Frequency-Domain Watermarking–Based QR Code Authentication Against Q-Phishing[*]

Taejin Jung[1†], Yeog Kim[1], Kiwook Sohn[1‡], and Changhoon Lee[1‡]

Seoul National University of Science and Technology, Seoul, Korea
taijin090511@seoultech.ac.kr, yeogkim@gmail.com
kiwook@seoultech.ac.kr, chlee@seoultech.ac.kr

Abstract

With the proliferation of QR-code-based services, the threat of Q-phishing is increasing. This paper proposes a method that verifies the provenance and integrity of QR codes by using an RSA signature over the URL's SHA-256 hash as a watermark and embedding it into the QR code via DWT-QIM. Verification is decided solely by checking whether the value obtained from signature verification matches the SHA-256 hash of the decoded URL. In experiments on 1,000,000 URLs, the proposed method achieved a 100% verification success rate, even under distortions such as JPEG compression, Gaussian noise addition, and watermark removal attacks.

Keywords— QR code authentication, Digital watermarking, Digital Signature

1 Introduction

As QR authentication has become commonplace in services such as shared bicycles and e-scooters, a new threat—Q-phishing, in which a legitimate QR code is replaced with a malicious one to redirect users to a phishing site—has emerged [1]. Because tampering is difficult to detect by sight, provenance and integrity verification are required. DCT/DWT watermarking lacks URL binding and is thus vulnerable to code-replacement [2]. Our approach treats the URL as metadata: it uses an RSA signature of the URL's SHA-256 hash as a watermark and embeds it with Discrete Wavelet Transform (DWT) and Quantization Index Modulation (QIM), thereby cryptographically binding the URL and the block watermark-reuse attacks. In addition, it leverages the robustness of frequency-domain watermarking to common distortions such as JPEG compression and Gaussian noise [3].

2 Proposed Method

This section describes the proposed digital watermarking scheme for guaranteeing the authenticity and integrity of QR codes. The core idea is to embed, into the QR code itself, a digital signature that only a trusted issuer can generate, using it as an imperceptible watermark. We adopt RSA-2048 because verification with a small public exponent (e.g., e=65,537) is typically faster than signing, thereby reducing scan-time latency [4]. The original URL is first hashed with SHA-256 to obtain a unique digest H_{URL}, which is then digitally signed with the issuer's RSA private key to produce the signature that serves as the watermark. A standard QR code is generated from the same URL. The QR code's grayscale region is transformed into the frequency domain via a DWT, and the watermark is embedded in the selected LL subband using QIM.

[*] Proceedings of the 9th International Conference on Mobile Internet Security (MobiSec'25), Article No. P-76, December 16-18, 2025, Sapporo, Japan.

[†] First Author

[‡] Corresponding Author

Figure 2 illustrates the watermark extraction process. The scanner decodes the URL' from the captured QR code and extracts the watermark by applying a DWT to the QR code's grayscale region and demodulating the selected subband using the inverse of QIM, yielding the extracted signature. The decoded URL is hashed with SHA-256 to obtain $H'_{\mathrm{URL}'}$. Using the public key, the watermark is decrypted to H_{URL}. If $H'_{\mathrm{URL}'}$ and H_{URL} match, verification succeeds and the QR code is deemed safe; otherwise, it is rejected.

Figure 1: Watermark embedding process.

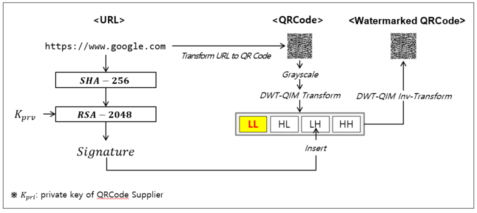
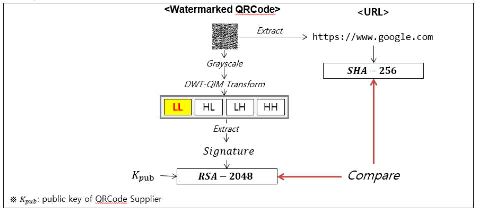


Figure 2: Watermark extraction process.



3 Experiments and Results

We evaluated the method using *The Majestic Million* dataset (1,000,000 URLs). For each URL, we generated a QR code, embedded the watermark, and then performed watermark extraction and verification. The watermarked QR codes achieved a Peak Signal-to-Noise Ratio (PSNR) of 47.27dB and a Structural Similarity Index Measure (SSIM) of 0.9972, indicating high visual quality and imperceptibility. To further assess robustness, we tested against JPEG compression, Gaussian noise, and watermark removal attacks using median blur. In all cases, the method achieved a 100% verification success rate, with the bit error rate (BER) remaining below 10%, demonstrating high resilience against common distortions.

Table 1: Experimental Results under Various Conditions

Condition	**Success Rate**	**Avg. BER**	**Avg. Time (ms)**
Ideal (No Attack)	100.00%	0.00%	20.00
JPEG Compression	100.00%	5.22%	17.63
Gaussian Noise	100.00%	0.63%	17.30
Median Blur	100.00%	5.47%	22.30

4 Conclusion

This paper proposed a QR code authentication scheme that combined digital signatures with DWT-QIM watermarking. The proposed method preserves the convenience of QR codes while adding a security layer, providing a practical solution for protected QR-code systems.

References

[1] F. Sharevski, A. Devine, E. Pieroni, and P. Jachim. (2022). "Phishing with Malicious QR Codes." In *Proceedings of the 2022 European Symposium on Usable Security (EuroUSEC 2022)*. New York, NY, USA: Association for Computing Machinery.

[2] S. Mandala, R. Rukman, and M. Irsan, "Mobile Payment Authentication using QR Codes Based on Combined DCT–DWT Digital Watermarking Scheme," in *Proc. 2023 IEEE Int. Conf. on Communication, Networks and Satellite (COMNETSAT)*, 2023, pp. 622–628, doi: 10.1109/COMNETSAT59769.2023.10420544.

[3] P. Kadian, S. M. Arora, and N. Arora. (2021). "Robust digital watermarking techniques for copyright protection of digital data: A survey." *Wireless Personal Communications*, 118, 3225–3249.

[4] D. Boneh, H. Shacham, and E. Rescorla. 2007. Authenticated encryption with RSA-OAEP. In Proceedings of the 2007 ACM Conference on Computer and Communications Security (CCS '07). Association for Computing Machinery, New York, NY, USA, 339–348.