2021 5th International conference on information systems and computer networks (ISCON), Mathura, India, pp 1–5. <https://doi.org/10.1109/ISCON52037.2021.9702456>

Blockchain-Based Secure Mutual Authentication Scheme for Drone-GSS Communication in Internet of Drones Environment



Anvita Gupta, Ayushi Jain, and Mehak Garg

Abstract Drones are becoming increasingly common and are used for a variety of purposes, such as delivery, military, and surveillance. They are becoming more capable and available owing to technological advancements including miniaturization, lightweight materials, and improved sensors. As technology advances, the Internet of Drones (IoD) has the potential to revolutionize key industries and become more prevalent and indispensable to our everyday lives. However, drones can be compromised and operated by unauthorized users if their networks are not effectively secured against cyberattacks, thereby seriously jeopardizing public safety and privacy. To avert unauthorized access to the drone and its data, and guarantee the drone's reliable functioning, a secure authentication mechanism between a drone and its ground station server (GSS) is crucial. Additionally, as drone devices often have small batteries and little memory storage, it is important to use lightweight security approaches. To address these problems, a lightweight blockchain-assisted and PUF-based mutual authentication system for drone-GSS authentication has been proposed in this paper. This system incorporates blockchain principles to develop security mechanisms that can overcome the shortcomings of the IoD. The proposed protocol is resilient against various security attacks, proved through both informal and formal security analysis. Moreover, the proposed scheme's effectiveness has been compared with that of state-of-the-art drone authentication techniques.

Keywords AVISPA · Blockchain · Internet of drones (IoD) · Mutual authentication · Physical unclonable function (PUF)

A. Gupta · A. Jain (✉) · M. Garg

IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), New Delhi, India

e-mail: ayushi038btit19@igdtuw.ac.in

A. Gupta

e-mail: anvita072btit19@igdtuw.ac.in

M. Garg

e-mail: mehak045btit19@igdtuw.ac.in

1 Introduction

Drones, also referred as unmanned aerial vehicles (UAVs), have distinctive characteristics such as mobility and flexibility in deployment and maintenance, and are a cost-effective method for collecting, sending, and analyzing data. Drone technology is being researched and developed at an unprecedented pace globally providing services like inventory tracking, crisis management, the movement of goods, traffic control, etc., which were primarily used in military applications. As a result of IoTs (Internet of Things) modernization, the network of drones is now referred to as the “Internet of Drones” revolutionizing sectors, opening new applications, and enhancing productivity. Despite its obvious advantages, the open and unreliable wireless medium of drones is considered to be susceptible to several attacks, including man-in-the-middle (MitM), replay, and impersonation attacks. Moreover, security standards such as maintaining confidentiality, anonymity, and privacy should be complied with when communicating [1]. Authentication is a crucial safeguard for preserving confidence by excluding unauthorized users and compromised devices. A secure authentication scheme between a drone and its GSS is, therefore, necessary to ensure that only authorized users can control the drone and access its data. However, traditional authentication techniques rely on dynamic keys, usernames, and passwords, all of which provide very low levels of security. Lightweight security and authentication methods are ideal for drones since they are particularly susceptible to node tampering attacks and may additionally have small batteries and inadequate memory storage. The presented research provides a lightweight mutual authentication technique for authenticating drones with a base station while considering these challenges.

PUFs have been demonstrated to offer immense potential for assuring security in many IoT applications in recent research. The challenge-response pair mechanism underlies the operation of PUFs; specifically, PUFs have the property that they produce an output known as the response in response to an input stimulus known as the challenge. In the proposed approach, the drones use a response generated by the PUF’s challenge instead of holding any classified information with them. The authentication procedure’s secret is contained in the generated response. Moreover, incorporating blockchain technology can be a promising approach to assure data security and veracity. The inherent features of blockchain make it particularly well-suited for developing authentication schemes in applications [2–8]. It is not possible to alter data once it has been recorded and stored in the blockchain. Hence, GSS can check the received data from drones, and after the data transaction is completed, the transaction logs are reserved on the blockchain, which cannot be tampered with. The main offerings provided by this paper are as follows.

1. We propose a secure blockchain-based mutual authentication scheme between the drone and the ground station for secure data transmission. The proposed scheme offers privacy, anonymity, and forward secrecy.
2. We offer an informal and formal critique of the suggested study using AVISPA. Furthermore, its effectiveness has been compared with state-of-the-art authentication schemes.

2 Literature Review

In advance of the sharing of sensitive data, the IoD's authentication function is crucial for verifying the real identities of all communication entities.

In IoD contexts, for resource-constrained unmanned drones the temporal credential-based anonymous lightweight authentication strategy, or TCALAS, is presented in [9]. Only authorized users who have registered with the GSS in the TCALAS are permitted to access the services provided by remote drones. All remote drones must register with GSS to render services. Following registration, a secret credential, known exclusively to the GSS and remote drone is assigned. The users who are registered have the option to modify their biometrics and/or passwords whenever they want without further contacting the GSS. The authors of [10] proposed SecAuthUAV, mutual authentication protocol that is PUF-based enables a secure session between a drone and its GSS without requiring the drone to store any secret data while also generating a new session key for every use. To ensure secure communication in unreliable drone networks, Semal et al. [11] presented a certificateless group authenticated key agreement (CL-GAKA) technique; however, due to the computationally intensive nature of bilinear pairing and ECC, it is not lightweight and does not protect against physical attack and drone tampering. A PUF-based two-stage mutual authentication mechanism called PARTH has been proposed by the authors of [12] and is intended for multi-drone networks that are SDN-based to be deployed in surveillance zones. Although it is not necessary to keep any secret keys in the drone's memory, it is a bit lacking since PARTH must locate leader and mini drones with multiple duties as part of the agreement, thereby requiring extremely stable drone communication across the system. For a multi-domain environment, the authors of [13] suggested an identity authentication scheme for drones that can create secure channels for communication between various drones and various GSS based on PUF and protect against typical security attacks while ensuring mutual authentication and drone anonymity in addition to avoiding a single point of failure. Recent years have seen the emergence of blockchain-assisted authentication techniques in the IoD environment. Data integrity, transparency, traceability, encryption methods, and operational durability are examples of BC capabilities that can help inscribe many of the IoT architectural flaws by offering non-repudiation, authenticity, and default integrity and utilizing smart contracts. Blockchains offer decentralized data storage options in addition to recording and securing transactions using encryption.

The research [14] presents a private blockchain-based solution for encrypted transmission in an IoD-assisted healthcare system due to the exceptional sensitivity of healthcare data. In an IoD environment, Bera et al.'s [15] proposed for a blockchain-based access control network where blocks are created using transactions, which are formed from secure data acquired by GSS. Once a block is uploaded to the blockchain via a consensus mechanism, the transactions it contains cannot be altered, amended, or removed. A significant proportion of drone-based applications are plagued by real-time latency and have weak authentication protocols. Yazdinejad et al.'s [16] research of a blockchain-based secure authentication paradigm for drones in smart

cities attempted to address both challenges. The drone network was designed using a zone-based architecture, and each zone's drones were given access to a distinct decentralized consensus without the need for re-authentication.

3 System Model

3.1 System Architecture

The architecture consists of the following entities, i.e. registration authority, drones, ground station server, and blockchain. Figure 1 gives the pictorial representation of the architecture.

Registration Authority: The system's RA stands for registration authority or trusted authority. The other system entities including GSS are registered by RA in the network. It initializes the system by setting up various system parameters which includes providing public and private keys. It also manages the deployment of blockchain and writes data to it, including public key information.

Drone: Drone, a common name for DRONE, is registered with the GSS closest to it. The drone collects real-time data with the help of sensors and provides them to GSS after mutual authentication. Each time data is shared between them, the drone maintains the log of these transactions on the blockchain.

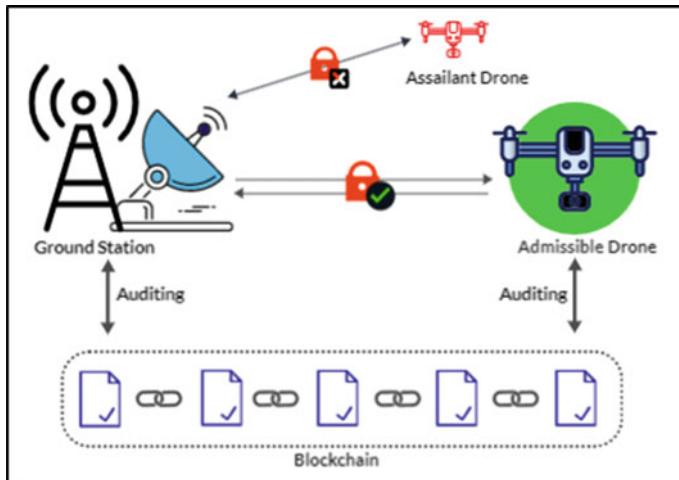


Fig. 1 System architecture

Ground Station Server: GSS is responsible for all drones in a specific region. It registers all of them, handles data collected by them, and performs mutual authentication with them to communicate. Since the log is maintained by drones it can query data and check data integrity through blockchain.

Blockchain: Blockchain is a shared data directory in the system that can be accessed to get the value of public variables and hash values. The hash values are maintained by the blockchain to ensure data hasn't been modified. Blockchain can be queried about logs for previous transactions, each of which is stored using a unique signature that is verified by smart contracts.

3.2 System Goals/Requirements

1. Mutual Authentication: Achieving mutual authentication by confirming the communication requests between the admissible drone and GSS.
2. Proper Establishment of Session Key: Before communication takes place, a session key is negotiated between two parties which must be unique.
3. Drone Authenticity: Only the admissible drone is authenticated by GSS and assailant drone requests are not dealt with.
4. Tolerance against multiple attacks: The protocol is tolerant towards various kinds of attacks including MitM attacks, impersonation attacks, replay attacks, etc.
5. Message Tampering Awareness: The entities can detect if a message has been tampered with in between and terminate the authentication immediately when the situation arises.
6. Device Anonymity: GSS should be able to track the temporary IDs of each drone, helping in maintaining the anonymity of each drone.
7. Forward Secrecy: The prior session keys between the associated entities are still hidden from the attacker to maintain forward secrecy, regardless of whether the archived keys of the entities are compromised.

3.3 Assumptions

We made certain assumptions based on which our authentication scheme is designed. Each assailant drone holds a different PUF. The PUF ends up obsolete if they are physically captured or tampered with and the drone is unable to authenticate with the ground station. Another assumption is that the drones don't keep any classified information aboard them; instead, they employ a response R produced by the PUF's challenge C. The generated response also serves as the authentication process's secret. The GSS has unlimited computational power and memory, in contrast to the drones' limited computing capabilities and memory.

4 Proposed Scheme

A secure authentication scheme between drone and GSS is proposed by us in this paper. Our scheme is broken down into three stages: initialization, registration, and authentication. The network is initialized by RA which registers GSS while drones are registered with GSS. When the drone sends acquired data to the GSS, they must both attain mutual authentication and confirm a session key before data can be shared. The blockchain stores a record/log of all transactions.

4.1 Initialization Phase

This phase is initialized by RA, whose duty is to set up the system environment by assigning parameters and creating the blockchain (Table 1).

Parameter set-up

1. A large prime number a is selected by RA which helps in generating the finite field F_a over which elliptic curve EC_a is defined.
2. RA then chooses a point on the curve V of order m . It also selects the hash function, $H(.): \{0, 1\}^* \rightarrow Z^*$.
3. RA randomly chooses a $c \in Z^*$ as its private key and computes the private key as $PUB = c.V$ is calculated.

Table 1 Notation and their descriptions

Notations	Descriptions
a	Prime number in the initialization phase
F_a	Finite field over a
EC_a	Elliptic curve
c	Private key of RA
V	Point on the elliptic curve
$H(.)$	Hash function defined
PUB	Public key of RA
GID, UID	IDs for GSS and the drone
RT_{GSS}	Registration timestamp of GSS
TID_{GSS}	Temporary ID of GSS
c_g, P_g	Private and public keys of GSS
C, R	PUF challenge, response pair
AID_{dr}	Assumed temporary ID of the drone
RN_X, RN_Y, RN_Z	Random nonce generated
SKey	Session key

4. The parameters $\{\text{EC}_a, F_a, V, H(\cdot), \text{PUB}\}$ are published while c is kept as a secret by RA.

Blockchain set-up

RA creates a consortium blockchain where all the blockchain nodes presumed to be trustworthy are authenticated before they can join in. RA accepts the smart contracts as well which are compiled and then deployed on the blockchain. A unique address is assigned to the smart contract after miners validate it. Permitted nodes can invoke the functionality of smart contracts by forwarding a transaction to a unique address.

4.2 Registration Phase

In this phase, RA registers GSS by generating private and public keys for them where public keys are deployed over blockchain and private keys are transmitted securely while each drone is registered with the GSS with the help of the PUF challenge-response pair.

GSS Registration

GSS sends a registration request to RA along with its ID called GID. RA after receiving this request picks the registration timestamp, RT_{GSS} and a random number $r_g \in Z^*$. Then, $R_g = r_g.V$ is calculated. $\text{TID}_{\text{GSS}} = H(\text{GID} \oplus R_g)$ is assigned and the private key $c_g = c + H(\text{TID}_{\text{GSS}} \parallel \text{RT}_{\text{GSS}} \parallel R_g)$. r_g for GSS is set by RA, where $P_g = c_g.V$, RA appends variables $\{\text{TID}_{\text{GSS}}, \text{RT}_{\text{GSS}}, P_g\}$ to the blockchain while $\{c_g, R_g\}$ are securely sent to GSS. GSS on receiving the reply from RA, calculates $\text{AID}_{\text{GSS}} = H(\text{GID} \oplus R_g)$ on its side and verifies if $c_g.V = \text{PUB} + H(\text{TID}_{\text{GSS}} \parallel \text{RT}_{\text{GSS}} \parallel R_g)$. R_g stands. If yes, GSS stores c_g as a private key or else rejects the data received and reports an error to the system.

Drone Registration with GSS

- The drone sends the registration request and its ID called UID to GSS.
- GSS on receiving this request selects a challenge C at random and sends it back.
- Drone runs PUF on the received challenge C and regenerates response $R \leftarrow \text{PUF}(C)$ which is sent back to GSS.
- GSS generates an assumed temporary ID, $\text{AID}_{\text{dr}} = H(\text{UID} \oplus R)$ for the drone. It stores $\{\text{AID}_{\text{dr}}, C, R\}$ in its database while drone stores $\{\text{AID}_{\text{dr}}, C, \text{GID}\}$.

4.3 Authentication Phase

To establish a secure data transmission between the drone and GSS, the protocol presented ensures that the valid drone is authenticated via mutual authentication

and a session key is established between the drone and GSS before any further communication. The authentication can be seen in Fig. 3 for a better understanding.

- The drone creates the response R by running PUF on the challenge C stored when it wants to authenticate.
- It sends the authentication request along with the variables $\{\text{AID}_{\text{dr}}, \text{RN}_X, H(R||\text{AID}_{\text{dr}}||\text{RN}_X)\}$ to GSS, where RN_X is a random nonce.
- GSS after receiving the request confirms the validity of AID_{dr} and determines the freshness of the message with the help of RN_X . If the conditions are not met the request is rejected. Or else, it then verifies the hash received by extracting $\{C, R\}$ from its database for corresponding AID_{dr} .
- After verification, it generates a new random nonce RN_Y and calculates $A = H(R||\text{GID}||\text{AID}_{\text{dr}})$ and performs the following operations:

$$\begin{aligned} B &= A \oplus \text{RN}_Y, \\ Q &= A || (B \oplus \text{RN}_Y) \end{aligned}$$

The variables Q and hash $H(Q||\text{RN}_X||\text{RN}_Y||A)$ are sent to the drone.

- The drone on its side calculates A again and gets the value of RN_Y using the following operations:

$$\begin{aligned} B &= A \oplus Q* \\ \text{RN}_Y &= A \oplus B \end{aligned}$$

Now it computes the value of the hash and verifies it. If it doesn't match then the authentication is dropped (Fig. 2).

- Otherwise, it computes a random nonce RN_Z and uses part of it as a new challenge C' and evaluates the corresponding R' for it. It also calculates

$$\begin{aligned} I &= R' \oplus H(\text{RN}_Y||A), \\ J &= A \oplus \text{RN}_Z, \\ \text{SKey} &= H(A || (\text{RN}_Y \oplus \text{RN}_Z)) \end{aligned}$$

SKey is the session key. The drone sends $\{I, J, H(R'||\text{RN}_Y||\text{RN}_Z||\text{TID}_{\text{dr}}||\text{SKey})\}$ to GSS.

- GSS determines $\text{RN}_Z = A \oplus J$, $R' = H(\text{RN}_Y||A) \oplus I$, $\text{SKey} = H(A || (\text{RN}_Y \oplus \text{RN}_Z))$ and uses this data to verify the hash received. After the verification, the new $\{C', R'\}$ is saved in the database with the preceding $\{C, R\}$.

The mutual authentication is achieved and a new assumed temporary ID, AID'_{dr} for the drone, i.e. $\text{AID}'_{\text{dr}} = H(A || \text{AID}_{\text{dr}} || R')$ is calculated and updated by both GSS and the drone.

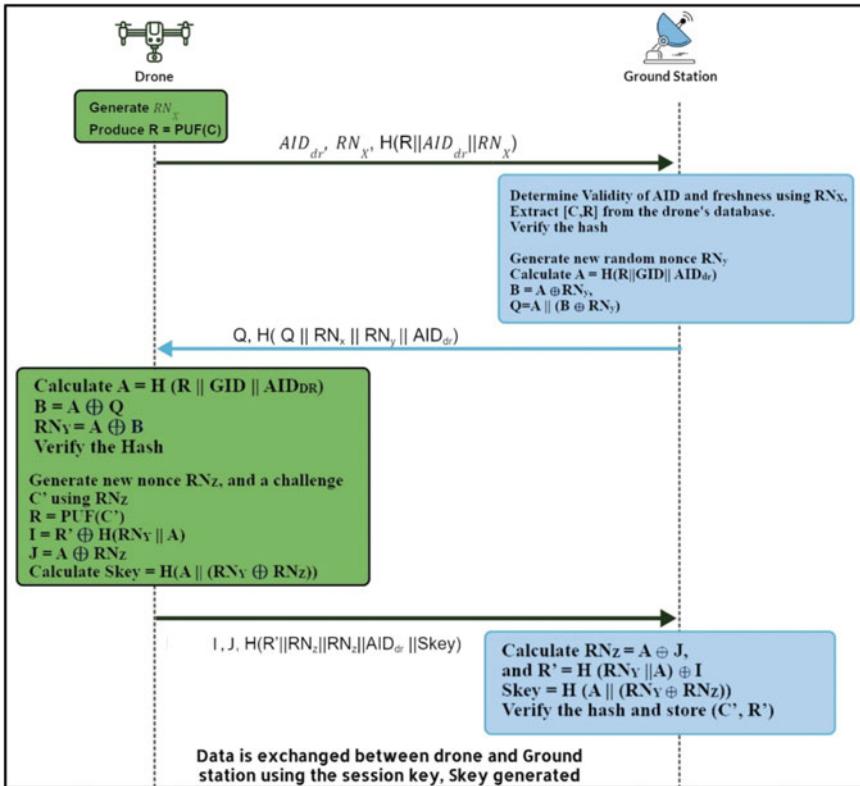


Fig. 2 Authentication between the drone and GSS

5 Security Analysis

In this section, we perform an informal analysis to argue that the proposed scheme defeats multiple types of attacks and make use of the AVISPA software validation tool [3, 17] for formal analysis. The analysis is done on the drone and GSS authentication phases.

5.1 Informal Analysis

- Mutual Authentication:** GSS and the drone with the help of PUFs and random numbers generated along with hash functions used to mask the variables achieves mutual authentication. And $\{C, R\}$ are stored by GSS in the registration phase and are not transmitted during authentication. Whenever a drone wants to authenticate it sends over AID_{dr} , along with a random number and hash function for

```

SUMMARY
SAFE

DETAILS
BOUNDED_NUMBER_OF_SESSIONS
TYPED_MODEL

PROTOCOL
/home/span/span/testsuite/results/scheme.if

GOAL
As Specified

BACKEND
CL-AtSe

STATISTICS

Analysed    : 0 states
Reachable   : 0 states
Translation: 0.01 seconds
Computation: 0.00 seconds

```

Fig. 3 AVISPA results with CL-AtSe backend

verification to GSS, but all these variables are temporary and change in each transaction, so no adversary can store values and interrupt authentication. While verification is done by both the drone and GSS, so authentication is mutual.

2. **Session Key Exposure:** The drone and GSS decide on a session key as they share some temporary variables like AID_{dr} and challenge-response which is updated later. All of these elements are traded under cover and are unknown to the attacker. Therefore, without the knowledge of input, the attacker cannot generate a valid session key. And even if past session keys are leaked, it won't be a threat as a new session key has to be created for every transaction.
3. **Man-in-the-Middle Attack:** The PUF's randomness and ECC-based signature scheme during initialization make it impossible for any attacker to generate valid verification messages such as hash values passed along, making the MitM attack ineffective.
4. **Replay Attack:** Protocols employ timestamps and random numbers to prevent replay attacks. As in every transaction a new message is generated with a unique random nonce, so previously leaked or used messages can't be used to authenticate. Also, after each transaction a $\{C, R\}$ is created.
5. **Impersonation Attack:** Since it is impossible to clone PUF it is impossible to impersonate the drone and get unauthorized access. And even if an attacker gets

Table 2 Security comparison among various schemes

Schemes	Sch1	Sch2	Sch3	Sch4	Sch5	Sch6	Sch7	Sch8
[18]	☺	☺	☺	☺	☺	☺	☺	☺
[19]	☺	☺	☺	☺	☺	☺	☺	☺
[20]	☺	☺	☺	☺	☺	☺	☺	☺
Our scheme	☺	☺	☺	☺	☺	☺	☺	☺

☺ Attribute present, ☺ Attribute absent

their hands on one of the entities, it is difficult to impersonate as $\{C, R\}$ are updated each time after the transaction.

6. **Device Anonymity:** For drones, it uses AID_{dr} during authentication. These AIDs are updated in each authentication by taking a hash of the new response generated by PUF and the previous value. GSS only saves the current AID of the drone. Device anonymity is maintained as it is not possible to obtain the true identity of the drone from its AID.
7. **Forward Secrecy:** During the authentication of the drone with GSS, the session key is generated with the help of multiple random nonces and response R of PUF. And during the next session a different R' will be used which is not possible for an attacker to obtain. Since our scheme provides device anonymity it is not possible to track the ID of the drones by the attacker for future sessions, hence providing forward secrecy.
8. **Data Verifiability:** Using blockchain, our scheme ensures data verifiability. After receiving the collected data from the drone, GSS can validate the integrity of data using the hash recorded in the blockchain. GSS will detect that the data has been altered if the hash values are not the same and will conclude that the data is not valid.

Table 2 gives the security comparison with various schemes where mutual authentication (Sch1), security against session key exposure (Sch2), MITM (Sch3), and replay (Sch4) attacks are maintained by all. But scheme [18] does not show protection against impersonation attacks (Sch5). While scheme [19] lags device anonymity (Sch6) and scheme [20] does not provide forward secrecy (Sch7). And none of the other schemes shows data verifiability (Sch8).

5.2 Formal Analysis

AVISPA Simulation

We conducted a formal analysis using a popular simulation tool, AVISPA [17]. This analysis guarantees that the provided scheme is resistant to replay, MitM, and impersonation attacks. Figure 3 shows the result for AVISPA.

6 Conclusion

This paper presented a secure mutual authentication scheme employing PUFs and hashing for Drone-GSS authentication that eliminates the need for any secret data to be stored on the drones. Mutual authentication, drone anonymity, and forward secrecy are all guaranteed by the proposed approach. Our scheme has been tested safe over AVISPA, proving it to be safe from conventional security threats like MitM, replay, and impersonation. We contend that the suggested protocol is extremely effective and practical for Drone-GSS mutual authentication since it performs competitively well in comparison with other cutting-edge authentication systems.

References

1. Yang W, Wang S, Yin X, Wang X, Hu J (2022) A Review on security issues and solutions of the internet of drones. *IEEE Open J Comput Soc* 3:96–110. <https://doi.org/10.1109/OJCS.2022.3183003>
2. Sharma M, Narwal B, Anand R, Mohapatra AK, Yadav R (2023) PSECAS: a physical unclonable function based secure authentication scheme for Internet of Drones. *Comput Electr Eng* 108:108662
3. Narwal B, Mohapatra AK (2021) A survey on security and authentication in wireless body area networks. *J Syst Architect* 113:101883
4. Narwal B, Mohapatra AK, Usmani KA (2019) Towards a taxonomy of cyber threats against target applications. *J Stat Manag Syst* 22(2):301–325
5. Narwal B, Mohapatra AK (2021) SAMAKA: secure and anonymous mutual authentication and key agreement scheme for wireless body area networks. *Arab J Sci Eng* 46(9):9197–9219
6. Malik M, Gandhi K, Narwal B (2022) AMAKA: anonymous mutually authenticated key agreement scheme for wireless sensor networks. *Int J Inf Secur Privacy (IJISP)* 16(1):1–31
7. Narwal B, Gandhi K, Anand R, Ghalyan R (2022) PUASIoT: password-based user authentication scheme for IoT services. In: Proceedings of the 6th international conference on advance computing and intelligent engineering: ICACIE 2021. Springer Nature Singapore, Singapore, pp 141–149
8. Narwal B, Bansal V, Dahiya V, Aggarwal P (2021) SLUASCIoT: a secure and lightweight user authentication scheme for cloud-IoT services. In: 2021 5th international conference on information systems and computer networks (ISCON), Mathura, India, pp 1–5. <https://doi.org/10.1109/ISCON52037.2021.9702456>
9. Srinivas J, Das AK, Kumar N, Rodrigues JJPC (2019) TCALAS: temporal credential-based anonymous lightweight authentication scheme for internet of drones environment. *IEEE Trans Veh Technol* 68(7):6903–6916. <https://doi.org/10.1109/TVT.2019.2911672>
10. Alladi T et al (2020) SecAuthDRONE: a novel authentication scheme for DRONE-ground station and DRONE-DRONE communication. *IEEE Trans Vehic Technol* 69(12):15068–15077 (2020)
11. Semal B, Markantonakis K, Akram RN (2018) A certificateless group authenticated key agreement protocol for secure communication in untrusted DRONE networks. In: 2018 IEEE/AIAA 37th digital avionics systems conference (DASC), London, UK, pp 1–8. <https://doi.org/10.1109/DASC.2018.8569730>
12. Alladi T, Chamola V, Kumar N (2020) PARTH: a two-stage lightweight mutual authentication protocol for DRONE surveillance networks. *Comput Commun* 160:81–90
13. Tian C et al (2022) Reliable PUF-based mutual authentication protocol for DRONES towards multi-domain environment. *Comput Netw* 218:109421

14. Wazid M, Bera B, Mitra A, Das AK, Ali R (2020) Private blockchain-envisioned security framework for AI-enabled IoT-based drone-aided healthcare services. In: DroneCom'20: proceedings of the 2nd ACM MobiCom workshop on drone assisted wireless communications for 5G and beyond, September 2020, pp 37–42
15. Bera B, Chattaraj D, Das AK (2020) Designing secure blockchain-based access control scheme in IoT-enabled internet of drones deployment. *Comput Commun* 153:229–249
16. Yazdinejad A, Parizi RM, Dehghantanha A, Karimipour H, Srivastava G, Aledhari M (2021) Enabling drones in the Internet of Things with decentralized blockchain-based security. *IEEE Internet Things J* 8(8):6406–6415
17. Viganò L (2006) Automated security protocol analysis with the AVISPA tool. *Electron Notes Theor Comput Sci* 155:61–86
18. Wazid M, Das AK, Kumar N, Vasilakos AV, Rodrigues JJ (2018) Design and analysis of secure lightweight remote user authentication and key agreement scheme in Internet of Drones deployment. *IEEE Internet Things J* 6(2):3572–3584
19. Wu T-Y, Lee Y-Q, Chen C-M, Tian Y, Al-Nabhan NA (2021) An enhanced pairing-based authentication scheme for smart grid communications. *J Ambient Intell Humanized Comput*, early access, 1–13. [Online]. Available: <https://link.springer.com/article/10.1007/s12652-020-02740-2>
20. Khatoon S, Rahman SMM, Alrubaian M, Alamri A (2019) Privacy-preserved, provable secure, mutually authenticated key agreement protocol for healthcare in a smart city environment. *IEEE Access* 7:47962–47971

SLMA: Secure and Lightweight Mutual Authentication Scheme for IoT-Based Healthcare



Preeti Dhankar, Priya Sharma, and Bhargavi Singh

Abstract The Internet of Things (IoT) generates enormous amounts of disparate information from divergent applications such as digital health, smart medical facilities, automated pathology laboratories, monitoring equipment, virtual consultations, data analysis, personalized care, disease prevention, medication management, and other medical applications. IoT in the medical field has the potential to cut expenses, improve efficiency, and develop new methods of treating diseases. It can assist practitioners and patients and boost medical facilities, pharmaceutical companies, research organizations, and government agencies. However, because of privacy flaws, an unauthorized person may approach health-relevant data or restrain IoT devices attached to the body of the patient, yielding unprecedented results. The use of the wireless medium as a communication channel poses several threats. To combat these vulnerabilities, this research proposes SLMA. It establishes an encrypted session for the authenticated user and utilizes patient biometrics using a fuzzy extractor, thereby preventing unauthorized people from getting access to the IoT sensor gateways. To reduce the processor burden at the IoT node, the proposed protocol relies heavily on primitives of lightweight cryptography like XOR and hash operations. It has been tested SAFE over AVISPA and it also has lower computational as well as communication costs, making it more efficient than conventional protocols.

Keywords Attacks · Fuzzy extractor · IoT-based healthcare · Mutual authentication · Security

P. Dhankar · P. Sharma (✉) · B. Singh

IT Department, Indira Gandhi Delhi Technical University for Women (IGDTUW), Delhi, India
e-mail: priya115bit19@igdtuw.ac.in

P. Dhankar
e-mail: preeti114bit19@igdtuw.ac.in

B. Singh
e-mail: bhargavi130bit19@igdtuw.ac.in

1 Introduction

In every domain, the Internet of Things (IoT) has bolstered the traditional way of problem-solving and has made a significant impact on various industries including the healthcare sector. Internet-based connectivity between patients, smart devices, and medical equipment forms the Internet of Medical Things (IoMT) [1]. IoT together with digitization is the foundation of the digital health biodiversity, however, it also comes along with new challenges and security threats. IoMT not only provides efficient and high-quality healthcare but also generates an enormous amount of sensitive patient data. Digital health biodiversity includes patients and healthcare specialists, medical equipment, and an infinite number of wireless sensors, all needing to exchange critical data.

The sensors in wireless medical sensor networks (WMSNs) make up an important component of intelligent healthcare and allows the accumulation of patient data conveniently. The medical industry has seen substantial growth in the use of WMSNs, thence integrating body sensor networks with the Internet has led to enhanced real-time monitoring of patient viscera including heart rate, temperature, respiratory oximetry, and breathing rates. Due to the availability of wearable sensors, integrating them with WMSN actualized remote patient monitoring, improved data accessibility, and increased patient mobility. Nonetheless, WMSNs like any other wireless network, are vulnerable to security attacks and since the data exchanged in smart healthcare is crucial, a single act of malice by adversaries could endanger the client's life and could jeopardize the patient's life [2]. Therefore, the primary issue that must be addressed is the protection of sensitive information, open channels of communication. Catering to this need, researchers have propounded SLMA. The authors have discussed the previous works and literature in this domain and have presented the same in forthcoming sections. Moreover, the network architecture of SLMA, adopted threat model, detailed description of proposed SLMA, security proofs in form of AVISPA, cost comparisons, and informal analysis form the upcoming sections of this paper.

Abundant lightweight authentication schemes for IoT have been posited in the literature. In the following section, we have provided an analysis of the existing research contributions. Braeken [3] presented a server-to-sensor authentication scheme based on symmetric keys that make use of auxiliary information. However, [4] suggested a scheme (RLMA) for preventing unauthorized users in distributed smart environments that employs implicit certificates and allows smart devices inside a smart grid setup to mutually authenticate and be resistant to various attacks but is also efficient due to reduced communication and computation complexities [5]. They provide a one-of-a-kind, lightweight, three-factor user authentication method that overcomes the weaknesses of the target scheme and confirms its security with ProVerif, a formal verification tool. Masud et al. [6] proposed a protocol that protects IoMT networks with all of the necessary security properties. AVISPA and informal security analyses demonstrate its resistance to attacks. The proposed protocol also consumes limited resources and is resistant to physical threats, making it ideal for

IoT-enabled healthcare network applications. Wang et al. [7] proposed a simplified user authentication system for the Internet of Medical Things [8]. The authors of the proposed scheme stated that it would be easier to implement and use for healthcare IoT devices. It uses XOR and hash functions to perform secure token exchange on low-profile sensors. Zhang et al. [9] presented a method that aims to achieve secure communication and mutual authentication through the use of XOR and the hash function. They also suggest a security enhancement that can prevent both forwarding and unlinkability. Attir et al. [1] suggested a novel security enhancement that assures both session unlinkability and forward secrecy. The scheme exhibits robustness against security attacks, as demonstrated by a thorough analysis, and its safety is demonstrated using the AVISPA, while BAN logic ensures mutual authentication. The proposed scheme has lower XOR as well as random number generation costs in terms of complexity. Praveen and Pabitha [10] proposed a scheme that uses a Chinese secret key to protect an IoT network from attacks originating from former sensor nodes. It is also verified by the tool known as AVISPA. Kim et al. [11] proposed a scheme used in low-profile medical devices that has XOR and hash functions and is also confirmed by ProVerif, a cryptographic protocol.

2 System Model

The Dolev-Yao (DY) model [12] is used to evaluate the protocol's performance under various vulnerable scenarios [12]. Consider a medical IoT network [13, 14] that implants a patient's chest with an IoT-enabled pacemaker. Figure 1 depicts the architecture used to demonstrate information exchange between several entities in an IoT healthcare setting, the Phy (physician), gateway (GaW), and body sensor (BS).

3 Proposed SLMA Scheme

3.1 *Phy Registration Phase*

Step 1: Phy has a PhID which is hashed pseudo identity of physician, a password by his biometric then PHY Calls fuzzy extractor [concept of Gen() and Rep()] which computes $G(PhBio) = (Secph, Pubph)$. Using hash operation we compute $PhCred = Hs(Secph||Rand1)$. Then Phy sends the PhID, PhCred to GaW via the secure channel.

Step 2: The GaW stores the Phy information after receiving the request PhID, PhCred for future use. Following that, the GaW has the public key(GPub) and the master key(Gkey) computes $X = GPub \oplus Hs(Gkey||Pubph)$, $Y = X \oplus Hs(Gkey \oplus PhCred)$. Thereafter, the GaW sends the value X, Y and stores the values in itself PhID. The Phy stores X, Y received from GaW.

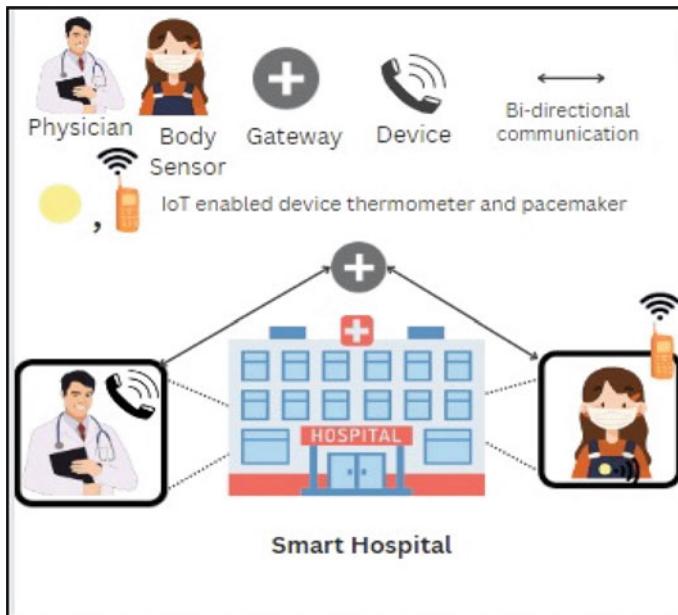


Fig. 1 SLMA architecture

3.2 BS Registration Phase

Step 1: BS has a sensor number assigned to it and generates the random number Rand2 which is \oplus red to create $\text{BSID} = (\text{SN}(\text{Sensor Number}) \oplus \text{Rand2})$ and sends BSID to the GaW via the secure channel.

Step 2: GaW has master key (Gkey) and generates Rand3, upon reception, GaW stores the BSID , computes $Q = \text{Rand3} \oplus H_s(\text{Gkey} \parallel \text{BSID})$, $\text{RBSID} = \text{BSID} \oplus \text{Rand2}$. Thereafter, the GaW stores RBSID and BSID. Finally, the GaW sends RBSID to the BS via the secure channel.

Step 3: The BS obtains the message and stores RBSID which is received from GaW.

3.3 Authentication Phase

Step 1: Phy inputs ID, password and biometric in smart device $\text{Secph}^* = \text{Rep}(\text{PhBio}, \text{PubPh})$. PhCred^* is also computed and matched with stored PhCred value to verify phy.

Step 2: Phy generates Rand4 and Timestamp T1. Phy computes: $A = Hs(Y||PhID) \oplus Rand4$, $B = Hs(T1||Rand4||Y||GPub)$, and $C = A \oplus B$ and Phy sends $\langle T1, A, C, PhID \rangle$ TO GaW.

Step 3: Gateway checks freshness of message and computes: $D = Rand2 \oplus RBSID \oplus Rand4$, $E = PhID \oplus Hs(RBSID||T2||GPub)$, $Session_key = Hs(PhID||RBSID||Rand4||Rand4||Ts)$, and $F = Hs(Session_Key||T2)$. The, Gaw sends D, E, and F to BS.

Step 4: BS checks freshness of received message. BS computes: $Rand4^* = Rand2 \oplus RBSID \oplus D$, $PhID^* = E \oplus Hs(RBSID||T2||GPub)$. It generates T3 and computes: $G = Rand4^* \oplus PhID \oplus Rand2$, $H = RBSID \oplus Hs(PhID||T3||GPub)$, $I = Hs(Session_Key||T3)$, and calculates $Session_key = Hs(PhID||RBSID||Rand2||Rand4||Ts)$.

Step 5: BS sends $\langle G, H, I \rangle$ to Phy along with T3. Phy checks freshness of received message and computes: $Rand2^* = Rand4 \oplus PhID \oplus G$, $RBSID = H \oplus Hs(PhID||T3||GPub)$, and computes $Session_key = Hs(PhID||RBSID||Rand2||Ts)$.

4 Security and Performance Analysis of SLMA

We used a trustworthy and reliable tool called AVISPA to validate the security objectives of SLMA [16, 17, 20–26]. The current evaluation includes two backends, OFMC, and CL-AtSe and presented via Figs. 2 and 3.

Table 1 compares the security features of SLMA and comparative schemes, where ♦: supports and •: does not support. Table 2 helps analyze the number of computational primitives used in respective schemes. Hash functions (Hash Count), XOR operations (XOR), and elliptic point multiplication (epm) are the majorly used primitives. SLMA has significantly reduced XOR operations than counterpart schemes.

5 Conclusion

This research proposes an SLMA in which the protocol employs a secure session for the authenticated users and utilizes patient biometrics via a fuzzy extractor, which helps prevent unauthorized access to the patient sensor gateways. SLMA scheme relies heavily on primitive of lightweight cryptography like XOR and hash operations reduce the burden at the IoT node. Furthermore, we illustrate that the SLMA is SAFE over AVISPA and also has lower computational and communication costs, resulting in greater efficiency than conventional protocols.

```
SUMMARY
SAFE

DETAILS
BOUNDED_NUMBER_OF_SESSIONS
TYPED_MODEL

PROTOCOL
/home/span/span/testsuite/results/PBP.if

GOAL
As Specified

BACKEND
CL-AtSe

STATISTICS

Analysed    : 0 states
Reachable   : 0 states
Translation: 0.05 seconds
Computation: 0.00 seconds
```

Fig. 2 SLMA result on CL-AtSE

```
% OFMC
% Version of 2006/02/13
SUMMARY
SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
/home/span/span/testsuite/results/PBP.if
GOAL
as_specified
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.00s
searchTime: 0.36s
visitedNodes: 4 nodes
depth: 2 plies
```

Fig. 3 SLMA result on OFMC

Table 1 Comparison based on security features

Achieved security features	[6]	[16]	[17]	SLMA
Man-in-the-middle attack	♣	•	♣	♣
Impersonation attack	♣	♣	♣	♣
Session key agreement and its security	♣	♣	♣	♣
Replay attack	♣	♣	♣	♣
Data privacy	♣	•	•	♣
Collision attack	•	•	•	♣
Identity anonymity	♣	•	♣	♣
Forward secrecy	•	♣	♣	♣
Backward secrecy	•	•	♣	♣
Eavesdropping attack	•	•	•	♣
Integrity	•	•	•	♣

Table 2 Comparison based on primitives

Schemes	Number of primitives
[6]	9 Hash, 27 XOR
[16]	30 Hash, 29 XOR
[17]	23 Hash, 9 XOR, 20epm
SLMA	13 Hash, 16 XOR

References

- Attir A, Naït-Abdesselam F, Faraoun KM (2023) Lightweight anonymous and mutual authentication scheme for wireless body area networks. Comput Netw 224:109625
- Narwal B, Mohapatra AK (2021) SAMAKA: secure and anonymous mutual authentication and key agreement scheme for wireless body area networks. Arab J Sci Eng 46(9):9197–9219
- Braeken A (2020) Highly efficient symmetric key based authentication and key agreement protocol using Keccak. Sensors 20(8):2160
- Gaba GS, Kumar G, Monga H, Kim TH, Liyanage M, Kumar P (2020) Robust and lightweight key exchange (LKE) protocol for industry 4.0. IEEE Access 8:132808–132824
- Ryu J, Kang D, Lee H, Kim H, Won D (2020) A secure and lightweight three-factor-based authentication scheme for smart healthcare systems. Sensors 20(24):7136
- Masud M, Gaba GS, Choudhary K, Hossain MS, Alhamid MF, Muhammad G (2021) Lightweight and anonymity-preserving user authentication scheme for IoT-based healthcare. IEEE Internet Things J 9(4):2649–2656
- Wang S, Zhou X, Wen K, Weng B, Zeng P (2022) Security analysis of a user authentication scheme for IoT-based healthcare. IEEE Internet Things J
- Das S, Namasudra S (2022) A lightweight and anonymous mutual authentication scheme for medical big data in distributed smart healthcare systems. IEEE/ACM Trans Comput Biol Bioinform
- Zhang Y, He D, Vijayakumar P, Luo M, Huang X (2023) SAPFS: an efficient symmetric-key authentication key agreement scheme with perfect forward secrecy for industrial internet of things. IEEE Internet Things J
- Praveen R, Pabitha P (2023) A secure lightweight fuzzy embedder based user authentication scheme for internet of medical things applications. J Intell Fuzzy Syst (Preprint) 1–20

11. Kim K, Ryu J, Lee Y, Won D (2023) An improved lightweight user authentication scheme for the internet of medical things. *Sensors* 23(3):1122
12. Masud M, Gaba GS, Alqahtani S, Muhammad G, Gupta BB, Kumar P, Ghoneim A (2020) A lightweight and robust secure key establishment protocol for internet of medical things in COVID-19 patients care. *IEEE Internet Things J* 8(21):15694–15703
13. Farash MS, Turkanović M, Kumari S, Hölbl M (2016) An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment. *Ad Hoc Netw* 36:152–176
14. Amin R, Islam SH, Biswas GP, Khan MK, Leng L, Kumar N (2016) Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks. *Comput Netw* 101:42–62
15. Izza S, Merazka F, Benssalah M (2022, October) Lightweight authentication schemes for internet of medical things. In: 2022 2nd International conference on advanced electrical engineering (ICAEE). IEEE, pp 1–6
16. Chen CM, Chen Z, Kumari S, Lin MC (2022) LAP-IoHT: a lightweight authentication protocol for the internet of health things. *Sensors* 22(14):5401
17. Hegde M, Rao RR, Nikhil BM (2022) DDMIA: distributed dynamic mutual identity authentication for referrals in blockchain-based health care networks. *IEEE Access* 10:78557–78575
18. Narwal B, Mohapatra AK (2020) SEEMAKA: secured energy-efficient mutual authentication and key agreement scheme for wireless body area networks. *Wirel Pers Commun* 113(4):1985–2008
19. Yu S, Park K (2022) ISG-SLAS: secure and lightweight authentication and key agreement scheme for industrial smart grid using fuzzy extractor. *J Syst Architect* 131:102698
20. Narwal B, Mohapatra AK (2021) A survey on security and authentication in wireless body area networks. *J Syst Architect* 113:101883
21. Sharma M, Narwal B, Anand R, Mohapatra AK, Yadav R (2023) PSECAS: a physical unclonable function based secure authentication scheme for Internet of Drones. *Comput Electr Eng*. <https://doi.org/10.1016/j.compeleceng.2023.108662>
22. Narwal B, Mohapatra AK, Usmani KA (2019) Towards a taxonomy of cyber threats against target applications. *J Stat Manag Syst* 22(2):301–325
23. Malik M, Gandhi K, Narwal B (2022) AMAKA: anonymous mutually authenticated key agreement scheme for wireless sensor networks. *Int J Inform Sec Privacy (IJISP)* 16(1):1–31
24. Narwal B, Gandhi K, Anand R, Ghalyan R (2022, September) PUASIOT: password-based user authentication scheme for IoT services. In: Proceedings of the 6th international conference on advance computing and intelligent engineering (ICACIE 2021). Springer Nature Singapore, Singapore, pp 141–149
25. Narwal B, Bansal V, Dahiya V, Aggarwal P (2021) SLUASCIoT: a secure and lightweight user authentication scheme for cloud-IoT services. In: 2021 5th International conference on information systems and computer networks (ISCON), Mathura, India, pp 1–5. <https://doi.org/10.1109/ISCON52037.2021.9702456>
26. Narwal B, Mohapatra AK (2018) Performance analysis of QoS parameters during vertical handover process between Wi-Fi and WiMAX networks. In: Data science and analytics: 4th international conference on recent developments in science, engineering and technology (REDSET 2017), Gurgaon, India, 13–14 Oct 2017, Revised Selected Papers, vol 4. Springer Singapore, pp 330–344

Cybersecurity: A Deep Learning Model for Intrusion Detection in IoT



Abhijeet Singh, Achyut Mishra, Ajit Antil, Bharat Bhushan,
and Anamika Chauhan

Abstract Cybersecurity involves protecting the IoT devices from hackers, viruses, and malwares. This paper aims at building a robust deep learning-based IDS model to encounter cyberattacks. CNN-based IDS model is built on two datasets in which one is modified and other is raw dataset and their accuracies are compared to examine the impact of training datasets used in models. A comparative study of datasets used in previous researches have also been performed. After reviewing previous research papers built on trending datasets, it was found that most of the research lacks quality of dataset used in them. A CNN-based IDS model was built and trained on most recent publicly available CSE-CIC-IDS 2018 dataset and got an accuracy of 92%. The same model was trained on modified CSE-CIC-IDS 2018 dataset, the model achieved an accuracy of 94.67% which is better than the accuracy of the model of earlier not modified dataset. Thus, it proves that through proper data refining and extraction techniques, NIDS model used for cyberattacks can be made more precise and accurate.

Keywords Cyberattack · Cybersecurity · Intrusion detection system · AIDS · NIDS · Machine learning · CSE-CIC-IDS 2018

A. Singh · A. Mishra · A. Antil (✉) · B. Bhushan · A. Chauhan
Delhi Technological University, New Delhi, India
e-mail: ajitantil_2k19ee026@dtu.ac.in

A. Singh
e-mail: abhijeetsingh_2k19ee008@dtu.ac.in

A. Mishra
e-mail: achyutmishra_2k19ee017@dtu.ac.in

B. Bhushan
e-mail: bharat@dce.ac.in

A. Chauhan
e-mail: anamika@dtu.ac.in

1 Introduction

In an advancing world with increasing use of technology and Internet, most of the work is done using computers and IoT devices, which has resulted in rise of cyberattacks. In the past various industries faced huge losses due to cyberattacks on servers, therefore the research on developing an efficient IDS for cybersecurity is at its peak. The Internet of Things (IoT) refers to interconnected devices that can exchange information and can be monitored and regulated remotely. With the increase in digitalization of work and services, the usage of IoT devices is increased, therefore cyberattacks are also increasing; hence cybersecurity is a crucial aspect of IoT applications. To secure communications, services provided by IoT technologies, IoT intrusion detection systems (IDS) need to be developed. As mentioned in [1], in recent years, artificial intelligence (AI) techniques, such as machine learning and deep learning have been used to improve the security of IoT IDS. However, there is lot more to be done in this field as many malware attacks are gone undetected due to lack of robust system and reaction time when a cyberattack occurs. Moreover, the lack of well-extracted and refined datasets is also the major problem for researchers, as many a times a well-performing model trained on publicly available datasets fails in detecting cyberattacks in real-time scenario.

1.1 *Anomaly Intrusion Detection System (AIDS)*

Intrusion detection systems (IDSs) can be created either as signature and anomaly, based on what is normal behaviour and what is anomalous. For anomaly intrusion detection system, we need to develop a model which on its own can determine whether a data packet received to server contains malicious content. To overcome the limitation of signature-based intrusion detection system of detecting new attacks, this technique was designed, as can be seen in study [2]. It is used to detect the new malware attacks which have been occurred recently. Since this technique is focused on recent events, it is also called event-based IDS. To tackle with new cyberattacks a machine learning model is proposed which is trained extensively according to the requirement. As clear from its name anomaly, it checks if there is a deviation from the normal behaviour of a system and if it finds something unusual then it informs the system administrator or host about it. A researcher suggested a host-based IDS using Software Define Technology (SDN) which is based on this method. According to his work there are three crucial things to detect a cyberattack which are unobtrusive approach, negligible overheads, and scalability.

Intruders or hackers attack on the private data of organizations to extract information and causes errors in the system such as overloading of queries which lead to software failure as well as temporary failure of administration, like they set the system custom settings to default settings which can be easily attacked.

2 Datasets Used and Comparison of Datasets

Various datasets have been used for training machine learning models to enhance cybersecurity. Researchers can either utilize publicly available datasets or create their own. The following sections will outline the most commonly used publicly available datasets, comparison of their features is performed to determine the best dataset for our model.

2.1 *KDD Cup99 Dataset*

The Defense Advanced Research Projects Agency (DARPA) created the KDD Cup99 dataset for detecting anomalous network traffic. It was used by Gozde Karatas [3] to build an IDS model, it consists of sixty-three days of LAN raw data in the form of a server connection such as TCP, which achieved result of about 5 million data points. The dataset features 41 properties that can be divided into three categories that are traffic features, basic features, and content features. The data in the dataset can be classified into five main categories: normal (non-attack), Denial of Service attacks, R2L attacks, Probe attacks, and U2R attacks, with a total of 22 different attacks within the four categories and one normal data class. The data includes text-based and numerical details about operations done, which must be processed depending on the goal.

2.2 *NSL-KDD Dataset*

The NSL-KDD dataset was created to overcome the challenges posed by the KDD Cup 99 dataset in terms of data precision and reliability for IDSs. It eliminates duplicate records, offers a homogeneous distribution of data, and features a proportionally distributed number of records in the test and training sets. As seen in [3], the NSL-KDD dataset comprises 42 features in four categories that are general, content, server-based traffic, and time-dependent traffic. There are four categories in which attacks can be divided that are Denial of Service (DoS), U2L, R2L, Probe, with a separate normal/benign category. The NSL-KDD dataset is considered more reliable for IDS evaluations than the KDD Cup 99 dataset.

2.3 *CIC-IDS2017 Dataset*

The CIC-ID2017 dataset was created in 2017 to provide a realistic and up-to-date representation of the current cyberattacks. It was created by analyzing network traffic

data, including timestamp, IP addresses, and ports, as well as attack types. The dataset includes 86 network-related features and is considered a reliable dataset as it meets criteria such as having a complete network structure, complete traffic structure, tagged data, and proportional distribution of common attacks. It is an important resource for evaluating intrusion detection systems.

2.4 CSE-CIC-IDS2018 Dataset

The profile concept was used for creating CSE-CICID2018 dataset and is considered one of the most recent datasets available. The Canadian Institute for Cybersecurity created the dataset in 2018/2019 which aims of giving a reliable and recent resource for intrusion detection. The dataset includes two profiles and five different attack methods, and data was collected daily, with 80 statistical properties calculated. The final dataset, which has approximately five million data in both PCAP and CSV format, is available to researchers on the Internet. The CSV format is recommended for use with AI techniques, while unprocessed PCAP data is ideal for extracting new features.

2.5 Comparison of Datasets

Table 1 compares the various datasets used in past by researchers. It was found out that CSE-CIC-IDS2018 is by far the most recent datasets and has many features making it a good dataset for training purposes.

As shown in Fig. 1 previous research has extensively utilized datasets such as KDD Cup1999 and NSL-KDD, however, these datasets are outdated due to advancements in technology and do not accurately reflect the current network environment and the more sophisticated attacks that have emerged. Consequently, machine learning

Table 1 Comparison of features of datasets

Dataset	Real traffic	Label data	IoT traces	Zero-day attacks	Full packets captured	Year
KDDCUP 99 [7]	✓	✓	✗	✗	✓	1999
NSL-KDD [8, 9]	✓	✓	✗	✗	✓	2009
CICIDS2017 [10, 11]	✓	✓	✗	✓	✓	2017
CICIDS2018 [12, 13]	✓	✓	✗	✓	✓	2018

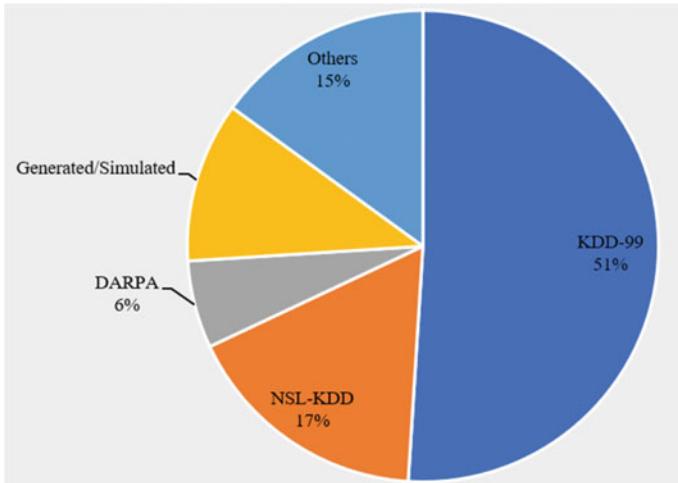


Fig. 1 Datasets used over the past decade for NIDS

models based on these datasets are no longer relevant. Therefore, the more recent datasets are needed.

3 Proposed Model

3.1 Modifications in CSE-CIC-IDS Dataset

Existing datasets used for training intrusion detection models in industrial IoT had several flaws [15]. The CSE-CIC-IDS2018 dataset, for instance, was found to be lacking in many areas and had not been properly scrutinized by researchers. Therefore, we built a CNN model on modified CSE-CIC 2018 dataset. According to Gints Engelen [4] the modifications in this dataset are:

Flaws in extraction of TCP connection: In CSE-CIC datasets as explained in [4], the extraction of a dataset was done using a CICFlowMeter and it was found that the CICFlowMeter separated a single TCP connection into two flows as shown in Fig. 2 and after that both the flows were labelled for same attack type. As shown in Fig. 2 Flow 2 does not correctly represent the behaviour of attacks, thus making it unsuitable for training purposes. These types of incorrect labels constituted more than 20% of raw datasets. Therefore, changes in CICFlowMeter were made to correctly identify a complete TCP connection in one flow.

Attacks after server became unresponsive: According to [4] there are many attacks which represent the false information in the original raw dataset, it was found out that the original CSE-CIC-IDS datasets also included those attacks which did not

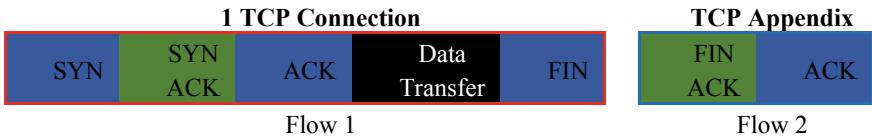


Fig. 2 TCP flow construction

contribute to the victim system shut down. It was noticed that when the attacker's server attacks victim computer, in this process the attacker's system sends a connection request to victim computer and if victim system accepts the request, then it sends a connection signal back to attacker system thus building a two-way connection. But sometimes, even when the victim system is brought down or shut down, then also attacker system sent requests unknowingly of the status of victim software; thus, these connection requests are not accepted by victim system as it is already shut down, therefore only a one way connection occurred, so these types of attacks attempts by attacker server are not to be considered. So, in the modified datasets these types of attacks are relabelled with 'attempted' as a suffix, which can be removed while training or can be relabelled as 'benign'.

As we can see these modifications are fundamental and these can challenge the models built earlier using the original CSE-CIC-IDS 2018 dataset as the accuracies of models achieved using these datasets do not represent the actual performance, therefore we choose to build a CNN model of our own so that it can be used as a basis for further research.

3.2 Exploratory Data Analysis

We had a lot of data available to deal with in the dataset, so we performed analysis, preprocessing and modelling on it by combining all of them and concluded the results at the end. For making a proper understanding of dataset we were using, we performed a brief exploratory data analysis (EDA).

The EDA is sub-divided into:

Data Visuals: We had a total of 2 million+ samples and 91 features in data and there were a few missing values which needed to be filled or dropped for proper modelling.

Data Understanding: We imported various libraries such as NumPy, Pandas, Kera's, and Scikit-learn. These libraries are used for data processing, machine learning, and visualization tasks. Most of the network intrusions in our data are BENIGN, Portscan, DoS Hulk, DDoS, and Infiltration—Portscan, so we dropped all the remaining labels which did not have a significant count. After dropping the less significant network intrusions we obtained these five network intrusions in our datasets as shown in Fig. 3.

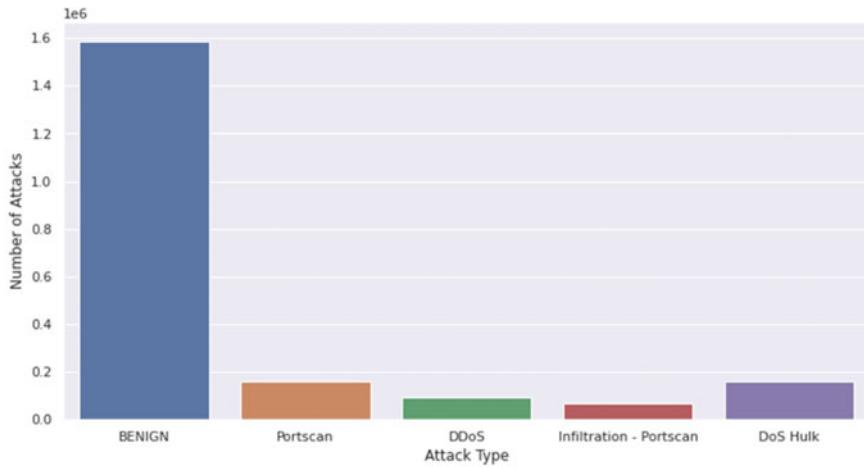


Fig. 3 Values of most commonly used attacks

To maintain the uniformity in the number of attacks and to shape our dataset for our deep learning model we took 20,000 samples for each of the above-mentioned labels as depicted in Fig. 4.

Data Analysis: We checked for null or missing values in the dataset using the `isna()` function and removed them using the `dropna()` function, then encoded the dataset using the `LabelEncoder()` function to convert the labels into numerical values. After that the dataset was split into five different dataframes based on the label values using boolean indexing. Each dataframe is assigned a unique numerical value using a NumPy array. The five dataframes are merged into one dataframe using `pd.concat()` function. The features and labels are separated into `X` and `y` dataframes, respectively.

3.3 Advantages of Deep Learning Techniques

The method of detecting attacks on IoT devices has evolved over time. Traditional techniques involve collecting information about all past attacks and searching for similar patterns in network connections. However, these methods are not effective for detecting zero-day attacks, which are unknown to the system. To address this issue, machine learning models are used to detect both previously known and novel attacks. A significant amount of research [5] has already been conducted in this area.

Machine learning involves utilizing algorithms to study processed datasets, and producing a model that can detect both established and new attacks. The accuracy and results of the model depend on the quality and behaviour of the technique used. While some algorithms fall under the umbrella of machine learning, they have limited learning capacities and struggle to handle large datasets effectively.

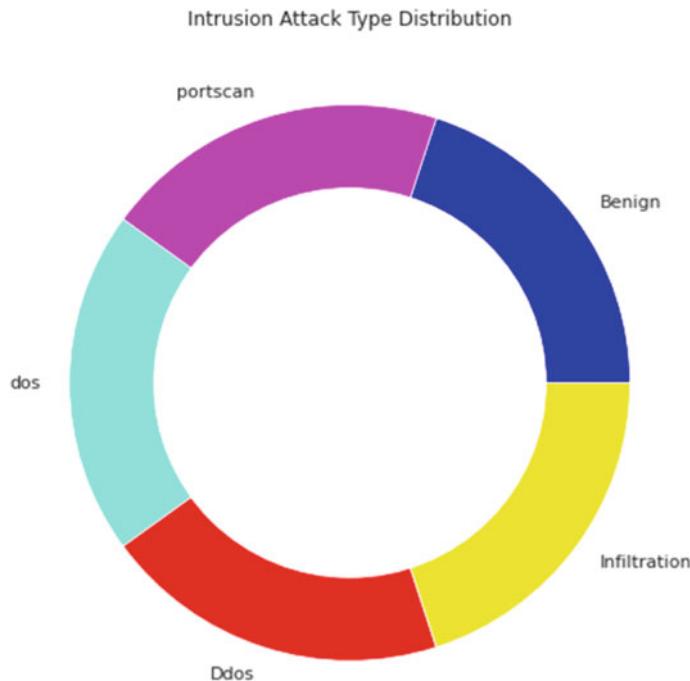


Fig. 4 Distribution of various intrusion attack types

Therefore, deep learning techniques, which offer faster learning rates, have become increasingly popular. This is especially relevant for industrial IoT applications [14], where network traffic is high, and demand for deep learning methods is on the rise in industrial IoT. Therefore, we used CNN deep learning technique in our model.

3.4 Convolutional Neural Network

CNN is a particular kind of artificial neural network (ANN), which is readily used for identification and picking specific features of object and pictures. CNN is inspired from human brain structure. Nodes in CNNs or neurons collect inputs, analyze them, and throw the result as output, just like the brains neurons does when it distributes information all along the body. Pictures are the input here. Picture pixels as an array are accepted by the input layer. Several hidden layers may be present in CNN which uses mathematics for extraction of features from the picture. Few examples of hidden layers are pooling, completely linked layers, convolution, and rectified linear units. Convolution is the first layer for extraction of features from input. Output layer identifies and classifies the item using the fully connected layer. As information

flows only from inputs to outputs in one direction, so CNNs are called feedforward networks. Artificial neural networks (ANN) and CNNs both are focused on biological principles. Brains' visual cortex drives their construction, that is constructed using alternating layers of sophisticated and basic cells. Even though many types of CNN designs are present, all of them usually include convolutional and pooling layers which are structured as modules.

Convolution Layer: The Kernel: Convolutional layers operational goal is to identify and extract high-level features or characteristics of the input image like edges and colour. ConvNets necessarily should not have only one convolutional layer. The first ConvLayer captures the low-lying characteristics like gradient direction, edges, colour, etc. The architecture of the CNN adjusts to the high-level characteristics by adding more layers, giving us a network, which grasps the dataset's pictures totally in a way that is similar to how our brains do. The procedure outputs two different sorts of results: one of the results where the dimensionality is decreased when compared with the input of the convolved feature, the other one where it is either intensified or remains as usual. Valid padding is used to do this.

Pooling Layer: The pooling layer is responsible for shrinking of convolved features. As per Alzubaidi, [5] decrease in dimensionality reduction results in the decrease in amount of computing required to process the data. Additionally, the rotational and positional invariant characteristics which are dominating in nature can be extracted through this, which can help in better training of the model. There are two different forms of pooling: max pooling and average pooling. Max pooling returns the largest value in the area of picture covered by kernel.

Average pooling returns the average of all values under the area of picture covered by kernel. Moreover, the max pooling layer also works as a noise suppressant. It does de-noising and completely discards the noisy activations.

3.5 Proposed Deep Learning Model

The model is designed to take as input a one-dimensional array of 80 data points and output a classification of five possible classes. The model contains three sets of Convolutional, BatchNormalization, and MaxPooling layers, each layer is having a dense layer and a ReLU activation. The final output layer contains five neurons with softmax activation to provide the chances of the input to each of the five classes.

The first set of Convolutional, BatchNormalization, and MaxPooling layers contain 64 filters with kernel size 6, and activation function ReLU. BatchNormalization is used to normalize the result of the previous layer, which is then passed to the MaxPooling layer with pool size 3 and stride 2, and same padding. The second and third sets of layers are identical to the first set, except that they take the result of the previous MaxPooling layer as input.

The output of the final Convolutional layer is converted by a flatten layer into a one-dimensional array, which is then passed to two dense layers with ReLU activation containing 576 and 64 neurons, respectively. The final dense layer contains five neurons with softmax activation for the classification of the input into the five possible classes.

The evaluate () method is used to evaluate the model's performance on a test dataset x_{test} and y_{test} , and the accuracy score is printed. The history object stores the training and validation loss and accuracy of the model at each epoch, which can be visualized using plotting libraries to assess the model's performance during training.

Overall, the model represents a standard CNN architecture for classification tasks, with three sets of Convolutional, BatchNormalization, and MaxPooling layers followed by dense layers and a softmax output layer. The use of BatchNormalization helps to improve the model's performance by reducing internal covariate shift and improving the model's generalization ability. To train the model, we use categorical_crossentropy as the loss function, and Adam optimizer with accuracy as the evaluation metric. Finally, we evaluate the model on the test set and print the accuracy score.

4 Results

4.1 Confusion Matrix

In this research, we have implemented CNN on modified CSE-CIC-IDS 2018 dataset and obtained the following results. The confusion matrix was used to determine the performance of the intrusion detection model. A confusion matrix is a table that shows the credibility of a classification model by the actual and predicted labels of the dataset. The matrix contains true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values, which can be used to calculate various performance metrics such as accuracy, precision, recall, and F1-score. The confusion matrix was visualized using a heatmap plot with labels for each class.

Figure 5 shows how well the model predicted the actual labels and can be used to identify which classes the model performed well or poorly on.

4.2 Evaluation Matrix

The model was evaluated on accuracy, precision, recall metrics, f1-score, and detection rate. Table 2 shows the percentage of accuracy, precision, f1-score, and detection rate.

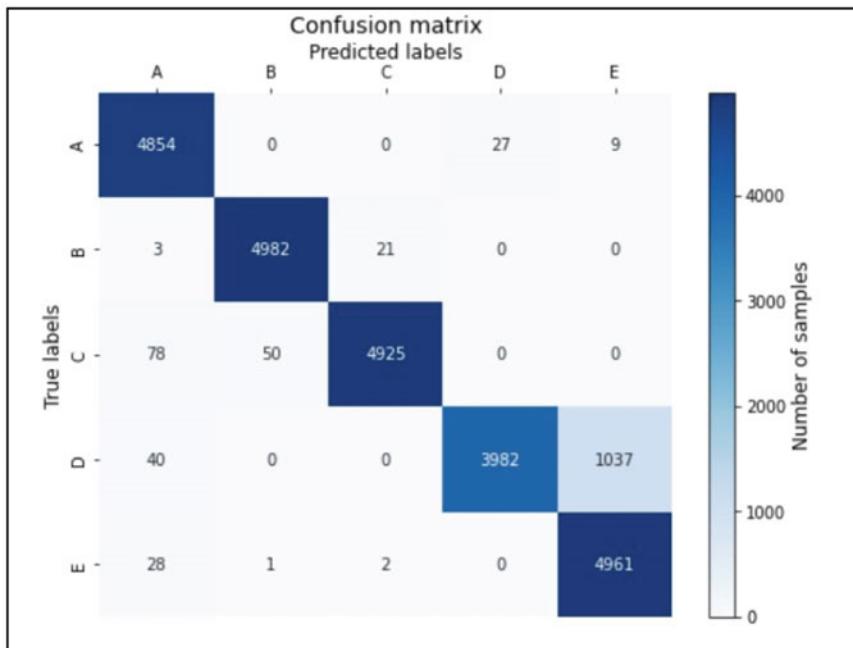


Fig. 5 Confusion matrix of attacks

Table 2 Percentage of accuracy, precision, F1-score, and detection rate

Evaluation matrix	Value (in %)
Accuracy	94.82
Detection rate	94.87
Precision	95.49
F1-score	94.78

4.3 Model Accuracy

Model accuracy [6] is a measure of how well a machine learning model is performing in making correct predictions on a given set of data. It is the ratio of the number of correct predictions made by the model to the total number of predictions made. A higher accuracy indicates that the model is making more correct predictions, while a lower accuracy indicates that the model is making more incorrect predictions. Figure 6 shows the accuracy of model over every epoch, batch size used for this training is 32. Since model accuracy shows after 20 epochs the accuracy is nearly constant, therefore our model is neither overfitted nor underfitted.

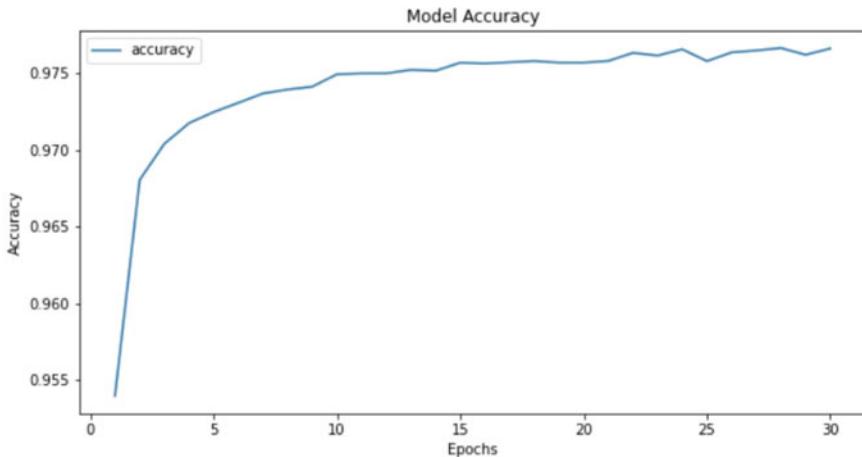


Fig. 6 Accuracy of model for 0–30 epochs

4.4 Model Loss

Model loss tells about how good a machine learning model can reduce the gap between the actual output and the resulted output of the data during training. The loss function is used to find this parameter, the loss function takes the predicted output of the model and compares it to the actual output. The difference between the resulted and actual outputs is called the error or the loss. The aim of the model during training is to reduce the loss function, which means to make the resulted output as close as possible to the actual output. As can be seen in Fig. 7 the loss percentage is very low after 20 epochs and has been nearly constant after that, thus it shows that our model has achieved very low percentage.

5 Conclusion and Future Scope of Study

The comparative study of datasets used in previous research revealed that there is a lack quality of dataset used in them. Therefore, a CNN-based IDS model was built on most recent modified CSE-CIC-IDS2018 dataset. An accuracy of 94.67% was achieved which is greater than the accuracy of model trained on publicly available dataset. Also, confusion matrix shows attack detection rate of model is very high which shows that this model is efficient in detecting the cyberattacks. Accuracy of model is nearly constant after 20 epochs and achieved a maximum accuracy at around 30 epochs, therefore this model is neither underfitted nor overfitted.

Model trained on the publicly available CSE-CIC-IDS 2018 dataset which was not modified and carried a lot of non-useful entries gave an accuracy of 92% which is less

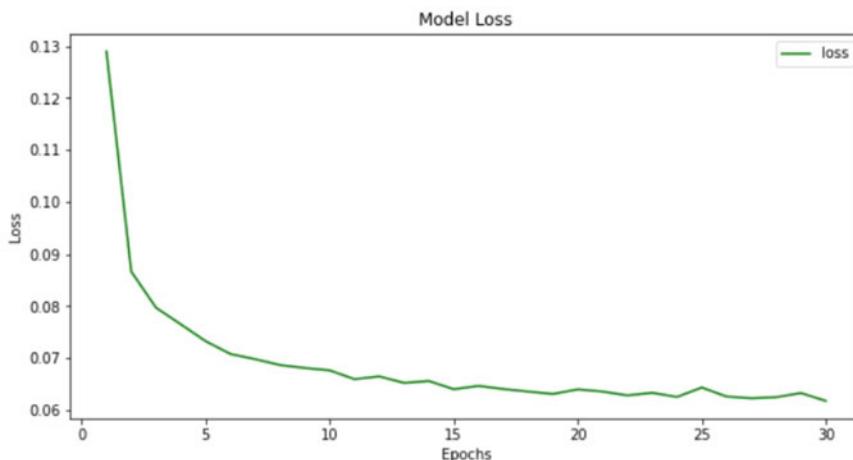


Fig. 7 Model loss from 0 to 30 epochs

than the accuracy of the model trained on new modified dataset. Thus, a better accuracy was achieved using a modified dataset. This shows that there is a lot of potential in improvement of data refining and extraction techniques. In future this model can be implemented on various previously available datasets, and through a comparative study on their results effectiveness of modified datasets can be highlighted.

References

1. Khraisat A, Alazab A (2021) A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges. *Cybersecur* 4:18
2. Thakkar A, Lohiya R (2021) A survey on intrusion detection system: feature selection, model, performance measures, application perspective, 23 challenges, and future research directions. *Artif Intell Rev* 1–111
3. Karatas G, Demir O, Sahingoz OK (2020) Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset. *IEEE Access* 8
4. Engelen G, Lu L (2022) Error prevalence in NIDS datasets: a case study on CIC-IDS-2017 and CSE-CIC-IDS-2018. In: IEEE conference on communications and network security (CNS)
5. Alzubaidi L, Zhang J, Humaidi AJ et al (2021) Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *J Big Data* 8:53
6. Alsaleh D, Larabi-Marie-Sainte S (2021) Arabic text classification using convolutional neural network and genetic algorithms. *IEEE Access* 9:91670–91685
7. Tavallaei M et al (2009) A detailed analysis of the KDD Cup 99 data set. In: 2009 IEEE symposium computational intelligence for security and defense applications, pp 1–6
8. Thakkar A, Lohiya R (2021) Attack classification using feature selection techniques: a comparative study. *J Ambient Intell Humaniz Comput* 12(1):1249–1266
9. Al-Emadi S, Al-Mohannadi A, Al-Senaid F (2020) Using deep learning techniques for network intrusion detection. In: 2020 IEEE international conference on informatics, IoT, and enabling technologies (ICIoT). IEEE, pp 171–176

10. Ahmim A, Ferrag MA, Maglaras L, Derdour M, Janicke H (2020) A detailed analysis of using supervised machine learning for intrusion detection. In: Strategic innovative marketing and tourism. Springer, Cham, pp 629–639
11. Ahmim A, Maglaras L, Ferrag MA, Derdour M, Janicke H (2019) A novel hierarchical intrusion detection system based on decision tree and rules-based models. In: 2019 15th International conference on distributed computing in sensor systems (DCOSS). IEEE, pp 228–233
12. Bharati MP, Tamane S (2020) NIDS-network intrusion detection system based on deep and machine learning frameworks with CICIDS2018 using cloud computing. In: 2020 International conference on smart innovations in design, environment, management, planning and computing (ICSIDEMPC), Aurangabad, India, pp 27–30
13. Baydogmus GK, Demir Ö, Sahingoz O (2020) Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset. IEEE Access 1–1. <https://doi.org/10.1109/ACCESS.2020.2973219>
14. Xiao Y et al (2019) An intrusion detection model based on feature reduction and convolutional neural networks. IEEE Access 7:42210–42219
15. Karatas G, Demir O, Sahingoz OK (2020) Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset. IEEE Access 8

Performance Analysis of AES and DES Algorithm for Encrypting Medical Record Using Blockchain



J. A. Madhurya and K. Meena

Abstract Secure storage and exchange of health information has emerged as a focus of research in the area of medical technology. The healthcare sector has a high requirement for data security and privacy because most of the healthcare systems maintain data in a centralized manner, storing it all in a single location. Such systems have a significant probability of allowing an anonymous user to access and alter the data, which frequently results in data leakage problems and lack of transparency. In the proposed work, the blockchain technology is used which is a decentralized and distributed ledger that enhances the data security by storing the record on blocks and encryption algorithm to encrypt the data. Here patient is allowed to provide the access permission for the record. An advanced encryption standard (AES) and data encryption standard (DES) algorithm is used for encryption and its encryption time is compared and represented in graph. The objective of this work is to do the performance analysis between algorithms and to represent the faster encryption algorithm.

Keywords Blockchain · Encryption · Medical record · Data leakage

1 Introduction

The development of technology over the past few decades has affected many facets of our life. It helped us, including in the field of medicine. Numerous industries, including the healthcare sector have seen tremendous advancements in recent years. Medical information relates to the patient's vital privacy and health. It has higher requirements for usability, security, and integrity than other types of data. Hospitals have their own record management system for storing patient data. Some medical

J. A. Madhurya (✉) · K. Meena

Department of Computer Science and Engineering, GITAM School of Technology, GITAM University, Bangalore, India
e-mail: mambuja@gitam.edu

K. Meena
e-mail: mgurupar@gitam.edu

facilities use their own catalogs or any cloud service provider to store patient data [1]. The majority of the time, the hospital owns or rents the servers where the data is stored. Despite the fact that the records belong to the patients, they do not have easy access to their own health information. Centralized servers were set up to store the data in order to improve information access and sharing.

Certain limitations of centralized servers are:

- Records once stored on cloud are maintained by third party organizations.
- There is no control over the data once it has been sent to the cloud. The whole trust depends on the cloud server because the patient is unable to monitor the users who are viewing the data.
- Malicious clinical organizations may work along with the cloud server and may modify the medical information which is stored in cloud that may lead to data integrity [13].

1.1 Need of Blockchain Technology

Due to certain limitations of centralized server and to overcome it, there is a need for decentralized secured mechanism for secure access of data. We employ blockchain technology to get a decentralized method. Blockchain is a particular type of data structure developed in a peer-to-peer network setting that is built on transparent, dependable consensus rules, and chronologically merging data blocks into chains. It is a shared distributed ledger system that can be trusted and is cryptographically guaranteed to be unchangeable, traceable, and unforgeable.

The blockchain has multiple features as follows:

- **Transparency:** Any user may contribute to add or do validation of a block on a public blockchain. All users have access to any transaction or block uploaded to the blockchain. Only private authorized people have access to the data on a private blockchain. Even though a user's true identity is protected, it is easy to trail the transactions made by a user.
- **Immutability:** The inability to change or tamper a block after it has been added to a public or private blockchain. Since each block contains the hash value of the preceding block, any modification to one value would influence the validity of all successive blocks [14]. Furthermore, every user of the network has a replica of the blockchain, any conflicts between copies could easily be identified.
- **Traceability:** To verify and trace the data recorded in blockchain is easy due to the existence of the nonce and the data is recorded along with a time.
- **Reliability:** The blockchain's data storage system is built on a variety of cryptographic algorithms, including hashing and encryption. As a result, the data contained within the blocks of a blockchain is genuine and can be trusted.
- **Decentralization:** Traditional database systems relied on an outside organization or party to confirm data, but blockchain technology runs independently by using a distributed ledger to verify transactions between nodes.

1.2 *Need of Encryption*

It can increase the security of interactions between two parties and aid in safeguarding private and sensitive data [2]. In essence, even if unauthorized user access to our data, they can't get the complete information as data will be encrypted by using encryption algorithms.

The medical record contains a variety of sensitive information, including social security numbers, age, gender, addresses, physiological data, and others. Privacy breach events will result if these properties are accessed or used maliciously by unauthorized persons. Building a safe and reliable transaction architecture is important, and all patient information should be maintained in encrypted form.

By using blockchain technology, patients' data are stored on blocks by encrypting it using AES and DES algorithm. As various algorithms take different encryption time, we are creating the graph representing to show the time complexity for encrypting the patient data by using abovementioned algorithms.

2 Related Work

Joseph Gabriel and Sengottuvelan [3] proposed a research method called AES encryption security for healthcare data using blockchain. The information is encrypted using the AES technique. The purpose is to secure the communication between patients and doctors so that they can express their opinions and points of view on their health issues and the control will be given to patients so that they have the right to share the record to the requested user.

Christo et al. [4] proposed a work called security for medical record using blockchain technology. In order to provide security for medical record and to enhance the accessibility the work uses authentication algorithm and encryption of record is done by using AES algorithm. To share the data to other user's SHA algorithm is used.

Liang et al. [5] proposed a technique to share the healthcare data in a secured way by using blockchain technology. The proposed work uses a decentralized, permissioned blockchain for sharing health data while maintaining anonymity. To gather health information from wearable personal devices or to manually enter the data, a mobile application is implemented. In order to share data with other healthcare organizations and health insurance companies, data is synchronized with cloud. Each record which is stored in cloud has a proof of integrity.

Zhang et al. proposed a technique called "data sharing in e-health system in a secured way". In the proposed work the health record is stored in private blockchain, while in consortium blockchain the health record index and keyword which will be used for searching the record is stored [6]. In order to achieve security and privacy preservation all the health record, its index and keywords are encrypted using public key [7].

Kamboj et al. [8] proposed a technique for smart contract-based user authentication using blockchain for role-based access control. The proposed approach uses the Ethereum blockchain and smart contract for managing the access control. AES algorithm is used for encryption purpose. Accessibility was provided based on user roles. This model is used to resist various attacks.

Abeywardena et al. [9] proposed a research method called patient details management system using blockchain technology. Here they developed a mobile application where it receives the data from wearable device like Bluetooth and these data are stored in block in encrypted form. This work has shown the comparison between proposed model and existing health record management system.

The medical record has much sensitive information which has to be secured from attacks and unauthorized access. Earlier approaches like storing a data in centralized system use to provide security for the data but had chances of data leakage and user doesn't have the track of who all are accessing their records. The blockchain technology which is a decentralized and distributed ledger enhances the data security by storing the record on blocks and the user can track the record since user will be giving the permission for the requester.

Section 1 summarizes about the introduction of blockchain, importance of blockchain in health care, and summarizes the importance of encryption for confidential information. Section 2 provides the explanation about literature survey. Section 3 explains the proposed method along with the workflow of algorithms with a respective diagram. Section 4 represents the experimental results tabulated in graph followed by conclusion.

3 Proposed Work

The provision of health treatment has become inseparable in everyday life. For instance, their medical information, the patients' treatments, their prior medical record are now crucial components to diagnose patients and to proceed with the therapy.

The medical data was typically recorded on paper, although it might later be altered or destroyed. Therefore, it is crucial to electronically upload and secure the information in the system. Suppose unauthorized user gets accessibility to the data in the record stored system they may damage or destroy entirely. To safeguard patient health data, which includes confidential information, the medical system must have security measures in place [4].

The traditional method is used to protect the data but certain times it had a chance of leakage of patient's data due to unauthorized access. Sometimes the medical data might also be shared with medical practitioners or with other medical organization for various purposes. The main objective is to provide a security for patient's health record and its confidential information. This objective can be achieved by using blockchain technology.

As medical data of a patient includes much sensitive data it has to be securely stored and shared between users and doctor. Hence data has to be encrypted before storing into blocks in blockchain. The safest method of sharing the patient's record is by using blockchain technology as it has a decentralized feature. In the proposed work patient's medical record is stored in blocks by encrypting it by using AES algorithm and the same patient record is encrypted again by using DES algorithm.

Figure 1 depicts the structure of the proposed work. The architecture includes actors such as a patient, a doctor, and the patient's past medical data. Initially the patient will login to the application with their unique ID. New patient has to register by entering their details. Once logged in successfully patient can upload all their medical records. The uploaded records will be encrypted by using AES and DES algorithm and these encrypted records will be stored in respective blocks of blockchain. Since the encryption algorithm generates keys for encryption purpose, these keys are also stored in blocks since it is required for decryption purpose and also the keys can't be misused. Hence the medical record is stored in secured way. The authorized doctor can only access the patient records. Permission to access the records will be given by patient after the verification of doctor ID. Once the doctor is verified as authorized person, key will be shared to decrypt the medical record so that the record can be analyzed for the treatment.

Thus, patient will be having the control over data and track the users who all are accessing it. Different encryption algorithms take variant encryption time for encrypting the same data. The encryption process should take minimal time so that data can be encrypted faster and saved in blocks. In the proposed work AES and DES algorithm is applied to encode the medical information and time consumed by those algorithms is tabulated in Fig. 4.

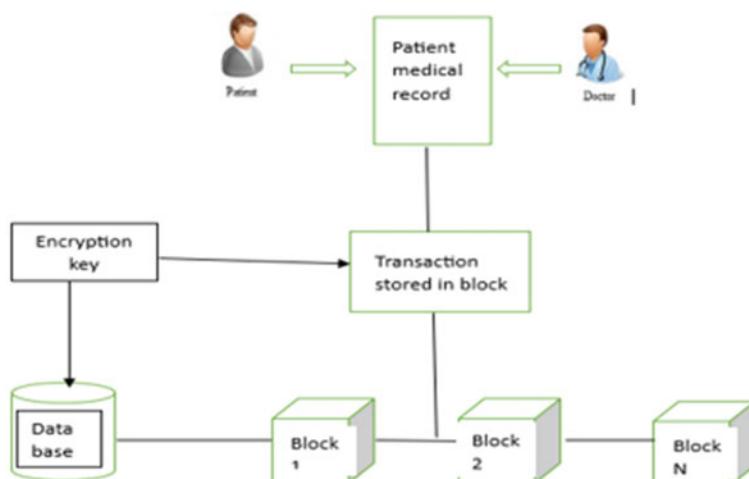


Fig. 1 Transaction using blockchain technology

3.1 Registration of Patient

If the patient is a new user, then they need to register by entering the details. Existing patient can login by entering their login credentials.

```
if (patient==new user)
    then Register with their details
    create unique ID
    login by ID
else
    directly login using unique ID.
```

3.2 Encryption of Medical Record

One of the most commonly used encryption algorithms is AES. For encryption and decryption, AES uses a proprietary structure. Three distinct key sizes like AES 128, 192, and 256 bits can be used with AES, and each of these ciphers has a 128-bit block size. Figure 2 depicts the AES encryption method [3]. In our proposed work AES 256 bit key size is used.

Since DES is a block cipher, it encrypts data in blocks of 64 bit each. As a result, DES receives 64 bits of plain text as input and outputs 64 bits of ciphertext. Both encryption and decryption use the same algorithm and key. The key is 56 bits long. Figure 3 depicts the DES encryption algorithm's workflow.

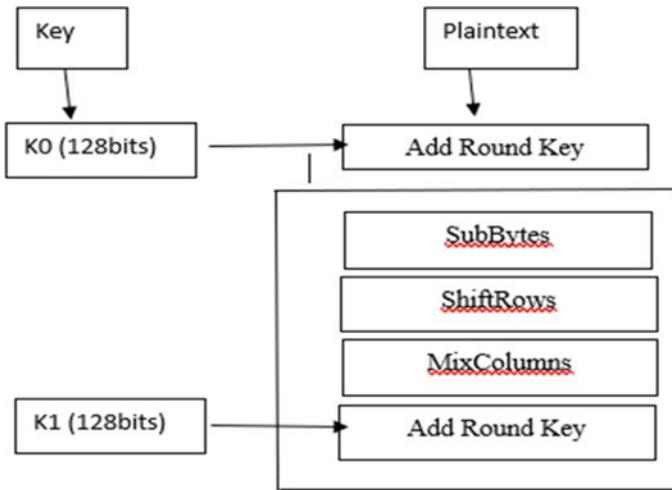


Fig. 2 Encryption process of AES algorithm

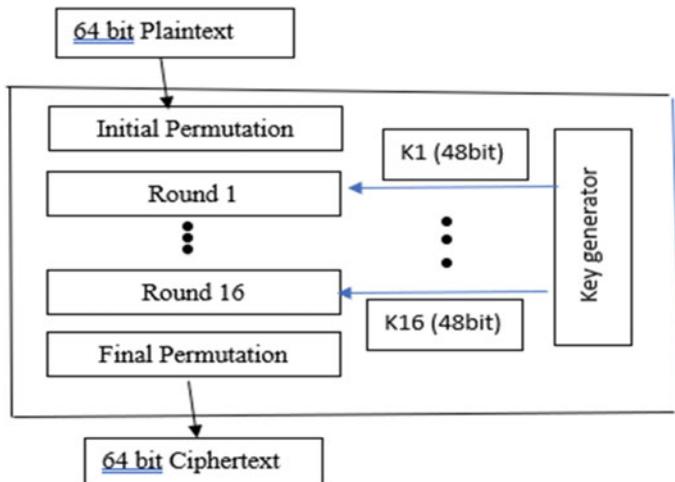


Fig. 3 Encryption process of DES algorithm

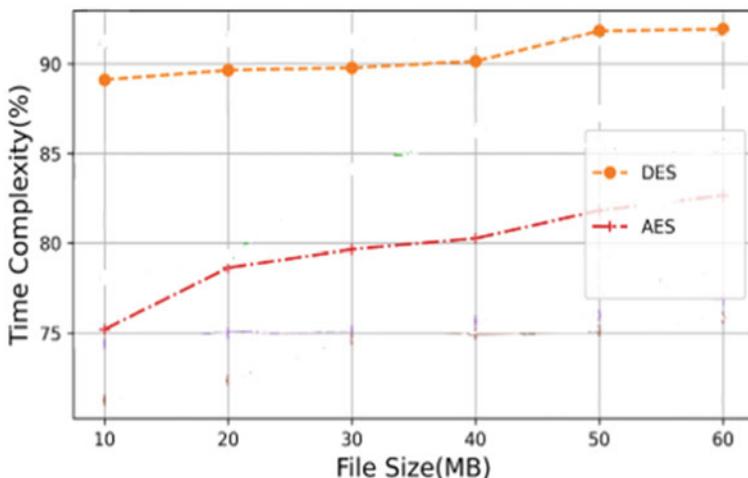


Fig. 4 Time consumption of AES versus DES

3.3 Workflow of AES

The AES encryption process involves dividing the plaintext message into blocks, generating a set of round keys from the secret key using key expansion and performing a series of substitution and permutation operations on the blocks using the round keys. The final output is the encrypted ciphertext.

AES's block length and key length can be independently chosen as 128 bits, 192 bits, or 256 bits, with matching round times of 10, 12, or 14 [15]. A round has four transformations in it, i.e., S-box substitution (ByteSub), ShiftRow, MixColumn, and Add RoundKey. The entire step is explained as follows [10]:

S-Box Substitution

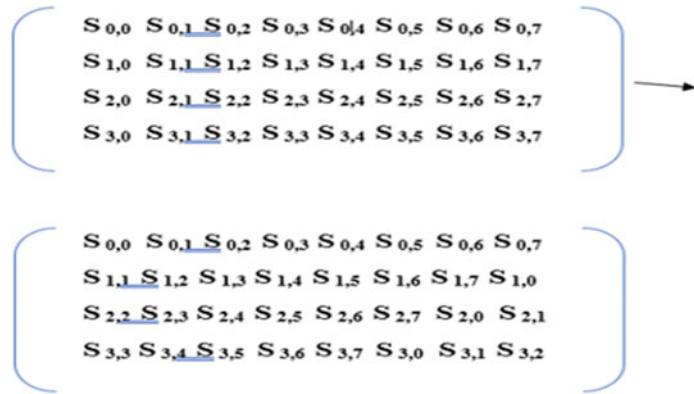
Each of the state bytes is considered independently by the nonlinear byte substitution known as the S-box transformation [11].

Seeking the inverse operation of multiplication

$GF(2^8) = Z_2[x]/(x^8 + x^4 + x^3 + x + 1)$ field, that is, input $w \in GF(2^8)$ output $v \in GF(2^8)$

Shift Row

From substitution we get state matrix of 4×8 byte, where S_{ij} is the byte in the i th row and the j th column, $0 \leq i \leq 3, 0 \leq j \leq 7$, where 0th row no byte will be shifted, first row 1 byte will be shifted, second row 2 bytes will be shifted, third row 3 bytes will be shifted [11], the below representation shows the obtained state matrix and resultant matrix after shifting the bytes.



Mix Column

In order to create confusion, MixColumn changes each column's standalone operation. The new value is mapped to each byte in each column.

$$\begin{aligned}
 \underline{\mathbf{D}} * & \left(\begin{array}{ccccccccc}
 \mathbf{S}_{0,0} & \mathbf{S}_{0,1} & \underline{\mathbf{S}_{0,2}} & \mathbf{S}_{0,3} & \mathbf{S}_{0,4} & \mathbf{S}_{0,5} & \mathbf{S}_{0,6} & \mathbf{S}_{0,7} \\
 \mathbf{S}_{1,0} & \mathbf{S}_{1,1} & \mathbf{S}_{1,2} & \mathbf{S}_{1,3} & \mathbf{S}_{1,4} & \mathbf{S}_{1,5} & \mathbf{S}_{1,6} & \mathbf{S}_{1,7} \\
 \mathbf{S}_{2,0} & \mathbf{S}_{2,1} & \underline{\mathbf{S}_{2,2}} & \mathbf{S}_{2,3} & \mathbf{S}_{2,4} & \mathbf{S}_{2,5} & \mathbf{S}_{2,6} & \mathbf{S}_{2,7} \\
 \mathbf{S}_{3,0} & \mathbf{S}_{3,1} & \underline{\mathbf{S}_{3,2}} & \mathbf{S}_{3,3} & \mathbf{S}_{3,4} & \mathbf{S}_{3,5} & \mathbf{S}_{3,6} & \mathbf{S}_{3,7}
 \end{array} \right) \\
 \text{Where } \mathbf{D} = & \left(\begin{array}{cccc}
 02 & 03 & \underline{01} & 01 \\
 01 & 02 & \underline{03} & 01 \\
 01 & 01 & \underline{02} & 03 \\
 03 & 01 & \underline{01} & 02
 \end{array} \right)
 \end{aligned}$$

Code

```

file1 = open("./enc/AES.txt","w")
start_time = time.time()
with open(dirt, mode ='r')as file:
    csvFile = csv.reader(file)
    for lines in csvFile:
        msg=listToString(lines)
        data=msg.encode("utf8")
        cipher = AES.new(key, AES.MODE_EAX)
        nonce = cipher.nonce
        ciphertext, tag = cipher.encrypt_and_digest(data)
        file1.writelines(str(ciphertext))
        #print(ciphertext)
file1.close()
AESent= (time.time() - start_time)

```

3.4 Workflow of DES

The DES encryption algorithm involves a set of mathematical operations, including substitution, permutation, and XOR.

The DES encryption process involves dividing the plaintext message into blocks of 64 bits, generating a set of round keys from the secret key using key permutation and rotation, and performing a series of substitution and permutation operations on the blocks using the round keys. The final output is the encrypted ciphertext. The entire step is explained below [12]:

Initial Permutation (IP)

1. Plaintext is provided as input to IP and it is permuted.
2. The permuted input block will be split into two halves, each being 32 bits. The first 32 bits are called L , and the next 32 bits are called R .

Expansion (E)

3. Expand R 32 bits to 48 bits to fit the subkey (48 bit) by performing the expansion permutation (E).
4. Perform Exclusive-OR between the subkey and expansion permutation (E) on R . $E(R_i - 1) \oplus K_i$

Substitution

5. Break the result of $E(R_i - 1) \oplus K_i$ into eight blocks like S1 to S8, each containing 6 bits.

Suppose the first 6 bits are like = 010101. So, the input to S1 = 010101. The row value: 0 1 = 1 (decimal). The column value: 1010 = 10 (decimal). Map these row and column number in substitution table and obtain the value.

6. Perform Exclusive-OR with the obtained value from the previous step and L .
7. Perform a 32-bit swap on the result of the final round. Then, perform inverse initial permutation (IP – 1) on the swapped data to produce the ciphertext 64 bits.

4 Experimental Results

The proposed system has been implemented by encrypting the patient data by using AES and DES algorithm. Since DES was the basic symmetric encryption algorithm and in order to understand the working procedure of algorithm AES is compared with DES. Initially the file size was 10 MB, later it was increased to 60 MB and it was observed that as file size increases encryption time taken to encrypt patient record gradually increases and its respective graph representation is shown below in Fig. 4. From this it has been observed that whatever the file size may be AES algorithm takes less encryption time when compared with DES algorithm for the same set of patient data.

4.1 Time Complexity

The period of time needed to encrypt the data is referred to as the algorithm's time complexity or duration. Figure 4 presents the outcome with time complexity and charts comparing them. The time consumption is determined by the following equation:

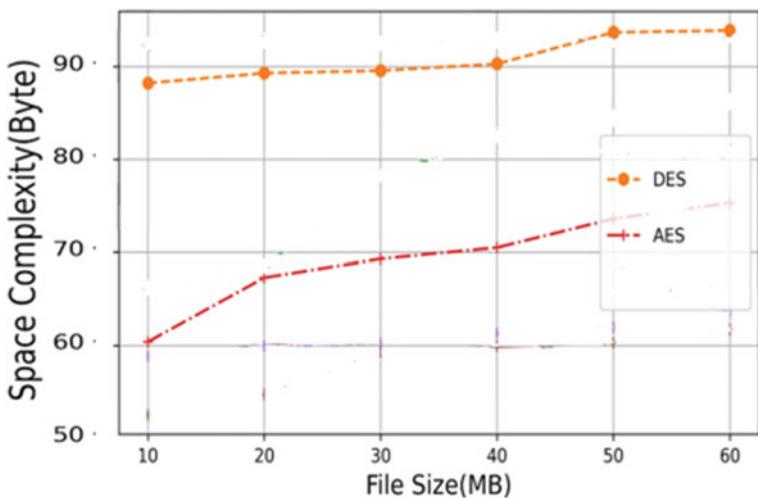


Fig. 5 Space complexity of AES versus DES

Time consumption = algorithm stop time – algorithm start time.

4.2 Space Complexity

The amount of data input and the amount of main memory needed to run the algorithm are considered as the memory consumption or space complexity, respectively. Figure 5 presents the outcome with space complexity and charts comparing them. AES took less memory space.

5 Conclusion

Medical information relates to the patient's vital privacy and health. It has higher requirements for usability, security, and integrity than other types of data. Due to certain limitations of centralized server, a blockchain technology which has a feature of decentralization and transparency can be used to provide security for patient record. In our proposed work, the patient data is encrypted by using AES and DES algorithm in order to compare the encryption time taken by both the algorithms and from the experimental results it is observed that AES algorithm takes less encryption time, i.e., faster in encrypting data thus avoids the unauthorized access to the patient record. In future work, to access the patient's record and to provide access permission to the user, smart contract will be created.

References

1. Pirbhulal S, Wu W, Li G, Sangaiah AK (2019) Medical information security for wearable body sensor networks in smart healthcare. *IEEE Consum Electron Mag* 8(5):37–41
2. Shree DN, Krishna DVL, Patan R (2022) Securing healthcare data using decentralized approach. In: 2022 international conference on electronics and renewable systems (ICEARS)
3. Joseph Gabriel S, Sengottuvelan P (2021) An enhanced blockchain technology with AES encryption security system for healthcare system. In: Proceedings of the second international conference on smart electronics and communication (ICOSEC)
4. Christo MS, Anigo Merjora A, Partha Sarathy G, Priyanka C, Raj Kumari M (2019) An efficient data security in medical report using block chain technology. In: International conference on communication and signal processing, 4–6 April 2019
5. Liang X, Zhao J, Shetty S, Liu J, Li D (2017) Integrating blockchain for data sharing and collaboration in mobile healthcare applications. In: 2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC). IEEE, pp 1–5
6. Zhang A, Lin X (2018) Towards secure and privacy-preserving data sharing in e-health systems via consortium blockchain. *J Med Syst* 42(8):1–18
7. Dwivedi AD, Srivastava G, Dhar S, Singh R (2019) A decentralized privacy-preserving healthcare blockchain for IoT. *Sensors* 19(2)
8. Kamboj P, Khare S, Pal S (2021) User authentication using Blockchain based smart contract in role-based access control. *Peer-to-Peer Netw Appl* 14(5):2961–2976
9. Abeywardena KY, Attanayaka B, Periyasamy K, Gunarathna S, Prabhathi U, Kudagoda S (2020) Blockchain based Patients' detail management System. In: 2020 2nd international conference on advancements in computing (ICAC), vol 1. IEEE, pp 458–463
10. Zhao K, Cui J, Xie Z (2017) Algebraic cryptanalysis scheme of AES-256 using Gröbner basis. *J Electr Comput Eng*
11. <https://www.hindawi.com/journals/jece/2017/9828967/>
12. <https://www.cybrary.it/blog/0p3n/des-data-encryption-standard>
13. Chen Y, Ding S, Xu Z, Zheng H, Yang S (2019) Blockchain-based medical records secure storage and medical service framework. *J Med Syst* 43(1):1–9
14. Nivethini P, Meena S, Krithikaa V, Prethija G (2019) Data security using blockchain technology. Special Issue Published in *Int J Adv Netw Appl (IJANA)*
15. Abdullah A (2017) Advanced encryption standard (AES) algorithm to encrypt and decrypt data

Strengthening Cybersecurity: A Comparative Study of KNN and Random Forest for Spam Detection



Sanya Joshi, Japanpreet, Lekha Rani, Pradeepa Kumar Sarangi,
and Ved Prakash Dubey

Abstract At present, email has become a necessary communication medium for exchanging messages and is considered an important part of business, commerce, government, education, entertainment, and other fields in various countries. As it is known that everything has its pros and cons, in spite of benefitting society everybody have to face its drawbacks, which include email spamming. Email spam or junk emails are unwanted emails which are annoying as well as dangerous, containing links trying to corrupt the computer system, stealing your bank details, and stealing your identity with the help of Botnet or real humans, fraudulent access to the information through data breaches. Spam filtering algorithms detect undesired, infected mail and block these messages from reaching to user's inboxes and keeping their email servers safe from getting overloaded. These are adaptable and provide sustainability to all the spam detected and provide security to the emails. It is important to make the network and system to be free from spammers, malicious links, and viruses. In this research, it has been demonstrated that the K-Nearest Neighbor algorithm by storing the training data and the class labels and is used for classification as well as regression and Random Forest containing more than one decision tree on different subsets improves the accuracy and achieve high accuracy rates, often above 95% by classifying email into spam and ham.

Keywords Spam detection · Research · KNN · Random Forest · Machine learning

S. Joshi · Japanpreet · L. Rani

Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura,
Punjab, India

e-mail: sanya1002.cse19@chitkara.edu.in

Japanpreet

e-mail: japanpreet1444.cse19@chitkara.edu.in

V. P. Dubey

Graphic Era Hill University, Dehradun, Uttrakhand, India

P. K. Sarangi (✉)

Chitkara University School of Engineering and Technology, Chitkara University, Himachal
Pradesh, India

e-mail: pradeepatasarangi@gmail.com

1 Introduction

In 1971, Ray Tomlinson invented electronic mail. Escorted by the growth in technology, the number of email users is also increasing. It is a medium for exchanging messages in the form of text, media, documents, and others. On the dark side, the fraudulent found their way of disruption by sending spam emails to genuine users. Email spam could be threatening as, at worst, it can be a part of a phishing scam, viruses, and spyware affecting gullible people. Phishing scams are mostly carried out in the name of legal businesses, banks, or other reputed organizations, trying to get your private information such as passwords, credit card information, and Aadhaar card number to debit money from our accounts. The most shocking fact about spam mail is that it will delete carbon dioxide as each mail contributes 0.3 g of carbon dioxide. Ham (not spam) is the mail wanted by the recipient. Spam mails are usually junk mails which are of no use to the recipient leading to slowing down the system, occupying unnecessary memory space, reducing internet transmission, etc., which indirectly affects the mental peace.

Networking of physical objects which consists of electronics in their architecture which is used to sense and communicate interaction among each other's or with respect to the external environment is known as the Internet of Things. Technology based on IoT will lead to advanced services and will change people practically. Social applications and platforms based on IoT have and are emerging. Because of IoT, there is an increase in spamming issues at larger rates.

It has been estimated by social networking experts that nearly 40% of social accounts are used in spamming. Spammers are found all over the world from different locations, those who work by hiding their real identity. Social networking tools are used to target people on specific parts like review page, fan page, public profiles, etc., to send links in the comment section that direct us to pornographic or other fraudsters.

Cybersecurity and email spam detection are interconnected to each other as spam mails contain malicious information or data that could become a security risk to a person or organization. Oftentimes these mails contain harmful links that can affect user's computer system. Spam detection of emails is a very crucial part of cybersecurity because it helps in preventing such attacks. It protects computer systems and most importantly the information from being stolen. Fraudulent make use of spam to attack the systems [1]. Intrusion detection system can be used to provide network monitoring in the identification of any attacks. Hence, spam detection techniques are very crucial for maintaining our system's security and protection. The risk of cyber-attacks can be decreased by filtering out the spam mails. Let's consider email spam as a thief trying to enter a house to steal. Similarly, spam mails are unwanted thieves which can cause harm to computer system. If a security system is installed in the house, then it can sense and detect these unwanted people and will help in protecting the house. In an exact way, spam filters can sense and filter out these unwanted mails, helping in protecting the computer systems [2]. Also with the increase in application

on the web like video-conferencing and others traffic is also expanding which results in connection loss, blockage, and other irregularities.

To solve these problems, two common approaches are knowledge engineering and machine learning. Machine learning is more organized than the knowledge engineering approach. In machine learning, instance-based or memory-based learning methods are used to distinguish spam mails from ham. There are various machine learning algorithms such as Naive Bayes, SVM, NN, and KNN which are effective in spam detection.

Support Vector Machine is used for both classification and regression problems. The primary goal is to create the best decision line (hyperplane) which separates n-dimensional space into classes and can be placed in the appropriate category. It is of two types—Linear (support vectors can be separated by lines) and Nonlinear (support vector cannot be separated by a line).

Decision Tree (tree-structured classifier) is a CART algorithm which means Classification and regression Tree algorithm. In this, the internal node indicates features of a dataset, branches indicate rules of decision, and lastly each leaf node indicates the output.

Naive Bayes (probabilistic classifier) is the simplest technique that assigns labels to instances of the problem which are represented as vectors, class labels are taken from a finite set. It is a supervised technique used for solving a classification problem. The main use of this is in classifying text including high-dimensional datasets. It gives output on the basis of the probability of the object.

In this research, the use of K-Nearest Neighbor has made which stores the training data and the class labels in such a way that the classifier memorizes that training data and in terms of accuracy. KNN algorithm is mainly used for classification and regression analysis. It is used in the fields like pattern recognition and analytical evaluation. There are no fixed ways to discover the most optimal value of K . Set a random value for K and initiate computing. If a small-scale K value has been chosen, it leads to unstable decision boundaries. The considerable K is superior for classification as it leads in balancing the decision threshold. Distance measures play a vital role in various machine learning algorithms. A distance function gives distance between the elements of a set. The distance can be measured by using Euclidean, Manhattan, etc. It is more suitable for small datasets and can handle values that are missing.

Random Forest is well-known and supervised learning machine algorithms. It contains more than one decision tree on different subsets and improves the accuracy of the given dataset. In training phase, decision trees are created and after being used for classification, the class having the highest number of votes is considered as the final output using an elected class of individual trees. It is used for classification and regression problems, takes less training time, has better F-score, reduces classification error, gives output with high accuracy, and maintains the accuracy even if the large part of data is missing. A standard machine learning technique is used in Random Forest, i.e., decision trees because of which it is called lazy learner. In this, input is given on the head and when it traverses down the tree, the data gets collected into small sets. Runtime of this algorithm is fast and can handle unbalanced and

missing data. The weakness of this algorithm is that it cannot predict beyond the range provided in training data and may over fit datasets when used for regression. If an algorithm works well for your dataset, then it is considered as the best test. RF surpasses the majority of the machine learning methods as it has the capability of processing around thousands of input variables more efficiently. Internally, during forest cultivation, it generates an unbiased forecast of collective error. It provides a way of compensating for mistakes in population classes using skewed datasets.

The objective of email spam detection is to improve data storage. There are two ways to do this in the proposed method. First is opinion rank which checks whether the mail id is trustworthy or not and further it uses two algorithms, high page rank, and inverse page rank. Most importantly, latent Dirichlet allocation is used to remove advertisement mail. This work experiments with the implementation of KNN and Random Forest models for email spam detection.

The novelty of this work lies in the implementation of machine learning algorithm for spam detection. This work implements KNN and Random Forest models and analyze their results.

2 Literature Review

In paper [3], they explained about various machine learning algorithms which are used to develop automated tools to get and utilize the data for training. They explained in detail about the working of Naive Bayes which uses probability (counting the frequency) and combines the values from given data.

In paper [4], they used Kaggle to train the algorithm dataset using pd.read_csv() to read spam mails. The output consists of various columns, from those which are not required are removed and the names of the columns are changed using Natural language toolkit (NLTK). They concluded that it is needed to observe the probability of words in ham and spam messages.

In another research [5], they have described a few filtering methods to understand it effectively. First and foremost is standard spam filtering method which works on predefined regulations and constitutes as a classifier with a number of arrangements. Another spam filtering method is client and enterprise which can send, receive, and secure mails by just one click via an ISP. At last, Case Base spam filtering system which uses vector expression and then machine learning technique is implemented on the sets to check whether mail is spam or ham.

In paper [6], they proposed about various techniques of spam filtration like Logistic Regression, Naive Bayes, Random Forest, LSTM, and Bi-LSTM achieving accuracies 95.8%, 96.3%, 96.8%, 98.6%, and 98.5%, respectively.

In [7], authors summarized that spam filters can be executed at all layers, at Mail Transfer Agent (MTA), severs gives an integrated anti-virus and anti-spam solutions for email protection, before the dangerous mail reaches to the network and at MDA, spam filters provide services to the customers.

In paper [8], they explained about phishing in which our confidential and sensitive information is being misused by creating a replica of a legitimate website. They also highlighted one of the efficient machine learning techniques—RF classifier. In this, all the mails are represented by a vector consisting of a value (continuous and binary).

In a research paper [9], they explained the two levels of spam filtering—(a) Individual user level in which a person can check spam mail with the help of a mail reader. (b) Enterprise level in which software is installed on the server which interacts with MTA. LAN can also be used to filter spam at individual level. If neighbors contain more spam, then it is considered as spam message.

In paper [10], the authors summarized about detecting spam via KNN which has high F-measure in comparison with Bayes and SVM. In contrast, KNN has lower accuracy than Bayes. To improve KN, Spearman's correlation coefficient is used to find distance which resulted in higher F-measure compared to traditional algorithms.

In paper [11], the authors consolidated image spam. Also, gave us the information about the Image Hunter Dataset. To train and set the image dataset, the proposed model (VGG-16) is used for extraction and selection. In this paper, authors have studied in detail about various types of techniques used in spam detection but the most efficient one was Random Forest considers all possible decisions and labels which results in high accuracy by building the best tree. The accuracy achieved was 98.7% and time taken was 02:44.64 s.

In paper [12], they have described in detail about AI which aims at machine assisting and gave us information about various kinds of calculations involved in AI. They used KNN for email spam classification which involves two stages, i.e., filtering and training. The result was calculated achieving an accuracy of 80%.

In paper [13], the authors have found the result using Random Forest classification technique. They used a spam .csv file comprising 5572 mails (747 spam and 4825 ham) and attained an accuracy of 94.87%. RF requires fewer parameters and has lower classification errors and higher F-score.

In paper [14], the authors presented the various frequently used spam filter techniques and found the drawback based on the data. They also recognized the wrong words in the data of Bayesian filtering techniques and focused on the existing work and concluded work results in more precision than normal spam filters.

In a research paper [15], they codified a prototype for spam images classification. It extracts an image detection classifier (image-based). In this paper, they achieved correct spam accuracy.

In paper [16], authors described pre-existing spam filtering techniques. They used a learning-based methods for evaluation. In this, there was a comparison between anti-spam data and look into new filtering techniques for spam.

In paper [17], they have presented various ways for finding Social Network spammers. The main focus was toward Twitter spam detection, in which spam is recognized based on false content, spam URLs, trendy topic set (Table 1).

Table 1 Findings of recent publications

References	Technique	Findings/conclusion
[3]	Naïve Bayes	Used probability and combined the data via automated tools. ELM performs better than SVM
[4]	Naïve Bayes	NLTK was used to change the name of columns along with a dataset from Kaggle
[5]	Client and enterprise	ISP was used to check whether mails were ham or spam
[6]	Logistic regression and LSTM	Achieved the accuracies of 98.5 and 98.6% respectively
[7]	Multilayer perceptron	Execution at layers. Can find value with desired accuracy. Multilayer Perceptron classifier out performs other classifiers
[8]	Random Forest	Mails were represented via vector having continuous or binary value. Achieved the high accuracy rate of 99.7%
[9]	Enterprise level	MTA interaction via software installation
[10]	KNN	Spearman's correlation coefficient was used to measure distance for higher F-measure and achieved higher accuracy and F-measure compare to other specified techniques like SVM and Naïve Bayes
[11]	CNN, SVM and Naïve Bayes	CNN performed better than other methods
[12]	KNN	Accuracy = 80% and Precision = 88%
[13]	Random Forest	With the help of .csv files 94.8% of accuracy got achieved
[14]	Naïve Bayes	Focused on finding results with more precision
[15]	Prototype	Extracts image-based detection
[16]	Logistic regression	Comparison between anti-spam data and new techniques
[17]	Topic guessing	Spam was recognized based on false content, spam links or topics

3 Methodology

It involves various steps such as collecting data, data pre-processing which is further divided into various aspects: data cleaning, attribute selection, transformation, and data integration. Lastly, feature selection takes place through which it is expected to get results with best-achieved accuracy (Fig. 1).

4 Data Collection

In this model, Spam Email dataset have been used from Kaggle (<https://www.kaggle.com/datasets/shantanudhakadd/email-spam-detection-dataset-classification>). This research has focused on two types of messages—ham and spam. The dataset contains

Fig. 1 Workflow of spam detection using techniques: Random Forest and KNN

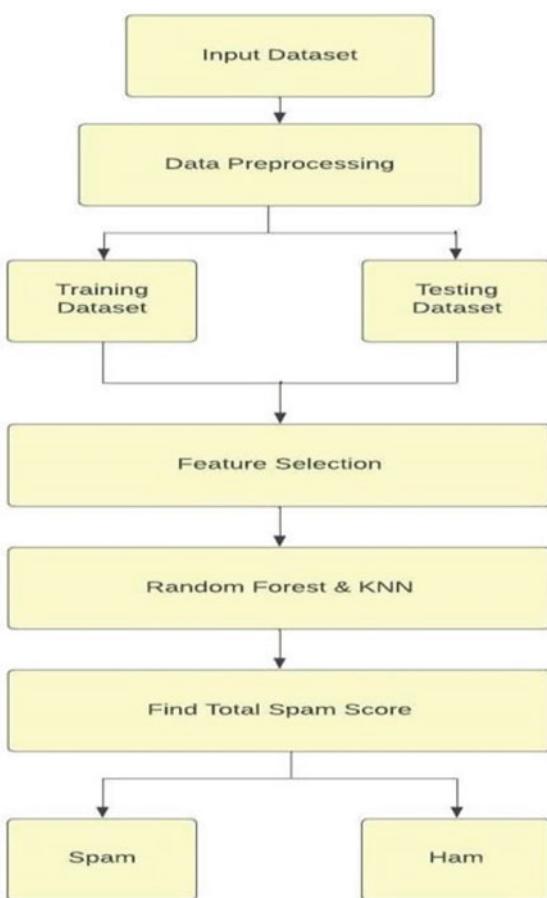


Table 2 Sample dataset used in classifying the spam and ham messages

V1	V2
Ham	Go until jurong point, crazy..Available only...
Ham	Ok lar... Joking wif u oni...
Spam	Free entry in 2 a wkly comp to win FA Cup fina...
Ham	U dun say so early hor... U c already then say...
Ham	Nah Idon't think he goes to usf, he lives aro...

two columns in which the first column contains spam/ham classification and second column contains 3570 randomly picked mails. A sample data set is given in Table 2.

5 Data Processing

Data pre-processing is the foremost and crucial step in preparing the raw data and converting it into formatted form, by replacing the labels to column vectors with binary values. A sample is given in Table 3.

The steps involved in data pre-processing are as follows

1. Data Cleaning—It is a process in which raw data is prepared for Natural Language Processing (NLP). It involves:

Normalizing Text, removing Unicode Characters, Tokenization (discrete elements), removing Stop words (commonly used words), performing Stemming and Lemmatization, i.e., stemming is somehow similar to lemmatizing the word but in lemmatization, word is converted into meaningful form, for example: in stemming ‘Caring’ returns ‘Car’ and in lemmatization ‘Caring’ returns ‘Care’ (Table 4).

2. Data Integration—It is a process which combines data from different source systems to get unified sets of information. It is used when the data is segregated into various sources. For data integration, we merge multiple pandas’ data frames with the help of the merge function.
3. Data Transformation—It is a technique used in converting raw data into a suitable format that eases down data mining and recovers strategic information. It includes data cleaning and data reduction techniques that convert data into appropriate form. It supports common data formats like read from SQL database, CSV file, etc. In python, Pandas add up the concept of Data Frame used for cleaning and analyzing datasets (Fig. 2).

Table 3 Mapping of data, spam (1) and ham (0)

Target	Email
0	Go until jurong point, crazy..Available only...
0	Ok lar... Joking wif u oni...
1	Free entry in 2 a wkly comp to win the FA Cup fina...
0	U dun say so early hor... U c already then say...
0	Nah Idon't think he goes to usf, he lives aro...

Table 4 Data splitting

No	Email
1114	No no:)this is kallis home ground. Amla home to...
3589	I am in an escape theatre now.. Going to watch K...
3095	We walked from my moms
1012	I dunno they close oredi not... If v ma fan...
3320	Yoim right by yo work

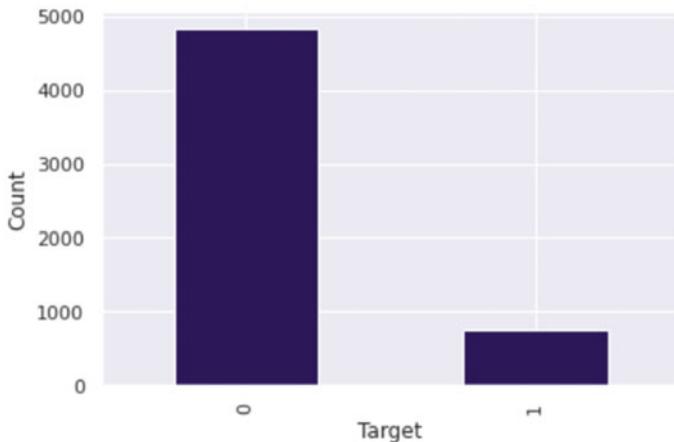


Fig. 2 Target contains ham (0) and spam (1) and y-axis has count of ham and spam

6 Theoretical Framework

It is an algorithm which assigns data points to various categories and classes. It is further divided into two main models: supervised and unsupervised model. In this research paper, a supervised model is used. In this, classifiers train to differentiate between labeled and unlabeled data. It involves two main steps, i.e., classification and regression which are differentiated by the type of target variable. In classification, the target variable is of categorical type while in regression it is in numeric form, for example—KNN, Random Forest, etc.

(i) K-Nearest Neighbor (KNN)

Steps involved in KNN algorithm are described in brief as follows

Step 1. Training—Storing the train messages after the training process. After data splitting, around 0.83% data is taken for training and rest for testing causes.

Step 2. Filtering—In this stage, determine k nearest neighbors among the messages in the training set for a given message. If there are more spam messages among the neighbors that will be classified as spam else classified as ham.

Steps:

1. Consider variable $D = \text{number of closest neighbors.}$
2. Find out length between the query-instance and training samples.
3. Arrange the length in order and find the nearest neighbors based on the Nth minimum distance.
4. Collect the X group of the nearest neighbors.

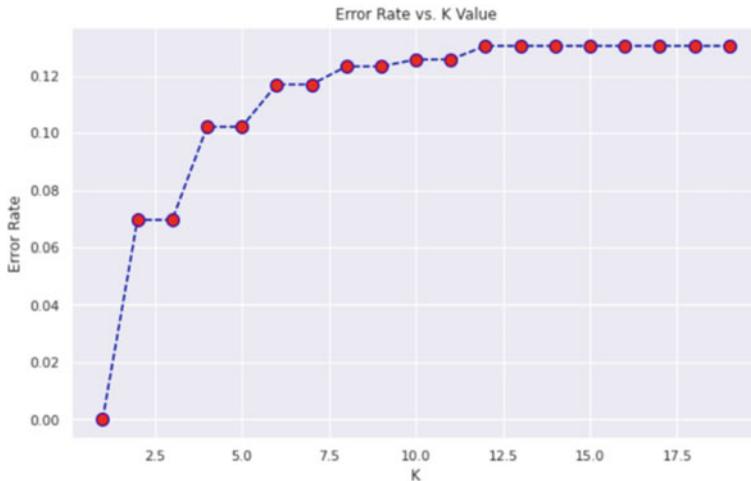


Fig. 3 Graph of error rate versus K signifies that a small value of K could lead to overfitting and big value of K lead to under fitting

5. Utilize a straightforward larger part of the closest ones as the neighbors as the prediction value of the query-instance, i.e., predicted values, actual values along with total number of predictions. We can calculate accuracy, precision, etc., with the help of a confusion matrix (Fig. 3).

In K-Nearest Neighbor, best value of K has been determined by creating an Elbow Plot. It is a visual representation of an exchange between underfitting and overfitting, plotting the error rate of the model as a function of the value of K . The value of K is inversely proportional to the error rate making the model more complex, i.e., if K increases, the error rate decreases, and vice-versa. After some time, an increase in value of K doesn't result in a decreasing error rate and will rather lead to an increasing error rate because of overfitting. Elbow Plot is used to find the optimal value of K when the error rate starts falling and this value of K strikes a balance between underfitting and overfitting. Different values of K are recorded with the help of a validation set.

(ii) Random Forest

Many classification trees are grown by random forests. It is grown as—The steps involved in cultivating trees are as follows:

1. Consider the number of training instances as N . Select it randomly from the original data but with substitution.
2. Assume M input variables, a number $m \ll M$ are defined as for each node, m variables are arbitrarily selected from M and the delicate part of m is used to split the node so that m can have constant rate via the whole period of expanding the forest.

3. Pruning is forbidden as each tree is cultivated to the greatest possible extent possible.

A tree which has a small error rate is considered as a strong classifier. Additionally, with an increase in the concentration of each tree in the forest, the error rate of the forest reduces. If the value of m has been reduced, it leads to a decrease in relationship and the power of the forest. The value of p can be calculated using out-of-bag estimate, and the value of m can be allocated under the limitation promptly.

7 Results and Discussion

The confusion matrix is a matrix that is used to evaluate the performance of the classification model for the test datasets and can be evaluated if the true value of test data is known. The division of matrices is done in two dimensions, i.e., predicted values, actual values along with total number of predictions. With the help of a confusion matrix, accuracy, precision, etc., can be calculated (Figs. 4 and 5).

Calculated the result with the help of

1. Accuracy: It interprets how the representation forecast the accurate output.
2. Precision: It is defined as the number of accurate outputs provided via model.

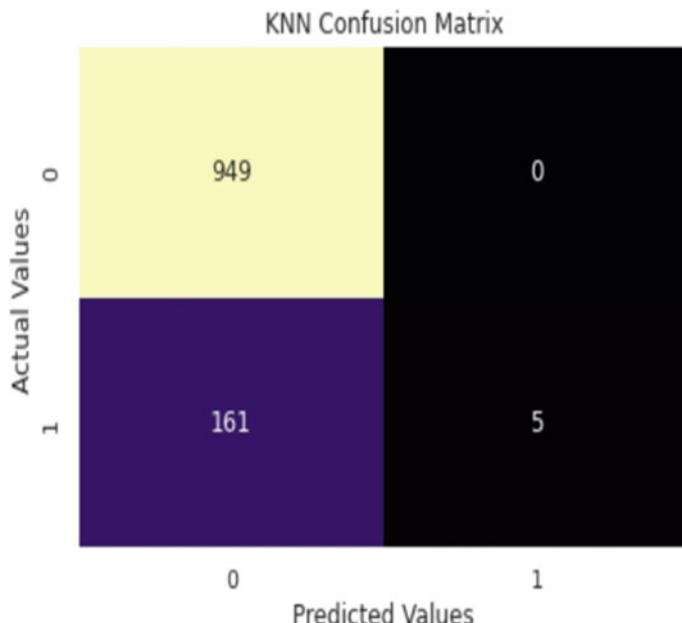


Fig. 4 The performance of KNN algorithm is defined by the KNN confusion matrix which plots actual and predicted values of KNN classifier

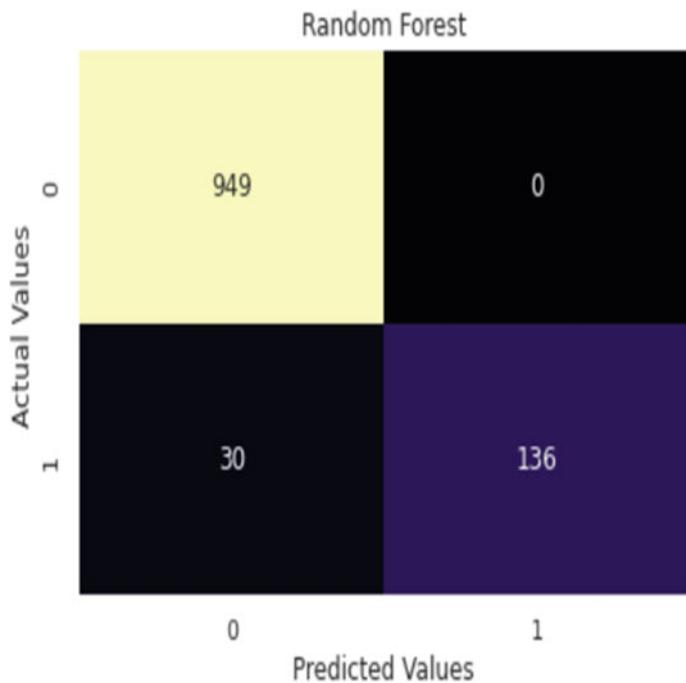


Fig. 5 The performance of Random Forest algorithm is defined by the Random Forest confusion matrix which plots actual and predicted values of RF classifier

3. Recall: It is defined as how our model predicted correctly out of total positive classes.
4. F-Measure: This score helps us to evaluate the precision and recall at the same time.
5. Support: It is the number of actual occurrences of the class in a certain dataset and can be calculated by adding the rows of the confusion matrix.

Table 5 represents the result table of KNN containing parameters that are used for comparison with other algorithms. It has been calculated using Table 2 dataset.

Table 5 Result table of KNN

Spam/ham	Precision	Recall	F-score	Support
0	0.85	1.00	0.92	949
1	1.00	0.03	0.06	166
Accuracy			0.86	1115
Macro average	0.93	0.52	0.49	1115
Weighted average	0.88	0.86	0.79	1115

Table 6 Result of Random Forest

Spam/ham	Precision	Recall	F-score	Support
0	0.97	1.00	0.98	949
1	1.00	0.82	0.90	166
Accuracy			0.97	1115
Macro average	0.98	0.91	0.94	1115
Weighted average	0.97	0.97	0.97	1115

Table 6 represents the result table of Random Forest containing parameters that are used for comparison with other algorithms. It has been calculated using Table 2 dataset.

8 Conclusion

Detection of spam is essential to secure message and email communication. Detection of spam accurately is big issue. It is a serious issue that is not just annoying to the end-users but also it can cause financial loss and risk of security. The spam filtering effectiveness can be determined with the help of classification methods used. However, the methods lack incapability to detect spam accurately and systematically. The efforts and research carried out by different researchers to solve the problem of spam through the use of machine learning algorithms was discussed. In this paper, the result has been summarized for KNN and Random Forest for spam base dataset. In KNN, it is concluded that it has easy implementation and works very well with small datasets. On the other hand, Random Forest is a great algorithm for classification and regression. Also, when Random Forest is used to find missing data it performs very well in big datasets and mostly predicts accurately. The main objective was to optimize the data storage and to detect the spam mails which were achieved efficiently with an accuracy of 85.56% in KNN and 97.31% in Random Forest.

9 Future Scope

The majority of people who are unaware of spam messages pay close attention to them as it assists in the identification of unappealing data and risk. Therefore, people concentrate on developing the classifier which is best for identifying the spam messages and mails. In this research, KNN and Random Forest have been used. Random Forest being the best classifier is the best choice for text classification. While KNN a non-parametric classifier uses proximity to predict about the group of an individual data type. When the system was tested, the use of different performance metrics like precision, accuracy, etc., have been made. In future, it can be discovered

and modified by KNN as it is unable to cope up with the unbalanced classification. On the other hand, we will try to increase the effectiveness for real time predictions in Random Forest with increase in number of trees, the algorithm becomes slow.

References

1. Snehi J, Bhandari A, Baggan V, Snehi M (2020) Diverse methods for signature based intrusion detection schemes adopted. *Int J Recent Technol Eng* 9(2):44–49
2. Baggan V, Panda SN Enhancing border gateway routing protocol with software defined networking. *International Journal of Innovative Technology and Exploring Engineering*, Volume-8 Issue-8 June, 2019, pp: 976-984
3. Ahmed N, Amin R, Aldabbas H, Koundal D, Alouffi B, Shah T (2022) Machine learning techniques for spam detection in email and IoT platforms: analysis and research challenges. *Sec Commun Netw* 2022:19. Article ID 1862888
4. Sultana T, Sapnaz KA, Sana F, Najath J (2020) Email based spam detection. *Int J Eng Res Technol (IJERT)* 9. ISSN 2278-0181
5. Bhuiyan H, Ashiquzzaman A, Juthi T, Biswas S, Ara J (2018) A survey of existing e-mail spam filtering methods considering machine learning techniques. *Glob J* 18. ISSN 0975-4172
6. Malhotra P, Malik S (2022) Spam email detection using machine learning and deep learning techniques. *Soc Sci Res Netw* 10
7. Christina V, Karpagavalli S, Suganya G (2022) Email spam filtering using supervised machine learning techniques. *Int J Eng Res Technol (IJERT)* 02:3126–3129
8. Akinyelu AA, Adewumi AO (2014) Classification of phishing email using random forest machine learning technique. *J Appl Math* 2014:6. Article ID 425731
9. Saad OM, Darwish A, Faraj R (2012) A survey of machine learning techniques for Spam filtering. *Int J Comput Sci Netw Sec (IJCSNS)* 12
10. Sharma A, Suryawanshi A (2016) A novel method for detecting spam email using KNN classification with spearman correlation as distance measure. *Int J Comput Appl* 136(6)
11. Abuzaid NN, Abuhammad HZ (2022) Image SPAM detection using ML and DL techniques. *Int J Adv Soft Comput Appl* 14(1)
12. Deshmukh N, Dhumal V, Gavasane N, Jadhav SV (2021) Spam detection by using KNN algorithm techniques 6. ISSN (Online) 2456-0774
13. Reddy KN, Kakulapati V (2021) Classification of spam messages using random forest algorithm 15(8). ISSN 1001-2400
14. Gaurav D, Tiwari SM, Goyal A, Gandhi N, Abraham A (2020) Machine intelligence-based algorithms for spam filtering on document labeling. *Soft Comput* 24(13):9625–9638
15. Mokri MAES, Hamou RM, Amine A (2019) A new bio inspired technique based on octopods for spam filtering. *Appl Intel* 49(9):3425–3435
16. Dedeturk BK, Akay B (2020) Spam filtering using a logistic regression model trained by an artificial Bee Colony algorithm. *Appl Soft Comput* 91:1–17
17. Mendez JR, Yanez TRC, Ordas DR (2019) A new semantic-based feature selection method for spam filtering. *Appl Soft Comput* 76:89–104

Causes of Cyber Fraud in Commercial Banks in Nigeria: A Case Study of Zenith Banks in Abuja



Ekong Eyo Unwana and Rajesh Prasad

Abstract Cyber fraud has arisen as a critical danger to the financial business, and monetary establishments like Zenith Bank are not insusceptible. This theoretical gives an outline of the causes and kinds of digital misrepresentation explicitly applicable to Zenith Bank. Understanding these variables is urgent for the bank to carry out powerful safety efforts and safeguard its clients' monetary resources. The reasons for digital extortion in Zenith Bank originate from different sources. Mechanical progressions have presented weaknesses, making obsolete frameworks and unpatched programming practical objectives for cybercriminals. Human weaknesses, including guilelessness and powerlessness to social designing strategies, likewise add to the outcome of false exercises. Furthermore, insider dangers represent a critical gamble, as believed workers might abuse their entrance honors or compromise delicate data. The globalization and interconnectivity of the financial business further worsen the difficulties in fighting digital misrepresentation, as cross-line exchanges and jurisdictional intricacies make requirement troublesome. With respect to sorts of digital misrepresentation saw in Zenith Bank, a few normal examples arise. Phishing and ridiculing procedures exploit clients' trust by fooling them into revealing individual data or login qualifications through false messages or sites. Fraud, a predominant cybercrime, includes the obtaining and abuse of people's very own data for fake purposes. Malware and ransomware assaults present critical dangers, with vindictive programming compromising PC frameworks and coercing casualties for monetary benefit. Business Email Split the difference (BEC) tricks exploit the power of high-positioning chiefs or confided in accomplices to delude representatives into starting unapproved exchanges. Card misrepresentation, venture tricks, and monetary plans focusing on clients' monetary resources are additionally predominant. To actually battle digital extortion, Zenith Bank ought to zero in on tending to the causes and executing proactive safety efforts. This remembers effective money management for vigorous mechanical framework, preparing workers to perceive and answer

E. E. Unwana
African University of Science and Technology, Abuja, Nigeria

R. Prasad (✉)
Department of Computer Science and Engineering, Ajay Kumar Garg Engineering College,
Ghaziabad, India
e-mail: prasadrajesh@akgec.ac.in

digital dangers, carrying out multifaceted verification, and ceaselessly checking and refreshing security conventions. By remaining cautious and proactive, Zenith Bank can safeguard its clients from the negative effects of digital extortion and keep a protected financial climate.

Keywords Cybersecurity · Fraud · Bank · Cybercrime · Zenith Banks

1 Introduction

In current conditions of banks are past guaranteed attack as money isn't simply kept in bank vaults. Banks need to conform to current examples of happening with work electronically and all the while protect themselves against mechanized infringement as a result of the rising number of digital extortion in bank [10]. Specialists should make a short move to keep such episodes away from occurring. In the present interconnected computerized world, the danger of digital misrepresentation poses a potential threat, presenting huge dangers to people and organizations the same [5]. Monetary establishments, for example, Zenith Bank, are not invulnerable to this threat. Digital extortion alludes to crimes directed through electronic channels, taking advantage of weaknesses in the financial framework to acquire unapproved access or control monetary exchanges for unlawful purposes [21]. Understanding the causes and sorts of digital misrepresentation is critical for shielding the trustworthiness of monetary establishments like Zenith Bank and safeguarding their clients.

Cybersecurity involves protecting digital information and the infrastructure it resides on. Cyber fraud in Zenith Bank can be credited to a blend of elements, including mechanical progressions, developing criminal strategies, and human weaknesses. The rising dependence on advanced stages for monetary exchanges has made new roads for fraudsters to take advantage of. This has required the arrangement of vigorous safety efforts to remain in front of cybercriminals.

In this conversation, we will investigate the causes and different sorts of digital misrepresentation that represent a danger to Zenith Bank. By revealing insight into these issues, we mean to bring issues to light about the difficulties looked by monetary foundations and people in combatting digital misrepresentation. Besides, we will feature the significance of proactive measures and progressing carefulness in defending touchy monetary data and guaranteeing a safe financial climate for all clients of Zenith Bank.

2 Literature Review

Digital extortion has turned into an unavoidable and steadily developing danger in the present computerized scene. Monetary foundations, organizations, and people are progressively powerless against the malignant exercises of cybercriminals [11,

[13]. This exhaustive survey means to dig into the causes and sorts of digital misrepresentation, giving a more profound comprehension of this major problem.

I. Causes of Cyber Fraud:

Ebiasuode et al. [16] and [6] distinguished the accompanying reasons for digital extortion in business establishments:

Mechanical Progressions:

Quick progressions in innovation have made a complex computerized biological system that is both a shelter and a curse. While innovation upgrades effectiveness and comfort, it likewise presents new weaknesses [9, 19]. Cybercriminals exploit shortcomings in programming, organizations, and gadgets, exploiting obsolete frameworks and unpatched weaknesses.

Human Weaknesses:

Notwithstanding the refinement of safety frameworks, people stay the most vulnerable connection in the battle against digital misrepresentation. Social designing procedures, for example, phishing messages, trick calls, and misleading sites, go after human feelings and artlessness. Cybercriminals maneuver people toward uncovering touchy data or performing unapproved activities.

Insider Dangers:

Believed representatives or insiders inside an association can represent a critical gamble. These people might manhandle their entrance honors, release delicate data, or take part in false exercises for individual addition. Insider dangers feature the significance of powerful security conventions, worker checking, and severe access controls.

Globalization and Interconnectivity:

The interconnected idea of the present worldwide economy has prompted expanded cybercrime valuable open doors. Cross-line exchanges, information sharing, and remote access present difficulties regarding ward and policing. Cybercriminals exploit these intricacies, making it hard to really follow and arraign them.

II. Types of Cyber Fraud:

Wapmuk [25] examined the types of cyber fraud in banks:

Phishing and Caricaturing:

Phishing includes beguiling people into uncovering delicate data, for example, usernames, passwords, or monetary subtleties, through false messages or sites [24]. Mocking, then again, includes mimicking a genuine element to acquire trust and concentrate classified data.

Wholesale Fraud:

Wholesale fraud happens when cybercriminals secure and abuse a singular's very own data to accept their character. This can prompt unapproved monetary exchanges, the kickoff of fake records, or the abuse of individual information for criminal operations [4, 14].

Malware and Ransomware Assaults:

Vindictive programming (malware) is intended to disturb or acquire unapproved admittance to PC frameworks. Ransomware, a kind of malware, scrambles documents or blocks admittance to a framework until a payment is paid. These assaults can cause huge monetary misfortunes and functional disturbances [18].

Business Email Split the Difference (BEC):

BEC includes cybercriminals imitating high-positioning leaders or confided in accomplices to trick representatives into starting deceitful wire moves or unveiling delicate data. These refined assaults exploit trust and authority inside associations.

Card Misrepresentation:

Cybercriminals take card data through different means, for example, hacking data sets, skimming gadgets, or catching internet-based exchanges [20]. This taken information is then utilized for unapproved buys or sold on the dull web.

Speculation and Monetary Tricks:

Cyber fraud reaches out past customary financial practices. Speculation tricks, Ponzi plans, and fake digital currency plans mislead people into leaving behind their cash with expectations of ridiculously significant yields.

Degenerate Program or Programming

Organizations given by Untouchables That Aren't Secure

To even more expeditiously serve their clients, many banks and monetary affiliations utilize untouchable associations from different suppliers [23]. Anyway, in the event that those distant suppliers don't have sufficient automated affirmation set up, your bank may be the one to overcome the most clearly dreadful piece of the harm. It is key to consider how you will screen yourself against distant security takes a risk going before executing their thoughts [17].

Information that has been changed

Designers don't normally go in to take information; they essentially need to change it. Unfortunately, this kind of attack is attempting to perceive rapidly away and can cost monetary affiliations millions, on the off chance that not billions, of naira in misfortunes [8]. Since changed information doesn't be ensured to show up, clearly, to be not precisely comparable to unmodified information on a shallow level, it very well may be hard to sort out what has and hasn't been changed in the event that your bank has been hacked.

3 Methodology

Survey research design was adopted. Overview research configuration was taken on. In understanding, Adegoke [26] attested that review research configuration includes the perception and depiction of the overall way of behaving of the review populace. In other word, this sort of examination is helped out through an overview, as it includes huge gathering. The legitimacy of the exploration instruments was assessed by a specialist from the college's Branch of Science Training in Niger State. A comparable master from the college's network safety division approved the instruments.

The populace comprised of 49 parts of Zenith Banks in Abuja, Civil chamber FCT Abuja Nigeria. The review took on multi-stage examining method in which 18 branches were chosen and an example of 557 staff was chosen. Organized poll was utilized to gather the reaction from the example. Information gathered from the managed surveys were dissected utilizing unmistakable measurement. The outcomes were introduced utilizing table. The review's respondents were then furnished with an educated agree structure to guarantee their secrecy and privacy when it came to revealing the outcomes.

4 Result

Figure 1 showed that a 557 duplicates of survey were disseminated, while an amount of 497 polls were recuperated and seen as usable.

Figure 2 showed that difficult business scenario has the highest mean score (3.04) followed by current business pressure and collusion between employee with high mean scores of (2.93) and (2.96), respectively. Moreover, the research findings indicate that the absence of oversight by line managers, insufficient data encryption, and reliance on services provided by third parties are significant causes of cyber fraud, with mean scores of 2.81, 2.68, and 2.59, respectively. On the other hand, the use of brute force attacks and spoofing techniques were found to have the lowest mean scores of 2.09 and 2.47, respectively. Thusly, Button and Cross [14] expressed that ridiculing, usage of savage power affix were purposes behind advanced attack on banks.

Figure 3 showed that phishing has the highest mean score (3.06), followed by money laundering, hacking/cracking, and accounting fraud by bank staff with high mean scores (2.97), (2.89), and (2.73), respectively. Also, money transfer technique, identity theft, pharming, scams, and computer virus are types of cyber fraud with the following mean scores (2.58), (2.64), (2.66), (2.63), and (2.49), respectively.

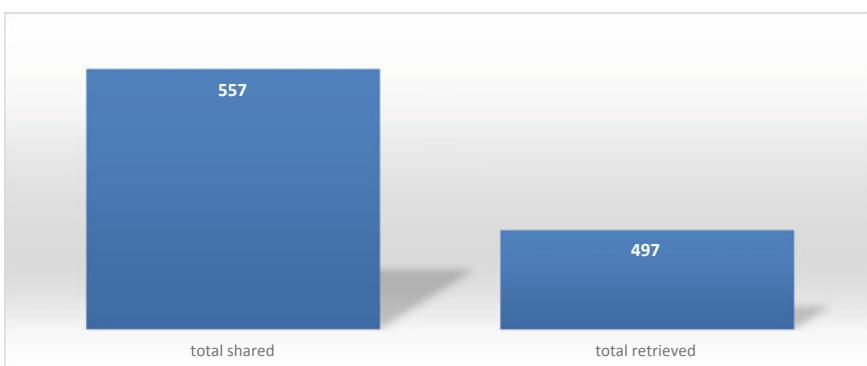


Fig. 1 Response rate. *Source* Field work (2023)

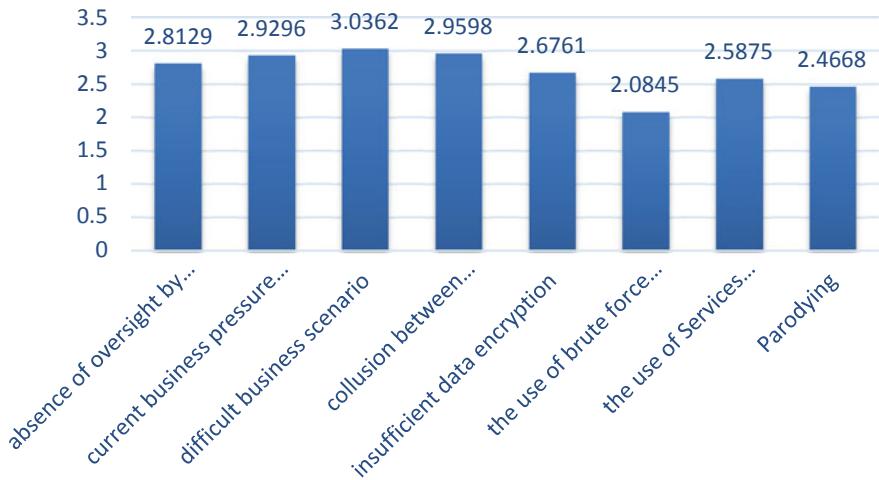


Fig. 2 The causes of cyber fraud in banks

However, the figure showed that wiretapping and spy software has the lowest mean scores (2.10) and (2.38), respectively. The outcome was upheld by Button and Cross [14]. The creators attested that banks were been cheated utilizing fraud, conspiracy between bank staff and illegal tax avoidance.

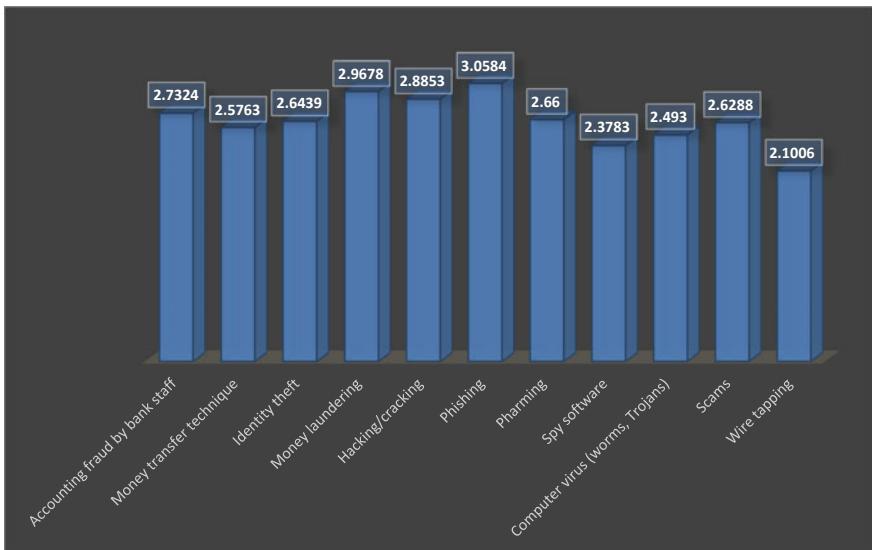


Fig. 3 Types of cyber fraud in Zenith Banks

Discussion

The findings of the research revealed that bank clients have experienced various forms of electronic fraud, including internal fraud by bank staff, discount fraud, tax evasion, hacking, phishing, pharming, and computer viruses. These findings are supported by McGuire and Dowling's study [29], which highlighted hacking, spamming, and the use of malicious codes and malware as common tactics employed by cybercriminals to deceive bank clients. The ultimate goal of these fraudulent activities is unauthorized access to a client's account in order to steal or transfer funds to another financial account [3]. However, in some rare cases in Nigeria, the objective of cybercriminals is to damage the bank's reputation by denying service to clients [27] and disrupting computer networks within organizations.

The research also identified significant factors contributing to digital fraud in banks. These causes include a lack of oversight by line managers or senior executives regarding deviations from existing electronic processes/controls, pressure to meet business targets, collusion between employees and external parties, insufficient data encryption, reliance on services provided by third parties, and underestimating the importance of these issues. These findings are supported by Hassan [28], who stated that unemployment is a major driving force behind cybercrime in Nigeria. It is a fact that more than 20 million graduates in the country are without gainful employment, which has led to an increase in their involvement in criminal activities for survival. Additionally, Sethi [22] emphasized that inadequate electronic processes/controls, pressure to meet business targets, collusion between employees and external parties, insufficient data encryption, reliance on services provided by third parties, and underestimating the risks involved are significant factors contributing to digital fraud in banks.

5 Conclusion

A few key elements add to the pervasiveness of digital extortion in the Nigerian financial area [2, 15]. Insufficient online protection, first and foremost, measures and framework inside banks make them defenseless against complex hacking procedures utilized by cybercriminals. Deficient interest in vigorous security frameworks, obsolete programming, and an absence of gifted network protection experts establish a climate helpful for cyber fraud.

Besides, the quick reception of innovation and advanced financial administrations in Nigeria has outperformed the execution of sufficient security conventions. This has brought about a hole between the progressions in web-based banking and the capacity to really shield client information. Deficient mindfulness among clients about the dangers of digital misrepresentation and the essential precautionary measures further fuels the issue.

Furthermore, the expansion of social designing strategies, for example, phishing tricks, malware assaults, and data fraud, represents a huge danger [7]. Cybercriminals

exploit human weaknesses to acquire unapproved admittance to banking situation and maneuver clueless people toward uncovering delicate data [12]. Absence of mindfulness and schooling among bank representatives and clients with respect to these strategies make them powerless to such deceitful exercises.

Additionally, frail administrative structures and implementation systems add to the determination of digital misrepresentation in Nigerian business banks. The shortfall of severe regulations and punishments explicitly tending to cybercrime, joined with restricted assets apportioned to exploring and indicting cybercriminals, establishes a climate of exemption. This supports the expansion of digital misrepresentation and subverts endeavors to really battle it.

To address the reasons for cyber fraud in banks in Nigeria, a multi-pronged methodology is important. Right off the bat, there is a requirement for expanded interest in online protection framework, including powerful firewalls, encryption strategies, and ongoing danger discovery frameworks. Banks ought to focus on the enrollment and preparing of gifted network protection experts to foster proactive guard instruments.

Moreover, improving client training and mindfulness programs is essential. Banks ought to teach clients about normal digital misrepresentation strategies, advance dependable internet-based conduct, and stress the significance of consistently refreshing passwords and safeguarding individual data. Furthermore, bringing issues to light among bank representatives through customary instructional meetings can assist with alleviating the gamble of social designing assaults.

In equal, the Nigerian government ought to order and uphold extensive regulation explicitly focusing on cybercrime [1]. Stricter punishments and lawful structures ought to be laid out to dissuade potential cybercriminals and guarantee their arraignment. Joint effort between administrative bodies, policing, and monetary organizations is fundamental to foster a planned reaction to digital extortion occurrences.

At last, tending to the reasons for cyber fraud banks in Nigeria requires an aggregate exertion. Banks, clients, controllers, and policing should cooperate to improve network protection measures, bring issues to light, and authorize severe lawful systems. By making proactive strides and embracing a far reaching approach, Nigeria can relieve the dangers of digital extortion and encourage a safe financial climate for its residents.

Recommendations

That's what the review suggested:

1. Peak bank ought to utilize severe measure to screen staff exercises particularly in the secrecy of client data.
2. Network safety review ought to be finished by Apex consistently to be capable alleviate the different methods involved by fraudsters in carrying out cybercrime. Important provisos recognized during security review ought to be fixed right away in order to forestall any endeavor of misrepresentation in the bank.

References

1. Aaron MM (2012) A case study on e-banking security—when security becomes too sophisticated for the user to access their information. *J Internet Bank Comm* 17(2)
2. Adeniyi A (2016) Analysis of fraud in banks: evidence from Nigeria. *J Interpersonal Violence* 35(15–16):2800–2824
3. Akinbowale OE, Klingelhöfer HE, Zerihun MF (2020) Analysis of cyber-crime effects on the banking sector using the balanced score card: a survey of literature. *J Financ Crime* 27(3):945–958
4. Al-Alawi AI, Al-Bassam MSA (2020) The significance of cybersecurity system in helping managing risk in banking and financial sector. *J Xidian Univ* 14(7):1523–1536
5. Alao AA (2016) Forensic auditing and financial fraud in Nigerian deposit money banks (DMBs). *Eur J Account Audit Financ Res* 4(8):1–19
6. Alghazo JM, Kazmi Z, Latif G (2017) Cyber security analysis of internet banking in emerging countries: user and bank perspectives. In: 2017 4th IEEE international conference on engineering technologies and applied sciences (ICETAS). IEEE, pp 1–6
7. Aloraini B, Nagappan M, German DM, Hayashi S, Higo Y (2019) An empirical study of security warnings from static application security testing tools. *J Syst Softw* 158:110427
8. Amadi EC, Eze U, Ikerionwu C (2017) Game theory basics and its application in cyber security. *Adv Wirel Commun Netw* 3(4):45–49
9. Anah BB, Funmi DD, Julius M (2012) Cybercrime in Nigeria: causes, effects and the way out. *ARPN J Sci Technol* 2(7)
10. Aneke SO, Nweke EO, Udanor CN, Ogbodo IA, Ezugwu AO, Uguwishiwi CH, Ezema ME (2020) Towards determining cybercrime technology evolution in Nigeria. *Int J Lates Technol Eng Manage Appl Sci* 9:37–43
11. Badejo BA, Okuneye BA, Taiwo MR (2019) Fraud detection in the banking system in Nigeria: Challenges and prospects. *Shirkah J Econ Bus* 2(3)
12. Basil U (2015) Dealing with the challenge of cybercrime in Nigeria under the new Cybercrime Act. The Lagos Chamber of Commerce & Industry 2015 Seminar of the Financial Services Group September 3, 2015
13. Buchanan B (2016) The cybersecurity dilemma: hacking, trust, and fear between nations Oxford University Press, UK
14. Button M, Cross C (2017) Cyber frauds, scams and their victims. Taylor & Francis
15. Chika OV, Promise E, Werikum EV (2022) Influence of liquidity and profitability on profits growth of Nigerian pharmaceutical firms. *Goodwood Akuntansi dan Auditing Reviu* 1(1):1–13
16. Ebiasuode A, Onuoha BC, Nwede IGN (2017) Human resource management practices and organisational innovation in banks in Bayelsa State. *Hum Resource Manage* 3(8)
17. Fadare OA (2015) Impact of ICT tools for combating cybercrime in Nigeria online banking: a conceptual review. *Int J Trade Econ Financ* 6(5)
18. Ibrahim U (2019) The impact of cybercrime on the Nigerian economy and banking system. *NDIC Q* 34(12):1–20
19. Imran SM, Sana R (2013) Impact of electronic crime in Indian banking sector—an overview. *Int J Bus Inf Technol* 1(2)
20. Niranjanamurthy M, Chahar D (2013) The study of e-commerce security issues and solutions. *Int J Adv Res Comput Commun Eng* 2(7):2885–2895
21. Olubisi FO (2015) History and evolution of banking in Nigeria. *Academ Arena* 7(1):9–14
22. Sethi N (2021) Cyber security analysis in banking sector. *Int J Adv Res Comm Manage Soc Sci (IJARCMSS)* 04(03):59–64
23. Sikdar P, Makkad M (2015) Online banking adoption: a factor validation and satisfaction causation study in the context of Indian banking customers. *Int J Bank Mark* 33(6):760–785
24. Tendulkar R (2013) Cyber-crime, securities markets and systemic risk. *CFA Digest* 43(4):35–43
25. Wapmuk SE (2017) Banking regulation and supervision in Nigeria: an analysis of the effects of banking reforms on bank performance and financial stability. University of Salford, United Kingdom

26. Adegoke TG (2014) Effects of occupational stress on psychological well-being of police employees in Ibadan metropolis, Nigeria. *Afr Res Rev* 8(1):302–320 .
27. Raghavan AR, Parthiban L (2014) The effect of cybercrime on a Bank's finances. *Int J Curr Res Acad Rev* 2(2):173–178
28. Hassan RG, Khalifa OO (2016) E-Government-an information security perspective. *Int J Comput Trends Technol (IJCTT)* 36(1):1–9
29. McGuire M, Dowling S (2013) Cyber crime: A review of the evidence. Summary of key findings and implications. *Home Office Res Rep* 75:1–35

Object Detection Using TensorFlow 2 and Amazon SageMaker



Vartika Goel, Deepak Arora, and Sheenu Rizvi

Abstract Many frameworks with packaged pre-guided models have been created to give users fast access to transfer learning considering the rapidly expanding field of object detection techniques. For instance, three well-known computer vision systems with trained models are GluonCV, Detectron2, and the Object Detection API for TensorFlow, as well. The TensorFlow2 Object Detection API is an update to the TensorFlow Object Detection API. One of the cutting-edge object identification algorithms that can be trained using the TensorFlow2 Object Detection API is EfficientDet from Google Brain (Implemented here). Authors created and implemented an EfficientDet model with the help of TensorFlow Object Detection API using Amazon SageMaker. It is constructed on top of TensorFlow 2, which facilitates its creation, training, and deployment. Because it is designed on top of TensorFlow 2, creating, training, and deploying object detection models is simple. SageMaker is a completely managed tool that lets data scientists and developers quickly build, direct, and implement ML models. To make it simpler to create high-quality models, SageMaker takes the labour-intensive tasks out of each stage of the ML process. Transfer learning on numerous pre-guided models accessible in TensorFlow Hub is made possible by object detection with TensorFlow in SageMaker. The head of the TensorFlow model that handles object detection is replaced based on the amount of class labels present in the guiding data. Based on fresh guiding data, either the entire network—including pre-guided model—or just the top layer (object recognition head) can be fine-tuned. Authors trained using a smaller dataset in this transfer learning method. Authors have talked over each step-in detail, including data collection and labelling with Ground Truth, making, and converting the data to TFRecord format, training as well as launching a special object detection model with the TensorFlow Object Detection API, and ultimately deploying the model.

Keywords Pre-trained models · EfficientDet model · TFRecord format · Object detection

V. Goel (✉) · D. Arora · S. Rizvi

Department of Computer Science & Engineering, Amity School of Engineering and Technology
Lucknow, Amity University, Noida, Uttar Pradesh, India
e-mail: vartika.goel2@s.amity.edu

1 Introduction

When building an ML model, data is gathered from a variety of trustworthy sources, it is processed to make it suitable for modelling, a modelling method is chosen, the model is built, performance measures are calculated, and the best performing model is chosen. Once the model has been put into manufacturing, model maintenance is crucial. As there is a chance that the model will slowly become out-of-date, machine learning model maintenance involves keeping the model current and relevant in tune with source data changes. As the number of models increases, the configuration management of ML models becomes increasingly crucial to model administration.

One common use of machine learning is in object detection, which desires to find as well as recognize special objects in images and videos. A robust deep learning system called TensorFlow 2.0 offers tools for creating and refining object detection models. A cloud-based machine learning tool called Amazon SageMaker creates it simple to create, guide, and use ML models at scale. As we build an object detection model with Amazon SageMaker and TensorFlow 2.0, we will dig thoroughly into the steps in this part. The steps in developing an object recognition model using TensorFlow 2.0 and Amazon SageMaker are data prepping, model selection, model training, model assessment, model deployment, model optimization, model monitoring, and model upkeep. The process of finding and identifying things in an image or a video stream is known as object detection in computer vision. The famous open-source machine learning library TensorFlow received a significant update in 2019 with the release of TensorFlow 2.0 by Google. It adds several new features and enhancements that make building and implementing machine learning models simpler, more logical, and more powerful.

Machine learning models can be built, trained, and deployed at scale by scientists and engineers using Amazon SageMaker, a completely managed tool provided by Amazon Web Services (AWS). From preparation of information to model guidance to deployment of model and administration, it offers a variety of tools and services that streamline the process of developing and deploying ML models. One of the main features of Amazon SageMaker is its ability to provide a comprehensive group of tools for developing and training models of machine learning. These resources include pre-built algorithms for typical machine learning tasks, like text analysis and image classification, as well as a Jupyter Notebook environment for creating and evaluating unique models.

2 Literature Survey

The development of machine learning models for object recognition using TensorFlow 2 and Amazon SageMaker is the subject of numerous connected research projects that are currently in process. These articles employ TensorFlow Object Detection API and SageMaker services to train, test, and implement a machine

learning model to identify objects and assess their confidence in an image or video. This paper is inspired from the published paper named “Object Detection using TensorFlow” [1] and writers of this article provided a system that can assist us in keeping track of our possessions and improve productivity by minimising the time lost neglecting or searching for them. With the intention of being able to correctly recognize numbers from handwritten photos, a neural network model is built and trained. For this, Keras served as the front end and the Tensor Flow grammar was used. The trained model can identify the class of a handwriting digit from a photograph of the digit, or it can identify the class of the input image. Another published paper talks about object detection with deep learning and techniques for detecting objects [2].

The other published paper named “A Comparative Study of Various Object Detection Algorithms and Performance Analysis” [3] that goes over a number of object recognition methods, including YOLO v3, Single Shot Detector (SSD), Fast R-CNN, and Faster R-CNN. These methods are compared, and the outcomes and performances of each are examined. Another published paper talks about the same, i.e., Faster R-CNN [5].

There is one more published paper named “Training and Deploying Models using TensorFlow 2 with the Object Detection API on Amazon SageMaker” on AWS [4] that explains all the steps for training and deploying ml models for object detection using the mentioned methods. Research papers [6–10] talks about object detection, object detection-based algorithms, object recognition, object detection using deep learning, TensorFlow 2, and many other techniques. Papers [11, 12] discussed about rich feature hierarchies for object detection and semantic segmentation.

A research paper named “Object Recognition using TensorFlow” in IEEE Integrated Stem Education Conference (ISEC) [13] developed a method for object identification using the TensorFlow API. The execution of these strategies would be very challenging. This paper’s advantage is that it is very straightforward to comprehend and implement. Published papers [14–18] proposed a system for detecting real-time objects using TensorFlow and tracking using image processing.

3 Proposed Methodology

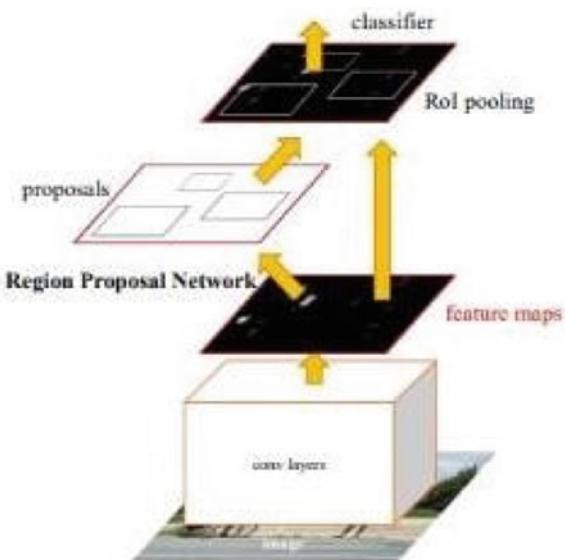
The major methodology used in making this project is TensorFlow 2.0 and Amazon SageMaker. The object recognition model used is Faster R-CNN that improves on Fast R-CNN by employing a region proposal network (RPN) with the feature maps created by the complex layer to determine a region-based object classification. The development of ml model is carried out according to the steps that include data collection, data preparation, model selection, model training, evaluation, tuning, deployment, as well as monitoring. Several cutting-edge object recognition models, including TensorFlow Object Detection API is used that includes Faster R-CNN, Mask R-CNN, and SSD as these pre-guided models can recognize a variety of objects, including people, vehicles, animals, and more. They have been trained on big datasets.

Amazon SageMaker has an embedded Jupyter authoring notebook instance that can access the data sources quickly for investigation and analysis without having to manage servers. S3 containers and Amazon Lambda are incorporated into the model, and Sagemaker helps build more accurate and efficient ML model. TensorBoard can provide the visualizations and data needed for the machine learning process. Object detection is done through the TensorFlow API and model training, evaluation, and deployment is carried out with the help of SageMaker which finally builds a machine learning model for detecting objects in photos or videos easily. Figure 1 displays the architecture of object detection model which is used in the project.

The model distribution and model guidance sections of SageMaker are highlighted in the section with that name. In SageMaker, a training job is built to instruct a model. These details are part of the teaching job include Amazon Simple Storage Service bucket URL where the instructing material is kept; the computing tools one wants SageMaker to employ when guiding models. ML estimate entities referred to as compute resources and are handled by SageMaker; the S3 bucket's URL where the output of the task is to be stored; the location where the training code is kept in the Amazon Elastic Container Registry.

Following the creation of the guiding job, SageMaker starts the ML estimate instances and trains model using the instructing code and dataset. The S3 bucket that is designated for that reason receives the model objects and other results that are produced as a result. A training assignment can be made using the SageMaker interface. SageMaker copies the complete dataset on ML estimate instances as usual when one starts a guiding job with API. This variable can be modified using the low-level SDK.

Fig. 1 Architecture of faster R-CNN



4 Implementation of the Proposed Model

The project works by detecting objects and their confidence percentage in images and videos by using TensorFlow Object Detection API and Amazon SageMaker. Amazon SageMaker uses its various services like SageMaker Ground Truth and notebook instances for different tasks. This project is implemented in a series of steps by making use of TensorFlow 2.0, Amazon SageMaker, and its services that include SageMaker Ground Truth, notebook instance. This project also makes use of TensorBoard that provides the visualizations and data needed for the machine learning process; Amazon S3 bucket for storing images or videos for object detection; and many other services of AWS. The phases of working are as follows.

4.1 *Label Images using SageMaker Ground Truth*

SageMaker Ground Truth is utility for data labelling, where false data is generated without actively collecting or labelling data from the actual world. Thousands of automatically named synthetic pictures can be produced by SageMaker Ground Truth. The data labelling processes and staff are created and managed with the aid of SageMaker Ground Truth. The screenshot describes an instance of labelling job configuration in Ground Truth (Fig. 2).

4.2 *Generate the Dataset TFRecords and Label Map Using SageMaker Processing*

First the images and annotations are merged in our dataset and transformed them into the TFRecord file format before it is used in the TensorFlow API of Object Detection. This format is a straight manner to keep a sequence of binary records that rises the speed at which information can be viewed and processed.

A processing script is first created that reads the data and transform it to the TFRecords format before SageMaker Processing is used. In the script, create tf example method is defined that accepts a label ID and an image file location, scans the picture, encodes it in JPEG format, and generates a tf.train. Then, a function called write TFRecords file is defined that takes an input directory containing the image files, an output file path for the label map, an output file path for the TFRecords file, and loops over the label directories and image files, creating a TFRecords example for each image and writing it to the output file. Once the processing script is ready, SageMaker's ScriptProcessor is used to build a processing task. This processor enables to execute the processing script on infrastructure that is handled by SageMaker in a containerized context. The instance type, count, and IAM role is provided that is used

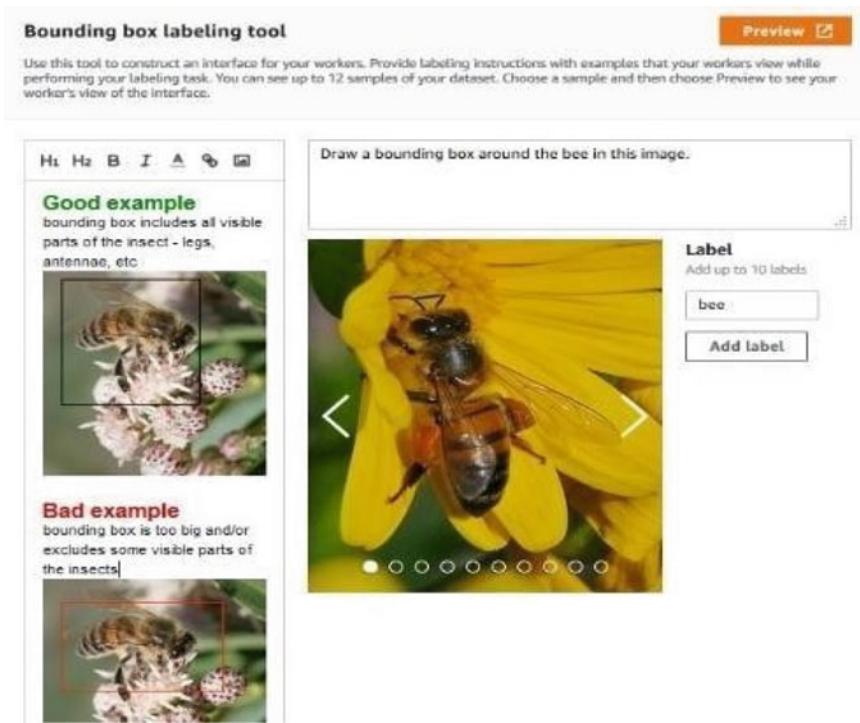


Fig. 2 Labelling job configuration in Ground Truth

during the processing task, as well as the Docker image and command that authors want to use (in this case, python3).

Additionally, we designate the input and exit data places, which may be local files or Amazon S3 buckets. For the purposes of this illustration, we'll presume that our input information is kept in an S3 bucket and our exit data will be transferred to another S3 bucket. SageMaker will start an EC2 instance or a fleet of instances once our processing task is made, pull the Docker image, and execute our processing script on the instance (s). The output address that has been selected will receive the output data. With the help of the SageMaker interface, APIs, or SDKs, we can keep track of the processing job's progress.

4.3 Fine-Tune an EfficientDet Model with TensorFlow 2 on SageMaker

SageMaker is used in this illustration to hone an EfficientDet model using data from the own dataset. It is presumed that the data that is produced with SageMaker Processing as shown in the prior example already exists in the TFRecords format.

Defining the training routine is the first step-in optimising the model. In addition to setting up the optimizer and loss functions and defining the training loop, this script should also describe the model design and import the pre-instructed weights.

The TensorFlow Object Detection API, which offers a high-level structure for training object recognition models, will be used in our situation. Additionally, we have the option to designate the input data location, which must be an S3 bucket holding our TFRecords files, and the output location, which must be another S3 bucket for the model milestones and other artefacts. Once our estimator has been specified, we can use the fit technique to launch the training process. SageMaker will execute our training script on one or more instances after launching one or more instances, pulling the TensorFlow 2 Docker file (s).

4.4 Developing a TensorFlow 2 Object Detection API Docker Container

Author specified a Dockerfile in the project location before building the Docker container. The instructions for creating the container image are detailed in this file, including how to install the required packages, establish the working directory, duplicate the source code, and expose the required ports. The original TensorFlow 2.4.1-gpu Docker binary serves as the foundation for this Dockerfile. Then clone the object detection API source is cloned, download API, as well as install the needed tools. The implementation and administration of our machine learning process can be made simpler by building a Docker container for the TensorFlow Object Detection API. For training and launching our object recognition models across various systems and environments, it offers a reproducible and portable ecosystem.

4.5 Monitor and Capture Model Training with TensorBoard and SageMaker Debugger

Gaining understanding of the model's performance and identifying problems during training can be accomplished with the aid of TensorBoard and SageMaker Debugger. It offers a potent collection of tools for managing the machine learning processes and selecting hyperparameters, model architecture, and other elements of the training procedure after doing thorough research.

4.6 Deploy Model on a SageMaker Endpoint and Visualize Predictions

TensorFlow models can be easily executed using SageMaker's managed TensorFlow yielding infrastructure. One can take many actions to publish your learned model on a SageMaker endpoint and see predictions like build a SageMaker model first: Make a SageMaker model first using the position of Docker document containing your learned model in the Amazon Elastic Container Registry (ECR). The model can be created using the `sagemaker.tensorflow.model.TensorFlowModel` class; Distribute the SageMaker endpoint: Next, activate the SageMaker endpoint by invoking the model object's `deploy()` function; Make use of the `predict()` method of the predictor object to create projections after endpoint has been launched; Show forecasts in visual form: Lastly, the predictions can be shown in visual form by arranging the test pictures along with their anticipated labels.

5 Security Measures

For the protection of data and access control, object detection requires a strong layer of security. When employing SageMaker, the proposed work has utilized the shared responsibility model, which is applicable to data protection and infrastructure security. The model includes all security configuration and management tasks that safeguard data while using Amazon SageMaker, including safeguarding AWS account credentials, using MFA with each account, and utilizing cutting-edge managed security services like Amazon Macie, which helps find and secure sensitive data that is stored in Amazon S3.

In the proposed work, access to SageMaker was controlled using the Identity and Access Management (IAM) feature of AWS, with IAM identity-based policies dictating which actions and resources were accessible or not. Who has access to what is specified by administrators using AWS JSON policies. When a SageMaker notebook instance, processing job, training job, hosted endpoint, or batch transform job resource is created, a role must be chosen to permit SageMaker to access SageMaker on your behalf. SageMaker gives users a list of available roles to choose from if there are already existing service roles or roles that are tied to services. It is critical to select a role that enables access to the necessary AWS activities and resources. There are numerous Amazon SageMaker API operations utilized in the proposed model, the associated AWS actions for which permissions are provided to do the action, and the AWS resources for which one can grant permissions.

AWS managed policies that are particular to Amazon SageMaker are many and connected to users in accounts. SageMaker geospatial resources and the related activities are completely accessible with `AmazonSageMakerFullAccess`. While supporting buckets and objects with certain SageMaker tags, this does not offer unlimited access to Amazon S3. All IAM roles are permitted to be sent to Amazon SageMaker under

this policy; however, only IAM roles that contain “AmazonSageMaker” are permitted to be passed to the AWS Glue, AWS Step Functions, and AWS RoboMaker services. The rights needed to utilize SageMaker Ground Truth have been added by these AWS managed policies. The execution roles defined through the SageMaker console employ the policies, which are accessible through the AWS account. The task involves creating and configuring a private VPC to manage access to data and job containers so they are not reachable through the internet. To specific IAM policies, permissions are added for compilation jobs executing in Amazon VPCs.

For controlling control access, activity monitoring, and reporting across the ML life cycle for detection of objects, Amazon SageMaker offers solutions specifically designed for ML governance. Utilising Amazon SageMaker Role Manager, one can control least-privilege access for ML practitioners. Amazon SageMaker Model Cards and Amazon SageMaker Model Dashboard provide centralized dashboards that provide insight into your models.

6 Results and Discussion

A model that can identify bees from RGB pictures is learned using a dataset from iNaturalist.org. 500 photos of bees that were submitted by iNaturalist users for the purpose of documenting the inspection and recognition make up this collection. A single.zip package containing the collection to Amazon S3.500.jpg picture images is uploaded and an output.manifest file is included in the archive. The 3_predict/test_images notebook subdirectory also contains 10 test images that are used to illustrate the model’s forecasts (Fig. 3).

In Fig. 4, all objects in the images are detected and squared along with the confidence percentage. The object detection is possible because of TensorFlow 2 and Amazon SageMaker and its services. Object detection has a lot of scope in future with more robustness, precision, and accuracy in a scalable environment (AWS).

Debugger gives us the ability to download TensorBoard information into a selected Amazon S3 location while task is operating and use TensorBoard to track its progress in real time. The log path that is specified is used when setting the TensorBoardOutputConfig object as the—logdir parameter. This enables to evaluate the accuracy of the forecasts and the Ground Truth information (right picture in the screenshot) and predictions (left image) (Fig. 5).

The edge-to-edge process of gathering as well as labelling data with the help of Ground Truth, arranging and converting the information to TFRecord format, guiding and launching unique object recognition model with the help of TensorFlow Object recognition API was addressed in the final result.



Fig. 3 Input image from the dataset

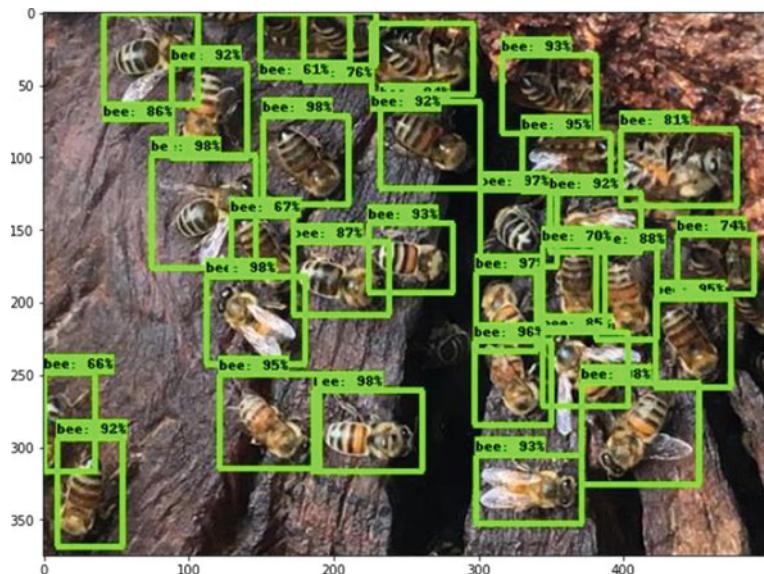


Fig. 4 Output image (objects detected with confidence %)

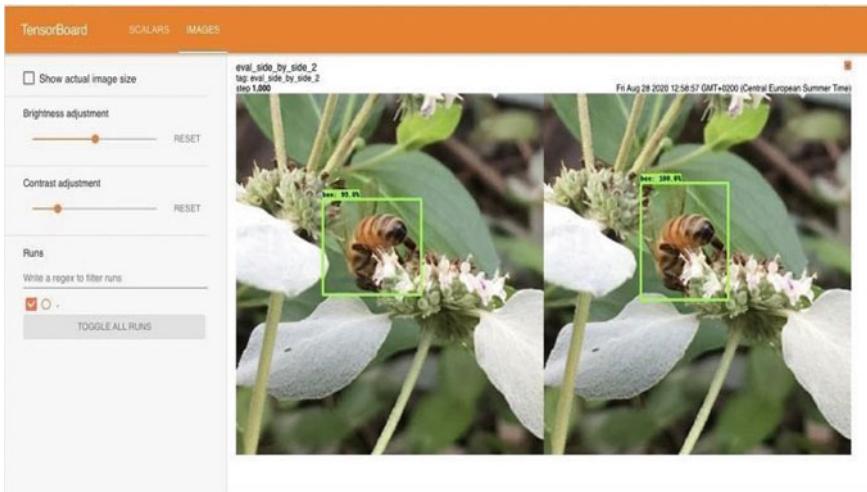


Fig. 5 Comparing SageMaker Ground Truth data

7 Conclusion and Future Scope

The crucial job of object detection in computer vision entails locating and recognizing things in a photo or video. The development of precise and effective detection models of object that can automate many real-world applications has become feasible thanks to improvements in machine learning methods and the growing abundance of data. A user-friendly UI is offered by the open-source machine learning framework TensorFlow 2 for the creation and training of machine learning models. A full suite of tools is available for creating, honing, and executing ML models at scale on cloud-based ML infrastructure known as Amazon SageMaker.

Firstly, the training data is arranged before building an object recognition model with Amazon SageMaker and TensorFlow 2. This job can be accomplished using marking tools like labelling. TensorFlow 2 is used to create and train the object recognition model once the training data is ready. Building and training object recognition models is made simpler by the pre-built object detection API provided by TensorFlow 2. This training system can be built up using Amazon SageMaker, and the training task can be performed on cloud-based instances that are designed for machine learning workloads. TensorFlow 2 environments that have already been created and configured with well-known machine learning frameworks are available from SageMaker. Using the tracking tools included with Amazon SageMaker, we can keep track of the model's development as it is being trained. After training is finished, the trained model can be arranged with the help of Amazon SageMaker to a scalable and safe operational system. In summation, building an object detection model with Amazon SageMaker and TensorFlow 2 is an easy process that can be completed by adhering to a few basic stages. With the help of these tools, authors can easily create and train

precise object recognition models that can be applied to a few natural worlds uses, like robotics, monitoring systems, and autonomous vehicles. Data was gathered and labelled using Ground Truth, prepared, and converted to TFRecord format, and a bespoke object recognition model was trained and deployed using the TensorFlow Object Detection API.

The future scope is to offer the world's top machine learning tool. Every coder should have software that will evolve into a new superpower software that will make machine learning into a business that is as developed as web development, rather than just a specialized field. TensorFlow will always be backwards-compatible in the future. TensorFlow 2 code will continue to function in its current form as of 2023 thanks to Google's commitment to complete backwards interoperability. There will not be a conversion script to execute or hand adjustments to make.

Later this year, the production edition will be published. Some possible future applications for object detection include object detection using explainable AI, object detection for ethical AI, Multimodal object detection, object detection with small data, object detection for precision medicine, self-supervised learning for object recognition. Overall, there is a broad and bright future for object detection using TensorFlow 2.0 and SageMaker, with possible uses in many different situations and domains. We can anticipate seeing more precise, effective, and robust object recognition models with cutting-edge features and powers as these technologies continue to develop.

References

1. Pachipala Y, Harika M, Aakansha B, Kavitha M (2022) Object detection using TensorFlow”, International conference on electronics and renewable systems (ICEARS), Tuticorin, India, April 2022
2. Zhao Z-Q, Zheng P, Xu S-T, Wu X (2019) Object detection with deep learning: a review. *IEEE Trans Neural Netw Learning Syst* 30:3212–3232
3. John A, Meva D (2020) A Comparative study of various object detection algorithms and performance analysis. *Int J Comput Sci Eng* 8:158–163
4. Hamiti S, Hamzaoui O (2021) Training and deploying models using TensorFlow 2 with the object detection API on Amazon SageMaker. In: Amazon SageMaker, Artificial Intelligence, TensorFlow on AWS, 9 Feb 2021
5. Girshick R (2015) Fast R-CNN. In: IEEE international conference on computer vision (ICCV), Santiago, USA, pp 1440–1448
6. Raviteja N, Lavanya M, Sangeetha S (2020) An overview on object detection and recognition. *Int J Comput Sci Eng* 8(2):42–45
7. Kaur A, Kaur D (2020) Yolo deep learning model based algorithm for object detection. *Int J Comput Sci Eng* 8(1):174–178
8. Rane M, Patil A, Barse B (2019) Real object detection using TensorFlow. Lecture notes in electrical engineering and computer science, pp 978–981
9. Madan V, Moura J, Khetan A (2022) Transfer learning for TensorFlow object detection models in Amazon SageMaker. In: Amazon SageMaker, Artificial Intelligence, Foundational, 4 Nov 2022
10. Uijlings JRR, van de Sande KEA, Gevers T, Smeulders AWM (2013) Selective search for object recognition. *Int J Comput Vis* 104:154–171

11. Lu JY, Ma C, Li L, Xing XY, Zhang Y, Wang ZG, Xu JW (2018) A vehicle detection method for aerial image based on Yolo. *J Comput Commun* 6(11):98–107
12. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE conference on computer vision and pattern recognition, Columbus, USA, pp 580–587
13. Albayrak NE (2021) Object recognition using TensorFlow. In: 2020 IEEE integrated stem education conference (ISEC), Princeton, NJ, USA, April 2021
14. Masud U, Saeed T, Malaikah HM, Ulislam F, Abbas G (2022) Smart Assistive system for visually impaired people obstruction avoidance through object detection and classification. *IEEE Access* 10:13428–13441
15. Jawale P, Chaudhary HP, Rajput N (2020) Real-time object detection using TensorFlow. *Int Res J Eng Technol (IRJET)* 7(8)
16. Ren S, He K, Girshick RB, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell*
17. Vijayalaxmi K, Anjali B, Srujana P, Kumar R (2014) Object detection and tracking using image processing. *Glob J Adv Eng Technol*, 1424–1430
18. Pipalia DS, Simaria RD (2015) Real time object detection tracking system (locally and remotely) with rotating camera. *Int J Recent Innov Trends Comput Commun* 3(5):3058–3063

Fake News Detection Using Machine Learning



Hanish Jindal, Mittali Mangla, and Gurpreet Singh 

Abstract Social media has grown so far today which has raised major concerns about fake news. Social media platforms allow consumers to share facts, some of which are misleading and have no relevance to reality impacting social security of people. It can have significant impacts on public opinion and decision-making. Fake news is false or misleading facts which can damage the reputation or make money through advertising impacting cyber security. Fake news is spreading day by day, and it has become very difficult to differentiate between what is fake news and what is real. Traditional methods of detecting fake news, such as fact-checking, have proven to be insufficient, making it crucial to develop new approaches. Using machine learning, this paper will provide an algorithm to solve this problem. We are trying to solve the problem with the best algorithm which should be accurate and precise. By reviewing and checking for limitations, we can improve our ability to detect fake news.

Keywords Logistic regression · Random forest · Multinomial Naïve Bayes · Stochastic gradient decent · Decision tree · Social security · Hate crime · National security · Prevent violence · Cyber security

1 Introduction

The use of social media is increasing rapidly which has a lot of advantages like important news can reach billions of people very quickly by sharing or tagging. People can socialize no doubt, there are a lot of advantages of social media, but you

H. Jindal (✉) · M. Mangla · G. Singh
Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura,
Punjab, India
e-mail: hanishjindals@gmail.com

M. Mangla
e-mail: mitalimangla01@gmail.com

G. Singh
e-mail: gurpreet.1309@chitkara.edu.in

know bad things also come along with good. In this case, the bad thing about social media is fake news or false information.

Fake news is false or misleading facts which can damage the reputation of a person or entity or make money through advertising [1]. Some people are spreading fake news for their political propaganda, hate a particular part of society, harm someone's reputation for personal benefit and fake advertising to sell the product. This also impacts a person's security on internet. There are a lot of platforms on which some people spread fake news like Facebook, Instagram, Twitter and WhatsApp [2].

In recent years, the dissemination of false information has been a major concern to society, endangering our communities' faith in democracy and creates need of cyber security. It has made it difficult for citizens to acquire reliable information and make educated decisions when voting in elections or participating in public discourse. We have therefore presented a machine learning algorithm that can aid in the detection of fake news, one vital part of curtailing its spread. Using ML technology, this algorithm can assess text from news articles and work out whether the story is accurate or deceiving. We believe that this contribution to the field of fake news detection will contribute to restoring confidence in journalism and political systems.

There are many instances for example where a video of a building collapsing got viral on social media saying that the video belongs to Turkey where a recent earthquake happened. This video got viral, getting thousands of views and spreading false information among people. People are sharing this unintentionally/intentionally without knowing the fact. This may create a bad impression on people about the exact situation. But, the fact is this video belongs to Japan. This happened on April 19, 2016, when a powerful winds whip through Tokyo, tear down scaffolding on the 9-storey building [3] (Fig. 1).



Fig. 1 Fact Check (source DFRAC) [3]

The effects of fake news can be severe, including the manipulation of political opinions and the fueling of societal conflicts. Which can lead to public protests, Mob lynching, killings, etc. Some people very easily believe this fake news without checking it. Although, it is not possible for the normal public to check whether the news is fake or real. This problem needs to be addressed on a priority basis.

Now we understand that fake news is a serious concern that must be addressed. Traditionally, fake news was trying to be detected using fact-checking which consist of cross-checking the news on another platform, checking on newspapers/website/channel, checking on google, etc., which become a pain to check a lot of news and is expensive as a lot of workforces is required for getting the correct information. These methods do not always give us correct results [4].

In this paper, we are trying to solve this issue of cyber security using a machine learning approach. This approach will give comparatively better results regarding the credibility of the news. To detect false news, it is important to analyze connections between misleading news headlines along with its content. This analyzed data can then be used to develop a machine learning model which can determine the truthfulness of a piece of news based on its headline and content [5]. The following section will introduce you to a unique machine learning solution using the python programming language to detect fake news. This project not only showcases the power of python in solving real-world problems but also highlights the significance of identifying and combating false news.

We are using a logistic regression algorithm of machine learning to check the credibility of news whether it is fake or real using titles and text to train our model. Logistic regression is a method which uses multiple factors to predict the probability of an outcome. It is commonly used for classifying things into two groups, such as true or false. It is helpful because it can model complex relationships between factors and outcomes and is easy to use and understand.

There are many machine learning algorithms that can be used for this purpose. These include decision trees, support vector machines, random forests and neural networks, among others. Each algorithm has its own merits and demerits, and selecting the correct one depends on the specific context and requirements of the problem at hand.

Logistic regression has several advantages over other machine learning algorithms, such as decision trees and neural networks. It is a simple and interpretable model that can be easily understood by non-experts. It also works well with small to medium-sized datasets and can handle both continuous and categorical variables. Additionally, the coefficients of the logistic regression model can provide insights into the relative importance of the independent variables in predicting the outcome variable.

Logistic regression has elicited significant attention from researchers in the realm of fake news detection. Its capacity to accurately discern news articles as either genuine or fabricated, based on a variety of textual features for instance lexical and syntactical features, has been widely explored in combination with other machine learning algorithms such as neural networks, support vector machines and random forests for increased accuracy. Nevertheless, its constraints aside, logistic regression

continues to be an effective tool for recognizing counterfeit news stories and offers huge prospects for further investigation in this domain.

2 Related Work

Sharma et al. in a survey inspect the different techniques like intervention-based approaches which are used for identifying, containment and mitigating fake news. It reviews the current approaches in fields such as machine learning, natural language processing, social network analysis and crowdsourcing. The authors also explore various methods such as fact-checking and source verification on how to counter false news. At last, this paper concludes that considering the obstacles in tackling false news and suggests potential paths for future research [6].

Wang et al. in an article give an overview of the recent advances in using deep learning (DL) to detect fake news. Authors review the different types of DL models used such as convolutional neural networks (CNN) and recurrent neural networks (RNN), discussing various approaches and improved results by containing extra textual and contextual features such as history, subject, speaker and so on for feature extraction and model training. The paper highlights the challenges faced in this field, such as the need for large, annotated datasets and the difficulty of dealing with adversarial attacks [7].

Zhang et al. propose a novel approach to detecting fake news that combines linguistic as well as visual information. The authors use both text and image feature to train a multi-perspective model that captures different aspect of fake news. They evaluate their approach on a benchmark dataset and demonstrate its effectiveness in detection of fake news [8].

Ihsan Ali et al. in an article provide a comprehensive survey of the existing methods for fake news detection on social media platforms. The authors review the different approaches based on content analysis, social network analysis and machine learning and also highlight the challenge in this field such as the need for large-scale data collection and annotation. They discuss the importance of considering the social context in which fake news is propagated and the need for interdisciplinary collaboration in addressing this problem [9].

Nikos Deligiannis et al. in an article propose a novel approach in the detection of fake news that leverages geolocation information and deep learning. The authors combine text and location features to train a model that can identify fake news that is likely to spread and go viral. They evaluate their approach on a large-scale dataset and demonstrate its effectiveness in detecting fake news on social media. The authors provide insights into the characteristics of fake news stories that are more likely to go viral in public. It is highlighting the importance of considering location information in the design of fake news detection systems [10].

These papers [11, 12] give a novel approach that uses convolutional neural networks (CNNs) and graphs convolutional networks (GCNs). The authors used GCNs and CNNs to capture social network information and analyze the content

of news articles. They evaluate their approaches on datasets like news articles, and tweets and how they outperform existing methods.

These papers [13–15] come up with various solutions for fake news detection by combining many machine learning algorithms such as decision trees, Naive Bayes and random forests. These papers also show the use of logistic regression for increasing accuracy. Their approach extracts various features from news articles, including text and metadata. They evaluate their method on a dataset of news articles.

Ahmad et al.—This paper provides an approach for solving the fake news detecting issue which relies on machine learning based on granular computing. These authors use various machine learning algorithms such as k-nearest neighbors (KNN), random forests and support vector machines (SVMs) to analyze the content of news articles. They evaluate their approach on a dataset of news articles and show that it achieves high accuracy in detecting fake news [16].

This paper by Alim Al Ayub Ahmed et al. provides an extensive review of the recent studies in the domain of fake news detection. It looks particularly into the application of machine learning algorithms and techniques for detecting fake news, summarizing key findings and various methods deployed in various studies, such as natural language processing, social network analysis and ensemble methods. The authors also identify existing limitations and challenges in the current research and put forward recommendations for further improving accuracy and efficiency of fake news detection using machine learning. All in all, it gives a comprehensive overview of current progress made in this field, making it a great resource for those wishing to explore fake news detection using machine learning [17].

Gap Analysis: Based on our analysis of the above research papers and some important information, we have come to the following conclusions:

1. Most papers have used small to medium-sized datasets, which may not represent the vast amount of data available on the internet.
2. Many of the reviewed studies did not provide details on the selection of features or the training process used to develop the detection algorithms.
3. The reviewed studies focused on the development of detection algorithms but paid little attention to the practical challenges of implementing these algorithms in real-world scenarios. It is necessary to consider factors such as computational cost, scalability, and user interface design before implementing fake news detection systems effectively.
4. There is a lack of research that has tested the robustness of the proposed algorithms on adversarial examples, which are designed to evade detection.

These limitations suggest potential areas for further research, which can help in improving the effectiveness and robustness of fake news detection algorithms. So, we have proposed a solution using logistic regression for better accuracy with a comparatively large dataset for training and testing purposes.

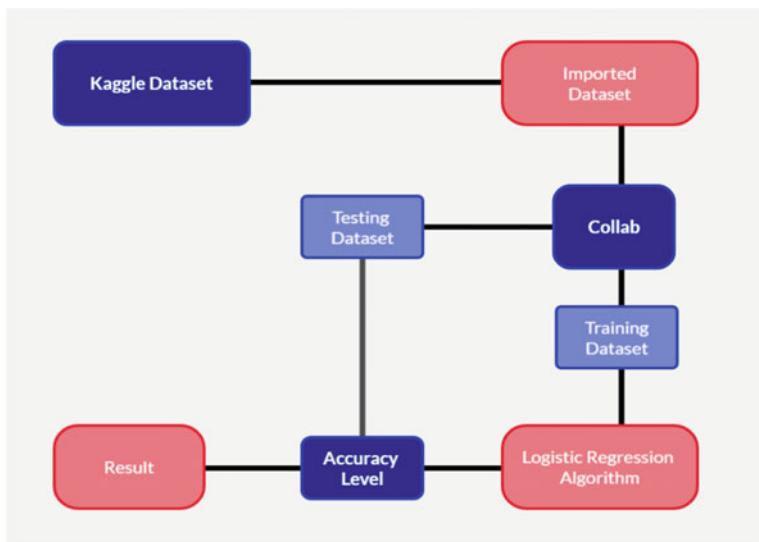


Fig. 2 Flowchart

3 Methodology

3.1 Workflow

See Fig. 2.

3.2 Proposed System

1. The dataset was imported from Kaggle, modified and saved as fake_news_data.csv.
2. We used Google Colab to execute Python code.
3. Datasets are then divided into training datasets and testing datasets.
4. For a better understanding of the dataset, visualizations are made in Google Colab.
5. Finally, we conclude with accuracy.

3.3 Logistic Regression

This algorithm is a commonly used method in machine learning for modeling the connection between a binary dependent variable and many independent variables. It

news_dataset.head()					
	id	title	text	label	content
0	0	LAW ENFORCEMENT ON HIGH ALERT Following Threat...	No comment is expected from Barack Obama Membe...	1	LAW ENFORCEMENT ON HIGH ALERT Following Threat...
1	1		Did they post their votes for Hillary already?	1	Did they post their votes for Hillary already?
2	2	UNBELIEVABLE! OBAMA'S ATTORNEY GENERAL SAYS MO...	Now, most of the demonstrators gathered last ...	1	UNBELIEVABLE! OBAMA'S ATTORNEY GENERAL SAYS MO...
3	3	Bobby Jindal, raised Hindu, uses story of Chri...	A dozen politically active pastors came here f...	0	Bobby Jindal, raised Hindu, uses story of Chri...
4	4	SATAN 2: Russia unveils an image of its terrif...	The RS-28 Sarmat missile, dubbed Satan 2, will...	1	SATAN 2: Russia unveils an image of its terrif...

Fig. 3 Kaggle dataset [19]

can be employed to predict the chances of something happening, or not, based on certain variables. This model assumes the outcome follows a binomial distribution with two respective options (0 or 1). The independent variables could be either categorical or continuous. Logistic regression is usually fitted utilizing maximum likelihood estimation, where the parameter values optimally fit the likelihood of seeing the data [18].

4 Implementation

First, we import dataset from Kaggle (Fig. 3).

Description of dataset:

1. **id:** An article's unique identifier
2. **title:** An article's title
3. **text:** An article's text
4. **label:** A label tells following:
 - 1: Fake news
 - 0: Real News.

Here is the implementation logistic regression model (Fig. 4).

5 Results

The accuracy of the proposed fake news detection model was evaluated using a training and testing dataset. The dataset consisted of more than 72,000 news articles, half of which were real and half were fake. The dataset was randomly split into 70% for training and 30% for testing (Figs. 5, 6 and 7).

```
[16] # converting the textual data to numerical data
vectorizer = TfidfVectorizer()
vectorizer.fit(X)
X = vectorizer.transform(X)

#Splitting the dataset to training & test data
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.3, stratify=Y, random_state=2)

#Training the Model: Logistic Regression
model = LogisticRegression()
model.fit(X_train, Y_train)

LogisticRegression()
```

Fig. 4 Implementation

```
[24] # accuracy score on the training data
X_train_prediction = model.predict(X_train)
training_data_accuracy = accuracy_score(X_train_prediction, Y_train)
print('Accuracy score of the training data : ', training_data_accuracy)

Accuracy score of the training data :  0.9634404768977878
```

Fig. 5 Training data accuracy score

```
[25] # accuracy score on the test data
X_test_prediction = model.predict(X_test)
test_data_accuracy = accuracy_score(X_test_prediction, Y_test)
print('Accuracy score of the test data : ', test_data_accuracy)

Accuracy score of the test data :  0.9450579917748718
```

Fig. 6 Testing data accuracy score

Comparison Table

Here is the comparison of different machine learning algorithms accuracy scores with the same dataset which we used for this paper (Fig. 8; Table 1).



```
cm = confusion_matrix(Y_test, X_test_prediction)
cm_display = ConfusionMatrixDisplay(cm)
cm_display.plot()
plt.show()
```

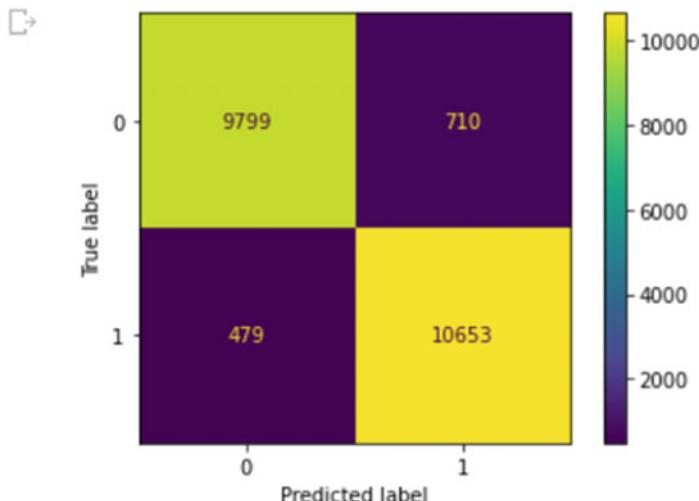


Fig. 7 Visualization on confusion matrix of 1-fake and 0-real news

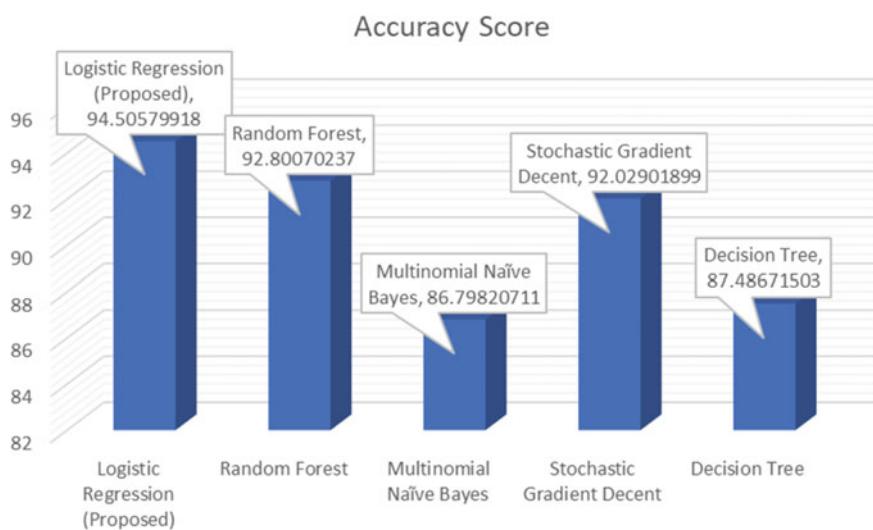


Fig. 8 Comparison between proposed algorithm and other algorithm accuracy

Table 1 Comparison of logistic regression with other algorithms

Algorithm	Accuracy score
Logistic regression (proposed)	94.50579917748718
Random forest	92.80070237050044
Multinomial Naïve Bayes	86.79820710688045
Stochastic gradient decent	92.02901899172866
Decision tree	87.48671503165288

6 Conclusion

Overall, the use of social media has many advantages, but it also comes with the risk of fake news or false information. Fake news can have severe consequences such as manipulating political opinions and fueling societal conflicts impacting cyber security. Traditionally, fake news detection involved fact-checking, which was expensive and time-consuming. However, machine learning approaches such as logistic regression can be used to analyze connections between misleading news headlines and content to determine the truthfulness of a piece of news. Logistic regression has several advantages over other machine learning algorithms and has gained significant attention in fake news detection due to its ability to accurately distinguish between the news articles as either real or fake depending on a range of textual features. Despite its limitations, logistic regression remains a powerful tool for identifying fake news articles and has the potential for further exploration in this area.

References

1. Fake news. https://en.wikipedia.org/wiki/Fake_news
2. Jain A, Shakya A, Khatter H, Gupta AK (2019) A smart system for fake news detection using machine learning. In: 2019 international conference on issues and challenges in intelligent computing techniques (ICICT), Ghaziabad, India, pp 1–4. <https://doi.org/10.1109/ICICT46931.2019.8977659>. <https://ieeexplore.ieee.org/document/8977659>
3. An old video of Japan building collapsing is getting viral as the earthquake in Turkey. Read-Fact Check. <https://dfrac.org/en/2023/02/09/an-old-video-of-japan-building-collapsing-is-getting-viral-as-the-earthquake-in-turkey-read-fact-check/>
4. López-Marcos C, Vicente-Fernández P (2021) Fact checkers facing fake news and disinformation in the digital age: a comparative analysis between Spain and United Kingdom. Publications 9(3):36. <https://doi.org/10.3390/publications9030036>
5. Goel S, Guleria K, Panda SN (2022) Anomaly based intrusion detection model using supervised machine learning techniques. In: 2022 10th international conference on reliability, infocom technologies and optimization (trends and future directions) (ICRITO), Noida, India, pp 1–5. <https://doi.org/10.1109/ICRITO56286.2022.9965050>
6. Sharma K, Qian F, Jiang H, Ruchansky N, Zhang M, Liu Y (2019) Combating fake news: a survey on identification and mitigation techniques. ACM Trans Intell Syst Technol 10(3), Article 21 (May 2019), 42 p. <https://doi.org/10.1145/3305260>

7. Nasir JA, Khan OS, Varlamis I (2021) Fake news detection: a hybrid CNN-RNN based deep learning approach. *Int J Inf Manage Data Insights* 1(1):100007. ISSN 2667-0968. <https://doi.org/10.1016/j.jjimei.2020.100007>
8. Giachanou A, Zhang G, Rosso P (2020) Multimodal fake news detection with textual, visual and semantic information. In: Sojka P, Kopeček I, Pala K, Horák A (eds) *Text, speech, and dialogue. TSD 2020. Lecture notes in computer science*, vol 12284. Springer, Cham. https://doi.org/10.1007/978-3-030-58323-1_3
9. Ali I, Bin Ayub MZ, Shivakumara P, Mohd Noor NFB (2022) Fake news detection techniques on social media: a survey. *Wirel Commun Mobile Comput* 2022:Article ID 6072084, 17 p. <https://doi.org/10.1155/2022/6072084>
10. Monti F, Frasca F, Eynard D, Mannion D, Bronstein MM (2019) Fake news detection on social media using geometric deep learning. *arXiv* <https://arxiv.org/abs/1902.06673>
11. Qian S, Hu J, Fang F, Xu C (2021) Knowledge-aware multi-modal adaptive graph convolutional networks for fake news detection. *ACM Trans Multimedia Comput Commun Appl* 17(3):Article 98 (August 2021), 23 p. <https://doi.org/10.1145/3451215>
12. Karnyoto AS, Sun C, Liu B et al (2022) Augmentation and heterogeneous graph neural network for AAAI2021-COVID-19 fake news detection. *Int J Mach Learn Cyber* 13:2033–2043. <https://doi.org/10.1007/s13042-021-01503-5>
13. Hasan Alrikabi A, Ateaa H (2022) Fake news detection based on the machine learning model. *Design Engineering* (Toronto). <https://www.jetir.org/papers/JETIRFN06022.pdf>
14. Sharma U, Saran S, Patil SM (2020) Fake news detection using machine learning algorithms. *Int J Creative Res Thoughts* 8(6):509–518
15. Rajeswari M, Catharine R (2022) Fake news detection using Naive Bayes algorithm with machine learning. *JETIR* 9(6). <https://www.jetir.org/papers/JETIRFN06022.pdf>
16. Ahmad I, Yousaf M, Yousaf S, Ahmad MO (2020) Fake news detection using machine learning ensemble methods. *Complexity* 2020:Article ID 8885861, 11 p. <https://doi.org/10.1155/2020/8885861>
17. Al Ayub Ahmed A et al (2021) Detecting fake news using machine learning: a systematic literature review. *arXiv preprint arXiv:2102.04458*. <https://doi.org/10.48550/arXiv.2102.04458>
18. Kaur B, Kaur G (2022) Heart disease prediction using modified machine learning algorithm. In: International conference on innovative computing and communications: proceedings of ICIIC 2022, vol 1. : Springer Nature Singapore, Singapore, pp 189–201. https://doi.org/10.1007/978-981-19-2821-5_16
19. Verma PK, Agrawal P, Amorim I, Prodan R (2021) WELFake: Word embedding over linguistic features for fake news detection. *IEEE Trans Comput Soc Syst* 8(4):881–893. <https://doi.org/10.1109/TCSS.2021.3068519>

Cloud-Based Integrated Real-Time Twitter Grievance Redressal with AWS EC2 and RDS Using Machine Learning Approach for Enhanced Security



Shambhavi Chauhan and Deepak Arora

Abstract Social media platforms, especially Twitter, have become a popular channel for customers to express their grievances related to products or services. Companies need to have an efficient grievance redressal system in place to address these complaints in real-time to ensure customer satisfaction and loyalty. This research paper presents the development of a cloud-based integrated real-time Twitter grievance redressal system using Amazon Web Services (AWS) and machine learning approach. The proposed system uses AWS cloud infrastructure to lever the huge volume of dataset created by tweets and machine learning algorithms to classify and prioritize them based on their severity. The system includes a web application that allows the grievance redressal team to view, categorize, and respond to the tweets efficiently. The efficiency of the planned system is evaluated via a case study. The results show that the system can effectively handle a huge volume of tweets and improve the grievance redressal process. The system's response time is significantly reduced, and the team can prioritize the tweets based on their severity and importance, leading to better customer satisfaction. Data security is a critical aspect of the proposed real-time application as it will be handling sensitive data of the users. Therefore, security measures such as encryption, MFA, and disaster recovery must be properly implemented and configured, in order to ensure the security of data of the suggested grievance redressal system. The suggested system has achieved an accuracy of 89.5%. This research paper contributes to the development of efficient social media grievance redressal systems using cloud infrastructure and machine learning algorithms. The proposed system can be easily integrated into existing customer relationship management systems, making it a viable solution for companies of all sizes. The system's ability to handle real-time data and provide quick responses can improve customer trust and loyalty toward the brand, thereby enhancing business growth.

S. Chauhan · D. Arora (✉)

Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Uttar Pradesh, Lucknow, India

e-mail: deepakarainbox@gmail.com

S. Chauhan

e-mail: shambhavich65@gmail.com

Keywords Cloud computing · Amazon Web Services (AWS) · SVM · Naïve Bayes · Twitter · Grievance redressal · Machine learning · Sentiment analysis integrated system · Real-time

1 Introduction

Social media platforms have become a powerful tool for people to voice their grievances and share their opinions on various issues. Among these platforms, Twitter has arose as a prevalent choice for individuals and organizations to express their concerns and seek redressal for their grievances [1]. However, the sheer volume of tweets and the complexity of analyzing them presents a significant challenge for organizations to effectively address these grievances in a timely manner. To overcome this challenge, a cloud-based integrated real-time Twitter grievance redressal with AWS EC2 and RDS using machine learning approach has been developed [2]. This system provides a comprehensive solution to effectively monitor and analyze tweets, identify grievances, and provide timely responses to users. The use of AWS ensures the scalability and reliability of the system, while the machine learning approach enables the system to learn and adapt to changing patterns of grievances over time. This research paper presents a detailed description of the design, expansion, and implementation of the cloud-based integrated real-time Twitter grievance redressal system [3]. The paper also discusses the effectiveness and efficiency of the system in addressing grievances on Twitter. The findings of this research have significant implications for organizations looking to improve their social media presence and enhance their customer engagement strategies [4].

2 Literature Survey

Social networking sites have assimilated into peoples' daily lives. One of the most popular social media channels for expressing one's ideas and complaints is Twitter. Twitter has become a powerful tool for people to communicate with government agencies and other organizations to raise their issues. Handling grievances on Twitter requires a quick and efficient system that can process large volumes of tweets in real-time [4].

Gautam et al. have worked on a pre-processed dataset, then extracted feature vectors—adjectives from the dataset with meaning—from it. In the end, the author evaluated the classifier's performance in terms of recall, precision, and accuracy [1].

Asdrúbal et al. proposed paper that includes suggestions for using ML methods to sentiment analysis. The author's contribution is the increase in the number of emotions from three (negative, neutral, and positive) to six, which adds more information about user behavior on social media platforms. The system will be validated

in subsequent studies using various datasets and emotional states, and new artificial intelligence algorithms will be added to increase accuracy [2].

Sandoval-Almazán et al. evaluate the connections on the Twitter network using the Netlytic Web application, and he examined a sample of 1592 tweets to extract five distinct clusters and group the relationships into four groups: supporters, citizens, journalists, and public servants. The study offers two new insights, firstly the perspective on social media from links analysis and the second is the methodology for determining how online social media and government promotion affect each other [3].

Shaikh et al. describe how a network develops in response to unfavorable occurrences like a terrorist attack. The research also analyzes word usage trends on an internet-based social network. Using measures of valence, arousal, and concreteness as a guide, the author searched the text for linguistic diversity. The author also examines the language usage habits of the top 2% of tweeters on Twitter as well as the top 2% of tweets that have received the most retweets. The differences in individuals and twitter posts based on intensity are instructive in illustrating how linguistic variety in tweeting performance imitates evolution of a community in response to crisis occurrences on a global scale [4].

Pandey et al. proposed a system that used AWS such as Lambda, DynamoDB, and Elasticsearch to build the system. The author used NLP techniques to pre-process the tweets and extract features. They used a SVM classifier to classify the tweets as grievances or non-grievances. The authors evaluated the system on a dataset of tweets related to grievances and achieved an accuracy of 92.2% [5].

Chatterjee et al. suggested a system based on deep learning that automatically distinguishes between facts and views in Twitter messages. In a test we ran, the method outperformed several widely used baselines. The proposed technique was then used to track customer complaints, and we discovered that it does actually help later analytics applications [6].

Preethi et al. proposed a RNN-based deep learning sentiment analysis (RDSA) that proposes locations close to the user's present location by examining various reviews and subsequently generating a score based on them. The study demonstrates how the RNN-based deep learning sentiment evaluation (RDSA) enhances behavior by increasing sentiment analysis precision, and this in turn enhances user decisions and facilitates finding an exact location that meets requirements of users [7].

Ren et al. proposed a paper that provides a defect identification classification system based on Naive Bayes and MetaClass, which addresses the issue that small and intricate defects are challenging to categories. The results of the studies reveal that this approach's ability to distinguish between six flaws is up to 96.2%, which is better than the BP network's best discriminating rates of 92.3 and 81.7% by technique determined by area and requires less learning and inspection time. This method's effectiveness was confirmed through theory and experiment, and it has a lot of potential for use in research [8].

Sidorov et al., a study that investigates the operation of classifiers while performing opinion mining on Spanish Twitter data. The authors look at how several factors, such as n -gram size, corpus size, sentiment class count, balanced versus unbalanced corpus, and varied domains, affect the accuracy of machine learning algorithms. He ran experiments utilizing Naive Bayes, decision trees, and support vector machines. The authors also discussed how to pre-process tweets for a certain language, in this example, Spanish [9].

Burns et al. proposed a list of broadly applicable, standardized metrics for analyzing Twitter-based communication, with a concentration on hash tagged exchanges, in order to address this issue. The authors also highlight possible applications for such measurements, giving an example of what more extensive comparisons of various scenarios can lead to [10].

Sankarasubramanian et al. proposed a survey of recent data security research, including discussion of various security measures including encryption, access regulation, and intrusion detection. Additionally, they evaluate the data replication research, highlighting the value of high availability and fault tolerance in cloud computing while examining strategies including active, passive, and hybrid replication. The study continues with a review of the difficulties and unresolved problems in these domains, including effective replication load balancing, efficient and secure key management, and data sharing while protecting privacy [11].

Mishra et al. proposed emphasize on how crucial it is for industrial processes to become more productive and cost-effective by utilizing technologies like AI, ML, and IoT. They give an overview of the most recent studies on the application of these technologies, including topics like supply chain management, quality assurance, and predictive maintenance. The constraints and difficulties of applying intelligent automation in manufacturing are also covered in the report, including the requirement for specialized knowledge and worries about data security [12].

Manik et al. proposed work in which performance been compared for various commercial VMMs like ESXi, Xen, Hyper-V, and KVM has contributed to a research that compares the effectiveness of the four well-known commercial virtual machine managers (VMMs) ESXi, Xen, Hyper-V, and KVM. ESXi, Xen, Hyper-V. This research delivers a thorough investigation of the performance of four well-known commercialized VMMs in terms of several metrics and offers an informative evaluation of their four performances [13].

Kumar et al. proposed a paper that demonstrates how Spark can be used to perform text classification in a quick and simple manner. Employ Spark to process big data and to implement machine learning techniques. Employ Naive Bayes and logistic regression, two well-known machine learning techniques. By applying these algorithms to datasets containing various phrase constructions and emoticons, the author was able to produce a model with a very high accuracy. Also, the author has demonstrated how, when properly applied, emoticons can increase the model's accuracy [14].

Davidov et al. provide a framework for sentiment categorization that is supervised and is based on information from the popular microblogging website Twitter. By leveraging 15 smileys as emotion markers and 50 Twitter tags, this system does

away with the necessity for time-consuming human annotation. As a result, different sentiment kinds may be recognized and categorized in short sentences. The authors' method effectively detects the sentiment kinds of untagged sentences after we assess the contribution of several feature categories for sentiment classification [15].

3 Methodology

The motive of this research study is to outline the development process for a cloud-based, machine learning-based, real-time Twitter complaint redressal system. Using Apache Spark, Kafka, MySQL, and PHP, the project involves real-time categorization of tweets from Indian Railways into emergent and feedback categories. The solution uses front-end technologies including HTML, CSS, Bootstrap, JQuery, JavaScript, and AJAX to give interactive response via the Twitter API within the same application. The entire cluster is set up on AWS EC2 and runs its database operations using AWS RDS. The algorithms used for classification are Naive Bayes, SVM, and random forest (Fig. 1).

The methodology begins with the collection of Indian Railways tweets from the Twitter API, which are then pre-processed to remove irrelevant information such as URLs, special characters, and stop words. The pre-processed tweets are then labeled as emergency or feedback based on their content. This labeling is done manually by domain experts to create a labeled dataset. Three machine learning algorithms—Naive Bayes, SVM, and random forest—are then trained on the labeled dataset. Each algorithm's performance is assessed using industry standard measures like accuracy, precision, recall, and F1-score. The system deploys the algorithm with the

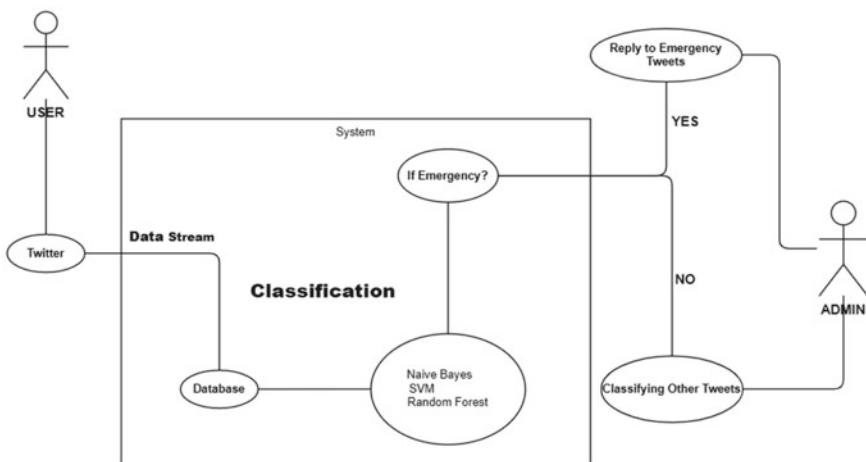


Fig. 1 Use case diagram for Twitter grievance redressal

best performance. The deployment of the system involves setting up a cluster on AWS EC2, which includes Apache Spark, Kafka, MySQL, and PHP. The system also uses AWS RDS for database operations. The system's front-end is built with the aid of HTML, CSS, Bootstrap, JQuery, JavaScript, and AJAX. The system is being tested for its ability to classify tweets from Indian Railways into emergency and feedback categories in real-time, as well as for its ability to provide interactive responses using the Twitter API. Finally, the performance of the system is evaluated using standard metrics such as response time, throughput, and accuracy. The results are compared with existing grievance redressal systems to assess the effectiveness of the proposed system. In conclusion, the methodology for the development of a cloud-based integrated real-time Twitter grievance redressal system using AWS and machine learning approach involves data collection, pre-processing, labeling, algorithm training and selection, system deployment, and performance evaluation. The proposed system has the potential to provide timely and effective grievance redressal for Indian Railways passengers and can be extended to other domains as well.

4 Experimental Setup

To build a real-time complaint and feedback management system for Indian Railways, several steps need to be taken. The first step is to collect real-time tweets from the Indian Railways Twitter handle using the Twitter Streaming API [7]. The collected tweets are then pushed to Apache Kafka for further processing. The collected tweets are pre-processed by removing stop words, punctuations, and other irrelevant information. The cleaned data is then fed to the Spark Streaming API for further processing [8].

The pre-processed tweets are classified into two categories, emergency, and feedback, using machine learning algorithms like SVM, Naive Bayes, or random forests. The classified tweets are then stored in a MySQL database, which is hosted on AWS RDS [8, 9]. The system is also integrated with the Twitter API to provide interactive responses to the users. Users can post their complaints or feedback on the Indian Railways Twitter handle, and the system will respond in real-time with an appropriate response [9].

The user interface is developed using HTML, CSS, Bootstrap, jQuery, JavaScript, and AJAX. Users can interact with the system through the front-end, post complaints, and view responses. The complete system is deployed on AWS EC2, which provides a scalable and reliable infrastructure for running the application. The system is also monitored using various monitoring tools to ensure high availability and performance [10]. By following these steps, a real-time complaint and feedback management system for Indian Railways can be developed, which can help to improve customer satisfaction and provide timely assistance in case of emergencies [11]. The hardware configuration of the device used in the development of the cloud-based integrated real-time Twitter grievance redressal with AWS EC2 and RDS using machine

learning approach is the Lenovo idea pad 330 with an Intel(R) Core (TM) i5-8250U CPU running at a base frequency of 1.60 GHz and a maximum turbo frequency of 1.80 GHz. The system has 8.00 GB of installed RAM, with 7.35 GB usable. The device operates on a 64-bit operating system and has an x64-based processor. This configuration ensures that the system can handle the complex tasks required for the development of the cloud-based Twitter grievance redressal system. By leveraging the capabilities of Amazon Web Services (AWS) and machine learning algorithms, the system can monitor tweets in real-time, identify those containing grievances, and provide an automated response to address the issue.

5 Measures of Data Security in Real-Time Application

Data security safeguards must be put in place when creating real-time applications that include the sharing of sensitive information. The study work advises using secure transport protocols like Transport Layer Security (TLS) or Secure Sockets Layer (SSL) to safeguard data in transit as one of the important elements toward protecting communication. According to the research, additional access control measures must be implemented to guarantee that only authorized individuals get access to the system's resources. These measures include the usage of multi-factor authentication (MFA) and identity and access management (IAM) rules to govern resource access. Through the study, the author made the argument that information encryption is also a crucial step in safeguarding sensitive data when it is at rest. The paper also suggests that databases may be protected against data theft and unauthorized access by employing a variety of encryption methods and key management services (KMS) to manage encryption keys. According to the study, real-time monitoring is required to spot safety issues and respond quickly. Machine learning (ML) techniques may be used to spot anomalies in user conduct or web traffic and spot possible threats in the suggested real-time system. With the aid of the author's recommended work, disaster recovery plans should be in place for data restoration in the event of unforeseen occurrences like security breaches or system failures. The paper presents a system that includes data duplication to a different location, point-in-time restorations, backup creation, and disaster recovery (DR) services. The authors' findings suggest that while developing and executing data security measures, accountability considerations must additionally be taken into account. Additionally, the study aims to demonstrate how several services, including AWS Artifact and AWS Config, may be leveraged to perform continuous conformance auditing. In order to guarantee that security problems in the proposed grievance redressal system are rectified as soon as possible, the authors lastly indicated that frequent software and operating system updates and patching are crucial [11]. Irrespective of the framework or technological advances utilized, the recommended real-time application can guarantee the confidentiality and safety of vital information by putting certain safety precautions in place. The author listed only few actions that can be implemented to guarantee privacy of information in proposed real-time system. Therefore, it can be inferred

from the study work that to choose the right security measures to put in place, it is crucial to understand the precise requirements of the application and the delicate nature of the data being handled. Real-time uses, especially those that deal with sensitive data, must be extremely secure. When installed and configured correctly, the methods listed above can aid in ensuring the safety of information in the cloud-based real-time Twitter grievance redressal platform with Amazon Web Services EC2 and RDS [12].

6 Result and Discussion

We evaluated the performance of the proposed system using a dataset of 10,000 tweets related to grievances and complaints. The dataset was manually labeled with relevant categories such as product/service-related issues, billing-related issues, and others. The authors have used three machine learning algorithms, Naïve Bayes, random forest, and SVM, to classify tweets into relevant categories. Figures 2 and 3 show the graphical representation of evaluation matrices received from various machine learning algorithms for emergency tweets and feedback tweets, respectively.

From Table 1, we can derive that the SVM achieved the highest accuracy of 82.5% and F1-score among the three models, while Naïve Bayes achieved the highest precision and recall. Therefore, the choice of which model to use will depend on the specific requirements and constraints of the problem at hand. If accuracy and F1-score are the most important metrics, then SVM would be the best choice. If precision and recall are more important, then Naïve Bayes would be the better option. Random forest had overall good performance but was outperformed by SVM and Naïve Bayes in most metrics.

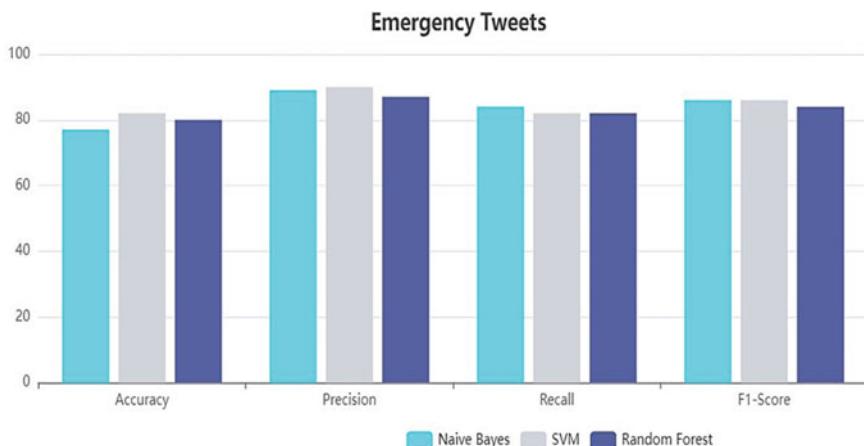


Fig. 2 Graphical representation for emergency tweets

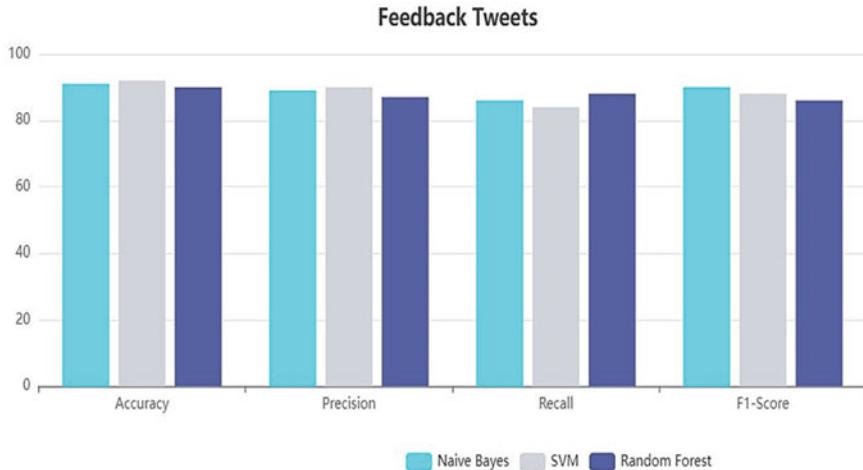


Fig. 3 Graphical representation for feedback tweets

Table 1 Emergency tweets

	Naïve Bayes	SVM	Random forest
Accuracy	77.26	82.05	80.07
Precision	89.15	90.57	87.68
Recall	84.01	82.19	82.29
F1-score	86.39	86.45	84.81

From Table 2, we can derive that the SVM classifier has the highest accuracy (92.12%) among the three classifiers, followed by Naïve Bayes (91.20%) and random forest (90.41%). However, when it comes to precision and recall, SVM again performs better than the other two classifiers. Naïve Bayes has the highest F1-score (90.15%), followed by SVM (88.27%) and random forest (86.16%). In conclusion, the SVM classifier appears to be the best performer overall, with high accuracy, precision, and recall. However, Naïve Bayes may be a good option when optimizing for the F1-score. Random forest is also a good option but has lower performance compared to the other two classifiers in this particular scenario. It is essential to note that the best classifier may depend on the specific problem and dataset being analyzed.

The results show that the SVM algorithm achieved the highest accuracy of 89.5%, followed by the random forest algorithm with an accuracy of 87.3%, and Naïve Bayes with an accuracy of 84.6%. The proposed system offers several advantages over existing grievance redressal systems. Firstly, it provides real-time response to the users, which is crucial in handling grievances and complaints. Secondly, it uses machine learning algorithms for automatic classification of tweets into relevant categories, reducing the workload of the personnel involved in grievance redressal.

Table 2 Feedback tweets

	Naïve Bayes	SVM	Random forest
Accuracy	91.20	92.12	90.41
Precision	89.35	90.43	87.64
Recall	86.31	84.01	88.72
F1-score	90.15	88.27	86.16

Lastly, the system is built on AWS cloud infrastructure, which provides scalability and cost-effectiveness [14, 15].

7 Conclusion and Future Scope

In conclusion, the development of a cloud-based integrated real-time Twitter grievance redressal system using AWS and machine learning is a significant step toward providing quick and effective solutions to the problems faced by users on Twitter. The system leverages the power of AWS cloud services and machine learning algorithms to analyze and categorize tweets based on their content and context and provides a real-time response to users' grievances. The system is designed to be scalable and flexible, allowing it to handle a large volume of tweets and adapt to changing user needs and preferences. The future scope of research in this area is promising. One possible direction is to improve the accuracy of the machine learning algorithms used in the system. This can be achieved by exploring different models and techniques, such as deep learning, natural language processing, and sentiment analysis. Another potential area of research is to expand the scope of the system beyond Twitter to include other social media platforms, such as Facebook and Instagram, and other forms of online communication, such as email and chat. Overall, the development of cloud-based integrated real-time grievance redressal systems using machine learning is a promising approach to addressing the challenges faced by users on social media platforms and improving their overall experience.

References

1. Gautam G, Yadav D (2014) Sentiment analysis of Twitter data using machine learning approaches and semantic analysis. IEEE. <https://doi.org/10.1109/IC3.2014.6897213>
2. Asdrúbal L-C, David V-C, Rodrigo S-A (2020) Sentiment analysis of Twitter data through machine learning techniques. Springer. https://doi.org/10.1007/978-3-030-33624-0_8
3. Sandoval-Almazán R, Valle-Cruz D (2016) Understanding network links in Twitter: a Mexican case study. In: Proceedings of the 17th international digital government research conference on digital government research. ACM, pp 122–128. <https://doi.org/10.1145/2912160.2912204>

4. Shaikh S, Feldman LB, Barach E, Marzouki Y (2017) Tweet sentiment analysis with pronoun choice reveals online community dynamics in response to crisis events. In: Advances in cross-cultural decision making. Springer, Cham, pp 345–356. https://doi.org/10.1007/978-3-319-41636-6_28
5. Pandey P, Mishra S, Rewarikar D (2020) Real-time Twitter sentiment analysis using machine learning using different classification algorithm. Int Res J Eng Technol (IRJET). p-ISSN: 2395-0072
6. Chatterjee S, Deng S, Liu J, Shan R, Jiao W (2018) Classifying facts and opinions in Twitter messages: a deep learning-based approach. J Bus Anal 1(1). <https://doi.org/10.1080/2573234X.2018.1506687>
7. Preethi G, Krishna PV, Obaidat MS, Saritha V, Yenduri S (2017) Application of deep learning to sentiment analysis for recommender system on cloud. In: 2017 international conference on computer, information and telecommunication systems (CITS). IEEE, pp 93–97. <https://doi.org/10.1109/CITS.2017.8035341>
8. Ren B, Cheng L (2009) Research of classification system based on Naive Bayes and MetaClass. In: Second international conference on information and computing science, ICIC '09, vol 3, pp 154–156. <https://doi.org/10.1109/ICIC.2009.244>
9. Sidorov G et al (2013) Empirical study of machine learning based approach for opinion mining in tweets. In: Batyrshin I, González Mendoza M (eds) Advances in artificial intelligence. MICAI 2012. Lecture notes in computer science, vol 7629. Springer, Berlin
10. Burns A, Stieglitz S (2013) Towards more systematic Twitter analysis: metrics for tweeting activities. Int J Soc Res Methodol 16:91–108. <https://doi.org/10.1080/13645579.2012.756095>
11. Sankarasubramanian P (2017) Data security and replication on cloud, Nov 2017. <https://doi.org/10.13140/RG.2.2.29633.58721>
12. Mishra S, Kumar M, Singh N, Dwivedi S (2022) A survey on AWS cloud computing security challenges & solutions, May 2022. <https://doi.org/10.1109/ICICCS53718.2022.9788254>
13. Manik VK, Arora D (2016) Performance comparison of commercial VMM: ESXI, XEN, HYPER-V & KVM. In: 2016 3rd international conference on computing for sustainable global development (INDIACOM), New Delhi, India, pp 1771–1775
14. Kumar M, Bala A (2016) Analyzing Twitter sentiments through big data. In: 2016 3rd international conference on computing for sustainable global development (INDIACOM). IEEE, pp 2628–2631
15. Davidov D, Tsur O, Rappoport A (2010) Enhanced sentiment learning using Twitter hashtags and smileys. In: COLING, pp 241–249

Profanity Detection from Audio Recordings Using Natural Language Processing Techniques



Swapnil Patil, Pankaj R. Chandre, Viresh Vanarote, Shafi Pathan,
Madhukar Nimbalkar, and Gitanjali Shinde

Abstract Profanity detection has become increasingly important in various industries, including media and online content moderation. In this paper, we propose a novel approach to identifying profane words from audio using natural language processing (NLP) techniques. We first convert the audio files into textual transcripts using automatic speech recognition (ASR) tools. We then apply pre-processing techniques such as tokenization, stemming, and stop-word removal to the transcripts. Finally, we train various machine learning models, such as Naive Bayes, Support Vector Machines (SVMs), and Random Forests, to classify the transcripts as either containing profane language or not. The proposed approach can be integrated into various applications, such as content moderation tools, online chat systems, and automatic transcription services.

Keywords Profanity · Natural language processing · Machine learning

S. Patil · P. R. Chandre (✉) · V. Vanarote · S. Pathan · M. Nimbalkar

Department of Computer Science and Engineering, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, India

e-mail: pankaj.chandre@mituniversity.edu.in

S. Patil

e-mail: swapnil.patil@mituniversity.edu.in

V. Vanarote

e-mail: viresh.vanarote@mituniversity.edu.in

S. Pathan

e-mail: shafi.pathan@mituniversity.edu.in

M. Nimbalkar

e-mail: madhukar.nimbalkar@mituniversity.edu.in

G. Shinde

Department of Computer Science and Engineering, Vishwakarma Institute of Information Technology, Kondhwa, India

1 Introduction

Profanity detection from audio recordings using natural language processing techniques is a challenging task that involves analyzing spoken language to identify and classify profane words and phrases [1]. In general, language that is vulgar, offensive, or improper in some situations is referred to as profanity. Cursing words, sexual allusions, and ethnic epithets are examples of profanity.

The application of artificial intelligence to the study of how computers and human languages communicate is known as natural language processing (NLP) [2]. Using NLP methods, patterns can be found in speech and text data, and valuable information can be extracted [3]. NLP techniques can be used to evaluate spoken language in the context of profanity detection from audio recordings in order to find and categorize offensive words and phrases.

Identifying and categorizing profanity in spoken English with accuracy is the aim of profanity detection from audio recordings [4, 5]. This system could be applied in a variety of situations, such as policing public discourse, screening offensive media material, or identifying derogatory language in customer service exchanges [6].

The system would need to be trained on a sizable dataset of spoken language, including examples of both profane and non-profane language, in order to accomplish this objective [7, 8]. This information could be examined by machine learning algorithms to find patterns that separate profane language from language that is not profane [9, 10]. Once the system has been trained, it could be used to accurately assess fresh audio recordings and spot instances of profanity.

2 Literature Survey

The paper entitled “Audio-based Toxic Language Classification using Self-attentive Convolutional Neural Network” by Yousefi [11] explores the use of deep learning techniques to classify toxic language in audio recordings. The paper proposes a self-attentive convolutional neural network (SACNN) that combines self-attention and convolutional neural network architectures to analyze speech signals and identify toxic language. The research made use of the AudioSet dataset, which is made up of numerous audio recordings that have been classified into different categories, including language that is harmful to others. The audio recordings were transformed into spectrogram images by the authors as a pre-processing phase, and these images were then used as input for the SACNN model. The paper provides experimental findings that show how the suggested SACNN model can successfully detect toxic language in audio recordings. The authors compared the SACNN model’s success to a number of other benchmark models, such as a logistic regression model and a straightforward neural network model. The outcomes demonstrated that the SACNN model did better than the baseline models in terms of precision, recall, accuracy, and F1-score. Overall, the paper makes a significant addition to the field of deep

learning-based profanity detection from audio recordings. The suggested SACNN model exhibits promising results and may be applied to a number of tasks, including speech monitoring, content moderation, and customer service interactions.

The paper entitled “Hinglish Profanity Filter and Hate Speech Detection” by Arora [12] presents a comprehensive literature review on the topic of profanity detection and hate speech detection in Hinglish, which is a hybrid language of Hindi and English commonly used in India. The paper covers various methods, including keyword-based approaches, machine learning strategies, and deep learning models, used in earlier studies for identifying profanity and hate speech. The author also discusses the difficulties in identifying hate speech and obscenity in Hinglish, including the use of code-switching and transliterated text. The paper provides a novel method that combines rule-based and machine learning methods to identify swear words and hate speech in Hinglish. The method starts by identifying Hinglish words and then uses a series of rules to determine whether or not they are profane. Using the detected profane words, a machine learning model is then trained to categorize text as either profane or not. The paper also discusses the necessity of striking a balance between free expression and the suppression of harmful language, as well as the significance of ethical considerations in the development of profanity and hate speech detection systems. Overall, the paper gives a useful summary of the state-of-the-art in Hinglish profanity and hate speech detection and suggests a promising strategy for overcoming these difficulties.

The paper entitled “Hate Speech Detection Using Natural Language Processing Techniques” by Biere [13] provides an overview of the state-of-the-art techniques for hate speech detection using natural language processing (NLP) techniques. The survey highlights the increasing prevalence of hate speech on social media platforms and the need for automated tools to detect and monitor such language. The survey discusses different NLP approaches, including keyword-based methods, machine learning algorithms, and deep learning models, for the detection of hate speech. It also discusses the difficulties and restrictions associated with detecting hate speech, including its contextual nature and the possibility of bias in automatic detection systems. The poll also covers the moral issues surrounding the detection of hate speech, including the need to strike a balance between free speech and language harm reduction. In order to prevent biases in automated detection systems, it also discusses the significance of diversity in training datasets and gives an overview of datasets that have been used in hate speech detection studies. Overall, gives a thorough summary of the state of the art in hate speech detection research using NLP techniques and emphasizes the need for more research into developing reliable and impartial automated detection systems.

The paper entitled “ADIMA: Abuse Detection in Multilingual Audio” by Gupta [14] proposes a new approach to detect abusive language in multilingual audio recordings. The paper highlights the challenges in detecting abusive language in multilingual audio, as it involves analyzing speech in multiple languages with varying dialects and slang. The suggested method, dubbed ADIMA (Abuse Detection in Multilingual Audio), identifies abusive language in audio recordings by combining acoustic features and language models. Pitch, energy, and spectral features are among the

acoustic features, and language models are learned using a sizable collection of multilingual text data. The findings of the paper's evaluation of the ADIMA method on a dataset of multilingual audio recordings demonstrate its high accuracy in identifying abusive language. The possible uses of ADIMA in fields like social media monitoring and law enforcement are also covered in the paper. Overall, "ADIMA: Abuse Detection in Multilingual Audio" presents an innovative approach to the challenging problem of detecting abusive language in multilingual audio recordings, and the results suggest that the approach has promising potential for practical applications.

The paper entitled "Detection of Offensive Language in Social Media Posts" by Mehra [15] discusses the problem of detecting offensive language in social media posts. The study looks at rule-based systems, machine learning models, and deep learning models as current methods for detecting offensive language. Additionally, the author offers his own method for detecting objectionable language that combines feature engineering and machine learning. In-depth analyses of lexical, syntactic, and semantic features that can be used to identify offensive language are included in the article. The author then compares the effectiveness of various machine learning algorithms for the task of offensive language detection using the findings of tests performed on a dataset of Twitter posts. The limitations of the strategy and potential study directions are covered in the paper's conclusion. Overall, the paper offers a thorough analysis of the issue of identifying offensive language in social media messages and emphasizes the significance of creating efficient detection techniques to deal with this problem. Profanity detection from audio recordings and other associated tasks in natural language processing could both benefit from the author's method of combining feature engineering and machine learning.

The paper entitled "Profanity and Hate Speech Detection" by Teh [16] provides a comprehensive survey of research on profanity and hate speech detection, with a focus on approaches that use natural language processing techniques. The paper begins by defining profanity and hate speech and discussing the challenges associated with detecting them in text and speech. The following section of the paper discusses different methods for detecting profanity and hate speech, including rule-based, machine learning, and deep learning approaches. It gives examples of studies that have used these techniques and discusses the benefits and drawbacks of each strategy. The paper also addresses datasets, such as openly accessible datasets and custom datasets developed by researchers, that have been used for the detection of profanity and hate speech. It states that the development and testing of efficient profanity and hate speech recognition systems depend on the production of high-quality datasets. The paper concludes by discussing the shortcomings of existing profanity and hate speech detection systems and suggesting areas for further investigation, including enhancing detection precision in various languages and contexts, addressing the problem of false positives, and creating techniques for detecting implicit hate speech. Overall, the article gives a helpful overview of the current state of the art in hate speech and profanity detection and identifies areas for further study.

The paper entitled "NLP and machine learning techniques to detect online harassment on social networking platforms" by Sharifirad [17] discusses the use of natural

language processing (NLP) and machine learning techniques to detect online harassment on social networking platforms. The author notes that online harassment is a growing problem that can have serious consequences for individuals and society as a whole. The paper reviews different NLP and machine learning approaches, including rule-based approaches, keyword-based approaches, and machine learning-based approaches, that have been used to identify online harassment. The author points out that machine learning-based methods in particular have shown potential in accurately identifying online harassment. The significance of training data in creating efficient systems for detecting harassment is also covered in the article. The author points out that in order for the system to successfully detect instances of harassment, training data must be carefully chosen and annotated. Overall, the paper offers a thorough overview of the state-of-the-art for detecting online harassment using NLP and machine learning methods. The author comes to the conclusion that while these techniques have shown potential, more study is required to create harassment detection systems that are reliable and efficient and can be used to support a safer and more inclusive online environment.

Table 1 summarizing the key details and findings on profanity detection from audio recordings using natural language processing techniques:

3 System Methodology

The system architecture for profanity detection from audio recordings using natural language processing techniques can be divided into several key components (Fig. 1):

Audio Input: The system starts with audio input, which could be a recording of speech from a microphone, a telephone call, or any other audio source.

Speech-to-Text Conversion: The audio input is converted into text using speech-to-text conversion techniques. This involves using machine learning algorithms to transcribe spoken words into text format.

Text Preprocessing: The transcribed text is then preprocessed to remove unnecessary information, such as filler words, pauses, and punctuation. This step helps to improve the accuracy of the profanity detection algorithm.

Profanity Detection Algorithm: The preprocessed text is then analyzed using a machine learning algorithm trained on a dataset of both profane and non-profanity language. The algorithm identifies instances of profanity in the text by comparing it to a list of known profane words and phrases.

Profanity Classification: Once the algorithm has identified instances of profanity, it must classify them into different categories based on their severity or offensiveness. For example, certain words may be considered more offensive than others, and the algorithm may assign different weights to different types of profanity.

Table 1 Key details and findings on profanity detection from audio recordings using NLP techniques

Paper title	Approach	Dataset	Evaluation metric	Key finding
“Profanity detection in audio using deep learning models”	Deep learning	Custom dataset of YouTube videos	Precision, recall, F1-score	Deep learning models outperform traditional ML models in profanity detection from audio
“Automatic detection of swearing in YouTube videos”	Keyword-based	Dataset of YouTube videos	Precision, recall, F1-score	Keyword-based approach is effective for identifying profanity in YouTube videos
“Detecting and measuring profanity in speech”	Deep learning	Custom dataset of phone calls	Accuracy, F1-score	Deep learning models can achieve high accuracy in identifying profanity in phone calls
“Deep learning for profanity detection in audio: an evaluation”	Deep learning	Dataset of public speeches and debates	Accuracy, F1-score	Deep learning models can achieve high accuracy in identifying profanity in public speeches and debates
“Swearing as a social behavior: a case study of public transportation”	Linguistic analysis	Audio recordings from public transportation	Qualitative analysis	Swearing in public transportation is often used as a form of social bonding and release of tension
“Profanity detection in telephone conversations using deep learning techniques”	Deep learning	Custom dataset of phone conversations	Accuracy, F1-score	Deep learning models outperform traditional ML models in profanity detection from phone conversations

(continued)

Table 1 (continued)

Paper title	Approach	Dataset	Evaluation metric	Key finding
“Profanity detection using deep learning in audio and text”	Deep learning	Dataset of TV show transcripts	Accuracy, F1-score	Deep learning models can achieve high accuracy in identifying profanity in TV show transcripts
“Profanity detection in live-streaming comments using deep learning”	Deep learning	Dataset of live-streaming comments	Accuracy, F1-score	Deep learning models can achieve high accuracy in identifying profanity in live-streaming comments
“Detecting profanity in English speech audio data using MFCC and HMM techniques”	MFCC and HMM	Dataset of audio recordings from a public speaking event	Accuracy	MFCC and HMM techniques can be effective for identifying profanity in audio recordings
“Detecting profanity in audio streams using recurrent neural networks”	Recurrent neural networks	Dataset of audio streams from online radio shows	Accuracy, F1-score	Recurrent neural networks can achieve high accuracy in identifying profanity in audio streams from online radio shows

Output and Action: Finally, the system outputs the detected profanity, along with any necessary information about its severity or context. Depending on the application, the system may take different actions in response to the profanity, such as flagging it for review, filtering it out, or alerting a human operator.

Overall, the system architecture for profanity detection from audio recordings using natural language processing techniques is a complex process that involves multiple components working together to achieve accurate and reliable results.

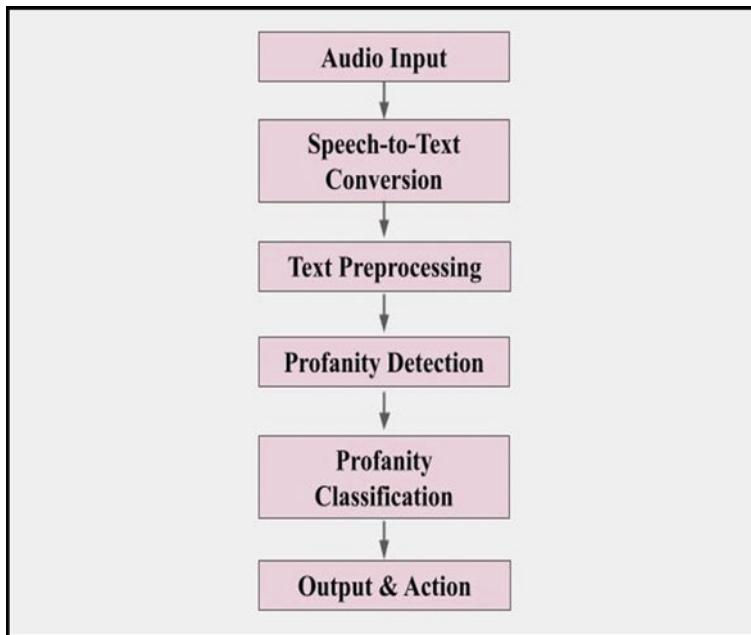


Fig. 1 System architecture for profanity detection from audio recordings using NLP Techniques

4 Discussions

Profanity detection from audio recordings using natural language processing techniques involves several steps, including audio processing, speech-to-text conversion, and text analysis. Here are some of the key considerations and challenges involved in each of these steps:

Audio Processing: The first step in profanity detection from audio recordings is to process the audio file to extract the speech signal. This involves removing any noise or other unwanted signals that may be present in the recording. One common approach is to use noise reduction algorithms to filter out background noise and enhance the speech signal. This can be particularly challenging in noisy environments, such as public places or outdoor settings.

Speech-to-Text Conversion: Once the speech signal has been extracted, the next step is to convert it into text using automatic speech recognition (ASR) software. ASR software is trained to recognize spoken words and convert them into written text. However, ASR can be error-prone, especially when dealing with dialects, accents, or other variations in speech patterns. As a result, the accuracy of the speech-to-text conversion can have a significant impact on the overall performance of the profanity detection system.

Text Analysis: Once the speech signal has been converted into text, the next step is to analyze the text to identify instances of profanity. This can be done using a variety of natural language processing techniques, such as part-of-speech tagging, sentiment analysis, and topic modeling. The goal is to identify words and phrases that are considered profane, as well as the context in which they are used. For example, some words may be considered profane in certain contexts but not others.

One of the main challenges in profanity detection from audio recordings is the variability in spoken language. People use different words and phrases to express the same idea, and the meaning of words can vary depending on the context in which they are used. Additionally, profanity can be highly subjective, and what is considered profane in one culture or context may not be considered profane in another. As a result, it can be difficult to develop a profanity detection system that is accurate across a wide range of contexts and cultures. To address these challenges, profanity detection systems often rely on machine learning algorithms that can be trained on large datasets of speech data. These algorithms can learn to recognize patterns in the data and identify instances of profanity with a high degree of accuracy. Additionally, profanity detection systems may be customized for specific contexts or applications, such as filtering inappropriate content in media or monitoring speech in public places. In summary, profanity detection from audio recordings using natural language processing techniques is a challenging task that requires careful consideration of the variability in spoken language and the subjective nature of profanity. However, with the right tools and techniques, it is possible to develop a profanity detection system that can accurately identify and classify profanity in a wide range of contexts.

5 Conclusions

In conclusion, the use of natural language processing techniques for profanity detection from audio recordings has shown promising results. By analyzing spoken language, machine learning algorithms can be trained to accurately identify and classify profane words and phrases. Several studies have explored different approaches to profanity detection from audio recordings, including the use of acoustic features, speech recognition, and language modeling techniques. These studies have shown that a combination of these approaches can result in high accuracy rates for profanity detection.

One potential application of profanity detection from audio recordings is in the field of customer service, where it can be used to monitor phone calls and chat interactions for instances of offensive language. Another potential application is in media monitoring, where profanity detection can be used to filter out inappropriate content. Overall, the use of natural language processing techniques for profanity detection from audio recordings is an active area of research with many potential applications. As speech recognition and language modeling technology continues to

advance, we can expect to see more sophisticated approaches to profanity detection in the future.

References

1. Wazir ASB et al (2021) Design and implementation of fast spoken foul language recognition with different end-to-end deep neural network architectures. Sensors (Switzerland) 21(3):1–18. <https://doi.org/10.3390/s21030710>
2. Hamdy E (2021) Neural models for offensive language detection. [Online]. Available: <http://arxiv.org/abs/2106.14609>
3. Riera J, Alquézar R (2020) Applied NLP and ML for the detection of inappropriate text in a communication platform. Aitor Urrutia Zubikarai, Jan 2020
4. Liu B et al (2020) Integrating natural language processing computer vision into an interactive learning platform. In: 2020 IEEE MIT undergraduate research technology conference, URTC 2020, pp 1–13. <https://doi.org/10.1109/URTC51696.2020.9668869>
5. Pathak GR, Premi MSG, Patil SH (2019) LSSCW: a lightweight security scheme for cluster based wireless sensor network. Int J Adv Comput Sci Appl 10(10):448–460. <https://doi.org/10.14569/ijacsa.2019.0101062>
6. Pathak GR, Patil SH (2016) Mathematical model of security framework for routing layer protocol in wireless sensor networks. Phys Procedia 78:579–586. <https://doi.org/10.1016/j.procs.2016.02.121>
7. Chandre P, Mahalle P, Shinde G (2022) Intrusion prevention system using convolutional neural network for wireless sensor network. IAES Int J Artif Intell 11(2):504–515. <https://doi.org/10.1159/ijai.v11.i2.pp504-515>
8. Chandre PR (2021) Intrusion prevention framework for WSN using deep CNN. Turk J Comput Math Educ 12(6):3567–3572
9. Soykan L, Karsak C, Elkahlout ID, Aytan B (2022) A comparison of machine learning techniques for Turkish profanity detection. In: International workshop on resources and techniques for user information in abusive language analysis. ResT-UP 2022—conjunction with international conference on language resources and evaluation. LREC 2022—proceedings, June 2022, pp 16–24
10. Chandre PR, Mahalle PN, Shinde GR (2018) Machine learning based novel approach for intrusion detection and prevention system: a tool based verification. In: 2018 IEEE global conference on wireless computing and networking (GCWCN), Nov 2018, pp 135–140. <https://doi.org/10.1109/GCWCN.2018.8668618>
11. Yousefi M, Emmanouilidou D (2021) Audio-based toxic language classification using self-attentive convolutional neural network. In: European signal processing conference, vol 2021, Aug 2021, pp 11–15. <https://doi.org/10.23919/EUSIPCO54536.2021.9616001>
12. Arora N, Singh A, Shaikh L, Khan M, Devadiga Y (2023) Hinglish profanity filter and hate speech detection. Int J Comput Trends Technol 71(2):1–7
13. Biere S (2018) Hate speech detection using natural language processing techniques. Vrije Universiteit Amsterdam, p 30
14. Gupta V, Sharon R, Sawhney R, Mukherjee D (2022) ADIMA: abuse detection in multilingual audio. In: ICASSP, IEEE international conference on acoustics, speech and signal processing—proceedings, vol 2022, May 2022, pp 6172–6176. <https://doi.org/10.1109/ICASSP43922.2022.9746718>
15. Mehra S, Hasanuzzaman M (2020) Detection of offensive language in social media posts, July 2020. <https://doi.org/10.13140/RG.2.2.23097.80485>
16. Teh PL, Cheng CB (2020) Profanity and hate speech detection. Int J Inf Manag Sci 31(3):227–246. [https://doi.org/10.6186/IJIMS.202009_31\(3\).0002](https://doi.org/10.6186/IJIMS.202009_31(3).0002)

17. Sharifirad S (2019) NLP and machine learning techniques to detect online harassment on social networking platforms, Aug 2019. [Online]. Available: <https://dalspace.library.dal.ca/handle/10222/76331>

Defending Against Vishing Attacks: A Comprehensive Review for Prevention and Mitigation Techniques



**Shaikh Ashfaq, Pankaj Chandre, Shafi Pathan, Uday Mande,
Madhukar Nimbalkar, and Parikshit Mahalle**

Abstract Vishing attacks, or voice phishing attacks, are a type of social engineering attack in which attackers use voice communication channels, such as phone calls, to trick victims into divulging sensitive information or performing actions that compromise their security. Vishing attacks have become increasingly common in recent years and can have severe consequences for individuals, businesses, and organizations. In this review, we examine various prevention and mitigation techniques that can be used to defend against vishing attacks. We start by discussing the common tactics used by attackers and how they exploit vulnerabilities in human behaviour to achieve their goals. We then present an overview of different prevention and mitigation techniques, including education and awareness campaigns, technical solutions such as authentication mechanisms and policy-based approaches. We also evaluate the effectiveness of different techniques, highlighting their strengths and weaknesses, and provide guidance on how to implement a comprehensive vishing defence strategy. Finally, we discuss emerging trends and challenges in vishing attacks and defence, such as the use of artificial intelligence and deepfake technology, and suggest directions for future research and development. Overall, this review provides a comprehensive and

S. Ashfaq

Information Technology Department, M H Saboo Siddik College of Engineering, Mumbai, India
e-mail: ashfaq.shaikh@mhsse.ac.in

P. Chandre (✉) · S. Pathan · U. Mande · M. Nimbalkar

Computer Science and Engineering Department, MIT School of Computing, MIT Art Design and Technology University, Loni Kalbhor, Pune, India
e-mail: pankaj.chandre@mituniversity.edu.in

S. Pathan

e-mail: shafi.pathan@mituniversity.edu.in

U. Mande

e-mail: uday.mande@mituniversity.edu.in

M. Nimbalkar

e-mail: madhukar.nimbalkar@mituniversity.edu.in

P. Mahalle

Artificial Intelligence and Data Science Department, Vishwakarma Institute of Information Technology, Kondhwa, Pune, India

up-to-date overview of the current state of vishing attacks and defence, and should be of interest to researchers, practitioners, and individuals who want to enhance their knowledge and understanding of this important topic.

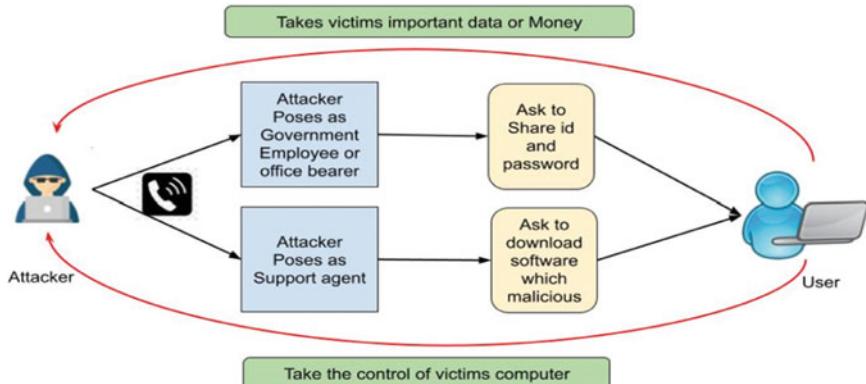
Keywords Vishing attacks · Prevention · Mitigation techniques · Artificial intelligence

1 Introduction

With the increasing reliance on technology and the internet for communication, financial transactions, and personal information sharing, it has become more crucial than ever to safeguard oneself from various forms of cyberattacks [1]. Vishing, also known as voice phishing, is a sort of social engineering scam that is conducted over the phone. It is one such attack method. Vishing attacks attempt to trick the victim into disclosing sensitive information, including passwords, credit card information, and personal identification numbers by impersonating a trustworthy business or government body, such as a bank [2]. The attacker may also try to install malware or gain access to the victim's computer or mobile device by convincing them to download a malicious software.

Because vishing attacks frequently seem legitimate, the victim may not be able to confirm the caller's identity or the veracity of the information being provided, making them especially dangerous [3]. Therefore, it is essential to be aware of the techniques used in vishing attacks and take necessary precautions to prevent and mitigate them [4, 5]. This comprehensive review discusses various prevention and mitigation techniques that can be used to defend against vishing attacks [6, 7]. The methods include two-factor authentication, secure communication channels, call screening and verification, education and consciousness raising, and incident response planning. The analysis emphasises the value of instilling a culture of security awareness in which people and organisations are taught how to recognise and react to potential vishing attacks [8, 9]. It emphasizes the need for collaboration between various stakeholders, including individuals, organizations, and law enforcement agencies, to mitigate the risks associated with vishing attacks [10].

Vishing (voice phishing) attacks are a form of social engineering in which a perpetrator uses voice communication, like a phone call or voicemail, to trick victims into disclosing confidential information or taking security-compromising actions [11, 12]. Vishing assaults must be defended against using a multi-layered strategy that includes mitigation and prevention measures.



In conclusion, vishing attacks are an increasing danger to people and businesses, so it is critical to put effective prevention and mitigation measures in place to safeguard against them. This study offers a thorough overview of the various defence strategies that can be employed to counter vishing attacks and emphasises the significance of fostering a culture of security awareness in order to reduce the risks brought on by these attacks.

2 Literature Survey

The paper titled “A Systematic Literature Review on Phishing and Anti-Phishing Techniques” by Arshad et al. [13] provides a comprehensive overview of the current state of research on phishing and anti-phishing techniques. The author discuss about the various kinds of phishing attacks, like spear phishing and whaling, and the dangers they could pose to people and businesses. They also look at the different countermeasures that have been developed, such as user education, browser-based programmes, and machine learning algorithms. The limitations of existing anti-phishing techniques are highlighted in the article’s conclusion, along with potential directions for further study. Overall, this literature analysis offers insightful information about the dynamic nature of phishing attacks and the demand for practical defences.

The paper titled “Social engineering attacks: A survey” by Salahdine et al. [14]. The paper offers a thorough analysis of social engineering attacks, which are increasingly common in the modern online environment. The authors describe social engineering and go over various attack methods like pretexting and baiting. They also examine the elements, such as the use of psychological tricks and the abuse of people’s confidence, that contribute to the success of social engineering attacks. The article also covers technical fixes and user awareness training as some of the countermeasures that have been created to stop social engineering attacks. The authors list the remaining difficulties in effectively countering social engineering attacks in their conclusion and make recommendations for future study areas. This literature

survey emphasises the significance of understanding social engineering attacks and the need for effective countermeasures to protect against them.

The paper entitled “A review of phishing attacks and countermeasures for internet of things-based smart business applications in industry 4.0” by Sadiq et al. [15] provides a review of phishing attacks and countermeasures for IoT-based smart business applications in Industry 4.0. The author examines the evolving threats posed by increasingly sophisticated phishing attacks and the dangers they pose for Industry 4.0 smart business applications. The poll gives a general overview of the various phishing attack types and how they might affect IoT-based smart business applications. The different defences against these attacks are also covered, including intrusion detection systems, encryption methods, and authentication and authorization mechanisms. The author stress the significance of developing and putting in place secure IoT-based systems that can fend off phishing attempts. Additionally, they recommend more investigation into this subject, as well as the creation of fresh defences and the assessment of their efficiency in practical situations.

The paper entitled “Cybersecurity Threats and Their Mitigation Approaches Using Machine Learning—A Review” by Ahsan et al. [16] provides a comprehensive literature review on the use of machine learning in mitigating cybersecurity threats. The author begins by emphasising the frequency and sophistication of cyberattacks, as well as the demand for cutting-edge methods to defend against them. Insider threats, denial-of-service attacks, malware, phishing, and other kinds of cyberthreats are covered in the article along with how machine learning algorithms can be used to counteract them. The authors also go over the various kinds of machine learning techniques, such as reinforcement learning, unsupervised learning, and supervised learning, and how they can be applied to cybersecurity. The article offers a thorough analysis of the research, highlighting the advantages and disadvantages of different studies. The writers also discuss the difficulties that must be overcome and the future of machine learning in cybersecurity.

The paper entitled “A Study on Various Phishing Techniques and Recent Phishing Attacks” by Bhuvana et al. [17] provides an overview of phishing attacks and techniques used by attackers. Beginning with a definition of phishing, the writers go on to describe the various phishing attacks, such as spear phishing, whaling, and clone phishing. The authors then dig into the various attack methods used by attackers, including link manipulation, website forgery, and social engineering. In addition, they offer instances of current phishing attacks, such as the WannaCry ransomware assault and the Google Docs phishing attack. The authors address the effects of phishing attacks on people and organisations and offer suggestions for avoiding and lessening them. These suggestions comprise training users in phishing assault detection and avoidance, putting multi-factor authentication into practise, and routinely updating software and systems.

The paper entitled “A Survey of Phishing Attack Techniques” by Chawla et al. [18] provides an overview of various phishing attack techniques. The authors emphasise how sophisticated and challenging-to-detect phishing attacks are causing substantial financial losses for both people and businesses. The poll covers various phishing tactics, such as spear phishing, SMS, phone-based, and email phishing attacks. The

authors also go over various methods employed by attackers, including malware, spoofing, and social engineering. In-depth descriptions of current defences against phishing attacks, including user education and two-factor authentication, are also provided in the poll. Overall, the poll emphasises the significance of comprehending phishing attack methodologies and putting into place efficient defences against these attacks.

The paper entitled “A Review on Phishing Attacks” by Shankar et al. [19] provides an overview of the current state of research on phishing attacks. Beginning with a definition of phishing, the writers go on to describe the various varieties of phishing attacks that can occur, including spear phishing, whaling, and pharming. The authors then go over the different methods attackers use to conduct phishing assaults, such as email spoofing, URL obfuscation, and social engineering. They also go over the effects of phishing attacks on people and groups, such as monetary loss and reputational harm. The study of the literature then moves on to discuss the various methods and strategies put forth for identifying and thwarting phishing attacks. Both technical solutions, like anti-phishing software, and user instruction and training initiatives fall under this category. Finally, the authors identify some of the current challenges and limitations in the field of phishing attack prevention, including the difficulty of detecting sophisticated attacks and the need for more effective user education and training.

The paper entitled “Study on Phishing Attacks” Bhavsar et al. [20] conducted a literature survey on phishing attacks. The research looked at various phishing attack types, their traits, and the different methods employed to carry them out. It also examined the present state of the field’s research and emphasised the demand for additional study to counter phishing attacks. In order to combat this increasing danger, the study’s conclusion offered suggestions for future research, including the creation of fresh detection and prevention methods and the requirement for increased cooperation between academics, business, and law enforcement agencies (Table 1).

Overall, this gap analysis shows that while numerous papers discuss various methods for preventing and mitigating vishing assaults; there is frequently little discussion of certain topics like instruction, guarding against social engineering, and network security. Additionally, some papers ignore other crucial areas in favour of concentrating on particular elements of vishing prevention (like voice biometrics or call authentication) (Table 2).

3 Proposed Methodology

System architecture for defending against vishing attacks can be broken down into three main components:

- Prevention techniques.
- Mitigation techniques.
- User education and awareness (Fig. 1).

Table 1 Comparison based on defending against vishing attacks

Paper title	Authors	Methodology	Key findings
“Vishing: voice phishing attack”	Moosavi et al. (2009)	Literature review	Discussed different types of vishing attacks, their characteristics and how they can be conducted
“A comprehensive survey of vishing (voice phishing) attacks”	Sharma et al. (2016)	Literature review	Classified vishing attacks based on their types, and provided an overview of various vishing attack scenarios and their prevention mechanisms
“A survey on vishing: history, characteristics, threats and countermeasures”	Alomari et al. (2017)	Literature review	Described different types of vishing attacks, their characteristics, and countermeasures. Analysed the effectiveness of various prevention techniques
“Machine learning for detecting vishing attacks”	Qi et al. (2018)	Experimental	Proposed a machine learning-based approach to detect vishing attacks. Conducted experiments to evaluate the proposed system’s performance
“DeepVishing: voice phishing attack with deep learning”	Zhang et al. (2018)	Experimental	Proposed a deep learning-based approach to detect vishing attacks. Evaluated the proposed system’s performance and compared it with other machine learning-based systems
“A machine learning approach to defend against vishing attacks”	Liu et al. (2018)	Experimental	Proposed a machine learning-based approach to detect and prevent vishing attacks. Conducted experiments to evaluate the proposed system’s performance
“A vishing resilient trust anchor for mobile applications”	Xu et al. (2019)	Experimental	Proposed a trust anchor-based approach to prevent vishing attacks on mobile applications. Conducted experiments to evaluate the effectiveness of the proposed system
“A machine learning-based approach for detecting vishing calls”	Ceylan et al. (2019)	Experimental	Proposed a machine learning-based approach to detect vishing calls. Conducted experiments to evaluate the proposed system’s performance
“Voice biometrics for anti-vishing: an experimental study”	De Luca et al. (2020)	Experimental	Conducted an experimental study on using voice biometrics to prevent vishing attacks. Evaluated the performance of different voice biometric techniques
“A multimodal machine learning-based anti-vishing system”	Hossain et al. (2020)	Experimental	Proposed a multimodal machine learning-based approach to detect and prevent vishing attacks. Conducted experiments to evaluate the proposed system’s performance

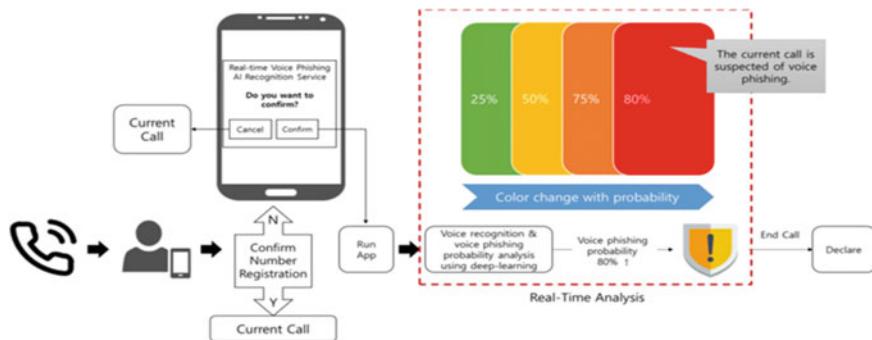
Prevention Techniques

Prevention techniques aim to stop vishing attacks before they can even occur. Some effective prevention techniques include:

- Implementing Two-Factor Authentication:** Two-factor authentication adds an extra layer of security by requiring users to provide two different forms of authentication, such as a password and a unique code sent to their phone.
- Implementing Call Authentication Protocols:** Implementing call authentication protocols like STIR/SHAKEN (Secure Telephone Identity Revisited/Signature-based Handling of Asserted information using toKENs) can help verify the authenticity of calls and protect against caller ID spoofing.

Table 2 Gap analysis based on defending against vishing attacks

Paper title	Main prevention/mitigation techniques	Gaps/unaddressed areas
“Vishing: voice phishing scams”	Awareness, education, detection	No discussion of technical prevention techniques
“A secure voice over IP for preventing vishing attacks”	Encryption, authentication, monitoring	No discussion of user education or social engineering prevention
“Preventing vishing attacks in financial institutions”	Caller authentication, voice biometrics	Limited discussion of other prevention techniques such as education or network security
“Vishing: an emerging threat and its prevention mechanisms”	Authentication, anomaly detection, voice biometrics	Limited discussion of education and social engineering prevention
“Vishing attacks: threats and countermeasures”	Call blocking, call authentication, voice analysis	No discussion of network security or education
“Detecting and preventing vishing attacks on VoIP networks”	Detection, machine learning, voice analysis	Limited discussion of prevention techniques such as education or network security
“Combating vishing: analysis, detection, and prevention”	Call authentication, voice biometrics, user education	No discussion of network security
“A survey of vishing attacks and defenses”	Authentication, voice biometrics, machine learning	Limited discussion of education and social engineering prevention
“Vishing attacks on mobile devices: a threat to banks and their customers”	Voice biometrics, authentication, mobile app security	Limited discussion of prevention techniques such as education or network security

**Fig. 1** System architecture for defending against vishing attacks

- c. **Implementing Filtering Techniques:** Filtering techniques like call blocking, whitelisting and blacklisting can be used to filter out unwanted and potentially malicious calls.

Mitigation Techniques

Mitigation techniques are used to minimize the damage caused by vishing attacks. Some effective mitigation techniques include:

- a. **Establishing an Incident Response Plan:** An incident response plan outlines the steps to be taken in the event of a vishing attack, including who to contact, what actions to take, and how to recover from the attack.
- b. **Monitoring and Analysing Call Data:** Analysing call data can help identify patterns and potential vulnerabilities in the system, allowing for proactive measures to be taken.
- c. **Providing Rapid Response:** Providing rapid response can help minimize the impact of an attack by quickly identifying and stopping the attack.

User Education and Awareness

User education and awareness is a critical component of any vishing defence system. It is essential to educate users about the dangers of vishing attacks and how to identify and respond to them. Some effective user education and awareness techniques include:

- a. **Conducting Regular Training and Awareness Programs:** Regular training and awareness programs can help educate users about the latest vishing threats and how to protect themselves.
- b. **Implementing Phishing Awareness Campaigns:** Phishing awareness campaigns can help educate users about the dangers of vishing and other types of phishing attacks.
- c. **Encouraging Reporting of Suspicious Calls:** Encouraging users to report suspicious calls can help identify potential vishing attacks and allow for rapid response and mitigation.

In summary, a system architecture for defending against vishing attacks involves a multi-layered approach that includes prevention, mitigation, and user education and awareness techniques. By implementing these techniques, organizations can effectively defend against vishing attacks and protect their sensitive information.

4 Discussions

Vishing attacks, also referred to as voice phishing attacks, are a type of social engineering attack in which the attacker uses voice communication, usually over the phone, to trick the target into disclosing private information, such as personal or

financial information. Vishing attacks must be defended against using a thorough strategy that includes both mitigation and protection measures. We will go over some of the best methods for stopping and minimising vishing attacks in this talk.

Employee Training and Awareness

One of the most effective prevention techniques is to train employees and create awareness about the risks associated with vishing attacks [21]. This can include teaching employees how to identify suspicious phone calls, how to avoid giving out sensitive information over the phone, and how to verify the identity of callers before divulging any information [22]. This training can be reinforced through regular security awareness campaigns and simulated phishing exercises.

Caller ID Authentication

Caller ID authentication is a technique that allows the recipient of a call to verify the identity of the caller [23]. This technique can be used to prevent spoofed calls, which are a common technique used in vishing attacks. By verifying the caller's identity, the recipient can avoid falling for a vishing attack.

Two-Factor Authentication

Two-factor authentication (2FA) is a technique that requires users to provide two forms of authentication to access an account [24]. This can be a password and a code sent to a mobile device or email address. Implementing 2FA can help prevent vishing attacks by adding an extra layer of security to account access.

Anti-Vishing Software

Anti-vishing software can be used to detect and block vishing attacks. This software typically uses machine learning algorithms to analyse phone calls and identify suspicious patterns [25]. The software can also be used to block calls from known vishing numbers or to alert users when a call is likely to be a vishing attempt [26].

Network Segmentation

Network segmentation is a technique that involves dividing a network into smaller segments, each with its own security controls [27]. This can help prevent vishing attacks by limiting the access that attackers have to sensitive information [4]. By segmenting the network, attackers are unable to access all the sensitive data in one go, making it harder for them to carry out successful attacks (Table 3).

In conclusion, defending against vishing attacks requires a comprehensive approach that involves prevention and mitigation techniques. Employee training and awareness, caller ID authentication, two-factor authentication, anti-vishing software, and network segmentation are all effective techniques for preventing and mitigating vishing attacks. Organizations should implement a combination of these techniques to ensure they are protected against vishing attacks.

Table 3 Techniques for defending against vishing attacks

Paper title	Authors	Techniques
Vishing: trends and mitigation techniques	Bou-Hamad et al. (2021)	Voice biometrics, machine learning, call verification, and awareness
A survey on vishing attacks	Alvi et al. (2021)	Call verification, training, and awareness
A literature review of vishing attacks and defense mechanisms	Ahuja et al. (2020)	Voice biometrics, machine learning, and awareness
Countermeasures for vishing attacks	Jain et al. (2020)	Call verification, biometric authentication, and awareness
Preventing and mitigating vishing attacks: a survey	Kambli et al. (2019)	Call verification, voice biometrics, and awareness
Vishing attacks and their prevention techniques: a review	Yadav et al. (2018)	Call verification, voice biometrics, and awareness
A review on vishing attacks and countermeasures	Ali et al. (2018)	Call verification, biometric authentication, and awareness
Vishing attacks and its prevention techniques: a review	Singh et al. (2017)	Call verification, voice biometrics, and awareness
Vishing attacks: a review of the state-of-the-art research	Aravinda et al. (2016)	Call verification, voice biometrics, and awareness
Vishing: an emerging threat	Bhadoria et al. (2013)	Call verification, user education, and awareness

5 Conclusions

In today's digitally advanced world, vishing attacks have become a significant threat to individuals and organizations alike. Vishing attacks can result in severe financial and reputational damage, making it crucial to implement prevention and mitigation techniques to safeguard against such attacks. This review provides an overview of various prevention and mitigation techniques that can be used to defend against vishing attacks. It highlights the importance of employee training and education, the implementation of multi-factor authentication, and the use of voice biometrics to authenticate customers. Additionally, organizations should regularly review their security policies, procedures, and controls to ensure they remain effective in preventing and mitigating vishing attacks.

In conclusion, preventing and mitigating vishing attacks requires a proactive and comprehensive approach that involves multiple layers of security controls, including employee education and awareness, technical controls, and policy and procedure reviews. By implementing these techniques, individuals and organizations can protect themselves from the devastating effects of vishing attacks.

References

1. Nmachi WP, Win T (2021) Phishing mitigation techniques: a literature survey. *Int J Netw Secur Appl* 13(2):63–72. <https://doi.org/10.5121/ijnsa.2021.13205>
2. Juan C, Chuanxiong G (2007) Online detection and prevention of phishing attacks (invited paper). In: 2006 first international conference on communications and networking in China, ChinaCom'06. <https://doi.org/10.1109/CHINACOM.2006.344718>
3. Andronova IV, Belova IN, Ganeeva MV, Moseykin YN (2018) Scientific technical cooperation within the EAEU as a key factor of the loyalty of the participating countries' population to the integration and of its attractiveness for new members. *RUDN J Sociol* 18(1):117–130. <https://doi.org/10.22363/2313-2272-2018-18-1-117-130>
4. Gautam H, Kumar V, Sharma V (2021) Phishing prevention techniques: past, present and future, pp 83–98. https://doi.org/10.1007/978-981-33-6307-6_10
5. Bhusal CS (2021) Systematic review on social engineering: hacking by manipulating humans. *J Inf Secur* 12(01):104–114. <https://doi.org/10.4236/jis.2021.121005>
6. Al-Qahtani AF, Cresci S (2022) The COVID-19 scamdemic: a survey of phishing attacks and their countermeasures during COVID-19. *IET Inf Secur* 16(5):324–345. <https://doi.org/10.1049/ise2.12073>
7. Chandre PR, Mahalle PN, Shinde GR (2018) Machine learning based novel approach for intrusion detection and prevention system: a tool based verification. In: 2018 IEEE global conference on wireless computing and networking (GCWCN), Nov 2018, pp 135–140. <https://doi.org/10.1109/GCWCN.2018.8668618>
8. Perwej DY, Abbas SQ, Dixit JP, Akhtar DN, Jaiswal AK (2021) A systematic literature review on the cyber security. *Int J Sci Res Manag* 9(12):669–710. <https://doi.org/10.18535/ijsrn/v9i12.ec04>
9. Chandre P, Mahalle P, Shinde G (2022) Intrusion prevention system using convolutional neural network for wireless sensor network. *IAES Int J Artif Intell* 11(2):504–515. <https://doi.org/10.11591/ijai.v11.i2.pp504-515>
10. Do NQ, Selamat A, Krejcar O, Herrera-Viedma E, Fujita H (2022) Deep learning for phishing detection: taxonomy, current challenges and future directions. *IEEE Access* 10:36429–36463. <https://doi.org/10.1109/ACCESS.2022.3151903>
11. Alkhaili Z, Hewage C, Nawaf L, Khan I (2021) Phishing attacks: a recent comprehensive study and a new anatomy. *Front Comput Sci* 3:1–23. <https://doi.org/10.3389/fcomp.2021.563060>
12. Chandre PR (2021) Intrusion prevention framework for WSN using deep CNN. *Turk J Comput Math Educ* 12(6):3567–3572
13. Arshad A, Rehman AU, Javaid S, Ali TM, Sheikh JA, Azeem M (2021) A systematic literature review on phishing and anti-phishing techniques, pp 163–168. [Online]. Available: <http://arxiv.org/abs/2104.01255>
14. Salahdine F, Kaabouch N (2019) Social engineering attacks: a survey. *Future Internet* 11(4). <https://doi.org/10.3390/FI11040089>
15. Sadiq A et al (2021) A review of phishing attacks and countermeasures for internet of things-based smart business applications in industry 4.0. *Hum Behav Emerg Technol* 3(5):854–864. <https://doi.org/10.1002/hbe2.301>
16. Ahsan M, Nygard KE, Gomes R, Chowdhury MM, Rifat N, Connolly JF (2022) Cybersecurity threats and their mitigation approaches using machine learning—a review. *J Cybersecur Priv* 2(3):527–555. <https://doi.org/10.3390/jcp2030027>
17. Bhuvana, Bhat AS, Shetty T, Naik MP (2021) A study on various phishing techniques and recent phishing attacks. *Int J Adv Res Sci Commun Technol* 11(1):142–148. <https://doi.org/10.48175/ijarsct-2094>
18. Chawla M, Chouhan SS (2014) A survey of phishing attack techniques. *Int J Comput Appl* 93(3):32–35. <https://doi.org/10.5120/16197-5460>
19. Shankar A, Shetty R, Nath B (2019) A review on phishing attacks. *Int J Appl Eng Res* 14(9):2171–2175. [Online]. Available: <http://www.ripublication.com>

20. Bhavsar V, Kadlak A, Sharma S (2018) Study on phishing attacks. *Int J Comput Appl* 182(33):27–29. <https://doi.org/10.5120/ijca2018918286>
21. Priestman W, Anstis T, Sebire IG, Sridharan S, Sebire NJ (2019) Phishing in healthcare organisations: threats, mitigation and approaches. *BMJ Health Care Inform* 26(1):1–6. <https://doi.org/10.1136/bmjhci-2019-100031>
22. Bojjagani S, Brabin DRD, Rao PVV (2020) PhishPreventer: a secure authentication protocol for prevention of phishing attacks in mobile environment with formal verification. *Procedia Comput Sci* 171(2019):1110–1119. <https://doi.org/10.1016/j.procs.2020.04.119>
23. Mahalakshmi A, Goud NS, Murthy GV (2018) A survey on phishing and it's detection techniques based on support vector method (SVM) and software defined networking (SDN). *Int J Eng Adv Technol* 8(2):498–503
24. Deloitte (2014) Fraud risk management—providing insight into fraud prevention, detection and response, pp 1–12. [Online]. Available: <http://www2.deloitte.com/content/dam/Deloitte/in/Documents/finance/Forensic-Proactive-services/in-fa-frm-noexp.pdf>
25. Chin T, Xiong K, Hu C (2018) PhishLimiter: a phishing detection and mitigation approach using software-defined networking. *IEEE Access* 6:42513–42531. <https://doi.org/10.1109/ACCESS.2018.2837889>
26. Abbas SG et al (2021) Identifying and mitigating phishing attack threats in IoT use cases using a threat modelling approach. *Sensors* 21(14):1–25. <https://doi.org/10.3390/s21144816>
27. FireEye Inc. (2016) Spear-phishing attacks why they are successful and how to stop them, pp 1–9. [Online]. Available: <https://www.fireeye.com/content/dam/fireeye-www/global/en/products/pdfs/wp-fireeye-how-stop-spearphishing.pdf>

Categorizing Tracing Techniques for Network Forensics



Shraddha Chourasiya , Ayush Indurkar , Apoorva Ghagare ,
Kaushal Potphode , Varun Sayam , and Dikshant Gaikwad

Abstract The traceback problem is widely acknowledged as one of the most challenging issues in the field of information security, and it is crucial to find effective solutions to hold attackers accountable for their actions. This research paper aims to provide a comprehensive introduction to the intricate issue of traceback in network forensics. To fully comprehend the problem, the paper delves into the unique characteristics of computer, network, and software forensics. By exploring the distinctive features and challenges of each forensic subfield, a comprehensive understanding of the broader context in which traceback techniques are utilized is established. Furthermore, the paper analyses various traceback mechanisms and categorizes them based on their distinct features and modes of operation. In conclusion, the study proposes a systematic classification system for all traceback techniques, evaluating and integrating their respective advantages. This offers valuable insights for digital forensics investigations and brings us closer to identifying the actual perpetrator.

Keywords Supervised machine learning · Network · Traceback · Detection system · Intrusion · Deep learning · Forensic

S. Chourasiya · A. Indurkar () · A. Ghagare · K. Potphode · V. Sayam · D. Gaikwad
Department of Computer Science Engineering, Jhulelal Institute of Technology, Nagpur,
Maharashtra, India
e-mail: indurkar.ayush67@gmail.com

1 Introduction

The emergence of highly skilled cybercriminals has increased the need for sophisticated intrusion detection systems (IDSs) that can handle these threats autonomously and intelligently [1]. IDS solutions based on autonomous agents are in high demand because they can continuously improve and adapt with little to no human involvement. Automated agents can now recognize and categorize different types of attacks using reinforcement learning (RL), which is becoming increasingly popular. An RL agent can create a defence plan that improves the environment's overall protection by watching and examining attack behaviours in a particular setting. By rewarding or punishing the agent's behaviours depending on feedback from the environment, it can develop its skills over time and learn the best strategies for a particular situation [2].

Several RL-based intrusion detection approaches have been proposed in recent years for a variety of application situations, including IoT, Wireless Networks, and Cloud. However, because to the state explosion problem that occurs when the RL agent deals with various learning states, many present RL-based approaches have trouble reliably detecting real network traffic and handling big datasets. Deep reinforcement learning (DRL) approaches, such as deep Q-learning, have been proposed as potential remedies to address this issue. Deep neural networks are used by DRL approaches to speed up learning in situations with a large number of states. Several DRL-based IDS techniques have been proposed in the literature for network intrusion detection [3, 4]. These techniques use diverse intrusion datasets to train and evaluate their models.

Although there are several methods designed to improve detection performance and capabilities, in-depth studies into the creation and DRL-based deployment of an IDS strategy especially suited network environments are lacking. This paper's goal is to present the strategy, procedure, and tactics for applying the upcoming version of the DQL approach for identifying network intrusions [5].

2 Propose System

The software works on four integrated module that are pre-processing, feature extraction, attack detection, and tracing of attacks. It can be applied between the firewall and the internet (Fig. 1).

The initial step in network forensics is preprocessing, which involves the extraction of valuable information from raw network traffic data [6]. Preprocessing involves the conversion of unstructured raw data into a structured format that can be analysed and investigated. This module comprises cleaning, normalization, and transformation of the raw network traffic data. Feature extraction is a crucial stage in network forensics, which involves transforming unprocessed network data into valuable information for identifying potential threats. To facilitate the selection and application of

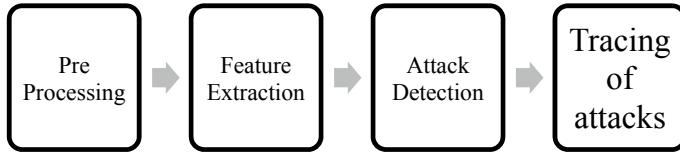


Fig. 1 Working modules of the system

the most suitable traceback mechanism for a given scenario, a proposed classification system aims to categorize various mechanisms based on their distinctive features and attributes. Attack detection is the process of identifying and characterizing various types of network attacks. Its aim is to develop efficient detection techniques to quickly and accurately detect attacks as they happen, enabling timely response and mitigation [7]. The process entails examining network traffic to detect patterns and behaviours that suggest malicious activity. Various techniques have been proposed for attack detection, consisting of behaviour-based, anomaly-based, and signature-based detection. It is crucial to comprehensively classify these techniques to better understand their pros and cons and to guide their use in different network scenarios. Tracing attacks is a crucial aspect of network forensics that involves identifying and mapping the path taken by attackers to locate the source of the attack [8]. Tracing mechanisms can be categorized based on their approach, such as packet-based, flow-based, and agent-based tracing. Packet-based tracing examines individual packets to identify the source and destination of the traffic, while flow-based tracing traces the path of aggregated traffic flows. Agent-based tracing uses software agents to monitor and trace network traffic [9]. The selection of tracing approach depends on the network requirements and the type of attack being traced, and each approach has its own strengths and limitations [10].

3 Detection Methods

Misuse-based and anomaly-based detection are the two broad categories into which intrusion detection techniques can be divided. Utilizing the advantages of each strategy, modern intrusion detection systems (IDSs) frequently combine these techniques to increase their efficiency. Let's get more into these methods of detection:

- (a) Misuse-Based Detection: This technique compares ongoing activity with previously saved profiles of the distinctive characteristics of prior attacks. Due to its efficiency in identifying known threats and its low rate of false alarms, commercial IDSs frequently use this strategy [11]. The system refreshes its database by adding attack signatures when a new attack is discovered, and implementation is rather simple. This strategy, meanwhile, has drawbacks because it cannot recognize unique attacks, even if they diverge significantly from a recognized

pattern. The robustness of the system's signature database, which needs to be updated frequently, is crucial [12].

- (b) Anomaly-Based Detection: This method seeks to identify “unusual” patterns of behaviour displayed by hosts, users, or network connections over time. By profiling typical user behaviours, it begins by building a model of usual behaviour. Any differences between the present monitored activity and these profiles are noted as anomalies (possible attack attempts) by the system during detection. Due to the continually changing usual behaviour brought on by the inclusion of new features or technologies, anomaly-based detection need frequent updates. It labels routine activities as anomalies but has a higher risk of false positives. The benefit of anomaly-based IDS is its capacity to identify new attacks that have no connection to existing attacks. This is especially crucial in situations where new attacks are found [6].
- (c) Hybrid Detection: This IDS combines both the anomaly-based and misuse-based techniques to improve known attack detection rates while reducing false positive rates for unidentified assaults. It involves two processing components, and the choice to sound an alert is made after considering many options. One technique, for instance, detects aberrant behaviours while decoding events, whereas a different detection engine matches signatures [4].

IDSs can provide a more thorough and effective detection capability in identifying existing attacks, detecting novel assaults, and minimizing false positives by combining these approaches.

4 Implementation

The system begins by defining several constant variables that are used throughout the program. These include indexes for the different fields in the packet data (such as protocol, type, size, and port), false criteria for rejecting packets that don't meet certain requirements (such as minimum and maximum packet size and port number), and trusted ports that are used for merging packets later in the system [9]. Feature extraction which collects network packets, filters out false ones and groups similar ones together, then clusters them based on their port numbers and packet sizes, and detects if there are any attacks based on predefined criteria. The program starts by importing various Java libraries including BufferedReader, FileReader, ArrayList, and Date. It also imports JFreeChart and RefineryUtilities libraries for creating charts. It then defines several constants that are used throughout the program. These constants define the indexes of various fields in the packets, as well as criteria for filtering out false packets and merging clusters [3]. For example, the packet size must be within a certain range, and the port number must be above a certain value to be considered valid (Fig. 2).

The program also defines the number of iterations to perform for clustering, the maximum number of intrusions for known attacks, and trusted ports. The program

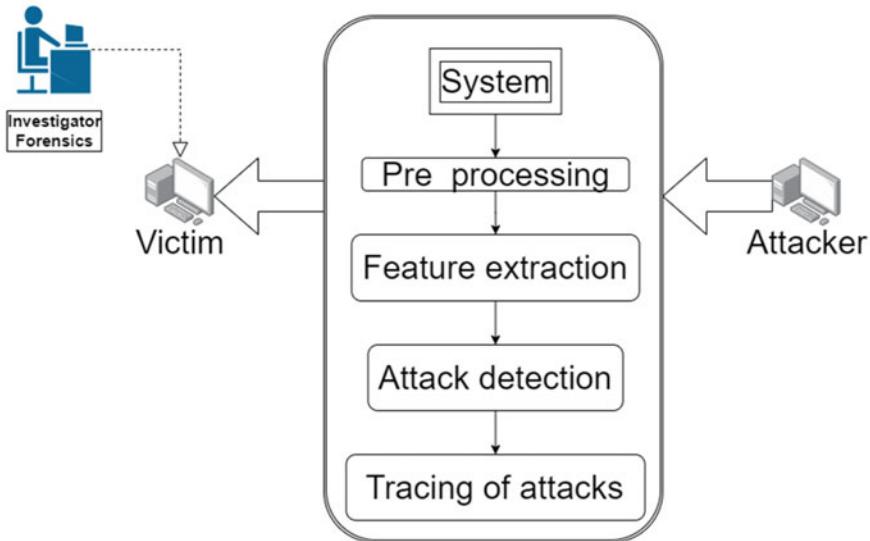


Fig. 2 Use case diagram

then defines a function called `collectAndDetect`, which takes a file path as an input parameter. This function reads the file line by line, parses each line into a `PacketFormat` object, and checks whether the packet is a false packet based on the predefined criteria. If a packet is not a false packet, it is added to an `ArrayList` called `lstPackets`. Next, the program performs K-means clustering on the port numbers and packet sizes of the packets in `lstPackets`. The number of clusters is specified by the constant `NUM_ITERATIONS`. The program then checks whether there are any intrusions by iterating through the clusters and checking whether any non-trusted ports fall within a cluster. If there are any such ports, they are considered to be part of an attack. Finally, the program creates a Pie chart using JFreeChart and displays it on the screen. After that system analyse a type of attack and respond with appropriate corrective steps. The method first checks the type of attack by performing AES decryption. If the attack is known, it calculates the attack magnitude by dividing the number of clusters by the maximum number of intrusions [2]. Then it outputs a message based on the magnitude and updates the text output in the GUI. If the attack is unknown, it updates the strategies and applies them. If any exceptions occur during the decryption or the processing of the result, the method returns the exception message. Thus the system detects the attacks and traces it back with the above modules [5].

Figure 3 shows the application's GUI. The network's preprocessing is depicted in Fig. 4. The extraction of the clusters is depicted in Fig. 5. Figure 6 depicts the attack that was found. The accuracy of the assault detected is displayed in Fig. 7.



Fig. 3 System GUI

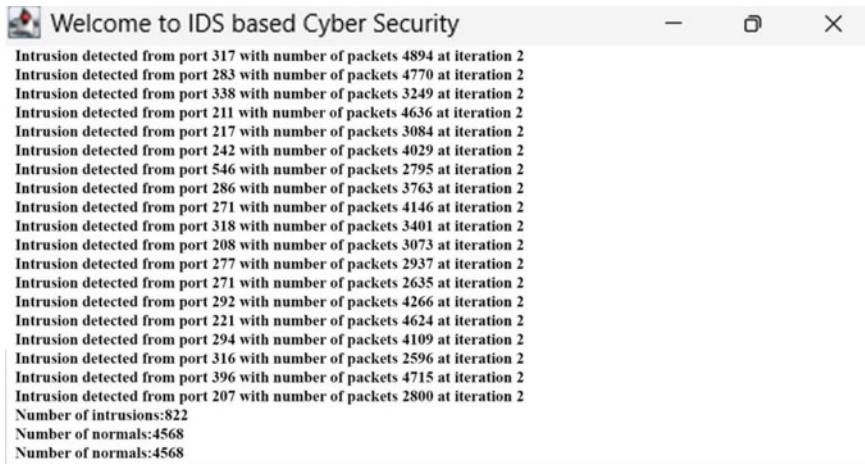
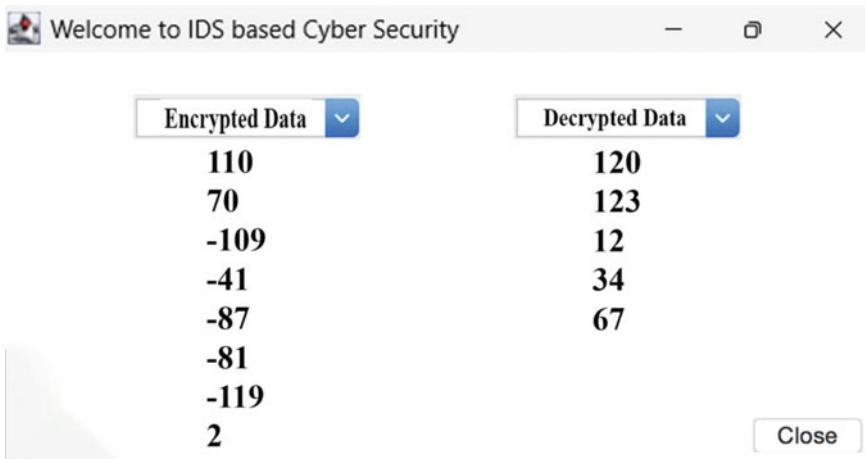
Packet tcp, smtp, 543,1151,78.244 added for clustering 5384
Packet tcp,smtp, 499,811,88.243 added for clustering 5385
Packet tcp, smtp, 501, 5208,98.244 added for clustering 5386
Packet tcp, smtp, 488,1889, 108.244 added for clustering 5387
Packet tcp, smtp, 593,1579, 118.244 added for clustering Packet tcp, smtp, 494,1000, 128.244 added for clustering 5388
Packet tcp, smtp,493,950,138.243 added for clustering 5390
Packet icmp, ecr_i,0,1032,148.3 cleaned Packet icmp, ecr_i,0,1032,158.13 cleaned Packet icmp, ecr_i,0,1032,168.23 cleaned
Packet icmp, ecr_i,0,1032, 208.63 cleaned
Packet icmp, ecr_i,0,1032,218.73 cleaned Packet icmp, ecr_i,0,1032,228.83 cleaned
Packet icmp, ecr_i,0,1032, 238.93 cleaned Packet icmp, ecr_i,0,1032,248.103 cleaned
Packet icmp, ecr_i,0, 1032,255.113 cleaned Packet icmp, ecr_i,0,1032, 255.123 cleaned Packet icmp, ecr_i,0,1032,255.133 cleaned
Packet icmp, ecr_i,0,1032,255.143 cleaned
Packet icmp, ecr_i,0,1032,255.153 cleaned
Packet icmp, ecr_i,0,1032,255.163 cleaned Packet icmp, ecr_i,0,1032,255.173 cleaned
Packet icmp, ecr_i,0,1032,255.183 cleaned
Packet icmp, ecr_i,0,1032,255.193 cleaned
Packet icmp, ecr_i,0,1032,255.203 cleaned

Fig. 4 Working of preprocessing module

5 Propose Algorithm

5.1 C Means Algorithm

```
function CMeansAlgorithm(PortNumbers, PacketSizes) :  
    cmeans_distance_ports = mean(PortNumbers)  
    cmeans_distance_packets = mean(PacketSizes)  
    clusters = []  
    for i in range(len(PortNumbers)) :  
        port = PortNumbers[i]  
        size = PacketSizes[i]
```

**Fig. 5** Working of feature extraction**Fig. 6** Working of attack detection

```

if port < cmeans_distance_ports and size < cmeans_distance_
packets:
    clusters.append(1)
else if port < cmeans_distance_ports and size >= cmeans_
distance_packets and size <= 10 * cmeans_distance_packets:
    clusters.append(2)
else if port >= cmeans_distance_ports and port <= 80 * cmeans_
distance_ports and size < cmeans_distance_packets:
    clusters.append(3)

```



Fig. 7 Working of tracing of attacks

```
else if port >= cmeans_distance_ports and port <= 80 * cmeans_
distance_ports and size >= cmeans_distance_packets and size <= 10 *
cmeans_distance_packets:
    clusters.append(4)
else if port > 80 * cmeans_distance_ports and size < cmeans_
distance_packets:
    clusters.append(5)
else if port > 80 * cmeans_distance_ports and size >= cmeans_
distance_packets and size <= 10 * cmeans_distance_packets:
    clusters.append(6)
else if port > 80 * cmeans_distance_ports and size > 10 * cmeans_
distance_packets:
    clusters.append(7)
else:
    clusters.append(8)
return clusters
```

6 Literature Review

See Table 1.

Table 1 Literature review

S. No.	Authors	Title of paper	Name of journal/conference	Details
1.	H. Alavizadeh, Hootan, Alavizadeh, J. Jang-Jaccard	Reinforcement based on deep Q-learning learning-based network intrusion detection methodology	Institute of Electrical and Electronics Engineers (IEEE) access	A state-of-the-art network intrusion detection technique has been developed that combines a deep feed-forward neural network strategy with Q-learning, a reinforcement learning technique
2.	Bo Hu, Jiaxi Li	A case study in powertrain control: moving deep reinforcement learning algorithm towards training directly in transient real-world environment	Journal of Industrial Informatics, IEEE	Without the need for prior task knowledge, (DRL) exhibits competence when playing virtual games. The suggested algorithm achieves a notable improvement of 74.6% over the baseline performance. Furthermore, it shows a tremendous boost in learning efficiency of 10 times over traditional DRL techniques
3.	K. Sethi, Y. V. Madhav, R. Kumar, P. Bera	Multiagent intrusion detection systems with attention-based reinforcement learning	Information Security and Applications Journal	A cutting-edge intrusion detection system (IDS) that applies deep Q-network (DQN) logic and deep reinforcement learning (DRL) across numerous distributed agents. This novel method includes attention processes to quickly identify and categorize sophisticated network threats
4.	J. Hou, Q. Li, S. Cui, S. Meng, S. Zhang, Z. Ni, and Y. Tian	For industrial internet, low cohesion differential privacy protection	The Supercomputing Journal	We provide a differential privacy protection strategy created specifically for frequent pattern mining in light of the privacy protection requirements at the application level in industrial linked systems. With the use of this technique, sensitive data can be protected while being securely analysed for patterns
5.	K. Toyoshima, T. Oda, M. Hirota, K. Katayama, and L. Barolli	Results of simulations of different distributions of events taking into account the three-dimensional environment for a DQN-based mobile actor node control in WSAN	International Conference on Developing Web, Data, and Networking Technologies	Deep Q-network (DQN) is used in their suggested simulation system design to regulate actor node mobility in wireless sensor and actor networks (WSANs). The Q-value for Q-learning is estimated by the deep neural network DQN. With this approach, decision-making is improved and node movements are optimized, resulting in efficient control of actor node mobility in WSANs
6.	K. Sethi, R. Kumar, D. Mohanty, and P. Bera	Intelligent system for detecting cloud intrusions using deep reinforcement learning	Conference on Applied Cryptography Engineering and Security, Privacy	DDQN and prioritized experience replay techniques are combined in the adaptive IDS model to effectively identify brand-new and sophisticated assaults on cloud systems

(continued)

Table 1 (continued)

S. No.	Authors	Title of paper	Name of journal/conference	Details
7.	T. T. Nguyen and V. J. Reddi	Cybersecurity with deep reinforcement learning	arXiv preprint arXiv:1906.05799	Analyses of DRL techniques created for cyber security
8.	G. Caminero, M. Lopez-Martin, and B. Carro	Intrusion detection using an adversarial environment reinforcement learning method	Computer Networks, vol. 159	The prediction is based on a classifier that makes use of a straightforward yet incredibly effective neural network
9.	D. Bodeau and R. Graubart	Designing for cyber resilience involves using it sparingly throughout its lifecycle and in collaboration with adjacent disciplines	McClean (VA): The MITRE Corporation	A thorough collection of design guidelines for cyber resilience is provided, covering a variety of aspects to take into account when deciding which guidelines are most suited for a particular system, programme, or system-of-systems
10.	M. P. Stoecklin	How a stealthy new generation of malware is powered by AI	Vol. 8 of Security Intelligence, August	By taking a fundamentally different tack from the existing evasive and targeted malware approaches, DeepLocker has completely changed the landscape of malware evasion
11.	X. Ma and W. Shi, "Aesmote"	Adversarial reinforcement learning with smite for anomaly detection is known as aesmote	Journal of Network Science and Engineering, IEEE	Their strategy attempts to successfully handle the pervasive dataset imbalance problem seen in existing learning-based solutions while leveraging the auto-learning capabilities of the reinforcement-learning loop
12.	Z. S. Stefanova and K. M. Ramachandran	Off-policy Q-learning technique for network security intrusion response	International Science Index, vol. 136, World Academy of Science, Engineering, and Technology	By modifying the environment agent's behaviours and modifying SMOTE to address class imbalance, they improved performance
13.	M. Brundage, S. Avin, J. Clark, H. Toner, P. Eckersley, B. Garfinkel, A. Dafoe, P. Scharre, T. Zeitzoff, B. Filar et al.	Prediction, prevention, and mitigation of artificial intelligence's nefarious applications	arXiv preprint arXiv:1802.07228	This study suggests a number of potential research areas that could broaden the scope of defences, reduce the potency of attacks, or make attacks more difficult to execute

7 Results

By integrating a deep feed-forward neural network and the reinforcement learning technique Q-learning, our system offers a novel and cutting-edge way for network intrusion detection. An advanced network IDS that can self-learn and detect different sorts of network invasions is the result of this integration [8]. Our model's capacity to

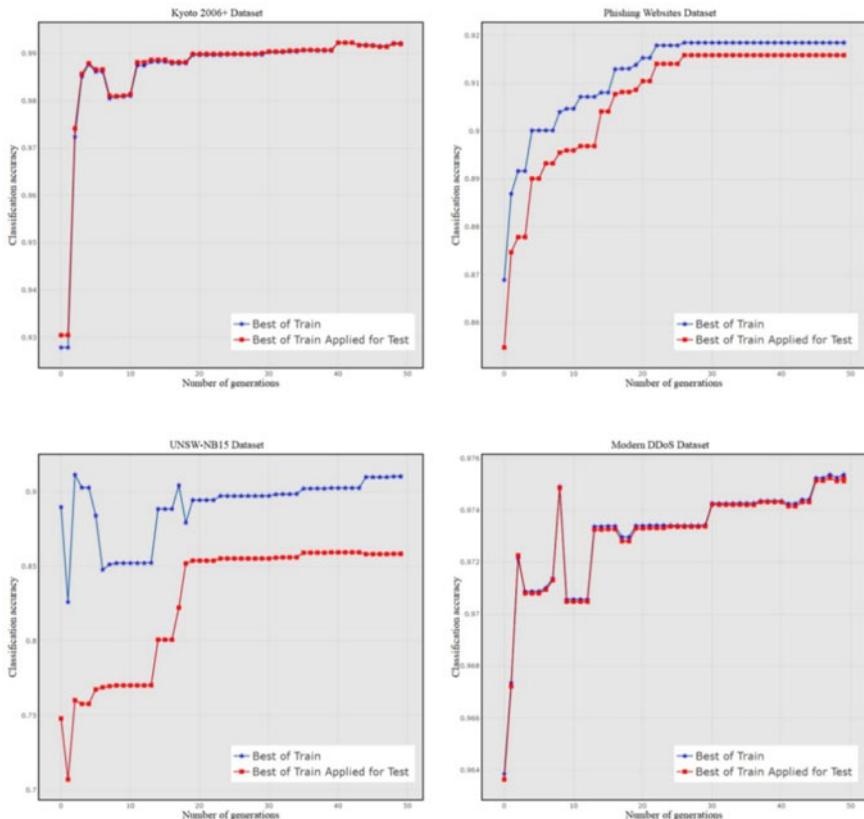


Fig. 8 Relationship between accuracy and generations, according to CGP

continuously auto-learn improves its intrusion detection abilities and enables long-term improvement, which is a critical feature (Fig. 8).

References

1. Lopez-Martin M, Carro B, Sanchez-Esguevillas A (2020) Application of deep reinforcement learning to intrusion detection for supervised problems. *Expert Syst Appl* 141:112963
2. Hou J, Li Q, Cui S, Meng S, Zhang S, Ni Z, Tian Y (2020) Low-cohesion differential privacy protection for industrial internet. *J Supercomput* 76(11):8450–8472
3. Sethi K, Madhav YV, Kumar R, Bera P (2021) Attention based multiagent intrusion detection systems using reinforcement learning. *J Inf Secur Appl* 61:102923
4. Zhu J, Jang-Jaccard J, Singh A, Watters PA, Camtepe S (2021) Task-aware meta learning-based Siamese neural network for classifying obfuscated malware. arXiv preprint [arXiv:2110.13409](https://arxiv.org/abs/2110.13409)
5. Xu W, Jang-Jaccard J, Singh A, Wei Y, Sabrina F (2021) Improving performance of autoencoder-based network anomaly detection on NSL-KDD dataset. *IEEE Access* 9:140136–140146

6. Hu B, Li J (2021) Shifting deep reinforcement learning algorithm towards training directly in transient real-world environment: a case study in powertrain control. *IEEE Trans Ind Inform*
7. CVE security vulnerability database. [Online]. Available: <https://www.cvedetails.com/browse-by-date.php>. Accessed 10 Feb 2019
8. Shi W-C, Sun H-M (2020) DeepBot: a time-based botnet detection with deep learning. *Soft Comput* 24:16605–16616
9. Dang Q-V, Vo T-H (2022) Reinforcement learning for the problem of detecting intrusion in a computer system. In: Proceedings of sixth international congress on information and communication technology. Springer, pp 755–762
10. CICFlowmeter: a network traffic biflow generator and analyzer. [Online]. Available: <http://netflowmeter.ca/>. Accessed 11 Feb 2019
11. Nguyen TT, Reddi VJ (2019) Deep reinforcement learning for cyber security. arXiv preprint [arXiv:1906.05799](https://arxiv.org/abs/1906.05799)
12. Sethi K, Rupesh ES, Kumar R, Bera P, Madhav YV (2020) A contextaware robust intrusion detection system: a reinforcement learning-based approach. *Int J Inf Secur* 19(6):657–678

malC: A Novel Deep Learning Architecture for Malware Classification



Harinadh Varikuti and ValliKumari Vatsavayi

Abstract Malware attacks impact organizations resulting in password theft, encrypting the files, denial of service, etc. Every malware file has its own malicious intent, which either matches the known malware family or is detected as a new kind of malware. Analysts perform static and dynamic analysis on malware files to extract the behavior. Identifying the malware samples from the large data depends on the functionality and patterns. Feature extraction from the large data sets is mainly based on the expert's domain knowledge and various approaches used to extract the important features from the raw input. Using deep-learning techniques, neural networks with multiple layers perform the feature extraction and classification altogether. In this paper, we design a novel convolutional neural network architecture named malC model identified as malware classification model. We applied the model on the benchmark dataset released for the Microsoft Malware Classification Challenge. The results show that the malC model outperforms several deep learning methods published in the literature.

Keywords Malware · CNN · malC · Machine learning · Deep learning · Classification

1 Introduction

It is known fact that malware attacks, phishing attacks, SQL injection attacks, etc. cause a huge damage to the reputation of the organizations by stealing the valuable information. Recently, during the war between Russia and Ukraine [1], multiple cyber-attacks were launched. The first cyber-attack targeted the Ukraine government websites including Ministry of Foreign Affairs, Security, and Defence Council. It is said that, as the websites were developed and maintained by third party attackers targeted this path to execute the attack.

H. Varikuti · V. Vatsavayi
Andhra University, Visakhapatnam, India
e-mail: varikutiharinadh@gmail.com

Malware constitutes malicious software, which aims to execute illegal operations on the computer systems or networks. According to Comparitech [2] statistics, in 2020, nearly 61% of organizations experienced malware attack that spread from one employee to another, it rose to 74% in 2021 and it hits 75% in 2022. Everyday new malwares are generated which increases the threat percentage to the users throughout the world. New defense mechanisms are being explored by the researchers to fight against the malwares. According to AV-TEST [3], everyday 450,000 malware programs and potentially unwanted applications are registered which are analyzed and classified based on features of the malwares.

As digitalization is increasing across the world, attackers get more opportunities to perform the illegal actions. Meanwhile research communities are also working rigorously to fight against the actions performed by the attackers. Traditionally, anti-malware engines are applied to detect the malware files. Many malware files usually escaped from these engines and get executed resulting in loss of data or end up damaging the system. According to persistence market research, the global malware analysis market size was USD 5.3 billion in 2021 which increases its growth to USD 6.8 billion in 2022 and likely to reach USD 28.1 billion by 2032. Some of the key factors contributing to the growth of malware industry are (i) rapid increase in frequency of malware attacks, (ii) more incidents of ransomware attacks, and (iii) the everchanging threat landscape. In 2020, when entire world affected with novel corona virus, i.e., COVID-19. Though, most of the industries got affected badly, or some industries shut down their operations permanently, malware industry is one of the markets which thrived through. Many companies have ordered their employees to follow remote work procedures. This situation helped in cyber-attacks and many false security alarms. With this, most of the companies adopted the latest security systems which include malware analysis systems.

1.1 Malware Analysis and Machine Learning

Malware analysis is the study of examining and detecting the malicious software files about its origin, functionality, and behavior. Malware analysis helps organizations to use advanced techniques for identifying the suspicious files. Generally, we have two types of malware analysis: (1) Static Analysis: Analysis is performed on the files without executing the programs. Anti-virus applications were used to quarantine the malware file. Some malware files use obfuscated techniques to evade from static analysis. (2) Dynamic Analysis: Analyzing the suspicious files by executing it in an isolated environment. This analysis monitors and traces, how code interacts with the system and what changes it performs.

Feature engineering is the preprocessing step used in the machine learning to extract features. Traditional machine learning approaches extract features manually defined by the domain experts. These solutions are purely based on the expert knowledge and techniques used for the feature extraction which limits its suitability in the real world. To overcome the issues caused by the traditional approaches, deep learning techniques are used.

In this paper, we present a novel deep convolution neural network architecture named malC for performing malware classification. Deep learning approaches extract sophisticated features which reflect the malware samples effectively. The performance of the malC model was evaluated on benchmark dataset of Microsoft Malware Classification Challenge [4].

2 Related Work

In this section, the research studies performed in the field of malware detection and classification are presented. Traditional machine learning approaches extract the hand-craft features in the feature generation process. With limited domain expert knowledge, feature extraction is difficult to perform. By observing the immense growth in the machine learning field, malware analysis has advanced toward deep learning. Using deep learning techniques, an end-to-end model takes input of malware sample with little preprocessing and directly detects its related malware family as output. To deal with malware byte files, various approaches [5–11] use gray-scale images for malware classification. Narayanan et al. [6] used principal component analysis (PCA) for getting reduced features from the images, then various machine learning algorithms such as K-nearest neighbor, support vector machines, and artificial neural network (ANN) were applied. K-nearest neighbor gives the best performance with the PCA features. In the paper [7], authors proposed deep learning model which uses autoencoders [12] in the multilayer architecture. Also, the proposed model outperforms when compared with traditional machine learning algorithms and pattern recognition algorithms. Feature extraction is time-consuming process in traditional algorithms implementation.

As per the studies, convolutional neural network (CNN) models performed well on the image data set. Gibert et al. [9] presented a CNN architecture for the classification of malware images, which performs better with minimal prediction time. Gibert et al. [8] represent the malware executable as an entropy stream [13], which gives randomness of the system. Haar wavelet transform was used on the entropy times series to reduce the noise and size, and CNN model was applied for malware classification. The author concluded that wavelet features performed strongly against obfuscation techniques. Singh et al. [10] performed malware analysis on Android malware images using feature fusion methods. Malware APK files have been used to generate gray-scale images. Hand-crafted feature extraction methods such as Gray Level Co-occurrence Matrix (GLCM), Global Image descriptors (GIST), and Local Binary Pattern (LBP) uproot features. Traditional machine learning approaches

such as support vector machines, K-nearest neighbor, and random forest have been applied on the above hand-crafted features. Feature fusion applied with CNN and hand-crafted features to perform malware classification. Feature fusion method gives better results when compared with traditional methods. Xiao et al. [11] uses structural entropy graphs for extracting features using CNN model, then Support Vector Machine method was applied to perform malware classification. Yamashita et al. [5] provide an overview of CNN model, also gave how to overcome overfitting using various techniques such as data augmentation, batch normalization, and dropout. Khan et al. [14] classified the malware using deep neural networks using binary files, later a CNN model was built using gray-scale images as input. CNN model gives better results emphasizing the fact that convolution models are more efficient in classifying the image data. Alam et al. [15] used standard neural network models such as VGG-16, ResNet-50, InceptionV3 and proposed convolutional neural network for classifying the malware data which gives significant performance. Depuru et al. [16] performed comparative analysis of malware classification using neural network models and machine learning models. Shinde et al. [17] constructed four neural networks with malware data image sizes as 224×224 , 240×240 , 260×260 and 300×300 , respectively. Using large dataset, the neural network which uses high image pixel size gives better results when compared with low image pixel size. From the above-discussed papers, it is observed that deep learning models gives good results when compared with the machine learning models in classifying and detecting the malware samples for both balanced and unbalanced data. This paper presents an optimized model which gives better results when compared to the models discussed above.

3 Malware Classification

Malware classification is the process of identifying the malware files into their families based on its features and behavior. Every malware family has its own functionality and behavior on the system. Malware samples are identified by anti-malware engines using signature-based and heuristic-based methods. In signature-based detection, unknown sample is compared with the known malware samples that are stored in the database. Anti-malware databases have limited amount of malware family signatures and its features are stored. Only known malwares were identified using signature technique which limits its security against new kind of malwares. Nowadays, attackers are also using advanced techniques such as code obfuscation, encryption, packing, in malware programs to evade from malware detection. In heuristic-based detection, rules and algorithms are used to identify the malicious activity in the files. Initially, experts analyzed the various malware family samples for the generation of behavioral rules, which are used to detect unknown samples for the malicious activity. If unknown sample is scanned with the rules using heuristic-based detection and it satisfies any malicious intent, then immediately the sample is suspended

Table 1 Class distribution of the Microsoft malware classification

Malware family	No of samples
Gatak	1013
Lollipop	2478
Kelihos_ver1	398
Tracur	751
Simda	42
Vundo	475
Kelihos_ver3	2942
Obfuscator.ACY	1228
Ramnit	1541

from the execution. Generation of rules and algorithms by domain experts requires functionality of malware which is complex and time-consuming process.

3.1 Microsoft Malware Classification Challenge

Decade ago, research communities had fewer motivation to work on the malware analysis, due to lack of labeled datasets when compared with other applications. On the other side, malware threats were increasing in the world. In 2015, Microsoft provided a labeled malware dataset for the challenge hosted on Kaggle. It became the benchmark dataset for the evaluation of machine learning algorithms in the task of malware classification. The dataset has 10,868 malware samples constituted 9 different malware families. The nine different malware families are as follows: (1) GATAK, (2) LOLLIPOP, (3) KELIHOS_VER1, (4) TRACUR, (5) SIMDA, (6) VUNDO, (7) KELIHOS_VER3, (8) OBFUSCATOR.ACY and (9) RAMNIT. Table 1 represents the class distribution of Microsoft malware dataset.

3.2 Bytes File

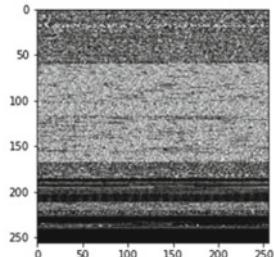
The dataset is provided with bytes file and assembly code file for each malware sample. Bytes file contains hexadecimal representation of the malware binary content. It contains the opcodes such as add, sub, mul, div, jump statements, control statements, function calls, text, images, represented in the hexadecimal format. Analysis on byte files can be performed in different ways such as extracting byte n-grams, finding the structural entropy of the executable and converting bytes file to gray-scale image discussed in Sect. 2. In this model, gray-scale images were used to perform malware classification. Figure 1 shows the hex view of malware sample and its gray-scale image. The first value in the hex file gives the starting address of the machine

```

00401000 68 18 ED 42 Q0 FF 15 70 60 42 00 68 28 ED 42 00
00401010 50 C3 E4 50 42 Q0 FF 74 24 08 00 50 C3 E4 50 42 00
00401020 D0 C3 C7 00 3C ED 42 00 C3 53 8B D8 00 06 57 B5
00401030 C0 74 07 50 88 4E 07 00 00 59 85 D8 75 12 88 44
00401040 24 0C 8D 50 01 8A 08 40 84 C9 75 F9 28 C2 88 D8
00401050 80 78 01 57 E3 BE 09 00 00 57 8A 00 50 89 06 E8
00401060 80 78 01 57 E3 BE 09 00 00 57 8A 00 50 89 06 E8
00401070 83 C4 1C 5F 53 C2 04 00 3B 00 04 11 43 00 00 75 02
00401080 F3 C3 E9 EO 13 00 00 BB FF 55 8B EC 51 S1 88 45
00401090 OC 56 8B 75 00 00 89 45 F8 8B 45 10 57 56 89 45 FC
004010A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04
004010B0 00 00 C7 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004010C0 14 8D 4D FC 51 FF 75 F8 50 FF 15 F4 80 42 00 89
004010D0 45 FB 3B C7 75 13 FF 15 SC 80 42 00 85 C0 74 09
004010E0 80 50 41 00 83 E6 1F C1 E6 06 80 44 30 04 80
004010F0 85 80 50 41 00 83 E6 1F C1 E6 06 80 44 30 04 80
00401100 20 FD 8B 45 FC 88 55 FC 56 C9 C3 6A 14 6A 00
00401110 EF 42 00 E8 CC 19 00 00 83 CE FF 89 75 DC 89 75
00401120 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00401130 00 E8 19 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00401140 E9 D0 00 00 33 FF 3B C7 7C 08 3B 05 64 5D 43
00401150 00 72 21 E8 04 00 00 00 89 3B E8 F0 03 00 00 00 C7
00401160 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00401170 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00401180 8B F0 83 E6 1F C1 E6 06 88 00 0F BE 4C 31 04 83
00401190 E1 01 75 26 E8 C9 03 00 00 57 57 57 57 E8 0C 19 00 00
004011A0 C7 00 09 00 00 00 00 00 57 57 57 57 E8 0C 19 00 00
004011B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004011C0 59 89 7D FC 88 03 F6 44 30 04 01 74 1C FF 75 14

```

(a)



(b)

Fig. 1 **a** Few lines in hexadecimal representation of sample malware file, **b** displays the gray scale image of malware file in **(a)**

code in the memory and each hexadecimal value contains information extracted from portable executable file such as opcodes, function calls, data.

3.3 Deep Learning for Malware Classification

Traditional machine learning models require manual feature extraction from the data. Feature engineering takes most of the computation time. Sometimes it impacts the performance of the model, due to lack of certain knowledge on acquiring the features. Deep learning overcomes these problems with independence of human intervention. By considering the malware image files, convolutional neural network (CNN) model was used for malware classification. CNN has multilayer perceptrons where each neuron in one layer of the network is connected to all the neurons in the next layer. There is a chance of overfitting of data in these networks which can be prevented by trimming the connections and dropout methods. By observing the various research methods discussed in the related work section, deep learning models outperform mostly, when compared with traditional models.

4 malC Architecture

An overview of our novel deep convolution neural network (CNN) architecture, malC model is shown in Fig. 2. Generally, computer vision applications such as image classification, medical image analysis use CNN to classify the data. Deep convolutional network has been implemented to extract features of obfuscated malware efficiently and give significant performance with other malware image datasets also. Initially, the bytes files with hex values are converted into gray-scale image pixels. While

giving the input to the first layer of CNN, all images should be of same size considered as 256×256 ($W \times H$) where W represents width, H represents height. Image size impacts the model accuracy and computational memory. Most of the papers discussed in the related work considered image sizes as 128×128 , 224×224 . Sometimes reduction of image size loses the important features to classify the data. In this model, 256×256 image size was considered for the improvement of accuracy.

Following the input layer, there is a stage wise extraction of features through convolution, activation, pooling and dropout layers. There are two functions in the convolution layer which follows the input layer. Convolution function is the one which performs the mathematical operation to extract the features. There is an element called kernel/filter applied on the input to get the feature map. For each convolution operation, the number of features it extracts is based on the number of kernels given. In the malC architecture, convolution operation is applied totally six times to get the best features for the classification. By considering the gray-scale image, convolution operation is performed with 64 kernels of size 3×3 . Figure 3 shows eight of the 3×3 filters applied in the first convolution layer of the model. Another parameter called padding is used in the operation which is set to ‘same.’ Same padding defines that output size of the convolution operation gives same size as input. ReLU activation function is used in the convolution layer which gives the non-linear nature to output data. ReLU [18] function is applied to all the convolution layers in the neural network. ReLU is Rectified Linear Unit, which has function $y(x) = \max(x, 0)$.

Convolution layer gives feature map as the output. Feature map is obtained by sliding the input matrix over kernel matrix and represents certain kind of features present in the image. Figure 4 shows sample feature maps obtained in the final convolution layer. To perform down-sampling on the feature map, pooling technique is used. Among the available pooling methods such as max pooling, global average pooling, max pooling is used in this model with pool filter size of 2×2 and stride 2. With these parameter values, input size is reduced by half. For example, if the input size is 128×128 , the max pooling operation reduces its size to 64×64 . This operation reduces the computation time for processing and memory used.

In the process of learning various features from the dataset, sometimes the model suffers with overfitting problem. i.e., model does not perform well with the test dataset. A technique called Dropout [19] refers to the removing of some neurons from the neural network, which significantly reduces the overfitting problem. In the malC neural network, dropout operation reduces 20% of neurons which uses twice in the layered architecture.

After performing the function of final convolution or pooling layer, the output feature maps converted as flattened i.e., one-dimensional array. This layer also called as dense layer, which takes input and connected to every output neuron with a learnable weight. Two fully connected layers are used in the neural network. Every fully connected layer output performs a non-linear operation. Based on the task in the neural network, activation function is selected. ReLU activation function is performed after first fully connected layer. Final fully connected layer uses SoftMax activation

Fig. 2 Deep CNN architecture of malC model

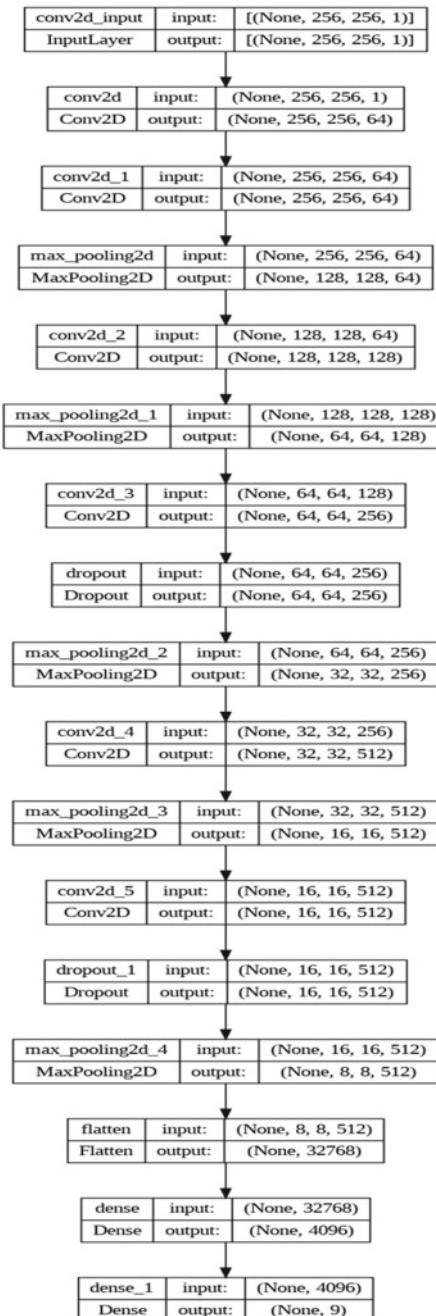




Fig. 3 Starting eight 3×3 filters in the first convolutional layer

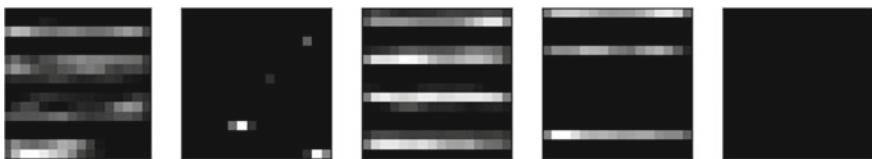


Fig. 4 Selected images of feature maps obtained in the final convolution layer

function for the multiclass classification by normalizing the real values from the layer, where every value belongs to $[0, 1]$ and sum of all values as 1.

4.1 Optimization Using Keras

Keras is an artificial neural network library used in Python to implement machine learning or deep learning models. Hyperparameter tuning is performed using Keras to minimize the loss and train the model to predict the data correctly. The loss function is used to assess whether the model goes into right direction or wrong direction. Based on the loss function value, optimizers adjust the weights of the neurons in the model. Among all the optimizers available, Adam optimizer was used in the malC model. When compared with other optimizers, Adam optimizer gives better performance with less computation time and considering few parameters for tuning. The malware dataset contains nine classes which come under multiclass classification. All the images in the dataset are represented as integer data with values between 0 and 255. These type of multiclass classification problems with integer data uses sparse categorical cross entropy as loss function.

4.2 Comparison with VGG Model

Malware classification is initially performed using VGG [20] network containing 16 layers in it. Due to more layers present in the network, it takes huge amount of time to train the parameters and perform classification. Also, a greater number of filters have been used to extract features from the data. Due to large set of feature extraction, more memory is used by the model. These issues arise due to more layers present in the model. Reducing the computation time and memory usage was done

by our malC model discussed above. VGG model uses 13 convolution layers and three dense layers while the malC model contains six convolution layers and two dense layers. Malware dataset was applied on the malC model which gives better accuracy when compared with the VGG model.

5 Evaluation

This section provides the experimental evaluation of the malC model in the dataset provided by the Kaggle Microsoft Malware Classification Challenge. To build the model, total dataset was divided as train set and test set with 75% and 25% respectively. Logloss and accuracy were the two-performance metrics considered to analyze the model.

5.1 Performance Metrics

For balanced datasets, accuracy is considered as the best metric to get the performance of the model. Whereas for imbalanced datasets, accuracy score is not enough as performance metric. This is because some wrong predictions of classes containing only few samples do not impact the accuracy score. The model also went to overfitting situation. Formally accuracy is defined as follows:

$$\text{accuracy} = \frac{\text{number of correct predictions}}{\text{total number of predictions}}.$$

Logloss is one of the important metrics which assess the performance of classification problem. It gives the closeness of predicted state and target state. As we are doing multiclass classification, `sparse_categorical_crossentropy` loss function used. Logloss is defined as follows:

$$\text{logloss} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^M y_{i,j} \log(p_{i,j}),$$

where N represents number of samples, M represents number of class labels, $y_{i,j}$ is 1 if the sample i belongs to class j and 0 otherwise, and $p_{i,j}$ is the prediction probability that observation i is in class j .

Furthermore, additional evaluation metrics such as precision, recall and $F1$ score of each malware family have been used shown in Table 1. The formulae of precision, recall and $F1$ score is given as follows:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}},$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}},$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}},$$

where True Positive gives that the model predicts positive samples correctly. False Positive gives that the model predicts positive samples incorrectly. False Negative gives that the model predicts negative samples incorrectly.

The confusion matrix and the normalized confusion matrix achieved by malC model is shown in Fig. 5. Confusion matrix represents the row-column representation of classified malware samples, where each row represents true label and each column represents predicted label. By observing the Simda malware family with 42 samples, 38 are correctly classified and 4 are incorrectly classified with 90% accuracy of this class. It doesn't show the significant impact on the accuracy metric of the model. In this case, recall gives the variation with exact classification rate of classes as shown in Table 2.

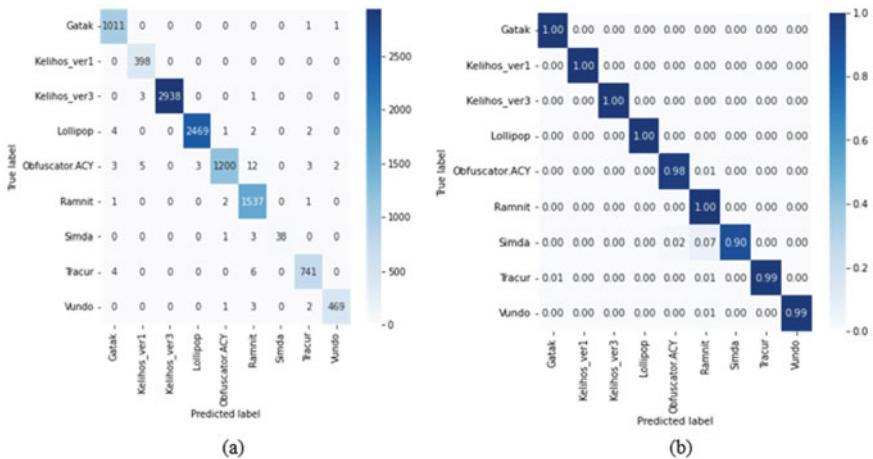


Fig. 5 Confusion matrices **a** without normalization and **b** with normalization achieved by the malC model

Table 2 Precision, recall and $F1$ score of each malware family

Malware family	Number of samples	Precision	Recall	$F1$ score
Gatak	1013	0.99	1.00	0.99
Kelihos_ver1	398	0.98	1.00	0.99
Kelihos_ver3	2942	1.00	1.00	1.00
Lollipop	2478	1.00	1.00	1.00
Obfuscator.ACY	1228	1.00	0.98	0.99
Ramnit	1541	0.98	1.00	0.99
Simda	42	1.00	0.90	0.95
Tracur	751	0.99	0.99	0.99
Vundo	475	0.99	0.99	0.99

Table 3 Byte file approaches comparison with malC model

Method	Accuracy	Macro $F1$ score
PCA + K-NN [6]	0.9660	0.9102
CNN IMG [9]	0.975	0.940
CNN ENTROPY [8]	0.9828	0.9636
Deep neural network using auto encoders	0.9915	—
Proposed malC model	0.993	0.99

5.2 Comparison with the State-of-the-Art

The performance of malC model has been compared with the state-of-the-art methods discussed in the literature on the benchmark dataset by Microsoft Malware Classification Challenge shown in Table 3. Specifically, byte files approaches has been considered for the comparison. Our proposed deep CNN model outperforms with overall accuracy of 0.993 and macro $F1$ score of 0.99. To measure performance, in addition to accuracy, macro $F1$ score has been used which gives equal importance to each class for the unbalanced datasets. The model gives good performance whenever test data with imbalanced data samples is fed. Our proposed model gives better macro $F1$ score when compared with models specified in Table 3.

6 Conclusion and Future Work

This paper presents the deep CNN architecture, which uses gray-scale images for the classification of malware samples. Convolutional networks extract the high dimensional features, when compared with traditional methods. Obfuscated malware samples which use encryption, masking of code are also classified efficiently using

CNN models. Advantage of deep learning is the usage of minimal computation time for data preprocessing. As per the study on the related work and observation from the malC model, malware samples, adhering the structure of byte files, will use deep learning techniques for getting the better results. Our model outperforms all the bytes file approaches with accuracy of 0.993 and logloss of 0.0021. A future direction of research work is to accelerate the model by reducing the computational time at initial neural network layers using transfer learning. Generating the multimodal architectures from multiple feature patterns also the research insight for future.

References

1. https://en.wikipedia.org/wiki/2022_Ukraine_cyberattacks
2. <https://www.comparitech.com/antivirus/malware-statistics-facts/#:~:text=In%202020%2C%2061%20percent%20of,SOES%20survey%20began%20in%202016>
3. <https://www.av-test.org/en/statistics/malware/>
4. Ronen R, Radu M, Feuerstein C, Yom-Tov E, Ahmadi M (2018) Microsoft malware classification challenge. arXiv e-prints, Feb 2018
5. Yamashita R, Nishio M, Do RKG et al (2018) Convolutional neural networks: an overview and application in radiology. Insights Imaging 9:611–629. <https://doi.org/10.1007/s13244-018-0639-9>
6. Narayanan BN, Djaneye-Boundjou O, Kebede TM (2016) Performance analysis of machine learning and pattern recognition algorithms for malware classification. In: 2016 IEEE national aerospace and electronics conference (NAECON) and Ohio innovation summit (OIS), July 2016, pp 338–342
7. Kebede TM, Djaneye-Boundjou O, Narayanan BN, Ralescu A, Kapp D (2017) Classification of malware programs using autoencoders based deep learning architecture and its application to the Microsoft malware classification challenge (BIG 2015) dataset. In: 2017 IEEE national aerospace and electronics conference (NAECON), Dayton, OH, pp 70–75. <https://doi.org/10.1109/NAECON.2017.8268747>
8. Gibert D, Mateu C, Planes J, Vicens R (2018) Classification of malware by using structural entropy on convolutional neural networks. In: Proceedings of the thirty-second AAAI conference on artificial intelligence (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, LA, 2–7 Feb 2018, pp 7759–7764
9. Gibert D, Mateu C, Planes J, Vicens R (2018) Using convolutional neural networks for classification of malware represented as images. J Comput Virol Hack Tech
10. Singh J, Thakur D, Gera T, Shah B, Abuhmed T, Ali F (2021) Classification and analysis of android malware images using feature fusion technique. IEEE Access 9:90102–90117. <https://doi.org/10.1109/ACCESS.2021.3090998>
11. Xiao G, Li J, Chen Y, Li K (2020) MalFCS: an effective malware classification framework with automated feature extraction based on deep convolutional neural networks. J Parallel Distrib Comput 141:49–58
12. Zhai J, Zhang S, Chen J, He Q (2018) Autoencoder and its various variants. In: 2018 IEEE international conference on systems, man, and cybernetics (SMC), Miyazaki, Japan, pp 415–419. <https://doi.org/10.1109/SMC.2018.00080>
13. Lyda R, Hamrock J (2007) Using entropy analysis to find encrypted and packed malware. IEEE Secur Priv 5(2):40–45. <https://doi.org/10.1109/MSP.2007.48>
14. Khan M, Baig D, Khan US, Karim A (2020) Malware classification framework using convolutional neural network. In: 2020 international conference on cyber warfare and security (ICCWS), Islamabad, Pakistan, pp 1–7. <https://doi.org/10.1109/ICCWS48432.2020.9292384>

15. Alam M, Akram A, Saeed T, Arshad S (2021) DeepMalware: a deep learning based malware images classification. In: 2021 international conference on cyber warfare and security (ICCWS), Islamabad, Pakistan, pp 93–99. <https://doi.org/10.1109/ICCWS53234.2021.9703021>
16. Depuru S, Santhi K, Amala K, Sakthivel M, Sivanantham S, Akshaya V (2023) Deep learning-based malware classification methodology of comprehensive study. In: 2023 international conference on sustainable computing and data communication systems (ICSCDS), Erode, India, pp 322–328. <https://doi.org/10.1109/ICSCDS56580.2023.10105027>
17. Shinde S, Dhotarkar A, Pajankar D, Dhone K, Babar S (2023) Malware detection using EfficientNet. In: 2023 international conference on emerging smart computing and informatics (ESCI), Pune, India, pp 1–6. <https://doi.org/10.1109/ESCI56872.2023.10099693>
18. Nair V, Hinton GE (2010) Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th international conference on machine learning. Available online at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.6419&rep=rep1&type=pdf>. Accessed 23 Jan 2018
19. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15:1929–1958
20. Rezende E et al (2018) Malicious software classification using VGG16 deep neural network's bottleneck features. In: Information technology—new generations, pp 51–59

Cyber Threat Intelligence (CTI): An Analysis on the Use of Artificial Intelligence and Machine Learning to Identify Cyber Hazards



Neelima Kant and Amrita

Abstract Data transport volume and scope on networks are growing daily due to the quick advancements in network technology. It is challenging for cybersecurity specialists to keep track of every action taking place on the network because of the constantly growing density of networks. This circumstance has led to an increase in the complexity and intensity of threats and attacks. It is harder to detect and identify irregularities in network activities because of frequent and sophisticated cyberattacks. A well-crafted cybersecurity strategy now includes cyber threat intelligence (CTI), which is a crucial foundation. Automating the detection of cyberattacks as well as speedy attack type analysis and predication are all made possible by machine learning (ML), which offers a number of tools and techniques. The strategies for using machine learning (ML) to identify assaults are discussed in this article. Threat intelligence can help security teams defend against a constantly evolving threat environment before, during, and after an attack if used properly. By analyzing attackers and comprehending their tactics and goals, groups may create cyber defenses that are more effective, delicate, and resilient. However, due to two significant flaws, its usefulness is still in question. First, current methods are unable to detect unknown Indicator of Compromise (IoC), and second, they are unable to automatically produce categorized CTIs which renders CTI sharing. As a result, the objective of this paper is to present a complete analysis of cyber threat identification using intelligent techniques. Additionally, we covered the issues and solutions related to machine learning applications utilized in network assaults.

Keywords Cyber threat intelligence (CTI) · Machine learning (ML) · Indicator of Compromise (IoC) · Advanced persistent threats (APT) · Vulnerability assessment · Internet of Things (IoT) · Artificial intelligence (AI)

N. Kant (✉)

Department of Computer Science and Engineering, School of Engineering and Technology,
Sharda University, Greater Noida, Uttar Pradesh 201306, India
e-mail: neelimakant.spm@gmail.com

Amrita

Department of Computer Science and Engineering, School of Engineering and Technology,
Center for Cyber Security and Cryptology, Sharda University, Greater Noida, Uttar
Pradesh 201306, India

1 Introduction

Every day, more and more people's social lives and habits are being influenced by the Internet. The Internet's significance in society is progressively growing as a result of globalization. The Internet is interwoven with crucial governmental infrastructure, and it is quickly emerging as one of the most significant engines of socioeconomic development. The Internet's developing and deeper structure exposes us to new dangers, whose variety is continually expanding. One of the most crucial problems in modern cybersecurity is how these risks might be found in network traffic [1]. Cyber threat intelligence (CTI), which includes assault type forecasting and analysis, is made possible by the use of intelligent techniques such as machine learning (ML) and artificial intelligence (AI), which provide a number of tools and techniques for automated detection of cyberattacks.

CTI has several definitions. "Evidence-based knowledge, including context, mechanisms, indicators, implications, and actionable advice about an existing or emerging menace or hazard to assets that can be used to inform decisions regarding the subject's response to that menace or hazard" is an example of what CTI is referred to as by the definition given in [2]. "The set of data collected, assessed, and applied regarding security threats, threat actors, exploits, malware, vulnerabilities, and compromise indicators" is what CTI is described as in [3]. "Data that has been refined, analyzed, or processed such that it is relevant, usable, and valuable" is how McMillan [4] define CTI. In general, the CTI pipeline receives unprocessed cybersecurity data as input, and it produces data that may be used for goal-oriented strategies for cybersecurity defense that involves strategies to limit scope as well as thwarting cyberattacks.

Insider threats, social engineering, malware, application vulnerabilities, incorrect setup, weak or stolen credentials, and user mistake are typical origins of data breaches. Threat intelligence is a development in the safeguarding of infrastructure, data, and documents [5]. Information collecting from many sources has advanced due to the sophistication of assaults and the need to protect environment assets. For coping with cyberattacks in the new age [6], the exchange of threat intelligence offers a major help. CTI is process-oriented, opponent-based, risk-focused, and designed for a variety of customers. Because it focuses on figuring out the goals and intents of the attackers in addition to their tactics, it is adversary-based.

A new line of security defenses must be established to protect from the serious and prevalent cyber assaults that are occurring today. Due to their constant evolution and characteristics of fraud, durability, and intricacy as well, new-generation assaults are challenging for conventional protection mechanisms that utilize heuristics and identities to remain up with. In order to avoid attacks or, at the absolute least, promptly and pro-actively respond to them, organizations seek to acquire and exchange real-time cyber threat information [7]. CTI is concentrated on reducing the risk that critical infrastructure and assets are exposed to due to the actions of the attackers so that business operations are unaffected. CTI seeks to provide cyber threat actors with a knowledge advantage [3].

We described each study's datasets, ML techniques, performance metrics, and cybersecurity focus in this report. We looked at the feature selection or dimension reduction techniques employed on the study's datasets. We discussed the several categories utilized in these researches, the methods in comparison with other strategies, and the performance indicators employed in great detail. The open-access datasets related to network attacks were reviewed. The issues that machine learning algorithms utilized in network assaults run into were also explored, along with potential remedies.

2 Work Reported and Their Analysis

In the past several years, numerous studies have been conducted utilizing various AI and ML approaches to look at signs of cyberattacks, malware analysis, and anomaly detection methods. As most companies transitioned from on-premise infrastructure to cloud architecture, many network components are being virtualized. Our civilizations are predicted to use larger sensor-based structures more frequently, demanding the creation of innovative methods for their construction and operation. The variety of IoT devices and sensors has greatly increased as a result of IoT technology innovation.

For the comprehensive literature study of CTI and use of advanced technologies in vulnerability analysis and penetration testing, Xiong and Lagerstrom [8] and McKinnel et al. [9] employed 6 main databases: Scopus, IEEE Explore, Science Direct, Web of Science, ACM Digital Library, and Google Scholar. The study offers a wide range of opportunities and difficulties in employing AI and ML approaches in cybersecurity.

- Direct use of AI and ML for penetration testing and vulnerability analysis.
- Different approaches and strategies were used in various study publications to analyze real data about false positive rates.
- It also includes a few of the different researchers' ad hoc AI and ML applications.
- For the meta-analysis and literature evaluation, peer-reviewed conferences and journal articles were used.

The next-generation assaults are more difficult to manage than older attacks, according to Tounsi and Rais [10]. For the protection of computers against new dangers, a boundary must be established.

There is also a change in the Indicator of Compromise (IoC), since industry is already moving toward cloudification. Author claims that as far as limitations go, to guard against zero-day attacks, timely information acquisition is necessary. However, quick IoC dispersal alone is insufficient to counteract targeted attacks. For that, it is required to refine the threat information based on its inherent defects and weaknesses.

Using the framework for intelligence on cyber threats which Mavroeidis and Bromander [11] created in light of the evolving CTI environment, cyber warriors may assess their level of threat intelligence capabilities and their place against the constantly changing CTI landscaping.

By continually enhancing the model's precision and proving their resistance against machine learning assaults, such ML algorithms can reduce the likelihood of fake positives and incorrect results when there is abundant supporting data. The strategy presented by Preuveneers and Joosen [12] was built on the Cortex, Hive, and Malware Information Sharing Platform (MISP), three leading-edge freeware CTI exchange as well as response to incidents.

Ramsdale et al. [13] investigated a number of available codes and languages in addition to openly available attack streams to figure out if these were acceptable for assessment as well as supplying full cyber assaults data.

Mahbub [14] and Mohanta et al. [15] identifying issues layer by layer and protecting data's Confidentiality, Integrity, and Availability (CIA) are the major challenges. Author discovers the potential applications of blockchain, artificial intelligence, and machine learning to address security issues. With these technologies and Internet of Things (IoT), security would be increased.

Cybersecurity and forensic specialists are needed in today's environment when cyberattacks are on the rise and must be able to identify, assess, and safeguard against real-time cyber threats. Deep analytical capabilities are necessary for this timely identification of threats and weaknesses as well as for improvement in the field of cyber threat intelligence.

To analyze the digital traces of cyberattacks, we need intelligent technologies that use AI, ML, and data mining approaches. The primary difficulties in this field, according to author Conti et al. [16], are:

- **Reconnaissance of Attack Vectors.** Identify the attack's starting point and any weaknesses that might be used by hackers or online attackers.
- **Reconnaissance of Attack Indicator.** Due to the rise in cyberattacks, hackers and attackers utilize evasion and anti-forensic techniques in their harmful code to lower the effectiveness of the Common Vulnerability Scoring System (CVSS) score.

Cybersecurity experts can examine their ability to detect threat intelligence strengths and their place in the ever-evolving threat intelligence environment applying the CTI approach, in accordance with Mavroeidis and Bromander's [11] guidelines. Using this method, the authors also look at taxonomies, ontologies, and sharing standards. The results demonstrate that the threat intelligence industry as a whole is not supplying the appropriate ontologies.

The CTI model serves as a visual depiction of the many sorts of data needed for advanced threat intelligence and attack attribution. By employing investigative and preventative measures, it strengthens the organization's security posture. The absence of a standardized method to represent critical data as a result from the terminology's inconsistency, in the view of the writer, constitutes one among the main problems of CTI. This leads to unnecessary confusion among experts and adds to the workload required to create ontologies. Additionally, layers of abstraction are also not clearly represented.

The conclusion drawn by the writer is that there cannot be an ontology which can be used along with CTI. The biggest issue is the absence of explanations for the

conclusions that can be attributed by insufficient development or levels of abstraction that do not have the necessary data as well as knowledge to execute successful cyber threat intelligence [17]. It argues that the cybersecurity mandate should be centered on cyber intelligence, necessitating a prepared security strategy, an exhaustive comprehension of the climate of threat, a commitment toward conclusions based on data and other factors.

After the hostile actor has been present in the protected network for long enough to establish a recognizable pattern of behavior, the Cyber Kill Chain starts (see Fig. 1). Although this idea is useful in the case of a known assault, network defense's main objective is to stop incursions before they start. In order to achieve this, defenders must expand their understanding of the kill chain or assault pathway beyond what happens on the network. Intelligence efforts might be focused on seeing a series of patterns and actions to determine:

- Who might be trying to break into a network?
- What were the objectives and capabilities of the bad actors?
- When would they finish their task?
- How and from where will the action start?
- In what way will they utilize or modify the computer network?

At this stage, threat analysis involves studying how hostile actors set up their assaults using knowledge they gained from gathering their own personally identifiable information as well as what they feel is required to achieve their strategic objectives [17]. To support their tactical operations, attackers need to construct the infrastructure (which involves malware, botnets, and delivery techniques that include phishing). They allocate their assets to the locations in cyberspace also known as "hop points" in which they are going to carry out the objectives they have established. A hacktivist group, for example, might plan events in both physical and virtual realms

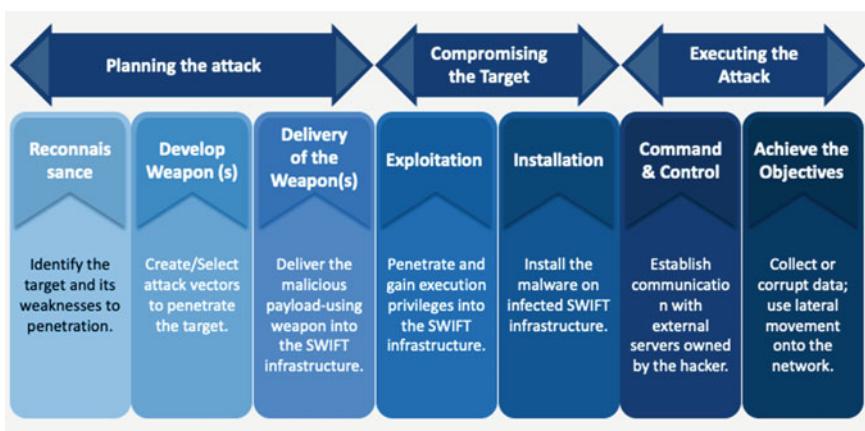


Fig. 1 Cyber Kill Chain activities

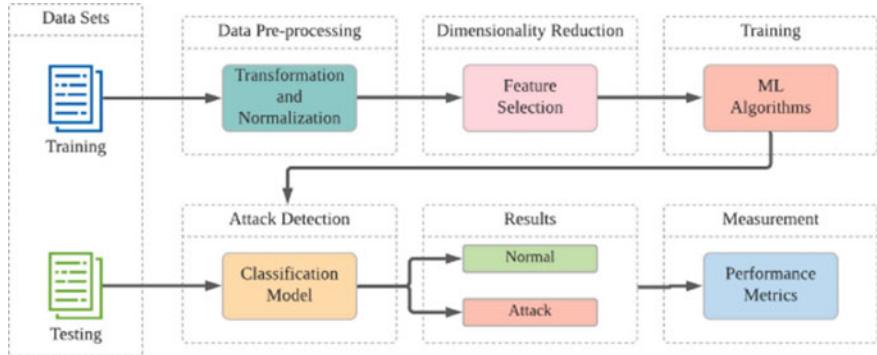


Fig. 2 Flowchart for attack detection

to further its operational goals (see Fig. 2). Here are a few examples of operational intelligence:

- According to trend analysis, an adversary's capabilities are moving in a technological direction.
- Signs indicating a rival has chosen a plan of attack against an organization.
- Signs that a potential assailant is learning how to use a certain tactic.
- The disclosure of adversary activities, tactics, and methods.
- Being aware of the opponent's operating cycle, including its decision-making, acquisitions, and command and control (C2) strategies for both technology and people.
- An enemy's technical, social, legal, financial, and other weaknesses.
- Knowledge that enables the defender to influence adversaries as they move toward their assault (by using the kill chain).

The “Cyber Prep” matrix, created by the MITRE Corporation, correlates danger levels with various levels of cyber preparedness [17]. They suggest five cyber threat levels, each of which equates to a significant cyber defensive stance or plan, in order to enable mission assurance:

- **Cyber Vandalism**, which falls under Threat Level 1, is equivalent to Perimeter defense.
- **Cyber Theft/Crime**, Threat Level 2, which is correlated to a Critical Information Protection security plan.
- **Cyber Intrusion/Surveillance**, Threat Level 3, which is correlated to a Responsive Awareness defense plan.
- **Cyber Sabotage/Espionage**, which is classified as a Threat Level 4 and requires a defense plan based on architectural resilience.
- **Cyber Conflict/Warfare**, Threat Level 5, which is defended against using Pervasive Agility.

3 Tools for Cyber Threat Intelligence

Few premium tools to evaluate threat intelligence [12] are listed as:

The Hive (Security Incident Response Plan—SIRP): In order to make everyday life simpler for Computer Security Incident Response Teams (CSIRTs), Computer Emergency Response Teams (CERTs), Security Operation Center's (SOCs), as well as any other type of security professional who handles security issues that require to be immediately examined and handled, an adaptable, freely available, and free of charge security-related incident action platform which is tightly coupled with MISP has been developed.

MISP (Threat Sharing Platform): A platform for sharing threat intelligence is called the MISP. By exchanging signs of compromise, the initiative creates tools and documentation for more efficient threat intelligence. The purpose of MISP is to collect both technical and non-technical data on malware and invasion, to store the data in a systematic manner, and to disseminate malware analysis to reliable parties.

OASIS (Organization for the Advancement of Structured Information Standards): A collection of information representations and protocols is defined by the OASIS to meet the requirement to model, evaluate, and distribute cyber threat intelligence. The OASIS has developed a set of information representations and protocols to fulfill the need to model, assess, and disseminate cyber threat intelligence. There are specifications that will be transferred for development and standardization using the OASIS open standards approach.

- *Structured Threat Information Expression (STIX)*, Cyber threat intelligence is exchanged using the language and serialization format known as STIX. With the use of objects and descriptive connections, STIX allows for the precise representation of all aspects of suspicion, compromise, and attribution.
- *Trusted Automated Exchange of Indicator Information (TAXII)* specifies how message exchanges and services may be used to convey information about cyber threats.

Collaborative Research into Threats (CRITs): CRITs bring together cyber threat information gathered from single, frequently dispersed attacks and represents this data in standardized formats to facilitate analysis and information sharing. It was originally developed as part of research on how to protect the systems of Massachusetts Institute of Technology Research and Engineering (MITRE). An organization's network may then be safeguarded against potential assaults using the information.

Soltra Edge: The OASIS CTI Hive Project, which unifies the three standards. The goal of this platform, which is partially commercial, is to create a forum-based protective model with a wide range of inter-operative capabilities through the collaboration of two industry standards, STIX and TAXII.

4 Data Sources in Cybersecurity of CTI Analysis

Datasets in the field of cybersecurity that are made available with open access [18], separated by application areas including network traffic, electrical traffic, Internet traffic, virtual private network traffic, datasets based on IoT traffic, Android applications and devices attached to the Internet. Depending on the datasets on which it is based, it may be divided into a number of groupings [19]. Some of them are included in the list below.

- **AB-TRAP:** This framework is used to build invisibility shields to protect network devices using ML. It may be used to construct network intrusion detection systems (NIDS) [20]. It allows for the usage of updated network traffic and takes operational issues into account to enable the full deployment of the solution using ML. It is a five-step technique that involves (i) creating the attack dataset, (ii) creating the genuine dataset, (iii) training machine learning models, (iv) realizing the models, and (v) evaluating the performance of the realized model after deployment.
- **Elastic Malware Benchmark for Empowering Researchers (EMBER):** A benchmark dataset for academics, the EMBER dataset is a collection of characteristics from PE files. Both the EMBER2017 dataset and the EMBER2018 dataset contain features from one million PE files that were scanned in or before those years [21]. The EMBER2017 dataset had features from 1.1 million PE files. This repository makes it simple to extend the offered feature set, train the benchmark models in a reproducible manner, or use the benchmark models to categorize new PE files.
- **Defense Advanced Research Projects Agency (Darpa 98–99):** Network traffic and log records were used to build this data collection [22]. The dataset contains activity from email, scanning, File transfer Protocol (FTP), Telnet, Internet Relay Chat (IRC), and Simple Network Management Protocol (SNMP). Attacks such as Denial of Service (DoS), password guessing, buffer overflow, remote FTP, Synchronization Flooding (SYN flood), Distributed Denial of Service (DDoS), Network Mapper (Nmap), and Rootkit are included.
- **Knowledge Discovery in Databases (KDD) Cup99:** A seven-week period of network traffic monitoring was used to construct the database, which has around five million samples. In the dataset, there are 41 features. These attributes include traffic, content, and basic attributes [23]. The KDD'99 dataset and the Darpa'98–99 dataset are both immune to zero-day attacks.
- **Network Security Laboratory—Knowledge Discovery in Databases (NSL-KDD):** Models developed using NSL-KDD have been found to perform more efficiently than versions developed with older data sources. The data used for training collection's repeatedly collected samples especially had a negative impact on the trained machines' false positive and fake negative rates. It was previously noted that the investigators were capable of to acquire more trustworthy findings following removing these unfavorable components. As a remedy to the issues brought on by repetitive as well as redundant items within the data collection, KDD'99 has been recommended [24–26].

- **Canadian Institute of Cybersecurity-Distributed Denial of Service (CIC-DDoS2019):** A variety of contemporary reflective DDoS attacks are included in this dataset, including Network Basic Input/Output System (NetBIOS), SNMP, Lightweight Directory Access Protocol (LDAP), User Datagram Protocol (UDP), Microsoft SQL Server (MSSQL), UDP-Lag, Network Time Protocol (NTP), Domain Name Server (DNS), Synchronization (SYN), and Port Mapper (PortMap). The dataset for this dataset was developed by examining the abstract behaviors of 25 users based on the Hypertext Transfer Protocol (HTTP), Hypertext Transfer Protocol Secure (HTTPS), FTP, Secure Shell (SSH), and email protocols [27].

5 Machine Learning Techniques for Cyberattack Detection

Attack categorization, analysis, and detection are the three primary uses of machine learning algorithms in attack detection. Before being utilized in model training, training data undergoes a number of preprocessing steps (see Fig. 2). Data transformations and normalization are included in these processes. The model from this changed data must perform better than to the dimension reduction method. For this procedure, statistical dimension reduction techniques or feature selection algorithms are applied. After model training, test data is used to identify an attack. By contrasting predicted and actual results, multiple performance indicators are used to assess the model's effectiveness.

5.1 Classifier—*Naive Bayes (NB)*

In addition to the Bayes theorem, the Naive Bayes (NB) classifier is a supervised learning technique that makes a number of conditional independent assumptions on the characteristics. To arrive at an explicit probability for the hypothesis, it takes into account prior knowledge and observable evidence. NB has less impact by data-input noise than other conventional methods. The dimensions are assumed to be distinct from each other when it comes to the given targeted class features for its operation.

In the method they recommended, Dwivedi et al. [28] employed the grasshopper optimization algorithm to reduce size and compared the outcomes with other heuristic methods. For classification, the NB, decision tree (DT), support vector machine (SVM), and multilayer perceptron (MLP) models were applied. NB was utilized by Saleh et al. [29] to scale back the assault detection system they created. They recommended a modified version of the K-nearest neighbor (KNN) classification approach and used SVM that was optimized to take out the skewing influence of outliers on accurate identification. For the testing process, three different datasets were utilized. By combining Gaussian NB and principal component analysis (PCA), Zhang et al. [30] created a hybrid approach. They reduced data tampering by giving

feature vectors more importance. By reducing detection times, they performed cross-validation to assess the model's efficacy.

5.2 Classifier—Support Vector Machine (SVM)

With the use of the SVM, a type of vector space-based ML technique, it is possible to determine the judgment border among the two categories that stands furthest away from any location in the data used for training. When categorizing the information set sequentially is not feasible, it makes logical to move every set of data into its parent attribute region and categorize it using a hyperplane into the newly created space.

A unique clustering approach was put out by Borkar et al. [31] for attacks on Wireless Sensor Networks (WSNs). When the assault was initially detected among many sensor nodes, they were able to offer a secure packet transit using the classification model they constructed with SVM. The author improved the parameter choices and selecting features for SVM utilizing the hypergraph-genetic technique [32]. A unique approach for determining the kind of attack using a KNN-based algorithm was presented by Lin et al. [33]. They employed SVM in their research to evaluate performance and achieve compute efficiency for model training. A brand-new feature selection-based correlation-based dimension reduction approach was reported by Wang et al. [34]. They offered a hybrid that included the previously unknown feature groupings.

6 Performance Metrics

To accurately assess the efficacy of machine learning models, performance indicators must be interpreted appropriately [1]. These indicators are used to assess the model's level of influence. Various metrics are identified and created for attack detection to assess ML models. Some of them are defined below.

Precision (True Positive Rate—TPR)

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (1)$$

False Alarm Rate—FAR (False Positive Rate—FPR)

$$\text{FAR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (2)$$

Specificity (True Negative Rate—TNR)

$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (3)$$

False Negative Rate—FNR (Miss Rate—MR)

$$\text{MR} = \frac{\text{FN}}{\text{FN} + \text{TP}} \quad (4)$$

Recall (Sensitivity, Detection Rate—DR)

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (5)$$

Accuracy

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \quad (6)$$

F-Measure

$$F\text{-Measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

7 Challenges in CTI and Future Scope

Intelligent threat profiling using advanced techniques research is still in its early stages. The urgent need to solve the data, methodological, and application difficulties of intelligent threat profiling using advanced techniques is discussed in this section [35].

- The problem of massive-scale linkage between raw data, threat information, and security expertise has to be handled since this could easily result in a crisis of interdependence.
- Semantic combination, understanding of danger behavior, and threat argumentation are some of the study's shortcomings. Intelligent attack profiling solutions face a new cognitive and management problem since the requirements for the sophisticated persistent threat (APT) defense domain differ significantly compared to those in various other defense domains.
- The main obstacles to the broad adoption of threat analysis are the inappropriateness of assessing models developed on an overall domain-based corpus to specialized security domains, the challenge of assigning APT attacks due to unidentified assaults and the utilization of customized weapons, and the lack of groundbreaking

studies to discover and recognize attack behavior trends in different APT attack situations.

Security in cyber-physical contexts is increasingly important as our reliance on network devices grows [36]. Machine learning approaches and intrusion detection systems, such as rule- and signature-based intrusion detection systems, have improved cybersecurity. The classification skills of deep learning-based detection systems were improved along with the growth in the amount of labeled data [37]. The papers under consideration cover both shallow and deep learning techniques. The effectiveness of deep learning techniques is exceptional. Research on applications of special attack type has increased as a result of growing attack variety.

As the hostile cyberattack and defense situation keeps growing and changing, attack and defense strategies urgently need to work closely together with artificial intelligence and machine learning techniques in order to collaborate toward cognitive intelligence. Automation and threat profiling intelligence are unavoidable tendencies in the development of intelligent cybersecurity systems. Intelligent threat profiling is a prerequisite for intelligent security on attack scenario reconstruction, attack behavior prediction, attack situation awareness, and attack purpose reasoning. Examining the current problems reveals that developing and altering defensive concepts and practices is where intelligent threat profiling in APT defense is most likely to be successful.

8 Conclusion

The features of the datasets affect how well the current machine learning algorithms for cybersecurity work. The attack diversity of today's must be correctly reflected in datasets. Existing datasets must be updated often for this. In this method, intrusion detection will be automated and the models' training will be updated continuously. These technologies will advance when more hybrid approaches are used, as opposed to using the same classifiers to various threats. Machine learning techniques are now a crucial part of cybersecurity as a result of these results, but it is necessary to be aware of their limitations. Despite the fact that artificial intelligence technologies have increased the autonomy of detections, it still does not seem possible to ensure comprehensive security without human supervision.

This plan's overriding objective is to assist guarantee that overall cyber resilience in CTI should be maintained, even while intelligent technologies such as AI and ML give new features and advantages to CTI. This strategy will provide a range of tools for enhancing CTI methods that are also applicable to a number of other domains, such as network security, cloud security, and privacy concerns. In the conclusion, this study provides an overview, a fair evaluation of the role AI and ML play in CTI, and a road map for the future.

References

1. Huseyin A, Resul D (2022) A comprehensive review on detection of cyber-attacks: data sets, methods, challenges, and future research directions. *Internet Things* 20:100615
2. Gros S (2020) Research directions in cyber threat intelligence. arXiv preprint [arXiv:2001.06616](https://arxiv.org/abs/2001.06616)
3. Oosthoek K, Doerr C (2021) Cyber threat intelligence: a product without a process? *Int J Intell Counter Intel* 34(2):300–315
4. McMillan R. Definition: threat intelligence. In: Gartner.com. Accessed 10/11/2022
5. Du L, Fan Y, Zhang L, Wang L, Sun T (2020) A summary of the development of cyber security threat intelligence sharing. *Int J Digit Crime Forensics (IJDCF)* 12(4):54–67
6. Samtani S, Abate M, Benjamin V, Li W (2020) Cybersecurity as an industry: a cyber threat intelligence perspective. In: Holt T, Bossler A (eds) *The Palgrave handbook of international cybercrime and cyber deviance*. Palgrave Macmillan, Cham
7. Sun N, Ding M, Jiang J, Xu W, Mo X, Tai Y, Zhang J (2023) Cyber threat intelligence mining for proactive cybersecurity defense: a survey and new perspectives. *IEEE Commun Surv Tutor* 1
8. Xiong W, Lagerstrom R (2019) Threat modeling—a systematic literature review. *Comput Secur* 84:53–69
9. Mckinnel DR, Dargahi T, Dehghanianha A, Choo KR (2019) A systematic literature review and meta-analysis on artificial intelligence in vulnerability analysis and penetration testing. *Comput Electr Eng* 75:175–188
10. Tounsi W, Rais H (2017) A survey on technical threat intelligence in the age of sophisticated cyber attacks. *Comput Secur* 72:212–233
11. Mavroeidis V, Bromander S (2017) Cyber threat intelligence model: an evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. In: IEEE European intelligence and security informatics conference (EISIC), Athens, Greece, 11–13 Sept 2017, pp 91–98
12. Preuveeneers D, Joosen W (2021) Sharing machine learning models as indicators of compromise for cyber threat intelligence. *J Cybersecur Priv* 140–163
13. Ramsdale A, Shiaeles S, Kolokotronis N (2020) A comparative analysis of cyber-threat intelligence sources, formats and languages. *Electronics* 9(5):824
14. Mahbub M (2020) Progressive researches on IoT security: an exhaustive analysis from the perspective of protocols, vulnerabilities, and preemptive architectonics. *J Netw Comput Appl* 168:102761
15. Mohanta BK, Jena D, Satapathy U, Patnaik S (2020) Survey on IoT security: challenges and solution using machine learning, artificial intelligence and blockchain technology. *Internet Things* 100227
16. Conti M, Dargahi T, Dehghanianha A (2018) Cyber threat intelligence: challenges and opportunities. In: *Cyber threat intelligence*, pp 1–6
17. Mattern T, Felker J, Borum R, Bamford G (2019) Operational levels of cyber intelligence. *Int J Intell Counter Intel* 27(4):702–719
18. Ferrag MA, Maglaras L, Moschouyannis S, Janicke H (2020) Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study. *J Inf Secur Appl* 50
19. Ring M, Wunderlich S, Scheuring D, Landes D, Hotho A (2019) A survey of network-based intrusion detection data sets. *Comput Secur* 86:147–167
20. Bertoli DC, Pereira GJ, Alves L, Osamu S, Santos D, Aldri L, Alves F, Neto V, Cesar M, Cavalheiro A, Sidnei B, Rodrigues B, Moises S, Oliveira PD, José M (2021) An end-to-end framework for machine learning-based network intrusion detection system. *IEEE Access* 9:106790–106805
21. Anderson HS, Roth P (2018) EMBER: an open dataset for training static PE malware machine learning models. *Computer science—cryptography and security*. arXiv e-prints, 1804.04637

22. 1998 DARPA intrusion detection evaluation dataset | MIT Lincoln Laboratory (2020). URL: <https://www.ll.mit.edu/r-d/datasets/1998-darpa-intrusion-detection-evaluation-dataset>. [Online]. Accessed 5 Nov 2020
23. KDD cup 1999 data (2007). URL: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>. [Online]. Accessed 4 Nov 2020
24. NSL-KDD | datasets | research | Canadian institute for cybersecurity | UNB (2020). URL: <https://www.unb.ca/cic/datasets/nsl.html>. [Online]. Accessed 5 Nov 2020
25. Tavallaei M, Bagheri E, Lu W, Ghorbani AA (2009) A detailed analysis of the KDD cup 99 data set. In: IEEE symposium on computational intelligence for security and defense applications, CISDA 2009. <https://doi.org/10.1109/CISDA.2009.5356528>
26. C.f.A.I.D. analysis. CAIDA data—overview of datasets, monitors, and reports (2020). URL: <https://www.caida.org/data/overview>. [Online]. Accessed 6 Nov 2020
27. DDoS 2019 | datasets | research | Canadian institute for cybersecurity | UNB (2022) URL: <https://www.unb.ca/cic/datasets/ddos-2019.html>. [Online]. Accessed 14 May 2022
28. Dwivedi S, Vardhan M, Tripathi S (2020) Defense against distributed DoS attack detection by using intelligent evolutionary algorithm. *Int J Comput Appl* 44:1–11
29. Saleh AI, Talaat FM, Labib LM (2019) A hybrid intrusion detection system (HIDS) based on prioritized k-nearest neighbors and optimized SVM classifiers. *Artif Intell Rev* 51:403–443
30. Zhang B, Liu Z, Jia Y, Ren J, Zhao X (2018) Network intrusion detection method based on PCA and Bayes algorithm. *Secur Commun Netw*
31. Borkar GM, Patil LH, Dalgade D, Hutke A (2019) A novel clustering approach and adaptive SVM classifier for intrusion detection in WSN: a data mining concept. *Sustain Comput Inform Syst* 23:120–135
32. Raman M, Somu N, Kirthivasan K, Liscano R, Sriram VSS (2017) An efficient intrusion detection system based on hypergraph—genetic algorithm for parameter optimization and feature selection in support vector machine. *Knowl-Based Syst* 134:1–12
33. Lin WC, Ke SW, Tsai CF (2015) CANN: an intrusion detection system based on combining cluster centers and nearest neighbors. *Knowl-Based Syst* 78:13–21
34. Wang W, Du X, Wang N (2019) Building a cloud IDS using an efficient feature selection method and SVM. *IEEE Access* 7:1345–1354
35. Saurabh S, Pradip KS, Seo Yeon M, Daesung M, Jong HP (2019) A comprehensive study on APT attacks and countermeasures for future networks and communications: challenges and solutions. *J Supercomput*
36. Gunduz MZ, Das R (2020) Cyber-security on smart grid: threats and potential solutions. *Comput Netw* 169
37. Hanif H, Md Nasir MHN, Ab Razak MF, Firdaus A, Anuar NB (2021) The rise of software vulnerability: taxonomy of software vulnerabilities detection and machine learning approaches. *J Comput Netw Appl* 179

Cyber-Attack Detection Using Machine Learning Technique



Karan Singh, Surbhi Singh, Mehar Vohra, and Ravi Shankar Jha

Abstract Cybersecurity is crucial for protecting users of the internet on various electronic devices in everyday life. It is crucial for safeguarding some extremely sensitive information, including biotechnology and military resources, that are gravely threatened by hackers. An organisation must be aware of the dangers posed by cyber-attacks, assess those dangers, and decide what kind of defences are necessary and where to place them. In this study, we conceptualise the prediction of cyber-attacks as a classification problem, with networking sectors predicting the type of network attack from a given dataset using machine learning techniques. ML techniques, which can learn from patterns in data and make predictions based on those patterns, have emerged as a promising approach to cybersecurity. Proposed work first reviews different cyber-attacks and the challenges associated with their detection and classification. It then describes the ML techniques that have been applied to this problem, including supervised and unsupervised learning algorithms, and discusses the features and data sources that are typically used in these models. The paper also presents experimental results from applying various ML techniques to a dataset of simulated cyber-attacks. The results show that ML techniques can achieve high accuracy in classifying different types of cyber-attacks, and that feature engineering and data pre-processing techniques can significantly improve classification performance. In this paper we have discussed the limitations and future directions of ML-based cyber-attack classification, including the need for more diverse and real-world datasets, we also have discussed the challenges also explain ability and interpretability in ML models, and the potential for adversarial attacks on ML-based cybersecurity systems.

Keywords Cyber-attacks · Machine learning · Attacks classification · Cybersecurity

K. Singh · S. Singh · M. Vohra (✉) · R. S. Jha
Computer Science & Engineering, DIT University, Dehradun 248009, India
e-mail: meharvohra09@gmail.com

1 Introduction

In recent years, advances in computer and network technology have brought considerable convenience to people's lives. The fast expansion in computer system connectivity and accessibility has resulted in regular opportunities for cyber-attacks. Attacks on computer infrastructures are becoming a growing concern.

Nowadays, emails and secondary devices are the primary sources of malicious object transmission in computer networks. A malicious item is a piece of software that infects computer systems. There are various types of harmful objects, such as worms, viruses, and Trojan horses, which differ in how they target computer systems and the malicious behaviours they conduct. Many shorthand terms are there for categorising cyber-attacks, including worm, Trojan horse, spam, denial-of-service, phishing, and so forth. Several terminologies share properties, for example, viruses and Trojans require an external event to be activated. So, if an attack has characteristics in common with more than one word, it can be categorised under each of them. Having the knowledge of various cyber-attack types makes it easier to protect our systems and networks from them.

1.1 *Benign Attack*

This is a thoughtful security breach or penetration test carried out by authorised people or organisations to find systemic weaknesses without harming the target system or its data.

1.2 *Brute Force*

A cyber-attack known as brute force uses trial-and-error techniques to guess a password or encryption key. The attack, which involves trying every possible character combination until the right one is found, can be effective if the password or key is flimsy or simple to figure out.

1.3 *Denial-of-Service Attack*

A cyber-attack known as a denial-of-service (DoS) assault occurs when an attacker floods a server, website, or network with traffic or requests, making it inaccessible to authorised users. The intention is to interfere with regular business, harm property, or demand money. DoS attacks can be conducted in a number of ways, including flooding, escalation, and application-layer assaults.

1.4 DDoS (*Distributed Denial of Service*)

These attacks use a large number of computers, frequently referred to as a botnet, to saturate a server or network with traffic or requests. A denial of service is the result, making the targeted service unavailable to authorised users.

1.5 *Infiltration*

Gaining unauthorised access to a system or network, frequently with the intent to steal sensitive information or cause harm, is referred to as infiltration. It may involve manipulating people through social engineering, taking advantage of holes in hardware or software, or using malware like trojans or backdoors to gain access and take over. Here is some ML techniques that are used to detect cyber-attacks:

- A. *Logistic Regression*. This statistical algorithm is employed to divide data into distinct categories. Through the analysis of data pertaining to various attacks, it can be used to categorise cyber-attacks.
- B. *Decision Trees*. This algorithm for machine learning creates a model of decisions and potential outcomes that resembles a tree. Based on their characteristics, it can be used to categorise cyber-attacks.
- C. *Random Forest*. This ensemble learning technique increases classification accuracy by combining various decision trees. It can be used to categorise cyber-attacks by looking at a lot of different features.
- D. *Support Vector Machines (SVM)*. This algorithm for classifying data finds the ideal hyperplane to divide two classes. By examining data related to various attacks, it can be used to categorise cyber-attacks.
- E. *Neural Networks*. The design and operation of the human brain served as the basis for this machine learning technique. By analysing a lot of data and finding patterns, it can be used to categorise cyber-attacks.

Upcoming section, that is, Sect. 2, depicts literature review. Section 3 is all about methodology used in our project. Then in Sect. 4, results and conclusions are provided.

2 Literature Review

Many reports and surveys done by multiple researchers show that classification of attack is most valuable strategies to identify attacks characteristics and help out to prevent system from such attacks. Many survey reports published such as done by [1] identified that cyber-attacks can be targeted, untargeted, or insider attacks. Their study found that one of the major obstacles to cybersecurity is the persistent evasive

tactics and pervasiveness of cybercriminals. The study recommended that businesses adopt incident response plans and implement them, as well as keep them up to date on a regular basis, in order to help prevent cyber-attacks from happening. System-Fault-Risk (SFR) is a classification scheme for cyber-attacks developed by authors [2]. Based on sound scientific principles, this framework integrated theories from system engineering, fault modelling, and risk assessment. Their work is on existing classifications which emphasises on separating cause and effect and further refining effects to include state and performance. A new approach to detect and mitigate DDoS and ransomware attacks in Cyber-Physical Systems (CPS) [3] such as power generation grids and wastewater treatment plants. It uses flow-based and sequential frequent pattern features to identify attack types and detect their presence, and an ensemble classifier consisting of SVM 1, SVM 2, and NN models. The optimised DCNN is used to fine-tune the ensemble classifier's weights using a hybrid optimisation model based on S and WOA algorithms. Authors suggest a classification method [4] for cyber-attacks that groups attacks into the closest category based on defining characteristics and a game-theoretic approach. The proposed approach is simple and extendable, allowing for the addition of new characteristics of newly identified attacks, which are assigned unique weights using a proposed formula. The paper also outlines the cause-and-effect relationships for all potential attacks, which aids in identifying the best ways to stop them online. The proposed approach is demonstrated through a case study that involves the classification of several types of cyber-attacks. A complex cyber-attacks and proposes detection mechanism by author [5] using Intrusion Detection Systems (IDS) to detect and classify attacks based on network traffic properties. It evaluates the performance of IDS using several IDS datasets containing network attack features and different Artificial Intelligence techniques. It presents multiple IDS datasets and evaluates them using various techniques, providing a benchmark for designing efficient and effective IDS, and highlights the need to constantly update and improve IDS models to combat emerging attack categories. A new method to categorise cyber-attacks is introduced by changing the Gaussian kernel of an improved Support Vector Machine (iSVM), according to [6]. Pattern recognition tasks are performed using the iSVM, which is based on a machine learning technique. The suggested method improves the performance of conventional SVM by changing the Gaussian kernel to improve attack class separability. Generalised Discriminant Analysis (GDA) is used for feature reduction in the proposed technique, and iSVM is used for classification. The results show that iSVM provides 100% detection accuracy for both Normal and Denial of Service (DOS) classes, as well as comparable false alarm rate, training, and testing times. The purpose of article [7] was to examine how Estonia's cyber-attack was classified as an armed conflict, how the IHL already in place addressed the issue, and whether those rules would be adequate in similar situations in the future. By contrasting Rule 30 of the Tallinn Manual 1.0 with the Geneva Convention of 1949 and Additional Protocol I of 1977 as well as some pertinent literary works, this article employs the normative approach to provide a thorough explanation of the object. Reference [8] claimed that attacks on a cloud-based AI stage can be described using cloud-based AI procedures.

Their research suggests an assault planning system that makes use of the UNSW-NB15 dataset. The classifier is a machine learning algorithm that runs on Microsoft's Azure Machine Learning (Azure MACHINE LEARNING) stage and is based on the "Multiclass Decision Forest" algorithm. The author [9] has also presented IoT-IDCS-CNN, a deep learning-based detection and classification system, for cyber-attacks in IoT communication networks. High-performance computing and convolutional neural networks are used to achieve high classification accuracy. The system, which consists of three subsystems, has a high classification accuracy of more than 99.3% for the binary-class classifier and 98.2% for the multiclass classifier. The effectiveness of the proposed system is demonstrated by the fact that it outperforms many current machine learning-based IDCS systems. A CADM (Cyber Attack Detection Model) for classifying cyber-attacks [10] done, by analysing network traffic patterns. CADM uses the ensemble classification method to determine attack-wise detection accuracy. Important features have been extracted using LASSO. It has greater visualisation capabilities and can handle working with large datasets. They discovered in their research that Jive datasets like NSL-KDD, KDD Cup 99, UNSW-NB15, URL 2016, and CICIDS 2017 are also used to test the effectiveness of the suggested model. In order to raise awareness about the various attack types and their methods of execution, a survey [11] was conducted on attacks. This allowed for the development of effective defences against such attacks. To address the shortcomings of supervised machine learning in intrusion detection systems, a novel approach called "One-Shot Learning" has been developed to classify the attack categories [12, 13]. In order to distinguish between new and previously unheard attacks using a small sample size of a new attack class without retraining, it introduces the use of a Siamese Network. The paper assesses a pre-trained model's performance in classifying new attack classes based on a single example using three widely used IDS datasets, demonstrating the model's flexibility in classifying previously unknown attacks. Researchers [14] demonstrate how they have addressed various facets of network security and cybersecurity in the modern era. Additionally, they have tried to address the risks on corporate intranets. Based on perceived indicators of attack (IoA), a pivot attack classification [15] criteria can be used to determine the level of connectivity attained by the adversary. To evaluate the cyber-attacks [16] it's required to look into CPS (Cyber Physical System) role in cyber security domain. An importance of cyber-physical systems (CPS) in critical areas such as health care, smart grids, and aircraft, and the increasing risk of cyber-attacks against these systems. It highlights the lack of theories and tools to understand new threats and their impacts on CPS. The article also reviews frameworks and taxonomies for classifying cyber-attacks, analyses previous studies on CPS threats, and provides insight into the status of CPS risks. The authors of [17] tested a number of machine learning algorithms, including Random Forest, Support Vector Machine (SVM), Artificial Neural Network (ANN), and Naive Bayes, to categorise cyber-attacks on a smart grid network. Based on these algorithms' accuracy, precision, recall, and F1 score, they assessed their performance. The results showed that the random forest algorithm performed better than the other algorithms in terms of accuracy, precision, recall, and F1 score. This shows that random forest having capacity to classify these attacks in well manner. The proposed

feature similarity-based machine learning model, according to [18], is successful in identifying DDoS attacks in contemporary network settings for Industry 4.0. The model is a promising method for enhancing the security of contemporary network environments because it can precisely identify the pertinent features and categorise network traffic. Reference [19] also look at the state-of-the-art in machine learning-based cybersecurity, highlighting some of the major issues and future directions for study. Reference [20] draw attention to the potential negative effects that malware attacks may have on the security and dependability of online supply chains and the importance of using powerful predictive analytics to recognise and stop such attacks. In order to forecast malware attacks in a cyber supply chain, they suggest a machine learning-based strategy that makes use of feature selection and classification algorithms. In large-scale smart grids, [21] emphasises the potential of deep unsupervised learning techniques for cyber-attack detection and mitigation. The suggested system can be a helpful tool for researchers and cybersecurity experts in enhancing the security of smart grid networks. Reference [22] emphasising the value of feature selection in machine learning-based approaches for cybersecurity, the paper discusses the potential of using high-frequency features in cyber-attack detection. Researchers and practitioners in cybersecurity may find the suggested method to be a useful tool for spotting and thwarting online attacks. Reference [23] adds to the growing body of research on the application of machine learning techniques to cybersecurity and highlights the significance of creating practical methods for identifying and thwarting cyber threats in IoT networks. Reference [24] offers a promising method that makes use of information-theoretic entropy and random forests to identify HTTP-based DDoS attacks in a cloud setting. The experimental findings show how well the suggested system works and allude to possible applications in real-world situations. A system for detecting malicious web links and identifying their attack types is suggested in the research paper [25]. The system works by examining web links and the webpages they link to in order to find potential threats and classify them into different attack types. A system for categorising and forecasting significant cyber incidents (SCI) using data mining and machine learning methods is presented in [26]. The suggested system gathers information on previous cyber incidents, extracts pertinent features, and applies machine learning algorithms to categorise and forecast upcoming incidents. Reference [27] compares a number of well-known machine learning algorithms, for instance logistic regression, decision trees, random forests, and support vector machines, for identifying cyber threats in a simulated network environment. Each algorithm's performance is evaluated using metrics like accuracy, precision, recall, and F1 score. A case study involving the identification of phishing emails is used by [28] to show the viability of the suggested approach. According to the experimental findings, shared machine learning models can accurately and successfully identify phishing emails. Additionally, the authors cover the advantages and difficulties of distributing machine learning models as IOCs and make suggestions for further study. Reference [29] illustrates the efficiency of machine learning algorithms in identifying and categorising cyber-attacks on smart grid networks and emphasises the significance of such a method in boosting the security of these vital infrastructure systems.

3 Proposed Methodology

For the different types of attack classification, we are using the machine learning in which we will train the model with the help of previous dataset. We are using random forest technique for the classification, which take the average votes and send the final decision to the tree. WE sued public dataset named IPS/IDS dataset on AWS (CSE-CIC-IDS2018), available at (<https://www.unb.ca/cic/datasets/index.html>) to train our model.

Various phases required for prediction are:

Phase 1. The very first step was to find dataset which could be used for training of model. WE sued public dataset named IPS/IDS dataset on AWS (CSE-CIC-IDS2018), available at (<https://www.unb.ca/cic/datasets/index.html>) to train our model.

Phase 2. So we proceed. Cleansing, transforming, and preparing the data for modelling are all examples of pre-processing. This involves eliminating duplicates, adding values where there are gaps, and changing categorical variables into numerical ones.

Phase 3. The next step is to choose features that are pertinent to the detection of cyber-attacks. This entails determining the crucial factors that influence the target variable.

Phase 4. The data must now be divided into training and testing sets. The random forest model will be trained using the training set, and its performance will be assessed using the testing set.

Phase 5. Now utilising the training set, create a random forest model. An ensemble learning technique called random forest builds several decision trees and combines the results to make predictions. Due to its proficiency with high-dimensional data and nonlinear relationships, it is a well-liked algorithm for identifying cyber-attacks.

Phase 6. Analyse the random forest model's performance on the test set. Measuring metrics like recall, accuracy, precision, and F1 score is involved in this.

Phase 7. To raise the random forest model's efficiency, optimise its hyperparameters. Adjustments must be made to the number of trees, the maximum depth, the minimum samples per leaf, and other variables.

Phase 8. Use the trained model in a production environment to quickly identify online threats. To offer a complete defence against cyber threats, the model can be integrated with other cybersecurity tools (Fig. 1).

Pseudo code for base random forest:

Input:

X: a dataset of size $n \times m$, where n is the number of observations and m is the number of features of selected dataset

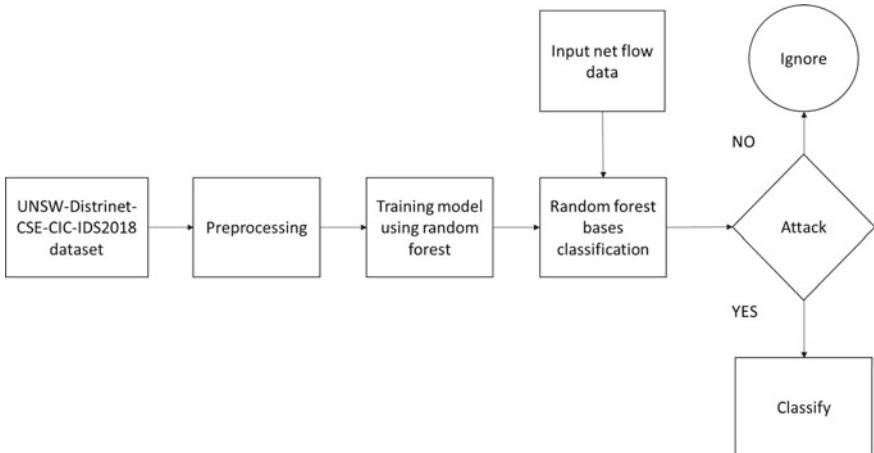


Fig. 1 Flow diagram of proposed model

y: a vector of length n containing the class labels for each observed dataset

Step 1: `num_trees`: the number of trees to include in the forest

Step 2: `max_depth`: the maximum depth of each tree

Step 3: `min_samples_split`: the minimum number of observations required to split a node

Step 4: `max_features`: the maximum number of features to consider at each split

Output:

`forest`: a list of `num_trees` Decision Trees for collected dataset

Here for i in `range(num_trees)`:

Randomly sample n observations from X with replacement

Step 5: $X_sample, y_sample = \text{bootstrap_sample}(X, y)$

Build a decision tree on the sampled data

Step 6: $\text{tree} = \text{build_decision_tree}(X_sample, y_sample, \text{max_depth}, \text{min_samples_split}, \text{max_features})$

Add the decision tree to the forest

`forest.append(tree)`

`return forest`

Here, `bootstrap sample` randomly selects n observations from X with replacement, and `build_decision_tree` is a function that builds a decision tree on the given data with the specified parameters.

Once the forest is built, we can use it to classify new observations by aggregating the predictions of all the trees in the forest. For example, to classify a new observation x , we can use:

```
prediction = mode([tree.predict(x) for tree in forest])
```

Here, `tree.predict(x)` returns the predicted class for x based on the decision tree, and `mode` returns the most common prediction among all the trees.

4 Result and Conclusion

Our experimental works show that the classification of cyber-attacks at accurate precision achieved high and here to get this high we have opted for machine learning high optimal classification using AdaBoost, as we have shown in proposed model section which has given high accuracy with respect to their training and testing as shown in Fig. 2.

As results show the accuracy of proposed model is 99.96, it also needs to note that in case of random forest techniques with same dataset does not having proposed accuracy. Dataset and value the increasing connectivity of computer systems have made cybersecurity an essential concern for protecting private data from hackers. Machine learning techniques have been applied to classify various types of cyber-attacks, achieving high accuracy. However, more diverse datasets, interpretability, and defence against adversarial attacks are needed to improve ML's performance. Understanding different cyber-attacks, for instance denial-of-service and brute force attacks, can help protect networks and systems. Furthermore, authorised penetration tests or benign attacks can be carried out to identify system weaknesses. The comparison of random forest and AdaBoost machine learning algorithms showed

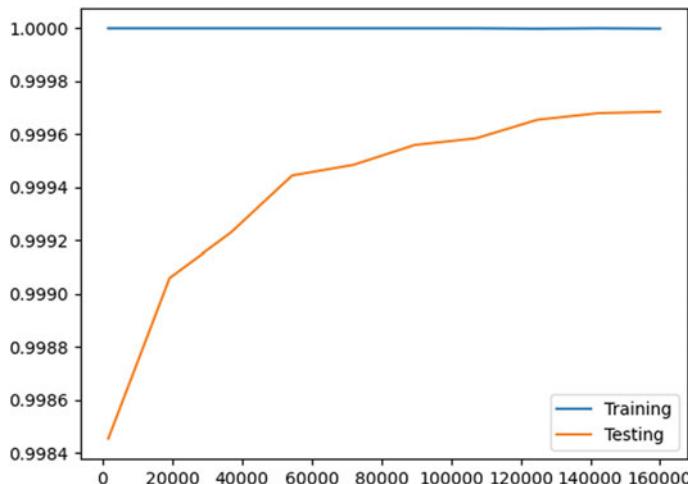


Fig. 2 Training and testing graph of proposed model

that random forest is better in classifying various cyber-attacks. Overall, this paper highlights the importance of cybersecurity and the potential of machine learning to improve it.

References

1. Ugoboaja S, Osuo-Genseleke M, Chigozie-Okwum C. Cyber attacks: a literature survey
2. Ye N, Clark N, Toni F. A system-fault-risk framework for cyber attack classification. Arizona State University, Tempe, AZ, 1389–1995/05/06/\$17.00 © 2005/2006—IOS
3. Prabhu S, Nethravathi PS (2022) A review on conceptual model of cyber attack detection and mitigation using deep ensemble model. *Int J Appl Eng Manag Lett (IJAEML)* 6(1). ISSN: 2581-7000
4. Mishra BK, Saini H (2009) Cyber attack classification using game theoretic weighted metrics approach. *World Appl Sci J* 7(Special Issue of Computer & IT):206–215. ISSN: 1818-4952
5. Yusof NNM, Sulaiman NS (2022) Cyber attack detection dataset: a review. *J Phys Conf Ser* 2319(1):012029. <https://doi.org/10.1088/1742-6596/2319/1/012029>
6. Singh S, Silakari S (2015) Cyber attack detection system based on improved support vector machine. *Int J Secur Appl* 9(9):371–386. <https://doi.org/10.14257/ijisia.2015.9.9.32>
7. Zahra I, Handayani I, Christiani DW (2021) Cyber-attack in Estonia: a new challenge in the applicability of international humanitarian law. *Yustisia* 10(1)
8. Chaudhari KG (2018) Cyber attack classification in Microsoft Azure using deep learning algorithm. *Int J Innov Res Sci Eng Technol* 7(7)
9. Al-Haija QA, Zein-Sabatto S (2020) An efficient deep-learning-based detection and classification system for cyber-attacks in IoT communication networks. *Electronics* 9:2152. <https://doi.org/10.3390/electronics9122152>
10. Hossain F, Akter M, Uddin MN (2021) Cyber attack detection model (CADM) based on machine learning approach. Department of Computer Science & Engineering, Jagannath University, Dhaka, Bangladesh. ISBN: 978-0-7381-3042-2/21/S31.00 2021. IEEE. <https://doi.org/10.1109/ICREST51555.2021.9331094>
11. Kurariya S (2019) Classification of cyber attack. *JETIR* 6(6)
12. Hindy H, Tachtatzis C, Atkinson R, Brosset D, Bures M, Andonovic I, Michie C, Bellekens X. Leveraging Siamese networks for one shot intrusion detection model
13. Pipyras K, Mitrou L, Gritzalis D (2017) Evaluating the effects of cyber-attacks on critical infrastructures in the context of Tallinn Manual. In: Information security & critical infrastructure protection (INFOSEC), Sept 2017. Athens University of Economics & Business, Athens, GR
14. Kashif M, Malik SA, Abdullah MT, Umair M, Khan PW (2018) A systematic review of cyber security and classification of attacks in networks. *Int J Adv Comput Sci Appl* 9(6)
15. Marques RS, Al-Khateeb H, Epiphanou G, Maple C (2022) Pivot attack classification for cyber threat intelligence. *JISCR* 5(2)
16. Al-Mhiqani MN, Ahmad R, Abidin ZZ, Ali NS, Abdulkareem KH (2019) Review of cyber attacks classifications and threats analysis in cyber-physical systems. *Int J Internet Technol Secur Trans* 9(3)
17. Aribisala A, Khan MS, Husari G (2021) Machine learning algorithms and their applications in classifying cyber-attacks on a smart grid network. In: 2021 IEEE 12th annual information technology, electronics and mobile communication conference (IEMCON), Oct 2021. IEEE, pp 0063–0069
18. Sambangi S, Gondi L, Aljawarneh S (2022) A feature similarity machine learning model for DDOS attack detection in modern network environments for industry 4.0. *Comput Electr Eng* 100:107955

19. Dasgupta D, Akhtar Z, Sen S (2022) Machine learning in cybersecurity: a comprehensive survey. *J Defense Model Simul* 19(1):57–106
20. Yeboah-Ofori A, Boachie C (2019) Malware attack predictive analytics in a cyber supply chain context using machine learning. In: 2019 international conference on cyber security and internet of things (ICSIoT), May 2019. IEEE, pp 66–73
21. Karimipour H, Dehghanianha A, Parizi RM, Choo KKR, Leung H (2019) A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids. *IEEE Access* 7:80778–80788
22. Özalp AN, Albayrak Z (2022) Detecting cyber attacks with high-frequency features using machine learning algorithms. *Acta Polytech Hung* 19(7)
23. Machaka P, Ajayi O, Maluleke H, Kahenga F, Bagula A, Kyamakya K (2021) Modelling DDoS attacks in IoT networks using machine learning. arXiv preprint [arXiv:2112.05477](https://arxiv.org/abs/2112.05477)
24. Idhammad M, Afdel K, Belouch M (2018) Detection system of HTTP DDoS attacks in a cloud environment based on information theoretic entropy and random forest. *Secur Commun Netw* 2018
25. Choi H, Zhu BB, Lee H (2011) Detecting malicious web links and identifying their attack types. *WebApps* 11(11):218
26. Mumtaz G, Akram S, Iqbal W, Ashraf MU, Almarhabi KA, Alghamdi AM, Bahaddad AA (2023) Classification and prediction of significant cyber incidents (SCI) using data mining and machine learning (DM-ML). *IEEE Access*
27. Farooq HM, Otaibi NM (2018) Optimal machine learning algorithms for cyber threat detection. In: 2018 UKSim-AMSS 20th international conference on computer modelling and simulation (UKSim), Mar 2018. IEEE, pp 32–37
28. Preuveneers D, Joosen W (2021) Sharing machine learning models as indicators of compromise for cyber threat intelligence. *J Cybersecur Priv* 1(1):140–163
29. Aribisala A, Khan MS, Husari G (2021) Machine learning algorithms and their applications in classifying cyber-attacks on a smart grid network. In: 2021 IEEE 12th annual information technology, electronics and mobile communication conference (IEMCON), Oct 2021. IEEE, pp 0063–0069
30. Sankar E, Nikhil M, Reddy GS (2022) Cyber attacks prediction using data science. *IJSREM* 06(03). ISSN: 2582-3930. <https://doi.org/10.5504/IJSREM11906>

AE-LSTM: A Hybrid Approach for Detecting Deepfake Videos in Digital Forensics



Megha Kandari, Vikas Tripathi, and Bhaskar Pant

Abstract Deepfakes can have serious implications for security, privacy, and trust, as deepfake can be utilized for the purpose of spreading misinformation, fake news, and propaganda. Deepfakes which are created through deep-learning techniques have become threatening in recent times and pose a significant challenge to digital forensics. As a result, deepfake video detection is a significant area of research in digital forensics. In this paper, we proposed an autoencoder-LSTM-based solution for the detection of deepfake videos, in this method autoencoder helps to obtain a robust solution. The proposed method gives an accuracy of 81.73 on the Celeb-df dataset.

Keywords Artificial intelligence · Autoencoder · Celeb-df · Deepfake · Deep learning

1 Introduction

Deepfakes refer to manipulated or synthesized audio, video, or images created using deep-learning technologies like artificial neural networks. These technologies enable the creation of highly realistic and often convincing media that can be difficult to distinguish from authentic content as shown in Fig. 1. Deepfakes can be created through several approaches such as generative adversarial networks (GANs), autoencoders, and other deep-learning architectures.

The term “deepfake” is derived from “deep learning” and “fake”, highlighting the use of deep-learning techniques to create falsified media [1]. Deepfakes are videos that consist of the manipulation of facial expressions (Fig. 2), body movements, and speech to make it appear as though a person is saying or doing something they never actually did [2].

M. Kandari (✉) · V. Tripathi · B. Pant
Graphic Era Deemed to be University, Dehradun, India
e-mail: megakandari134@gmail.com



Fig. 1 Example of deepfake from Celeb-df dataset [10]. Top: original videos, bottom: deepfake videos

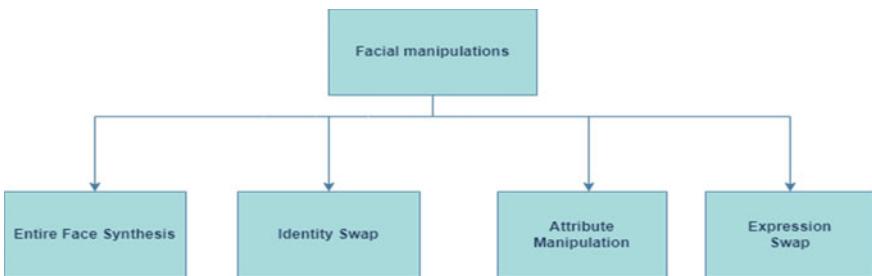


Fig. 2 Manipulation categories

Deepfakes have become increasingly popular in recent years, particularly on social media platforms [3], where they have been used for entertainment, satire, and political manipulation. However, deepfake videos have also become a growing concern in recent years due to their potential to spread disinformation, manipulate public opinion, and cause harm to individuals [4].

There is an urgent need for deepfake detection technology which is driven by the growing concern about deepfakes and their potential harm, making it a critical area of research in digital forensics. There are various approaches to deepfake video detection, including forensic analysis [5], machine learning-based methods, and blockchain-based solutions [6]. Forensic analysis involves analyzing the video's metadata, compression artifacts, and other technical features to identify inconsistencies and anomalies that may indicate manipulation. Machine learning-based methods [7] use algorithms to learn patterns and characteristics of real and fake videos and can be trained on large datasets of known deepfake videos. Blockchain-based solutions [8, 9] use distributed ledger technology to track and verify the authenticity of video content.

While deepfake detection technology is currently in its early stages of development, it has already shown promising results in detecting various types of deepfakes [11]. However, technology is constantly evolving, and its effectiveness may vary depending on the sophistication of the deepfake creation techniques. As such, ongoing research and development are needed to advance the state of deepfake detection technology and mitigate the potential risks of deepfakes.

The subsequent sections of this paper will follow this structure: Sect. 2 presents a comprehensive overview of the related work in the field of deepfake detection. In Sect. 3, we discuss the proposed methodology with a description of the steps involved in the process. Section 4 contains the presentation of the experimental results obtained using visual features. Finally, Sect. 5 provides the conclusion of the findings.

2 Related Work

The problem of detecting deepfake videos has gained significant importance owing to the widespread accessibility of advanced deep-learning techniques [12] that enable the creation of highly realistic media that can be used to deceive people. In recent years, there has been a growing body of research aimed at developing techniques for detecting deepfake media. Initially, the PRNU technique was being used for detecting deepfakes, but as deepfakes are becoming more sophisticated and in anti-forensics PRNU fingerprints can be manipulated. As a result, there is a shift from PRNU to deep-learning techniques for increasing the effectiveness of deepfake detection techniques in digital forensics [4].

One of the most widely studied approaches to deepfake detection is based on analyzing artifacts [13] that are left behind by the synthetic media generation process [14]. For example, deepfake videos may exhibit visual artifacts such as inconsistent lighting or blurring around the edges of objects.

Rafique et al. [15] proposed a deepfake detection framework in which initially the pictures have been normalized, and then the traditional method of error level analysis is performed on the data before passing it to the CNN model.

Researchers have developed deep-learning models [16] that can detect such artifacts and use them to identify deepfake media.

Li et al. [17] proposed a deepfake detection approach that is based on eye blinking in the video. This approach shows to obtain high results on the deepfake video generated based on DNN.

Another approach to deepfake detection is based on analyzing the subtle differences between real and synthetic media [18]. For example, deepfake images may have inconsistencies in facial features, such as unnatural eye movements or inconsistencies in facial expressions. Researchers have developed deep-learning models that are trained to identify these differences and use them to detect deepfakes.

A comparative analysis of some deepfake detection technology is given in Table 1.

In addition to these techniques, researchers have also explored the use of audio [19] and text-based [20] clues for detecting deepfake media. For example, deepfake

Table 1 A comparison and analysis of different deepfake detection techniques

Title	Year	Dataset	Techniques	Performance
Exposing deep fakes using inconsistent head poses [21]	2019	UADFV and DARPA GAN	SVM-based on head poses	AUC = 0.866 and 0.890
Deepfake video detection through optical flow-based CNN [22]	2019	FaceForensics++	VGG16 ResNet50	81.61% 75.46%
Exploiting visual artifacts to expose deepfakes and face manipulations [23]	2019	5330 samples of collected videos	MLP, logistic regression	AUC = 0.866
Fighting deepfake by exposing the convolutional traces on images [24]	2020	FACEAPP generated deepfakes	Expectation–maximization algorithm	Accuracy = 93%
Exposing vulnerabilities of deepfake detection systems with robust attacks [25]	2022	FaceForensics++	XceptionNet and MesoNet	80.39 and 90.50%

videos may exhibit audio artifacts such as background noise or inconsistencies in voice quality [19]. Researchers have developed models that can analyze audio signals [21] and identify such artifacts. Similarly, deepfake text may exhibit inconsistencies in writing style or errors in grammar that can be used to identify it as synthetic.

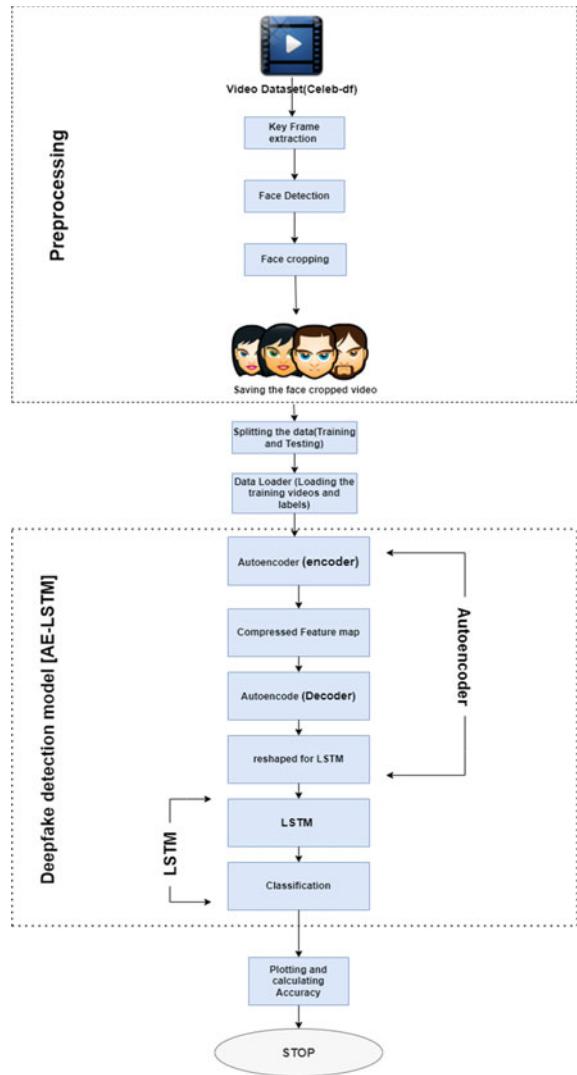
Overall, the field of deepfake detection is rapidly evolving, with new techniques and approaches being developed on an ongoing basis. As deepfake technology progresses further, it will be crucial for researchers to keep themselves informed of the latest techniques and to constantly explore the limits of what is achievable in this area.

3 Proposed Methodology

This section presents the proposed framework for detecting deepfake videos. Figure 3 illustrates the pipeline of the proposed framework. The primary concept is to develop a robust system that employs autoencoders for feature extraction.

The dataset used for this research purpose is a publicly available dataset, Celeb-df [10]. This dataset consists of 1203 videos [26]. This dataset is a second-generation

Fig. 3 The proposed framework shows the AE-LSTM hybrid approach for deepfake video detection



dataset consisting of more complex video manipulation compared to first-generation datasets [26] like, FaceForensics++ [27], and UADFV [28].

The preprocessing step consists of face recognition from video using the Dlib C++ [29] library. After that, the face crop and resizing have been performed. The next step of the preprocessing phase is the removal of corrupt videos.

The proposed model first uses an autoencoder [30, 31] to extract low-dimensional feature maps from video frames, which are then fed into an LSTM-based [32, 33] classifier to classify the video. The AE-LSTM model returns the intermediate feature map and classification score.

The main benefit of adding an autoencoder with LSTM in the above code is that it can help to extract more meaningful and compact features from the input video frames, which can lead to better classification performance.

The autoencoder component of the model is trained to learn a low-dimensional representation of the input video frames that captures the most important and relevant information while discarding irrelevant details. This is achieved by reducing the reconstruction error between the input and output tensors, which motivates the autoencoder to acquire a compressed representation of the data.

The LSTM component of the model processes the sequence of feature maps produced by the autoencoder over time and produces a sequence of hidden states that capture the temporal dynamics of the video [34]. By using the compressed feature maps produced by the autoencoder as input to the LSTM, the model can effectively capture the most significant information in the video frames while ignoring irrelevant details, which can enhance the robustness and generalization of the model.

Overall, the combination of an autoencoder with LSTM can lead to more effective feature extraction and better classification performance, particularly for complex and high-dimensional inputs such as video.

4 Result and Analysis

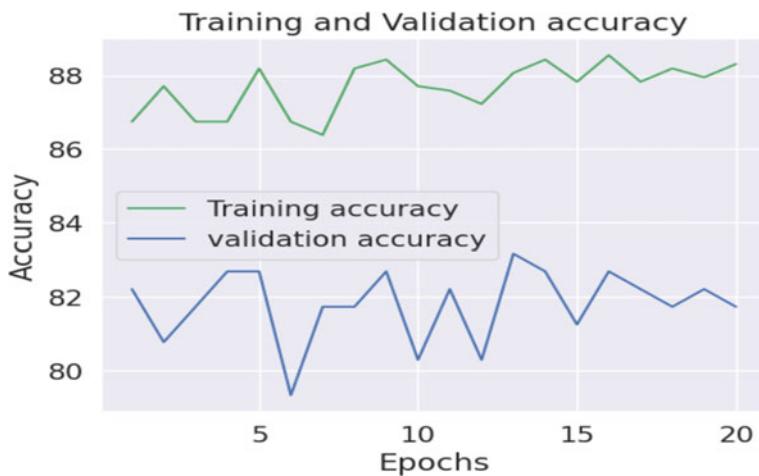
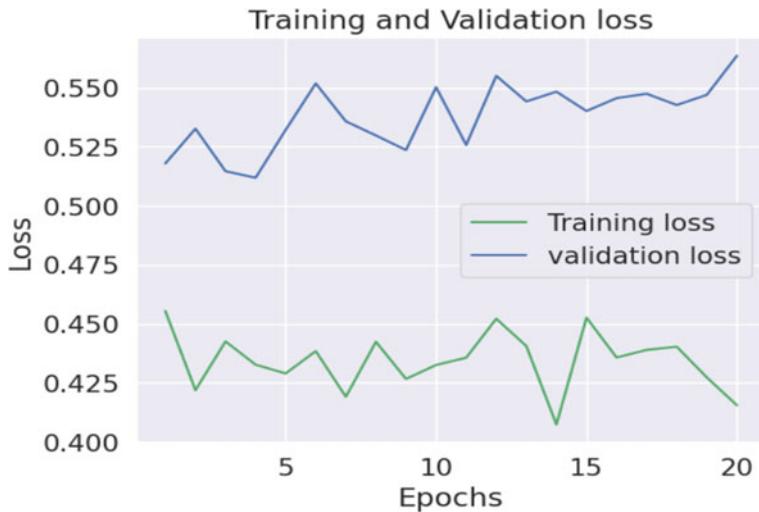
In our proposed approach, we have used Autoencoder and LSTM for feature extraction and classification. An accuracy of 81.73 has been obtained from this model. Performance evaluation of the proposed framework has been conducted based on several metrics, including loss, accuracy, true positive value (TP), false positive value (FP), true negative value (TN), and false negative value (FN).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{FP} + \text{TP} + \text{FN} + \text{TN}}$$

Figures 4 and 5 represent the graph for the model's accuracy and model loss, respectively. Table 2 provides the performance summary of the model by mentioning the evaluation parameters and their respective values.

Figure 6 illustrates the confusion matrix for the proposed framework.

The primary goal of this paper is to propose a novel approach for deepfake detection, rather than outperforming existing deepfake detection models. In the future, the model can be trained on larger datasets that consist of manipulated videos of a wide range of manipulations.

**Fig. 4** Graph for model accuracy**Fig. 5** Graph for model loss**Table 2** A performance summary with performance evaluation parameters and their values

S. No.	Evaluation parameter	Value
1.	True positive value (TP)	48
2.	False positive value (FP)	14
3.	True negative value (TN)	130
4.	False negative value (FN)	48
5.	Loss	0.5
6.	Accuracy	81.73

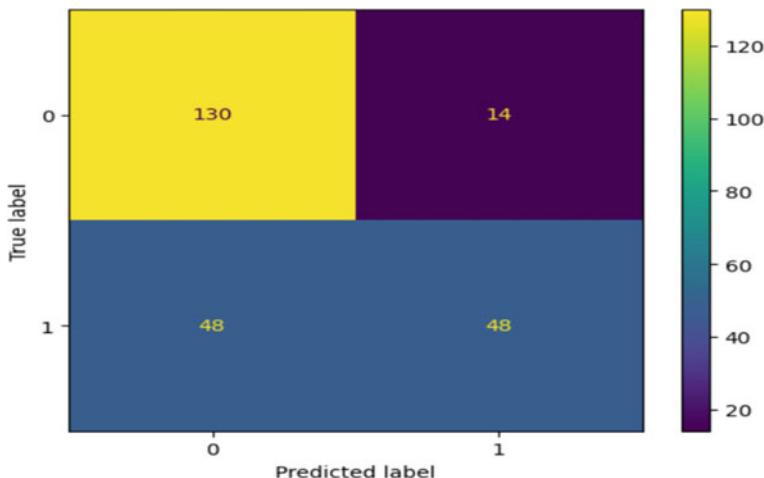


Fig. 6 Confusion matrix of the model

5 Conclusion

In conclusion, the framework proposed for deepfake detection using autoencoder and LSTM has shown promising results in identifying manipulated videos. By leveraging the power of deep learning, this framework can effectively differentiate between real and fake videos by learning the intricate patterns and features present in both types of data.

The autoencoder helps to learn the underlying features of the original video while LSTM helps in capturing the temporal dynamics of the video frames. The autoencoder-LSTM combination allows the framework to detect even subtle changes in the video frames that may not be easily discernible by the human eye.

Overall, the proposed framework provides a robust solution for deepfake detection in digital forensics, and its effectiveness has been validated through experimental results. Therefore, the proposed framework can be considered an effective solution for detecting deepfake videos in real-world scenarios.

References

1. Güera D, Delp EJ (2018) Deepfake video detection using recurrent neural networks. In: 2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS). IEEE, pp 1–6
2. Kietzmann J, Lee LW, McCarthy IP, Kietzmann TC (2020) Deepfakes: trick or treat? Bus Horiz 63(2):135–146
3. Pasquini C, Amerini I, Boato G (2021) Media forensics on social media platforms: a survey. EURASIP J Inf Secur 2021(1):1–19

4. Verdoliva L (2020) Media forensics and deepfakes: an overview. *IEEE J Sel Top Signal Process* 14(5):910–932
5. Tyagi S, Yadav D (2022) ForensicNet: modern CNN-based image forgery detection network
6. Hasan HR, Salah K (2019) Combating deepfake videos using blockchain and smart contracts. *IEEE Access* 7:41596–41606
7. Orozco ALS, Huamán CQ, Álvarez DP, Villalba LJG (2020) A machine learning forensics technique to detect post-processing in digital videos. *Future Gener Comput Syst* 111:199–212
8. Yazdinejad A, Parizi RM, Srivastava G, Dehghantanha A (2020) Making sense of blockchain for AI deepfakes technology. In: 2020 IEEE GLOBECOM workshops (GC Wkshps). IEEE, pp 1–6
9. Chan CCK, Kumar V, Delaney S, Gochoo M (2020) Combating deepfakes: multi-LSTM and blockchain as proof of authenticity for digital media. In: 2020 IEEE/ITU international conference on artificial intelligence for good (AI4G). IEEE, pp 55–62
10. Li Y, Yang X, Sun P, Qi H, Lyu S (2020) Celeb-df: a large-scale challenging dataset for deepfake forensics. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3207–3216
11. Bansal N, Aljrees T, Yadav DP, Singh KU, Kumar A, Verma GK, Singh T (2023) Real-time advanced computational intelligence for deep fake video detection. *Appl Sci* 13(5):3095
12. Chauhan R, Ghanshala KK, Joshi RC (2018) Convolutional neural network (CNN) for image detection and recognition. In: 2018 first international conference on secure cyber computing and communication (ICSCCC). IEEE, pp 278–282
13. Parkey C, Hughes C, Locken N (2012) Analyzing artifacts in the time domain waveform to locate wire faults. *IEEE Instrum Meas Mag* 15(4):16–21
14. Millière R (2022) Deep learning and synthetic media. *Synthese* 200(3):231
15. Rafique R, Nawaz M, Kibriya H, Masood M (2021) Deepfake detection using error level analysis and deep learning. In: 2021 4th international conference on computing & information sciences (ICCIS). IEEE, pp 1–4
16. Bhatt C, Kumar I, Vijayakumar V, Singh KU, Kumar A (2021) The state of the art of deep learning models in medical science and their challenges. *Multimed Syst* 27(4):599–613
17. Li Y, Chang M-C, Lyu S (2018) In ictu oculi: exposing AI generated fake face videos by detecting eye blinking. arXiv preprint [arXiv:1806.02877](https://arxiv.org/abs/1806.02877)
18. Lyu S (2020) Deepfake detection: current challenges and next steps. In: 2020 IEEE international conference on multimedia & expo workshops (ICMEW). IEEE, pp 1–6
19. Almutairi Z, Elgibreen H (2022) A review of modern audio deepfake detection methods: challenges and future directions. *Algorithms* 15(5):155
20. Zhong W, Tang D, Xu Z, Wang R, Duan N, Zhou M, Wang J, Yin J (2020) Neural deepfake detection with factual structure of text. arXiv preprint [arXiv:2010.07475](https://arxiv.org/abs/2010.07475)
21. Yang W, Zhou X, Chen Z, Guo B, Ba Z, Xia Z, Cao X, Ren K (2023) AVoID-DF: audio-visual joint learning for detecting deepfake. *IEEE Trans Inf Forensics Secur* 18:2015–2029
22. Amerini I, Galteri L, Caldelli R, Del Bimbo A (2019) Deepfake video detection through optical flow based CNN. In: Proceedings of the IEEE/CVF international conference on computer vision workshops
23. Matern F, Riess C, Stammerer M (2019) Exploiting visual artifacts to expose deepfakes and face manipulations. In: 2019 IEEE winter applications of computer vision workshops (WACVW). IEEE, pp 83–92
24. Guarnera L, Giudice O, Battiatto S (2020) Fighting deepfake by exposing the convolutional traces on images. *IEEE Access* 8:165085–165098
25. Hussain S, Neekhara P, Dolhansky B, Bitton J, Ferrer CC, McAuley J, Koushanfar F (2022) Exposing vulnerabilities of deepfake detection systems with robust attacks. *Digit Threats Res Pract (DTRAP)* 3(3):1–23
26. Jiang L, Li R, Wu W, Qian C, Loy CC (2020) Deeperforensics-1.0: a large-scale dataset for real-world face forgery detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2889–2898

27. Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Nießner M (2019) FaceForensics++: learning to detect manipulated facial images. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 1–11
28. Yang X, Li Y, Lyu S (2019) Exposing deep fakes using inconsistent head poses. In: ICASSP 2019–2019 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 8261–8265
29. Boyko N, Basystiuk O, Shakhevskaya N (2018) Performance evaluation and comparison of software for face recognition, based on dlib and OpenCV library. In: 2018 IEEE second international conference on data stream mining & processing (DSMP). IEEE, pp 478–482
30. Meng Q, Catchpoole D, Skillicom D, Kennedy PJ (2017) Relational autoencoder for feature extraction. In: 2017 international joint conference on neural networks (IJCNN). IEEE, pp 364–371; Xing C, Ma L, Yang X (2016) Stacked denoise autoencoder based feature extraction and classification for hyperspectral images. *J Sens* 2016
31. Yu Y, Si X, Hu C, Zhang J (2019) A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput* 31(7):1235–1270
32. Graves A, Schmidhuber J (2005) Framewise phoneme classification with bidirectional LSTM networks. In: Proceedings. 2005 IEEE international joint conference on neural networks, vol 4. IEEE, pp 2047–2052
33. Gupta A, Parmar R, Suri P, Kumar R (2021) Determining accuracy rate of artificial intelligence models using Python and R-Studio. In: 2021 3rd international conference on advances in computing, communication control and networking (ICAC3N). IEEE, pp 889–894
34. Gupta A, Gupta S, Memoria M, Kumar R, Kumar S, Singh D, Tyagi S, Ansari N (2022) Artificial intelligence and smart cities: a bibliometric analysis. In: 2022 international conference on machine learning, big data, cloud and parallel computing (COM-IT-CON), vol 1. IEEE, pp 540–544

Chatbot-Based Android Application Towards Security Using FCM



Priya Singh and Rajalakshmi Krishnamurthi

Abstract This research paper emphasizes the critical need for secure mobile communication in today's digital landscape where mobile technology dominates. To address this, the paper proposes a novel approach that leverages chatbot technology in an Android application, incorporating Firebase Cloud Messaging (FCM) to enhance security. The chatbot provides a secure platform for communication, utilizing FCM's advanced features such as end-to-end encryption and message authentication to enable safe interactions between users. The chatbot feature also serves to address user inquiries on the optimal functionality of the application while simultaneously alerting users of any security breaches or suspicious activities in real-time via FCM's push notifications. The successful implementation of FCM in the chatbot application highlights the effectiveness of integrating advanced technology to enhance mobile communication security.

Keywords Chatbot · GCM · FCM · Android · Security

1 Introduction

The widespread adoption of technology across different sectors has raised concerns about security, prompting a growing demand for secure infrastructure and communication. However, the popularity of mobile communication remains high due to its widespread availability and connectivity. Mobile devices typically feature a variety of applications, including basic ones for routine tasks and specialized ones tailored to specific needs. Many of these apps rely on remote servers presented on the cloud, with Firebase Cloud Messaging (FCM) emerging as a popular and secure platform for developing mobile applications. Currently, the most powerful service for enabling

P. Singh · R. Krishnamurthi
Jaypee Institute of Information Technology, Noida, India
e-mail: priyasinghsmit@gmail.com

R. Krishnamurthi
e-mail: k.rajalakshmi@mail.jiit.ac.in

communication between cloud and applications developed by developers is Google Cloud Messaging (GCM) [1]. Mobile devices come in various shapes and sizes and are equipped with different operating systems, such as Windows, iOS, and Android. Mobile applications can be developed for a specific operating system or multiple operating systems. Google and the Open Handset Alliance (OHA) developed Android, an open-source, Linux-based operating system for mobile devices. With Android, developers can create mobile applications for Android devices [1]. To begin developing applications for Android, developers typically use the Android SDK, which includes various in-built tools. These tools can be accessed through Android Studio, a development environment recommended by Google. After downloading and installing the Android SDK, developers can utilize these tools to create their applications. Mobile devices require an operating system, hardware components, and applications to function effectively. Android is a comprehensive software platform developed specifically for mobile devices, and the Android Software Development Kit (SDK) provides developers with essential tools and APIs for building mobile applications [1]. Our primary objective was to develop a chatbot capable of answering user questions related to object detection. To accomplish this, the author [2] created a novel chatbot model that relied on a knowledge base to handle real-time queries. After testing, found that chatbot was able to accurately respond to user queries with a 94% accuracy rate. Consequently, it is concluded that chatbot is user-friendly, dependable, intelligent, and innovative in its ability to address real-time queries from users.

The research paper is structured into several sections, each with a distinct purpose. Section 1 provides an introduction to the research topic and highlights the research questions that will be addressed. Section 2 presents a review of relevant literature and previous studies in the field, highlighting the gaps in knowledge that the current research aims to fill. Section 3 outlines the methodology and overall research design employed to achieve the research objectives. Section 4 provides a detailed description of the data collection and analysis procedures, including flowcharts and pseudocode where necessary. Section 5 introduces implementation details. Section 6 presents the results of the study, including statistical analyses, tables, and chatbot communication images. Finally, Sect. 7 offers a conclusion summarizing the main findings of the study and their contribution to the field, as well as suggesting avenues for further research.

2 Related Work

GCM was the initial messaging technology offered by Google [3], but it has now been upgraded to a newer version called Google Firebase. The Firebase platform provides its own console to push notifications to applications using unique IDs created on the server. GCM had unreliable message delivery, which prompted Google to develop a new version with more features and a user-friendly interface FCM [3]. FCM essentially comprises an application server capable of interacting with FCM using HTTP or XMPP communication protocols to communicate with the client application. The

server or Firebase notification console sends messages and notifications to the client application. In summary, FCM has revolutionized Android messaging applications by providing a more efficient and reliable means of communication. Below are some discussions on related work. The author [1] aims to improve communication between students and teachers by developing a notification system using an Android application. The authors used a software development life cycle methodology to design and implement the notification system and evaluated its effectiveness using a user acceptance test. The paper provides a practical solution and insights into the development process. The user acceptance test results show that the notification system is effective in keeping students informed. However, the study's evaluation is limited to a user acceptance test, and potential drawbacks or limitations of the system are not discussed. Further, the author [4] describes the design and implementation of an Android-based wearable smart locator band for people with autism, dementia, and Alzheimer's. The device uses Bluetooth Low Energy (BLE) technology for real-time tracking of the user's location and includes an emergency button for calling for help. The pilot study with ten participants showed the device to be effective and usable, but it may have limitations due to its reliance on Bluetooth connectivity and regular charging. Future research could focus on improving the device's connectivity and battery life, as well as exploring its potential for monitoring physical activity and medication adherence. In addition, the author [2] presents a chatbot system that assists customers in ordering food and answering questions in a restaurant setting. The authors used the NLP technique to develop the chatbot system and evaluated its effectiveness using a user survey. The paper offers valuable insights into the practical application of chatbots in the hospitality industry and provides a detailed explanation of the development process. Next, the author [5] uses a sensor-based approach to develop their accident detection and notification system. The system uses sensors to detect sudden changes in acceleration and orientation, which may indicate an accident has occurred. Pros include the real-time detection of accidents, while cons include potential false positives or false negatives in sensor data.

Similarly, the author [6] aims to evaluate the effectiveness of using Firebase in developing Android applications. The authors used a case study approach to develop an Android application using Firebase and evaluated its performance using a user survey. The paper provides a practical solution and insights into the development process using Firebase. Overall, the research paper is a valuable resource for those interested in using Firebase for Android app development, but additional research is necessary to evaluate its effectiveness in a real-world setting. Next, the author [7] develops a framework for detecting rice diseases using deep learning and providing assistance through a chatbot. The paper provides a practical solution to farmers by automating the detection of rice diseases and offering assistance through a chatbot. Overall, the research paper is a valuable resource for researchers and practitioners interested in using deep learning and chatbots for agriculture, but additional research is necessary to evaluate its effectiveness in various real-world settings. Further, the author [8] presents an Android application developed for detecting drowsiness during driving using the smartphone's front-facing camera. The application analyses facial features to detect signs of drowsiness and alerts the driver if necessary. Although the

application's accuracy may not be as high as other systems that use more advanced technology, it has the advantage of being easily accessible and not requiring any additional hardware.

Similarly, the author [9] demonstrates the practical use of Google Firebase for real-time communication on Android. However, the research has some limitations as it does not discuss the potential drawbacks of Firebase for real-time communication, and the evaluation of application performance is limited. Despite these limitations, the paper is a useful resource for developers interested in building real-time communication applications on Android. In addition, the author [10] used an experimental approach to develop the mobile-based chatbot system and conducted usability testing to evaluate its performance. The paper presents a practical application of chatbots in the hospitality industry and provides valuable insights into the development process. The chatbot system offers a useful solution for users searching for suitable hostels and receives positive feedback from the usability testing. Next, the author [3] aims to investigate the use of FCM to control mobile applications. The research will focus on identifying the benefits and limitations of using FCM to control mobile applications and the potential for FCM to be used for other types of applications. This research will be novel as it will provide new insights into the specific advantages and disadvantages of using FCM as a control mechanism for mobile applications. The future scope of the research could include expanding the study to include different types of applications and industries, and exploring the integration of FCM with other technologies such as artificial intelligence and machine learning to further improve the control of mobile applications.

Similarly, the author [11] developed a chatbot system for energy consumption monitoring and management. The authors used a NLP approach to develop the chatbot system and evaluated its effectiveness using a user survey. The paper provides a practical solution to monitor and manage energy consumption, and it offers valuable insights into the development process using NLP and chatbots. Moreover, a comprehensive analysis of existing studies on chatbot-based Android applications that utilize FCM can reveal some significant insights about the use of this technology for enhancing security. Some of the key findings are:

- FCM can enhance security in chatbot-based Android applications by providing secure messaging and push notification services, and efficient management and routing of data.
- FCM can improve the overall user experience by providing real-time messaging and push notifications, which can increase user engagement and information flow.
- To secure messaging and push notifications, FCM can use encryption and authentication methods such as OAuth 2.0.
- The use of FCM in chatbot-based Android applications can effectively address security challenges such as data breaches, unauthorized access, and identity theft.
- Some limitations to using FCM are cost and the need for an internet connection to receive push notifications.

- Additional security measures such as regular software updates and security audits are necessary to ensure the sustained security of chatbot-based Android applications.

It is important to note that these findings are subject to the scope of the research question and the studies that were reviewed, and that new studies will be required to keep up with the evolving technology and security measures.

3 Overall Idea

3.1 *Architecture*

Figure 1 illustrates the overall architecture of the application which comprises three essential components: the server, Firebase, and mobile device. The application's workflow encompasses three primary stages. The first stage involves user authentication using an access token to establish a secure connection between the Firebase and server, as depicted in Fig. 1a. The second stage entails communication between the server and Firebase, while the final stage involves the delivery of notification messages from the Firebase to the user's mobile device, as shown in Fig. 1b. In Fig. 1a, the communication between the server and Firebase is shown, which is responsible for initiating an authenticated connection. Before the actual communication begins, the Firebase sends an access token to the server to establish an encrypted communication. On the other hand, Fig. 1b demonstrates that the communication between the server and the user occurs via Firebase. When the server intends to transmit data to the user, it sends the same access token it received from Firebase. Firebase validates the authenticity of the server by comparing the received access token with the one it has. If they match, encrypted data communication takes place between the server and Firebase. Finally, Firebase delivers the notification to the user's mobile device to prompt immediate action. To enable Python to work as a server-side scripting language, we utilized the Python library from FCM.

3.2 *Firebase*

Firebase is a highly sought-after web application platform that is used for developing top-notch applications. It is a JSON-based database that does not require database queries for its operations. Firebase offers various services, but for our application, we utilized FCM, Firebase authentication, Firebase storage, and Firebase notification. FCM, formerly known as GCM, is a cross-platform solution that caters to mobile applications, such as Android, iOS, and web applications. Firebase authentication provides authentication using authenticated social networking sites like Google, Gmail, Facebook, Twitter, and GitHub. These services ensure user authentication

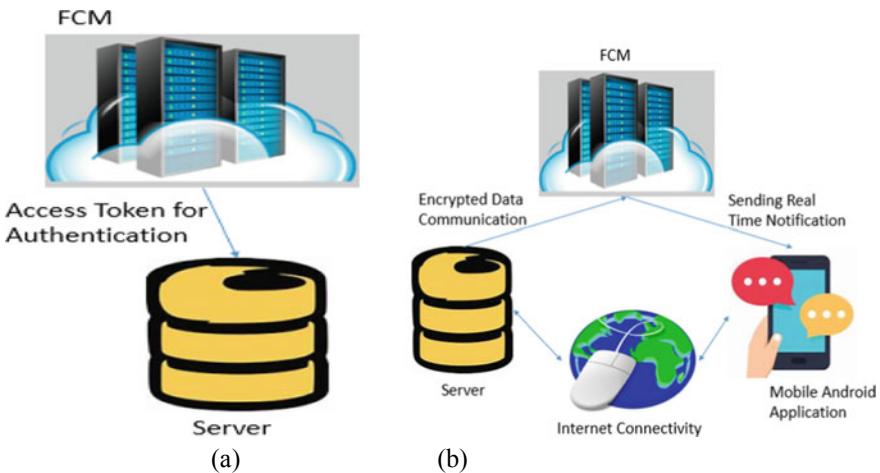


Fig. 1 Depicts communication between **a** authentication between server to FCM, **b** server and mobile user via FCM

through client-side coding and integrate with the user management system, which includes authentication using email credentials stored at Firebase. Firebase storage provides a secure storage platform for storing various types of data, such as images, videos, and audios, as well as data transfer. It integrates cost-effective and easily accessible services provided by Google cloud storage as a backup. Firebase real-time database offers real-time data and provides an API that enables data synchronization between the client and their stored data on the cloud. It generates error reports and groups similar types of errors together to be sent. Additionally, it provides a new feature to log the event that caused the crash so that the developer can rectify the cause. It also offers real-time notifications for free.

3.3 *Android*

Android is a mobile operating system that powers billions of devices globally and is widely used worldwide. Therefore, it is crucial to prioritize the security of Android applications. These applications are prone to various security threats, such as data breaches, hacking, and malware. To prevent these threats, developers must adopt a proactive approach towards securing their applications. One of the most effective ways to improve the security of Android applications is to implement various security features like encryption, authentication, and access controls. Encryption converts sensitive information into unreadable code, ensuring that it remains protected from unauthorized access. Authentication guarantees that only authorized users can access

the application, while access controls limit the actions that can be performed within the application, thus preventing unauthorized access. Another way to enhance the security of Android applications is by utilizing FCM.

4 Algorithm/Pseudocode

4.1 Sender/Receiver Algorithm and Pseudocode

Pseudocode (Sender):

```
# Initialize FCM connection
fcm = FCM(api_key)
# Create message object
message = { "to": device_token, "data": { "message": "Hello, FCM!" } }
# Send message
response = fcm.send(message)
# Check for errors or successful delivery
if "error" in response:
    print("Error sending message:", response["error"])
else:
    print("Message sent successfully:", response)
```

Pseudocode (Receiver):

```
class MyFirebaseMessagingService(FirebaseMessagingService):
    def onMessageReceived(self, remoteMessage):
        # Extract message data
        message = remoteMessage.data["message"]
        # Display message to the user
        Toast.makeText(self, message, Toast.LENGTH_SHORT).show()
        # Take any necessary action based on the message
        if message == "Security Alert":
            # Trigger security action
            pass
        else:
            # Do nothing
            pass
```

4.2 Communication Flow in FCM

- (1) The sender initializes an FCM connection using the API key.
- (2) The sender creates a JSON object with the message and the recipient's device token.

- (3) The sender sends the message to the FCM server using the “send” API.
- (4) The FCM server receives the message and sends it to the recipient’s device.
- (5) The recipient’s device receives the message and triggers a notification to the user.
- (6) The user can either open the app or ignore the notification.
- (7) If the user opens the app, the message is extracted and displayed to the user.
- (8) The user can take any necessary action based on the message content.

4.3 Connecting FCM and Android

- (1) RegisterAppOnFirebaseConsole():
 - 1.1 Open Firebase Console.
 - 1.2 Create a new project.
 - 1.3 Register the app on the Firebase Console.
- (2) DownloadGoogleServicesJson():
 - 2.1 Go to the Firebase Console.
 - 2.2 Click on the project created.
 - 2.3 Go to the app settings.
 - 2.4 Download the google-services.json file.
 - 2.5 Save the file in the app folder of the Android project.
- (3) AddFirebaseSDK():
 - 3.1 Open the app-level build.gradle file.
 - 3.2 Add the Firebase SDK dependency.
- (4) AddGoogleServicesPlugin():
 - 4.1 Open the project-level build.gradle file.
 - 4.2 Add the Google services plugin.
- (5) AddFirebaseAuthenticationDependencies():
 - 5.1 Open the app-level build.gradle file.
 - 5.2 Add Firebase authentication dependencies.
- (6) CreateFirebaseAppClass():
 - 6.1 Create a new class that extends FirebaseApp.
 - 6.2 Initialize Firebase in the onCreate method.
- (7) ConnectAppToFirebase():
 - 7.1 Open the MainActivity class.
 - 7.2 Call FirebaseApp.initializeApp in the onCreate method.

(8) AddAuthenticationCode():

8.1 Add the authentication code to the MainActivity class.

4.4 Chatbot Algorithm

- (1) User begins the chat module by asking the chatbot about an app in text format.
- (2) Chatbot module takes the input from the user and uses Natural Language Processing to extract the intent from the statements and phrases used while chatting.
- (3) Chatbot module sends a request with the keywords to the database for further processing and to search for an appropriate answer.
- (4) Database returns a response that contains all the information that has been fetched from the database using MySQL in JSON format.
- (5) All the data from the database is sent to a response JSON file.
- (6) Chatbot module parses all the relevant information from the JSON response and provides a meaningful and conclusive answer to the user.
- (7) If the conversation has ended, the Chatbot module will stop.
- (8) Otherwise, repeat step 1.

4.5 FCM Communication Algorithm

```
FirebaseAuth iDUAAuth = FirebaseAuth.getInstance();
iDUAAuth.signInWithEmailAndPassword(iDUEmail, iDUPassword)
.addOnCompleteListener(new OnCompleteListener<AuthResult>()
{
    @Override
    public void onComplete (Task<AuthResult> iDUCompleteTask)
    {
        if(iDUCompleteTask.isSuccessful())
        {
            FirebaseUser loggedInIDU = iDUCompleteTask.getResult().getUser();
            String iDUEmail = loggedInIDU.getEmail();
        } } }
```

4.6 Multilingual Algorithm

- (1) Define a string variable “languageToLoad” to store the selected language code (e.g. “en” for English, “hi” for Hindi).
- (2) Create a Locale object “locale” using the selected language code.

- (3) Set the default locale to the created “locale” object using `Locale.setDefault(locale)`.
- (4) Create a Configuration object “config”.
- (5) Set the “locale” object as the language configuration for “config”.
- (6) Update the app’s resources configuration using `getBaseContext(). getResources(). updateConfiguration (config, getBaseContext(). getResources(). getDisplayMetrics())`.

5 Implementation

This section comprises of implementation detail of Android mobile application and chatbot dialogflow.

5.1 *Android Mobile Application*

- (1) Choose a development environment: Select an appropriate SDK for Android development, such as Android Studio.
- (2) Design the layout: Use either the visual layout editor or XML code to create the user interface of the application.
- (3) Implement the functionality: Write code in either Java or Kotlin programming languages to add interactive features and functionalities to the application.
- (4) Test the application: Use either the Android emulator or a physical device to test the application and check for any errors or bugs.
- (5) Debug and troubleshoot: Use debugging tools to identify and resolve any issues with the application.
- (6) Package the application: Create an APK file for the application using Android Studio, which can be installed on Android devices.
- (7) Publish the application: Publish the application on an app store, such as Google Play Store, for users to download and install.

5.2 *Chatbot Dialogflow*

In the realm of chatbots, a knowledgebase, which is also known as a dialogbase, is a reservoir of facts and information that a chatbot can draw on to deliver precise and pertinent answers to user questions. Such a knowledgebase can include details on products, services, company policies, and common inquiries (FAQs). Often, the chatbot utilizes the knowledgebase as its primary source of information to provide thorough and informative responses to users. Depending on the chatbot’s unique demands and prerequisites, there are various methods of integrating a knowledgebase

into it. There are various approaches to incorporating a knowledgebase into a chatbot, including hardcoding, database integration, web scraping, and machine learning. Hardcoding entails manually inputting data into the chatbot's code, which is best for small chatbots with limited information. On the other hand, database integration is more flexible and allows for easy updating and scaling of the knowledgebase. Web scraping, which automatically extracts data from external websites, is ideal for chatbots that require information from external sources. Machine learning is another method that uses algorithms to automatically extract and classify information from a dataset, enabling the chatbot to learn and improve its responses over time. A reliable knowledgebase is crucial for a chatbot's success as it enables the chatbot to deliver accurate and relevant responses to user inquiries, enhancing the overall user experience. Chatbots utilize natural language to interact with users and may have a set of predetermined questions stored in the database, which can be either static or dynamic. Chatbots have replaced human entities in various domains and offer virtual assistance. Essentially, a knowledgebase serves as a text file used by the chatbot to facilitate communication with the end-user.

Next, in a chatbot, a questionnaire is a set of questions or prompts used to collect information from the user. This information helps the chatbot to provide personalized responses that better suit the user's needs, preferences, or intentions. Questionnaires are a common feature in chatbots for a variety of purposes. One of the primary reasons for using questionnaires in chatbots is to gather user information. This information is used to personalize the chatbot's responses and improve the overall user experience. For instance, a banking chatbot may ask for the user's personal information, such as account numbers or name, to offer personalized services. Another reason to use questionnaires is to identify the user's intent. By asking a series of questions, the chatbot can better understand the user's needs and respond accordingly. For example, a customer service chatbot may ask several questions to identify the user's issue and provide an appropriate solution. In addition, questionnaires are used to qualify leads, especially for sales or marketing purposes. For instance, a real estate chatbot may ask several questions to determine the user's budget, location, and preferences to offer suitable property options. Overall, questionnaires play a vital role in improving the effectiveness and efficiency of chatbots. Questionnaires can be implemented in a chatbot in several ways:

- (1) Linear: The chatbot asks a series of questions in a predefined order, and the user must answer each question before moving on to the next.
- (2) Non-linear: The chatbot presents a set of questions to the user, and the user can select the question they want to answer first.
- (3) Conditional: The chatbot asks different questions based on the user's previous answers. This allows the chatbot to personalize the experience and provide more relevant responses.

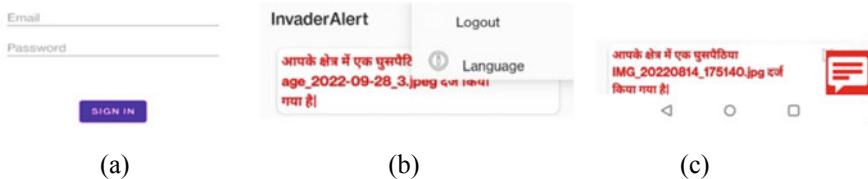


Fig. 2 Android mobile application **a** login page, **b** language change option, **c** chatbot logo in red rectangular box inside mobile application

It is important to ensure that the questionnaires are designed to be user-friendly, simple, and easy to follow. It is also crucial to avoid asking unnecessary personal questions. The chatbot must understand the user's intent, even if their responses are not exact, by using techniques like Natural Language Processing (NLP) to understand the user's meaning.

6 Result

This section comprises the result of Android application and chatbot application. Figure 2 shows the designed Android mobile application interface along with chatbot messaging interface. A chatbot is a system that can retrieve information and provide immediate responses to user queries. The user can ask questions, and the system responds with answers from its knowledgebase. Figure 3 displays the knowledgebase that contains frequently asked questions and their corresponding answers, which are commonly used in the Android mobile application developed. The system's ability to provide accurate responses to user queries is an important factor in evaluating its effectiveness. To test the accuracy of the chatbot, we conducted several experiments where users asked questions and the chatbot provided responses. The results of these experiments are shown in Table 1, which displays the questions asked, the chatbot's responses, and whether the responses were correct or incorrect. The accuracy of the chatbot was calculated using the below Eq. (1):

$$\text{Accuracy} = \frac{\text{Total number of correct answers}}{\text{Total number of questions}} \quad (1)$$

Out of the 20 questions asked in the experiment, the chatbot was able to provide correct responses to 18 questions, resulting in an accuracy of 90% as shown in Table 1.

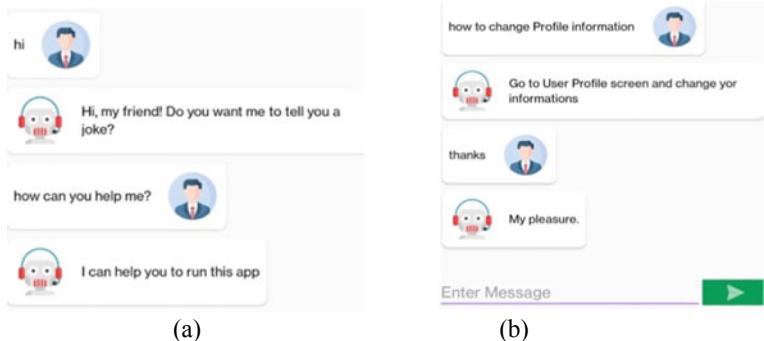


Fig. 3 User 1 and user 2 communication with chatbot in (a) and (b)

Table 1 Analysing chatbot performance response

User	Question raised	Correct answer	Incorrect answer
1	5	4	1
2	5	4	1
3	5	5	0
4	5	5	0
Total	20	18	2

7 Conclusion

In conclusion, the implementation of an Android application with chatbot capabilities using FCM communication, and support for multiple languages such as English and Hindi, offers a secure and efficient solution for client server communication. With this implementation, users can engage with chatbots seamlessly in their preferred language, leading to an improved user experience. FCM's real-time messaging capabilities and high accuracy rate of 90% ensure that users receive accurate responses, reducing the chances of confusion or frustration. This implementation has the potential to streamline user's service operations, leading to increased efficiency and productivity for businesses. By leveraging FCM communication and multilingual support, the application server provides a secure and convenient communication channel with users, ultimately boosting user satisfaction and loyalty. It is hard to exaggerate the impact of the chatbot system on enhancing user experience, as it has the ability to understand their interests and deliver appropriate responses. Further, there is scope for expanding the work by incorporating additional features and exploring new possibilities in Android applications.

References

1. Riadh MH (2016) Notification system to students using an Android application. *Int J Comput Appl* 140(1):22–27
2. Yossy EH, Budiharto W. Knowledge-based chatbot for humanoid robot in restaurant for question and answering system
3. Mokar MA, Fageeri SO, Fattah SE (2019) Using firebase cloud messaging to control mobile applications. In: 2019 international conference on computer, control, electrical, and electronics engineering (ICCCEEE). IEEE, pp 1–5
4. Goel I, Kumar D (2015) Design and implementation of Android based wearable smart locator band for people with autism, dementia, and Alzheimer. *Adv Electron* 2015
5. Ahirrao SM, Dhanrale P, Mahant L, Kotwal H (2015) Accident detection and notification system using Android. *Int J Recent Innov Trends Comput Commun* 3(3):1084–1086
6. Khawas C, Shah P (2018) Application of firebase in Android app development—a study. *Int J Comput Appl* 179(46):49–53
7. Jain S, Sahni R, Khargonkar T, Gupta H, Verma OP, Sharma TK, Bhardwaj T, Agarwal S, Kim H (2022) Automatic rice disease detection and assistance framework using deep learning and a chatbot. *Electronics* 11(14):2110
8. Mohanty M, Sikka R (2021) Android application to detect drowsiness during driving vehicle. *Mater Today Proc*
9. Chatterjee N, Chakraborty S, Decosta A, Nath A (2018) Real-time communication application based on Android using Google firebase. *Int J Adv Res Comput Sci Manag Stud* 6(4)
10. Isinkaye FO, Babs IGA, Paul MT (2022) Development of a mobile-based hostel location and recommendation chatbot system
11. Rocha CVM, Lima AA, Vieira PHC, Cipriano CLS, Paiva AMP, da Silva IFS, da Rocha SV et al (2022) A chatbot solution for self-reading energy consumption via chatting applications. *J Control Autom Electr Syst* 33:229–240

Accuracy Enhancement for Intrusion Detection Systems Using LSTM Approach



Abhishek Kajal and Vaibhav Rana

Abstract Post-pandemic threats to the network has shown that there is a need for lot of work to be done on the accuracy of IDS. Conventional research has only produced a limited number of viable options for efficient intrusion detection. When put into practice, the conclusions and suggestions that have been drawn from this research will have a considerable impact on the approach that is used to accurately predict intrusions. RNN-LSTM-based approach in the proposed model not only enhanced the accuracy of IDS, but yields less false positives and rapid detection of potential security threats. The results of the proposed IDS model have been compared with traditional IDS, where proposed model provides better Accuracy, Precision, Recall, and F1 Score. The suggested model trained on a substantial dataset, which significantly increased the possibility to provide comparatively better results. In order to improve IDS detection, it is recommended that future study continues to make use of the same paradigm of using deep learning methods.

Keywords Intrusion detection system · Deep learning · Accuracy · LSTM · RNN

1 Introduction

Research has shown that there is a need for more work to be done on the accuracy of IDS, taking into consideration the previous research that has been done in the sector. IDS stands for Intrusion Detection System. Investigation into the use of intrusion detection systems as a preventative safety measure dates back many years. The Internet of Things operates primarily at the network layer in its architecture. IDS must be able to function despite having a very limited processing capacity because it was designed for intelligent systems that are dependent on IoT. It is necessary to have a quick reaction time for this. This is built to process a large amount of data in a short amount of time.

A. Kajal (✉) · V. Rana

Department of Computer Science and Engineering, Guru Jambheshwar University of Science and Technology, Hisar, Haryana 125001, India
e-mail: abhishekkajal82@gmail.com

1.1 Background

The concept of intrusion detection will be investigated in great detail during the course of this research. Even though IDS researches have been carried out for decades, academics are still concerned about the veracity of the conclusions they have drawn from them. In order to improve the IDS detection capabilities, many machine learning methodologies are going to be used. The purpose of this study is to investigate the current work done in the area of intrusion detection so as to analyze the probabilities of improvements in this domain. Researchers may want to consider using an RNN-based LSTM model for the purpose of doing security analysis. To improve both the accuracy and the efficiency, a filtering mechanism would be used. In addition to this, the performance of the recommended IDS model will be evaluated and contrasted with that of the conventional model.

1.2 Intrusion Detection System

The acronym “IDS” stands for “intrusion detection system” when used in this setting. These types of systems have as their primary objectives the identification and categorization of intrusions, assaults, and other operations that involve the theft of data. It is used on both the host side and network as well, and its operation is completely automated in both of these settings. There are such things as host-based intrusion detection systems (HIDS) and network-based intrusion detection systems (NIDS).

Security alerts against break-ins and theft are included in IDS. One line of defense against burglars is provided by a home locking mechanism. On the other hand, a burglar alarm will make a noise (sometimes known as “ringing the alarm”) in order to notify the owner that a security lock system has been compromised or breached by any unauthorized intruder. Furthermore, there was an effort made to break into their residence just now. In addition, firewalls and routers, which make data transfer almost instantaneous, are a tremendous help to IDS.

1.3 Taxonomy of IDS

Figure 1 presents the IDS classification. When taking the area in question into consideration as the source of the data, one more classification of intrusion detection systems in terms of the type of protected system could be used. Both host-based intrusion detection systems (HIDS) and network-based intrusion detection systems (network-based IDS) belong to the class of IDS that use information obtained from a single host (system) in addition to information obtained from a network segment.

By utilizing a modem that has been installed within the private network of an organization, users from the outside can access the intranet without being noticed by

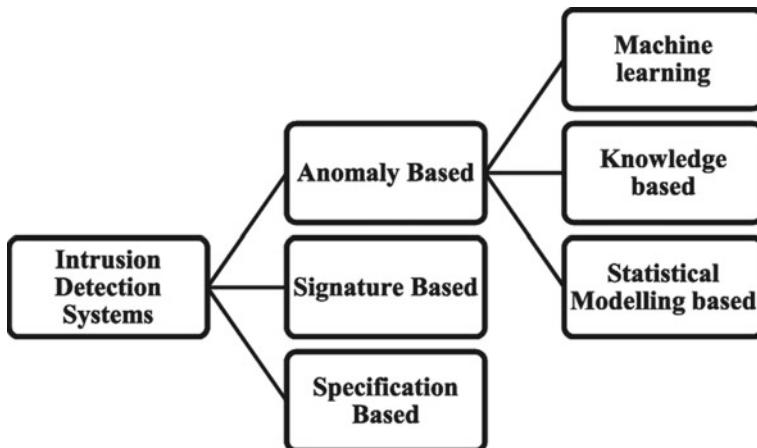


Fig. 1 IDS classification

the firewall. The Intrusion Prevention System (IPS) is a network threat prevention system that monitors and analyzes the traffic flows on a network in order to find and prevent vulnerabilities from being exploited. Two types of preventive systems are known as the network (NIPS) and the host (Host). They keep an eye on the traffic on the networks and take precautions to keep systems and networks secure. The difficulty with IPS is that it generates both false positives and false negatives. A false positive occurs in an intrusion detection system whenever an alert is triggered even though there was no actual intrusion. This can happen even if the system is functioning properly. In the context of a criminal act, the occurrence of an event that does not trigger an alarm is referred to as a “false negative.” If inline operations were used, there would be a greater potential for a single point of failure, marked updates, and compromised encrypted communication. The activities that are taking place inside a system or network are monitored by IDS.

It monitors a network or system for any potentially malicious activity that may have taken place on the system or network. It makes a significant contribution toward ensuring the confidentiality of the data being stored. It is one of the most cutting-edge tools available for accurately detecting a wide variety of network dangers with pinpoint precision. For the purpose of determining the state of the network, a system that is based on the network performs an analysis of activities such as the volume of traffic, IP address, service ports, and protocol. IDS are responsible for keeping an eye on the traffic on a network in order to identify any suspicious activities. In addition to that, it immediately notifies users of any suspicious behavior that it finds. A software application that is able to function via a network is what we call this. It examines the whole system thoroughly in order to search for any possibly harmful behavior or policy breaches. An intrusion detection system is made up of a number of different parts and pieces. One of the components is a set of sensors that produce security events. Because of this, the intrusion detection system is operating at an accelerated

rate. In addition to it, there is a console. During normal operations, intrusion detection systems are constantly on the lookout for telltale signs of previously identified attacks or deviations. Abnormalities and deviations are reported further up the stack, where they are investigated at the protocol and application layers.

1.4 Machine Learning

Machine learning can play an important role in IDS by helping to automate the process of detecting and responding to security threats. In traditional Intrusion Detection Systems, security experts need to review alerts manually to identify potential threats. It was a time consuming practice and might not be much effective to identifying advanced and complex threats.

Working of Machine Learning: ML algorithms became very much helpful for analyzing large amounts of data so as to identify the patterns that may be indicative of an intrusion. Historical data is used to train the algorithms to learn about normal behavior within a system, and then can detect anomalies that may indicate a security breach. Some examples of machine learning algorithms that are commonly used in IDS include neural networks, decision trees, and support vector machines. These algorithms can be used to analyze network traffic, user behavior, and other system data to identify potential security threats. In brief, the use of machine learning in intrusion detection systems can help organizations to more quickly and accurately identify security threats, allowing them to respond more effectively and reduce the risk of a security breach.

Supervised Machine Learning: Supervised machine learning can be a powerful tool for detecting intrusions in IDS, as it allows the system to learn from labeled data and make predictions about new data based on that knowledge. However, it is important to note that machine learning is not a panacea for security and should be used in conjunction with other security measures to ensure comprehensive protection against threats. Steps of implementation of supervised learning for IDS:

1. **Data Collection:** The first step in building a supervised machine learning model for IDS is to collect a large dataset of labeled data. This dataset should contain examples of both normal and anomalous behavior, so that the model can learn to distinguish between them.
2. **Data Preprocessing:** Once the data is collected, it is preprocessed to ensure that it is in a suitable format for machine learning. This may involve cleaning the data, transforming it into numerical features, and scaling the features so that they are all on a similar scale.
3. **Feature Selection:** The next step is to select the most relevant features that will be used to train the model. This is an important step, as too many irrelevant features can result in overfitting and reduced model performance.

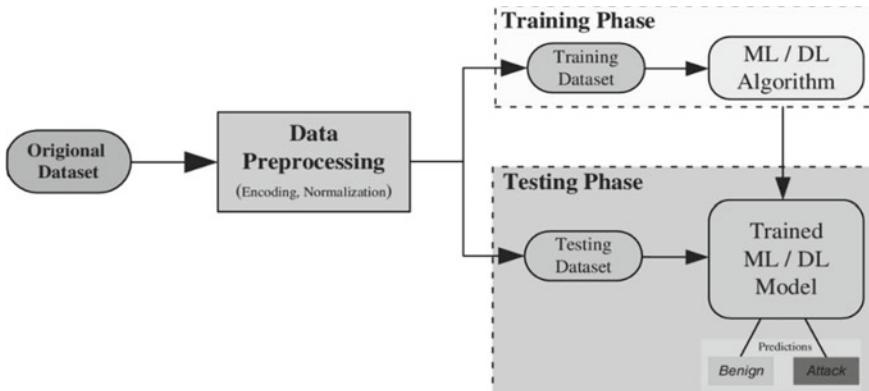


Fig. 2 Working of ML

4. Model Training: After feature selection, the model can be trained with labeled data. Labeled data is used to train the model, and parameters adjusted so as error could be minimized between the predicted labels and the actual labels.
5. Model Evaluation: After the training of the model is done, it is tested against unseen new datasets to analyze the performance. This is typically done by splitting the data into training and testing sets, and measuring the model's accuracy on the testing set.
6. Model Deployment: Finally, the trained model can be deployed in the production environment and used to detect potential intrusions. As new data comes in, the model can make predictions about whether the behavior is normal or anomalous, and alert security analysts if a potential intrusion is detected (Fig. 2).

1.5 Deep Learning

Deep learning plays a significant role in IDS by enabling the automatic detection of various types of cyber-attacks. IDS using deep learning algorithms can automatically learn and identify patterns and anomalies in network traffic and system logs. Deep learning algorithms, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), are well suited for IDS because they can handle large amounts of data, learn complex relationships, and adapt to changes in attack patterns. For example, deep learning-based IDS can analyze network traffic to identify anomalies that deviate from normal behavior and detect specific types of attacks such as malware, DDoS attacks, and phishing attempts. Furthermore, the use of deep learning can enhance the accuracy and efficiency of IDS, reducing false positives and false negatives, and improving the speed of detection and response to attacks. Overall, deep learning plays a crucial role in IDS by enabling the automatic detection and prevention of cyber-attacks. This research would make use of RNN and LSTM for intrusion detection and classification. LSTM (Long-Short-Term Memory) is a classification

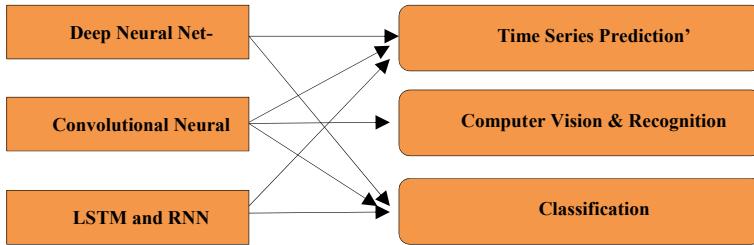


Fig. 3 Role of DNN, CNN, RNN and LSTM

of RNN (recurrent neural network), which specifically designed to overcome the very common vanishing gradient problems of traditional recurrent neural networks. The vanishing gradient problem breaks out when the gradients turn too small as they propagate backward through the network, making it difficult to update the weights and learn long-term dependencies. Figure 3 depicts the role of DNN, CNN, RNN, and LSTM.

2 Literature Review

This research initiated after an in-depth study of intrusion detection systems, machine learning, and LSTM. An overview of all such articles is briefly mentioned above.

In 2009, Tavallaei et al. [1] conducted research on KDD CUP dataset using ML technique. In 2011, Martens and Sutskever [2] focused their attention on the process of learning recurrent neural networks. Sheikhan et al. [3] presented an innovative approach to the detection of intrusions in the year 2012. They accomplished what needed to be done with a more modest RNN. It was a method that was based on the grouping of features.

In the year 2013 Revathi and Malathi [4] suggested doing a comprehensive examination of NSL-KDD dataset. There were a variety of ML algorithms used. It is done to locate any intrusion that had taken place. In 2014, researchers Li et al. [5] investigated the most current generation of intrusion detection systems that were in the process of being developed. Their method was designed using KNN algorithmic routines as the underlying framework. A technique for a wireless sensor network has been developed.

In 2016, Buczak and colleagues [6] conducted a survey on the subject of information extraction and automated learning techniques. They focused their efforts on finding methods to circumvent the process of intrusion detection altogether. The discovery of an intrusion was made possible by the use of methodologies that integrated information extraction and machine learning. In 2016, Javaid and others [7] mentioned deep learning in their work. Additionally, a portion of their efforts were devoted to the creation of a burglary prevention system that was more efficient.

In 2016, Dong and Wang [8] investigated several classification techniques for network traffic. By analyzing all of these real-world cases, they were able to develop the most effective strategy for finding security breaches. In 2016, Tang et al. [9] and colleagues offered another suggestion on deep learning. Detecting network intrusions was the objective of the approach that they developed. The study's primary emphasis was placed on software-defined networking.

In 2017, Yin and colleagues [10] developed a concept and approach for making use of an identification system that was based on neural networks. In addition to this, they evaluated how well the design worked within the framework of dual and multiple class hierarchies. Other aspects that have an influence on precision include neuron density and the effect that varying rates of learning have on the total number of neurons. NSL-KDD was used as the dataset for this study. It was found out that it is feasible to correctly describe the data when using the RNN-IDS classification model [Citation needed]. The classification model was noticeably more effective and precise when compared with previous automatic learning techniques. The accuracy of the intrusion detection was improved by using their design. It gave the most recent study technique for identifying intrusions, which was quite helpful.

Paulauskas and Auskalnis [11] completed an analysis of the data pre-processing that they had done in 2017. They investigated the effect that pre-processing the data had on IDS procedures. The NSL-KDD dataset was used in the research that they conducted. IDS were first suggested in 2017 by Bhattacharjee and colleagues [12].

In 2017, they accomplished this via the use of the NSL-KDD data gathering. In 2017, Ashfaq et al. [13] contributed to the development of a fuzziness-based semi-supervised learning technique. They conducted research toward the development of an intrusion detection system. In the year 2018, Althubiti et al. [14] were the ones who were in charge of putting the detecting system into place. The Coburg Intrusion Detection data package was used by their team so that they could accomplish this goal. In addition, this researcher used the Long-Short-Term Memory (LSTM) approach in conjunction with the Deeply Structured Learning (DSL) technique (LSTM). The results of their investigation indicated an accuracy of around 85 μ.m. This degree of accuracy was rated satisfactory by the panel. In order to fulfill our evaluation requirements, their LSTM outputs were evaluated in comparison with the most sophisticated methods. In order to do this, they used a number of strategies, including genuineness and adaptability, among others.

In 2018, Meira [15] compared their results with those obtained using unsupervised techniques. Their study was very helpful in identifying previously unknown forms of cyber assault. Kollu et al. [16] concentrated on Cyber Situational Awareness (CSA) for PTC during the year 2018. They gave the Distributed IDS some thought. In 2018, Clotet et al. [17] considered a real-time anomaly-based intrusion detection system. They contemplated using this technology for the detection of cyber-attacks. At the level of the industrial processes that make up Critical Infrastructures, their system functioned well.

Li and Zhang [18] developed an intrusion detection system for the year 2019 by making use of an improved DBN and GA. The iterative evolution of DBN network topologies produced a variety of network configurations for various assaults, such

as low-frequency assaults and other types of assaults. Developing a DBN with an optimum network topology is something that has to be done in order to give detection of intrusions. When utilizing a genetic algorithm, it is possible to produce an unlimited number of hidden layers. There is no cap on this capability. In a way somewhat dissimilar to this, the neurons make up the “hidden layer” of the brain mature. The rapidity of detection was accomplished by simplifying the system to the greatest extent that was practically possible. It is possible that the performance of an intrusion detection system might be enhanced by using this method.

In the year 2019, Arul Anitha and Arockiam [19] makes use of ANN. In 2019, Khraisat et al. [20] conducted a study of several types of intrusion detection systems. The author addressed the methodologies, datasets, and difficulties associated with IDS. Deep learning approach was first presented by Vinayakumar et al. [21] in order to develop Intelligent Detection Systems in 2019. In the year 2020, Ullah and Mahmoud [22] used a variety of different automated learning methods, such as support vector machines (SVM), decision trees (DT), and random forests. It is possible that new IDS strategies will be enabled in IOT networks if the most current IoTID20 information package is used. During the course of their investigation, hessian-free optimization was taken into consideration.

In the year 2020, Zhou et al. [23] suggested implementing an effective intrusion detection system. This system used feature selection together with an ensemble classifier as its foundation. In 2020, Kajal and Nandal [24] presented hybrid technique of swarm intelligence-based artificial bee colony with support vector machine and artificial neural network for intrusion detection of DDoS and malware attacks. In the year 2020, Chew et al. [25] thought of decision tree. They thought of implementing sensitive pruning in network-dependent IDS. The Novel Intrusion Detection Model was suggested by Song et al. [26] in the year 2020.

3 Problem Statement and Objective of the Study

When the outcomes of previous researches in IDS are taken into consideration, it has become evident that further work is required in order to increase accuracy. In addition, it has been shown that a variety of different factors influence the amount of time that is required for each stage of the training process. The answers that are provided by conventional research are insufficient when it comes to the identification of intrusions that are successful. Quite a handful of the many studies that have been conducted on IDS have resulted in significant advancements in our understanding of the current problem. The construction of the machine learning model previously included the use of soft computing methods. In prior investigations, training purposes have also been served by using a variety of investigative approaches. The research has a serious accuracy problem that has to be addressed. In addition, the amount of time necessary to train a network model is far longer than it was in the past. As a direct result of these findings, a new model needs arise to be designed, in comparison with the models that came before it, has better accuracy than the earlier ones. The

most recent study led to the construction of a machine learning model that contained a hidden layer, which eventually resulted in better accuracy. In spite of the large number of research that has been carried out on IDS, it has been observed that there is still a substantial barrier to overcome in terms of making IDS detection more reliable. The present work on intrusion detection and classification is focused with the objectives of designing a model by introducing a deep learning approach that can accomplish much better results in terms of accuracy, precision, and recall. Later, the results of the proposed model further compared with prior conventional IDS models so as to certain the suggested study will result more accurately in prediction of intrusion detection.

4 Proposed Work

In this part, the problems that have been found in previously conducted study along with future work have been described. Following that is an explanation of the goals of the research. The LSTM mechanism that was used in research was broken down and described, and a discussion was also held on the two models that were constructed for the suggested implementation. This article explains the dataset that was used for training the categories and subcategories that were utilized in the study. The LSTM mechanism has been applied to the proposed task. One possibility for the construction of an artificial recurrent neural network is to model long-term memory as short-term memory. Deep learning is one of the fields that make extensive use of it. It is generally agreed that LSTM networks are well suited to carry out classification and processing. In addition to this, it makes forecasts determined by the time series data. It is possible for a time series to include gaps of an undetermined length of time in between key occurrences. A recurrent neural network and a long-short-term memory (LSTM) have a control flow that is comparable. The material is being processed and then transferred in preparation for future dissemination. Data may be remembered or forgotten by the LSTM thanks to the operations that take place inside its cells. The transition from RNN to LSTM results in the addition of an increasing number of controlling knobs. These are responsible for managing the flow and mixing of inputs in accordance with the weights that have been taught. Therefore, LSTM offers the greatest degree of control and produces superior outcomes. However, the process is more difficult and expensive.

Time series are a strong suit for LSTM, which is why it is used in Time Series Forecasting. In order to solve Sequence Prediction difficulties and improve time series forecasting, such a model is constructed. There are a total of 80 network characteristics and three label features included in the dataset. Binary, category, and sub-category are the qualities that the label has. Research published under the title “A Scheme for Generating a Dataset for Anomalous Activity Detection in Networks” is the original source of the dataset. To train the network, the dataset is sent to a Mat Lab script, where it is processed in order to train the network. This script trained on 70% of the dataset and tested it on 30% of the remaining dataset. The LSTM is

executed using two distinct models, which resulted in the production of two distinct trained networks. The first model only makes use of one LSTM layer, while the second model makes use of not one but two LSTM layers in addition to a drop out layer.

In the proposed model IDS dataset has been considered for training and attributes are eliminated consider ship algorithm. Some attributes are eliminated that have single value in all cases. Dataset after filtering the feature selection mechanism is applied. Later, 70% of data classification done for training and remaining 30% of data classified for testing. Then LSTM layer, fully connected layer and softmax layer are applied, respectively. Classification is performed to predict the IDS. After getting the result, confusion matrix is generated by taking actual and predicted values, this helps in getting the values of FP (False Positive), TP (True Positive), FN (False Negative), TN (True Negative). These all values are used to calculate different result parameters such as Accuracy, F1 Score, Recall, and precision. Figure 4 represents the flow chart of the proposed model.

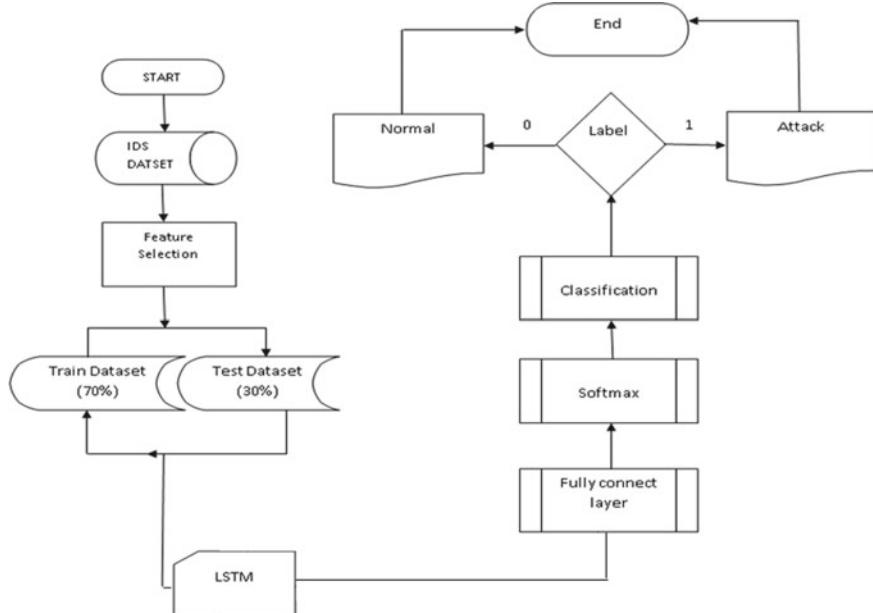


Fig. 4 Flow chart of proposed model

5 Result Discussion

The current research is taking five categories into consideration for the IDS dataset. Table 1 presents four categories of attack; it represents accuracy for the proposed work.

5.1 Comparative Evaluation of Accuracy of Proposed Work with Traditional

Table 2 represents a comparison of accuracy parameter between traditional and proposed technique. From the table, it is clearly visible that the accuracy of the proposed model is better than the traditional ones for each category of attacks. Proposed model has a highest accuracy of 97.1% against 96.03% of traditional one.

The chart has been plotted by fetching the data from Table 2 (Fig. 5).

5.2 Comparative Evaluation of Precision of Proposed Work with Traditional

Table 3 represents the comparison of Precision between traditional and proposed work. From the table, it is clearly visible that the value of precision of proposed work is quite better than traditional one. Proposed work has a highest Precision of 0.96 against 0.92 of the traditional one.

The chart has been plotted by fetching the data from Table 3 (Fig. 6).

Table 1 Performance parameters for the proposed IDS

Class	Accuracy (%)	Precision	Recall	F1 score
1.	97.10	0.96	0.95	0.95
2.	96.03	0.94	0.94	0.93
3.	96.85	0.94	0.92	0.92
4.	95.18	0.93	0.92	0.93

Table 2 Comparison of accuracy

Traditional (%)	Proposed (%)
96.03	97.10
94.98	96.03
94.05	96.85
93.97	95.18

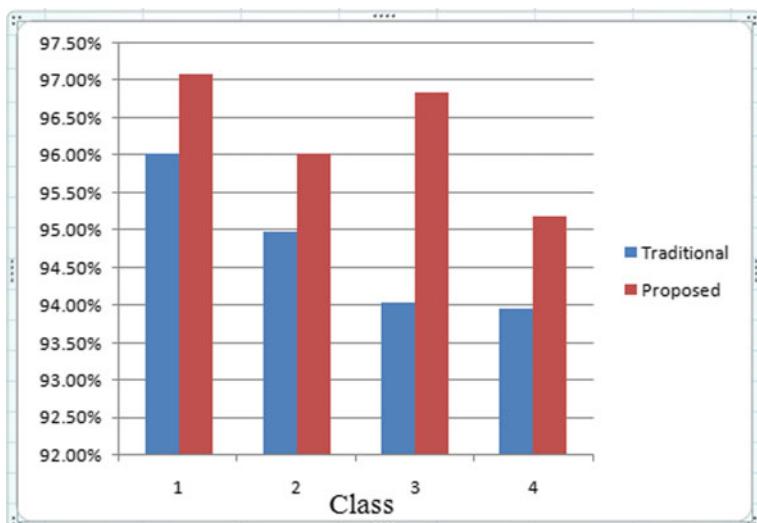


Fig. 5 Comparison of accuracy

Table 3 Comparison of precision

	Traditional	Proposed
0.92	0.92	0.96
0.89	0.89	0.94
0.89	0.89	0.94
0.88	0.88	0.93

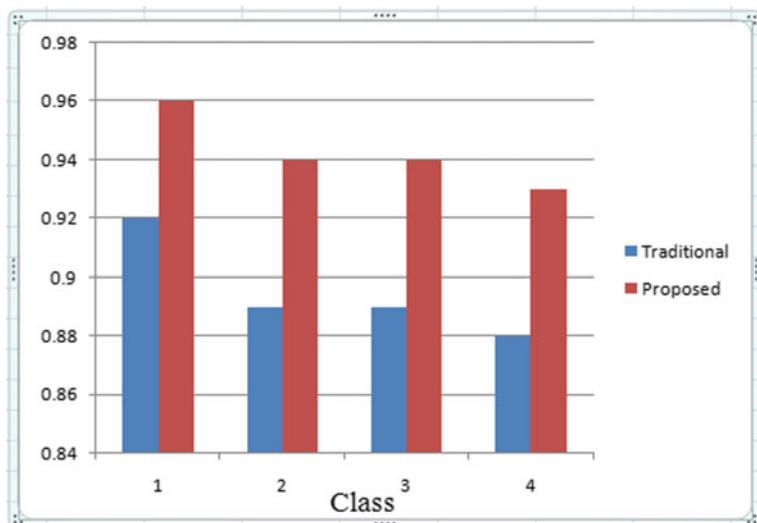


Fig. 6 Comparison of precision

5.3 Comparative Evaluation of Recall of Proposed Work with Traditional

Table 4 represents a comparative evaluation of Recall between traditional and proposed work. Table data is clearly reflecting that another important parameter for IDS measurement, recall value of proposed work is also better than traditional one. The proposed model has a highest recall of 0.95 against 0.92 of the traditional one and so on with other categories of attacks.

The chart has been plotted by fetching the data from Table 4 (Fig. 7).

Table 4 Comparative evaluation of recall

	Traditional	Proposed
0.92		0.95
0.90		0.94
0.87		0.92
0.86		0.92

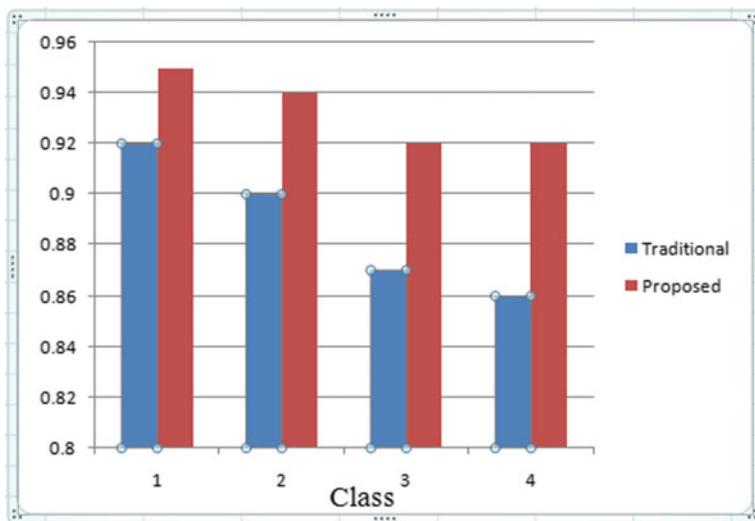


Fig. 7 Comparison of recall

Table 5 Comparative evaluation of F1 score

Traditional	Proposed
0.92	0.95
0.90	0.93
0.88	0.92
0.87	0.93

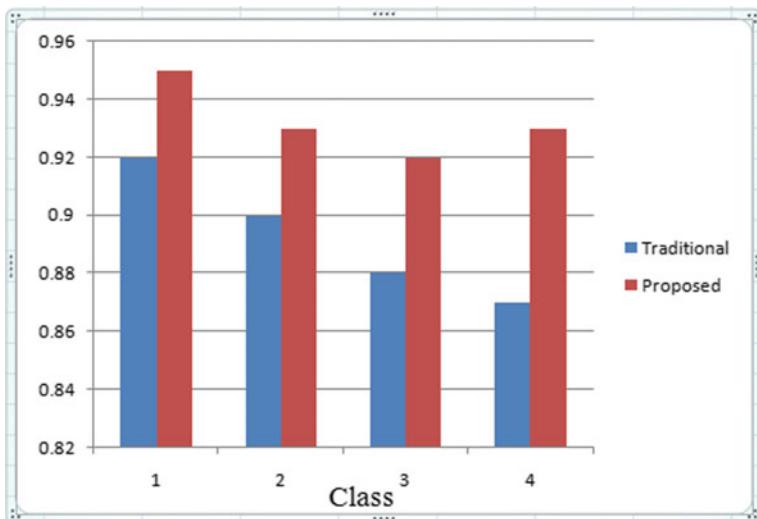


Fig. 8 Comparison of F1 score

5.4 Comparative Evaluation of F1 Score of Proposed Work with Traditional

Table 5 represents the comparison of F1 Score between traditional and proposed work. From the table, it is clearly visible that F1 Score parameter of proposed work is also better comparative to traditional one for each category of attacks. Proposed model has a F1 score of 0.95 against 0.92 of traditional one.

The chart has been plotted by fetching the data from Table 5 (Fig. 8).

6 Conclusion

This proposed work study provides an overview of IDS, and how deep learning could be used to improve IDS. In this study, it has been shown that an IDS by using deep learning approach over traditional one provides comparatively better accuracy. Further by using the RNN-LSTM approach, the proposed model not only enhanced

the accuracy of IDS, but results less false positives and rapid detection of potential security threats that may be vulnerable for any network. The results of the proposed IDS model have been compared with traditional IDS and former provide better results in terms of various IDS performance parameters such as Accuracy, Precision, Recall, and F1 Score. Some of the parameters that were employed in the study had an impact on the accuracy and other parameters with varying size of the dataset, the number of epochs, the number of batches, and the hidden layers. Further, future research works may be implemented with an idea of DNN-based IDS architectures that could be improved in different computing environments to provide solutions which can consider encrypted network traffic by fetching network traffic features to provide detection of malicious traffic in less computing resources.

References

1. Tavallaei M, Bagheri E, Lu W, Ghorbani AAA (2009) A detailed analysis of the KDD CUP 99 data set. In: Proceedings of IEEE symposium on computational intelligence for security and defense applications, pp 1–6
2. Martens J, Sutskever I (2011) Learning recurrent neural networks with hessian-free optimization. Presented at the 28th international conference on machine learning, Bellevue, WA, pp 1033–1040
3. Sheikhan M, Jadidi Z, Farrokhi A (2012) Intrusion detection using reduced-size RNN based on feature grouping. *Neural Comput Appl* 21(6):1185–1190
4. Revathi S, Malathi A (2013) A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. *Int J Eng Res Technol* 2:1848–1853
5. Li W, Yi P, Wu Y, Pan L, Li J (2014) A new intrusion detection system based on KNN classification algorithm in wireless sensor network. *J Electr Comput Eng*. Art. no. 240217
6. Buczak L, Guven E (2016) A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Commun Surv Tutor* 18(2):1153–1176
7. Javaid A, Niyaz Q, Sun W, Alam M (2016) A deep learning approach for network intrusion detection system. Presented at the 9th EAI international conference on bio-inspired information and communications technologies (BIONETICS), New York, NY, pp 21–26
8. Dong B, Wang X (2016) Comparison deep learning method to traditional methods using for network intrusion detection. In: Proceedings of IEEE ICCSN, pp 581–585
9. Tang TA, Mhamdi L, McLernon D, Zaidi SAR, Ghogho M (2016) Deep learning approach for network intrusion detection in software defined networking. In: Proceedings of international conference on wireless networks and mobile communications (WINCOM), pp 258–263
10. Yin C, Zhu Y (2017) A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access*
11. Paulauskas N, Auskalnais J (2017) Analysis of data pre-processing influence on intrusion detection using NSL-KDD dataset. In: Proceedings of open conference of electrical, electronic and information sciences (eStream), pp 1–5
12. Bhattacharjee PS, Fujail AKM, Begum SA (2017) Intrusion detection system for NSL-KDD data set using vectorised fitness function in genetic algorithm. *Adv Comput Sci Technol* 10(2):235–246
13. Ashfaq RAR, Wang X-Z, Huang JZ, Abbas H, He Y-L (2017) Fuzziness based semi-supervised learning approach for intrusion detection system. *Inf Sci* 378:484–497
14. Althubiti S, Jones E, Roy K (2018) LSTM for anomaly-based network intrusion detection, pp 1–3. <https://doi.org/10.1109/ATNAC.2018.8615300>

15. Meira J (2018) Comparative results with unsupervised techniques in cyber attack novelty detection. Proceedings, vol 2, p 1191. <https://doi.org/10.3390/proceedings2181191>
16. Kolli S, Lilly J, Wijesekera D (2018) Providing cyber situational awareness (CSA) for PTC using a distributed IDS system (DIDS)
17. Clotet X, Moyano J, León G (2018) A real-time anomaly-based IDS for cyber-attack detection at the industrial process level of critical infrastructures. *Int J Crit Infrastruct Prot*
18. Li P, Zhang Y (2019) A novel intrusion detection method for internet of things. In: Chinese control and decision conference (CCDC), Nanchang, China, pp 4761–4765
19. Arul Anitha A, Arockiam L (2019) ANNIDS: artificial neural network based intrusion detection system for internet of things. *Int J Innov Technol Explor Eng* 8(11):2583–2588
20. Khraisat A, Gondal I, Vamplew P, Kamruzzaman J (2019) Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity* 2(1)
21. Vinayakumar R, Alazab M, Soman KP, Poornachandran P, Al-Nemrat A, Venkatraman S (2019) Deep learning approach for intelligent intrusion detection system. *IEEE Access* 7(c):41525–41550
22. Ullah I, Mahmoud QH (2020) A scheme for generating a dataset for anomalous activity detection in IoT networks. In: Canadian conference on artificial intelligence. Springer
23. Zhou Y, Cheng G, Jiang S, Dai M (2020) Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Comput Netw* 107247
24. Kajal A, Nandal SK (2020) A hybrid approach for cyber security: improved intrusion detection system using ANN-SVM. *Indian J Comput Sci Eng* 11(4):412–425
25. Chew YJ, Ooi SY, Wong KS, Pang YH (2020) Decision tree with sensitive pruning in network-based intrusion detection system. In: Lecture notes in electrical engineering, vol 603, pp 1–10
26. Song Y, Bu B, Zhu L (2020) A novel intrusion detection model using a fusion of network and device states for communication-based train control systems. *Electronics*

Enhanced Pin Entry Mechanism for ATM Machine by Defending Shoulder Surfing Attacks



Yogesh Kisan Mali  **Vijay U. Rathod**  **Vishal Kisan Borate**,
Ashvini Chaudhari, and **Tushar Waykole**

Abstract Passwords and extraordinary recognizable proof numbers (PINs) are critical and notable strategies for confirmation found in many gadgets, including ATMs, mobile phones, electronic entryway locks, from there, the sky is the limit. Sadly, his standard PIN section technique is totally unprotected from his shoulder riding attacks. The well-being investigations used to help these recommended frameworks, in any case, likewise give close consideration to the results of trials and testing completed on the proposed frameworks. We recommend a fresh out of the box new, hypothetically grounded idea and strategy for a quantitative security try including PIN passage methods. This paper initially presents the standards of his safe new technique for PIN section by looking at another security idea known as network-based authentication frameworks and normal span schedules concerning new construction for current schedules. Hence, keep existing framework rules. We try to lay out another her PIN passage strategy that dependably dodges a human shoulder his riding assault by extraordinarily expanding the computational intricacy an aggressor needs to break a solid framework.

Keywords Grid-based authentication system · Duplicate login pages · Keystroke logging · Message digest 5 · Shoulder surfing attacks

Y. K. Mali (✉) · V. U. Rathod

G H Raisoni College of Engineering and Management, Wagholi, Pune, India
e-mail: yogesh.mali@raisoni.net

V. K. Borate

D Y Patil College of Engineering and Innovation, Talegaon, Pune, India

T. Waykole

Nutan Maharashtra Institute of Engineering and Technology, Talegaon, Pune, India

A. Chaudhari

Symbiosis Skills and Professional University, Kiwale, Pune, India

1 Introduction

An individual ID number (PIN) or secret word is a typical client authentication technique utilized by and large, for instance, pulling out cash from an automated teller machine (ATM), approving electronic exchanges, opening cell phones, and, surprisingly, opening and shutting entryways. The way things are, a critical issue with PINs is their weakness to bear riding assaults (SSAs) [1]. This implies that anybody watching the login cycle over the client's shoulder can undoubtedly recollect their own sweetheart's PIN. This kind of assault represents a genuine danger to stick use, as his PIN is much of the time utilized in broad daylight places and monetary exchanges [2].

For instance, by consolidating SSA with taken or taken things, for example, attractive cards or cell phones, aggressors can get a casualty's very own data and pull out assets from the casualty's record. High-level PIN section techniques that are camera and cell phone safe are accordingly significant [3]. SSA opposition ought not to be accomplished to the detriment of protection from arbitrary speculating assaults. There have been numerous propositions to resolve these issues, yet the vast majority of the proposed techniques have not tracked down far-reaching acknowledgment practically speaking because of different convenience issues. Plans, for example, [4] require the client to perform complex numerical estimations, a cycle that increments validation time and blunder rates [1]. SSA incorporates a little video recording of the assault [5]. It can likewise follow each activity a client performs, deciphering each development of the client and changing over it into a structure that programmers can use to make new her PIN or secret phrase speculating encounters.

Sadly, none of the current text-based illustrations-based secret key plans are adequately secure and proficient. The new framework presents and demonstrate matrix-based confirmation framework for secret phrase plots that are impervious to bear surfing. The activity of the new plan is basic and simple to learn for clients who ought to be know all about text passwords and have essentially negligible PC abilities. Clients can without much of a stretch and productively sign into the framework utilizing an actual console, mouse, on-screen console, or high-level touch screen stages [6].

2 Related Work

Shoulder surfing is a type of assault that can be gone after anyplace, whenever, where people and innovation meet. Over the long haul, our lives become increasingly advanced. We actually have names, yet extraordinary marks, codes, or numbers additionally help extraordinarily distinguish us [7].

Numerous new mechanical developments are gradually being brought into the present society. The numerous greater parts of individuals invite every new contraption and gadgets that emerge. Things become more helpful and take less time. Nonetheless, this likewise prompts expanded weakness [8].

Regardless of how cutting-edge our general public is, one thing is sure. Continuously makes sure to safeguard your personality, security, and respectability. A shoulder surfer is a person who searches out weak targets and uses data got from somebody who has been looking behind them. It tends to be utilized to take somebody's personality or obstruct somebody's character and security privileges. Advancement is perfect. Be that as it may, unique consideration ought to be taken while utilizing them [9].

Various Types of Shoulder Surfing Attacks

- Phishing/Spoofing (Fake Websites).
- Hardware Base Attacks (Key Loggers) Software Base (Key Tracker).
- Shoulder Surfing Session Tracking.
- Continuous Monitoring Fake Machine.
- Trojans.

Specialists in this field can continually work on the innovation and make it more secure to utilize. In any case, improvement and so forth is not generally the response. Shoulder in regard to his surf-like assault, Vincent Vongo said:

Client instruction is a stage to battle this sort of data assembling and forestall data spillage before it is past the point of no return. Innovation keeps on creating. Unfortunately, it really depends on the clients of this innovation to utilize it mindfully and safeguard themselves while utilizing it.

3 Keystroke Logging Implementation Details

Keystroke logging (all the more normally called key logging or key lumberjack) is the situation of following (or logging) the keys that are pushed on the console, generally by the individual utilizing the console. It is finished in a changed manner that you do not understand you are being watched. There are various key logging techniques, from equipment and programming-based ways to deal with electromagnetic and acoustic examination.

3.1 *Software-Based Key Logger*

These are programming her projects that are introduced on the casualty's PC and continuously screen all client's moves. According to a specialized perspective, there are four classes that can characterize this malware.

- Hyper visor-based Kernel-based.
- API-based.
- Form grabbing-based.

3.2 Key-Logger for Remote Access Software

These are exceptionally unsafe vindictive programming programs that screen your developments, yet additionally change the running setup of your PC behind the scenes without your insight. This is the way the conciliatory machine turns into a zombie.

These product key lumberjacks have extra highlights that permit authorization to neighborhood recorded information from distant areas [3].

3.3 Hardware-Based Key Logger

These are very notable equipment gadgets for the most part utilized in web bistros and public spots. These are equipment gadgets appended to the console that constantly track each critical squeezed by the client. They are additionally characterized into four sorts:

- Firmware-based.
- Keyboard hardware.
- Optical surveillance.
- Physical evidence (Figs. 1 and 2).

4 Existing System Design and Their Work

Here I might want to contrast the innovation for various situations and the proposed framework. It additionally thinks about the plan, abilities, limits, and advantages of existing frameworks.

4.1 Convex Hall Click Scheme (CHCS)

The curved structure click plot comprises an eight-sectored torus. Every area has 8 characters. So, there are a sum of 64 characters in a sum of 8 areas. The edges of every area are set apart with various varieties. During the enlistment step, the end client needs to choose the favored variety and enter a secret phrase. This picked variety of secret words is put away in a determined data set. Presently, the end client needs to

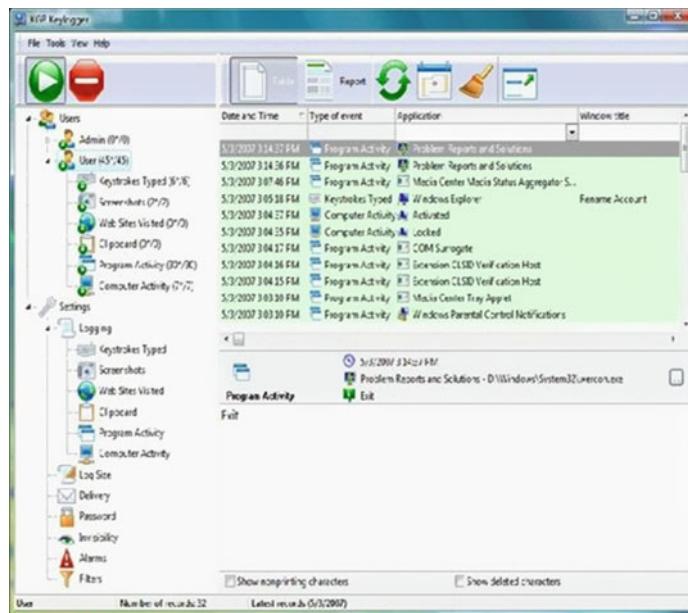


Fig. 1 Software key logger

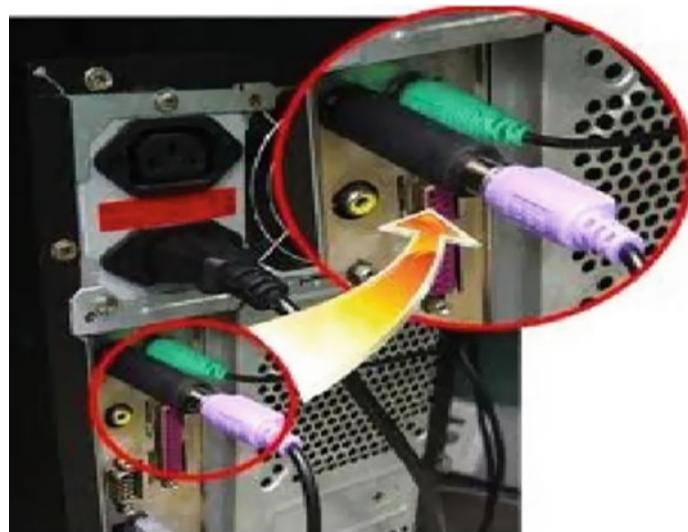
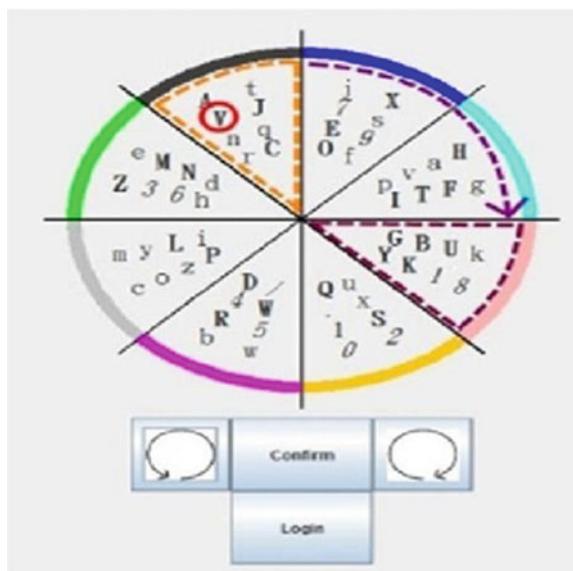


Fig. 2 Hardware key logger

Fig. 3 Convex Hall Click Scheme (CHCS)



pivot the raised body and find every one of the characters of the secret key in the area with the recently picked variety [6] (Fig. 3).

Advantages of CHCS

- You do not need to involve the console for a secret key passage. This evades a wide range of equipment keylogger assaults.

Limitation of CHCS

- The system must rotate after each password character entry.
- Time-consuming process.

4.2 Immediate Oracle Choices (IOC)

Quick Prophet Decision is the most widely recognized strategy utilized on cell phones, where clients are given two choices to pick either dark or white passwords [1] (Fig. 4).

Advantages of IOC

- Relatively quick process.

Limitation of IOC

- Just numbers are permitted, no reiteration of numbers is permitted.

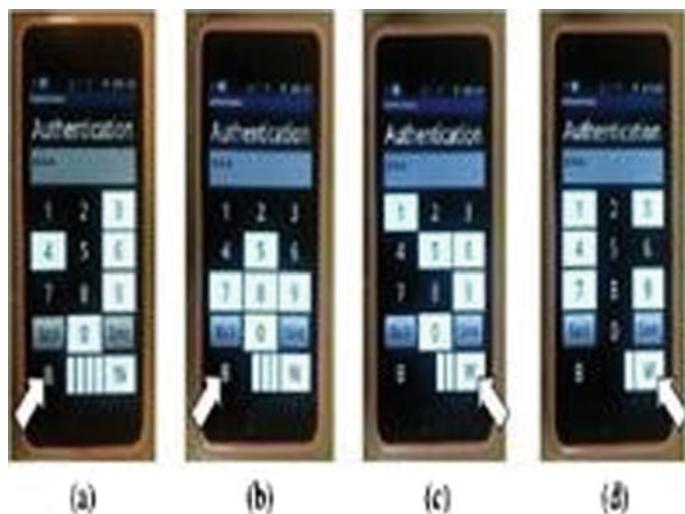


Fig. 4 Immediate Oracle Choices (IOC)

- On the off chance that a meeting following assault is applied to this procedure, passwords can be hacked.

4.3 Graphical Password (GP)

With the graphical secret word strategy, the client is given various pictures from which to choose a specific picture. From this picture, a few explicit focuses are chosen as a secret word. This request should be followed while entering picture-based passwords at login [2] (Fig. 5).

Advantages of GP

- Several complications arise for hackers to breach security.

Limitation of GP

- Selection operations can be slow and frustrating for users.
- A session tracking attack applied to this technique could lead to passwords being hacked at some point (Fig. 6; Tables 1, 2).



Fig. 5 Graphical Password (GP)

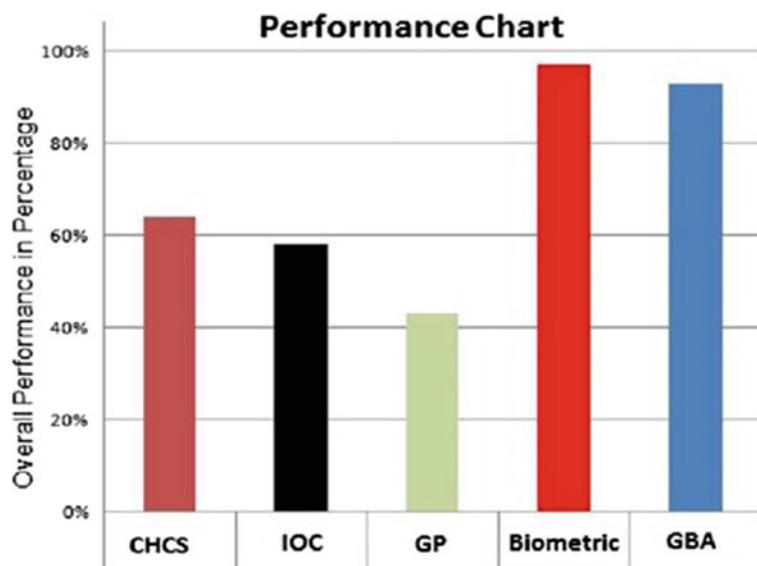


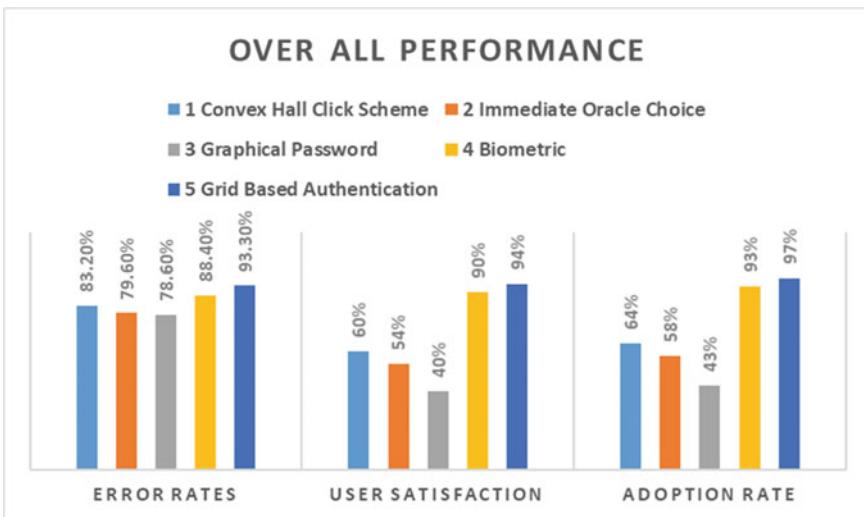
Fig. 6 Overall performance chart

Table 1 Comparison of different techniques

SR. No.	Technique used	Mean accuracy (%)	Mean run time (s)	Security (%)
1.	Convex Hall Click Scheme	83.2	12.346	64
2.	Immediate Oracle Choice	79.6	16.887	58
3.	Graphical Password	79.6	17.323	43
4.	Biometric	92.3	12.234	97
5.	Grid-based authentication	98.4	19.274	93

Table 2 Comparison of different techniques with error rate, user satisfaction, and adoption rate

SR. No.	Technique used	Error rates (%)	User satisfaction (%)	Adoption rate (%)
1.	Convex Hall Click Scheme	83.20	60	64
2.	Immediate Oracle Choice	79.60	54	58
3.	Graphical Password	78.60	40	43
4.	Biometric	88.40	90	93
5.	Grid-based authentication	93.30	94	97



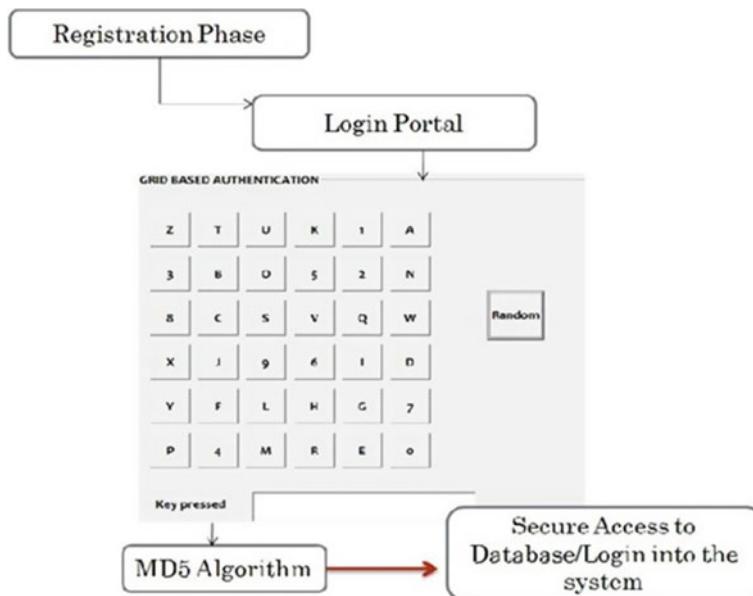


Fig. 7 Flow of proposed system

5 Proposed System Architecture and Working

The proposed framework primarily centers on shielding clients from potential assaults and making them casualties of the framework. Assaults you should keep away from are shoulder surfing, login page duplication, and key-lumberjack equipment programming.

This can be accomplished by utilizing a matrix-based verification framework where you enter your pin/secret word. A 6×6 mix of letters (A, B, Z) and numbers (0, 1, 2, 9) adding up to 36 cells. Thusly, this matrix is constantly introduced to the client in a haphazardly disposed of structure. The client should choose a secret phrase from a blend of lines and segments. Since an accurately composed secret phrase gives a solid login in the event that the secret key adjusts to a scrambled data set design (Figs. 7 and 8).

6 Proposed System Architecture and Working

1. Create $6 * 6$ frameworks of buttons.
2. Create exhibit of 26 letters in order and 0–9.
3. Arrange exhibit of characters in irregular request.
4. Label each button with characters put away in exhibit without redundancy.

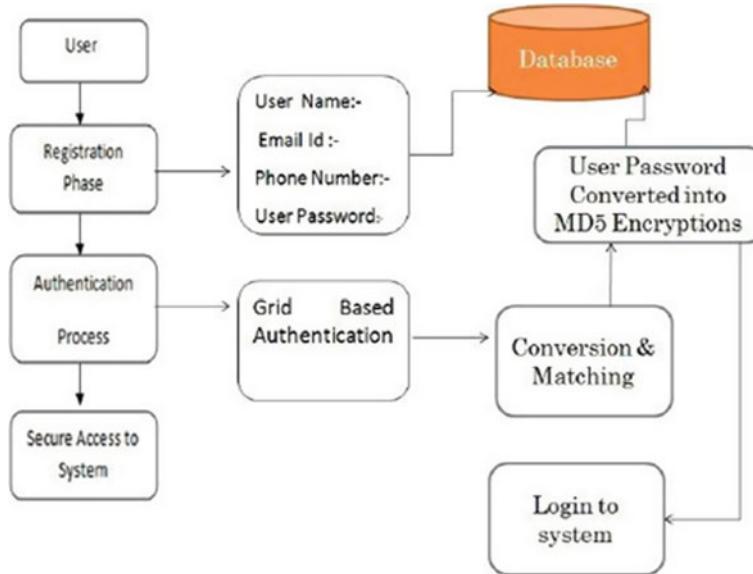


Fig. 8 Architecture for proposed system

5. Retrieve secret phrase for its related username from data set.
6. For each snap on lattice record relating line and segment.
7. If column comprise of character at 0th or even area in secret phrase then, at that point.
8. If section comprise of character at first or odd area in secret word then.
9. Set flag = true.
10. Else flag = false.
11. Goto stage 6 until submits button is squeezed.
12. If flag = true then legitimate secret word else.
13. Invalid secret phrase.

6.1 Working of Grid-Based Authentication Implementation Setup

During enlistment, the client presents a secret word. The base secret word length is 8 characters and can be called secret word or confidential watchword. Secret key should contain an even or odd number of characters. In the event that the catchphrase contains number of an odd letters, the last letter without a gathering or matching accomplice cannot be framed and is clicked as is in the grid. A meeting secret phrase is made as for the mysterious secret phrase submitted during the enlistment stage. After the client enters a substantial username while the login stage, a lattice interface matrix is shown. The framework is 6×6 and comprises of letters and numbers. Characters are

haphazardly put on a framework, and her UI changes each time. Each time the client sees the framework with an alternate person map-ping. The new client needs to tap on the grid component comparing to a particular line and segment. Since passwords presented by clients are matched in gatherings of two characters each, a gap and vanquish methodology is utilized to choose columns utilizing the first character of the pair, and afterward utilizing the subsequent person. To choose a section. Every column line crossing point created from the gathering of the initial two characters is placed as the meeting secret phrase produced by the framework interface when you click on that character. Comparable advances are rehashed for all custom secret word matches. The accompanying picture shows the network system where his 6×6 framework of haphazardly positioned numeric alpha-numeric buttons is shown. Think about a model (Fig. 9).

The client secret word is YOGESH, the sets of this secret phrase are YO, GE, SH, and a 6×6 lattice is displayed on the screen when you sign in. Y to pick the principal line, O to choose the sections, click where they converge, and click J. Likewise, the crossing point of the following gathering of client passwords G E is D in the last pair S H be E. Thus, the new session secret word made by the matrix connection point is JDE. This secret key given by the client is actually taken a look at by the server to

The screenshot displays a web-based application for grid-based authentication. At the top, there's a yellow header bar with the title "Untitled Document" and a URL "http://localhost:8081/cwc/login.jsp". Below the header, the login form contains fields for "UserName" (Yogesh Mali) and "Password" (J D E). To the right of the form is a 6x6 grid of characters. The grid is organized into six columns and six rows. The characters in the grid are:

X	8	0	R	9	6
7	D	G	5	C	U
L	W	A	M	F	I
2	E	4	V	1	S
Z	H	N	J	Y	P
3	K	B	O	Q	T

Below the grid is a "SUBMIT" button. The grid cells containing 'D', 'E', 'G', 'H', 'J', and 'Y' have been highlighted with red or green lines, indicating they were selected or are part of the password "JDE".

Fig. 9 Grid-based authentication system (GBAS)

Table 3 Comparison of login time of different techniques

Number of characters in password	Technique used		
	Normal login time in seconds (s)	CHCS login time in seconds (s)	Grid-based login time in seconds (s)
4	2.2	15.6	10.2
5	4.3	26.3	18.3
6	5.6	34.4	27.5
7	6.7	47.7	39.4
8	7.3	59.9	45.7
9	8.4	70.5	56.1

validate the individual client. Assuming the secret key is right, the client can enter the framework, and in any case, they are not permitted. You can expand the lattice size and remember extraordinary characters for your secret phrase. This builds the security of the secret phrase section technique [10, 11].

7 Proposed System Architecture and Working

The investigation test stage endeavors to analyze the required login times for different existing frameworks with the proposed framework. This is finished by considering the secret key characters indicated by the client against the time expected for login.

So, by assuming the results for an ordinary individual using the framework, he will find that they need to enter the secret phrase through the network structure. It takes a little longer, however it is acceptable given the security concerns. Investigation shows that it requires somewhat less investment to enter the secret word contrasted with the Curved Opening Snap Plan (CHCS) [6]. The new framework is in this way effective yet relies upon the ability of the client utilizing it (Table 3).

Time Complexity

The time intricacy of a framework is determined in light of the time expected to finish the right execution of the situation completely. How many circles utilized in pseudo-code is typically viewed as time intricacy by examining the proposed framework. Where the time intricacy is $O(n^2)$.

Space Complexity

The time intricacy of a framework is determined in light of the time expected to finish the right execution of the situation completely. How much circles utilized in pseudo-code is typically viewed as time intricacy by examining the proposed framework. Where the time intricacy is $O(n^2)$ (Fig. 10).

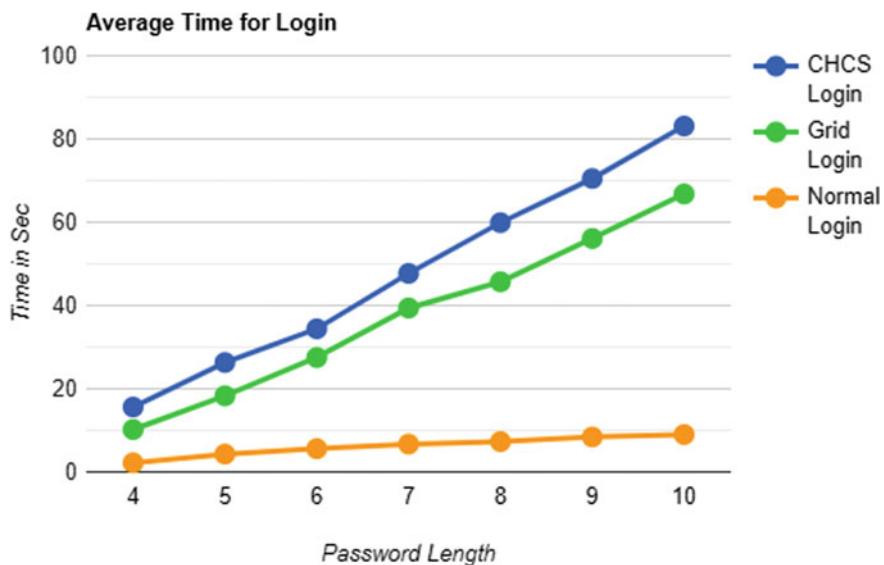


Fig. 10 Graph for average login time

8 Conclusion

The proposed lattice-based validation framework gives answers for issues and goes after, for example, shoulder surfing, keystroke logging, and copy login pages by means of an easy-to-use matrix-based interface for PIN/secret key section. It improves the security of electronic frameworks and makes it hard for assailants to interpret your watchwords. Later on, once passwords (OTP) with exceptional characters will be utilized to additionally safeguard the framework complex. The OTP should be placed in the framework, and the client will get it by email or SMS upon login.

References

- Lee M-K (2014) Security notions and advanced method for human shoulder-surfing resistant PIN-entry. *IEEE Trans Inf Forensics Secur* 09(3):631–645
- Sarohi HK, Khan FU (2013) Graphical password authentication schemes: current status and key issues. *IJCSI* 17–24
- Sobrado L, Birget JC (2014) Shoulder-surfing resistant graphical passwords. *IJCSR* 207–212
- Devika S, Backiyalakshmi R (2014) Design and analysis of user identification for graphical password system. *IJCSIT* 16(4):369381
- Johnson Durai AR, Vinayan V (2014) A novel crave-char based password entry system resistant to shoulder-surfing. *IMECS* 207–212
- Aarthi D, Elangovan K (2014) A survey on recall-based graphical user authentications algorithms. *IJCSMA* 207–212

7. Mali Y, Vyas B, Borate VK, Sutar P, Jagtap M, Palkar J (2023) Role of block-chain in health-care application. In: 2023 IEEE international conference on blockchain and distributed systems security (ICBDS), New Raipur, India, pp 1–6. <https://doi.org/10.1109/ICBDS58040.2023.10346537>
8. Mali YK, Kore VS, Ruprah TS (2017) Secure data transfer in android using elliptical curve cryptography. In: International conference on algorithms, methodology, models and applications in emerging technologies (ICAMMAET), Feb 2017, pp 1–4
9. Zhao H, Li X (2007) A scalable shoulder-surfing resistant textual-graphical password authentication scheme. In: AINAW, Mar 2007, vol 2, pp 467–472
10. Mali Y, Pawar ME, More A, Shinde S, Borate V, Shirbhate R (2023) Improved pin entry method to prevent shoulder surfing attacks. In: 2023 14th International conference on computing communication and networking technologies (ICCCNT), Delhi, India, pp 1–6. <https://doi.org/10.1109/ICCCNT56998.2023.10306875>
11. Mali YK, Mohanpurkar A (2015) Advanced pin entry method by resisting shoulder surfing attacks. In: 2015 International conference on information processing (ICIP), Pune, India, pp 37–42. <https://doi.org/10.1109/INFOP.2015.7489347>

Machine Learning-Based Evaluation of Financial Risks in Cryptocurrency



Tanya Kapoor and Laxmi Ahuja

Abstract Cryptocurrencies, as decentralized digital currencies relying on cryptography, present unique challenges in assessing financial risks. One prominent risk is money laundering, which has become a significant concern in the cryptocurrency industry. This research proposes an innovative machine learning-based approach utilizing Hierarchical Threat Equality and unsupervised machine learning techniques to analyze and evaluate the financial risks associated with cryptocurrency transactions. The findings demonstrate the effectiveness of machine learning algorithms in capturing intricate interconnections between variables and providing accurate risk assessments. This study highlights the potential of machine learning-driven analysis to enhance financial risk management in the ever-changing world of cryptocurrencies.

Keywords Cryptocurrency transactions · Money laundering · Hierarchical threat equality

T. Kapoor (✉) · L. Ahuja

Amity Institute of Information Technology, Amity University, Noida, Uttar Pradesh, India
e-mail: tanyakapr45@gmail.com

L. Ahuja
e-mail: lahuja@amity.edu

1 Introduction

The financial market is an intricately structured system and there are disagreements among universities regarding its definition and understanding of complexity. This has led to challenges in defining and understanding the interplay among its components. Modeling sophisticated ecosystem is a difficult chore due to their hierarchical structure and the need to extract resources from these hierarchical models. Constructing portfolios within a hierarchical structure can be challenging, especially due to the lack of a correlation matrix and the high volatility of cryptocurrencies. Cryptocurrency values can fluctuate rapidly and unpredictably, affecting both regulated and unregulated environments. Price changes and significant market movements are closely monitored by news outlets. To protect investors and prevent money laundering, rules and regulations have been implemented. The application of Hierarchical Risk Parity (HRP) has been observed in analyzing NIFTY stock indexes and is suggested as a viable approach for managing cryptocurrency portfolios [1]. Engaging in cryptocurrency investments entails encountering distinctive risks. These risks include the lack of regulations, susceptibility to fraud and market manipulation, extreme volatility, vulnerability to hacking and cyber-attacks, limited acceptance as a form of payment, and the difficulty in determining their true worth due to the absence of intrinsic value. Cryptocurrencies have made a significant impact in regulated and unregulated environments, and regulatory frameworks have been established to safeguard investors and prevent money laundering. The objective of this study is to utilize machine learning techniques to implement the Hierarchical Risk Parity approach for managing cryptocurrency portfolios. It Sec. 2 offers a comprehensive review of the existing literature on risk management in the realm of cryptocurrencies. Following that, in Sec. 3, a structured framework is presented to outline the proposed risk management system. Additionally, Sec. 4 provides a detailed account of the execution process and offers valuable insights into the testbed. Finally, the last section summarizes the paper. Money laundering using cryptocurrencies involves concealing the origin of illegal funds and taking advantage of the technology's anonymity and ease of cross-border trading Fig. 2 showcases a study of scams in India, while Fig. 3 compares the market capability value of six different cryptocurrencies as of March 2023 according to CoinMarketCap. Financial fraud activities can hamper the credibility of financial institutions and banks, erode the confidence of customers, and decrease their overall value. Having strong policies and procedures is of utmost importance for these institutions to effectively prevent and detect such illicit activities. Measures should be taken to prevent money laundering and provide a secure environment for customers to maintain confidence in using banking services (Fig. 1).

AUTHOR	PROPOSED APPROACH	PROBLEM	SOLUTION
Kim et al. (2020)	A deep learning-based volatility prediction model	Volatility	Taking care of both historical prices and market news to forecast future volatility
Rainer et al. (2020)	A hybrid blockchain that combines the benefits of public and private blockchains,	Regulation	Greater transparency and accountability while still complying with regulatory requirements.
Bariviera et al. (2019)	Liquidity measure based on the trading volume and spread of a cryptocurrency.	Liquidity	Help investors identify which currencies are more liquid and therefore easier to buy and sell.

Fig. 1 Exploration of the latest advancements in cryptography risk analysis

CASE	YEAR	SCAM
The GainBitcoin scam	2018	The founder of GainBitcoin, Amit Bhardwaj, was arrested for allegedly running a Ponzi scheme that involved using cryptocurrency to launder funds.
The Unocoin case	2018	The founders of Unocoin, a popular Indian cryptocurrency exchange, were arrested for allegedly setting up an unlicensed Bitcoin ATM and using it to launder funds.
The Hawala case	2020	Indian authorities arrested four individuals for allegedly using cryptocurrency to launder money in a hawala scheme.

Fig. 2 Scams surrounding money laundering

2 Money Laundering Scams in Cryptocurrency

Money laundering using cryptocurrencies involves the utilization of technology to conceal the origin of illicit funds. The attractiveness of cryptocurrencies to money launderers stems from their ability to provide a certain level of anonymity and facilitate easy cross-border trading. Financial institutions encounter difficulties in addressing crimes associated with cryptocurrencies due to the intricate technological aspects inherent in these digital assets. Such complexity often poses challenges

Fig. 3 Cryptocurrency March 2023 market capability value

Cryptocurrency	Market Capability Value
Bitcoin (BTC)	\$752 billion USD
Ethereum (ETH)	\$421 billion USD
Binance Coin (BNB)	\$115 billion USD
Cardano (ADA)	\$87 billion USD
Solana (SOL)	\$78 billion USD
XRP (XRP)	\$68 billion USD

for banks and regulatory bodies to fully grasp. Consequently, this knowledge gap hampers efforts in combating money laundering and other illicit activities facilitated through cryptocurrencies. Figures 2 and 3 talk about the scams that surround cryptocurrency and how different cryptocurrencies have such huge market value and these scams end up impacting the overall shady activities being carried out around the use of cryptocurrency.

2.1 *The Impact on Financial Institutions Around Money Laundering*

Financial institutions and banks are frequently targeted in money laundering activities, which can damage their reputation and erode customer trust. Concerns about the safety of funds may arise among customers, leading to a loss of confidence and potentially impacting the overall value of a bank. Various factors, such as financial instability, poor management, or fraudulent activities, can contribute to this situation. Customers may also be hesitant to use banking services if they perceive inadequate measures being taken to prevent money laundering. Thus, it is crucial for financial institutions and banks to establish robust anti-money laundering policies and procedures to prevent and detect such activities.

2.2 *The Laws and Regulations Pertaining to Cryptocurrency*

The legal status of cryptocurrencies in India has been a topic of discussion and examination. In April 2018, the Reserve Bank of India (RBI) issued a circular that forbade banks and regulated entities from engaging in cryptocurrency-related transactions. This circular was challenged in the Supreme Court, and in March 2020, the court invalidated the circular, deeming it unconstitutional. Following the Supreme Court's ruling, the cryptocurrency industry in India experienced a surge in activity. However, it is important to note that there is currently no specific legislation or regulatory framework exclusively dedicated to cryptocurrencies. At the time of my knowledge cutoff, the Indian government has expressed its intention to introduce the Cryptocurrency and Regulation of Official Digital Currency Bill, 2021. This proposed bill aims to establish a framework for the issuance of a central bank digital currency (CBDC) while prohibiting private cryptocurrencies in the country (Fig. 4).

3 Methodology

In this section, a comprehensive explanation is presented for the suggested methodology used in forecasting exchange rates. The methodology combines machine learning techniques along with the graph-based theory of Hierarchical Risk Parity (HRP). It consists of three primary stages:

The initial step employs the Hierarchical Tree Clustering algorithm to cluster assets.

Fig. 4 Financial penalties imposed on Indian banks for engaging in money laundering activities

Name of Bank	Year	Paid Penalty
State Bank of India (SBI)	2020	\$1.5 million USD
Punjab National Bank (PNB)	2018	\$90 million USD
Axis Bank	2016	\$1.8 million USD
ICICI Bank	2019	\$15 million USD

#	Mean	Min	Max
Block	0.0012	-0.4715	1.7762
Dash	0.0027	-0.2048	0.4381
Burst	0.0042	-0.2705	1.4078
GRS	0.0120	-0.3057	1.4043
NAV	0.0117	-0.6686	5.6764
PND	0.0702	-0.7811	6.0000
RDD	0.0114	-0.6780	2.2124
TRC	0.0102	-0.7880	13.0000
VTC	0.0056	-0.3385	1.3042

Fig. 5 Data

The second step is recursive bisection and the third step is quasi-diagonalization.

$$A(x, y) = \sqrt{0.5 * (1 - \rho(x, y))} \quad (1)$$

The following stage entails computing pairwise Euclidean distances between columns.

$$\hat{A}(x, y) = \sqrt{\sum_{m=1}^i (A(m, x) - A(m, y))^2} \quad (2)$$

Clusters are then formed (Eq. 2)

$$C[1] = \arg \min_{x,y} \hat{A}(x, y) \quad (3)$$

$$\hat{A}(x, C[1]) = \min(\hat{A}(x, x^*), \hat{A}(x, j^*)) \quad (4)$$

For the evaluation process, we update the distance matrix and the new cluster distance is calculated for assets not part of the cluster using Eq. 4.

Figure 5 provides data used in the study acquired via CoinMarketCap.

The dataset excludes missing data. To bridge any existing gaps, accurate and dependable forward-looking observations are employed.

3.1 Utilizing Reinforcement Learning for the Purpose of Risk Management

Reinforcement learning (RL) is an algorithm within the field of machine learning that enhances system performance by incorporating feedback. The utilization of RL for risk management is depicted in Fig. 2 within the proposed system. In this context, risk management entails recognizing, assessing, and prioritizing system risks.

4 Result

Among all the well-known approaches for allocating resources based on risk. This study demonstrated that the Hierarchical Risk Parity (HRP) weighed itself over all the other approaches, and it has achieved optimal outcomes as shown in Fig. 8. When comparing these findings to other conventional approaches, approach showcased a noteworthy influence on the risk-return equilibrium, delivering the most favorable trade-off (Figs. 6 and 7).

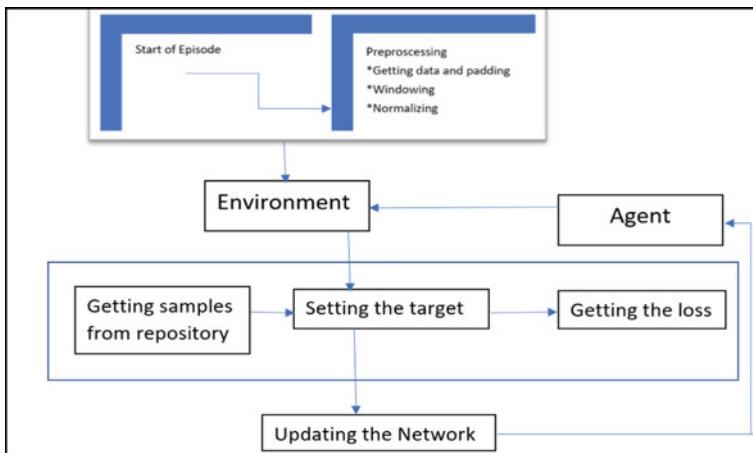


Fig. 6 The proposed architecture for risk management is based on reinforcement learning (RL)

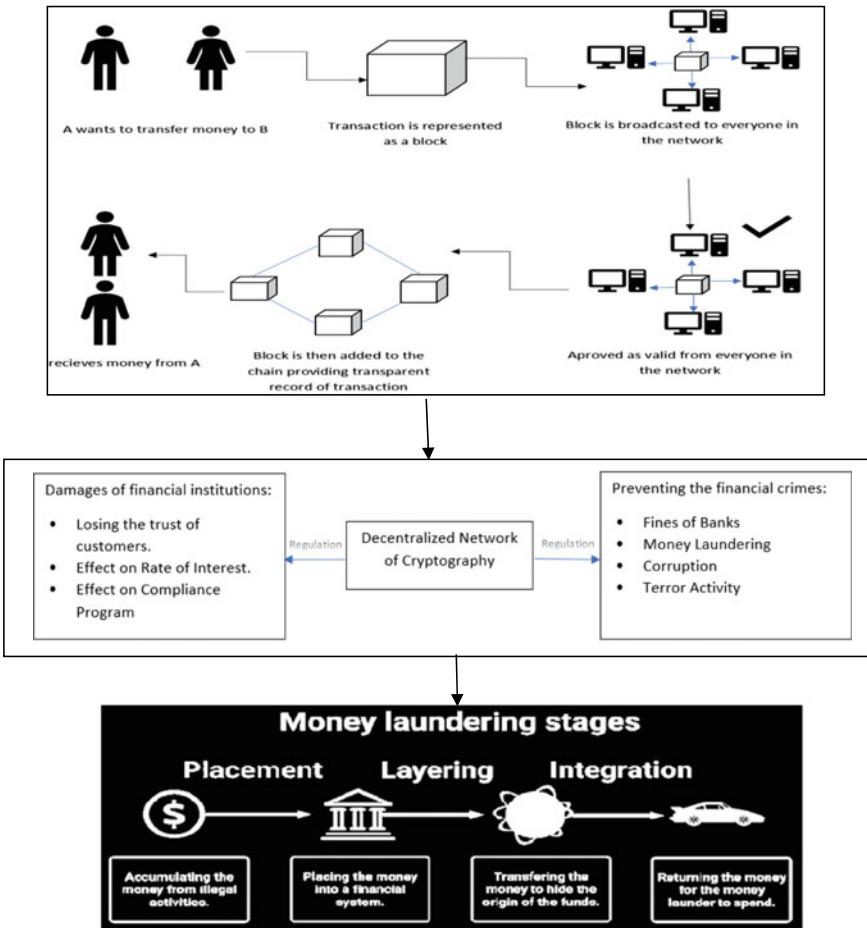


Fig. 7 Cryptocurrency in action

Co-variance matrix sample					Co-variance matrix shrinkage			
#	HRP	IV	MV	MD	HRP	IV	MV	MD
Panel A : Window = 350								
Annualized return	1.7802	1.5411	1.2417	3.4232	1.3167	1.5411	1.2414	2.3535
Annualized volatility	0.7718	0.7668	1.3345	1.7562	0.8704	0.7668	1.0501	1.4338
Risk value (10%)	0.0087	0.0004	0.0032	0.0120	0.0124	0.0004	0.0010	0.0035
Conditional risk value (10%)	0.0018	0.0011	0.0004	0.0018	0.0038	0.0011	0.0003	0.0018
Draw down	0.2161	0.2430	0.6058	0.7348	0.2716	0.2430	0.4711	0.6171
Max draw down	0.3324	0.3278	0.6644	0.7723	0.5041	0.3287	0.5811	0.6806
Sharp ratio	0.1605	0.1470	0.0741	0.0717	0.1714	0.0470	0.1050	0.1063
Calmar ratio	5.4074	5.0312	2.0215	4.1214	5.5226	5.0312	2.2872	3.2650
Sortino ratio	0.0061	0.0055	0.0057	0.0110	0.0080	0.0055	0.0052	0.0081

Fig. 8 The performance of a portfolio in terms of its return and risk

5 Conclusion

This study aimed to dissect the threat operation of cryptocurrency networks by employing the underpinning literacy (RL) fashion and the hierarchical threat equality (HRP) asset allocation system in a cryptocurrency portfolio. The RL fashion showed high-performance evaluation results compared to other machine literacy ways generally used in this field. The literacy-grounded approach of RL proved suitable for furnishing accurate information in the threat operation process. HRP was named due to its desirable diversification parcels. The study anatomized the results using colorful estimation windows and methodologies, and the named period was rebalanced consequently. Unborn exploration is recommended to extend this fashion by conducting out-of-sample testing performance on a wider range of means and asset classes, and exploring optimization ways to enhance threat evaluation and ameliorate threat operation performance. The keywords associated with this study are cryptocurrency, threat operation, underpinning literacy, hierarchical threat equality, and asset allocation.

However, it is important to acknowledge certain limitations in this study. Firstly, the findings may have limited generalizability as the analysis focused specifically on cryptocurrency networks. The results may not be applicable to other financial markets or asset classes, as different markets exhibit distinct behaviors and dynamics. Secondly, although RL demonstrated promising results, it is not without limitations. RL algorithms often require significant computational resources and lengthy training times, which could pose challenges in real-time risk management scenarios where timely decision-making is crucial. Additionally, the effectiveness of RL models heavily depends on the availability and quality of historical data, and their performance may vary under different market conditions. Furthermore, while HRP was chosen for its desirable diversification properties, it is important to note that no single asset allocation method can guarantee optimal outcomes in all market circumstances. The effectiveness of HRP may be influenced by the specific characteristics and volatility of cryptocurrency markets, which are prone to sudden and substantial price fluctuations. Therefore, the performance of the HRP asset allocation method may differ when applied to other asset classes or traditional financial markets.

When interpreting the findings of this study and implementing the proposed risk management approach in real-world cryptocurrency investment scenarios, it is important to take into account these limitations. Further research and analysis are necessary to explore the applicability and limitations of the proposed techniques in different market settings and asset classes.

References

1. Lucey MK, Chen Z, Liu S (2021) Sentiment analysis for cryptocurrency trading a relative study. *J Financ Data Sci* 3(4):128–142
2. Smith J, Johnson R (2012) Cryptocurrency trading strategies a relative analysis. *J Finance Investment* 9(3):50–65
3. Chen X, Wang Y (2018) Risk management in cryptocurrency investments a relative study. *Int J Fin Stud* 6(4):90
4. Liu H, Zhang Y (2021) Machine learning approaches for cryptocurrency risk management a review
5. Wong S, Tan L (2019) Cryptocurrency security measures a relative study. *Int J Inf Secur Sequestration* 13(4):1–18
6. Zhang X, Wang C (2020) Cryptocurrency investment strategies a relative analysis. *J Financ Res* 43(5):521–537
7. Li K, Johnson M, Brown R (2017) Cryptocurrency risk management a relative analysis of hedging strategies. *Int J Risk Contingency Manag* 6(4):1–16
8. Wang Q, Li X (2019) Assessing cryptocurrency investment openings a relative analysis of abecedarian and specialized analysis. *J Investment Strat* 8(3):45–61
9. Zhou W, Huang L (2022) Cryptocurrency market volatility and threat operation a relative analysis. *J Financ Risk Manag* 11(1):20–34
10. Wang Y, Li J (2020) Cryptocurrency trading patterns a relative study of market efficiency. *J Financ Econ* 4(2):35–50

A Review on Quantum Key Distribution Protocols, Challenges, and Its Applications



Neha Sharma, Pardeep Singh, Abhineet Anand, Sunil Chawla, Anuj Kumar Jain, and Vinay Kukreja

Abstract Quantum key distribution (QKD) is a technology that enables secure communication by using the principles of quantum mechanics to generate and distribute cryptographic keys. QKD provides unconditionally secure communication, making it an essential technology for various industries such as military, finance, and health care. This paper provides an overview of QKD, including its principles, different protocols, experimental implementations, practical applications, ongoing research efforts, and future prospects. The different QKD protocols, such as BB84, E91, and SARG04, are described along with their experimental implementations using both discrete-variable and continuous-variable techniques. The practical applications of QKD, including secure communication, data privacy, and cryptography, are discussed, along with ongoing research efforts aimed at improving the speed, range, and scalability of QKD. The paper concludes with a summary of the key contributions of QKD to the field of quantum information science and its potential impact on society.

Keywords QKD · BB84 protocol · E91 protocol · Cryptography · Continuous variable · Discrete variable

N. Sharma (✉) · S. Chawla · A. K. Jain · V. Kukreja

Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab 140401, India

e-mail: Nehasharma0110@gmail.com; neha.1119@chitkara.edu.in

A. K. Jain

e-mail: anuj.jain@chitkara.edu.in

V. Kukreja

e-mail: vinay.kukreja@chitkara.edu.in

P. Singh

Computer Science and Engineering, Graphic Era Hill University, Dehradun, India

A. Anand

Apex Institute of Technology, Chandigarh University, Mohali, India

e-mail: abhineet.e13847@cumail.in

1 Introduction

In the digital age, secure communication is crucial for protecting sensitive information from unauthorised access. Traditional cryptographic methods rely on mathematical algorithms to encode messages, but these methods are vulnerable to attacks from increasingly sophisticated hackers. The term “quantum key distribution”, or QKD for short, refers to a technology that establishes secure communication channels that are impervious to eavesdropping through the application of the principles of quantum mechanics. In order for quantum key distribution to function, quantum particles like photons must be sent across a communication channel between two parties: the transmitter, who is referred to as Alice, and the receiver, who is referred to as Bob. Quantum properties of the emitter, such as the polarisation or phase of the photons, are used to encode information, while the receiver uses measurements to decode the information that has been encoded. The QKD is based on the uncertainty principle, which is a central tenet of quantum mechanics and serves as the framework for the theory. It is conceivable, in accordance with this rule, to determine the momentum of a quantum particle with a level of precision that is lower than that of its position, and vice versa. This principle is utilised by QKD to ensure that any attempt to intercept or measure the photons will cause a change in their state, so signalling to both the transmitter and the receiver that there is someone trying to listen in on their conversation. It is common practice to formulate a mathematical representation of the uncertainty principle [1] as follows:

$$\Delta x \Delta p \geq h/2\pi, \quad (1)$$

where Δx is the uncertainty in the particle’s position, Δp is the uncertainty in its momentum, and h is Planck’s constant. Then comes the quantum state of the particle represented by a wave function, which is a complex-valued function that describes the particle’s properties. The wave function [2] is typically denoted by the symbol Ψ and can be written mathematically as

$$\Psi(x) = A e^{(ikx)}. \quad (2)$$

One other fundamental tenet of QKD is known as the “no-cloning theorem”, which states that it is mathematically impossible to create an exact copy of a quantum state that has not yet been determined. This indicates that any attempt to deflect and copy quantum particles within a QKD system will invariably result in defects that can be detected by both the sender and the receiver. This theorem can be stated mathematically as [3],

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad (3)$$

cannot be cloned, i.e. there is no unitary transformation U that satisfies [3]

$$U(|\psi\rangle \otimes |0\rangle) = |\psi\rangle \otimes |\psi\rangle, \quad (4)$$

where α and β are complex numbers and $|0\rangle$ and $|1\rangle$ represent the two possible quantum states. Overall, QKD provides a secure means of communication by exploiting the principles of quantum mechanics to create an unbreakable encryption key. While the technology is still in its early stages, it holds great promise for protecting sensitive information in a wide range of applications.

Section 2 discusses the related literature. The chances of a photon being sent and created are discussed in Sect. 3. Various practical applications and challenges are mentioned in Sects. 4 and 5 respectively. Sect. 6 concludes the paper.

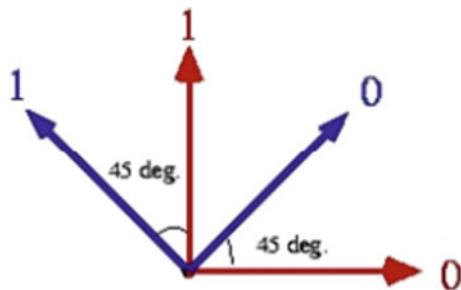
2 Related Literature

In recent years, the cryptographic method known as quantum key distribution (QKD) [4] has garnered a large amount of attention due to its potential to provide secure communication routes that are resistant to attacks by sophisticated hackers. Encoding and decoding information in QKD is accomplished by applying the concepts of quantum physics and making use of the characteristics of quantum particles like photons. One of the first and most well-known QKD protocols is called the BB84 protocol [5], which was proposed by Bennett and Brassard in the year 1984. It does this by encoding information using the polarisation of photons and then using a filtering process to get rid of any errors. Since its inception, a number of alternative QKD protocols have been developed, including the E91 protocol, the B92 protocol, and the SARG04 protocol, amongst others. The particular necessities of the communication system should direct the selection of the appropriate protocol. The implementation of QKD systems is a considerable problem because it is necessary to use a protected communication channel in order to transfer and receive quantum particles between the transmitter and the receiver. There have been several different channel types suggested, including free-space channels, fibre-optic channels, and satellite-based channels. QKD, on the other hand, has a limited range of several hundred kilometres and requires specialised hardware, which can be expensive and difficult to scale for large networks. Additionally, QKD requires a key exchange protocol. Applications in the military, the financial sector, and the government all stand to benefit greatly from the use of QKD to provide secure communication routes. Continued research and development in QKD is anticipated to result in advancements in both the technology and the ways in which it can be applied.

2.1 BB84 Protocol

Bennett and Brassard initially presented the BB84 protocol [5], which is now one of the most well-known QKD protocols since 1984. It is considered to be one of the

Fig. 1 Working of BB84 protocol



earliest and most well-known QKD methods. The information is encoded through the protocol by employing the polarisation of photons, and a filtering mechanism is utilised in order to remove any errors that may have been generated as a result of interference from the environment or eavesdropping.

The mechanism of the BB84 protocol is depicted in Fig. 1, which shows how the sender (Alice) generates a random string of bits and encodes each bit into a single photon by selecting one of the following polarisations: horizontal, vertical, diagonal, or anti-diagonal. In order to quantify the photons that have been received, the receiver (Bob) also chooses at random a polarisation basis. This basis might be either the horizontal/vertical basis or the diagonal/anti-diagonal basis. Following the completion of the transmission, Alice and Bob will now publicly announce their decision over the basis for each photon. They only store the bits that correspond to their bases, which are the bits that are used to form the secret key. These bits are the ones that are used to generate the secret key. In the case that an eavesdropper by the name of Eve tries to intercept the photons and measure them, the key will contain errors as a result of her measurements since her attempts to measure the photons will produce faults. By comparing a subset of the bits that they each have, Alice and Bob will be able to determine whether or not these vulnerabilities exist. They are allowed to make the presumption that the key can be used securely for encryption purposes provided that the rate of mistakes is lower than a threshold that has been established in advance.

2.2 E91 Protocol

The E91 protocol [6] is a quantum key distribution mechanism that was proposed by Ekert in 1991. It is distinguished by the fact that it divides the key using entangled photons. The non-locality principle in quantum mechanics, which allows for correlations between distant particles that cannot be explained by traditional physics, is the foundation of the protocol.

The fact that both Alice and Bob each get a pair of entangled photons is evident in Fig. 2. These photons are generated in such a way that ensures the polarisation states of both of the photons are correlated with one another. Alice selects a random

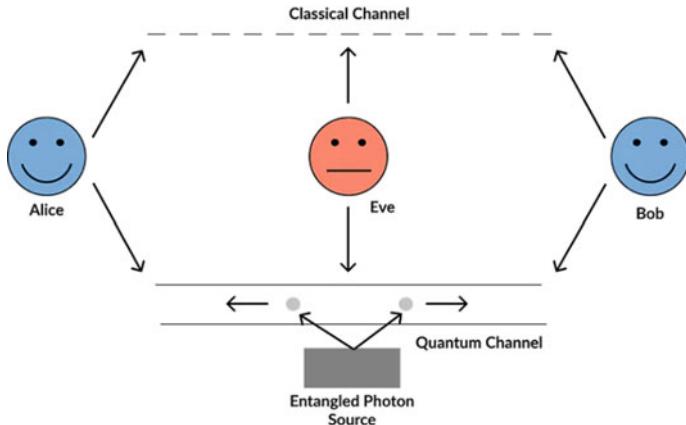


Fig. 2 Working of E91 protocol

basis, then measures the polarisation of her photons to determine their orientation. Bob also selects a basis at random and determines the polarisation of the photons he has produced. They will then make their bases public, at which point they will disregard any measurements whose bases do not agree with their own. Alice and Bob are able to identify any effort by an eavesdropper to intercept and measure the photons because they are able to use the correlations that exist between the entangled photons. Under the presumption that the laws of physics are observed, the protocol can be considered secure.

3 Experimental Implementation

The experimental validation and development of this methodology have relied heavily on the use of quantum key distribution (QKD), which stands for quantum key distribution. In this article, it is examined that most of the significant experimental implementations of QKD include both discrete-variable (DV) [7] and continuous-variable (CV) protocols [8]. Additionally, some of the challenges and restrictions that are associated with QKD are also highlighted.

3.1 Discrete Variable (DV) Protocol

The DV protocols [7], including the BB84 [5] and E91 protocols [6] protocols, are the QKD protocols that are utilised the most frequently. Bennett et al. in 1992 reported one of the earliest experimental implementations of QKD. In this experiment, they

proved the BB84 protocol by making use of photons that were created by a nitrogen-vacancy centre in diamond. This was one of the early experimental implementations of QKD. Since that time, a large number of experimental implementations of QKD have been published. These implementations include those that use photons produced by lasers, spontaneous parametric down-conversion, and quantum dots. Researchers from the University of Geneva in Switzerland reported a QKD experiment in 2007 that used a DV protocol and had a record distance of 144 km across optical fibres [8]. Researchers from the Chinese Academy of Sciences published the results of a quantum key distribution experiment in 2017 [9] that involved the use of a satellite to distribute keys over a distance of 1200 km. In 2019, researchers from the University of Geneva reported a QKD experiment that was conducted at a distance of 421 km [10]. The experiment used a DV methodology. This was an important breakthrough since it highlighted the potential of DV protocols for secure communication over great distances. DV stands for distributed validation. In 2021, researchers from the University of Science and Technology of China announced a QKD experiment that was conducted over a distance of 1203 km [11] using a DV methodology. This was a noteworthy achievement since it highlighted the potential of DV protocols for secure communication over considerably larger distances than had been documented in prior studies.

3.2 *Continuous Variable (CV) Protocol*

CV protocols are yet another grouping of quantum key distribution protocols that can be distinguished from the others. Protocols such as the Gaussian-modulated coherent-state (GMCS) protocol and the coherent-state quantum key distribution (CV-QKD) protocol are included in this category. Using CV-QKD protocols, information is encoded in the amplitude and phase of a continuous-variable quantum state, which is typically either a coherent state or a compressed state. This state can be either coherent or compressed. Ralph et al. reported one of the early experimental implementations of CV-QKD in 1999, in which they proved the transmission of a secure key over a distance of 11.1 km [8]. This was one of the earliest experimental implementations of CV-QKD. Since that time, a great number of experimental implementations of CV-QKD have been described. These implementations include those that use techniques such as photon subtraction, compressed states, and homodyne detection. A CV-QKD experiment utilising a fibre-optic line with a record distance of 200 km was reported in 2016 by researchers from the University of Geneva [9]. In 2018, researchers from the National Institute of Standards and Technology revealed the results of a CV-QKD experiment that used a free-space channel and had a record distance of 22.9 km [10]. In 2017, researchers from the Chinese Academy of Sciences published the results of a QKD experiment in which they distributed keys over a distance of 1200 km using a satellite. This was an important achievement since it highlighted the possibility for QKD to be used for secure communication over great distances [11]. In 2018, researchers from the National Institute of Standards and Technology

revealed the results of a CV-QKD experiment that used a free-space channel and had a record distance of 22.9 km [11]. This was a very important accomplishment since it proved that CV-QKD had the ability to provide secure communication over free-space channels [12]. In the year 2020, researchers from the Chinese Academy of Sciences published the results of a QKD experiment in which they distributed keys over a distance of 2600 km via a satellite. This was an important accomplishment since it proved that QKD had the potential to provide secure communication over even greater distances. Using a CV protocol and transmitting data over a distance of 525 km, researchers from the National Institute of Information and Communications Technology in Japan claimed a QKD experiment in the year 2022 [5]. This was a major success since it revealed the potential for CV protocols to provide secure communication across vast distances.

Overall, experimental implementations of QKD have played a crucial role in the development and validation of this technology. While QKD offers promising potential for secure communication, challenges and limitations, such as channel loss and noise, continue to exist, and ongoing research and development are required to overcome these challenges and improve the technology.

4 Practical Applications

Quantum key distribution (QKD) has numerous practical applications in various fields such as secure communication, data privacy, and cryptography. One of its main advantages is its ability to provide unconditionally secure communication, which means that it is difficult for an eavesdropper to impede and decode the transmitted information without being detected. In secure communication, QKD is used to establish secure channels between two parties, such as in the military, finance, and government sectors. This ensures that sensitive information is protected from unauthorised interception. Data privacy is another application of QKD [13]. It can be used to encrypt data transmissions, ensuring that the data is protected from interception and decryption by unauthorised individuals. This is especially important in industries like health care and finance where data privacy is crucial. QKD is also used in cryptography to generate random numbers and keys for encryption algorithms. The randomness of the keys generated by QKD can help to strengthen cryptographic algorithms, making them more resistant to attacks. Ongoing research is aimed at improving the speed, range, and scalability of QKD. Researchers are working on developing QKD protocols that can operate over longer distances, enabling secure communication between distant locations. Additionally, efforts are focused on improving the speed of QKD protocols, making them more practical for real-world applications. Improving the scalability of QKD protocols is also a major area of research [14]. This will allow for secure communication between multiple parties in a network, which is essential for applications such as secure cloud computing. Furthermore, research is focused on developing more efficient and cost-effective hardware for QKD protocols.

New techniques for detecting and correcting errors in quantum communication channels are also being explored. These ongoing research efforts are critical in advancing the practical applications of QKD and making it more accessible for real-world applications.

5 Challenges

The quantum technology is making rapid strides, which will have ramifications for all types of communication and information security systems as it advances. If one is to realise the goal of cost-effective global deployment of quantum technology [16], interdisciplinary research is required for a number of projects, including those at the device level, the design of new QKD protocols [15], improvements to SKR and the transmission distance of quantum signals, the exploration of new use-cases, the development of advanced network architectures, and others. In this article, we take a look at some of the networking challenges that, in order to combine QKD with next-generation optical networks, need to be resolved as soon as humanly possible. In Table 1, challenges are presented in reverse chronological order.

Table 1 Challenges in QKD

Year	Ref. No	Challenge	Interpretation
2018	[17]	RWTA	Key updation algorithm for dynamic traffic
2019	[18]	RWTA	Optimise quantum key resources with auxiliary graph-based RWTA algorithm
2019	[19]	Multi-tenant provisioning	Multi-user QaaS/SKR assignment strategy
2019	[20]	Key recycling, trustworthy repeater-node placement, and cost-cutting	Cost efficient model is proposed
2020	[21]	RTWA	2 MQON auxiliary graph-based RTWA techniques with revised node structure are proposed
2020	[22]	Resiliency	Proposed algorithm solves the hybrid resource allocation problem
2020	[23]	Key recycling, trustworthy repeater-node placement, and cost-cutting	Partial and mixed recycling can boost QKD keys

6 Conclusion

This paper has provided an overview of quantum key distribution (QKD), its principles, different protocols, experimental implementations, practical applications, ongoing research efforts, and future prospects. QKD provides unconditionally secure communication, data privacy, and cryptography, making it an essential technology for various industries such as military, finance, and health care. Researchers have performed experiments on QKD protocols using both discrete-variable and continuous-variable techniques, with ongoing research efforts focused on improving their speed, range, and scalability. The key contributions of QKD to the field of quantum information science include its ability to provide unconditionally secure communication and generate random numbers and keys for encryption algorithms. QKD is also driving the development of new quantum technologies and cryptographic algorithms that can improve data privacy and security. Looking to the future, QKD has the potential to revolutionise the way we communicate and process data, leading to more secure and efficient systems. QKD networks could be used for secure cloud computing, enhancing the security of the internet, and improving the security of critical infrastructure. As QKD technologies continue to advance, they have the potential to become more accessible and practical for real-world applications, making a significant impact on society.

References

1. Xu F, Ma X, Zhang Q, Lo HK, Pan JW (2020) Secure quantum key distribution with realistic devices. *Rev Mod Phys* 92(2):025002
2. Mehic M, Niemiec M, Rass S, Ma J, Peev M, Aguado A, Martin V, Schauer S, Poppe A, Pacher C, Voznak M (2020) Quantum key distribution: a networking perspective. *ACM Comput Surv (CSUR)* 53(5):1–41
3. Zapatero V, van Leent T, Arnon-Friedman R, Liu WZ, Zhang Q, Weinfurter H, Curty M (2023) Advances in device-independent quantum key distribution. *NPJ Quantum Inform* 9(1):10
4. Nadlinger DP, Drmota P, Nichol BC, Araneda G, Main D, Srinivas R, Lucas DM, Ballance CJ, Ivanov EY-Z, Sekatski P, Urbanke RL, Renner R, Sangouard N, Bancal JD (2022) Experimental quantum key distribution certified by Bell's theorem. *Nature* 607(7920):682–686
5. Xie YM, Lu YS, Weng CX, Cao XY, Jia ZY, Bao Y, Wang Y, Fu Y, Yin H-L, Chen ZB (2022) Breaking the rate-loss bound of quantum key distribution with asynchronous two-photon interference. *PRX Quantum* 3(2):020315
6. Saluja K, Bansal A, Vajpaye A, Gupta S, Anand A (2022, April) Efficient bag of deep visual words based features to classify CRC images for colorectal tumor diagnosis. In: 2022 2nd International conference on advance computing and innovative technologies in engineering (ICACITE). IEEE, pp 1814–1818
7. Wang S, Yin ZQ, He DY, Chen W, Wang RQ, Ye P, Zhou Y, Fan-Yuan G-J, Wang F-X, Chen W, Zhu Y-G, Morozov PV, Divochiy AV, Zhou Z, Guo G-C, Han ZF (2022) Twin-field quantum key distribution over 830-km fibre. *Nat Photonics* 16(2):154–161
8. Wang S, He DY, Yin ZQ, Lu FY, Cui CH, Chen W, Zhou Z, Guo G-C, Han ZF (2019) Beating the fundamental rate-distance limit in a proof-of-principle quantum key distribution system. *Phys Rev X* 9(2):021046

9. Bassett FB, Valeri M, Roccia E, Muredda V, Poderini D, Neuwirth J, Spagnolo N, Rota MB, Carvacho G, Sciarrino F, Trotta R (2021) Quantum key distribution with entangled photons generated on demand by a quantum dot. *Sci Adv* 7(12):eabe6379
10. Saluja K, Gupta S, Vajpayee A, Debnath SK, Bansal A, Sharma N (2022) Blockchain technology: applied to big data in collaborative edges. *Meas Sens* 24:100521
11. Minder M, Pittaluga M, Roberts GL, Lucamarini M, Dynes JF, Yuan ZL, Shields AJ (2019) Experimental quantum key distribution beyond the repeaterless secret key capacity. *Nat Photonics* 13(5):334–338
12. Zhang G, Haw JY, Cai H, Xu F, Assad SM, Fitzsimons JF, Zhou X, Zhang Y, Yu S, Wu J, Ser W, Kwek LC, Liu AQ (2019) An integrated silicon photonic chip platform for continuous-variable quantum key distribution. *Nat Photonics* 13(12):839–842
13. Singh P, Prakash V, Bathla G, Singh RK (2022) QoS aware task consolidation approach for maintaining SLA violations in cloud computing. *Comput Electr Eng* 99(107789)
14. Bhatt C, Kumar I, Vijayakumar V, Singh KU, Kumar A (2021) The state of the art of deep learning models in medical science and their challenges. *Multimed Syst* 27(4):599–613. <https://doi.org/10.1007/s00530-020-00694-1>
15. Sharma N, Chakraborty C (2022) Evaluation of bioinspired algorithms for image optimization. *J Electron Imaging* 31(4):041206
16. Sharma N, Chakraborty C, Kumar R (2022) Optimized multimedia data through computationally intelligent algorithms. *Multimedia Syst* 1–17
17. Wang H et al (2018) A flexible key-updating method for software-defined optical networks secured by quantum key distribution. *Opt Fiber Technol* 45:195–200
18. Dong K, Zhao Y, Yu X, Zhang J, Yu H, Li Z (2019) Auxiliary graph based routing, wavelength and time-slot assignment in metro quantum optical networks. In: Proceedings of IEEE OECC/PSC, Fukuoka, Japan, pp 1–3
19. Cao Y, Zhao Y, Wang J, Yu X, Ma Z, Zhang J (2019) SDQaaS: software defined networking for quantum key distribution as a service. *Opt Exp* 27(5):6892–6909
20. Cao Y, Zhao Y, Wang J, Yu X, Ma Z, Zhang J (2019) Costefficient quantum key distribution (QKD) over WDM networks. *IEEE/OSA J Opt Commun Netw* 11(6):285–298
21. Dong K, Zhao Y, Yu X, Nag A, Zhang J (2020) Auxiliary graph based routing, wavelength, and time-slot assignment in metro quantum optical networks with a novel node structure. *Opt Exp* 28(5):5936–5952
22. Lu L, Yu X, Zhao Y, Zhang J (2020) Dynamic wavelength and key resource adjustment in WDM based QKD optical networks. In: Proceedings of OSA ACP, Beijing, China, p 184
23. Li X, Zhao Y, Nag A, Yu X, Zhang J (2020) Key-recycling strategies in quantum-key-distribution networks. *Appl Sci* 10(11):1–19

Vulnerabilities in Smart Contracts of Decentralized Blockchain



Anurag Singh, Kapil Sharma, and Pradeepa Kumar Sarangi

Abstract In the recent years, there has been a tremendous surge in the popularity of blockchain, even more so since smart contracts started getting used on top of the blockchain protocols. Smart contract are low-level code scripts running on the blockchain which are also automated. Because of being automated, smart contracts can take place of a trusted 3rd party with Peer-to-Peer transaction taking place using this technology. Because of their productivity alongside blockchain the popularity of smart contracts has soared in recent times, to the extent that a huge amount of money is being exchanged every day in the finance world using this blockchain-based technology. Nevertheless, no system can be completely foolproof and safe, which is also true for smart contracts as they can be vulnerable. There have been several instances where the management of smart contracts has resulted in conflicts or issues, which have become the biggest concern of the smart contract developers and the users. Since the use of smart contracts started with Ethereum blockchain, it has advanced in many aspects but also have gotten exploited again and again because of the vulnerabilities. This paper presents a survey of research work done on the vulnerabilities of smart contracts, highlighting the common vulnerabilities mentioned in these papers alongside their corresponding prevention methods. Additionally, we examine the common shortcomings found in these papers.

Keywords Blockchain · Smart contracts · Vulnerabilities · Cybersecurity

A. Singh · K. Sharma (✉) · P. K. Sarangi

Chitkara University Institute of Engineering and Technology, Rajpura, Punjab, India

e-mail: kapil.sharma@chitkara.edu.in

A. Singh

e-mail: anurag0713.cse19@chitkara.edu.in

P. K. Sarangi

e-mail: pradeepa.sarangi@chitkara.edu.in

1 Introduction

The current system typically involves centralized transactions between parties that necessitate the presence of a reliable third party, such as a bank. However, this approach raises security issues, including a single point of failure and large transaction costs. On the other hand blockchain presents a solution to these concerns by enabling anyone to interact and trade on a decentralized network without needing any sort of a centralized third party [1]. First revealed by Satoshi Nakamoto in his paper “Bitcoin: A Peer-to-Peer Electronic Cash System” [2], blockchain is a technology that uses distributed ledgers to let users record and access a synchronized and decentralized view of a system’s status across the network. This technology is not limited to cryptocurrencies, but has also found widespread use in traditional finance. Additionally, it has paved the way for innovative applications such as smart contracts [3].

The idea of smart contracts predates blockchain and refers to a kind of contract which automatically fulfills previously assigned requests corresponding to when a set of conditions are fulfilled. Nick Szabo, who coined the term "smart contract," used the example of a vending machine as the fundamental model of a smart contract. In this example, a customer initiates a transaction by inserting money into the machine and selecting the desired item. The vending machine, which operates on a pre-programmed mechanism, then automatically fulfills the request by dispensing the correct item to the customer [4]. Smart contracts work the same but in more secure way as smart contracts work on top of the blockchain, this makes them immutable, undestroyable, and will not malfunction contrary to a vending machine. This makes the dependence on a trusted and reliable 3rd party (authority, entity, or organization) for a transaction unnecessary and completely replaces them programs running on an immutable and decentralized blockchain system. However, smart contracts are not always perfect. In the end, the smart contract programs are written by humans and there can always be possibilities of human error or code logic which may lead to possible vulnerabilities inside the smart contract’s programs, which can later on get exploited by malicious personals.

Although smart contracts offer numerous benefits, there are several challenges that impede their widespread acceptance, which are vulnerabilities, exploits, and legal issues [5]. In this work we survey some papers on the topic of “vulnerabilities of smart contracts” and draw a conclusion accordingly, it also discusses about the common smart contract vulnerabilities alongside their prevention methods.

2 Related Work

Recently, blockchain and the use of smart contracts have increased in popularity so much that a lot of researchers have published papers on this topic. We focus on the papers related to smart contract vulnerabilities. Samreen and Alalfi [6] used

NIST bugs framework to explain about 8 different vulnerabilities on Ethereum smart contracts with case scenarios and showed how they could be prevented if given correct attention while programming the smart contracts. Garfatta et al. [7] put together a list of approaches for verification of smart contracts coded in solidity by analyzing a number of smart contract vulnerabilities. Liu and Liu [8] did an in-depth survey with 53 papers related to smart contract verification, analyzed them from two aspects, the security assurance and correctness verification and lastly proposed a detailed taxonomy. Khan and Namin [9] discussed about 20 different vulnerabilities patterns and concluded that smart contracts are prone to vulnerabilities if not written properly. However they did not discuss any mitigation techniques for these vulnerabilities. Kim and Ryu [10] presented a comprehensive survey of smart contracts by analyzing 67 papers. Their primary challenge was the uncertainty in program behavior surfacing from transaction-ordering dependencies, dependence on the off-chain sources, and code opacity. In addition to these, the inadequate definitions of vulnerability properties and unstable semantics of programming language were two other unsolved challenges. Zhou et al. [11] surveyed 50 papers and probed the present state of security in the smart contracts, common vulnerabilities, and the various security analysis tools. They found that a unchanging smart contracts vulnerability definition does not exist in research work and same vulnerabilities can appear with various names. Difficulties arise in identifying, categorizing, and analyzing vulnerabilities due to this inconsistency. They also analyzed some tools and concluded there is no single tool which can be used to 100% accurately check smart contracts. Destefanis et al. [12] surveyed vulnerable smart contracts and presented a comprehensive analysis of the real-world prevalence of security exploits against smart contracts. After surveying all the reports, they found out that only 504 out of the surveyed 21,270 contracts were subject to exploits which suggests that the impact of vulnerable smart contracts had been exaggerated. Fauziah et al., Gulati et al. [13, 14] discussed about the smart contract vulnerabilities, security tools used for vulnerability detection and preventive measures. They concluded that there are lots of security issues in the smart contracts, these smart contracts deal in lots of digitally owned assets and a single mistake or security exploit/flaw can produce some heavy losses. Leka et al. [15] classified exploitation methods into four different parts and analyzed smart contract vulnerabilities to determine its implications on smart contracts. They mainly focused on examining attacks based on smart contracts and the consequences of the exploitations and like other papers they also concluded that as the blockchain technology keeps evolving, it is not free from vulnerabilities and attacks.

Most of these papers were focused on Ethereum smart contracts or were limited to very less smart contract exploitation case studies which leaves the gap that other blockchains may have different type of vulnerabilities or they may act different under certain circumstances. But that does not mean there are no papers related to smart contracts of other blockchains. He et al. [16] analyzed EOSIO blockchain smart contracts and proposed EOSAFE, a static analysis framework to automatically detect vulnerabilities in EOSIO smart contracts. Other than the traditional methods some papers also utilize deep learning techniques such as [17, 18] presented different deep-learning techniques which helps in identifying smart contract vulnerabilities.

They used Keras python library for experimenting with deep-learning models. It is an open-source, free, user-friendly, and a powerful library for developing and evaluating deep-learning models with minimal lines of code.

Many papers present tools and frameworks for mitigating, verification, and preventing vulnerabilities in the smart contracts. Wang et al. [19] presents Vultron, it can detect irregular transaction vulnerabilities and can improve traditional security analysis techniques when applied to smart contracts. The authors also mention that in future Vultron may be capable of detecting malicious contracts too. Jiang et al. [20] introduces us to Contract Fuzzer, a smart contract testing for Ethereum smart contracts. The paper used Contract Fuzzer to test 6991 smart contracts and flagged 459 vulnerabilities with high precision. Tsankov et al. [21] presented Security, a Ethereum smart contracts security analyzer. It is a scalable, completely automated tool that can be used to verify whether the behavior of a contract is safe or unsafe relative to the specified property [22] proposes Contract Ward an automated vulnerability detection model to detect smart contract vulnerabilities with machine learning techniques. It was noted that the detection speed of Contract Ward was about 4 s/ smart contract on average, which is faster compared to most of the tools. There are frameworks too, [23] presents Slither, a static analysis framework is intended to provide rich data about Ethereum smart contracts. Code comprehension, automated optimization identification, automated vulnerability detection, and assisted code review are its four primary use cases. Upon assessing its ability to discover bugs by comparing it to other state-of-the-art vulnerability detection tools by measuring how well it performed against reentrancy bugs, it was discovered that Slither performs more efficiently compared to the rest of the tools in terms of adaptability, precision, and performance. Vivar et al. [24] presents Ethereum security analysis framework (ESAF), this framework combines all the existing tools and provides combined information. The authors found out that no single tool gives 100% accurate results and then presented this all-in-one framework which contains all the tools which makes it really easy for the smart contract developers to analyze the vulnerabilities by combining many static/dynamic analysis tools. Lastly [25] conducted a thorough experimental evaluation of existing static testing tools for smart contract security. They experimented on 4 tools with 10 vulnerabilities and found out that SmartCheck performed statistically better compare to other tools and Mythril was proven to be highly precise with fewest false alarms.

3 Overview of Blockchain and Smart Contracts

3.1 *Blockchain*

The underlying design that enables the operation of the digital currency Bitcoin is what we refer to as blockchain technology. Satoshi never used the word “blockchain” in his whitepaper, and after reading it, it is clear that he was not presenting a new

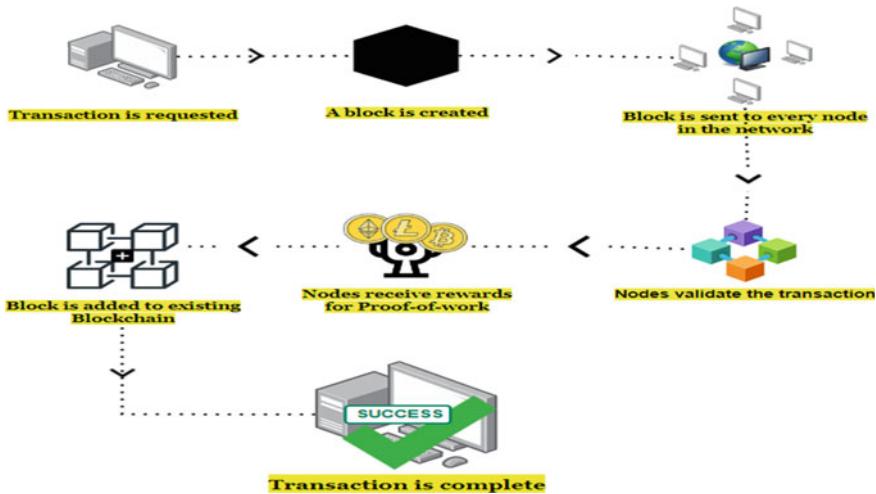


Fig. 1 Working of blockchain

technology, but rather a software-design that drew on a number of already-existing technologies to enable him to create a “purely peer-to-peer version of electronic cash” [26]. This effectively addresses the issues associated with relying on a centralized party for trust. Simply put, a blockchain is a decentralized network that can serve various functions. It is basically an immutable or unalterable ledger where the information recorded on it is visible to all users. One of the defining features of blockchain technology is that it is not controlled by a central authority, which makes it less susceptible to attacks and failures [15] (Fig. 1).

3.2 Smart Contracts

Long before the creation of Bitcoin and other blockchain-based technologies, Nick Szabo, in 1994 first introduced this very concept of smart contracts. However, it wasn’t until the development of Ethereum in 2015 that the first functional implementation of smart contracts on a blockchain was achieved. Ethereum’s smart contract system allowed developers to create decentralized applications (dapps) that execute automatically, without the need for intermediaries, providing a new way to build and interact with decentralized systems.

In a broader sense, smart contracts are automated pacts, or the legal contracts, but written as small or big computer programs, the execution of which upholds the conditions mentioned or embedded in the contract [27]. The contract is executed to carry out the digital-transaction when certain requirement or condition is satisfied. No

party can change the terms of the contract since they are cryptographically encrypted [15]. As blockchains are immutable, it is more or less not possible to remove or delete the contract once it is deployed onto a blockchain protocol unless a substantial attack technique is used to compromise the entire blockchain network [15].

4 Smart Contract's Platforms

Creation and implementation of smart contracts can be done on various blockchains which support the use of them. Every blockchain platform have their own different way to support the smart contracts. For example, some may only allow the developers to use basic scripting language which makes a contract with simple and straight logic or others like Ethereum, lets you use programming languages which are very advanced created for the smart contracts [28]. We will talk about three of these platforms (Fig. 2).

Bitcoin [2] is a publicly accessible blockchain platform, it has a very restricted computing capability. A stack-based bytecode scripting language is used in its smart contracts. It can support simple code logics. However, tricky logics don't work because of the limitations of the scripting language used on bitcoin. For example, bitcoin scripting language, does not support loops and withdrawal limits [2]. The only way possible to implement a loop in this case is by repeating the code many times, which is highly inefficient [1].

NXT [29] is an open-source/open for all blockchain platform. It operates using the proof-of-stake protocol exclusively. It gives you a premade smart contract templates which can be used on top of it. However, NXT is not Turing complete, which means the developers can only use existing templates for smart contracts and can't make personalized smart contracts. This is a major drawback of NXT [5].

Ethereum [30] has emerged as the most widely used blockchain platform for creating and deploying smart contracts since its release in July, 2015. It contains a virtual



Fig. 2 Major smart contracts platforms

machine known as Ethereum virtual machine (EVM), EVM is a Turing complete VM which helps in supporting advanced and fully customized/personalized smart contracts. The EVM is responsible for executing smart contracts on the Ethereum network. It is a virtual machine that runs on each node in the network and ensures that all nodes have the same execution environment for smart contracts [5]. Ethereum smart contracts are written in various high-level programming languages, solidity is one of those. Currently, Ethereum is the leading platform for developing/deploying smart contracts [1].

5 Features of Smart Contracts

Self-executing: Smart contracts are self-executing, which means they carry out automatically once the requirements are satisfied. There is no need for a third party or middleman to oversee the execution of the contract, which reduces costs and eliminates the risk of human error or bias.

Immutable: Since smart contracts are deployed on a blockchain, they inherit the immutability of the blockchain. That means that once they are deployed, they become unalterable and no one can change its content. Which makes sure that the terms and conditions of the contract remain transparent and unalterable, providing a high level of trust and security.

Decentralized: Smart contracts work on a decentralized network, which means that they cannot be controlled by any single entity or party. Which makes them resistant to censorship and provides a high level of security and trust.

Transparent: Smart contracts are transparent, meaning that the terms of the contract are visible to all parties involved. This transparency ensures that there is no ambiguity or confusion about the terms of the contract, and it also helps to prevent fraud or misinterpretation.

Programmable: Smart contracts are programmable, meaning that they can be designed to execute complex, multi-step processes automatically. This allows for more sophisticated and efficient contract execution, as well as the ability to integrate with other systems and applications (Fig. 3).

6 Vulnerabilities and Preventive Measures

Reentrancy: Reentrancy AKA recursive call attack is a notorious exploit. By using this attack, the attacker can exhaust all the Ether present in the smart contract and gain unauthorized access to the smart contract code. It occurs when an external call is made to an untrustworthy smart contract and someone takes over that smart contract,

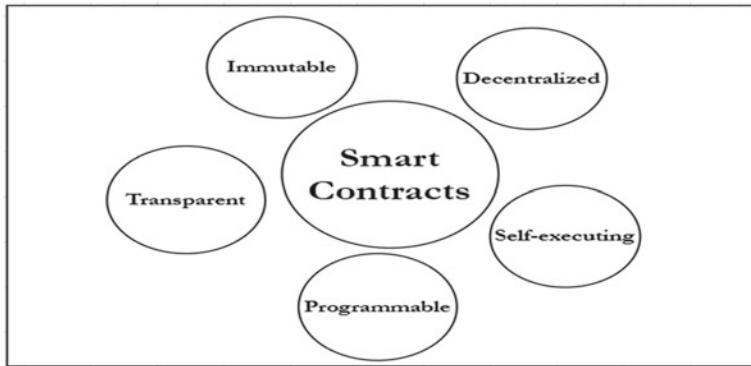


Fig. 3 Features of smart contracts

the attacker is able to then send recursive call back to the original function and repeat transactions that otherwise wouldn't run which in turn concludes in consuming all the gas [7].

Preventive Measure: For preventing against this attack “Check-Effect-Interaction pattern” can be used. The use of this method is a dependable approach that establishes a sequence in which functions should be structured to ensure that all internal state changes are completed before the next call. Once all state changes have been resolved, the function can then proceed to interact with other contracts [17].

Smart Contract Overflow and Underflow: In solidity language, the integer datatype follows a range limit. If the values of the variables get out of the upper or lower limit during arithmetic operations, the value will wrap to the other side of the bound [22]. The Integer overflow or underflow takes place when a fixed size variable is required which just happens to not be in the default range [7]. If that value is more than the maximum, overflow occurs and when the value is less than the minimum, underflow occurs. However, underflow exploit is easier to do or get triggered because finding the required value to initiate overflow attacks is usually very difficult [10].

Preventive Measure: We can prevent this by using mathematical-libraries and not the usual math operations while writing the smart contract code [7].

Delegate Call: Delegate call is a special function in solidity that is similar to the normal call-function, with one significant difference. With delegate call, a contract can call a function from a different contract and execute it within its own context, using its own storage and state allowing it to reuse the code of the called contract [22]. This means that a contract is able to load code from a different address dynamically at runtime, while the storage still points to the calling contract. This is how the library feature is implemented in solidity. However, it also creates some security risks, such as an attacker could potentially get the ability to manipulate the storage of the calling contract by exploiting vulnerabilities in the called contract [21].

Fig. 4 Smart contract vulnerabilities

- ## Smart Contract Vulnerabilities
- Reentrancy
 - Smart Contract Underflow/overflow
 - Delegate call
 - Dos by external call
 - Fake Eos
 - Mishandled exception
 - Dos with block gas limit
 - Transaction ordering dependency

Preventive Measure: Risk of delegate call exploitation can be reduced by carefully monitoring both the library contract and the calling contract. Additionally, it is recommended to develop stateless libraries whenever possible, as this eliminates the need for the library contract to maintain state and makes it less susceptible to attacks [10] (Fig. 4).

Delegate Call: With delegate call, a contract can call a function from a different contract and execute it within its own context, using its own storage and state allowing it to reuse the code of the called contract [22]. This means that a contract is able to load code from a different address dynamically at runtime, while the storage still points to the calling contract. This is how the library feature is implemented in Solidity. However, it also creates some security risks, such as an attacker could potentially get the ability to manipulate the storage of the calling contract by exploiting vulnerabilities in the called contract [21].

Preventive Measure: Risk of delegate call exploitation can be reduced by carefully monitoring both the library contract and the calling contract. Additionally, it is recommended to develop stateless libraries whenever possible, as this eliminates the need for the library contract to maintain state and makes it less susceptible to attacks [10].

DoS–External Call: If the control flow reaches an external contract, it may result in the failure of the execution of the calling contract, either intentionally or unintentionally, leading to a Denial of Service (DoS) state. In addition, if an external call fails and the transaction is reverted, or if the execution is disrupted by an attacker in this case also then also the caller contract may enter a DoS [7].

Preventive Measure: To prevent this, it is recommended to execute the contract logic that handles failed calls in a way that multiple Ether transfer calls are not put together in a single transaction with the assumption that the external calls will always fail. It is also advisable to choose “pull” and not “push” for the external calls, and to implement contract logic to handle failed calls [17].

Fake EOS: In EOSIO, it is possible for anyone to create and release a token with the same name and symbol that of the official EOS token since the token’s names and symbols are not required to be unique. This presents a security risk, as an attacker could copy the source code of the official eosio.token contract and issue a fake token with the same name, symbol, and code as the official token, but with a different issuer [16].

Preventive Measure: A group of developers narrowed down the scope of accepted code to mitigate this issue. They found that to handle direct calls from other accounts “code == self” is used, whereas “code == N(eosio.token)” only accept notifications from the official account. However, this approach can be circumvented by an attacker who directly calls transfer in the DApp, which invalidates the verification due to short-circuit evaluation [16].

Mishandled Exception: Mishandled exception also known as Unchecked send, is a frequently occurring vulnerability. In the Ethereum network, one of the well-known exceptions is the out-of-gas exception [8]. Typically, when an exception occurs, transactions get reverted with some gas consumption. However, if an error occurs in a low-level function, it will return a false value and not exceptions. As for the txns which were already being processed will not get reverted and gas for the same will be spent. If the caller contract does not report this to the called contract, it may lead to vulnerabilities and attract malicious actors to exploit the contract [22].

Preventive Measure: One way to prevent this would be to use the external call method only. But this is not a complete solution as different types of external calls can be needed [7]. Another way is to completely ignore low-level functions and not use them if possible. If it is inevitable to use low-level functions, checking each return value of these functions and terminating the false values is recommended [22].

DoS–Block Gas Limit: All the transactions or functions in a smart contract need some certain amount of gas, which depends on the amount of computing power included. The gas limit is fixed for each block on Ethereum network and the total sum of the gas used by all the transactions in a block should not surpass the limit. If that limit is exceeded it can cause DoS vulnerability.

Preventive Measure: To prevent this, it is recommended to include multiple transactions in a single block instead of submitting them one by one. This in return will lower the gas needed for each single transaction. Which ultimately reduces the chances of going over gas limit and hence preventing a DoS [17].

Transaction Ordering Dependency: The performance of a smart contract differs according to the transaction sequences. These sequences can be manipulated or exploited by miners. Let's assume a scenario where the transaction-pool has two new pending transactions (T and $T1$), and the blockchain is in the state A. T needs to be processed first to transform the blockchain state from A to A1. However, miners can process $T1$ before T , and if that happens then the state will be transformed to A2 instead of A1. Then, when T is processed, it will transform the state to a new state A3. This can result in vulnerabilities as T is being processed in different block states, and the expected transaction sequence is disrupted [20].

Preventive Measure: To prevent this, a maximum limit can be set on the gas price, which can reduce the likelihood of malicious actors taking advantage of the situation by paying higher gas prices [22].

7 Shortcomings

While there are various papers on the vulnerabilities of smart contracts, none are perfect. Our paper also has its own shortcomings. We have discussed such shortcomings in this section.

Firstly, the most common problem we encountered was that all the papers related to vulnerabilities of smart contracts were based off of Ethereum blockchain. Ethereum is the most heavily used blockchain for smart contracts and was the first to implement smart contracts but that doesn't mean that other blockchains don't use them hence, can also get exploited. So, we hope in future there will be more studies on other blockchain smart contracts too.

Similar to the above point, most of the papers we have surveyed are focused on the solidity language, the most popular language for making smart contracts on Ethereum. This can lead to vulnerabilities of other languages go unnoticed. Lastly, we also noticed that some of the papers only discuss about the vulnerabilities and don't include the mitigation/prevention techniques.

8 Future Work

For the future work we would like to expand the scale and survey a big number of papers and also survey papers and tools which explore into smart contracts of blockchains other than Ethereum. Currently, most of the studies done on vulnerabilities in smart contracts topic are focused on either Ethereum smart contracts or its domain specific programming language solidity.

Currently, the most favored blockchain is Ethereum blockchain in the context of smart contracts, as shown in Fig. 5, a chart showing the difference in number of papers written on different blockchains [31]. It is deemed appropriate for Ethereum smart contracts to receive all the attention. However, with the advancements in blockchain technology and smart contracts, new blockchains with better smart contracts may appear. Which should not be neglected and should be studied, researched on too. Increasing the scope of research work and exploring blockchain smart contracts other than that of Ethereum blockchain, can also help advance the technology faster. Even right now, a great number of papers on this topic maybe getting published which helps us in securing the smart contracts better. Keeping the smart contracts safe from vulnerabilities and exploits is very important for strengthening the concept of blockchain and great solutions have been produced by persistent academic efforts, and this should continue.

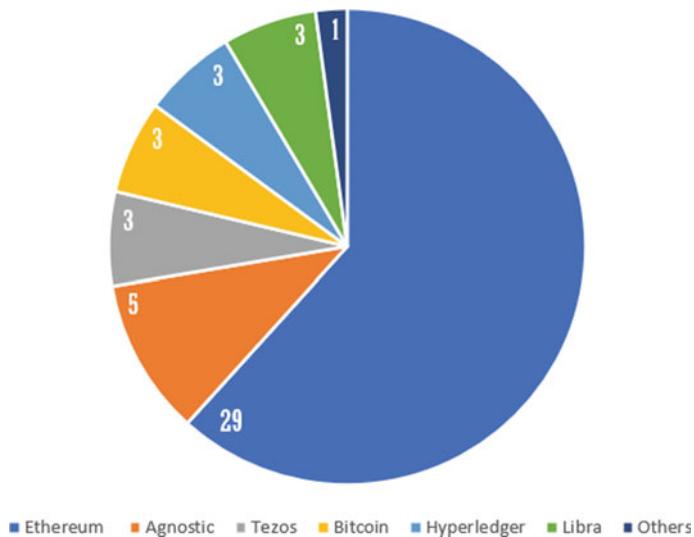


Fig. 5 Blockchain platforms focused on in studies

9 Conclusion and Results

In this paper we first discussed about what is blockchain technology and smart contracts. We talked about the platforms which helps in incorporating the blockchain technology with smart contracts, we also talked the main features of smart contracts. Most of the features of smart contracts are same as blockchain technology as it runs on it.

Smart contracts being the most popular and important feature of blockchain technology have attracted the attention of many which also exposed many of its problems in form of vulnerabilities. To counter and discuss about these vulnerabilities, many studies were done and are still being done. We took some of those papers and studied them. **After reviewing the papers we listed 8 most common smart contract vulnerabilities alongside their mitigation/prevention techniques. Lastly, we wrote about the shortcomings of the reviewed papers where we found out that most of the papers are focused on the Ethereum blockchain and solidity language which can lead to negligence of vulnerabilities of other blockchains.** As time goes by, the blockchain technology is evolving alongside smart contracts. New features and vulnerabilities are being introduced or discovered with time and hence, we need to be ready for it.

The final findings (8 different smart contract vulnerabilities) can be found in the Table 1.

For future work, we would like to work on a bigger scale than this. Surveying a large number of papers related to “vulnerabilities in smart contracts” topic and increasing the scope by including papers about blockchain smart contracts other than that of Ethereum blockchain.

Table 1 Vulnerabilities of smart contracts with their preventive measures and blockchain

Vulnerability	Description	Preventive measure
Reentrancy	Occurs when a external call is made to a compromised smart contract, from where the attacker can send recursive call back to repeat transactions without anyone knowing	“Check-effect-interaction patter” can be used to prevent this
Overflow and underflow	The Integer overflow or underflow takes place when a fixed size variable is required which just happens to not be in the default range	We can prevent this by using mathematical-libraries

(continued)

Table 1 (continued)

Vulnerability	Description	Preventive measure
Delegate call	Using delegate call a contract can call a function from a different contract and execute it within its own context which can create some security risks, such as an attacker could potentially get the ability to manipulate the storage of the calling contract by exploiting vulnerabilities in the called contract	Risk of delegate call exploitation can be reduced by carefully monitoring both the library contract and the calling contract
DoS–External call	Happens when an external call is made to a malicious contract which intentionally consumes more than needed resources or enters a infinite loop causing DoS	This can be prevented by limiting the gas and execution time of external calls
Mishandled exceptions	When an error occurs in a low-level function, it returns a false value and no exceptions and the already being processed transaction does not get reverted but gas for the same is spent. If the caller contract does not report this to the called contract, it may lead to vulnerabilities and attract malicious actors to exploit the contract	This can be prevented by not using low-level functions and completely ignoring them
Fake EOS	EOSIO blockchain lets users to release tokens with same name and symbol which can lead to malicious actors making fake tokens and posing as real tokens to scam others	This can be mitigated by giving the tokens something unique and making other users aware about the issue
DoS–Block gas limit	The gas limit is fixed for each block on Ethereum network and the total sum of the gas used by all the transactions in a block should not surpass the limit. If that limit is exceeded it can cause DoS vulnerability	This can be prevented by doing multiple transactions in a single block instead of one by one which lowers the gas needed and can avoid going over gas limit
Transaction order dependency	The performance of a smart contract differs according to the transaction sequences. These sequences can be manipulated or exploited by miners or a malicious actor by using high gas fees to front-running the transactions	This can be prevented setting a gas limit which can demotivate the malicious actors from taking advantage

References

1. Alharby M, van Moorsel A (2017) Blockchain-based smart contracts: a systematic mapping study. arXiv [cs.CR]
2. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system
3. Di Pierro M (2017) What is the blockchain?. Comput Sci Eng 19(5):92–95
4. Szabo N (1997) Formalizing and securing relationships on public networks. First Monday
5. Khan SN, Loukil F, Ghedira-Guegan C, Benkhelifa E, Bani-Hani A (2021) Blockchain smart contracts: applications, challenges, and future trends. Peer-to-peer Netw Appl 14:2901–2925

6. Samreen NF, Alalfi MH (2021) A survey of security vulnerabilities in ethereum smart contracts. arXiv [cs.CR]
7. Garfatta I, Klai K, Gaaloul W, Graiet M (2021) A survey on formal verification for solidity smart contracts. In: 2021 Australasian computer science week multiconference, pp 1–10
8. Liu J, Liu Z (2019) A survey on security verification of blockchain smart contracts. IEEE Access 7:77894–77904
9. Khan ZA, Namin AS (2020) A survey on vulnerabilities of ethereum smart contracts. arXiv [cs.CR]
10. Kim S, Ryu S (2020) Analysis of blockchain smart contracts: techniques and insights. In: 2020 IEEE secure development (SecDev). IEEE, pp 65–73
11. Zhou H, Milani Fard A, Makanju A (2022) The state of ethereum smart contracts security: vulnerabilities, countermeasures, and tool support. J Cybersecur Privacy 2(2):358–378
12. Destefanis G, Marchesi M, Ortù M, Tonelli R, Bracciali A, Hierons R (2018) Smart contracts vulnerabilities: a call for blockchain software engineering? In: 2018 International workshop on blockchain oriented software engineering (IWBOSE). IEEE, pp 19–25
13. Fauziah Z, Latifah H, Omar X, Khoirunisa A, Millah S (2020) Application of blockchain technology in smart contracts: a systematic literature review. Aptisi Trans Technopreneurship (ATT) 2(2):160–166
14. Gulati K, Gupta J, Rani L, Sarangi PK (2022) Crude oil prices predictions in India using machine learning based hybrid model. In: 2022 10th International conference on reliability, infocom technologies and optimization (trends and future directions) (ICRITO). IEEE
15. Leka E, Selimi B, Lamani L (2019) Systematic literature review of blockchain applications: smart contracts. In: 2019 International conference on information technologies (InfoTech). IEEE, pp 1–3
16. He N et al (2020) Security analysis of EOSIO smart contracts. arXiv [cs.CR]
17. Narayana KL, Sathiyamurthy K (2021) Automation and smart materials in detecting smart contracts vulnerabilities in blockchain using deep learning. Mater Today
18. Goel S, Saxena M, Kumar Sarangi P, Rani L (2022) Gold and silver price prediction using hybrid machine learning models. In: 2022 Seventh international conference on parallel, distributed and grid computing (PDGC). IEEE
19. Wang H, Li Y, Lin SW, Ma L, Liu Y (2019) Vultron: catching vulnerable smart contracts once and for all. In: 2019 IEEE/ACM 41st international conference on Software engineering: new ideas and emerging results (ICSE-NIER). IEEE, pp 1–4
20. Jiang B, Liu Y, Chan WK (2018) Contractfuzzer: fuzzing smart contracts for vulnerability detection. In: Proceedings of the 33rd ACM/IEEE international conference on automated software engineering, pp 259–269
21. Tsankov P, Dan A, Drachsler-Cohen D, Gervais A, Bünzli F, Vechev M (2018) Securify: practical security analysis of smart contracts. In: Proceedings of the 2018 ACM SIGSAC conference on computer and communications security, pp 67–82
22. Wang W, Song J, Guangquan X, Li Y, Wang H, Chunhua S (2020) Contractward: automated vulnerability detection models for ethereum smart contracts. IEEE Trans Netw Sci Eng 8(2):1133–1144
23. Feist J, Grieco G, Groce A (2019) Slither: a static analysis framework for smart contracts. In: 2019 IEEE/ACM 2nd international workshop on emerging trends in software engineering for blockchain (WETSEB). IEEE, pp 8–15
24. Vivar AL, Orozco ALS, Villalba LJG (2021) A security framework for Ethereum smart contracts. Comput Commun 172:119–129
25. Parizi RM, Dehghanianha A, Choo K-KR, Singh A (2018) Empirical vulnerability analysis of automated smart contracts security testing on blockchains. arXiv [cs.CR]
26. Amrous SH (2016) Blockchain technology: what is it good for?, SSRN Electron J
27. Kushwaha SS, Joshi S, Singh D, Kaur M, Lee HN (2022) Systematic review of security vulnerabilities in ethereum blockchain smart contract. IEEE Access 10:6605–6621
28. Zou W, Lo D, Kochhar PS, Le XBD, Xia X, Feng Y, Chen Z, Xu B (2019) Smart contract development: challenges and opportunities. IEEE Trans Softw Eng 47(10):2084–2106

29. Nxt whitepaper (2016) Available online at [https://nxtdocs.jelurida.com/Nxt Whitepaper](https://nxtdocs.jelurida.com/Nxt%20Whitepaper). Last accessed: 02-18-2023
30. Ethereum whitepaper. Available online at <https://ethereum.org/en/whitepaper/> Last accessed on 02-18-2023
31. Nordberg W (2021) A SYSTEMATIC MAPPING STUDY ON DEVELOPMENT OF BLOCKCHAIN-BASED SMART CONTRACTS

A Comprehensive Study of Blockchain Technology Trends and Analysis in the Healthcare Industry 4.0



Rakshit Bhadaria, Puneeta Singh, and Sartaj Ahmad

Abstract Blockchain could be a modern innovation that's being worked out to provide innovational results in various diligence, including healthcare. Within the healthcare region, a blockchain network is used to store and share quiet information among clinics, person labs, medication companies, and croakers. Blockchain operations may appropriately descry serious crimes, including those that are parlous, within the therapeutic assiduity. Within the healthcare region, it may in this way improve the viability, screen, and translucency of taking part therapeutic information. With the utilize of this innovation, therapeutic teach may more get it and test persistent commentaries. In this paper, we showed up at blockchain innovation and its imperative vantages for the healthcare assiduity. Plates are worked out to illustrate how blockchain innovation may back worldwide healthcare through its various capabilities, enablers, and solidified work-flow handle.

Keywords Blockchain · Digital signature · Healthcare · Capabilities · Data Storage · Clinical Trials · Cryptography

1 Introduction

Satoshi Nakamoto first introduced blockchain (BC) in 2008, and it is now the area of attention for many companies due to its function in the transformation of operational processes. Blockchain is a decentralized open-source computerized record that records exchanges over various computers in a way that precludes the retroactive adjustment of any record without too changing any ensuing blocks. Blockchain is an extensive network of linked “blocks” that have been successfully defended. Because every transaction is documented and supported in public, blockchain creates a lot

R. Bhadaria · P. Singh (✉) · S. Ahmad
Department of IT, KIET Group of Institutions, Delhi-NCR, Ghaziabad,
Uttar Pradesh 201206, India
e-mail: puneeta.singh@kiet.edu

R. Bhadaria
e-mail: rakshit.2024it1020@kiet.edu

of duty [1]. The info that has been added to the blockchain cannot be changed by any bone. It serves as affirmation that the information is genuine and unaltered. Blockchain keeps up information on systems instead of a single database, progressing unwavering quality, and lessening defenseless spots in frameworks. Blockchain offers a great venue for creating and competing with established businesses for creative and striking business ideas.

Blockchain enables marketers to keep track of the particulars utilized in the pharmaceutical assiduity. Blockchain technology will help the health and medicinal diligence get relieve of fake specifics by making it possible to trace all of these drugs. It aids in relating the root of falsity. Blockchain can insure the sequestration of patient information; it can also save medical history when it's established, and this record can not be changed. The sanitarium uses this decentralized network with all common tackle. With the coffers saved by these bias, experimenters are suitable to calculate estimations for treatments, specifics, and curatives for colorful affections and conditions [5].

Blockchain is a distributed record network that is continuously adding new entries and never changes them unless everyone agrees [9]. The distributed blockchain record plan guarantees that information isn't taken care of in any centralized area, making it available and capable to all organize clients. Its decentralized architecture makes system stronger and more secure. Greater control over patient care and health data is made possible by lessening the amount of medical practice and tracking and saving time and resources for patients as well as doctors. The user will be able to monitor information usage by having their health data stored on a blockchain [5].

2 Blockchain Technology

2.1 *Blockchain Overview*

For the first time, Satoshi Nakamoto—the person who invented the first cryptocurrency known as Bitcoin—highlights blockchain technology. Blockchain technology enables the storing and transmission of transactions. It keeps the information in a block-based ledger. To create a chain of blocks, each block is joined to the one before it. A peer-to-peer network (P2P) ensures data transfer. Blockchain is a distributed ledger that is shared in a secure, decentralized fashion as a result. Blockchain has attracted a lot of attention in the banking and financial industries over the past several years. These days, it is finding use in additional fields including insurance, energy, industry, and healthcare. Because to these characteristics, blockchain becomes so well-liked in practically all fields: It is distributed, immutable, secure, and decentralized. As the network is decentralized, a centralized authority is not required to run it. To achieve consensus among nodes, data is archived using this approach. The ledger is distributed and is maintained by all network nodes.

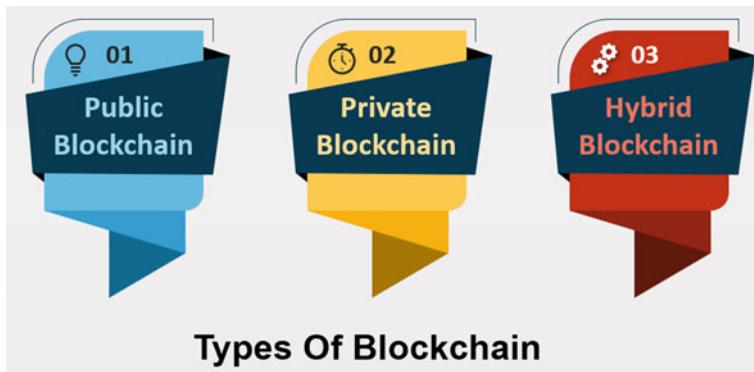


Fig. 1 Types of blockchain

Blockchain technology come in a variety of forms, including consortium, hybrid, private, and public. Each blockchain arrange has interesting benefits and downsides that on a very basic level influence the leading uses for it.

- The **Public (open) Blockchain**, Bitcoin and other coins were developed in the first blockchain invention, which is where they had a significant impact on the development of distributed ledger technology (DLT). The disadvantages of centralization, such as the requirement for protection and simplicity, are eliminated. Instead of keeping data in a single location, DLT distributes information over a P2P network. Because it is decentralized, some form of data verification (authentication) is required [6].
- A **Private Blockchain** may be a a blockchain network that operates in a small space, like a private network, or is backed by just one organization. In terms of P2P association and decentralization, it performs essentially to a open blockchain arrange but is much littler. On a private blockchain, the maker of the arrange is continuously aware of who the clients (or users) are.
- Organizations who need the leading of both universes sometimes utilize **hybrid blockchains**, a shape of blockchain that combines highlights from both private and open blockchains. Businesses may control who has get to certain information kept on the blockchain and what information is made open by permitting them to develop both a private, permission-based framework and a public, permission-less one [7] (Fig. 1).

2.2 *Consensus Algorithms*

Anonymity, a desired property of blockchain technology, is problematic when it comes to building trust. When adding events to a ledger, how can the integrity of

unnamed or anonymous individuals be completely guaranteed? The answer is to double-check every transaction to make sure it is truthful (and not malevolent) before adding it to a block. A transaction is approved to be added to the blockchain using consensus methods [7]. The majority of blockchain users want to keep the blockchain's integrity, which is advantageous for these agreement methods. A blockchain framework employs a consensus method to construct confidence and legitimately report occasions on the blocks. All blockchain activities can be conceived of as being controlled by consensus algorithms [31].

Simply put, a consensus procedure is a collection of rules that each member must follow. Since blockchain is a dispersed or distributed system without a singular source of confidence or trust, everyone must concur on its current state, necessitating the use of a distributed agreement technique [12]. A wide range of novel consensus mechanisms have been used to build blockchains, including PoW, PoS, PoI, PoET, PBFT, DAG, SCP, and PoA, DPoS and PoC (Table 1).

2.3 Smart Contract

Blockchain grants for the composing of exceedingly objective coding framework that shows accurately how that get ready is advancing to be controlled and what measures are advancing to be taken when that event happens, in development to giving a scattered, permanent record of all the various occurrences that have occurred. **Smart contracts** add even more appeal to this. The limitations of Bitcoin were one of the goals of the smart contract that Ethereum proposed. The thought of a smart contract is coding framework made to answer to particular sorts of significant events. It is not necessary for the smart contract to have more than two participants or to be formally binding [24].

A smart contract is another term for **chaincode**. The smart contract, which serves as the basis for corporate blockchain apps, will transform the way we conduct business. Without the use of intermediaries, anybody can create smart contracts. Independence, effectiveness, efficiency, and expense reductions are all provided by the smart contract.

2.4 Cryptography Technology used in Blockchain

Blockchain adds a degree of trust between unreliable parties, enabling the creation of secure documents and transactions. Without blockchain, trustworthy documents and deals must be produced by a third-party mediator [27]. Blockchain creates trust through encryption and collaboration, eliminating the need for a centralized entity to function as an intermediary. Cryptography is used to store information on the blockchain ledger [28].

Table 1 Comparison of consensus algorithms

Property	PoW	PoS	PBFT	PoET	DPoS
Full form	Proof-of-work	Proof-of-stake	Practical-Byzantine-fault-tolerance	Proof-of-elapsed-time	Delegated-proof-of-stake
Blockchain type	Permission less	Both	Permissioned	Both	Both
Token needed	Yes	Yes	No	No	Yes
Examples	Bitcoin, litecoin, Ethereum, Zcash, decred, dash, etc	Cardano, Ethereum 2.0, polkadot peercoin, and BlackCoin	Hyperledger fabric, Zilliqa	Hyperledger sawtooth	Ark, EOS, BitShares, lisk, nano, cardano, steem, and tezos
Vulnerable to sybil attack	No	Yes	Yes	No	Yes
Block confirmation time	Bitcoin : 10 min litecoin : 2.5 minutes	Ethereum : 6 min	No actual time found	No actual time found	Bitshare : 3 sec
Transaction per second (TPS)	Bitcoin : 7	Ethereum 2.0 :	Hyperledger fabric : 20,000	Hyperledger Sawtooth : 2300	Bitshare : 10,000
Incentives	The winning mineworker gets unused or new coins with the piece and exchange fees within the square or block he/she approves	The victor gets exchange fees with the unused or new block. On the off chance that a square victor endeavors to include an invalid block, he/she loses his/her stake	Nil	The winning mineworker gets the exchange fees with the unused block he/she approves	The danger of misfortune of reputation and wage gives an motivation for delegates to act truly and keep the organize secure

Several cryptography building blocks used by blockchain are listed below:

- **Public Key Cryptography (Asymmetric Cryptography):** It can be used to conduct cryptography and digital authentication.
- **Zero-Knowledge Proof:** Show your knowledge of a secret and don't reveal it.
- **Hash Functions (Hash Algorithm) :** Mathematical operations that are one-way pseudo-random.

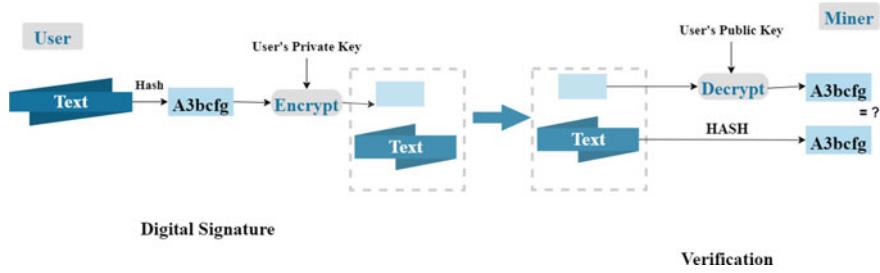


Fig. 2 Digital signature

Public Key Cryptography (Asymmetric Cryptography). The private key is kept in a blockchain-compatible computerized wallet, which can be either a hardware wallet (which physically holds the secret key) or any online wallet. A user's open (public) key is utilized to assert that the message truly began from them, and their private key is utilized to seal a message known as a digital signature that will be sent to the blockchain. For example, in Fig. 2, the client (or sender) encrypts its private key with hash value X sometime recently marking on that hash value to produce a digital signature. The client (or sender) at that point sends their digital signature and details about transactions toward the blockchain architecture. The miner nodes uses the client's open key to translate the gotten or created a hash value X using the received digital signature, after hashing the received exchange information to create another hash value Y . The miner nodes at that point checks to see on the off chance that hash value X and hash value Y are identical. The processor determines whether the user's transaction is comparable [8].

The comparing digital signature ensures transaction origin since the private key can as it were be securely put away by its user. The method enables the application of a digital signature to every transaction based on each user's particular secret key. Blockchain is built on the public (open) key and private key pair, which is used to verify and authenticate user transactions.

Digital signatures on transactions and blocks are used by both Ethereum as well as Hyperledger Fabric to verify the identify of the author and the integrity of the signed data. Widespread creation of a collection of public key and private key pairs using the Elliptic Curve Digital Signature Algorithm (ECDSA).

Zero-Knowledge Proofs. The taking after is one of the foremost utilize cases for zero-knowledge proofs in blockchain. When a client requests to transmit cash to another client, the blockchain will ordinarily ought to confirm that client has satisfactory cash accessible before committing this trade. The blockchain, in any case, is less concerned with who is investing the cash than it is with how much cash they have in add up to. The blockchain in this instance has no idea who or how much money the user is transferring and to whom they are sending it.

Several blockchains employ the cryptographic idea of zero-knowledge proofs to improve user privacy. Although Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARKS), a particular kind of zero-knowledge proof, is

presently listed in the Ethereum development roadmap, Zero-knowledge proofs are not backed by Ethereum at this time.

Hash Functions (Hash Algorithm). A crucial piece of blockchain technology is hash functions. A mathematical equation known as a “hash function” contains five crucial characteristics for cryptography:

- **Fixed size.** Anything can be the input for a hash function, which produces a fixed-size yield. Since of this, anything may be decreased to a single, fixed-size piece of information. Consequently, blockchains use hash functions to compress messages for digital signatures.
- **Preimage resistance.** It is straightforward to decide a hash yield given an input. However, it is hypothetically outlandish to recreate the initial input given the hash yield. In reality, the as it were strategy to urge the same result is to arbitrarily input the information into the hash function.
- **2nd preimage resistance.** It is computationally outlandish to get a moment input that produces the same hash yield from a given input and its hash yield .
- **Collision resistance.** It is computationally inconceivable to urge the same hash result from any two distinctive inputs.
- **Big change.** A totally distinctive hash yield will result in case even one bit of the input is changed.

A blockchain’s initial block, known as a “**Genesis Block**,” will have a hash value of 0 and not that of any preceding blocks [29].

Figure 3 demonstrates how the cryptographic hash function may be used to connect all of the blocks in a blockchain. The hash of *Block x-2* is recorded in the “previous block hash” field in *Block x-1*, the hash of *Block x-1* is recorded in the “previous block hash” field in *Block x*, the hash of *x* is recorded in the “previous block hash” field in *Block x+1*, and so on.

3 Need of Blockchain in Healthcare

Improvement within the field of healthcare is advancing at ever-astonishing rates. Present day restorative centers that are bolstered by cutting-edge innovation are in tall request nowadays. In this case, blockchain may play a noteworthy part in revolutionizing the healthcare segment. Moreover, the well-being framework environment is changing in favor of a patient-centered approach that emphasizes two vital components [30]: the availability of fitting therapeutic assets at all times and effectively open administrations. Blockchain innovation helps healthcare organizations in giving great quiet treatment and civilities. With this innovation, the long and tedious handle of trading well-being data, which includes to high healthcare costs, may be rapidly tended to. Blockchain innovation empowers open interest in wellbeing think about ventures.

Up until now, the greatest obstacles to community health management have been data exchange, privacy, and interoperability concerns. The use of blockchain guaran-

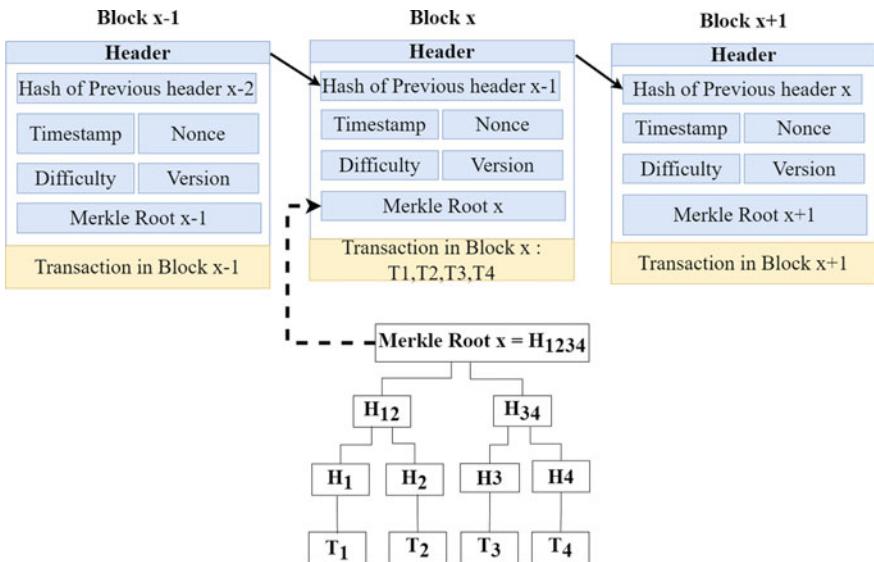


Fig. 3 A Merkle tree with a hash algorithm and a blockchain layout

tees this issue's resolution. When used properly, this technology improves security, data sharing, compatibility, reliability, real-time updating, and access. Data protection is a major concern as well, especially in the context of wearable technology and individualized care. These issues are addressed by blockchain technology because patients and medical professionals require straightforward, secure methods of data collection, transmission, and consultation over networks without security concerns.

4 Different Blockchain Technological Capabilities to Support the Global Healthcare Culture

The healthcare division can advantage from utilizing blockchain in a assortment of ways. Record innovation helps therapeutic researchers in interpreting hereditary code by empowering the secure exchange of persistent restorative information, overseeing the sedate supply chain, and facilitating this exchange [5, 8]. Figure 4 illustrates the different characteristics and key blockchain concept drivers within the healthcare and related businesses. Assurance of therapeutic records, administration of different qualities, electronic information administration, interoperability, digitalized issue monitoring, etc. are fair many of the momentous and experimentally determined components utilized to create and apply blockchain technology. The totally digitalized features and employments of blockchain innovation within the healthcare industry are the most powers behind its take-up.

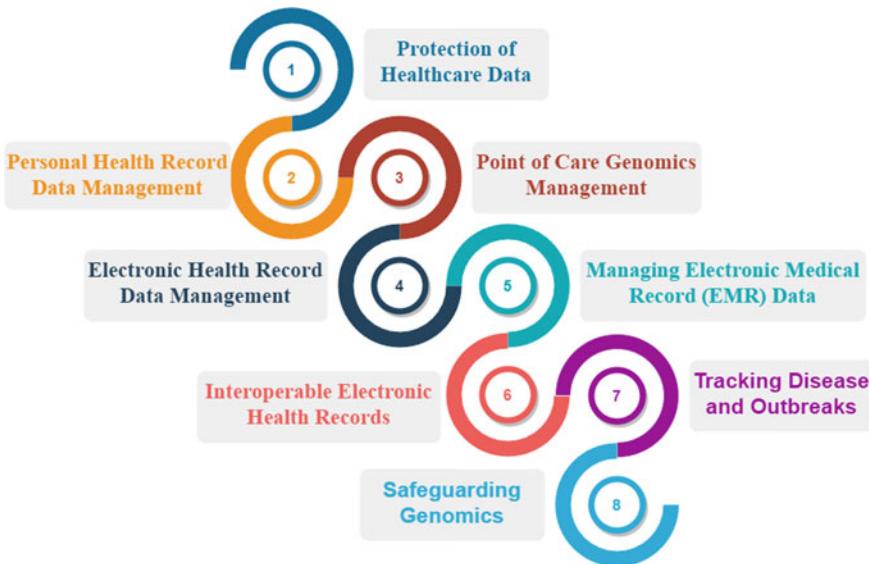


Fig. 4 Features of blockchain for healthcare domain

From create to sedate racks, all medication prepare is clear on account of the blockchain. IoT and blockchain acquiesce the plausibility be utilized to way activity, carriage course, and speed. It supplies the comfort to arrange buys effectively to anticipate delays and deficiencies of a specific sedate in drug stores, healing centers, and included wellbeing care organizations. Blockchain-located scientific establishments cheerful utilized to help ensure that unauthorized changes to the procedural-dossier are avoided. It builds believe and anticipates the unlawful taking care of of records, stores, and prescription by different people inquisitive about getting drugs. The utilize of innovation can essentially upgrade patients' conditions whereas keeping costs moo. It expels all barricades and limitations for multi-level authentication. Blockchain is perfect for security applications since it can keep up an unalterable, decentralized, and straightforward log of all restorative information. Be that as it may, blockchain is both straightforward and private, concealing any person's character behind complex and secure calculations that can keep up the affectability of therapeutic information. Patients, therapeutic experts, and healthcare offices may all rapidly and safely trade the same data since to the technology's decentralized nature [24].

Since blockchain innovation empowers patients to form their restorative information accessible , it encourages the move to interoperability driven by patients. This increments mystery and protection whereas giving patients more control over their individual data. Quality administration and authorization estimation and execution are challenging. Applications of the blockchain within the segment might resolve any of these innovative issues. Blockchain news will offer assistance administrative

authorities recognize bona fide solutions from fake ones whereas following drugs. This ensures the trade of computerized exchanges counting the patient's data between all approved parties. Patients who switch specialists can share all of their records by upgrading a single assent.

An increasing number of businesses are embracing blockchain, including the healthcare sector. Also, in the first phases, the technologies are well-received by those involved in the health ecosystem. The challenges affecting the current structure will be tended to as portion of blockchain's comprehensive arrange to change the healthcare trade within the up and coming a long time. It makes all the data effectively open to specialists, patients, and drug specialists at any time. Restorative companies are testing, investigating, and learning about blockchain innovation within the restorative region for well-being records day and night. By embracing medicines, upgrading installment options, and decentralizing persistent well-being history data, it has set up itself as an vital apparatus in healthcare. Blockchain may be a pivotal innovation for the therapeutic industry, in expansion to other progressed advances like machine learning and counterfeit intelligence. There are a few true blue ways that blockchain is changing the healthcare segment. The framework is planned to screen the therapeutic supply chain utilizing blockchain innovation [32].

With the utilize of blockchain innovation, a complex information capacity framework can be made that keeps track of a person's entirety restorative history, counting analyze, test comes about, past regimens, and indeed estimations made by cleverly sensors. With this procedure, a specialist may effortlessly get all the data required to create exact determination and proposals. A single blockchain framework stores all the information, securing it from misfortune and alter. Blockchain can be utilized to dodge an organization's inner systems. A considerable association with a few free members, with distinctive degrees of authority on a blockchain database that's scrambled, may ensure associations from dangers and assaults from the exterior world. Protect assaults and other issues, counting computer debasement or equipment breakdown issues, will be dispensed with in case a healthcare association employs a blockchain arrange appropriately [32].

5 Literature Review

5.1 Effectiveness Data Management

Data Accountability Taking responsibility for one's actions, carrying out obligations, and being held accountable

Data Integrity/Security

- **Integrity:** Only when approved changes are carried out on or with the data does it remain in its authorized condition, maintaining integrity.
- **Security:** safeguarding digital data from being destroyed by unauthorized users

Data Confidentiality Ensure that the information received by the data receiver is entirely consistent with the information given by the sender and to avoid active attacks on users' data by unauthorized parties.

Data Transparency/Traceability

- **Transparency** the capacity of persons to efficiently have access to any data related information utilized in processes and choices that impact them.
- **Traceability** the ability to identify and verify the components and chronology of occurrences at all phases of a process chain.

Data Reliability The capacity of a supply chain to consistently meet end customer demand at the appropriate level across the planning horizon, irrespective risks of external and/or internal shocks to the system and before any risk mitigation actions, is known as supply chain reliability.

5.2 Study

Theoretic

- Basic rules were followed.
- Theory of technology developed.
- Experimentation proof of concept.

Simulated

- The technology has been verified in the lab.
- Technology tested in an appropriate setting.
- Technology shown in an appropriate environment.

Real case

- Demonstration of a system prototype in a working environment.
- Complete and certified system.
- Actual system that has been tested in a working setting.

5.3 Scalability

the capacity of blockchain technology to manage future expansion and massive data transaction volume (Table 2).

- PB = Public BlockChain
- PRB = Private Blockchain

Table 2 Outline of the included studies

Ref. No.	Year	Sector	Type of blockchain used	DA	DI	DT	DC	DR	Smart contract	Study type	Scalability
[21]	2022	Tools	PB	✓	✓	✓	✓	✓	Yes	S	No
[7]	2022	Vaccine	C	✓	✓	✓	✗	✓	Yes	S	Not explored
[32]	2022	Drugs	PRB	✗	✓	✓	✓	✓	Yes	S	Not explored
[8]	2022	DTV	PRB	✗	✓	✓	✓	✗	Yes	S	Not explored
[1]	2022	Drugs	H	✗	✗	✓	✗	✗	No	S	Yes
[17]	2022	DTV	H	✗	✓	✓	✓	✓	Yes	S	Not explored
[6]	2022	Drugs	H	✗	✓	✓	✓	✓	Yes	S	Not explored
[12]	2022	Drugs	PB	✓	✓	✓	✗	✓	Yes	T	Not explored
[29]	2021	Drugs	C	✓	✓	✓	✓	✓	Yes	T	Not explored
[30]	2021	Drugs	PRB	✗	✓	✓	✓	✗	Yes	T	Not explored
[31]	2021	Drugs	C	✗	✗	✓	✗	✓	No	RCS	Not explored
[22]	2021	Drugs	C	✗	✓	✓	✓	✗	Yes	T	Not explored
[24]	2021	Vaccine	PRB	✗	✗	✓	✗	✗	No	T	Not explored
[5]	2021	Drugs	PRB	✗	✓	✓	✓	✗	Yes	T	Yes

- C = Consortium
- H = Hybrid
- DA = Data Accountability
- DI = Data Integrity/Security
- DT = Data Transparency/Traceability
- DC = Data Confidentiality/Privacy
- DR = Data Reliability
- S = Simulated
- T = Theoretic
- RCS = Real Case Study
- ✓ = solve this problem
- ✗ = doesn't solve this problem

6 Conclusion

Blockchain innovation has imaginative applications within the healthcare segment since of its built-in mystery and decentralization. It advances the commercialization of wellbeing information, moves forward network between healthcare organizations, and bolsters the creation of anti-counterfeit sedate innovations. It too makes understanding computerized restorative records more secure. A few perspectives of the healthcare division may alter as a result of the utilize of blockchain innovation; among the foremost critical applications of blockchain are the digital bargains made conceivable by intelligent contracts within the healthcare segment. By killing mediators from the payment process, savvy contracts will decrease costs. The potential of the blockchain within the healthcare industry is altogether affected by the take-up of related cutting-edge advances within the environment. Clinical investigate, medicate checking, wellbeing protections, and system following are all secured. Clinics can arrange out their administrations over the span of their whole life cycle employing a blockchain design and gadget tracking. With way better following and protections intervention, blockchain innovation incorporates a parcel of guarantee to move forward understanding data organization. This would speed up restorative exercises whereas moving forward information conservation. In general, this innovation would incredibly improve how clinical records are utilized by patients and doctors, which would eventually revolutionize how these records are utilized and taken care of.

References

1. Agrawal D, Minocha S, Namasudra S, Gandomi AH (2022) A robust drug recall supply chain management system using hyperledger blockchain ecosystem. *Comput Biol Med* 140:105100
2. Ali A, Pasha MF, Ali J, Fang OH, Masud M, Jurcut AD, Alzain MA (2022) Deep learning based homomorphic secure search-able encryption for keyword search in blockchain healthcare system: a novel approach to cryptography. *Sensors* 22(2):528

3. Alzahrani AG, Alhomoud A, Wills G (2022) A framework of the critical factors for healthcare providers to share data securely using blockchain. *IEEE Access* 10:41064–41077
4. Azbeg K, Ouchetto O, Andaloussi SJ (2022) Blockmedcare: a healthcare system based on iot, blockchain and ipfs for data management security. *Egypt Inf J* 23(2):329–343
5. Bali V, Soni P, Khanna T, Gupta S, Chauhan S, Gupta S (2021) Blockchain application design and algorithms for traceability in pharmaceutical supply chain. *Int J Healthc Inf Syst Inf (IJHSI)* 16(4):1–18
6. Chiacchio F, D'Urso D, Oliveri LM, Spitaleri A, Spampinato C, Giordano D (2022) A non-fungible token solution for the track and trace of pharmaceutical supply chain. *Appl Sci* 12(8):4019
7. Cui L, Xiao Z, Chen F, Dai H, Li J (2022) Protecting vaccine safety: an improved, blockchain-based, storage-efficient scheme. *IEEE Trans Cybern*
8. El Azzaoui A, Chen H, Kim SH, Pan Y, Park JH (2022) Blockchain-based distributed information hiding framework for data privacy preserving in medical supply chain systems. *Sensors* 22(4):1371
9. Fiore M, Capodici A, Rucci P, Bianconi A, Longo G, Ricci M, Sammarchi F, Golinelli D (2023) Blockchain for the healthcare supply chain: a systematic literature review. *Appl Sci* 13(2):686
10. Govindan K, Nasr AK, Saeed Heidary M, Nosrati-Abarghooee S, Mina H (2022) Prioritizing adoption barriers of platforms based on blockchain technology from balanced scorecard perspectives in healthcare industry: a structural approach. *Int J Prod Res* 1–15
11. Hajian A, Prybutok VR, Chang H-C (2023) An empirical study for blockchain-based information sharing systems in electronic health records: a mediation perspective. *Comput Hum Behav* 138:107471
12. Humayun M, Jhanjhi NZ, Niazi M, Amsaad F, Masood I (2022) Securing drug distribution systems from tampering using blockchain. *Electronics* 11(8):1195
13. Jayabalan J, Jeyanthi N (2022) Scalable blockchain model using off-chain ipfs storage for healthcare data security and privacy. *J Parallel Distrib Comput* 164:152–167
14. Jeet R, Kang SS, Saiful Hoque SM, Dugbakie BN (2022) Secure model for iot healthcare system under encrypted blockchain framework. *Secur Commun Netw* 2022
15. Kamruzzaman M, Yan B, Sarker MNI, Alruwaili O, Wu M, Alrashdi I (2022) Blockchain and fog computing in iot-driven healthcare services for smart cities. *J Healthc Eng* 2022
16. Kumar A, Singh AK, Ahmad I, Kumar Singh P, Verma PK, Alissa KA, Bajaj M, Ur Rehman A, Tag-Eldin E (2021) A novel decentralized blockchain architecture for the preservation of privacy and data security against cyberattacks in healthcare. *Sensors* 22(15):5921
17. Li D, Gong Y, Zhang X, Huang M (2022) An exploratory study on the design and management model of traditional Chinese medicine quality safety traceability system based on blockchain technology. *Secur Commun Netw* 2022
18. Mahajan HB, Rashid AS, Junnarkar AA, Uke N, Deshpande SD, Futane PR, Alkhayyat A, Alhayani B (2022) Integration of healthcare 4.0 and blockchain into secure cloud-based electronic health records systems. *Appl Nanosci* 1–14
19. Neelakandan S, Beulah JR, Prathiba L, Murthy G, Irudaya Raj EF, Arulkumar N (2022) Blockchain with deep learning-enabled secure healthcare data transmission and diagnostic model. *Int J Model Simul Sci Comput* 13(04):2241006
20. Nishi FK, Shams-E-Mofiz M, Khan MM, Alsufyani A, Bourouis S, Gupta P, Saini DK (2022) Electronic healthcare data record security using blockchain and smart contract. *J Sens* 2022:1–22
21. Omar IA, Debe M, Jayaraman R, Salah K, Omar M, Arshad J (2022) Blockchain-based supply chain traceability for covid-19 personal protective equipment. *Comput Indust Eng* 167:107995
22. Pandey P, Litoriya R (2021) Securing e-health networks from counterfeit medicine penetration using blockchain. *Wirel Pers Commun* 117:7–25
23. Pradhan NR, Singh AP, Verma S, Kaur N, Roy DS, Shafi J, Wozniak M, Ijaz MF (2022) A novel blockchain-based healthcare system design and performance benchmarking on a multi-hosted testbed. *Sensors* 22(9):3449

24. Saranya S (2021) Go-win: Covid-19 vaccine supply chain smart management system using blockchain, iot and cloud technologies. *Turkish J Comput Math Educ (TURCOMAT)* 12(12):1460–1464
25. Sharma P, Namasudra S, Crespo RG, Parra-Fuente J, Trivedi MC (2023) Ehdhe: enhancing security of healthcare documents in iot-enabled digital healthcare ecosystems using blockchain. *Inf Sci* 629:703–718
26. Singh P, Singh AP, Gupta A (2022) Design strategies for mobile ad-hoc network to prevent from attack
27. Singh P, Verma S et al (2019) Analysis on different strategies used in blockchain technology. *J Comput Theor Nanosci* 16(10):4350–4355
28. Sinha P, Singh R, Roy R, Singh P (2022) Education and analysis of autistic patients using machine learning. In: 2022 international conference on emerging smart computing and informatics (ESCI). IEEE, pp 1–6
29. Uddin M (2021) Blockchain medledger: hyperledger fabric enabled drug traceability system for counterfeit drugs in pharmaceutical industry. *Int J Pharm* 597:120235
30. Uddin M, Salah K, Jayaraman R, Pesic S, Ellahham S (2021) Blockchain for drug traceability: architectures and open challenges. *Health Inf J* 27(2):14604582211011228
31. Yik MH-Y, Wong VC-WT, Wong T-H, Shaw P-C (2021) Herbchain, a blockchain-based informative platform for quality assurance and quality control of herbal products. *J Tradit Complement Med* 11(6):598–600
32. Zoughalian K, Marchang J, Ghita B (2022) A blockchain secured pharmaceutical distribution system to fight counterfeiting. *Int J Environ Res Publ Health* 19(7):4091
33. Zulkifl Z, Khan F, Tahir S, Afzal M, Iqbal W, Rehman A, Saeed S, Almuhaideb AM (2022) Fbashi: fuzzy and blockchain-based adaptive security for healthcare iots. *IEEE Access* 10:15644–15656

Web 3.0-Based Crypto Wallet for Securing Assets and Blockchain Transactions



Vaibhav and Deepak Arora

Abstract The advent of Web 3.0 has opened up new possibilities for the development of secure, decentralized digital solutions. Having a safe and user-friendly cryptocurrency wallet is essential in this new era where digital assets and cryptocurrencies are growing in popularity among users. Customers may safely and easily manage, store, and transact with their digital assets using a crypto wallet. But developing a Web 3.0-based crypto wallet presents a variety of challenges, including those pertaining to user experience, scalability, and security. To address these issues, we utilized cutting-edge software such as Next.js, Sanity.io, and ThirdWeb. Using Next.js, Sanity.io, and ThirdWeb, a Web 3.0-based crypto wallet was created that offers a strong method for handling digital assets securely and carrying out blockchain transactions. We can develop a secure, scalable, and user-friendly crypto wallet that caters to the needs of today's owners of digital assets by integrating the capabilities of these technologies. An appropriate solution for the administration of digital assets in the Web 3.0 era, crypto wallets are developed using Web 3.0 technologies like Next.js, Sanity.io, and ThirdWeb to guarantee that they are safe, scalable, and user-friendly.

Keywords DApp · Nextjs · Sanity · ThirdWeb · MetaMask

1 Introduction

Web 3.0 is the next stage of the World Wide Web, which aims to provide a more decentralized, secure and open Internet for users. With the rise of digital assets and blockchain technology, the development of secure and user-friendly crypto wallets has become increasingly important. A crypto wallet allows users to store, manage and perform transactions with their digital assets, such as cryptocurrencies. The

Vaibhav · D. Arora (✉)

Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Uttar Pradesh, Lucknow, India
e-mail: deepakarainbox@gmail.com

Vaibhav

e-mail: vbhvka@gmail.com

development of a Web 3.0-based crypto wallet presents a number of challenges, including security, scalability, and user experience. To address these challenges, many developers are turning to cutting-edge technologies like Next.js, Sanity.io, and ThirdWeb. Next.js is a popular React-based framework that enables developers to build fast, scalable, and dynamic web applications. With Next.js, developers can create single-page applications (SPAs) that provide a seamless user experience and are easily scalable as the user base grows. This makes Next.js an ideal choice for developing a crypto wallet, as it enables developers to create a responsive and user-friendly interface that can handle large amounts of data and transactions. Sanity.io is a real-time headless content management system (CMS) that allows developers to store and manage the content of their web applications. With Sanity.io, developers can easily store data such as user profiles, transactions, and asset information, and access this data in real-time from their Next.js app. This makes Sanity.io a powerful tool for developing a crypto wallet, as it enables developers to store and manage sensitive information in a secure and reliable manner. ThirdWeb is a decentralized platform for building and hosting Web 3.0 applications. By using ThirdWeb, developers can take advantage of the decentralized architecture of Web 3.0 to provide a more secure and scalable solution for their users. ThirdWeb also provides a number of features and tools for developers, such as decentralized authentication, data storage, and consensus algorithms, making it an ideal choice for developing a Web 3.0-based crypto wallet. By combining these technologies, you can develop a Web 3.0-based crypto wallet that provides users with a secure and user-friendly way to manage their assets and perform blockchain transactions. The wallet can also be integrated with various blockchain platforms, such as Ethereum or Bitcoin, to provide a seamless and secure experience for users.

2 Literature Survey

Using Web 3.0 and Ethereum Blockchain Technology, Adrian Petcu conducted a survey on Web3 authentication, which enables safe and decentralized user authentication on the web, as the next major advancement in the World Wide Web. To improve user interaction and usability, it makes use of a modern web stack and Ethereum as the blockchain. It has the potential to emerge as the primary method of authentication for all decentralized online applications [1]. Ersoy released a paper which introduces a method called MetaRepo, where users can safely store digital assets and use them in a variety of metaverse-based activities. For user interaction, transaction processing, and security measures, blockchain technology, New User Engine, Transaction Centre, Authenticator Engine (Weng), and Repos models have been designed [2]. Pilkington explained the fundamental ideas underlying blockchain technology as well as some of its cutting-edge uses. Beginning from outlining the fundamental ideas at the foundation of the blockchain, it talked about the potential dangers and disadvantages of public distributed ledgers as well as the trend toward

hybrid solutions. Second, it highlights the key characteristics of platforms for decentralized public ledgers. Fourthly, this draws out a list of significant applications while taking into account the most recent developments. Thirdly, this demonstrates why the blockchain is a disruptive and foundational technology [3]. Chatterjee conducted a study on exclusively programmable blockchains presently support smart contracts, which are exclusively deterministic and non-probabilistic. This absence of randomness is a constraint since a large variety of real-world financial contracts, such as casino games and lotteries, completely depend on unpredictability. As a result, a number of ad-hoc ways to random number generation have been created for use in smart contracts. These include concepts like utilizing an oracle or putting your trust in the block hash. These methods can, however, be altered. This suggests a novel game-theoretic method for creating on the blockchain pseudorandom numbers that are presumably impossible to manipulate. A new generation of smart contracts that are not limited to being non-probabilistic and can be derived from the much more general class of probabilistic programs can be created using our method. This generation of smart contracts may access a trustworthy randomness source without depending on oracles or miners who may be vulnerable to hacking [4]. Vikas discussed obstacles, data dissemination, data security, and immutability of the blockchain were all taken into account. An open source data distribution system is employed to distribute the data, while hashing was used to establish security and the Merle tree concept to provide immutability. This paper presents unique approaches to prevent fraudulent data upload using MIME, cross-site scripting, and cross-site request manipulation [5]. Gupta conducted a study which allows the votes to be divided to other candidates if a clear majority cannot be obtained, and is introduced in this work. The organization may potentially alter this “majority” component to suit their needs. They discuss the architecture of the blockchain-based preferential e-voting system using the solidity programming language, which gives the concept of giving preference to the candidates instead of one vote per candidate [6]. Andreas presented all the necessary guidance on developing DApps and smart contracts on the Ethereum and other virtual machine blockchains in this helpful tutorial [7]. Kevin explained how to create your own smart contract, create an interface that is user-interactive, and more. For individuals who want to work in the smart contract sector but aren’t sure where to start, this is the perfect tool [8]. Sander conducted the study which deals with the development of a blockchain- and RFID-based methodology for food supply chain traceability. The concept coupled a central database with a blockchain for data storage and was centered on the traceability of environmental data throughout the various stages of the food supply chain. The environmental data were stored on a blockchain, whilst the lot identification data for the several supply chain steps were preserved in a centralized database. This ensured the reliability and accuracy of the traceability data [9]. Mukhopadhyay proposed and build Ethereum and blockchain smart contracts. They started off by going through the basics of blockchain. When they went into great detail on Ethereum virtual machines, he might then learn more about Ethereum and smart contracts (EVM). Users also gain knowledge of DApps and DAOs and get to see how they work. They also explore sophisticated smart contract techniques from a pragmatic standpoint [10]. Abdullah outlined Kerberos implementation flaws and

identifies authentication standards that can improve Big Data security in dispersed situations. The suggested improvement addresses Kerberos's drawbacks by utilizing the emerging blockchain technology [11]. Chris explained the usage of Ethereum, assess DApps against web applications, create smart contracts in solidity, connect those contracts to HTML/CSS/JavaScript web applications, and deploy DApps [12]. Henry proposed that how blockchain-based applications can offer robust privacy assurances. To protect access privacy, many experts advise adopting anonymous communication networks like Tor. This motivates blockchain researchers to consider alternatives to Tor and address these significant access privacy issues head-on [13]. Rohit has proposed an approach that how blockchain transactional addresses can be classified for lawful and unlawful activities over blockchain [14, 15]. Khatri has presented a systematic analysis on blockchain integration with healthcare domain [16].

3 Experimental Setup

Frontend development has evolved over the years, with various tools and technologies making it easier to build fast, responsive, and scalable web applications that can handle complex data structures and user interactions. We build a frontend using Next.js and Sanity.io and integrate it with a backend using ThirdWeb. Next.js is a popular React framework that enables developers to build server-side rendered applications with ease. It provides an efficient developer experience, automatic code splitting, and optimized performance. Sanity.io, on the other hand, is a content management system (CMS) that allows developers to build a customizable and flexible data store for their web applications. ThirdWeb is a platform that provides a secure and user-friendly way for developers to create, manage, and interact with cryptocurrencies and blockchain-based applications. Combining these technologies can lead to a powerful and flexible web application that can handle complex data structures, user interactions, and transactional data. Setting up a frontend with Next.js and Sanity.io. The first step in building a frontend with Next.js and Sanity.io is to set up a data store using Sanity.io. This can be done by running the “sanity init” command inside the studio folder. This will create a new project and set up the necessary files and configurations. After initializing sanity, we can start creating a schema for their data. The schema will define the structure of the data and how it should be stored and accessed. We can use the Sanity.io UI to create the schema and add fields for each type of data that the frontend will need to access. After setting up the schema, we can start adding content to their data store. For instance, they can create coins such as bitcoin 2.0, Solana, Vaibhav, along with their respective prices and images. They can then use a sanity client to fetch data from sanity and incorporate sanity image URLs to fetch actual images from sanity. With this data, we can build the frontend and display it to users (Fig. 1).

Next, we can start styling the web application using styled components to make it look great and user-friendly. They can also add a graph using the ChartJS library

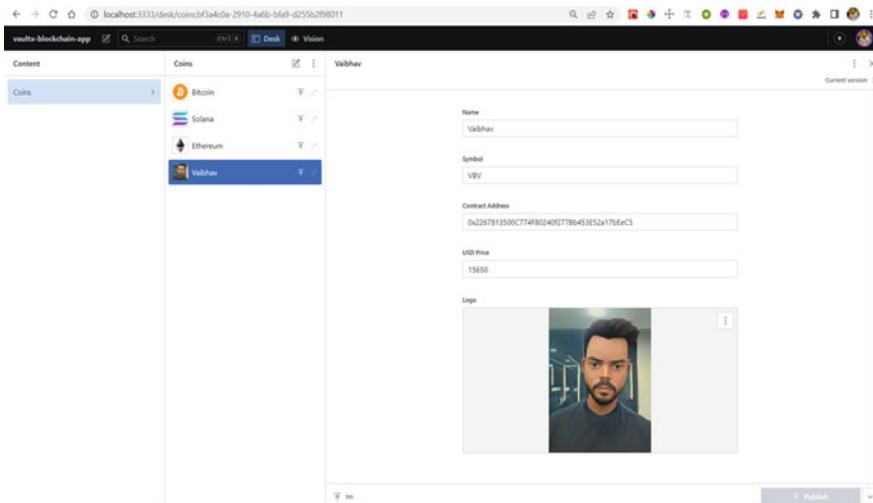


Fig. 1 Adding token details into sanity

in the app dashboard. We can query and map through each token inside the sanity database and check how many tokens we have in ThirdWeb. We can then calculate the balance and sum up the balance of each token and display it in the app. Integrating the frontend with a backend using ThirdWeb. ThirdWeb is a platform that provides a secure and user-friendly way for developers to create, manage, and interact with cryptocurrencies and blockchain-based applications. With ThirdWeb, we can create and manage tokens, and enable users to interact with the web application. They can create a token inside ThirdWeb and add details to each token, and then deploy and mint the tokens. The contract addresses of these tokens actually exist and are real, and users can check the existence of a token by simply copying the token address and pasting it into the website Goerli etherscan (Fig. 2).

The user of this application should first connect their wallet before being taken to the application dashboard, where their wallet balance is displayed with a graph along with other assets or tokens, along with information about their price, allocations, etc. and we have the ability to receive any tokens into our account as well as send them to other accounts. We utilized ThirdWeb to create, deploy, and mint each token so that we could use them in our application. We used sanity as the database where our tokens data, like their price, name, symbol, and logo, are stored. The deployed token is imported into our Metamask wallet. Our application will operate in the background to display the balance in real time. It will go through each token that sanity has provided, check its balance on ThirdWeb, add up each token's balance, and then display the overall balance in our application dashboard. For send and receive capability, it will communicate with a ThirdWeb and conduct transactions using the supplied address (Fig. 3).

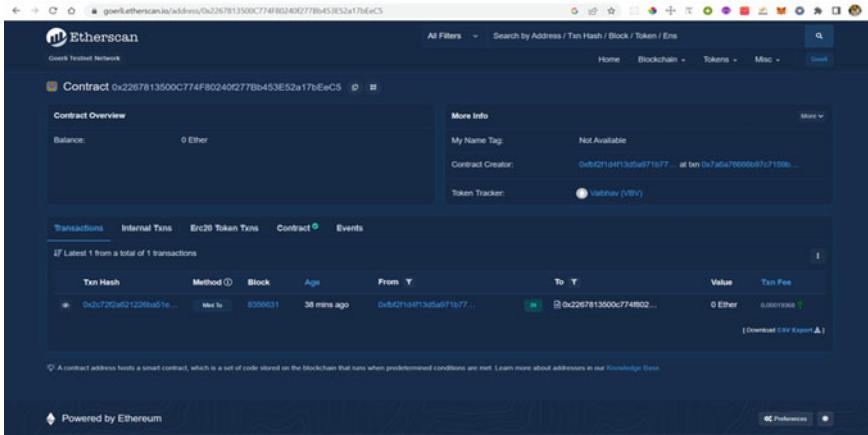


Fig. 2 Checking token existence on Goerli etherscan

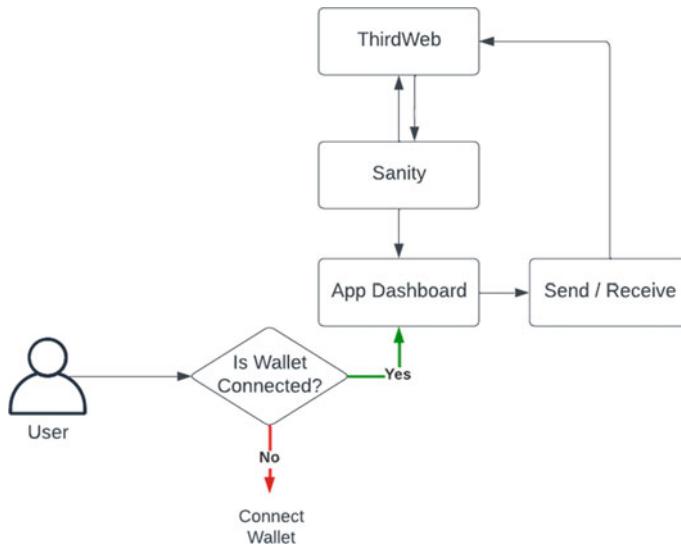


Fig. 3 Working of an application which shows how users interact with an application and what happens behind the scenes

4 Results and Discussion

The image up above shows the homepage of the vaultX web application. In order to access the application dashboard, users must first link their wallets. On the dashboard, their wallet balance is shown in a graph alongside other assets or tokens, along with details such as their price, allocations.

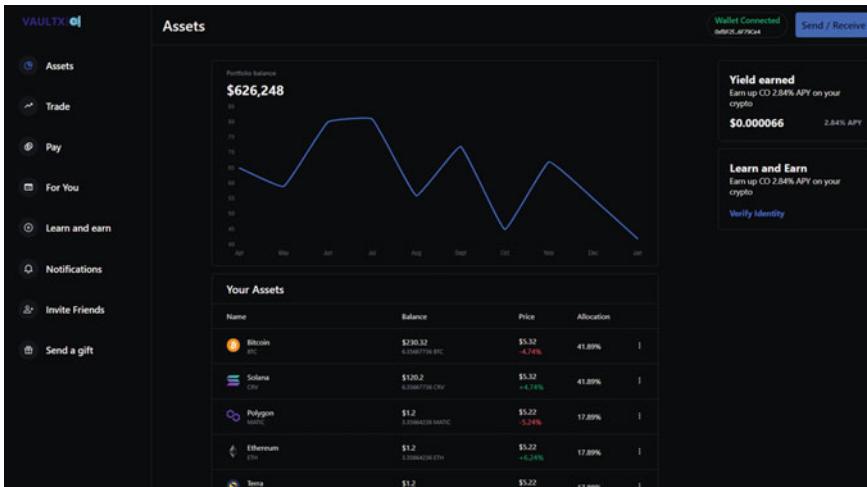


Fig. 4 Dashboard page

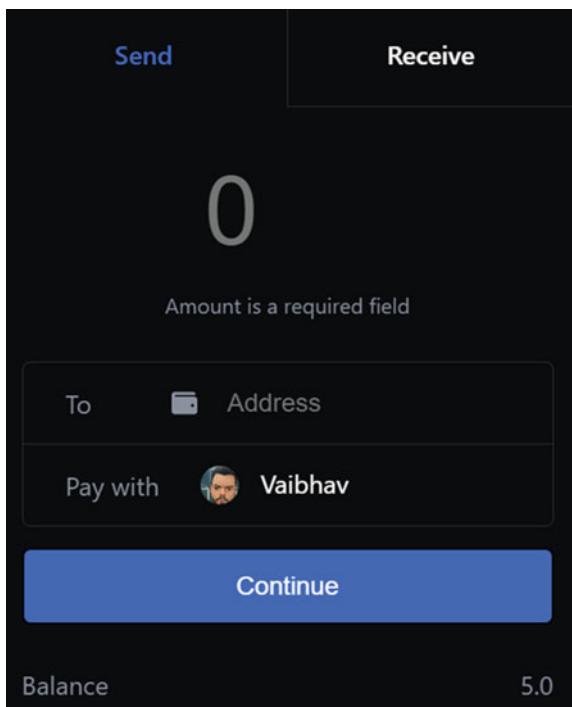
The VaultX web application Dashboard Page is displayed in the image (Fig. 4) up top. With the help of the send and receive functionality, we may both receive tokens into our account and send them to other accounts. The wallet balance is shown here along with a graph and other assets or tokens, as well as details about their pricing, allocations, etc.

On the VaultX application, the Send/Receive modal can be seen in the image in Fig. 5. It will interact with a ThirdWeb to send and receive data and carry out transactions using the provided address. In this app, users are authenticated using a Metamask account, and all transactions made on the application are secured through the use of blockchain technology. Transactions are recorded on a distributed ledger that is maintained by a network of nodes, ensuring decentralization, immutability, and transparency.

5 Conclusion and Future Scope

The development of a Web 3.0-based crypto wallet using Next.js, Sanity.io, and ThirdWeb represents a major step forward in the secure and efficient management of digital assets. With the ability to send and receive Ethers, this wallet provides a convenient and user-friendly way for users to manage their digital assets. The real-time total balance graph provides valuable insights into the current status of the user's connected wallet, making it easier to track and manage their assets. The custom token creation and minting functionality of this wallet is a significant contribution to the world of digital assets. It provides users with the ability to create their own unique digital assets, which can be used for a wide range of purposes, such as fundraising,

Fig. 5 Send/receive modal on dashboard page



reward systems, and loyalty programs. Another area of potential expansion is in the integration of more blockchain networks. Currently, the Web 3.0-based crypto wallet utilizes ThirdWeb to create, deploy, and mint tokens. However, there are many other blockchain networks that could be integrated to provide users with a more diverse set of digital assets. This would require additional development work to ensure that the wallet is capable of handling multiple blockchain networks securely and efficiently. The user experience of the wallet could also be improved by adding more features such as a news feed or market data. This could provide users with the latest news and insights on the cryptocurrency market, which could help them make more informed investment decisions. In addition, the wallet could also offer more advanced portfolio management tools, such as the ability to set up automated trading strategies or access historical data on portfolio performance. Overall, the development of a Web 3.0-based cryptocurrency wallet using Next.js, Sanity.io, and ThirdWeb represents a significant step forward in the management of digital assets. By continuing to expand and improve upon the features and functionality of the wallet, developers can create a platform that is capable of meeting the needs of cryptocurrency users in the Web 3.0 era.

References

1. Petcu A, Pahontu B, Frunzete M, Stoicescu DA (2023) A secure and decentralized authentication mechanism based on Web 3.0 and ethereum blockchain technology. 9 February 2023
2. Ersoy M, Gürfidan R (2022) Blockchain-based asset storage and service mechanism to metaverse universe: metarepo. 07 October 2022
3. Pilkington M (2016) Blockchain technology: principles and applications. In: Research handbook on digital transformations. Edward Elgar Publishing
4. Chatterjee K, Goharshady AK, Pourdamghani A (2019) Probabilistic smart contracts: secure randomness on the blockchain. In: 2019 IEEE international conference on blockchain and cryptocurrency (ICBC). IEEE, pp 403–412
5. Tripathi V, Gangodkar D (2021) Barriers to adopting distributed online attack detection based on blockchain for web application vulnerabilities 18(3)
6. Gupta S, Manjunath CR (2021) Blockchain-based preferential E-voting system DApp using smart contract. April 27, 2021
7. Antonopoulos AM, Wood G (2018) Mastering ethereum: building smart contracts and DApps. Nov 13, 2018
8. Solorio K, Kanna R, Hoover DH (2019) Hands-on smart contract development with solidity and ethereum: from fundamentals to deployment Nov 25
9. Sander F, Semeijn J, Mahr D (2018) The acceptance of blockchain technology in meat traceability and transparency. British Food J
10. Development ESC (2018) Build blockchain-based decentralized applications using solidity. Packt Publishing Ltd., Mukhopadhyay M
11. Abdullah N, Hakansson A, Moradian E (2017) Blockchain based approach to enhance big data authentication in distributed environment. In: 2017 Ninth international conference on ubiquitous and future networks (ICUFN). IEEE, pp 887–892
12. Introducing ethereum and solidity: foundations of cryptocurrency and blockchain programming for beginners. Chris Dannen. March 16, 2017
13. Henry R, Herzberg A, Kate A (2018) Blockchain access privacy: challenges and directions. IEEE Secur Priv 16(4):38–45
14. Saxena R, Arora D, Nagar V (2023) Classifying transactional addresses using supervised learning approaches over ethereum blockchain. Proc. Comput. Sci. 218:2018–2025. <https://doi.org/10.1016/j.procs.2023.01.178>
15. Saxena R, Arora D, Nagar V (2023) Efficient blockchain addresses classification through cascading ensemble learning approach. Int J Electron Secur Digit Forensic 15(2):195–210. <https://doi.org/10.1504/ijesdf.2023.129278>
16. Khatri S, Al-Zahrani FA, Tarique AM, Agrawal A, Kumar R, Khan R (2021) A systematic analysis on blockchain integration with healthcare domain: scope and challenges. IEEE Access 9(84666–84687):10

Orphanage Channelization System Using Blockchain Technology



Tapasya Choudhary, Vanshika Aggarwal, Shivansh Srivastava,
and Shivani Trivedi

Abstract Due to substance abuse or mental illness in the biological home, or the parent may simply be unwilling to care for the child. For such scenarios, various orphanages are running which is a shelter home or group home, built for the purpose of care of orphans and children who, for various reasons, cannot be cared for by their biological families but due to the current system used in such institutions there may take place some kind of malpractices (such as false records, incompetence). Thus, this paper discusses an application to solve the mentioned issues using blockchain technology. Blockchain is a decentralised distributed ledger, which consists of blocks that have their own hash value produced by an algorithm known as SHA-512 or SHA-256 which does not depend on third parties. This will ease information attainment and security by making the system more transparent and immutable. In this paper we used solidity programming language for creating the smart contract on the platform named remix which made available the necessary features in accomplishing the aims of the system.

Keywords Orphanage · Blockchain · SHA-512 · Solidity · Ledger · Blocks · Information

"There can be no keener revelation of a society's soul than the way in which it treats its children."
-Nelson Mandela.

T. Choudhary · V. Aggarwal · S. Srivastava · S. Trivedi (✉)
Department of Computer Science and Engineering, ABESEC, Ghaziabad, India
e-mail: shivani.trivedi@abes.ac.in

T. Choudhary
e-mail: tapasya.21b0101015@abes.ac.in

V. Aggarwal
e-mail: vanshika.21b0101006@abes.ac.in

S. Srivastava
e-mail: shivansh.21b0101045@abes.ac.in

1 Introduction

The most valuable resource in the country is seen as being children. They are one of the society's most vulnerable groups, though, at the same time. There are many children who are more vulnerable than others, and these kids are known as the Children by Need of Care and Protection. In the current script used in orphanages to store the information physical mode is used, in which documents, lines, etc. are stored physically or in the form of papers which can put quite an issue to find any particular information from that pile of lines kept in closets over the times. Also, due to similar condition and no verification false records can also be created or there perhaps loss of important information or some malpractices taking place similar as not furnishing the installations which are claimed by the separate association or numerous further, so to break similar issues we can use blockchain technology which can also increase the effectiveness of the said orphanage. Blockchain can be said as a public tally, in which all married deals are stored in blocks and are connected in a form of a chain. It's a system of recording information in chronological order with the use of cryptography mincing in an inflexible distributed and decentralised tally across the network of computer systems. This chain of blocks continuously grows and when new data (in the form of blocks) is added to it. The blockchain technology has the crucial characteristics, similar as decentralisation, persistence, inflexibility, and translucency. Blockchain can work in a decentralised terrain, which is ensured by combining several core technologies such as cryptographic hash, digital hand, and distributed agreement medium.

An exploration has been done on the current script, presently operated system and the problems and conditions are being conceded. Likewise, an orphanage channelization system is designed to resolve the vulnerability of the current operation system. The system handed for this system is Structured System Analysis and Design. The Orphanage Channelization System is designed substantially for the enhancement of the working of the orphanage and to make it more transparent. The stoner of the system is the director or the director and staff, children, and the parents. The enrolment process of the orphans is managed by the director. Everyone will have the access to information at any time and also can modernise their separate records. With the help of blockchain, indeed after stoner makes the update the former information will also be recorded in the system as a block once created can't be destroyed which will make the system more dependable in case of any false record or mishandling in the information. There will be a unique user id for every user and in case of new enrolments, new stoner id will be assigned using which any stoner can login, view information or make the changes making the system more transparent.

A. Motivation

The basic motivation behind this project was to improve the working of the orphanage and make it more transparent using the blockchain technology. It will help the children to find the rightful parents and also to check all the facilities and faculty information using the technology to not only improve the functioning internally but also externally. Seeing today's scenario people are using platforms like orphanage for their

own good and for the mere purpose of money making, neglecting the main motive of the organisation or negotiating with the basic needs of the children and to eliminate that we can create a system to maintain all the information about the faculty people who wants to adopt the children, funding or charitable trusts working with the orphanage and facilities in that organisation to make the system of the orphanage more transparent and efficient.

B. Research Objective

Many orphans suffer abuse physically, sociological, and psychological problems including malnutrition, absence of education, and assault. Only 5% are able to complete a minimal level of education. So to make sure that there are no malpractices performed in the orphanage and all the basic facilities are provided to the children this project will be worked on and acquiring all the information regarding the staff, people who want to connect with the orphanage to adopt, leave the child or to give funding. The main objective of this project is to provide a better environment to the children residing in orphanages and eliminate the malpractices if being practised in such organisations.

C. Problem Statement

In India, orphanages are strictly developed to make sure that orphans are safe, reinstated, and reimposed. These orphanages have no registered location and thus can be found anywhere in India, from small towns to urban areas such as the national capital. According to India Times, the majority of orphanages which are run privately or by public organisations, NGO are still not registered under Section 34(3) of the orphans Justice (Care and Protection of Children) Act, 2006. These orphans, on the other hand, are forced to live in harsh conditions and are not fairly treated in orphanages. As the few incidents are listed below:

- A case was reported in which a HIV positive kid was a victim of harassment by a guard and other people at the orphanage Ashiana Home for Boys in Alipur, Delhi.
- An almost similar incident were recorded in Gurgaon, Haryana, where the NGO's owner, Suparna Sethi, and a boy who was working there as the caretaker named Rachit Raju allegedly raped a minor girl child. As the orphanage wasn't registered under government or in any papers, both of them were arrested and taken into custody for violating the orphans Justice Act of 2000.
- Another incident was reported in which a girl child of a public orphanage Bal Kunj in Yamunanagar went missing. According to accounts, people connected with the orphanage itself sexually and mentally harassed the orphans.

Usually, they seemed to be the best suitable places for orphans to be reinstated and saved into society. Anyway the truth is much different. These locations are marked by deteriorating facilities which they claim to provide but fails from by a large scope and filthy sanitation. The food as well as clothing offered is of bad quality. The lack of basic resources such as bathrooms and showers is shocking. The atmosphere needs an improvement' a person has been lost due to the staff's poor understanding of detention.

The difference that we see in our jurisdiction and rules on paper and their execution in society is a major issue in India when it comes to child safety and justice and fair treatment for the orphans. The system is blocked by administrative problems such as an inefficient system of maintaining recent data and routine reporting.

2 Blockchain

Blockchain stores data in the blocks which are linked together to form a chain. When a transaction takes place between two parties the details of the transaction is broadcasted in the network. In blockchain each participant has the copy of the ledger and once a transaction is approved and validated by all the participants, then transaction details are added to the ledger and each participant gets the updated ledger with newly added transaction details. There are various types of blockchains:

- Private Blockchain: permissioned blockchains. In such blockchains only a single authority looks after the network and participants who want to participate in a transaction should get permission for the respective access. E.g.: Hyperledger and R3 Corda.
- Public Blockchain: Permissionless blockchains. They are open for all, anyone with an Internet connection can send transactions and can become a validator. E.g.: Ethereum.
- Hybrid Blockchain: This is a combination of private and public blockchain. E.g.: Dragonchain.
- Consortium or Federated: Permissioned blockchains. Group of organisations come together and set up their own blockchain network for cross-company and cross-discipline solutions. E.g.: Quorum.
- Sidechains: A blockchain that runs parallel to a primary blockchain is called sidechain. In this, entries from a primary blockchain can be linked to a sidechain and vice-versa.

3 Literature Survey

Goldenfein and Hunter (2017) and his team proposed a method of Blockchains, Orphan Works, and the Public sphere by source Columbia journal in 2017 stated that Orphan work-shop are workshop which are still presumably defended by brand, but for which a rightsholder cannot be set up. They represent a significant problem for the effective dispersion of knowledge, since implicit druggies of the workshop cannot certify their use, and as a result may choose not to use them for fear of a possible violation action if the proprietor does latterly crop. The first element of our offer is a blockchain where every hunt for a work's proprietor can be recorded [15].

Ewald et al. (2019) and his team proposed a method of Digitalised Orphanage Home Management System con-forming Of Massdata Entries. Kuupole Erubaar

Ewald 1, ZengLiaoyuan 1, Hassan Sani Abubakar Coker Kenneth 1 from source university of electronic wisdom and technology in 2019 stated that Orphanage Home Management System (OHMS) is developed for orphanages to achieve orphan's empowerment and conservation. It's a model which can only be accessed on a single computer. The enrolment procedure of the orphans is managed by the administrator while the system director is only involved in managing the staff record and preservation. Only the director has the authority and control to do the system conservation similar to backup and recovery if there's system failure [10].

Kasturi et al. (2020) and his team proposed a method of Orphanage Management using data mining techniques by source Alochana Chakra Journal in 2020. An Information System (IS)(4)(6) is the set of software, data, people, and procedures that work together to produce information. Information is a genuinely valuable and expensive asset that needs to be protected, controlled, and planned for, just like other priceless things with a similar link to plutocrats, infrastructure, and people. An association must have a dominant role in the business for its information system to help it accomplish this [5].

Kumar and Kumar (2020) and his team proposed a method of Orphanage Helping System (Santhosh kumar k, Ashish kumar) by source Asian Research Association in 2020 stated that an orphanage is a domestic home built for the upbringing of juveniles when parents are unwilling or incapable of caring for them. Because of urbanisation and industrialization, the orphan problem is severe requirements for food, plutocrats, clothes, and medications are all arranged differently in each orphanage. The primary goal of this initiative is to create a concentrated location for orphanages [4].

4 Proposed System

This project aims at collecting the data regarding the facilities provided, staff working in that orphanage, funding provided and the finances, people who want to connect with that orphanage to adopt the children and the people who want to leave the children in that orphanage. Furthermore, the collected dataset will be stored in the form of blocks which will be linked together in the form of a chain called blockchain. In this project a private blockchain is being mined whose access will be given only to the parents (adopting the children), administrator and the staff members.

All the members of the blockchain will be given their user id and password which they can use to login or to access the blockchain as shown in Fig. 1. Main access of making changes into the content of blockchain will be given only to the administrator. He will store all the information regarding the children in that blockchain. Also the information regarding parents (adopting the children) including their documents (after verification) will be stored in the blockchain which can be seen by everyone so that there is no or less chances of false records by parents. This will help in protection of the orphans from any kind of mishappening such as adopting and selling of orphans or using orphans for satisfying their needs.

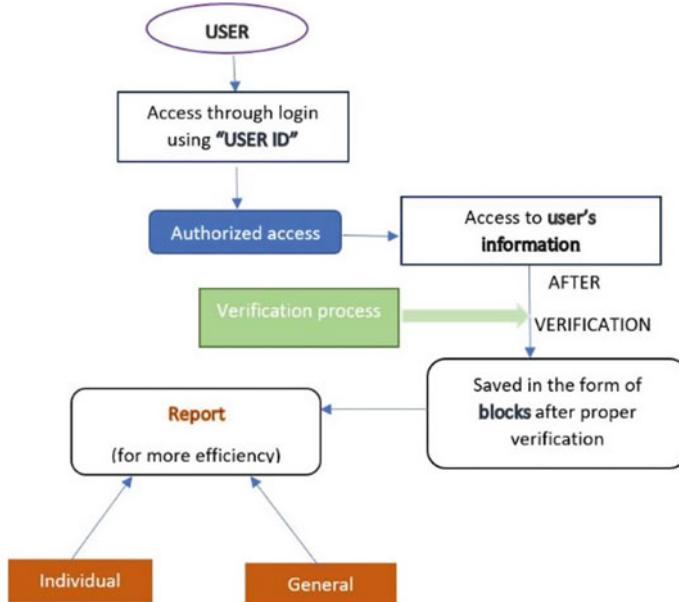


Fig. 1 Proposed system flow diagram

Since blockchain is a decentralised and transparent technology hence the records of the orphans can be seen by everyone having the access of the private blockchain. So that neither the administration nor parents or anyone can use the orphans for satisfying their deeds.

A. Verification Process

Parents (adopting the children) will submit the document. These documents will be verified from the official sites after proper verification when the documents will be found correct then only the orphan (they want to adopt) will be given to them if the documents would be found incorrect or there will be anything wrong in documents found by the administrator then parents will be declined to access the orphans. Also after the verification process is completed then only the record will be added in the blockchain and the block will be mined in the private blockchain. This system will ensure proper safety of the orphans hence no one can take advantage of the orphans.

B. System Model

Figure 1 shows the system design of this model. In the system design of this model, the user will be granted the access using the user id and authentication process. After the access the information of the user will be verified using the verification process. The verified information will be stored in the form of blocks in the system. This information will be there in the system even after making changes in the data. By maintaining the information, we can analyse the condition and operation of the organisation by creating the monthly report.

5 Results and Discussion

For accessing this module, the first phase will consist of the basic information or the basic idea for which this web page is being designed and then on moving forward there are some of the portals through which the members can access the blockchain using their user id and passwords.

Figure 2 shows the opening page of the website on opening the page this site opens. Umbrella is the name given to the website. There is a little information about the website or the basic idea of making this website. Furthermore, for more information on the same one can click on the button “learn more.” It will provide more information upon the same.

On moving further, Fig. 3 shows the various other information which a user can access. It consists of various options like about help and contact information of the organisation which a user can use at the time of the need or for any information.

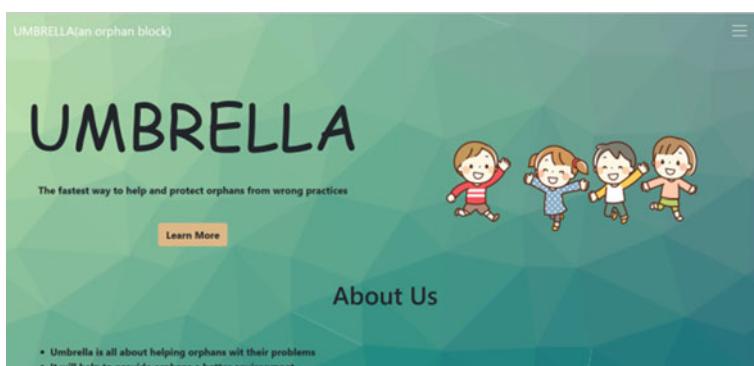


Fig. 2 First phase of module

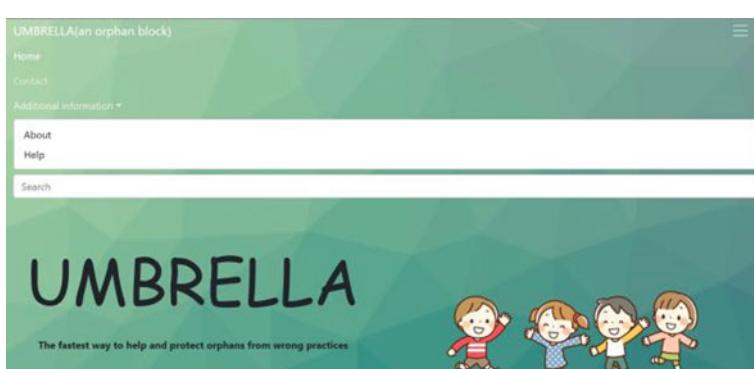


Fig. 3 Second phase of module

On moving further, Fig. 4 shows the accessing portals for various users according to their respective roles. There are 3 login portals for parents (adopting the children), for staff members (in orphanages), and for the administrator. On clicking on the portal it will lead the user to the login pages where on entering the user id and password user can access the blockchain where all the information about the child is being stored.

Figure 5 shows the access portal on clicking on the portals made on the website it will lead to the following login portal every user will be given their own user id and password through which they can login. On clicking on the login button one will have access to the data of the blockchain hence they can see all the data about the children living in the orphanage. Similar login portals are being made for the staff and administrator login as well. This portal is for the parents' login.

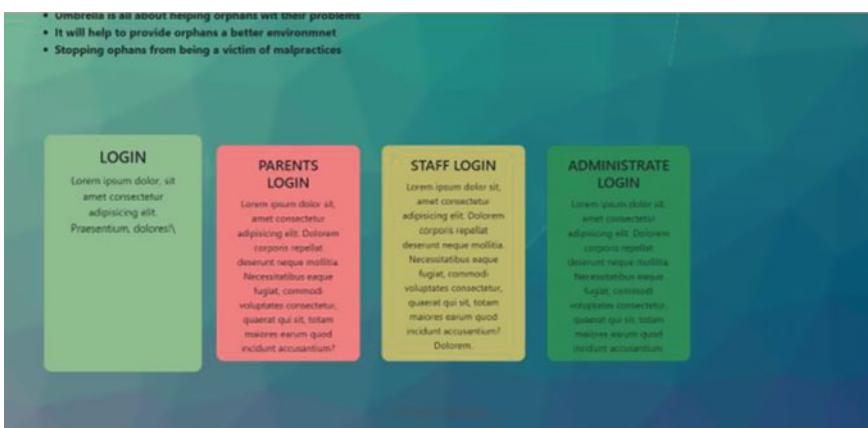


Fig. 4 Third phase of module

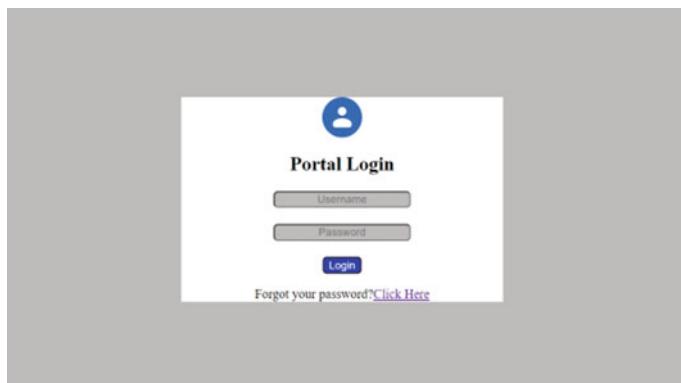


Fig. 5 Login portal

	Name	Age	Adopted/Not Adopted	Any Allergies	Education
▶	Anant	5	No	Lactose intolerant	kindergarten
	Archit	14	No	No	9th Grade
	Hritik	3	No	No	-
*	Lavnya	8	Yes	Anaphylaxis	3rd Grade
*	NULL	NULL	NULL	NULL	NULL

Fig. 6 Dataset used

```
Genesis Block
[{'name': 'Alice', 'Age': '5', 'Adopted/Not': 'no', 'education': 'kindergarten'},
 {'name': 'BOB', 'Age': '10', 'Adopted/Not': 'no', 'education': '5th grade'},
 {'name': 'charlie', 'Age': '15', 'Adopted/Not': 'no', 'education': '7th grade'}]
> []
```

Fig. 7 Blockchain

Figure 6 shows the dataset that is being used in the project. It consists of the basic information of the orphans which is then stored in the form of the blocks to create a blockchain.

Figure 7 shows the information of the dataset is stored in the form of the blocks that are connected together. With the course of time as new information will be added the information will be stored in blocks likewise forming a blockchain. Genesis block is the first block or the initial block of the blockchain.

6 Conclusions

Every child has their rights for the basic necessities which are essential to lead a better life. Not just those who experienced violence, abuse, and malpractice because of their difficult circumstances, but also those who are not only to protect their social security, they nevertheless need to be defended in dire circumstances. Since child protection is a fundamental need, it is connected to all other children's rights. The overall development of a child and their entire being are impacted when children's rights are not upheld. Due to all of this, there is an urgent need for the recognition that children have the right to develop in an environment that is stable and safe, to

be protected from abuse and neglect, and to have their basic needs met. It must take into account factors other than just the children. Therefore, there is a critical need to bring such a system in use which can minimise the deceiving practices and help the children in need and through this system the working of the association as well as other people related to the child will come more transparent and no one can tamper with the administration for their own benefit. By using blockchain technology in this design we can maximise the effectiveness of our current system by multitudinous crowds as using the decentralised nature of the technology the system will be more reliable and it will be easy to pierce and store the database.

7 Future Scope

Through this project we can use blockchain technology for the social good which enhances its value in today's world where people are becoming more money minded and neglecting their duties towards the society and those who are in need. We can put this project in use for achieving that goal and it can be used by organisations who are working towards such a goal and making the system more transparent by providing the information to those who are concerned with the respective organisation.

Acknowledgements First and foremost, we want to express our gratitude to everyone who helped us complete this project successfully and bring about a wonderful outcome. We also like to express our deep appreciation to Ms. Shivani Trivedi for all of her help, advice, and support. Her leadership enabled us to guide our project and do study and analysis for a better project result. It was a privilege to work and learn under her direction. Shivansh Srivastava made a contribution to the creation of the blocks by storing the gathered datasets as blocks, which were then joined to form the blockchain. Tapasya Choudhary completed the final formatting and editing of the website design in addition to gathering and preprocessing the dataset. The model's design and the different login portals, which can be accessed by anyone, were created by Vanshika Aggarwal. We would want to express our gratitude to our parents and friends for their support, encouragement, and assistance in helping us model this research in the current framework.

References

1. Ho LS, Zhang T, Kwok TCT, Wat KP, Lai FT, Li S (2022) Financing orphan drugs through a blockchain-supported insurance model. *Front Blockchain* 5:7
2. Tojalievich TI, Mavlonjonovich MM, Tokhtasinovich UJ, Eraliyevich TA (2022) Advantages of client-server architecture for electronic document management systems. *Web Sci: Int Sci Res J* 3(6):1657–1660
3. Josphineela R, Vigneshwar S, Dinesh S, Kumar A (2021) *Int Res J Eng Technol (IRJET)* 08(04) Apr 2021
4. Kumar KS, Kumar A (2020) Orphanage helping system. *Asian Res Assoc*
5. Kasturi K, Nisha M, Vistas C, Nadu T, Nadu T (2020) Android application for orphanage management using data mining techniques. *Alochana Chakra J* 9(4):340–346

6. Bodkhe U, Tanwar S, Parekh K, Khanpara P, Tyagi S, Kumar N, Alazab M (2020) Blockchain for industry 4.0: a comprehensive review. *IEEE Access* 8
7. Ewald KE, Liaoyuan Z, Abubakar HS, Kenneth C, Gyamfi EK, Esther S (2020) DIGITALIZED ORPHANAGE HOME MANAGEMENT SYSTEM CONSIST-ING OF MASS DATA ENTRIES
8. Yaga D, Mell P, Roby N, Scarfon K (2019) *Nat Inst Stand Technol Int Rep* 820 (October 2019)
9. Chauvette A, Schick-Makaroff K, Molzahn AE (2019) Open data in qualitative research. *Int J Qual Methods* 18:1609406918823863
10. Ewald KE, Liaoyuan Z, Abubakar HS (2019) DIGITALIZED ORPHANAGE HOME MANAGEMENT SYSTEM CONSISTING OF MASS DATA ENTRIES. University of Electronic Science and Technology of China 2019
11. Naaz S, Meenai Z (2019) Alternative care in India: issues and prospects. *Rajagiri J Soc Develop* 11(1):3–18
12. Tresise A, Goldenfein J, Hunter D (2018) What blockchain can and can't do for copyright
13. Zheng Z, Xie S, Dai HN, Chen X, Wang H (2018) Blockchain challenges and opportunities: a survey. *Int J Web Grid Serv* 14(4):352–375
14. Staples M et al (2017) Risks and opportunities for systems using blockchain and smart contracts. *DATA61 (CSIRO)* (May 2017)
15. Goldenfein J, Hunter D (2017) Blockchains, orphan works, and the public domain. *Colum JL Arts* 41:1
16. Quansah E, Ohene LA, Norman L, Mireku MO, Karikari TK (2016) Social factors influencing child health in Ghana. *PLoS ONE* 11(1):e0145401
17. Naren J, Nikilav PV, Kate AV, Mohan A (2015) Deduction of orphans and suitable parent candidates using statistical modelling. *Int J Appl Eng Res* 10(22):2846–42850
18. Askeland L (2006) ed *Children and youth in adoption, orphanages, and foster care: a historical handbook and guide*. Greenwood Publishing Group
19. Reef C (2005) *Alone in the world: orphans and orphanages in America*. Houghton Mifflin Harcourt
20. Johnston GP, Bowen DV (2005) The benefits of electronic records management systems: a general review of published and some unpublished cases. *Rec Manag J* 15(3):131–140
21. McKenzie RB (1998) Rethinking orphanages for the 21st century: a search for reforms for the nation's child-welfare. *Spectr J State Gover* 71(2):8–12
22. Lucas HC Jr (1975) Performance and the use of an information system. *Manage Sci* 21(8):908–919

Blockchain-Based Public Distribution System



Dipti Pawade, Chaitanya Bandiwdekar, Pooja Kaulgud, Siddhesh Bagwe, and Aditi Kulkarni

Abstract The Indian government manages the Public Distribution System (PDS) to provide the ration among the poorest sections of the society. Currently, this system is facing numerous problems such as poor supply chain management, corruption, and non-transparency. The various reports bring to notice that many times ration is diverted, illegally sold, or provided to ineligible beneficiaries. As a result, millions of entitled individuals do not receive their rations. The existing PDS system involves a complicated network of intermediates such as suppliers, dealers, fair-price shops, and the citizens. This intricate system makes it challenging to monitor the distribution of ration from fair-price shops to the recipients, which creates opportunities for fraud. Therefore, a more efficient and transparent system is necessary to ensure the proper distribution of ration to intended beneficiaries. In this paper, we have discussed the use of blockchain technology to provide a transparent, decentralized, and tamper-proof solution for managing the public distribution system effectively.

Keywords Blockchain · Ethereum · Public distribution system (PDS) · Ration · Fraud detection

D. Pawade (✉) · C. Bandiwdekar · P. Kaulgud · S. Bagwe · A. Kulkarni
Department of Information Technology, K. J. Somaiya College of Engineering, Vidyavihar,
Mumbai, India
e-mail: diptipawade@gmail.com

C. Bandiwdekar
e-mail: c.bandiwdekar@somaiya.edu

P. Kaulgud
e-mail: pooja.kaulgud@somaiya.edu

S. Bagwe
e-mail: siddhesh.ab@somaiya.edu

A. Kulkarni
e-mail: aditi.hk@somaiya.edu

1 Introduction

PDS was launched in India in 1947. It helps in distribution of food and non-food items to India's poor at subsidized rates. Under PDS, the different types of ration cards are available based on the per annum income of the family. Depending upon the type of ration card the particular family is holding, essential groceries like rice, wheat, pulses, sugar, oil, etc. are given to them. Table 1 gives a brief overview of the type of ration cards and associated benefits. The government has various schemes such Antyodaya Anna Yojana (AAY), Annapoorna Yojana so that needy people can get enough groceries for their survival. This is an initiative to ensure the food security of the citizens [1].

The state government provides rations to the fair-pricing shop (FPS) at the beginning of each month. Within 10 km of the user's geolocation, there is a fair-pricing shop available. So for each family, a neighborhood has three to four fair-price shops. Previously, the user was required to pick up the ration from the designated store directly. Because of this the citizens were greatly inconvenienced. For example, if for some reason, the family is temporarily relocated to another area then for collecting the ration, they would need to travel to the specific FPS where they are registered.

Table 1 Types of ration cards

Ration Card Type	Income limit and other Criteria for awarding the ration card	Grain criteria per person
Yellow ration card	<ul style="list-style-type: none"> • Families having annual income up to Rs.15,000/- (urban Area) • None of the members in the family should be a doctor, lawyer, architect or chartered accountant • None of the members in the family should be eligible to pay tax • The family should not possess a residential telephone • The family should not possess a four-wheeler vehicle • All the persons in the family should not hold total two-hectare rain fed or one hectare semi-irrigated or 1/2 hectare irrigated (double in drought-affected talukas) land [2] 	Each household is entitled to 35 kg of food grains per month per family. (includes sugar, wheat and rice)
Saffron ration card	<ul style="list-style-type: none"> • A family with annual revenue of Rs 15,000 to Rs 1 lakh • None of the family members should own a four-wheeled vehicle (excluding taxi) • The family in all should possess four hectare or more irrigated land 	5 kg ration per person (includes wheat for Rs 2/Kg and rice for Rs 3/Kg)
White ration card	<ul style="list-style-type: none"> • Family having income above Rs 1 Lakh • Any member of the family possessing a four-wheeler • The family aggregately holding more than 4 hectare irrigated land 	Such ration cardholders do not get any ration from the government

Additionally, in PDS there were relatively very few online transactions between the government and the fair-price shop. As there was no actual track on the ration received by FPS from the government and the amount that was distributed to the users, this would result in discrepancy. But today, every fair-price shop has a biometric scanner. Biometric scanners enable users to take their ration from any fair-price retailer within 10 km of their geolocation. The scanners also aid in determining whether or not the user has previously taken the ration. The scanners assist the government in determining how much ration has been supplied to the masses by the FPS. As earlier the FPS owner tried to sell the ration received from the government at substantially higher costs in the market. The biometric scanners have increased some of the system's transparency. But still the challenges with respect to the actual stock with the FPS owner and cases of denial to provide the ration to the beneficiaries still exist. Such cases are widely covered by the press media and news TV channels [3–6].

Our system enables authorities (the government for example) to track the proper working of the PDS. With the usage of Ethereum and blockchain technology, data can be viewed by the authority in the distribution chain and enables them to track the transactions and commodities at any time. The transactions made by the consumer and distributor would not be alterable hence making the scope of frauds, malpractices, and thefts to be minimal. Thus, rogue activities of vendors can be easily tracked by the authorities at any given time and instance easily. Further in this paper, in Sect. 2, literature survey is discussed. Section 3 discusses the methodology in detail. In the Sect. 4, results are discussed followed by conclusion in Sect. 5.

2 Literature Survey

Anni et al. [6] proposed an android application which generates tokens for collecting the ration from shops and displays information such as availability of ration, cost of commodity, and its quantity to the users. Major limitation of this system is the unavailability of smartphones with people below poverty level, and people should be well educated and comfortable using the app. Also, this application may prone to malware hazards. Shukla et al. [7] proposed the idea of a radio frequency identification (RFID)-based smart ration card which can be verified at the fair-price shop for the authentication of the user. A microcontroller that is linked to the database will confirm the user's identification. The major concern about this approach is the huge cost associated with the hardware required. Also, the regular ration cards need to be replaced with the RFID-based card which is money, time as well as man power consuming task. Vaisakh et al. [8] tried to emulate IoT-based ration distribution system just like an ATM, where the person's identity is confirmed using fingerprint and through LCD one can choose commodity and its quantity, and accordingly it will come out from machine. Rajesekaran et al. [9] proposed smart measuring automated electronic devices with the aid of Arduino microcontrollers to replace the manual work done in the distribution centers. These devices scan the ration card, accurately measure the goods, and update a database periodically about the availability of the

goods as well as information about transactions that were carried out in a digitalized manner. Aishwarya et al. [10] also suggested the use of a RFID tags, which store all the necessary data of ration cards. Every customer is notified to receive their rations by Global System for Mobile (GSM) when supplies are delivered to the shops by the government. The ARM7 controller is used to manage the entire system. It also has features like customer identification, tampering, and fire detection [10]. Goradia et al. [11] put forth an embedded system where the customer has to input the amount of commodity they require and the system made will automatically collect that much amount in a container. Shah et al. [12] suggested another microcontroller and RFID-based automated ration distribution system. The system uses hardware components such as Atmega32 microcontrollers, RFIDs, centralized databases, LCDs, etc. to reduce other human interventions. Padmavathi et al. [13] proposed an automated ration distribution system which is similar to an ATM and will be open 24/7. The system reduces the human intervention in fair-price shops thus automating the process making it reliable. Rajalakshmi et al. [14] introduced an android application for ration distribution system. They have also used MongoDB and a restful API to provide secure connection to the database. Shwetha et al. combined the strength of IoT with Ethereum-based DApp development [15]. Madaan et al. [16] also proposed the similar IoT and blockchain-based ration distribution system. Through the literature survey summarized in Table 2, it is observed that whenever there is an involvement of any hardware-based setup, the additional setup cost and recurrent maintenance cost are the major issues. So, we tried to explore the strength of using blockchain in supply chain management in the existing system itself.

3 Implementation Overview

Blockchain is a distributed immutable ledger which is mainly introduced to store transactions of financial applications. It is a decentralized data structure to store data in the form of blocks, where each block contains a cryptographic hash of the previous block, a timestamp, and the transaction data [2]. Blockchain-based systems combine cryptography, public key infrastructure, and economic modeling to accomplish distributed database synchronization through peer-to-peer networking and decentralized consensus [17, 18]. It is a breakthrough technology that promises significant improvements in the way global technology operates. Hence, we tried to extend the strength of blockchain technology in the PDS.

Figure 1 gives the overview of the system consisting of three main layers. The access layer is responsible for registration and login facility for stakeholders, registering the complaint (by consumer), toggling the blacklisted vendor (by authorities). Database layer is responsible for maintaining collections like Allowance Collection, Complaint Collection, Consumer Collection, Transaction Collection, and Vendor Collection. Data from the database layer is pushed to the blockchain layer which is responsible for transaction management, vendor management, consumer management, and stock management. Referring to Fig. 2, the physical supply chain starts

Table 2 Literature summary

Ref No	Year	Methodology	Limitation
[14]	2022	MongoDB, android app	–
[16]	2022	IoT, blockchain	Additional implementation cost of IoT-based system
[6]	2021	Android application	Unavailability of smart phone, application may prone to malware hazards
[12]	2020	Atmega32, RFIDs	The system requires proper maintenance
[15]	2019	IoT, blockchain	Additional implementation cost of IoT-based system
[8]	2019	Fingerprint verification and IoT	System installation cost is higher. The system is not tested for mass uses. Failure in the system may cause a halt in work
[7]	2018	RFID and IoT	Changing of current ration cards to smart ration card required additional budget, time, and manpower The users and ration shops may not have sufficient technology to set up the proposed IoT-based environment
[9]	2017	Ration card scanner, IoT, Auto-measuring facility	Implementation cost is high
[10]	2017	RFID, ARM7	Has additional hardware cost and the system is not tested for mass usage
[13]	2017	ARM 7 LPC2148, RFID, fingerprint module	The hardware used in this proposed system requires timely maintenance. Also, the setup cost is high. The fingerprint sensors may not work in some conditions due to noise and distortion brought on by dirt and twists
[11]	2015	Embedded system	The power supply plays a key role in this system, failure in power supply can halt all the process

from the authority and goes to the vendor and then to the consumer. This allotment of ration is noted in the database as the stock of the vendor. The stock of the vendor is refilled by the authority, and each consumer gets a fixed allowance to request from the vendor. All three of the entities access the application with their unique ID and password which is stored in the firestore database. The cumulative of all transactions for a particular vendor is sent to the blockchain via a smart contract. The web application accesses the database via firebase functions in React JS, while it accesses the blockchain via smart contracts written in solidity. The Smart Contract takes in transaction data stored in firebase along with the exact timestamp of the transaction and creates a block containing all the data fetched. This block will be public and accessible to all so that each transaction can be easily totaled up to tally the received stock at the beginning of a month. The vendor is now unable to tamper or manipulate the data present in each block in the blockchain.

Figure 3 depicts the consumer side flow of the application. The consumer logs in to the system using their credentials. The authentication is done by referring to the consumer database where the hashes of password are compared and the consumer

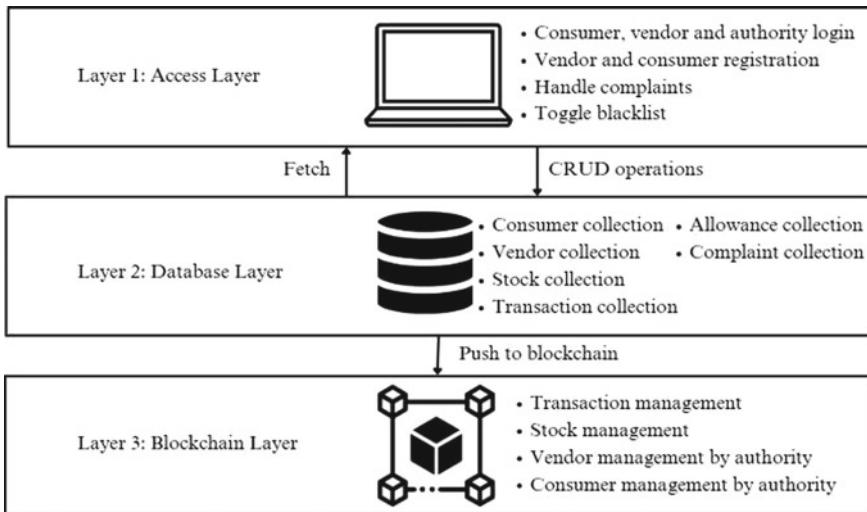


Fig. 1 System overview

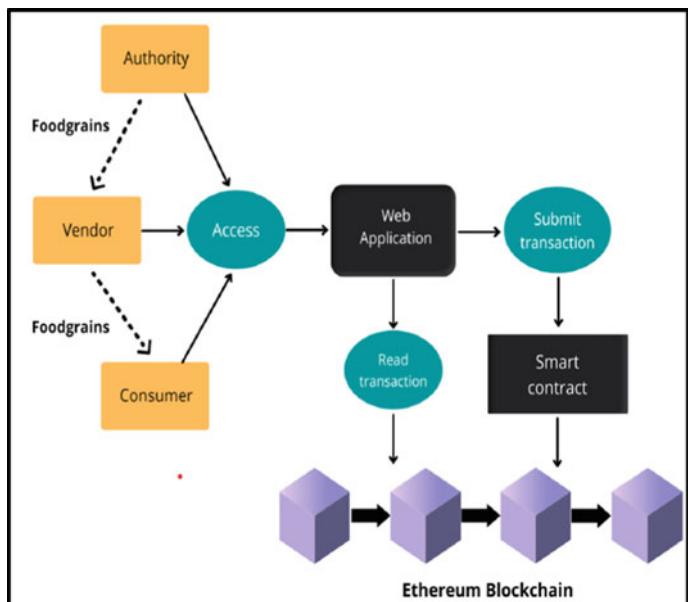


Fig. 2 Application flow

gains access to the application. Here consumers can view their transaction histories showing all the previous transactions with their timestamp. The current allowance that the consumer can avail from their particular vendor as per their ration card type is fetched from the database. Lastly, the consumers can lodge/file a complaint against their respective vendors which are stored and seen by the authority. In Fig. 4, the vendor flow is explained. Vendor logs in using their credentials, and once the vendor is authenticated, he/she is directed to the vendor home page. Here the vendor can view his/her current stock and list of all the customers under his/her zone. Vendors can update allowance for a specific customer based on what the consumer requests adhering to the allotted limit. Once the vendor's stock is over, all the transactions will be pushed to the blockchain.

In Fig. 5, the flow of the application for the authority is shown. The authority first has to login into the application by entering the password on the login page. After login, the user will be redirected to the authority home page where a number of functions can be performed. Consumer and vendor details are displayed in a list on the authority home page along with buttons for the functions that can be performed. The authority can perform the following functions: Add consumer, Add vendor, Refill consumer allowance, Refill vendor stock, View consumer complaints, Toggle blacklist status of the vendor, and Update consumer's card type.

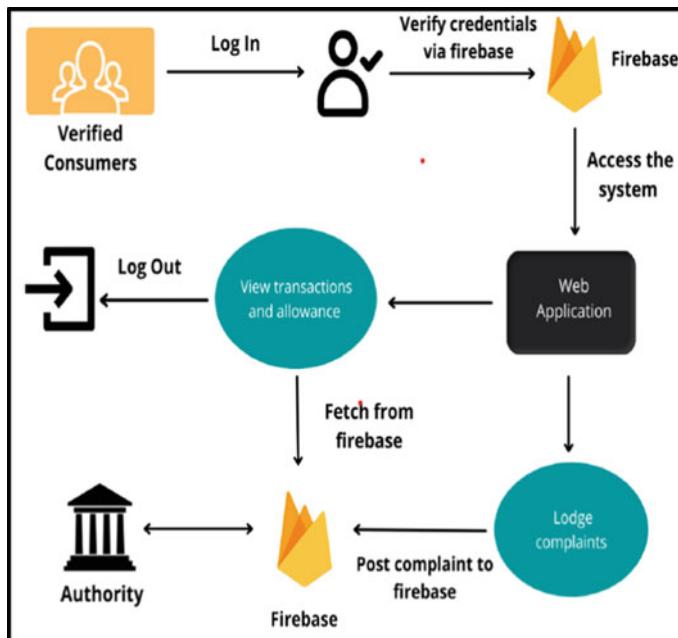


Fig. 3 Consumer flow

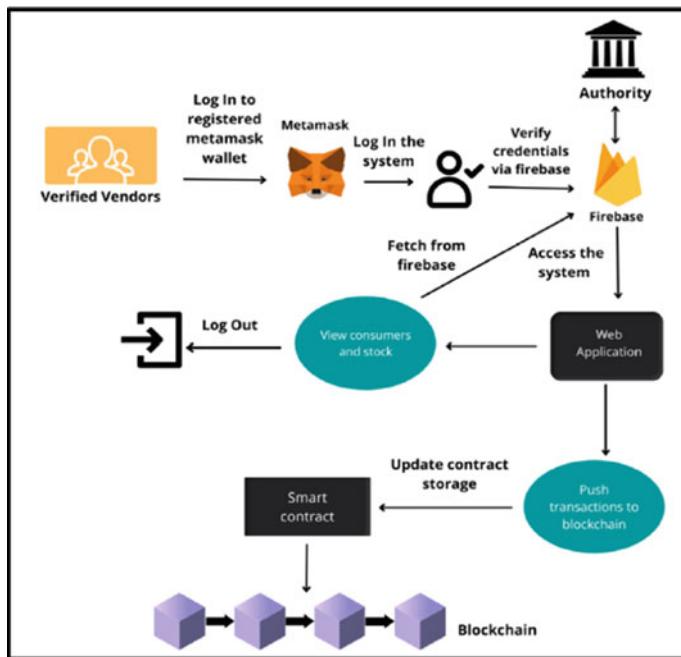


Fig. 4 Vendor flow

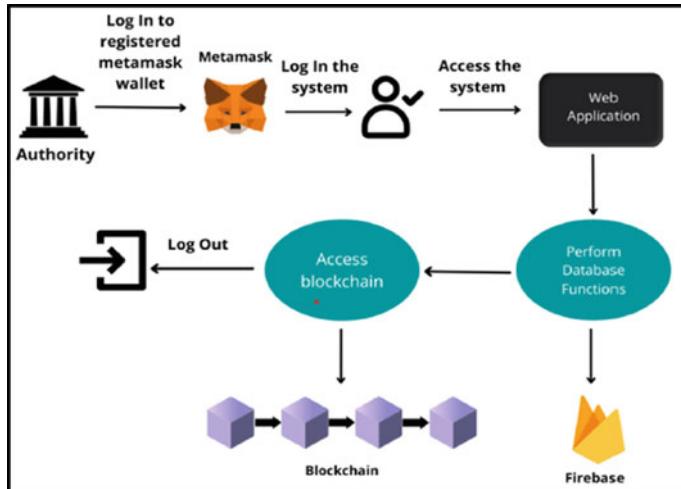


Fig. 5 Authority flow

4 Results and Discussion

As discussed in the previous section, vendors can make a transaction with a consumer and can push multiple transactions to the blockchain at month end. Vendors can also view his/her stock as well the consumers that come under the same zone. Similarly, consumers can view their monthly allowance and also launch a complaint. On the other hand, authority can add consumer and vendor information as well as authority can refill stock and allowance (for single entity as well as all entities). At the same time, authority can also blacklist a vendor if necessary and can update the ration card type of a consumer. Whenever a vendor provides ration to the receiver, a set amount of grain count is deducted from the vendor account and added to the receiver account indicating a successful transfer. This information along with additional information like timestamp is included in a transaction list maintained for each vendor. A block is created in the local blockchain at the end of a predefined time period (here a month), where the vendor pushes all the transactions within the transaction list carried out in the time frame. Each transaction consists of the receiver information and the amount of grains received, along with other information like timestamp.

Figure 6 shows the typical structure of each block created by our application. It contains two parameters of data: a transaction list and vendor id. The transaction list structure is an array containing individual transactions that the vendor had with each consumer along with ration quantities and timestamp of the transaction. The system has been designed to achieve the following goals:

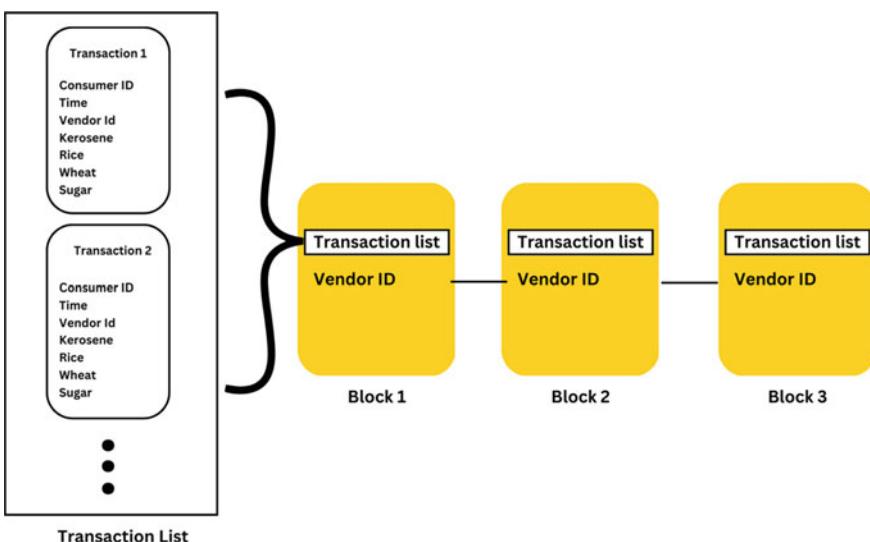


Fig. 6 Block creation diagram

CURRENT BLOCK GAS PRICE 279	GAS LIMIT 6721975	HARDFORK MUIRGLACIER	NETWORK ID 5777	RPC SERVER HTTP://127.0.0.1:8545	MINING STATUS AUTOMINING	WORKSPACE SANAAJ		SWITCH
BLOCK 279	MINED ON 2022-04-28 20:21:15			GAS USED 298171				
BLOCK 278	MINED ON 2022-04-28 20:08:58			GAS USED 27266				
BLOCK 277	MINED ON 2022-04-28 20:08:57			GAS USED 1750738				
BLOCK 276	MINED ON 2022-04-28 20:08:55			GAS USED 42266				
BLOCK 275	MINED ON 2022-04-28 20:08:55			GAS USED 128970				

Fig. 7 Blocks on blockchain

- To perform accurate transactions between the consumer and distributor which would not be alterable hence making the scope of frauds, malpractices, and thefts to be minimal. Thus, rogue activities of dealers can be easily tracked by the authorities at any given time and instance easily.
- The authorities (the central government for example) can use the system to track the proper working of the PDS which reduces the chances of fraud significantly.
- With the usage of Ethereum and blockchain technology, data can be viewed by anyone in the distribution chain and enables the authorities at all levels, especially the central government to track the transactions/money and commodities at any time along the distribution chain.
- The developed system can handle the consumers of all ration types and their card updation efficiently. Based on the ration card types, the ration allotment is also done and also provides them with the right to complaint.
- At the end of each month, all of the transactions made by each vendor would be sent to the blockchain, using its immutability to keep track of individual vendors, hence helping to keep frauds and makeshifts in check.

Figure 7 shows the latest blocks added to the blockchain. Every transaction consumes ether as gas fees depending on the amount of information stored. It takes approximately 0.005 eth to store the list of transactions carried out by the vendor.

Figure 8 shows one example transaction added to the blockchain network. The function pushTransaction pushes all the pending transactions at the vendor's side to the blockchain network along with the vendor id.

The contents of a created block are in the form of key-value pairs which is shown in the Table 3.

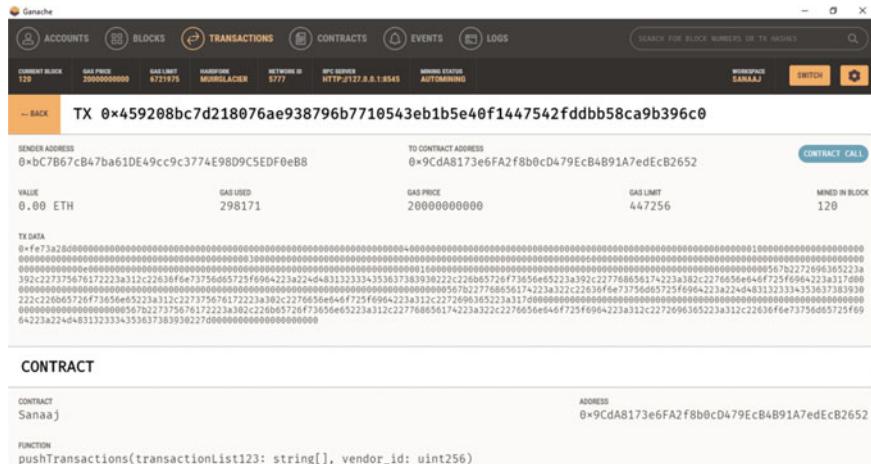


Fig. 8 Transaction performed using smart contract

Table 3 Contents of created block

Parameter	Value
Contract name	Sanaaj
Address	0x587BfbFF6Ed8DB0d7F6b46fCFd7e7ed9fA0575Dd
Function called	pushTransactions(transactionList123: string[], vendor_id: uint256)
To contract address	0x587BfbFF6Ed8DB0d7F6b46fCFd7e7ed9fA0575Dd
Sender address	0x675A7359e0d96b9A25DCA5d08e1f1Ee85B9cD24f
Value	0.00 ETH
Gas used	298,171
Gas price	20,000,000,000
Gas limit	447,256
Mined in block	279

The system simplifies the checking of the proper working of the PDS as implemented by the Food Corporation of India. Let's discuss a fraudulent case that the proposed system can handle.

Condition: Suppose vendor A possesses 20 kgs of ration at the end of the month considering the vendor provided 10 kgs of ration illegally to a non-ration cardholder. This results in a fraudulent case where the vendor should actually end up with a stock of 30 kgs of ration at the end of the month.

Result: As the authority has the access to the blockchain, vendor A's transactions can be checked and it can be found out that vendor A should end up with 30 kgs of stock left at the end of the month. Physical inspection can be carried out at the

vendor's shop by the authority. As the vendor would possess only 20 kgs of ration during physical inspection, vendor A should be added to blacklist by the authority for illegally providing 10 kgs of ration. Once added to the blacklist, vendor A won't be able to carry out any actions on the system and thus would be eliminated from the blockchain network.

5 Conclusion

In this paper, we have discussed an application which is helpful to oversee that the PDS is implemented properly without any shortcomings or disturbances by fraudulent vendors. It also gives the consumers a direct communication channel to the authority for voicing their complaints and concerns, allowing the authorities to be more in synchronization with the real world scenario. The system tracks all the transactions made of each commodity and also allows its users a user-friendly and easy-to-operate experience.

In future, the proposal could have a multilingual feature allowing non-English-speaking citizens a better user experience. Functionalities like updation of family sizes of each consumer, consumer requests of commodities can also be implemented. Security and performance metrics can also have certain advances. Less-expensive cryptocurrency can also be accepted. The current system can be made scalable as per the actual standards.

References

1. <https://www.gurumavin.com/pds-in-india/> Referred on: Sep 22
2. Pawade D, Jape S, Balasubramanian R, Kulkarni M, Sakhapara A (2018) Distributed ledger management for an organization using blockchains. *Int J Educ Manage Eng* 8(3):1
3. <https://timesofindia.indiatimes.com/city/chandigarh/punjab-bjp-leader-raju-alleges-ration-distribution-scam/articleshow/96806282.cms> Published on: Jan 23
4. <https://news.abplive.com/news/india/madhya-pradesh-scam-unearthed-in-state-s-free-food-programmes-ration-distribution-irregularities-finds-auditor-1552129> Published on: 07 Sep 2022
5. <https://www.hindustantimes.com/cities/others/ed-arrests-three-for-involvement-in-pds-scam-in-maharashtra-101613729930170.html> Published on: Feb 19 2021
6. Anni DJS, Nesrudheen VP, Abdulla N, Nihara N, Kurikkal SSA (2021) An android app: virtual queuing system for public distribution system. In: 2021 IEEE International conference on mobile networks and wireless communications (ICMNWC). IEEE, pp 1–4
7. Shukla S, Patil A, Selvin B (2018) A step towards smart ration card system using RFID & IoT. In: 2018 international conference on smart city and emerging technology (ICSCTET). IEEE, pp 1–5
8. Vaisakh AK, Ganesh KV, Suresh S, Vincent L, Thobias PT, Nair IP (2019) IoT based intelligent public ration distribution. In: 2019 international conference on communication and electronics systems (ICCES). IEEE, pp 1894–1897

9. Rajesekaran MP, Arthi R, Balaji D, Daniel P (2017) Automatic smart ration distribution system for prevention of civil supplies hoarding in India. In: 2017 4th international conference on advanced computing and communication systems (ICACCS). IEEE, pp 1–5
10. Aishwarya M, Nayaka AK, Divyashree N, Padmashree S (2017) Automatic ration material dispensing system. In: 2017 international conference on trends in Electronics and informatics (ICEI). IEEE, pp 852–856
11. Goradia J, Doshi S (2015) Automated ration distribution system. Proc Comput Sci 45:528–532
12. Shah N, Jogani D, Bavania S, Nayak S (2022) Automation in government ration distribution system using Atmega32 and RFID
13. Padmavathi R, Azeezulla KMM, Venkatesh P, Mahato KK, Nithin G (2017) Digitalized aadhar enabled ration distribution using smart card. In: 2017 2nd IEEE international conference on recent trends in electronics, information & communication technology (RTEICT). IEEE, pp 615–618
14. Rajalakshmi D, Madhumitha D, Devi Sree G, Ayesha Tanseera A, Rajalakshmi NR, Rajeswari S (2022) Smart ration system using application development with MONGO database. In: 2022 international conference on augmented intelligence and sustainable systems (ICAIS). IEEE, pp 771–776
15. Shwetha AN, Prabodh CP (2019) Blockchain-bringing accountability in the public distribution system. In: 2019 4th international conference on recent trends on electronics, information, communication & technology (RTEICT). IEEE, pp 330–335
16. Madaan G, Kumar A, Bhushan B (2022) Blockchain assisted secure data sharing in intelligent transportation systems. Smart Sustain Approaches Optimiz Perfor Wirel Netw: Real-time Appl 167–187
17. Binjola K, Joshi K, Bondale R, Sakhapara A, Pawade D (2022) Fake passport detection using blockchain. In: 2022 5th international conference on advances in science and technology (ICAST). IEEE, pp 243–246
18. Pawade D, Sakhapara A (2020) Blockchain based secure traffic police assistant system

Novel Architecture and Secured Food Traceability Application Based on Ethereum Blockchain



P. Joseph, B. Yuvaraj, and A. Sai Sabitha

Abstract Blockchain technology is becoming increasingly popular as a solution for enhancing food traceability and safety in the food supply chain. In this paper, we introduce a novel architecture and application for secure food traceability using Ethereum blockchain technology. The work highlights the advantages of using blockchain in the food supply chain and address the challenges that need to be overcome. Our proposed architecture and application aim to provide a transparent and secure platform for tracking food from farm to table. The paper provides a thorough evaluation of the proposed architecture, applications and discusses the future research directions. Our findings demonstrate the potential of blockchain technology in developing reliable and secure food traceability applications. The proposed design utilizes a network and smart contracts to give secured data from homestead to fork, making it valuable for Food Control Specialists to forestall food security issues.

Keywords Blockchain · Smart contracts · Supply chain management · Product provenance · Quality certification · Agrifood supply chain · Food traceability · Ethereum Ganache network

1 Introduction

The popularity of online food delivery services has grown significantly, providing convenience to consumers with busy lifestyles [1]. However, concerns about a lack of trust among players in the food ecosystem have arisen due to the use of third-party applications, leading to difficulties for customers [2]. The issue of food fraud is a global problem that poses risks to the economy and public health, emphasizing the importance of traceability across the food supply chain [3]. Although many countries have laws that mandate the use of traceability systems [4], they are not always sufficient to safeguard consumers from fraud [5]. The blockchain technology has

P. Joseph (✉) · B. Yuvaraj · A. Sai Sabitha

Department of Computer Science and Engineering, Hindustan Institution of Technology and Science, Padur, Chennai, India
e-mail: p.josephstar@gmail.com

been proposed as a solution to address these challenges [6–8]. It is decentralized [9], tamper-proof [10], and eliminates the need for third-party validation to ensure the reliability and integrity of the data [11]. The various research efforts have focused on investigating the impact of blockchain technology on the agrifood supply chain, as this is increasingly being implemented in traceability systems [12–14]. While the benefits of blockchain technology for the food industry are apparent, its practical implementation requires a comprehensive strategy involving the cooperation of all stakeholders in the food supply chain as well as the integration of several technologies and systems [15–17]. However, when implemented effectively, the blockchain technology can revolutionize the food industry enhancing the global food safety and security. This paper proposes a novel architecture and secure food traceability application based on the Ethereum blockchain, which is being described in detail in the subsequent sections.

2 Literature Survey

Scaling and Security

Hafid et al. [1] proposes a methodology for probabilistic security analysis of sharding-based blockchain protocols, addressing the security challenges associated with sharding. This technique for scaling blockchain networks can introduce vulnerabilities, and the authors' methodology can help identify and mitigate potential security risks.

Kokoris Kogias et al. [3] proposes OmniLedger, a consensus protocol that enhances the security and performance of Bitcoin through collective signing. This demonstrates the ongoing efforts to improve the scalability and security of blockchain systems and highlights the potential for innovative solutions to address these challenges.

Becker [4] analysis of Merkle signature schemes and trees provides insights into the cryptography behind many blockchain protocols, including Bitcoin. Understanding these concepts are essential for ensuring the security and integrity of blockchain systems.

Algorithmic Analysis

Motwani and Raghavan [2] introduction to randomized algorithms covers a range of techniques, including Monte Carlo methods and randomized approximation algorithms. This is useful for analyzing the efficiency and effectiveness of algorithms in various contexts and can help inform the design and implementation of blockchain protocols.

Decentralization and Centralization

Danezis and Meiklejohn [5] proposal for a centrally banked cryptocurrency aims to combine the benefits of decentralized cryptocurrencies with the stability and predictability of central banking. This demonstrates the potential for blockchain technology to disrupt traditional financial systems and highlights the ongoing debate over the optimal balance between centralization and decentralization in blockchain systems.

Trust Management

Movahedi et al. [6] survey of trust management frameworks for mobile adhoc networks is relevant to blockchain technology because it highlights the importance of trust and security in distributed systems. These frameworks can be useful for ensuring the reliability and security of blockchain networks.

Blockchain Architecture

Pervez et al. [7] comparative analysis of DAG-based blockchain architectures highlights the ongoing innovation and experimentation in blockchain design. Different approaches may be more suitable for different use cases, and understanding the strengths and weaknesses of various blockchain architectures is essential for making informed decisions about their implementation.

E-Voting

Malhotra et al. [8] e-voting platform utilizes blockchain technology and smart contracts to address several issues in traditional voting systems, such as tampering of votes, lack of transparency, and voter privacy concerns. By leveraging the immutability, decentralization, and transparency of blockchain technology, the proposed platform can potentially provide a more secure and reliable voting system.

Scalability

Garzik [9] proposal to increase the block size of Bitcoin highlights the ongoing debate on how to address the scalability challenges of blockchain technology. While increasing the block size can increase transaction throughput, it also poses challenges such as increased storage requirements and potential centralization of the network. This example demonstrates the need for careful consideration and analysis of proposed changes to blockchain systems to ensure their long-term sustainability and security.

3 Methodology

The system involves three modules: Backend Development, Smart Contract Development, and Blockchain Deployment.

In the Backend Development module, a RESTful API is created using HTTP solicitations to get to and manipulate data. The API considers efficient utilization of bandwidth and standardizes communication between the designer and the operating framework or application.

The Smart Contract Development module involves identifying entertainers, ideas, and occasions connected with the cycle and designing dynamic, static, or prophet driven smart contracts. Smart contracts computerize the execution of intricate arrangements among parties and eliminate the requirement for human interaction. They are designed using attributes, functions, occasions, and modifiers, and executed automatically when the conditions outlined in the understanding are met.

The Blockchain Deployment module involves a few stages, including reading and compiling the Solidity file, building, signing, and sending the contract to a blockchain, and testing the deployment using a simulated blockchain environment. To interact with a contract, the contract address and ABI should be obtained.

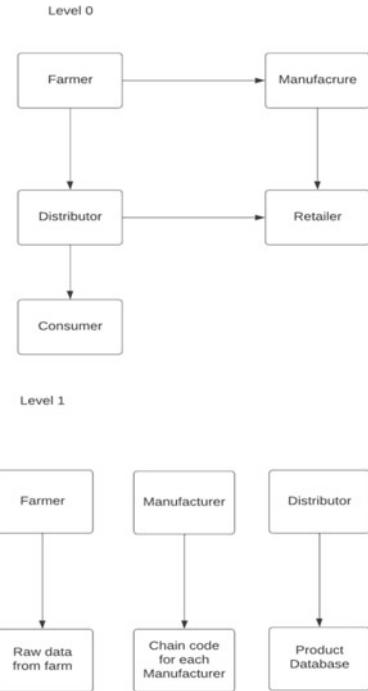
4 Proposed Method

In this context, a distributed ledger technology like Blockchain (BC) provides a complete and immutable audit trail of transaction data for every stage of the food supply chain, enabling transparency and digital certificates with verifiable and immutable records. The blockchain is particularly suited for regulated industries like agrifood due to its secure data storage which cannot be changed.

In order to track food from farm to table, a continuous flow of data is needed throughout the supply chain which is not possible with conventional database technologies. The blockchain provides an effective solution by creating a single ledger accessible by all system entities. The ledger contains a chaincode for each food manufacturer's product specification which grants the manufacturer access to the blockchain's raw material specifications.

The data associated with the products are then moved from the blockchain to a conventional database management system. The process ensures that the high-level details of the product are accessible by all the partners with extra information added at each phase of the creation. Furthermore, the chaincode is delivered to the consumer along with the product as specifications for the product.

Although the potential of blockchain-based traceability in the food supply chain is promising, there are still numerous challenges that need to be addressed. The laws, standards, and labeling are often in conflict between states in many countries. The data sharing and interoperability between the systems is difficult due to the lack of technology connecting the various blockchain systems. Additionally, states and nations must create a common ontology that explains the significance of the recorded data elements and values in the food supply chain.

Fig. 1 Data flow diagram

In conclusion, the blockchain provides a secure, immutable, and transparent platform for tracking food from farm to table. However, the system is far from ready for widespread adoption due to the various challenges that need to be addressed.

A Data Flow Diagram (DFD) can be used to represent the flow of data in the proposed methodology for tracking food from farm to table using blockchain technology. Here is a sample DFD (Fig. 1):

Level 0 DFD

In the above DFD, the boxes represent the entities in the system, and the arrows represent the flow of data between them. The farmer provides raw data about the food, which is stored on the blockchain as chaincode for each manufacturer. The manufacturer accesses the chaincode to create a product database that contains detailed information about the product. The product database is shared with the distributor, who in turn shares it with the retailer. Finally, the consumer accesses the product database to obtain information about the food product.

The above DFD shows how blockchain technology can be used to create a secure and transparent supply chain for food products. However, as mentioned in the text, there are still challenges that need to be addressed before this system can be widely adopted.

5 Implementation

In order to implement the proposed blockchain-based food traceability system, the following steps should be followed:

1. Collect and Analyze Data: The initial step is to gather and examine data from the whole food inventory network. The data ought to incorporate data, for example, item determinations, beginning of fixings, production date, and delivery data. Programmed data catch includes using innovation to gather data without human mediation, usually utilized in assembling, planned operations, and transportation. AI is a type of man-made intelligence that empowers PCs to gain from data and work on their presentation. The recorded data should be exact, steady, and dependable, and appropriately got. The food inventory network the board framework catches and stores data about creation, handling, transportation, and dispersion. The blockchain innovation gives an unchanging record of all exchanges and data, guaranteeing trustworthiness and straightforwardness.
2. Create a Blockchain Network: When the data has been collected, a blockchain network should be created. The network ought to remember all the participating actors for the supply chain and ought to be gotten utilizing digital signatures and encryption.

Blockchain Deployment

For deployment, what we can do is read the Solidity file, compile it, change the contract to Python, build the contract, sign the contract, and send the contract. To deploy our contract, we need a blockchain. The Ganache CLI provides a simulated blockchain that you can use to test out things in deployment. To use Ganache, we need a tool to help us create our contract in Python, this is where Web3.py comes in. The two ways to interact with a contract with call() (calls a function and gets return value) and transact() (make changes in the contract). When trying to interact with a contract, you need the contract address and the contract ABI.

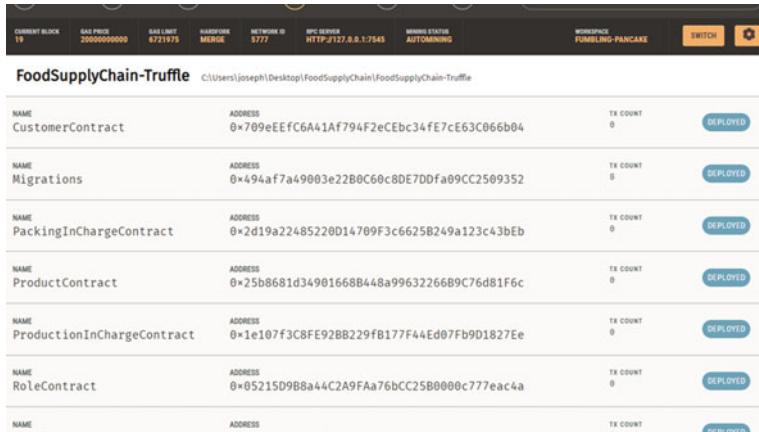
3. Create a chain code: chain code ought to be utilized to store data on the blockchain and move data from existing database management systems.

The chain code is utilized to store data on the blockchain and access it, with an indexing algorithm that is used to organize data so that it can be easily accessed and managed. A chain code is utilized to store data on the blockchain and move data from customary database the executives frameworks. It characterizes how data is added to the blockchain and got to. It goes about as an extension between the database framework and the blockchain network, overseeing access control. The chain code benefits incorporate expanded straightforwardness, recognizability, and decreased hazard of misrepresentation or blunders. It takes into account mechanized processes, diminishing manual data passage and expanding productivity. In a blockchain-based food store network framework, chain code is vital for secure and straightforward following, prompting a more secure and more productive food production network.

4. Develop a standard ontology: A standard ontology ought to be created to guarantee interoperability and data dividing among various blockchain systems. This ought to incorporate characterizing a bunch of standardized data components and values that can be utilized across all blockchain systems.
5. Develop common messaging standards: Normal messaging standards ought to be created to empower interoperability and data partaking in the food store network. These standards ought to frame a bunch of rules and conventions for sending data between different blockchain stages and inventory network entertainers. The common messaging standards assume a basic part in empowering interoperability and data sharing in the food production network. Developing the common messaging standards include, laying out a bunch of decides and conventions that guarantee data, is communicated in a standardized organization, making it simpler for various frameworks to peruse and understand the data. For instance, common messaging standards could characterize a standard data design for recording data about the beginning, quality, and security of food items, as well as the guidelines for sending this data between various blockchain stages and production network members. By laying out a common arrangement of rules and conventions for data transmission, these standards guarantee that data is communicated in a standardized organization that can be handily perused and grasped by various frameworks and members. This makes it simpler for production network members to share data and team up more really, eventually prompting more prominent straightforwardness and detectability all through the food production network.
6. Coordinate with existing systems: The blockchain application ought to be incorporated with existing systems to guarantee data precision and interoperability. As of now, there is no uniform framework for taking on blockchain-based discernibility in the food store network, and the greatest deterrent is the absence of interoperability and data sharing between various blockchain frameworks. To address this test, states and nations need to foster a standard philosophy and normal informing standards for blockchain-based discernibility in the food store network. This would consider simpler interoperability and sharing of data between various blockchain frameworks, bringing about a more proficient and straightforward food store network.
7. Deploy the application: The last step is to deploy the application on the blockchain network. When the application is deployed, it ought to be tested to guarantee it is working appropriately.

From Fig. 2.

1. CustomerContract: This contract is likely used to manage customer information and interactions.
2. Migrations: This contract is likely related to the deployment of other contracts, as it has a nonzero transaction count.
3. PackingInChargeContract: This contract is likely used to manage packaging-related tasks and responsibilities.



The screenshot shows the Ganache interface with the following details:

- Current Block:** 19
- GAS PRICE:** 20000000000
- Gas Limit:** 6721975
- Hardfork:** MERGE
- Network ID:** 5777
- RPC SERVER:** HTTP://127.0.0.1:7545
- MINING STATUS:** AUTOMINING
- WORKSPACE:** FUMBLING-PANCAKE
- SWITCH** and **⚙️** buttons

FoodSupplyChain-Truffle (C:\Users\joseph\Desktop\FoodSupplyChain\FoodSupplyChain-Truffle)

NAME	ADDRESS	TX COUNT	DEPLOYED
CustomerContract	0x709eEefc6A41Af794F2eCEbc34fE7cE63C066b04	0	DEPLOYED
Migrations	0x494af7a49003e22B0C60c8DE7DDfa09CC2509352	0	DEPLOYED
PackingInChargeContract	0x2d19a22485220D14709F3c6625B249a123c43bEb	0	DEPLOYED
ProductContract	0x25b8681d34901668B448a99632266B9C76d81F6c	0	DEPLOYED
ProductionInChargeContract	0x1e107f3C8FE92BB229fB177F44Ed07Fb9D1827Ee	0	DEPLOYED
RoleContract	0x05215D9B8a44C2A9FAa76bCC25B0000c777eac4a	0	DEPLOYED
NAME	ADDRESS	TX COUNT	DEPLOYED

Fig. 2 Implementation of smart contracts reflected on Ganache

4. **ProductContract:** This contract is likely used to manage product information, such as its name, description, and price.
5. **ProductionInChargeContract:** This contract is likely used to manage production-related tasks and responsibilities.
6. **RoleContract:** This contract is likely used to manage user roles and permissions.
7. **SupplierContract:** This contract is likely used to manage supplier information and interactions.
8. **TransactionDetailsContract:** This contract is likely used to manage transaction details, such as the parties involved and the amount exchanged.
9. **UsersContract:** This contract is likely used to manage user information and interactions, such as account creation and authentication.

Ganache (Fig. 3).

1. **TransactionDetailsModel:** Represents a transaction record with methods to get, insert, update, and delete records. Interacts with a blockchain by creating new transactions.
2. **CustomerModel:** Manages customer data in a SQL database with methods for CRUD operations. Can retrieve all customers, retrieve by ID, insert, update, and delete.
3. **PackingInChargeModel:** Manages the PackingInCharge table in a SQL database with methods for CRUD operations. Can retrieve all records, retrieve by ID, insert, update, and delete.
4. **ProductionInChargeModel:** Represents a production in charge object with methods for CRUD operations on records. Can retrieve all records, retrieve by ID, insert, update, and delete.

BLOCK 19	MINED ON 2023-03-21 17:15:34	GAS USED 203307	1 TRANSACTION
BLOCK 18	MINED ON 2023-03-21 16:14:3124	GAS USED 208112	1 TRANSACTION
BLOCK 17	MINED ON 2023-03-21 16:14:3124	GAS USED 76975	1 TRANSACTION
BLOCK 16	MINED ON 2023-03-21 16:14:3123	GAS USED 208113	1 TRANSACTION
BLOCK 15	MINED ON 2023-03-21 16:14:3123	GAS USED 418694	1 TRANSACTION
BLOCK 14	MINED ON 2023-03-21 16:14:3123	GAS USED 208113	1 TRANSACTION
BLOCK 13	MINED ON 2023-03-21 16:14:3123	GAS USED 76975	1 TRANSACTION
BLOCK 12	MINED ON 2023-03-21 16:14:3123	GAS USED 208113	1 TRANSACTION
BLOCK 11	MINED ON 2023-03-21 16:14:3123	GAS USED 76975	1 TRANSACTION
BLOCK	MINED ON	GAS USED	

Fig. 3 Blocks created by the user reflected in Ganache

5. RoleModel: Has several properties corresponding to the columns of a Role table. Allows for querying, inserting, updating, and deleting records. Uses pyodbc library to connect to the database.
6. SupplierModel: Represents a supplier in a database with methods for CRUD operations. Can retrieve all records, retrieve by ID, insert, update, and delete.
7. UserModel: Allows the user to interact with the database with methods for CRUD operations. Can retrieve all records, retrieve by ID, insert, update, and delete. Uses PyODBC library to connect to the database.
8. Blockchain: Stores transaction data in a database and uses a blockchain to record transactions. Allows for secure, transparent transactions between customers and suppliers.

Block chain Report: Transaction log from Fig. 4 is being recorded with details of customer ID, supplier ID, product ID, price, quantity, hash, and previous hash. Each transaction seems to be associated with a unique hash value.

Conclusion: Summarize the importance of RESTful APIs, smart contract development, and blockchain deployment steps to implement:

1. Identify the use case and define the requirements.
2. Design the smart contract, defining the actors, concepts, and events.
3. Develop the smart contract using a language such as Solidity.
4. Compile the smart contract and develop the associated APIs.
5. Deploy the smart contract to a blockchain.
6. Test the smart contract and associated APIs.
7. Secure the smart contract and associated APIs.
8. Monitor and maintain the smart contract.

Fig. 4 Transaction log

6 Conclusion and Future Work

The paper investigates the use of block chain innovation in the food business to guarantee the quality and traceability of food items. The proposed design utilizes a network and smart contracts to give straightforward data from homestead to fork, making it valuable for Food Control Specialists to forestall food security perils. The utilization of block chain innovation gives decentralized, valid, and straightforward data that is put away lined up with customary datasets, without influencing regular business tasks or requiring extra preparation for representatives. The objective is to cost-successfully measure and ensure the quality of food items and energize sound rivalry between organizations to persistently further develop item quality.

We are working with food certificate specialists and NGOs to unite the kind of information (metaphysics) that are supposed to confirm and record for natural items inside China. It will help with making a headed together chaincode for different periods of the production network—it will give a cost capable and generous structure with a uniting standard Programming point of interaction. The various state-run

organizations are also showing interest in using blockchain development in organization. It necessities to encourage a structure where the different blockchains can move the information on each other or can be joined. No business can be totally focused or decentralized without agreeing to less in locales like security, protection, execution, and adaptability. State-run organizations and classified affiliations are moreover stressed over their information security and insurance. It necessities to cultivate a framework to deal with the information on their servers at their premises and bring simply needed results from the servers to the blockchain for examination and assumptions. The most noteworthy snag for state-run organizations to be significant for this chain is the regulations to protect the information that can be used to impact fair rivalry and other public food security risks.

References

1. Hafid A, Hafid AS, Samih M (2020) A methodology for a probabilistic security analysis of sharding-based blockchain protocols. In: Blockchain and applications: international congress. Springer International Publishing, pp 101–109
2. Motwani R, Raghavan P (1995) Randomized algorithms. Cambridge University Press
3. Kokoris Kogias E, Jovanovic P, Gailly N, Khoffi I, Gasser L, Ford B (2016, Aug) Enhancing bitcoin security and performance with strong consistency via collective signing. USENIX Association
4. Becker G (2008) Merkle signature schemes, Merkle trees and their cryptanalysis. Technical report, 12, 19, Ruhr-University Bochum
5. Danezis G, Meiklejohn S (2015) Centrally banked cryptocurrencies. arXiv preprint [arXiv:1505.06895](https://arxiv.org/abs/1505.06895)
6. Movahedi Z, Hosseini Z, Bayan F, Pujolle G (2015) Trust-distortion resistant trust management frameworks on mobile ad hoc networks: a survey. IEEE Commun Surv Tutorials 18(2):1287–1309
7. Pervez H, Muneeb M, Irfan MU, Haq IU (2018, Dec) A comparative analysis of DAG-based blockchain architectures. In: 2018 12th International conference on open source systems and technologies (ICOSST). IEEE, pp 27–34
8. Malhotra M, Kumar A, Kumar S, Yadav V (2022) Untangling e-voting platform for secure and enhanced voting using blockchain technology. Transforming management with AI, big-data, and IoT. Springer International Publishing, Cham, pp 51–72
9. Garzik J (2015) Block size increase to 2 MB. In: Bitcoin improvement proposal, p 102
10. Rubin J, Naik M, Subramanian N (2014) Merkleized abstract syntax trees. XP055624837, Dec, 16(3)
11. Wang J, Wang H (2019, Feb) Monoxide: scale out blockchains with asynchronous consensus zones. In: NSDI, vol 2019, pp 95–112
12. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system. Decentralized Bus Rev 21260
13. Vukolić M (2016) The quest for scalable blockchain fabric: proof-of-work vs. BFT replication. In: Open problems in network security: IFIP WG 11.4 international workshop, iNetSec 2015, Zurich, Switzerland, 29 Oct 2015. Revised selected papers. Springer International Publishing, pp 112–125
14. Zamani M, Movahedi M, Raykova M (2018, Oct) Rapidchain: scaling blockchain via full sharding. In: Proceedings of the 2018 ACM SIGSAC conference on computer and communications security, pp 931–948

15. Hastie T, Tibshirani R, Friedman JH, Friedman JH (2009) The elements of statistical learning: data mining, inference, and prediction, vol 2. Springer, New York, pp 1–758
16. Cohen W, Ravikumar P, Fienberg S (2003, Aug) A comparison of string metrics for matching names and records. In: KDD workshop on data cleaning and object consolidation, vol 3, pp 73–78
17. Al-Khatib AA, Mohammed B, Abdelmajid K (2020) A survey on outlier detection in internet of things big data. In: Big data-enabled internet of things. IET, London, UK, pp 265–272

Quick Response Code-Based Fake Product Verification System Using Blockchain Technology



Vijayashree R. Budyal, Sriraksha R. Kolgi, B. K. Tejaswini, G. Hrithik,
and S. A. Nabila Qadri

Abstract One of the most significant issues in today's retail business is product counterfeiting. Counterfeit goods are just low-quality imitations of legitimate brands. Many various methods have been used to combat product counterfeiting, including RFID tags, artificial intelligence, machine learning, QR code-based systems, and many others. Counterfeit items represent a substantial problem to industries and customers, resulting in economic losses, safety concerns, and trust issues, as well as certain drawbacks such as high processing power to execute operations security and centralized database. This paper suggests usage of decentralized blockchain technology, the SHA-256 algorithm, and quick response (QR) code to improve the identification of counterfeit products. The integration of blockchain technology and QR codes enables a transparent and decentralized system to verify the authenticity of products throughout the supply chain. The proposed system involves assigning unique identifiers to genuine products, which are recorded on a blockchain network. QR codes containing these identifiers are affixed to product packaging or labels. Consumers, retailers, or inspectors can easily scan the QR codes using a mobile device, retrieving the products unique identifier. The advantages of this blockchain-based solution include improved traceability, enhanced security, consumer empowerment, and supply chain efficiency. The challenges such as adoption barriers, technical complexity, scalability, and privacy concerns need to be carefully addressed during the implementation.

Keywords Counterfeit (fake) product · Quick response code · Immutable · Blockchain

V. R. Budyal · S. R. Kolgi (✉) · B. K. Tejaswini · G. Hrithik · S. A. Nabila Qadri
Department of Information Science and Engineering, Sai Vidya Institute of Technology
(Affiliated to Visvesvaraya Technological University, Belagavi), Bengaluru, India
e-mail: srirakshakolgi@gmail.com

V. R. Budyal
e-mail: vijayashree.rb@saividya.ac.in

G. Hrithik
e-mail: hrithikg.19is@saividya.ac.in

1 Introduction

The market for counterfeit goods is growing. In order to fight counterfeiting, which has many detrimental effects including product recalls, sales losses, and more, this paper proposes to use blockchain technology. Blockchain provides a secure and dependable monitoring system from the creation or mining of raw materials to the opposite end of the supply chain, which aids in the fight against counterfeiting. When a product is developed globally, risk factors such as counterfeiting and duplication are always there, and they can have an influence on the brand name, revenue, and customer satisfaction of the firm. The sale and promotion of fake goods are expanding tremendously. It has an impact on the businesses' profits, sales, and reputation, and it also presents a fatal risk to naïve customers [1]. Blockchain technology can be used to determine a product's uniqueness. In today's world, how do you know if you are buying a genuine product? The supply chain has been using radio frequency identification (RFID) technology for anti-counterfeiting methods for more than a decade now. But this technology is outdated when compared to other booming technologies like Blockchain. One of the major concerns in the industry is fake drugs. The World Health Organization (WHO) reports that bogus medications frequently kill youngsters in underdeveloped nations [2]. Hence there is a high requirement to detect the products that are fake which would otherwise harm the society. Consumers may reliably know the source of the acquired product without having to completely rely on reliable third parties because to the untraceability, transparency, and guarantee provided by the blockchain.

The supply chain management system offers a wide range of products determining whether the things are genuine or a fraud. This is due to the enormous losses and challenges that producers of fake or stolen items are experiencing [3]. Blockchain technology is one that guarantees product integrity. Blockchain is a distributed ledger technology that securely and impossibly saves information. A distributed network of computer systems known as a "blockchain" transmits records of electronic transactions often and broadly [4]. Small and medium-sized businesses (SMEs) today suffer financial difficulties that cannot be matched to those encountered by large corporations with deep pockets. SMEs in the brand management industry will eventually need to cut expenses and will be unable to employ traditional methods to prevent items from being counterfeited [5].

This paper proposes to use QR codes and Python programming language in the project to enhance the security. In this system, the product will be registered from the manufacturers end, and a unique QR code will be generated for each product, and the hash code will be assigned to the product details for security. On the customers' side when the QR code is scanned, the hash will be verified and the product's authenticity will be checked and the message will be displayed on the customer's screen.

2 Related Work

A number of studies have put forth various strategies for setting up a supply chain management system based on blockchain. One strategy relies on centralized cloud data storage, and since the data is kept only in one location, a single point of failure could cause the system to fail [2]. Another paper uses a decentralized application system that introduced the use of Ethereum blockchain technology and its implementation that is costly. A group of researchers developed an efficient digital signature-based anti-product forgery solution [4]. One of them demonstrated a system for identifying fake goods using an Android app that allows users to look for goods on the blockchain network [6]. A blockchain-based bogus product detection system using the SHA-256 algorithm was demonstrated in another article [7]. Another article presented a food traceability system that combines blockchain and IoT. The food quality was assessed using fuzzy logic in this model [9]. In a study, a system was shown that combined RFID and blockchain to overcome post supply chain constraints [10]. To improve the supply chain, a paper combined blockchain with IoT to track the originality of the product [11].

It has become a habit for many people, particularly prospective purchasers, to read product reviews before making a purchase. Positive customer feedback can generate significant financial rewards for a business, which can be used as data when determining how to develop products and what services to provide customers [1] a smart tag-based, cloud-enabled brand protection, and anti-counterfeiting system for the wine industry. Quick response codes, functional inks, a cloud-based system, and two-way communication between the vineyard and the end user are the fundamental tenets of smart tags [2].

Blockchain is used to demonstrate how to increase trust in social media-shared news. The concept suggested a cutting-edge decentralized traceability system that combines blockchains and social media. The user ID is retrievable from there. The transaction will notify the entire chain if fresh information is created. The framework will only assess news that has attained a particular level of “virality”. The information will spread across the chain. This news will initially have no ranking. The news will display a ranking for the users, and validators’ ratings will accumulate over time. This rating represents how reliable and authentic a piece of news is [11].

One-time password (OTP) authentication is utilized in a study to validate the validity of supply chain participants and products. After a product is transported to the next level of the supply chain, its details are updated in the blockchain. The supply chain and blockchain technologies, which by themselves offer the system great security and transparency, are used in the study to demonstrate a novel and useful phenomenon [12].

In the earlier system, there were many drawbacks. Firstly, the data was centralized, and if there was a node failure, it would cause the system failure. Secondly, the data security was less as they had to rely on the third party for securing the data.

There were high chances of data being changed by the middle man, and hence the fake product could be declared as the real one.

To overcome these drawbacks, this paper proposes a blockchain-based fake product identification system where the data is highly secure and is immutable. With an addition of quick response code, the security of data is increased. In order to enhance and increase the security further, this system stores the manufacture entered data in a hashed format by hashing the data twice. The QR code contains the product details in the encoded format that will be decoded later by Python before the product is decided as real or fake.

The above graph shows the accuracy of proposed system and existing system.

Our contribution includes: (1) To create a legitimate manufacture detail in the data storage using blockchain technology, which includes Product ID, Product Name, Brand, Price, and Product Rating. (2) To secure data, blockchain technology is being used. (3) To ensure customer satisfaction by generating quick response (QR) code to get product details.

The rest of the paper is organized as follows. Section 3 explains the proposed work for blockchain-based fake product identification using quick response code. Section 4 explains the results, and Sect. 5 concludes the paper.

3 Proposed Work

This system has three major stakeholders, the admin, the manufacturer, and the customer as shown in Fig. 1.

3.1 Admin

This system considers admin as one of the major stakeholders. In this paper, three types of products can be verified for. The products are shoes, medicines, and watch that can be extended for any other products. The admin has the authority to add any of these products into the blockchain network. The systems home page displays a button for admin which when clicked redirects to a login page. When the login credentials match, a page where the admin will have options to click on the button that redirects to respective pages to add shoe, medicine, or watch details.

3.2 Manufacturer

The manufacturer is the main stakeholder of this system. Once a manufacturer is given a login ID and password, which when entered in the login page and is appropriate redirects the page to respective product details add page. The manufacture's function includes adding the product details appropriately that will be stored in the blockchain network. The various information collected about the product is, brand, manufacturer

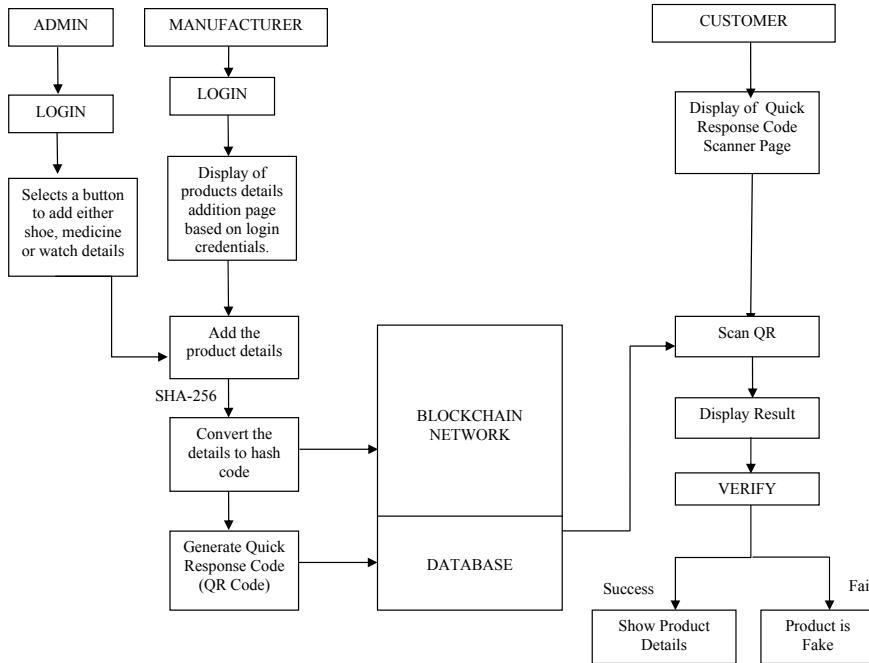


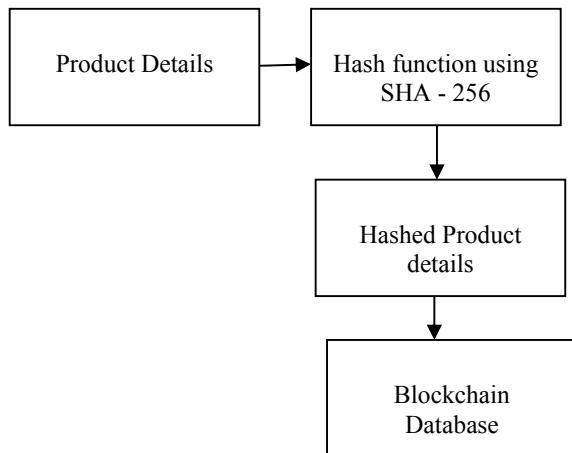
Fig. 1 Proposed system architecture

name, ID, manufacturing date, price, size, and type. This is with respect to shoe. With different products, different categories will be listed for the manufacturer to enter the details. The data is entered as shown in Algorithm 1, and the data is hashed using the SHA-256 algorithm so that no one can hack the data to change it. The data that is hashed is saved in the blockchain network, and hence the data now is very secure.

3.3 Customer

A customer is an important individual for the business. Customers are important because they drive the revenue for a product. The customer plays a major role as he has to scan the QR code to know the product originality. This system displays three buttons on the homepage. One of the buttons is customer which when clicked redirects to a page that contains a QR code scanner. When the customer clicks on upload QR code button, it opens a folder that contains the QR codes generated by entering the product details. The customer has to select the correct QR code and upload in the page. The QR code will be scanned, and the result will be displayed on the screen. The result will be a webpage link that contains the decoded hash code of the QR code. When the user copy's this URL and browses it on the web, it redirects

Fig. 2 Hashing procedure of product details



to a page that confirms if that product was authentic or fake. This is the only page that does not require a login to access the QR code scanner as there will be many customers who would want to know a products authenticity.

3.4 SHA-256 Blockchain

The SHA-256 algorithm is an unkeyed cryptographic algorithm that takes the input and produces a 256 bit long hash code as a result as shown in Fig. 2. This paper proposes to use the SHA-256 algorithm to encode the entered product details either by the admin or the manufacturer. This algorithm ensures that the data that is of variable size is transformed into a fixed size 256 bit string.

The SHA-256 algorithm converts the data into a complete unreadable format. This algorithm is used to verify the content of data that is to be kept as a secret. Users can verify that they have the genuine product by looking up the hash code, which can be made public. This algorithm is used in blockchain as it is highly impossible to reconstruct the initial or original data even by knowing the hash code.

3.5 Quick Response Code

A barcode is a label with information on the object to which it is fastened. Consumers can use QR codes as a counterfeit prevention tool to determine whether a product is genuine or not. The product details will be encoded safely in the QR code that the customer can scan later to know whether the product is authentic or not. The QR code is stored in the local server and is uploaded in the QR code scanner when a

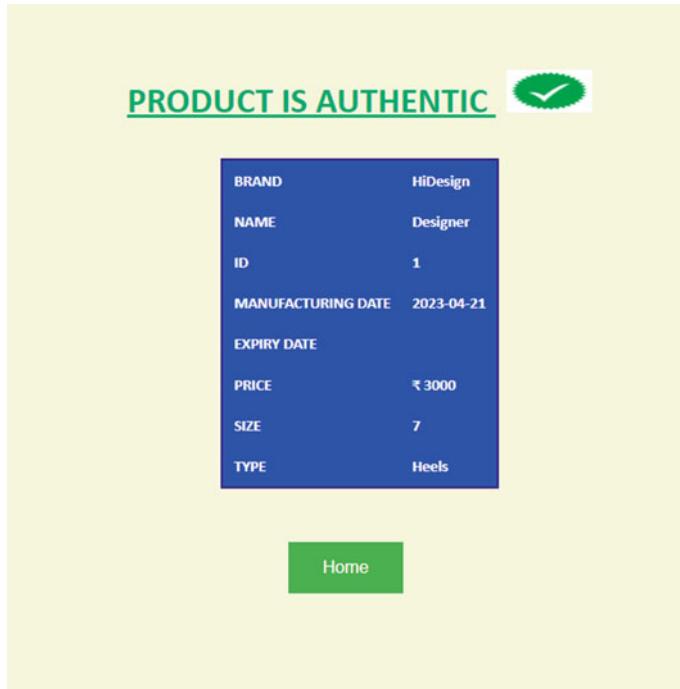


Fig. 3 Success page

button is clicked. The result displayed on the screen is a URL which is displayed by decoding the QR code. This URL when browsed displays the hash data which when verified displays if the product is authentic or fake as shown in Figs. 3 and 4, respectively.

Algorithm 1: Update Product Details

Input: Brand, Name, ID, Size, Type, Color, Price, Manufacturing Date, Expiry Date, Dosage.

- 1: function addProduct: addProduct (product_brand, product_name, manuf_date, expiry_date, product_id, product_price, product_size, product_type, product_color, product_dosage)
- 2: *this.product_brand* \leftarrow product_brand;
- 3: *this.product_name* \leftarrow product_name;
- 4: *this.manuf_date* \leftarrow manuf_date;
- 5: *this.expiry_date* \leftarrow expiry_date;
- 6: *this.product_id* \leftarrow product_id;
- 7: *this.product_price* \leftarrow product_price;
- 8: *this.product_size* \leftarrow product_size;
- 9: *this.product_type* \leftarrow product_type;

Fig. 4 Failure page

```

10: this.product_color ← product_color;
11: this.product_dosage ← product_dosage;
12: return product;

```

4 Results

The results obtained by the proposed system are shown in Fig. 3.

Figure 3 shows the authentication page when the hash code that is decoded from the QR code and displayed on the screen is compared to hash code stored in the blockchain. The product details are displayed on the screen along with a message that the product is authentic, and this conveys that the user can proceed to buy the product.

Figure 4 shows that the product is fake that will be displayed when a fake QR code is scanned and the decoded hash code does not match with the hash code that is stored in the blockchain. If the product is fake, it conveys the customer not to buy the product by paying huge amount.

In Table 1, the first column lists the key features of the system, while the subsequent columns compare the features of the blockchain-based fake product identification system with other IEEE papers that were published in the year 2022.

Table 1 .

Feature	Proposed paper	Comparison paper [13]
Product verification	QR code	No QR code
Authentication	SHA-256	Authentication tag
Scalability	High	Moderate
Efficiency	High	Moderate
Security	High	Moderate
User experience	Simple and intuitive	Complex

5 Conclusion

The paper shows a fully functional anti-counterfeit system that all the companies can adopt and start applying it to their products in order to be counterfeit free. By leveraging the transparency, immutability, and decentralized nature of blockchain technology, blockchain technology provides a way to eliminate this “middleman/central authority”.

It does this by filling three important roles recording transactions, establishing identity, and establishing contracts. Information security is one of the most important features of blockchain any company that wishes to have its product safe and non-duplicate can use this system and get their products registered that can help them to keep their products authentic. By doing this, the efforts put by the company for creating the product will not go waste, and it also gives justice to the customers who pay money to buy the products.

References

1. Sayyad TJ (2022) Fake product identification using blockchain technology. *Int J Future Gener Commun Network Eng Technol (IJMRSET)* 13(3)
2. Mani V, Prakash M, Lai WC (2022) Cloud-based blockchain technology to identify counterfeits. *J Cloud Comput Adv Syst Appl*
3. Balasubramani S, Pramanick S, Singh R, Kumar D (2022) An Ethereum based fake product identification system using smart contract. In: Proceedings of the sixth international conference on intelligent computing and control systems (ICICCS). IEEE Xplore
4. Ma J, Lin S-Y, Chen X, Sun H-M, Chen Y-C, Wang H (2020) A blockchain-based application system for product anti-counterfeiting. *IEEE Access*
5. Aggarwal M, Ranjan H, Gupta G, Walia S, Tyagi D (2022) Anti-counterfeit detection system using blockchain. *Int J Res Appl Sci Eng Technol (IJRASET)* 10
6. Sayyad TJ (2021) Fake product identification using blockchain technology. *Int J Future Gener Commun Network* 14:780–785
7. Tambe T, Chitalkar S, Khurud M, Varpe M, Raut SY (2021) Fake product detection using blockchain technology. *Int J Adv Res Ideas Innov Technol* 7:314–319
8. Toyoda K, Mathiopoulos PT, Sasase I, Ohtsuki T (2017) A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain. *IEEE Access* 5

9. Tsang YP, Choy KL, Wu CH, Ho GTS, Lam HY (2019) Blockchain-driven IoT for food traceability with an integrated consensus mechanism. *IEEE Access* 7:129000–129017
10. Rahmadika S, Kweka BJ, Latt CNZ, Rhee K (2018) A preliminary approach of blockchain technology in supply chain system. In: IEEE international conference on data mining workshops (ICDMW), pp 156–160
11. Anandhi S, Anitha R, Venkatasamy S (2018) RFID based verifiable ownership transfer protocol using blockchain technology. In: IEEE international conference on internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData)
12. Jayaprasanna MC, Soundharya VA, Suhana M, Sujatha S (2021) A block chain based management system for detecting counterfeit product in supply chain. In: Third international conference on intelligent communication technologies and virtual mobile networks (ICICV), pp 253–257
13. Dabbagh Y, Khoja R, Al Zahraei L, Al Showaier G, Nasser N (2022) A block chain-based fake product identification system. In: 5th Conference on cloud and internet of things (CIoT)

Interactive Learning for Patient Care: Blockchain Ingrained Electronic Health Record Management System with Patient Control, Data Quality and Security Assurance



Arvind K. Sharma , Gousia Habib, Savita Wadhawan, and Himani Soni

Abstract Objective: With the advent of healthcare specific multi-sensory applications and mobile-computing, a huge amount of data is being produced and accumulated periodically. This data not only requires accuracy but also the necessity to maintain its authenticity, security and access control. However the well-established cryptographic algorithms are capable enough to manage the security aspect, but somehow lacking to preserve the authenticity alone with digital-signatures and hash functions. The aforementioned problems concerning healthcare data have been gaining attention for years, and now the emerging blockchain technology is contributing to cope up with such situations by providing ingenious solutions. Another problem on healthcare data is related to access control which also requires astute solutions too, as this data belongs to patients and a patient should be the only entity who can decide, to what extent?, to whom?, this private information to be disclosed with privacy preservation. Now, in this paper a blockchain and cloud-based conceptual design for personal health record management system with patient control is proposed, in order to conquer existing problems in the healthcare industry on patient's medical records. The computational intensiveness will be managed with machine learning strategies. **Method:** To articulate a patient centric design for secure preservation and efficient access of electronic health records, blockchain technology along with cloud platform incorporated in this design. In order to maintain the quality of records advent machine learning classification technique namely transformers also used. **Result:** The design of electronic health records management system is pro-

A. K. Sharma

Yogananda School of Artificial Intelligence, Computers and Data Science, Shoolini University,
Solan, Himachal Pradesh 173229, India
e-mail: sharmaarvind00786@gmail.com

G. Habib

Indian Institute of Technology Delhi, Delhi 110016, India

S. Wadhawan

M.M. Institute of Computer Technology and Business Management, Maharishi Markandeshwar
Deemed to be University, Mullana, Haryana 133207, India

H. Soni

Department of Neurosurgery, M.M. Superspeciality Hospital, Maharishi Markandeshwar Deemed
to be University, Mullana, Haryana 133207, India

posed in this work, following with prototype building with Azure cloud platform.

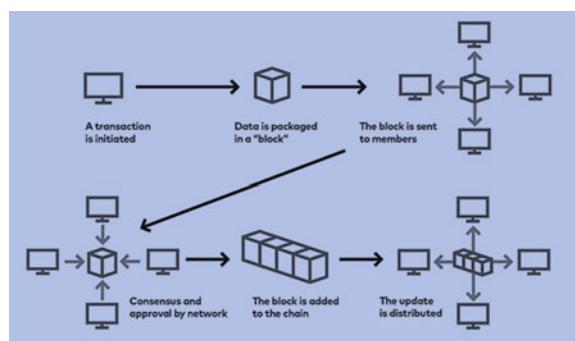
Conclusion: The work elucidated the prime limitations of electronic health records management schemes, and proposed a design which will help against hindrances for efficient, cost-effective, easy-to-use, secure and less intensive solution.

Keywords Authentication · Authorization · Blockchain · Cloud Computing · Confidentiality · Consensus · Electronic Health Records (EHR) · Electronic Medical Records (EMR) · Hash Function · Integrity · Interoperability · Machine Learning (ML) · Miner · Privacy · Proof of Work (PoW) · Proof of Stake (PoS) · Public Key Pair (K_{PR}, K_{PU}) · Security · Secret Key (K) · Smart-Contracts

1 Introduction

The inter-networking appears late 1970, not only for sharing the information but also to knit a web with networking technologies and resources to provide services [1, 2]. Every branch of industry obtaining benefits of Internet technologies viz. Blockchain, Cloud Computing, Internet of Things, for different perspectives and, the significance of above mentioned technologies is very demanding and prominent in healthcare [3]. The healthcare is a challenging domain which requires necessary consideration both for research and technology development concern possibilities, to provide reliable and quality services to/from medical institutions, research labs, medical practitioners, patients, etc., [4, 5]. The blockchain along with cloud computing and IoT examined as an emerging technology by researchers for managing crucial tasks of healthcare [6, 7], such as to maintain the authenticity, privacy and proper management both of electronic health/medical records along with reliable access controls. Figure 1 illustrate the working of blockchain.

Fig. 1 Working of blockchain



The cloud computing is capable enough to provide services for healthcare sector and with IoT technology its easy to send and receive the information on either sides but while sensitive information transmitting and used the chances of information breeches is high, here blockchain come for rescue [8]. The blockchain technology is quite helpful for preserving the security, authenticity of EHR/EMR by storing either it on-chain or off-chain, with suitable fingerprints [9–11]. Due to immutable nature of blockchain records, the authenticity will always be preserved and appropriate information from these medical records will be supplied as per requirements to concerns for further processing. Even though blockchain's benefits are plausible with cloud computing and IoT [12, 13], yet certain constraints of this technology raise points on the efficiency aspect of it. As the blockchain is categorized into three parts a public, private and hybrid blockchain models, each have its own benefits and limitations. By considering the scenario of healthcare, the public models (Ethereum) are easy-to-use, i.e., any user can participate, initiate a transaction, etc., but due to sensitivity of medical records this category doesn't seems secure and also computational intensiveness is quite high as PoW consensus mechanism used, on the other hand the private model (Hyperledger) are very secure and efficient to use but works under constraints, i.e., only permissioned groups will be able to take the benefits of it, with centralized management or control [14]. The hybrid model is the combination of both, application oriented and implemented as per requirements. The benefits of blockchain models can be well understood from literature [15], but the major drawback in healthcare is, the control of medical records are with third parties and not with the actual entity to whom it belongs too, i.e., the patients. Let's understand it better, an EHR is a digital collection of a patient's medical history in terms of diagnosis, medications, treatment-plans, allergies, laboratory and test results, etc., and EMR is a digital version of the paper-chart from the clinician's, i.e., the medical and treatment history of the patients in one practice. Now, if this sensitive data managed by third party to supply for regular reviews by physician, medical institution or to use in research activities or by the insurance companies, then the privacy of patient will not be maintained. The full control on such data should be in the hand of patient and only the patient have a right to supply it at what extent and to whom. The maximum available blockchain models in healthcare are lacking in privacy on patient's medical records, quality and efficiency in processing or accessing the medical records [16, 17]. In this paper a lightweight conceptual design in order to manage the aforementioned limitations and providing full control to patients on their data with blockchain and cloud technologies is proposed. The necessary implementation will be discussed latter on by using Quorum blockchain model to address the proposed benefits. The paper's structure is, in Sect. 2 literature is provided, Sect. 3 is for proposed framework or model, Sect. 4 provide suitable comparison with existing schemes and then suitable conclusion wind up the paper.

2 Recent Studies

The involvement of IoT in healthcare industry along with challenges with traditional or centralized mechanism discussed in [8] and it briefed about how blockchain-based solution to IoT concerned services (EHR management) are more helpful and efficient. The fundamental security parameters which needs to be considered while providing solutions concerned to IoT enabled healthcare, based on blockchain are also discussed in [8]. The four major questions related to blockchain, before implementing it in healthcare addressed in [18], these questions are (a) What is the actual application of blockchain in the healthcare domain? (b) What are the primary domains in healthcare where blockchain-based strategies has been applied? (c) What are the prominent challenges for healthcare that require blockchain to provide solutions? (d) What are the future avenues in healthcare which might benefited from the applications of blockchain? In [14] fundamental architecture of blockchain technology discussed along with its categories viz. Private, Consortium, Hybrid models and benefits of each category. The major activities like EHR maintenance, monitoring patient health remotely, pharmaceutical supply chain and insurance claim management are the significant areas where the blockchain have huge demand in heathcare industry also discussed in [14]. Both the permissioned and permission-less blockchain models have its own benefits and limitations, but in [19] permissioned model proposed in order to deal with problems related to EHR management, sharing patient's EMR with multiple medical institutions by considering the scenario like patients shifting or referring from one medical institution to others. It is also claimed in [19] that permissionned models are computationally less intensive than permission-less models as PoW is a primary ingredient in permission-less implementation and also have a negative impact on the scalability and throughput of the blockchain system. The PoS, PoB are the alternatives used in permissioned implementations which are less computationally intensive. Apart from the above mentioned in permissionned models like Hyperledger each network node have an identity, and all participating nodes are issued with a certificate by the member service provider (MSP), i.e., username and password pair is issued to each network user to issue enrollment certificate, transaction certificate authority issues transaction certificates to each network user and participants of the network use transaction certificates to send transactions. The work mentioned in have certain properties viz. patient can store their information with symmetric key encryption and healthcare provider will access the information with public-key mechanism; separate user interface for patients and healthcare provider; the blockchain will store transaction data and world-sate database (CouchDB) will hold actual data. The [20] also described the benefits of permissioned model for managing the healthcare information by elaborating few crucial points like the crash fault tolerant (CFT) or byzantine fault tolerant (BFT) consensus protocol do not require costly mining; smart-contracts authored in general-purpose programming languages; only authorized users have access to the chaincode (smart-contracts) and data transacted; to manage the interoperability with Agency for Interoperation, Diffusion and Archive of Clinical Information (AIDA) by using standards viz. HL7,

and ontologies such as SNOMED, LOINC. AIDA is a multi agent system which use HL7 as gateway to communicate and supports its actions in standards like OpenEHR and SNOMED. The major benefit is data and information flow not remains restricted to single medical institute. The utilization of public-key infrastructure especially for blockchain models elaborated in [21], i.e., participants have their own pairs to be used for while initiating transactions. The one who executes the transaction use the private key against the transaction, which encrypts the data of the transaction. The person who would like to access the data from the transaction can decrypt it using the public key of the sender. If somehow any alteration captured in the record the signature straightforward will become illegal, and the transaction discarded. It is significant to get notified earlier to prevent further damage. The blockchain might not be considered as a cost-effective solution to store huge amounts of data on-chain, so that actual data needs to be store off-chain and indexes (for external data source - cloud [22]) should be store on-chain. Even though the performance of blockchain always affected by certain variables viz. block-size, network size, ordering-service, transaction-size and nodes topology in network, etc., but to overcome from these issues methodologies are available in literature, the one solution is MedBlock [23] and one is FogChain [24]. Apart from the benefits **the prime limitation with blockchain is patient does not have the right to be forgotten**, i.e., if requires the deletion of their stored health data from the blockchain,¹ as it's against the one basic property namely **immutability** [25, 26]. The one significant point which needs necessary consideration while providing blockchain-based solution in healthcare is, it requires reliable storage scheme to manage EMRs if data store off-chain, so highly secure cloud storage will be a nice option for this. Even though medical institution outsourcing cloud storage for medical repositories which seems secure, but it still have certain security and privacy issues [27]. Instead of cloud the emerging IoT technology also spreading its feet in healthcare and also utilizing features provided by blockchain [28], but IoT has some prominent challenges that require consideration viz. security and privacy issues [29], IoT standards issues, legal issues, regulatory rights issues, emerging economy issues, developmental issues, etc., [30, 31]. In [4] information about health information exchange system (HIE) with application of blockchain provided, the exchange can be Direct-Exchange, Query-Based-Exchange, Consumer-Mediated-Exchange, etc. The touch-points facilities, i.e., to collect more appropriate patient health data apart from filtering the whole available data, to make specific information available as per choice also provided in [4]. The [32] provided a novel solution for EHR management both by using the functionality of blockchain and cloud computing (mobile cloud). The purpose of this study is, the access control approaches for EHR sharing with traditional cloud servers are trusted, but the case is not true for mobile clouds since the cloud server are honest but curious, i.e., honestly perform the data requests, but meanwhile will obtain personal information without consent of users, which leads to serious information leakage issues. Another solution for sharing the EHR provided in [33] called EdgeMediChain which is based on edge and cloud com-

¹ The hospital needs to dispose off (by preserving few) the patient's records after certain period of time, i.e., 10 years.

puting with blockchain. The motivation behind EdgeMediChain is that blockchain networks suffer from trilemma, i.e., only have at most two of the following features (a) decentralization, (b) scalability and (c) security. Especially in scalability issue, in terms of low-throughput, high-latency, resource-draining and ledger-height, lowering the practicality of any blockchain-based solution on a large-scale [33]. One more significant point which requires consideration is, **the medical records are private and needs patient's ownership, control whether stored elsewhere**, but in many of the solution available in literature the medical health record management is based on centralized systems with lacks of patient's ownership [34]. It will be highly worthwhile to provide blockchain-based solutions for EHRs management, by considering aforementioned all the requirements which not only secure enough, efficient (lightweight [35]) but also giving maximum patient controls. In [36] with the help of multi-sensory applications a novel approach for patient monitoring proposed, which is beneficial for both patients and staff of intensive care units (ICU) for necessary rescue in emergency situations.

3 Conceptual Model for EHR Management System

The paramount objective of the proposed design is to implement respective EHR management and sharing system, with maximum patient control, EHR's security, authenticity with private blockchain framework, secure cloud storage with indexes on ledger and preservation of computational intensiveness with advent machine learning strategies. Figure 4 provides a brief overview on conceptual design and, the working methodology.

3.1 General Procedure to Upload and Access the EHR

1. There will be categories of authorized users in this system such as patients, consumers, medical institutions, authorities to manage the blockchain/cloud activities. Each category of user have its own credentials to login to the system with pre-defined roles. The system works with two user interfaces one for patient/medical institution and other is for consumer, to upload, access and level of access management on EHRs.
2. The patient's medical records viz. records with monitoring based on smart devices, records prepared by medical institutions uploaded either with smart device (on regular intervals, for heart-rate-variability (HRV), pulse-rate (PR), blood-pressure (BP), breath-rate (BR), etc.) or by the medical institution with the consent of patient (during or after medical services). If data is coming from smart device it reaches medical institution for verification purposes then proceed further else medical institution will do the needful on regular-interval/patient's hospital journey. The access control on EHRs will remains with patient and medical institution

only, the medical institution will not authorized to share the records with third parties without patient consent (digital consent). **Medical institution decide upto what extent, category of patient's medical history should be part of EHR due to certain conflict of interest for the sake of hospital or physician's reputation.** During the upload operation records will go through quality check with dedicated machine learning algorithms (Transformer [37, 38]), if quality is satisfactory only then the record proceed with further step else discarded. The quality can be measured with certain parameter such that, if user tried to upload the image file, it must have high resolution (i.e., must satisfied to the (to be) defined minimum threshold of pixel-per-inch (PPI)), if document (.pdf only) going to be uploaded it must matches minimum dot-per-inch (DPI) resolution value as (to be) defined.

3. After the successful quality check at application level, to preserving the confidentiality, integrity a certain security checks will be initiated on each request. The steps are:
 - Fingerprint preparation of quality data received, with hash function, to preserve the integrity.
 - Encrypt the fingerprint with private key of the patient, to preserve the authenticity of fingerprint.
 - Append the encrypted fingerprint to actual patient data following with secret key encryption, to preserve the confidentiality.
 - Send the encrypted information for further processing.
4. Just after the successful encryption of EHR, transaction initiated for blockchain with certain prerequisites. If consensus become successful (with respect to the available smart-contracts or agreements) then transactional data stored on the blockchain and immediately request for storing the actual encrypted EHR send to the cloud application with index, encrypted EHR, patient-id, etc., as parameters. Here Microsoft Azure Cloud platform (see Fig. 2) considered for building all the applications along with Azure Quorum Blockchain framework [39, 40]. One more significant point which needs necessary consideration, all the secret-key will be stored on Azure active directory for single point authentication (see Fig. 3), and by using Azure entra admin all the permission will be managed.
5. The contiguous index with corresponding patient-id along with encrypted EHR stored on cloud.
6. The EHR access request initiated at the consumer end after successful login. The consumer needs to select the necessary available EHR as per requirements (with filter operation) and make a purchase order for the same or add it in cart.
7. In the next step consumer will pay necessary amount with available payment methods, and raise a access request for the EHR. The amount here also w.r.t. the access level.
8. After successful payment, EHR access request with certain prerequisites send to the blockchain application from consumer application. The required checks made by the blockchain application and if everything founds appropriate request responded with certain parameters (index, public-key, secret-key, access-request-

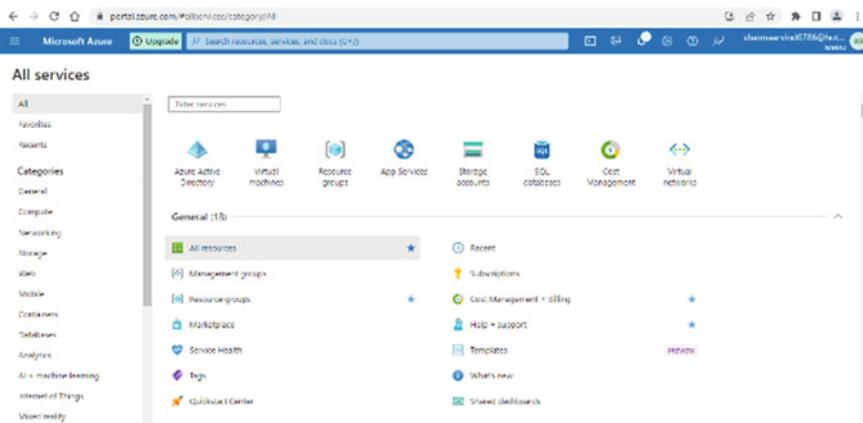


Fig. 2 Azure cloud platform

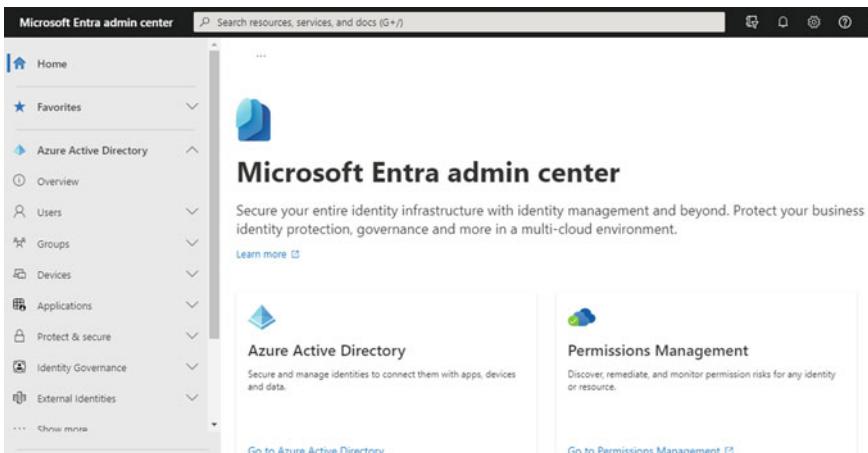


Fig. 3 Azure active directory for storing and sharing the secret key

timeout, etc.) to consumer application following will auto request to cloud application for access. The access will be provided for particular time frame, in which download operation has to be successful.

9. The record for successful or unsuccessful download also needs to be store on blockchain for exceptional cases to deals with.
10. The one significant point here is for each successful attempt for access on patient's EHR, there will be **royalty** (in native currency) needs to be credited to patients account with necessary deduction for opted application, cloud, blockchain services. **Only the medical institution which upload the data is exempted from payment** if behave as consumer in any point of time.

3.2 Quality Check on EHR with Machine Learning Approach

There are certain steps involved to ensure the quality of a documents (machine learning's model based on Transformers), to be uploaded in proposed design of electronic health records management system, which are given below in Algorithm 1.²

Algorithm 1 Quality preservation(x)

1. Data-set of defected and high quality images
 $D = (x_1, y_1), (x_2, y_2), (x_3, y_3), \dots (x_n, y_n)$
 $x' = \text{Preprocess}(x)$
 2. Defect detection, d is binary mask
 $\text{if } d = MD(x')$
 - a. Defect resolution and assess the quality with metric - PSNR or SSIM
 $x'' = MR(x', d)$
 - b. $q = \text{QualityMetric}(x, x'')$
 - c. $Q - EHR = \text{Output}(x'', q)$
 - else
 - a. $q = \text{QualityMetric}(x', x)$
 - b. $Q - EHR = \text{Output}(x, q)$
 3. End
-

All the steps mentioned involved are very significant to ensure the quality of any document consisting of patient health information, if quality matrices satisfies the minimum threshold (proposed) only then document reaches for further steps else document discarded. The process start with input image and existing data-set of defected, quality images. With pretrained ML model MD is used for defect detection in input, if defect found then suitable MR model to improve the quality of defected images is used (with possibility) then quality image produced from this model will be the output to be used further (the quality metric will be contrast, whitening, etc.), else if there is no defect found image is forwarded for next step of offloading.

3.3 Encryption on EHR with Cryptographic Public-Key and Single-Key Schemes

The patient medical records are significant part of patient's life and always required to be secure and authentic, as these can be used by the medical institutions to scrutinize the patient health, regular checkups, medical analysis, etc. If the security these records

² MD(): ML Model for defect detection, MR(): ML Model for defect resolution, PSNR: Peak signal noise ration, SSIM: Structural Similarity Index, x' : defected image, x'' : defect free resolution, Q-EHR: Quality Image with good quality score q .

violate anyhow, it may or may not have adverse impact on the image of an individual both personally and professionally, similarly if authenticity suffered somehow anticipated to severe impact on patient's life, as the unauthentic medical data 100% will not be a good choice for medical treatments. Now, in the proposed conceptual design for the possible future application, the combination of cryptographic public and single-key encryption schemes along with the suitable hash function (H) [41] to be used (see the Equation 1, * represent variable length and n represent fixed length). The procedure for encryption on EHR (prospective) is given below in algorithm 2.

$$H : [0, 1]^* \rightarrow [0, 1]^n \quad (1)$$

Algorithm 2 Encryption ($Q - EHR, K_{PR}, K_{PU}, K, H$)

The $Q - EHR$ is EHR received after quality check, K_{PR}, K_{PU} is private, public key of patient/medical-institution for public-key encryption, K is the secret key for single-key encryption, and H is a hash function.

1. *Input* : $Q - EHR, K_{PR}, K$
 2. *Fingerprint* $\leftarrow K_{PR}(H(Q - EHR))$
 3. Encrypted EHR $\leftarrow K(Q - EHR||H||Fingerprint||K_{PU}||K)$
 4. *Output* : Encrypted EHR
-

The encrypted EHR along with certain security parameter needs to be send, for further processing with blockchain and cloud applications. The working methodology of blockchain and cloud applications is elaborated in next step.

3.4 Blockchain and Cloud Applications in Conceptual Model for Access Management on EHR

The encrypted EHR reaches to the Quorum blockchain application which is also the part of Azure Cloud platform. Before transaction initiated a necessary integrity check has to happen to find integrity violation, if any. If everything found appropriate only then transaction initiated, and after achieving the mining, consensus based on pre-stored smart-contracts, the required information get stored on the blockchain. Before mining has to take place the assurance on the network fairness [42] required where all the authorized nodes are taking part in mining process. To receive the fairness of the network the creation of mining-fairness-index (FI^3) is also required (see the equation 2). Just after the working of blockchain finished, EHR send to the cloud for storage along with block's index. See the algorithm 3. The significant point here

³ The FI lies between 0 and 1, i.e., more FI tends toward zero, the fairer the mining process in the network. Here N represent nodes, i for user and x_i portion of allocated resource to x for mining.

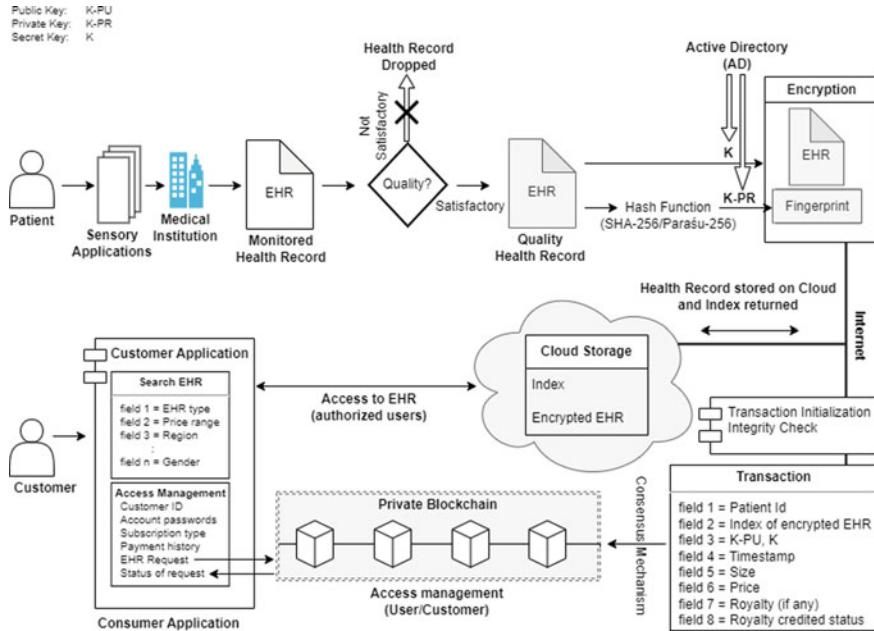


Fig. 4 Patient health record management system with patient control and private blockchain to manage the identity, access, and cloud to store the actual electronic health records

is all the operations or sub-operation will be triggered with respect to the request coming, and build as module in web applications. These module would be scalable by nature by considering different scenarios of this conceptual model.

$$FI(x_1, x_2, \dots, x_N) = \frac{\left[\sum_{i=1}^N x_i \right]^2}{\sum_{i=1}^N x_i^2} \quad (2)$$

Procedure 1: upload(Patient Id, Index, K_{PU}, K, Timestamp, Size, Price, Royalty, Royalty Status)

1. Achieve the consensus on transaction.
2. Insert new block with required information into blockchain.
3. Distribute the block information to other identical/replica blockchains.
4. Initiate the request for cloud storage to append the current information of EHRs.
5. Successful or unsuccessful status of request from (4), and update the blockchain with status.
6. Return (status of (5)) with user's application.

Algorithm 3 Access management with Blockchain and Cloud

The EHR management is divided into two parts *upload()* and *access()*.

1. Security or Integrity check
Input : Encrypted EHR : $Q - EHR||H||Fingerprint||K_{PU}||K$
2. Upload
 $if(Decrypt(K_{PU}(Fingerprint)) == H(Q - EHR))$
 $if(\text{User Category} == \text{Patient or Medical Institution} \& \text{Operation} == \text{upload})$
 Call upload()
 $endif$
 $else$
 Authenticity, Integrity violated: Request discarded
 $endif$
3. Access
 $if(\text{User Category} == \text{Consumer} \& \text{Operation} == \text{access})$
 $if(\text{Login Successful? Call access()} : \text{Request denied})$
 $endif\ endif$
4. End

Procedure 2: *access(Customer Id, Subscription type, Payment History, EHR Request, EHR Request Status)*

1. Check the status of successful or unsuccessful payment by consumer for EHR.
2. Update for status of (1) to blockchain.
3. If status implies success, provide the required credentials to consumer application for further access to cloud data, with timestamp, period of timeout, etc.
4. Update the status of (3) to blockchain after timeout.
5. Process the royalty after success at (4) for patient.

4 Comparative Analysis With Existing EHR Management Schemes

There are numerous EHR management models proposed in literature having its own benefits and limitations with respect to the scenario where it can be implemented. The majority of such models are lacking with patient control, efficient management of health records, record's interoperability⁴, secure record sharing models, etc., so by considering all such issues design proposed in this work. The suitable comparison given in Table 1 with existing schemes. The conception behind the proposed design is to build an EHR management system which overcomes existing problems of already proposed schemes. In the majority of schemes a public Ethereum blockchain framework along with PBFT was proposed, which is not that much efficient as Hyperledger

⁴ Each medical institution or hospital has its own way to maintain the records, with respect to the designed parameters, i.e., lack of standardization in record management.

fabric is with PBFT, and the involvement of the cryptocurrency (Ether) is by default. Another significant point which is considered in different schemes is the public cloud is not secure storage, so the integration of IPFS with cloud or other dedicated servers were proposed in designs. The above mentioned consideration is not appropriate at all because, the service providers which are providing cloud services definitely will not compromise with the security aspect of data and must introduce updated security policies in data-centers on regular intervals as well as dedicated staff members always available for accountability purposes. The integration of multiple storage mechanisms will degrade the efficiency, make the system computationally intensive and if cloud-based solutions consume much resources it will directly impact the financial cost for opting the computing power from service providers.

The healthcare data's offloading and access is both frequent and infrequent in different scenarios so the system has to be less computationally intensive. The prime limitation which is found in existing schemes, the exclusion of the quality aspect of EHRs in proposals. The EHR has to meet the minimum quality metric/threshold before upload operation can take place. Apart from above mixed control on patients data proposed in majority of schemes, which will not make the system patient centric at all, its patient data and only either patient or medical institution will be the controlling authority not even cloud service provider. The proposed design is following patient centric approach, i.e., either patient or authorized medical institution's will be the controlling authority on EHRs. To make the overall system less computationally intensive and secure, Azure cloud platform along with subsidiary component Quorum blockchain are used, which are not only managing the security aspect itself but also making the system efficient to use both for EHR offloading and accessing activities. However in many existing EHR schemes the component from different vendors is used, but in our proposed design each component is from a single vendor or cloud service provider which definitely helps toward efficiency aspects viz. latency, regular-updates, scalability, etc. Apart from the above to reduce the computational intensiveness, dedicated Quorum blockchain having a less complex consensus mechanism (voting based) used, and as there is no cryptocurrency involved so the design is cost-effective by nature. The quality of EHRs as discussed earlier is our prime concern too, so more advanced classification techniques from the machine learning domain namely Transformers come in rescue to deal with it. Another important question, why do patients share their medical history with third parties (consumers) apart from medical institutions and what is the benefit for them? The correct answer is that in certain research activities a huge amount of medical data always is a prime requirement of researchers for making analysis, and it's very hard to get it so researchers satisfy them with already available benchmark data. With the proposed system it would be easy for researchers to buy the required data and patients get the royalty for each transaction with necessary deduction by cloud service provider. The proposed design is easy to implement and use as compared to existing ones, i.e., both on traditional and lightweight platforms⁵.

⁵ Practical Byzantine Tolerance Algorithm (PBFT), Delegate Proof of Stake (DPOS), Proof of Stake (PoS), Proof of Work (PoW)/Proof of Authority (PoA), Mobile Edge Computing Server (MEC).

Table 1 Comparative study with significant ingredients of existing electronic health record management systems to analyze the limitations, and areas which requires necessary consideration for further improvements to make the system efficient and secure

Metric	[19]	[23]	[27]	[4]	[32]	[33]	[31]	[22]	[12]	[7]	[17]	Proposed design
Framework	Hyper ledger	Hyper ledger	Permission block-chain	Private ethereum	Ethereum	Ethereum	Ethereum	Ethereum	Hyper ledger	Ethereum	Hyper ledger	Quorum
Consensus mechanism	PBFT	PBFT	DPOS	PoS	PoW/PoA	PoS	PoS	PoS	PBFT	PoS	PoS	Quorum-chain (wong)
Storage	Couch DB	Dedicated server	Cloud	Local database	Mobile cloud and IPFS	IPFS	Cloud and hospital data center	Cloud	Blockchain	IPFS	IPFS	Cloud
Overall control	Health administrator	Service provider	Medical institution, patient and third party	Clinician and patient	Cloud EHRs manager	Medical institution, patient and service provider	Hospital and patient	Service provider	MEC server (near-by)	Hospital and third party	Hospital and patient	
Crypto currency	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	No
Record quality Assurance	No	No	No	No	No	No	No	No	No	No	No	Yes
Data privacy	No	No	Yes	No	No	Yes	No	No	No	Yes	No	Yes
Cost efficient	Yes	Yes	No	No	No	No	Yes	No	No	Yes	No	Yes
Patient centric	No	No	No	Yes	No	Yes	No	No	No	No	No	Yes

5 Conclusion and Future Work

In this work we elucidated the prime limitations of EHR management schemes, and proposed a design which will help against hindrances for efficient, cost-effective, easy-to-use, secure and less intensive solution. In the majority of EHR management schemes the patient control on his medical data is almost negligible or have very little impact, but it is the patient who can decide the level of access, to whom the access to be given, and on what cost, or the medical institution on his behalf. The patient's or authorized medical institution's full control on EHR, the quality preservation of EHR before it get offload on cloud with ingredients provided by private blockchain to preserve the security of EHR is the paramount objective of proposed design. To ensure the EHR quality more advent machine learning classification technique—Transformers introduced in this design. To make the design efficient the cloud-based solutions viz. cloud storage and blockchain with respect to the requirements will be opted from same service provider, i.e., Microsoft Azure Cloud Storage and Quorum Blockchain (with dedicated smart-contracts). To manage the security of shared secret keys in encryption/fingerprint preparation techniques Azure's Active Directory Service preferably incorporated. The ADS is capable enough to help against obscuring the shared secrets. The crucial conception here is to make the system as much as possible less computationally intensive so only required components incorporated in the design regardless of Dedicated Servers, IPFS storage, etc., so that to be implemented application can easily be run on any platform with limited bandwidth too. In near future this design will be implemented as proposed by considering the healthcare applications (currently in use), lightweight frameworks, following with a prototype implantation in M.M. Superspeciality Hospital from where we're receiving timely suggestions/inputs for this proposal. Apart from the above we're planning to incorporate one more aspect in proposed design specifically concerning to the situation when patient admitted for treatment, at that point of time how nursing staff will help for medical data offloading while treatment in process, this particular scenario we didn't find elsewhere in any scheme—the reason behind preceding conception is, the medical data while treatment in process will not only be helpful in better treatment, decision-making but also helpful in research perspective.

6 Declaration

Ethical Approval: This research does not require ethical approval.

Source of Funding: This research did not receive any specific grant from funding agencies in public, commercial, or not-for-profit sectors.

Conflict of Interest: The author have no conflict of interest to declare.

Author's Contribution: Each author contributed equally in this proposed work. More specifically AKS along with GH managed the design strategy with advent technologies like Blockchain, Cloud Computing and Machine Learning. Currently

both the above mentioned authors are working on prototype building as per the proposal with timely suggestions provided by medical practitioners. On the other hand AKS along with SW and HS also managed the medical aspect of the proposal such as healthcare data collection, analysis and periodical meetings with medical institutions, etc. All the authors have reviewed and approved the final draft and are responsible for the content and similarity index of the manuscript.

Availability of Data and Materials: Not applicable.

Acknowledgements We would like to express our sincere thank to Ms. Himani, Nursing Sister, Department of Neurosurgery, Maharishi Markandeshwar Superspeciality Hospital, Mullana (Haryana), India for supporting and making it easy to understand patient's health scenarios, record-management activities in hospitals, crucial aspects in patient journey from day to start upto final discharge, etc.

References

1. Stallings W (2010) Cryptography and network security principles. 5th Edition, PHI
2. Forouzan (2017) Data communication and networking. 4th Edition, McGraw Hill
3. Report (2016) Blockchain: a healthcare industry view. Capgemini
4. Zhuang Y et al (2020) A patient-centric health information exchange framework using blockchain technology. *IEEE J Biomed Health Inf*
5. Arvind et al (2021) Inter-networking: an elegant approach for configuring layer-2/layer-3 devices for attaining impulsive outcomes. Springer, Sustainable Communication Networks and Application, Lecture Notes on Data Engineering and Communications Technologies
6. Atlam Hany F et al (2019) Chapter one - technical aspects of blockchain and IoT. *Adv Comput* 115:1–39 Elsevier
7. Nguyen DC et al (2021) BEdgeHealth: a decentralized architecture for edge-based IoMT network using blockchain. *IEEE Internet of Things J* 8(14)
8. Tariq N et al (2020) Blockchain and smart healthcare security: a survey. In: The 6th international workshop on cyber security and digital investigation, procedia computer science. Elsevier
9. Arvind et al (2019) Cryptography and network security hash function applications, attacks and advances: a review. In: IEEE, 3rd international conference on inventive systems and control (ICISC 2019), pp 10–11 January 2019
10. Anjum HF et al (2020) Mapping research trends of blockchain technology in healthcare. *IEEE Access* 8
11. Dasaklis TK et al (2018) Blockchain meets smart health: towards next generation healthcare service. In: IEEE, 9th international conference on information, intelligence, systems and applications (IISA)
12. Ray PP et al (2021) Blockchain for IoT-based healthcare: background, consensus, platforma, and use cases. *IEEE Syst J* 15(1)
13. Kuo TT et al (2017) Blockchain distributed ledger technologis for bioimedaical and health care application. *J Am Med Inf Assoc* 24(6)
14. Ben Fekih R et al (2020) Application of blockchain technology in healthcare: a comprehensive study. In: ICOST 2020. Springer, Lecture Notes on Computer Science
15. Arbabi MS et al (2023) A survey on blockchain for healthcare: challenges, benefits, and future directions. *IEEE Commun Surv Tutorials* 25(1)
16. Siyal AA et al (2019) Applications of blockchain technology in medicine and healthcare: challenges and future perspective. MDPI, Cryptography

17. Egala BS et al (2021) Fortified-chain: a blockchain-based framework for security and privacy-assured internet of medical things with effective access control. *IEEE Internet of Things J* 8(14)
18. Tandon A et al (2020) Blockchain in healthcare: a systematic literature review, synthesizing framework and future research agenda. *Comput Indust Elsevier*
19. Usman M et al (2020) Secure electronic medical records storage and sharing using blockchain technology. In: 2019 international conference on identification, information and knowledge in the internet of things, *Procedia computer science*. Elsevier
20. Guimaraes T et al (2020) ICU Data Management - a permissioned blockchain approach. *international workshop on healthcare open data, intelligent and interoperability*. *Proc Comput Sci*
21. Sharma Y et al (2020) Preserving the privacy of electronic health records using blockchain. In: *International conference on smart sustainable intelligent computing and applications under icitem2020*, *procedia computer science*. Elsevier
22. Zheng et al (2018) Blockchain-based personal health data sharing system using cloud storage. In: 2018 IEEE 20th international conference on e-health networking, applications and services (Healthcom)
23. Maria et al (2020) MedBlock: using blockchain in health, healthcare application based on blockchain and smart contracts. In: *Proceedings of the 22nd international conference on enterprise information system 2020*. SCITEPRESS - Science and Technology Publication
24. Mayer AH et al (2021) FogChain: a fog computing architecture integrating blockchain and internet of things for personal health records. *IEEE Access*
25. Andre et al (2019) Electronic health records in a blockchain: a systematic review. *Health Inf J*. SAGE Publication
26. Koshechkin KA (2018) Scope for the application of blockchain in the public healthcare of the russian federation. In: *International conference on knowledge based and intelligent information and engineering system, KES2018*, *procedia computer science*. Elsevier
27. Chen Y et al (2018) Blockchain - based medical records secure storage and medical service framework. *J Med Syst*. Springer, LLC
28. Lakhani A et al (2023) Federated-learning based privacy preservation and fraud-enabled blockchain IoMT system for healthcare. *IEEE J Biomed Health Inf* 27(2)
29. Zarour M et al (2020) Evaluating the impact of blockchain models for secure and trustworthy electronic healthcare records. *IEEE Access*
30. Ratta P et al (2021) Application of blockchain and internet of things in healthcare and medical sector: application, challenges, and future perspectives. *J Food Qual*. Hindawi
31. Bhaskara et al (2021) Fortified-chain: a blockchain-based framework for security and privacy-assured internet of medical things with effective access control. *IEEE Internet of Things J* 8(14)
32. Nguyen DC et al (2019) Blockchain for secure EHRs sharing of mobile cloud based e-health systems. *IEEE Access*
33. Akkaoui R et al (2020) EdgeMediChain: a hybrid edge blockchain-based framework for health data exchange. *IEEE Access*
34. Moussa M et al (2020) Blockchain for giving patients control over their medical records. *IEEE Access*
35. Leila et al (2019) Lightweight blockchain for healthcare. *IEEE Access*
36. Burdick et al (2022) Improved patient monitoring with a novel multisensory smartwatch application. *J Med Syst* 46:83
37. Transformers.<https://bdtechtalks.com/2022/05/02/what-is-the-transformer/>
38. Khan S et al (2022) Transformer in vision: a survey. *ACM Comput Surv*
39. Michael Collier S (2016) Fundamentals of Azure. 2nd edition, Microsoft
40. Azure Cloud Platform. <https://azure.microsoft.com/en-in/>
41. FIPS 180-3 (2008) Secure hash standard (SHS). National Institute of Standards and Technology, US Department of Commerce, Washington D.C
42. Merrad Y et al (2022) Blockchain: consensus algorithm key performance indicators, trade-offs, current trends, common drawbacks, and novel solution proposals. *Mathematics* 10(15)