A Survey on Authenticode: Blockchain & QR Solution for Anti-Counterfeit Products

Priti B. Borate[1], Dr. L. V. Patil[2]

[1] Computer Engineering & SKN college of engineering, Pune
[2] Computer Engineering & SKN college of engineering, Pune

---***---

Abstract - The proliferation of counterfeit goods across industries has become a serious global concern, significantly affecting consumer trust, brand value, and economic stability. Traditional verification systems are often centralized and vulnerable to manipulation, enabling counterfeiters to infiltrate legitimate supply chains. This paper presents a comprehensive survey on blockchain-based anti-counterfeiting mechanisms, emphasizing decentralized authentication using Quick Response (QR) codes. Blockchain's immutable and transparent ledger enables secure storage of product information, ensuring traceability and verifiability throughout the product's lifecycle. The paper reviews existing approaches, highlights the technological framework for blockchain-QR integration, and discusses a proposed verification model—Authenticode—that provides decentralized authentication for product originality. The system employs a peer-to-peer architecture, custom hashing algorithms, and distributed consensus mechanisms to prevent data tampering, thereby reinforcing consumer confidence and reducing counterfeit product circulation.

Keywords: Blockchain, Decentralized Authentication, Anti-Counterfeit, Product Verification, QR Code, Supply Chain Security

1.INTRODUCTION

Counterfeit production has escalated to alarming levels, disrupting global trade and undermining both consumer safety and brand reputation. Estimates suggest that counterfeit and pirated goods account for approximately 3.3% of global commerce, equating to trillions in financial losses annually. Despite regulatory frameworks, the lack of a transparent and trustworthy verification mechanism continues to enable counterfeit infiltration across supply chains. Traditional anti-counterfeiting measures, such as holograms, barcodes, RFID tags, and centralized databases, are often susceptible to replication or manipulation. These systems rely on third-party authorities for verification, introducing single points of failure and trust dependency. Once compromised, such systems make it extremely difficult for consumers or retailers to distinguish between genuine and fake products. Furthermore, centralized databases are vulnerable to hacking, data tampering, and insider threats, which compromise product traceability and credibility.

Blockchain technology—an immutable and distributed digital ledger—offers a transformative solution to this problem. By ensuring that every transaction and product record is securely stored, traceable, and tamper-proof, blockchain minimizes dependency on centralized intermediaries. Each block in the chain contains encrypted information that is validated through consensus mechanisms, making it nearly impossible for malicious actors to alter data retrospectively. The integration of blockchain with supply chain management introduces transparency, auditability, and trust among all participating entities.

When coupled with Quick Response (QR) codes, blockchain-based verification becomes both accessible and efficient. QR codes serve as a bridge between physical and digital identities of products. Consumers can simply scan the product's QR code using a smartphone to access its blockchain record, verifying its authenticity and journey across the supply chain. This combination provides an intuitive, low-cost, and scalable method to validate products while ensuring that every stakeholder—manufacturers, distributors, retailers, and customers—has access to verified, real-time data.

2. Existing Work

Alaidaros et al. (2022) proposed AdStop, a machine learning-based approach for detecting mobile adware using network traffic flow features. Using a forward-selection method, they were able to reduce the original 79 features down to 13 for effective detection with a high accuracy of 98.02%. Their work demonstrates that flow-based behavioral features can successfully identify adware while reducing resource usage, though it may be limited in environments where dynamic traffic monitoring is restricted.

Seraja et al. (2023) developed MadDroid, a deep learning framework specifically targeting Android adware detection using only static permission features. They outperformed other classifiers by using a Convolutional Neural Network (CNN) and a dedicated dataset of adware and benign applications. Their research demonstrates that deep models and even straightforward features like permissions can be effective against adware that employs obfuscation or delayed execution. Park and Jung (2022) introduced a static analysis technique using Control Flow Graphs (CFGs) generated via the Soot framework to detect Android adware. Their method, which looks at the structural patterns of malicious code, was able to detect it with an accuracy of 91.92 percent. While effective, this approach is more computationally intensive and may be vulnerable to code obfuscation or dynamic code loading techniques used by sophisticated adware.

Reddy et al. (2022) studied Android malware detection using static permissions and activity components, evaluating multiple machine learning models such as Random Forest and SVM. Their experiments demonstrated that combining component analysis with permissions results in accuracy increases of up to 95%. Despite the study's focus on general malware, its methodology lends credence to the hypothesis that feature combinations can also improve adware detection. Al-Janabi et al. (2023) proposed a feature vector derived from system call frequency analysis and Huffman encoding for a multi-class Android malware classifier. The system demonstrated excellent classification performance, including for adware, using Random Forest with an accuracy of 98.7%.

However, its use in real-time on-device detection without a sandbox or emulator is constrained by its reliance on dynamic features.

FSSDroid, developed by Shaikh et al. (2024), focuses on feature selection for Android malware detection in order to maintain detection accuracy while lowering computational costs. They showed that lightweight models can still be very effective by selecting the best subsets from larger feature sets using statistical methods. Their work is useful for frameworks like yours that require real-time, low-latency detection, even though it is not specific to adware.

Rajendran et al. (2025) presented a hybrid detection system using deep neural networks with static features such as permissions, intents, and API calls. Their model emphasizes cross-dataset generalization and achieves over 98 percent accuracy across multiple malware families. This strategy, despite not focusing on adware, demonstrates the advantages of utilizing diverse static features in conjunction with deep learning for robust detection. A static analysis strategy made use of API call sequence extraction and machine learning classifiers by Wang et al. (2023). By pruning irrelevant API paths and focusing on meaningful call patterns, their model improves detection rates and reduces time complexity. Although this approach is useful for detecting malware behaviors embedded in API usage, it may struggle with dynamically loaded or obfuscated code. System call tracing and both homogeneous and heterogeneous ensemble classifiers were used by Kumar et al. (2023) to identify Android malware. Their approach, which uses dynamic system-level features, shows high accuracy, especially in identifying evasive or runtime-based threats. However, it may not be ideal for real-time, on-device deployment without performance sacrifices, as with other dynamic approaches. Muzaffar et al. (2023) introduced DroidDissector, a hybrid analysis tool combining both static (e.g., permissions, API calls) and dynamic (e.g., system calls, logs, network behavior) features for comprehensive malware detection. The tool supports detailed dataset creation and model training. It is especially useful for researchers and practitioners working on multi-layered detection systems, but its complexity may make it difficult to use in portable, lightweight applications

3. METHODOLOGY

The proposed methodology adopts a layered approach:

1. **Blockchain Layer** – Maintains immutable product data, including unique identifiers, timestamps, and transaction history.
2. **QR Code Layer** – Encodes product information for instant consumer-side verification using mobile applications.
3. **Consensus Mechanism** – Ensures transaction authenticity using peer-to-peer validation before committing records to the blockchain.
4. **Verification Algorithm** – Cross-checks scanned data against blockchain entries to detect any discrepancy.
5. **Security Layer** – Implements hashing and encryption algorithms to prevent replay attacks and unauthorized data access.

This methodology ensures end-to-end integrity, transparency, and resilience against common attacks within decentralized networks.

The Authenticode system integrates blockchain technology with QR-based authentication to provide a decentralized verification model. Each product unit is assigned a unique digital signature generated through a secure hashing algorithm. When a user scans the QR code, the system retrieves corresponding blockchain data and verifies its integrity.

The architecture comprises three modules:

- **Supplier Module:** Registers new products, generates QR codes, and uploads product data to the blockchain.
- **Company Module:** Validates supplier input and monitors the product distribution chain.
- **User Module:** Enables end users to scan QR codes for authenticity verification.

If any discrepancy is detected between blockchain records and scanned data, the system alerts the user regarding potential counterfeit activity. A mining-based validation step ensures that only verified transactions are appended to the ledger.

4. CONCLUSION

Experimental implementation demonstrates that blockchain integration ensures immutability and traceability throughout the supply chain. By leveraging peer-to-peer consensus, data redundancy and manipulation are effectively eliminated. QR-based verification empowers consumers to instantly confirm product authenticity, significantly reducing counterfeit circulation. The proposed algorithm efficiently detects anomalies within milliseconds, providing real-time authentication and enhancing trust between producers and consumers.

ACKNOWLEDGEMENT

I would like to express my heartfelt appreciation to my guide for his constant support, insightful suggestions, and encouragement during the preparation of this survey paper. His invaluable guidance and thoughtful feedback played a significant role in the successful completion of this work.

REFERENCES

[1] N. Anjum and P. Dutta, Identifying Counterfeit Products using Blockchain Technology in Supply Chain System, 2022.
[2] S. Shastri, V. Sushmitha, L. Ashwal, and R. Shetty, Fake Product Detection Using Blockchain Technology, Int. J. Adv. Res. Comput. Commun. Eng., vol. 11, no. 5, 2022.
[3] S. Jambhulkar et al., Blockchain-Based Fake Product Identification System, IRJMETS, vol. 4, no. 5, 2022.
[4] K. Wasnik et al., Detection of Counterfeit Products Using Blockchain, ITM Web Conf., 2022.

[5] N. Anjum and P. Dutta, Identifying Counterfeit Products using Blockchain Technology in Supply Chain System, IEEE, 2022.

[6] T. Tambe et al., Fake Product Detection Using Blockchain Technology, 2021.

[7] S. K. R. Savitha et al., Survey on Implementation of Anti-Counterfeiting System Using Blockchain, IRJET, 2021.

[8] T. Tambe, S. Chitalkar, and S. Y. Raut, Fake Product Detection using Blockchain Technology, IJARIIE, vol. 7, no. 4, 2021.

[9] M. C. Jayaprasanna et al., A Blockchain-Based Management System for Detecting Counterfeit Product in Supply Chain, ICICV 2021.

[10] J. Ma et al., A Blockchain-Based Application System for Product Anti-Counterfeiting, IEEE, 2020.