The MARK-B.L.U. 1.0 Architecture: A Quantum Hash Function Framework for Research and Implementation of Security and Verification of Agent Identity and Outputs

Bhavyadhirr V. Bharadwaj[1] and Harshvardhan Bhosale[2]

[1, 2] GENORROW
enterprises@genorrow.com

Abstract. Quantum Computation & Artificial Intelligence have been progressing from mere theoretical promises to experimental realities. The cryptographic and encryption mechanisms, more than anything else, must thereby adapt to this expansion and scaling of AI infrastructure and the mostly underdeveloped security aspects of the said autonomous ecosystem. In this work, or odyssey of sorts, is introduced MARK-B.L.U. (Base Layer Unification) 1.0: a novel quantum security architecture; of which, in this paper we look at the hashing mechanism being utilized, which is constructed using parameterized quantum circuits, serving as the bed-rock for Secure Agent Identity and Verifiable Autonomous AI communication-output logs system. Unlike classical hash functions or existing quantum proposals that assume fault-tolerant architectures, the MARK-B.L.U. operates within the practical limitations of current hardware, balancing circuit depth, gate fidelity, and entropy spread. Presented here is also a comprehensive evaluation of the proposed mechanism's behavior under classical-to-quantum input encodings, simulated randomness profile through entropy distribution, and sensitivity analysis through collision resistance, avalanche effect, and bit independence tests. Furthermore, the research positions this architectural framework within the broader landscape of quantum cryptography by identifying the niche it fills — that of a modular, scalable, and implementation-rich security architecture, which is operable "today"; aiming to catalyze both further inquiry and implementation of quantum hashing on real autonomous ecosystems (hardware or software alike), while offering a stepping stone towards future fault-tolerant cryptographic primitives.

Keywords: Quantum Cryptography, Quantum Hashing, Cryptanalysis, Agent Identities, AI and Autonomous Infrastructure Security.

1 Introduction

1.1 The True Question

The motivating matter that must be kept at the cerebral forefront while attempting at fathoming and imbibing this research work needs to be that of "**identity**" [1-2]. In this

ever-growing and advancing world that is permeating itself with AI and autonomous infrastructure, how exactly do we keep track of who does what? The answer to that question is found in a phenomena that revolves around what is termed as identities. That is, each individual or cluster-entity being represented and expressed with a unique string value, a 'name' per se (that's by its virtue unique and distinct to it), most often; what shall be now associated loosely yet definitely as its identity. The foundational systems and spaces that, one, takes care of these actions and reactions of the entities, and two, where the existence of these very identities reside in, are the Identity Governance and Administration Environments (referred to as IGAE) [3-4].

1.2 From the Quantum Need to the Quantum Answer

The MARK-B.L.U. 1.0 responds to this with a fundamentally different philosophical disposition. Rather than treating quantum computation as a future threat to be defended against classically, it treats quantum mechanics itself as the source of security. The unpredictability embedded within the output of a parameterized quantum circuit is not algorithmic — it is *physical*. It arises from the superposition and entanglement of quantum states, the interference patterns they produce during circuit evolution, and the irreversible collapse of the wavefunction upon measurement. These are phenomena not subject to computational inversion because they are not computational in nature — they are ontological.

1.3 The Objective Scope

The architecture of MARK-B.L.U. 1.0 is a NISQ-compatible (Noisy Intermediate-Scale Quantum) quantum hashing design. It operates using statevector simulation; a noiseless classical emulation of quantum circuit behavior, rather than physical quantum hardware. This is an intentional staging decision. The purpose of the 1.0 is to:

- establish the mathematical correctness of the architecture,
- empirically validate its cryptographic properties under ideal conditions, and
- provide a reproducible open-source baseline from which hardware-deployed iterations can proceed.

The architecture makes no claim of post-quantum security in the formal complexity-theoretic sense. It rather claims information-theoretic unpredictability grounded in quantum mechanical indeterminacy; a property that does not rely on the computational limitations of an adversary but on the physical impossibility of predicting or replicating quantum measurement outcomes.

Purely classical hashing mechanisms, derive their security from computational hardness assumptions. Their one-wayness and collision resistance are predicated on the infeasibility, for classical computers, of inverting specific algebraic trapdoor constructions. This assumption is increasingly fragile in a world trending toward quantum computational advantage, where even fundamental works of the likes of Grover's algorithm reduces the effective security of a 256-bit hash to approximately 128-bit classical equivalence, and Shor's algorithm dissolves the foundations of RSA and ECC entirely. The security premise of classically-governed AI agent identity systems

is therefore not a question of *if* it will break, but of *when* — and in environments already operating within partially quantum-enabled adversarial contexts, that horizon may have already arrived for the most sensitive applications. The aspect in prime focus here, however, is more so of that of the intrinsic "variability factor" of autonomy and uncertainty that marks the growing advancement in autonomous Artificial Intelligence. The forthcoming section of background deliberation is thereby seminally crucial to fathom the actual nuance of what MARK-B.L.U. seeks to achieve, and its eventual paramount capability and need.

2 Background

2.1 The Autonomous Autonomy

Now, broadly, when the database systems came to be (due to man's increasing indulgence with cyber-spaces), there were Human Entities and Traditional Non-Human Entities that populated these data continua [5]. And the emergence of these Non-Human Identities (NHI) that were attributed to, and thus utilized to represent the Non-Human Entities did not particularly make the security dynamics as critically dire, as was the case with what came to be known as "Agent Entities" [6]. This was due to the deterministic and largely unvarying nature of NHI due to their predisposed rule-based or purely algorithmic existential predisposition. This was in complete contrast with Agent Identities, which are dynamic and adaptive entities at their core; their engagements, interactions, way of information transaction, may they be conveyances, or receivals, are complex, dynamic and largely varying [7]. Even manifesting physically, quite literally, in cases of physical agent-mechanisms (or robots), that increase the variable of the "order of uncertainty" [8]. In regard to the discussion about an entire AI infrastructure, what stands even more prominent is how Artificially Intelligent Agents aren't merely autonomous systems with autonomy, but rather a **variable degree of autonomy (V.D.A.)**.

2.2 True Transparency or True Security?

To ensure strong and efficient governance for this new mingle of agent identities and agentic systems, including early AI systems, certain baseline strategies do provide a fundamental basis; in an overview, these are:

- Unique provision, authentication, and even "**evolution**" ensured by having Unique Individual Identities (UII);
- Dynamic access policies [9-10] in accordance (or in awareness of, quite literally) the varying contextual attributes, and properties of these attributes, establishing essentially Context-Aware Access;
- Having evolving and almost abysmally inconsistent access flows, for creating ephemeral access pipelines (so what if we keep changing the identity flow itself indefinitely?);

- Restricting agents to a subset of tasks they are allowed to do, resulting in a policy of task segmentation and control isolation [11], which in turn breeds independent but explicit command chains in hierarchical infrastructures when scaled to actual, on-ground, heavy applications.
- And alas, transparent observability [12]; but then is it truly secure?

2.3 The "Classical" Was Sufficing

Classical means do, and have until now securing the identity architectures alongside mitigating the high-stake challenges; may it be the evolution of Identity Propagation Patterns (IPP) to ensure how an identity can be propagated securely across agentic flows [13]. Defining what explicitly can the "identity" do, correlating to the strategy of isolation and segmentation; introducing the system of token exchanges [14] for verification of identities; leveraging data to defining context, scope, and fandom (particular identity or sets of them) for better defining access flows [15]; utilizing APIs for token exchanges [16], and much more.

2.4 Humanly Algorithmic or Algorithmically Human?

However, with the new era of Artificially Intelligent Machines/Systems/Entities, the challenges far surpass even the yardsticks of the security mechanisms that currently are in place. Defining the access scope becomes chaotically less viable to the degree of implausibility in multi-nodal dynamic environments, like battlezones and war situations, where the sheer input nodes are undefined in their quality as well as quantity. The plausibility of impersonation of identity, skyrocket, unconfirming any transitive layer of trust. Transparent observability becomes evermore susceptible of diluting to additive layer of unwanted or unwarranted surveillance. All this ruckus and ineptness due to the very lack of definition, or an evolving one rather, of the existential nature of these systems. Are they just software? They are not [17-18]. Do we categorize them as purely persistent or purely ephemeral? [19-20] How to define the true balance of security and transparency? [21-22] And the greatest question — do we acknowledge these entities in concern as "Humanly Algorithmic"? The answer to this, in particular, tilts towards a rather definite yes [23-25]. And this is exactly where the "quantum need" arises from.

2.5 The 1.0

The solution, in part, is a dynamically evolving system for a dynamically uncertain environment. A "translucent layer" instead of transparent observability. A quantum-based hashing and encryption architecture for identity and communication "shapeshifting" for security and verification of agent-cluster identity and outputs in AI and Autonomous Infrastructure, that is all increasingly becoming native to hostile, and zero-trust environments.

This is MARK-B.L.U.; a "Base Level Unifier" par se, of layers of quantum hashing pipeline and classically dynamic identity and signature-evolution database struc-

ture. The architecture in itself leverages quantum-entropy extraction and parameterized quantum circuits to generate quantum-based identity signatures and verifiable decision (communication) logs.

In an era where classical cryptographic measures face obsolescence due to the ill-defined existential nature of AI mechanisms, and increasing operation of these autonomous agents in adversarial, and colossally uncertain environments and use-cases, more potently like defense and finance, this framework all round ensures that communications, decisions, and outputs from any agent or AI system; may it be drones, robots, cyber-physical algorithms, autonomous AI systems, and more, remain cryptographically authentic, tamper-evident, and provable quantum secured.

To further understand the technological front of the model layout, it's seminal to note the positioning of the architecture as a critical security layer for:

- Securing agent identity (and) systems;
- Tamper-proof hashing of mission-critical instructions;
- Shapeshifting and evolving "data" predisposition;
- Quantum-verifiable command chains across contested networks;
- Traceable and verifiable AI outputs in high-stake environments and applications.

The subsequent section unfolds the full mathematical and architectural detail of how this is achieved; from the SHA-512 pre-stretching of the classical input, through angle-encoded parameterized gates in a 16-qubit 6-layer brick-wall circuit, to the SHAKE-256 post-processing that yields the final 256-bit hash output. Each design choice is motivated by both theoretical and empirical considerations documented in full.

3 Methodology

Having shed light over the contextual basis for MARK-B.L.U. in the previous section (especially, the need and lacking of the current identity governance and security aspect in AI and Autonomous Infrastructure), this section shall aim particularly at detailing the actual technological architecture of the 1.0 (MARK-B.L.U. 1.0). As was thoroughly mentioned earlier, this framework is a quantum substitute for the current identity methods, which is built for eventual deployment in actual high-octane, uncertain, and zero-trust environment applications. The entire architecture has two broad components; first being the *quantum hashing mechanism* responsible for generating quantum-derived cryptographic identities, and the other being the *agent badge generation and rotation system* built upon that quantum core. The 1.0 project at large has been structured for clarity, modularity, and extensibility. The reason for the election of the said approach was quite straightforward — to keep the project portable, and bolster non-complex interpretability and developmental efficiency for further advancement inputs.

3.1 The Hashing Architectural Design

As has been discussed, the 1.0 transforms a classical input of arbitrary length into a quantum-derived, hash-like output of 32 bytes using parameterized quantum circuits. The core hashing flow proceeds primarily through three stages:

- **Input Encoding:** Where the classical byte sequences are scaled and mapped to quantum gate parameters using the input encoder module.
- **Parameterized Quantum Circuit Execution:** Here, a multi-layer quantum circuit transforms the initial state using input-dependent gates and entangling operations.
- **Measurement and Hash Extraction:** The full complex statevector is decomposed into real and imaginary amplitude streams, which are absorbed by a SHAKE-256 extractor to generate the 256-bit quantum output digest.

Each of these phases is encompassed into singular, distinct modules, which are: ***input_encoder, circuit_builder,*** and ***hash_core*** respectively. What follows is a deeper, and more thorough insight into and account of each of these processes individually. Fig.1 represents the hashing pipeline.

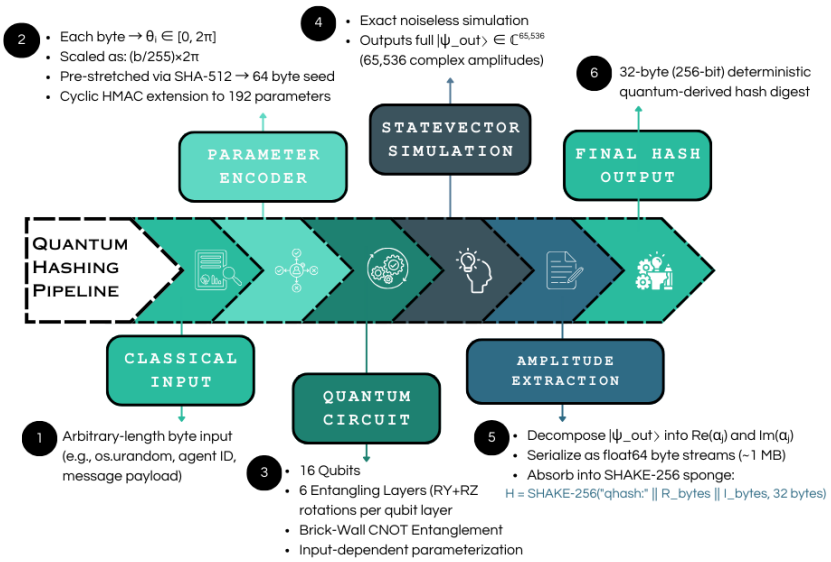

Fig.1. The MARK-B.L.U. 1.0 Quantum Hashing Pipeline. The six-stage architecture maps an arbitrary-length classical byte input through SHA-512 pre-stretching, 16-qubit 6-layer parameterized quantum circuit execution, exact statevector simulation, and SHAKE-256 amplitude extraction to produce a 256-bit quantum hash digest.

Input Encoding. The input byte sequence m is first pre-stretched via SHA-512 to yield a canonical 64-byte seed:.

$$s = \text{SHA-512}(m),\ s = (s_0, s_1, \ldots, s_{63}),\ s_i \in [0, 255] \tag{1}$$

Each byte s_i is then mapped to a rotation angle θ_i via the following linear scaling function:

$$\theta_i = \frac{s_i}{255} \cdot 2\pi,\ \ \theta_i \in [0, 2\pi] \tag{2}$$

Each resulting angle is then assigned to a corresponding rotation gate parameter (RY or RZ). The encoder ensures coverage of all 192 parameter slots (16 qubits × 2 gates × 6 layers) via cyclic HMAC-SHA-512 extension, guaranteeing that every gate in the circuit receives a unique, input-dependent angle regardless of the original input length. This cyclic encoding is performed in ***encode_input_to_params***, and the output is a dictionary data type (***dict***), mapping each ***Parameter*** object to its corresponding angle.

Circuit Construction. The quantum circuit design of the 1.0 is a 16 qubit, 6 layer parameterized one, concocted by utilizing a blend of single and dual qubit operations, which are as follows:

- **Single-Qubit Layers:** Each qubit receives a $R_Y(\theta)$ rotation followed by a $R_Z(\varphi)$ rotation; both parameterized via the input encoding system that has been previously established. The quantum register is initialised in the computational ground state:

$$|\psi_0\rangle = |0\rangle^{\otimes 16} \in (\mathbb{C})^{\otimes 16},\ \dim \mathcal{H} = 2^{16} = 65{,}536 \tag{3}$$

The Y-axis and Z-axis single-qubit rotation gates act on each qubit as follows:

$$R_Y(\theta) = \exp(-\tfrac{i\theta Y}{2}) = \begin{pmatrix} cos\frac{\theta}{2} & -sin\frac{\theta}{2} \\ sin\frac{\theta}{2} & cos\frac{\theta}{2} \end{pmatrix} \tag{4}$$

The Z-axis rotation gate introduces input-dependent complex phase:

$$R_Z(\varphi) = \exp\left(-\tfrac{i\varphi Z}{2}\right) = \begin{pmatrix} e^{-i\varphi/2} & 0 \\ 0 & e^{i\varphi/2} \end{pmatrix} \tag{5}$$

- **Entanglement Layers:** A brick-wall pattern of CNOT gates creates entanglement across the register by alternating between two sub-layers per layer; an even-pair sub-layer acting on qubit pairs (q_0,q_1), (q_2,q_3), …, (q_{14},q_{15}), and an odd-pair sub-layer acting on pairs (q_1,q_2), (q_3,q_4), …, (q_{13},q_{14}). This alternating topology ensures bidirectional information propagation across the full register within two consecutive layers. The two-qubit CNOT entanglement gate acts on a control qubit c and target qubit t as:

$$\mathrm{CNOT}|c,t\rangle = |c, t \oplus c\rangle,\ c,t \in \{0,1\} \tag{6}$$

Post this, the complete unitary for a single layer ℓ is:

$$U_\ell(\theta) = \mathrm{CNOT}_{odd} \cdot \mathrm{CNOT}_{even} \cdot \otimes_{\{q=0\}}^{\{15\}} \left[R_Z(\varphi_{q,l}) R_Y(\theta_{q,l})\right], l \in \{1, \dots, 6\} \tag{7}$$

The full six-layer circuit unitary is the ordered composition:

$$U(\boldsymbol{\theta}) = U_6(\boldsymbol{\theta}) \cdot U_5(\boldsymbol{\theta}) \cdot U_4(\boldsymbol{\theta}) \cdot U_3(\boldsymbol{\theta}) \cdot U_2(\boldsymbol{\theta}) \cdot U_1(\boldsymbol{\theta}) \tag{8}$$

Thereby, the output statevector is accordingly:

$$|\psi_{\text{out}}\rangle = U(\boldsymbol{\theta})|0\rangle^{\otimes 16} = \textstyle\sum_{j=0}^{65{,}535} \alpha_j |j\rangle,\ \alpha_j \in \mathbb{C},\ \ \sum_j |\alpha_j|^2 = 1 \tag{9}$$

Each of these layers follows the same structure; stacked in a sequence to amplify the quantum interference and state entanglement. The result thereby, is a variationally deep, and input-sensitive quantum transformation.

Core Hashing Structure. The ***hash_core*** module integrates the entirety of the aforementioned pipeline. It is done so by first initializing the 16-qubit quantum circuit via ***build_parameterized_circuit()***, after which the 192 encoded parameters are bound to their corresponding gate positions. The final statevector $|\psi_{\text{out}}\rangle \in \mathbb{C}^{65,536}$ is then generated exactly via ***Statevector.from_instructions()***. The statevector is subsequently decomposed into its real and imaginary amplitude components:

$$\alpha_j = \text{Re}(\alpha_j) + i \cdot \text{Im}(\alpha_j),\ \forall j \in \{0, \dots, 65{,}535\} \tag{10}$$

Both component arrays are serialized as IEEE 754 float64 byte streams, yielding two streams of 524,288 bytes each (~1 MB total):

$$\mathbf{R} = \text{serialize\_f64}(\text{Re}(\alpha_0), \text{Re}(\alpha_1), \dots, \text{Re}(\alpha_{65,535})) \tag{11a}$$

$$\mathbf{I} = \text{serialize\_f64}(\text{Im}(\alpha_0), \text{Im}(\alpha_1), \dots, \text{Im}(\alpha_{65,535})) \tag{11b}$$

They are then absorbed into a SHAKE-256 extractor with a domain-separation prefix:

$$H = \text{SHAKE-256}(\text{"qhash:"} \,||\, \mathbf{R} \,||\, \mathbf{I}, 32),\ H \in \{0,1\}^{256} \tag{12}$$

This yields the final 256-bit (32-byte) hash digest. The SHAKE-256 sponge construction is employed specifically because it absorbs both amplitude streams, which are correlated by the unitarity constraint $\sum|\alpha_j|^2 = 1$, into a single shared permutation state, eliminating the systematic output bias that arises while independently hashing correlated streams and combining them via XOR.

Hash Invocation. The ***main*** module acts as the main entry point for testing and demonstration. It overarchingly does so in the following manner:

- Initializes a dummy 32-byte input (for example's sake, *byte(range(32))*);
- Invokes the ***quantum_hash_function();***
- Outputs the resulting hash.

To re-emphasize, the motivation is for the architecture to be a modular interface, that is thus able to ensure that the hash function is callable as a black-box module in other pipelines, infrastructural elements, and benchmark scripts, etc.

Statistical Evaluation Framework. Even though the details of the exact and particular evaluation measures and inferences are discussed in the section subsequent, this sub-section is delegated with the task of elaborating over the particular statistical modules that were built for conducting the evaluation proceedings. To validate the cryptographic strength of the 1.0's quantum hashing capability, the following statisti-

cal evaluation code modules were created, and implemented, each collecting significantly meaningful quantitative metrics:

- ***test_entropy***: Computes Shannon's Entropy per output over 500 random samples.
- ***test_collisions***: Tests the collision resistance by comparing outputs across 1000 distinct inputs.
- ***test_avalanche***: Evaluates the sensitivity by flipping one input bit and counting the output bit changes.
- ***test_bit_independence***: Tracks per-bit flip frequency across many samples to verify the randomness spread.
- ***test_hamming_distance*:** Evaluates whether distinct inputs produce statistically independent outputs with no pairwise structural correlation.

Visual Analysis. An additional ***visualize_circuit*** module was compiled to generate the quantum circuit diagrams for the parameterized circuit of 1.0, to document gate structures. These visual additions further serve as a refinement for the empirical justification of the circuit design. Fig.2 highlights the quantum parameterized circuit construction for 1.0.

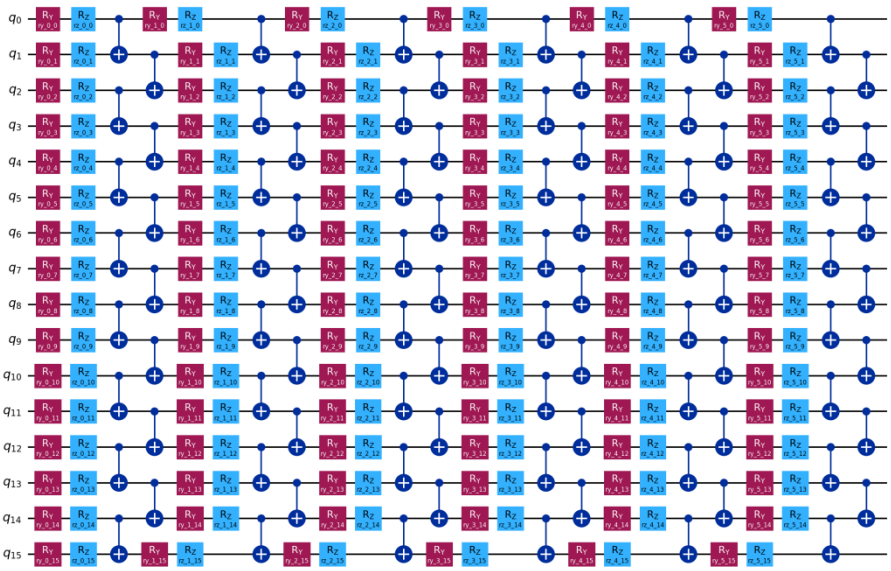

Fig.2. Full parameterized quantum circuit of the MARK-B.L.U. 1.0 quantum hashing mechanism. The circuit comprises 16 qubits (q_0–q_{15}) evolved through 6 sequential layers, each consisting of a single-qubit rotation sub-layer followed by a brick-wall two-qubit entanglement sub-layer. Crimson gates denote $R_Y(\theta)$ rotations (Y-axis); cyan gates denote $R_Z(\varphi)$ rotations (Z-axis); filled circle–ringed-plus pairs connected by vertical wires denote CNOT entanglement gates. Gate labels follow the convention 'gate_type'_'layer_index'_'qubit_index' (e.g., ry_2_5 denotes the R_Y gate on qubit q_5 in layer 2). Each rotation angle is uniquely derived from the SHA-512 pre-stretched,

cyclically HMAC-extended input encoding, yielding 192 independent input-dependent parameters across the full circuit (96 R_Y + 96 R_Z). Brick-wall entanglement alternates between even-pair sub-layers — CNOT on (q_0,q_1), (q_2,q_3), …, (q_{14},q_{15}) — and odd-pair sub-layers — CNOT on (q_1,q_2), (q_3,q_4), …, (q_{13},q_{14}) — producing 90 entangling gates in total. The full circuit depth is 24 time steps. No measurement gates are present; the complete statevector $|\psi_{out}\rangle \in \mathbb{C}^{65,536}$ is extracted coherently via statevector simulation and subsequently processed by SHAKE-256 to yield the 256-bit quantum hash digest.

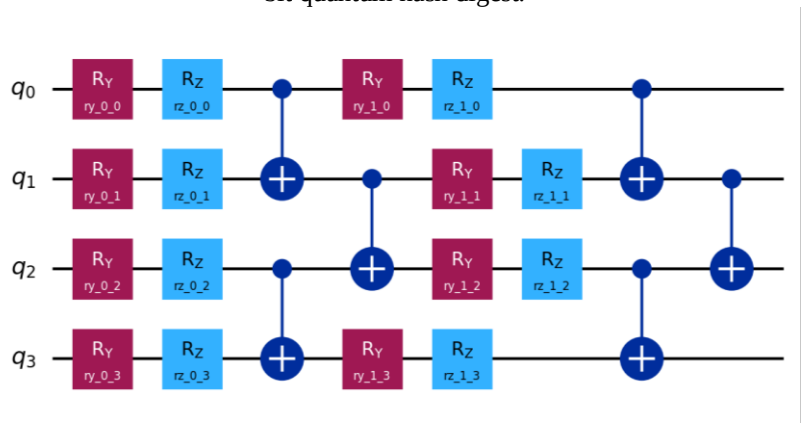

Fig.3. Architecture diagram of the MARK-B.L.U. 1.0 brick-wall parameterized quantum circuit, shown here in a pedagogically reduced 4-qubit (q_0–q_3), 2-layer form. All structural features of the full production circuit (Fig. 2) are faithfully represented at this scale. Crimson gates denote $R_Y(\theta)$ rotations (Y-axis); cyan gates denote $R_Z(\varphi)$ rotations (Z-axis). In each layer, the R_Y–R_Z sequential composition on every qubit first populates the real amplitude components of the Bloch sphere state (R_Y) and then introduces input-dependent complex phase (R_Z), together spanning the full complex amplitude structure of the Hilbert space. Brick-wall CNOT entanglement alternates between an even-pair sub-layer in Layer 0 — CNOT(q_0,q_1) and CNOT(q_2,q_3) — and an odd-pair sub-layer in Layer 1 — CNOT(q_1,q_2) — establishing bidirectional information propagation across the qubit register. The single absent CNOT connection at boundary qubit q_0 in Layer 1 illustrates the natural edge asymmetry of the brick-wall topology, which is architecturally inconsequential at the full 16-qubit, 6-layer scale. Gate labels follow the convention 'gate_type'_'layer_index'_'qubit_index'. This reduced diagram is intended as a structural companion to Fig. 2, making the circuit's architectural grammar visually legible independently of the complexity of the full 282-gate implementation.

Deployment and Reproducibility. A very thorough attempt was made to have the design structure adhere to best practices in reproducible research, with the code being open-source, and categorized in singular callable modules. The requirements are pinned in the ***requirements*** sheet. Further, as an intricate addition, the 1.0's circuit operating on less than just 16-qubits, meets the NISQ constraints, while also fulfilling its purpose as a first iterative version of the MARK-B.L.U.; counterintuitive to the

prevalent notion of higher qubit-scale. Even though that is a potent header for future developments of subsequent models of the architecture, current methodology works just well for making a more than enough of a strong case for its implementability, and conceptual potency.

Table 1 brings forth, in succinct, all the different features of MARK-B.L.U. 1.0 that have been discussed over up until this point, and which thereby make it a novel, feasible, and impactful technological model in securing AI and Autonomous Infrastructure via the utilization of a quantum hashing mechanism for securing Agent Identity.

Table 1. MARK-B.L.U. 1.0 Features' Summary.

Feature	Detailed Contribution Impact
Modular Parameterized Quantum Circuit	Fully reproducible framework that integrates parameterized gate encoding and quantum measurement-based randomness harvesting.
On-Ground Evaluative Parameters	Architectural evaluations against cryptographic metrics and empirical results on entropy distribution, avalanche effect, and output collision behavior.
Entropy Harvesting Design for Implementation	Bridging theoretical design and NISQ-compatible quantum cryptography through an open-source implementation of the framework for AI security research use.
Outlining Actual Emphasis Areas	Identification of future scoping, for use-case in securing high-stake AI infrastructure in the context of multi-nodal dynamic environments, like defense applications and financial proceedings, by including hybrid-classical improvements, random extractors, and robustness to adversarial noise.

3.2 Agent Badge Generation and Rotation System

The 1.0's agent identity layer introduces a time-variant badge mechanism, where each agent holds a quantum-derived cryptographic badge that rotates periodically. Unlike conventional static authentication schemes, this dynamically "shapeshifting" and "evolving" identity lifecycle prevents entity-trackability, thus enforcing forward secrecy and allowing for a singular, centralized verification proceeding, without exposing operational keys.

The quantum hashing core of the architecture earlier discussed, provides the entropy foundation for badge generation. This quantum badge functions as a non-reproducible identity credential rather than a classical encryption token; subsequently processed via SHA-256 for classical key derivation. The advantage with this design

strategy lies in the separating of quantum identity generation from classical message encryption; in addition to the time-variant rotation scheme, forming a "two-fold-x" (two-fold multiplied by x), i.e. a layered defense model. The diagram below outlines the agent badge generation and rotation pipeline.

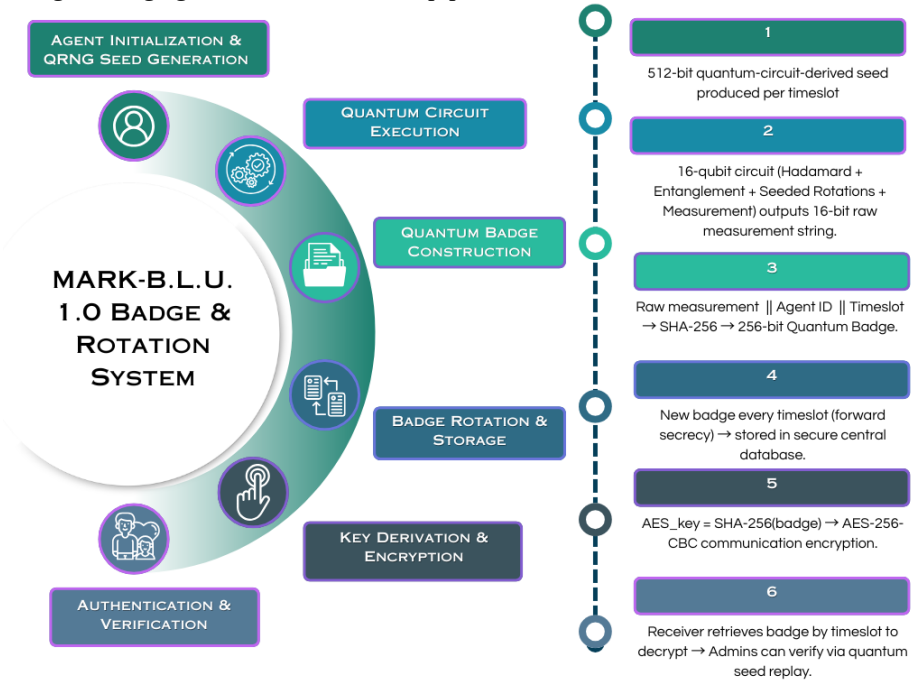

Fig.4. MARK-B.L.U. 1.0 Agent Badge Generation and Rotation System. The six-stage cyclic architecture generates a 256-bit quantum-derived agent badge per timeslot through QRNG seed generation, 16-qubit quantum circuit execution, SHA-256 badge construction, time-bound rotation with forward secrecy, AES-256-CBC key derivation and encryption, and cryptographic verification by authorised administrators.

Agent Identity and Lifecycle. Each agent is assigned a persistent serial identifier (for example's sake, AGENT-001) upon registration. The agent's serial ID remains constant, whereas its quantum badge rotates across fixed duration timeslots; with the default variant of 5 minutes.

This separation between stable identifiers and ephemeral quantum badges forms the foundation of the temporal identity masking approach. Since each badge originates from quantum measurements, its unpredictability is grounded in quantum indeterminacy rather than algorithmic randomness, yielding information-theoretic unpredictability that is unattainable through classical pseudo-random sources. The centralized database maintains:

- **identities:** Serial IDs, Timeslot Indices, Badge Values, Timestamps
- **communication_logs:** Encrypted Message Records

- **admins:** Verification-Authority Credentials

Badge Generation Protocol. Each badge derives from a 512-bit QRNG (i.e. a Quantum Random Number Generator) seed. The seed initializes a 16 qubit circuit comprising of primarily four operational layers, which are:

- **Hadamard Initialization:** All qubits are transformed to $|+\rangle$ states to create an equal superposition.
- **Entanglement:** Controlled Z (CZ) gates between adjacent qubits establish non-local correlations.
- **Seeded Rotations:** The QRNG seed-segments determine rotation angles for RZ and RX gates, introducing seed-dependent quantum phase variations.
- **Measurement and Hashing:** Qubits are measured in the computational basis; with the 16-bit measurement outcome concatenated with the agent's serial ID and timeslot index, is passed through SHA-256 to yield a 256-bit badge.

This quantum-classical hybrid procedure ensures that the quantum measurement contributes genuine physical randomness, while the classical hashing layer amplifies entropy further and embeds contextual metadata for badge uniqueness.

To again reiterate, the quantum-classical separation is very much deliberate, where the ***quantum layer*** generates non-deterministic measurement output; the ***classical layer*** expands, stores, and derives encryption keys; and the ***encryption layer*** employs AES-256 for secure message exchange.

Even if hypothetically assuming that SHA-256 were theoretically inverted, only the measured 16-bit outcome (not the originating state or seed) would be exposed, as a quantum collapse irreversibly destroys pre-measurement information.

Automated Badge Rotation. Operational time is discretized into synchronized timeslots defined by the following function:

$$timeslot = \left\lfloor \frac{(t_{\text{current}} - t_{\text{epoch}})}{t_{\text{interval}}} \right\rfloor \tag{13}$$

At each slot transition the agents autonomously detect the new timeslot, generate a fresh 512-bit QRNG seed, execute the quantum circuit, derive a new 256-bit badge, and store the badge with metadata in the central database, while purging the previous badge from local memory. Each badge generation is an independent quantum event (no persistent state is retained), ensuring a perfect forward secrecy policy and preventing correlation between consecutive identities.

Communication Encryption and Authentication. The messages of communication between agents and agent-human are encrypted using AES-256-CBC. The encryption key is derived as follows:

$$\text{Key}_{\text{AES}} = \text{SHA-256}\big(\text{badge}_{\text{quantum}}\big), \tag{14}$$

with a 128-bit random initialization vector ensuring cipher-text uniqueness.

Encryption proceeds via standard CBC chaining with PKCS#7 padding, and the transmitted cipher-text includes the initialization vector, sender ID, and timeslot metadata.

Upon receiving a message, the recipient retrieves the sender's badge for the specified timeslot from the database, derives the decryption key, and attempts decryption. Successful decryption constitutes a cryptographic proof of sender authenticity, since only the legitimate agent possessed that badge during the relevant timeslot.

This hybridization retains the classical computational efficiency while embedding quantum unpredictability at the initial identity layer, ensuring that even total knowledge of past badges yields no predictive leverage over future ones.

Administrative Verification and Database Security. Administrative authorities maintain an immutable badge record indexed by serial number and timeslot. Verification here, involves re-executing the quantum circuit with the stored QRNG seed (under statevector simulation) to confirm that the output reproduces the recorded badge, providing a holistically verifiable quantum provenance of badge creation.

The database, to be specific, incorporates and features a quantum-classical hybrid cryptographically protected admin authentication, immutable audit logs, asynchronous badge uploads for offline operation, and automatic key rotation enforcing forward secrecy.

Security Properties. The badge rotation system achieves five principal guarantees in the form of ***temporal unlinkability***, ***forward secrecy***, ***communication authentication***, ***replay prevention***, and ***quantum unpredictability***. This structure collectively resists eavesdropping, impersonation, replay, and linkage attacks, while containing potential breaches to the then current timeslot only. Table 2 outlines the aforementioned security descriptors in detail.

Table 2. MARK-B.L.U. 1.0 Security Descriptors.

Property	Description	Quantum Contribution
Temporal Unlinkability	Distinct timeslot badges prevent cross-session correlation.	Independent quantum measurements ensure zero statistical linkage.
Forward Secrecy	Compromise of current badge reveals nothing about prior ones.	Measurement collapse erases prior quantum state information.
Message Authentication	Valid decryption with timeslot badge proves message origin.	Badges are un-forgeable due to measurement irreproducibility.
Replay Prevention	Expired badges invalidate old cipher-texts.	Quantum measurement irreproducibility ensures statistical-lynon-repeating badge sequences.
Quantum Unpre-	Entropy derives from physical	Security grounded in quantum

dictability	indeterminacy.	mechanics, not computational hardness.

Furthermore, essentially summarising the hybrid quantum-classical synergy within the architecture is as follows:

- **Quantum Layer**: Badge generation via irreproducible quantum measurement.
- **Classical Layer**: Key derivation and AES based encryption.
- **Database Layer**: Secure storage and retrospective verification.
- **Application Interface**: Verification-first administrative dashboard and monitoring.

The novelty of MARK-B.L.U. 1.0 lies in employing quantum measurement outcomes as the root entropy source for autonomous AI agent identity management, yielding information-theoretic unpredictability, measurement irreversibility, and verifiable provenance while retaining classical operational practicality.

4 Evaluation

After having gone over the contextual basis and well recently, the technological basis in just the previous section, this part focuses on the evaluation of the abilities of the 1.0. As stated earlier briefly, to assess the cryptographical and statistical quality of the architecture, a series of evaluation metrics standardly utilized to benchmark cryptographic quality of hash functions were implemented. These include entropy analysis, collision detection, avalanche effect, and bit independence criterion (BIC). All experiments were conducted over multiple samples of uniformly random byte inputs across varying sample sizes per metric, using Qiskit's exact *statevector* simulator backend. Each individual metric quantitatively probes a distinct aspect of the hashing processes' reliability, unpredictability, and resilience to structural vulnerabilities.

4.1 Entropy Preservation

Shannon entropy is essentially a measure of uncertainty or specific to our use-case — information content in the hash output. For a perfectly uniform distribution over 8-bit output values, the maximum achievable Shannon entropy is 8 bits/byte, corresponding to a theoretical maximum of 256 bits for a 32-byte output.

However, the practically attainable per-sample entropy for a 32-byte output is bounded at approximately 4.9–5.0 bits/byte, since with only 32 byte observations the empirical byte-frequency distribution cannot be uniform over 256 values. The authoritative uniformity metric is therefore the pooled entropy computed over a large population of outputs.

For this research case, the entropy was computed over 100 randomly generated 32-byte input samples, and the results came out to be:

- *Sample size:* **500 randomly generated 32-byte inputs**
- *Mean per-sample Shannon entropy:* **4.884 bits/byte (σ = 0.081)**
- *Theoretical per-sample ceiling:* **~5.0 bits/byte**
- *Pooled entropy (16,000 bytes):* **7.9886/8.00 bits/byte** (99.86% of maximum)
- *Byte uniformity χ^2 statistic:* **292.58** (df = 255; critical value at p = 0.05: 293.25; null hypothesis of uniform distribution not rejected)

The per-sample entropy of 4.884 bits/byte is at the statistical ceiling for a 32-byte output, and not an indicator of entropy deficiency. The pooled entropy of 7.9886 bits/byte confirms near-uniform output byte distribution across the full sample population, with the chi-squared test failing to reject uniformity at p = 0.05. These results confirm that the SHAKE-256 post-processor effectively extracts and distributes the structural entropy of the quantum circuit across the full 256-bit output space.

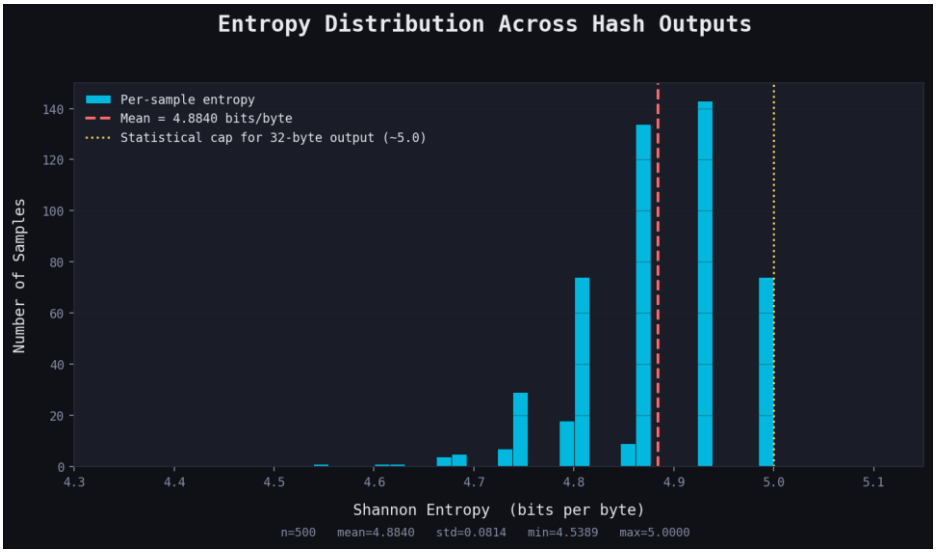

Fig.5. Per-sample Shannon entropy distribution over 500 random 32-byte inputs. The histogram shows mean entropy of 4.884 bits/byte (σ = 0.081), consistent with the theoretical per-sample ceiling of ~5.0 bits/byte for a 32-byte output. No zero-entropy samples were observed.

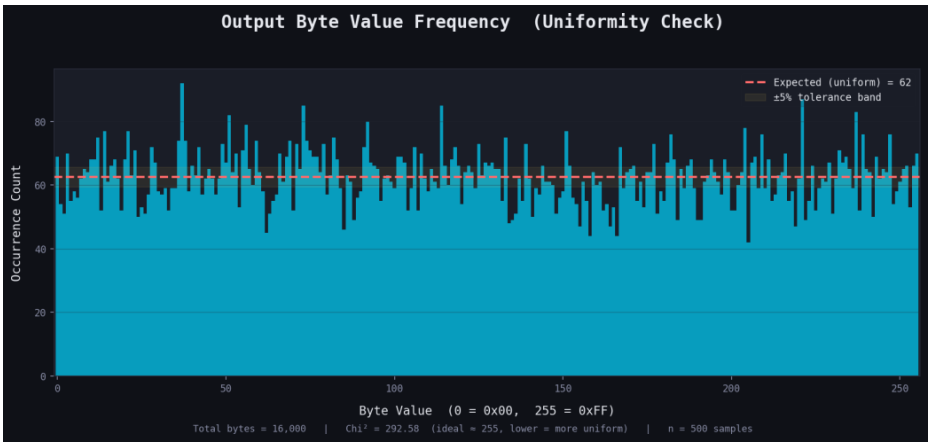

Fig.6. The Pooled byte-frequency distribution over 16,000 output bytes (500 samples × 32 bytes). The distribution exhibits near-uniform coverage across all 256 byte values, yielding a pooled Shannon entropy of 7.9886 bits/byte and a chi-squared statistic of $\chi^2 = 292.58$ (df = 255), below the rejection threshold of 293.25 at $p = 0.05$.

4.2 Collision Resistance

It basically ensures that no two different inputs hash to the same output. This was tested by hashing 1,000 distinct random 32-byte inputs and checking for duplicates. Following were the results:

- *Sample size:* **1,000 distinct random 32-byte inputs**
- *Collisions detected:* **0**
- *Collision rate:* **0%**
- *Expected collisions below the birthday bound (2^{128} for a 256-bit output):* **0**

Zero collisions across 1,000 distinct inputs is consistent with the birthday bound for a 256-bit output space, which predicts the first expected collision only after approximately $2^{128} \approx 3.4 \times 10^{38}$ inputs. The result confirms the absence of structural weaknesses, such as reduced effective output space or internal aliasing, at accessible input scales.

4.3 Avalanche Effect

A hash function is said to satisfy the avalanche criterion if a single bit-change in the input causes roughly half of the output to flip. To evaluate this, at random, a single bit in a 32-byte input was flipped, and the Hamming distance between the original and perturbed outputs was measured. The results were as follows:

- *Single-trial observed bit flips:* **127/256 bits (49.6%)**
- *Ideal target:* **128/256 bits (50.0%)**

- *Deviation from ideal:* **0.4 percentage points**
- *Heatmap evaluation (150 trials, 16 input bit positions×256 output bits):* mean flip rate **49.6%**, no structured spatial patterns detected

The observed flip rate of 49.6% deviates from the 50% ideal by only 0.4 percentage points, satisfying the strict avalanche criterion. The 150-trial heatmap confirms spatially uniform bit-flip distribution across all input-output bit position pairs, with no banding or structured bias detectable.

The result is directly attributable to the 6-layer brick-wall entanglement topology, which ensures that any single-parameter perturbation propagates through two independent entanglement paths before reaching the output statevector.

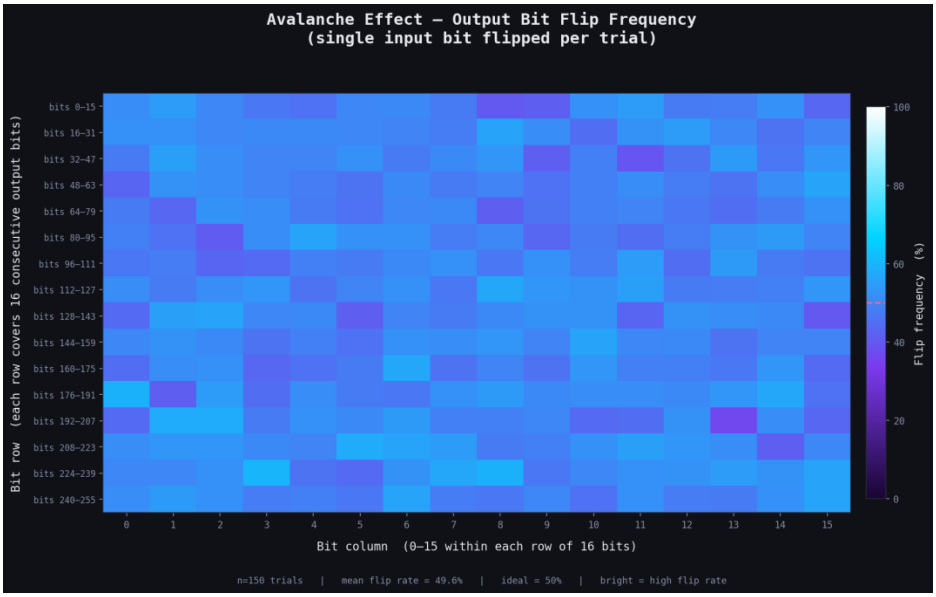

Fig.7. Avalanche effect heatmap for the MARK-B.L.U. 1.0 quantum hash function, evaluated over 150 trials across 16 input bit positions (x-axis) and 256 output bit positions (y-axis). Cell colour encodes the fraction of trials in which flipping input bit i caused output bit j to flip. Mean flip rate: 49.6%. No structured spatial patterns or banding are present, confirming satisfaction of the strict avalanche criterion.

4.4 Bit Independence Criterion (BIC)

The BIC quantifies how evenly individual output bits toggle across a broad sample of random inputs. Ideally each bit position should have a 50% probability of being 1. For testing 1.0, 1,000 hash operations were run and the deviation of each bit's activation frequency from 50% was computed. The following were the results:

- *Sample size:* **1,000 random inputs, tracking all 256 bit positions**

- *Average deviation from 50%:* **1.14 percentage points**
- *Maximum single-bit deviation:* **5.10 pp at bit position 38** (byte 04, bit 6)
- *Theoretical noise floor at n = 1,000:* $\sigma_p = \sqrt{(0.25/1000)} \approx$ **1.58 pp**

The average deviation of 1.14 pp falls below the theoretical noise floor of 1.58 pp, which is the expected statistical variance of a perfect uniform hash at n = 1,000 samples; indicating that the architecture's bit-balance is indistinguishable from ideal at this sample size. The maximum single-bit deviation of 5.10 pp at bit 38 corresponds to a z-score of approximately 3.2; in a system of 256 independent bit positions, approximately 0.33 bits are expected to exceed 3σ by chance, making this outlier statistically consistent with a structurally unbiased hash function.

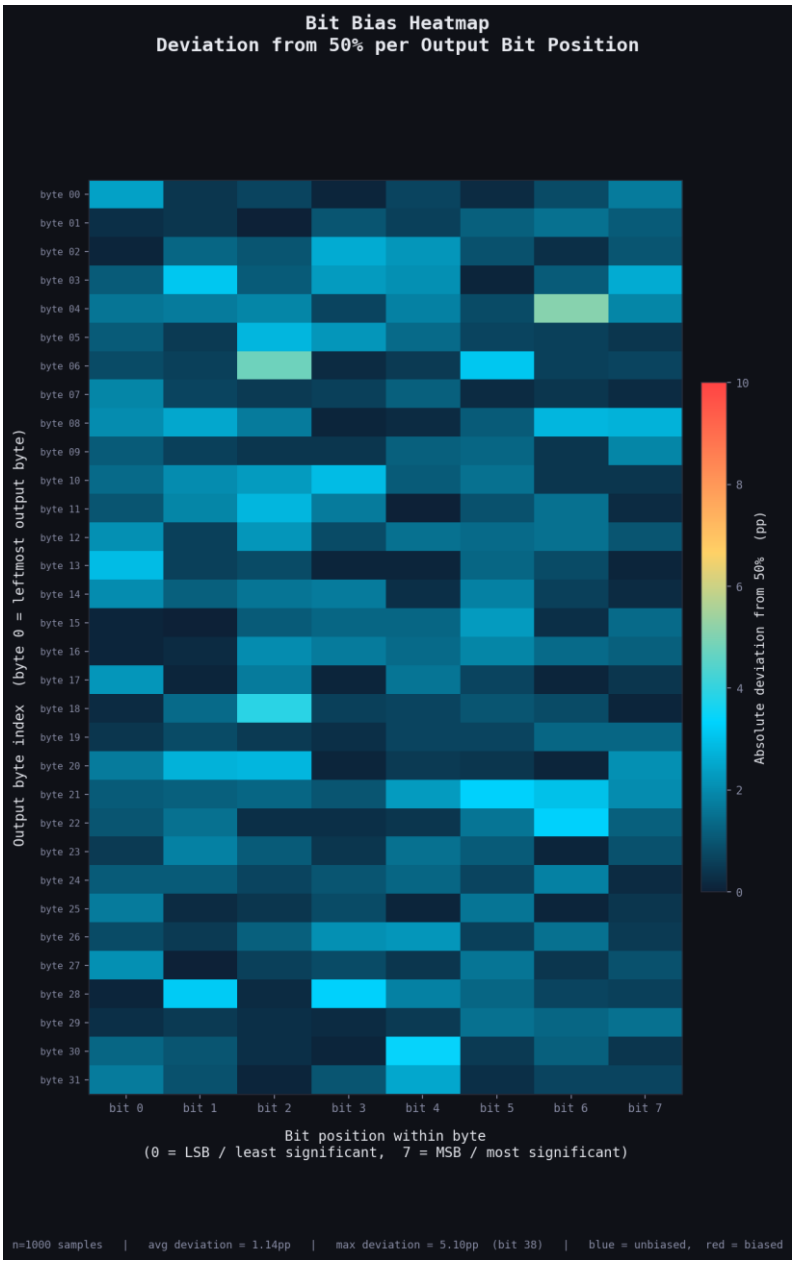

Fig.8. Bit independence criterion (BIC) heatmap for the MARK-B.L.U. 1.0 quantum hash function, evaluated over 1,000 random input samples across all 256 output bit positions (32 bytes × 8 bits). Cell colour encodes the deviation of each bit's ones-frequency from the 50% ideal. Mean deviation: 1.14 percentage points (below the theoretical noise floor of 1.58 pp at n = 1,000). Maximum deviation: 5.10 pp at bit position 38, within the expected 3σ range for 256 independent bits.

4.5 Hamming Distance Distribution

To assess pairwise output independence, an additional evaluative header was tested in the form of the Hamming distance between 300 randomly selected distinct-input pairs was computed. For a cryptographically ideal 256-bit hash function, pairwise Hamming distances should follow a binomial distribution B(256, 0.5) with mean $\mu = 128$ bits and standard deviation $\sigma = 8.0$ bits. The following were the results:

- *Sample size:* **300 randomly selected distinct-input pairs**
- *Mean Hamming distance:* **128.34 / 256 bits (50.1%)**
- *Standard deviation*: **σ = 8.46 bits** (theoretical: 8.0 bits)
- *Observed range:* **[102, 155] bits**

The mean deviation from the theoretical ideal is 0.34 bits (0.13% of total output length), and the empirical standard deviation exceeds the theoretical value by 5.75%. The distribution is visually consistent with B(256, 0.5), confirming that distinct inputs produce statistically independent outputs with no pairwise structural correlation.

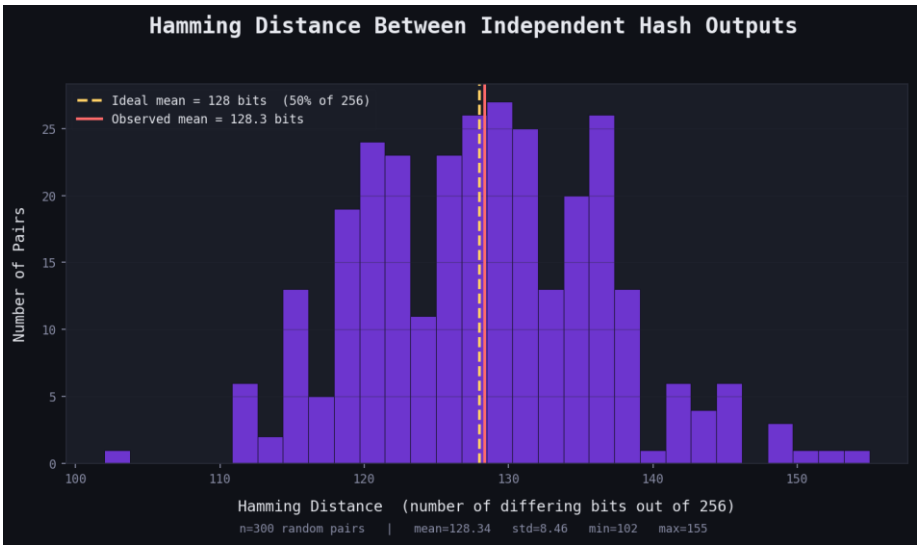

Fig.9. Pairwise Hamming distance distribution for the MARK-B.L.U. 1.0 quantum hash function, evaluated over n = 300 randomly selected distinct-input pairs. The x-axis denotes the number of differing bits between two 256-bit hash outputs; the y-axis denotes the number of pairs observing that distance. The dashed yellow vertical line marks the theoretical ideal mean of 128 bits (50% of 256), corresponding to the expected value of a binomial B(256, 0.5) distribution; the solid red vertical line marks the observed mean of 128.34 bits. Observed distribution statistics: mean = 128.34 bits, σ = 8.46 bits, min = 102 bits, max = 155 bits. The empirical distribution is visually consistent with B(256, 0.5) (theoretical σ = 8.0 bits), confirming that distinct

inputs produce statistically independent 256-bit outputs with no pairwise structural correlation.

Table 3. MARK-B.L.U. 1.0 Evaluation Comparative Analysis.

Metric	Observed Value	Ideal/Benchmark
Per-sample Shannon entropy (500 samples)	4.884 bits/byte (mean, σ = 0.081)	~5.0 bits/byte (per-sample ceiling for 32-byte output)
Pooled Shannon entropy (16,000 bytes)	7.9886 / 8.00 bits/byte (99.86%)	8.0 bits/byte
Collisions (in 1,000 runs)	0	0
Avalanche bit-flip rate (single trial)	127 / 256 bits (49.6%)	50% (strict avalanche criterion)
Avalanche heatmap mean (150 trials)	49.6%, no spatial structure	45–55%, no banding
BIC average deviation (1,000 samples)	1.14 pp	$\leq$ 1.58 pp (noise floor at n = 1,000)
BIC maximum single-bit deviation	5.10 pp at bit 38	1 outlier $\geq 3\sigma$ expected in 256 bits
Hamming distance mean (300 pairs)	128.34 / 256 bits (50.1%)	128 bits (50%)
Hamming distance std deviation	σ = 8.46 bits	Theoretical σ = 8.0 bits

5 Results and Interpretation

The 1.0, as aforementioned, was subjected to five cryptographic evaluation metrics: entropy analysis, collision resistance testing, avalanche effect, bit independence criterion (BIC), and pairwise Hamming distance distribution. The results offer actual-time quantitative insights into the very behavior and robustness of the parameterized quantum architecture.

The entropy analysis demonstrated that, on average, each output byte yielded a per-sample Shannon entropy of **4.884 bits/byte (σ = 0.081)** across 500 random input samples; consistent with the theoretical per-sample ceiling of approximately 5.0

bits/byte for a 32-byte output, and therefore at the statistical maximum attainable at this output length. The pooled entropy computed over 16,000 output bytes reached **7.9886/8.00 bits/byte (99.86% of the theoretical maximum)**, with a byte-uniformity chi-squared statistic of $\chi^2 = 292.58$ **(df = 255)**, falling below the rejection threshold of 293.25 at p = 0.05. These results confirm that the SHAKE-256 post-processor effectively extracts and distributes the structural entropy of the quantum circuit across the full 256-bit output space, with no instances of zero entropy observed.

No collisions were detected across 1,000 distinct random input trials, providing strong empirical evidence for collision resistance at the tested scale. This is consistent with the birthday bound for a 256-bit output, which predicts the first expected collision only beyond approximately 2^{128} inputs. Further, a **49.6% bit-flip rate** under a single-bit input perturbation, deviating from the 50% ideal by only 0.4 percentage points, confirms that the hash function satisfies the strict avalanche criterion. The 150-trial heatmap further confirmed spatially uniform flip distribution across all input-output bit-position pairs, with no structured banding detected, indicating that small input changes propagate widely and unpredictably across the full quantum circuit.

The bit independence test revealed an average deviation of **1.14 percentage points from the ideal 50% bit-toggling frequency** across all 256 output bit positions, falling below the theoretical noise floor of 1.58 pp for a perfect uniform hash at n = 1,000 samples. The maximum single-bit deviation was 5.10 pp at bit position 38, statistically consistent with the expected 3σ behavior for 256 independent bits.

Finally, pairwise Hamming distance evaluation over 300 randomly selected distinct-input pairs yielded a mean distance of **128.34/256 bits (50.1%)** with σ = 8.46 bits, in close agreement with the theoretical binomial B(256, 0.5) distribution. This confirms that distinct inputs produce statistically independent outputs with no pairwise structural correlation.

6 Future Scope and Prospects

The MARK-B.L.U. establishes a foundational step towards practical, dynamic, and evolving security models in agent identities and communication for the growing Artificial Intelligence and Autonomous Infrastructure. Having said that, there remain open avenues for further refinement, expansion, and integration.

First, future iterations of the model may explore ***purely measurement-based hash extraction*** in place of statevector simulation, seeking to leverage real quantum hardware backends to realise entropy from physical shot-level randomness rather than structural circuit complexity. Such an upgrade would transition the architecture's unpredictability guarantee from a computational basis to an information-theoretic one grounded in quantum mechanical indeterminacy, enhancing its cryptographic realism and aligning its behavior with stochastic quantum phenomena.

Second, the integration of ***more comprehensive hybrid classical-quantum post-processing layers***; such as Toeplitz randomness extractors or multi-round sponge

constructions, which may improve statistical uniformity and entropy concentration without compromising the quantum-first nature of the model.

Third, deeper exploration into ***quantum optimal control, variational parameter tuning, and adversarial noise modelling*** may reveal pathways to optimise circuit design for stronger cryptographic performance under real-time uncertainty and real-world environmental constraints. This includes embedding the MARK-B.L.U. circuit within verifiable quantum protocols, or as a subroutine in blockchain-based proof-of-work systems, towards the development of a broader quantum-secured agent infrastructure ecosystem.

Being clear with the objectivity of this architecture is paramount while making an attempt at advancing the technological basis. The core purpose and value of MARK-B.L.U. as a strategy and approach remains to:

- Develop quantum entropy-driven identity signatures for autonomous agents;
- Designing the foundational hashing protocols resistant to quantum attacks and adversarial spoofing;
- Prototyping a quantum-verifiable command chain for agent instruction authentication;
- Establishing output verification protocols for AI decision-making pipelines;
- Evaluating security guarantees via entropy analysis, statistical randomness tests, and adversarial robustness checks.

All in all, the MARK-B.L.U. 1.0 architecture is a foundational and fundamental stepping stone in the direction of a broader class of cryptographic quantum primitives and AI governance approaches, which are secure by design, explainable deeply by theory, and above all, executable, for mitigating the great impending need of the current time, which is, to again state, of governance, security, and thorough considerability in the forth-coming age of Autonomous AI.

7 Conclusion

Now, to summarize this entire “odyssey”, this particular work explores MARK-B.LU. (Base Layer Unification) 1.0, a fully implemented quantum hashing architecture constructed using parameterized quantum circuits, leveraging quantum-derived entropy, and designed explicitly for agent identity dynamicity and verifiable autonomous AI communication logs, thereby securing AI and autonomous infrastructure in high-stakes, uncertain, and multi-nodal dynamic environments.

The architecture comprises two components: a quantum hashing mechanism serving as the core layer responsible for generating quantum-derived cryptographic identities, and an agent badge generation and rotation system built upon that quantum core.

Demonstrated was also its effectiveness across five cryptographic benchmarks — entropy preservation, collision resistance, avalanche effect, bit independence criterion, and pairwise Hamming distance distribution; all of which support its feasibility as both a meaningful quantum cryptographic primitive and a novel approach for securing AI and autonomous ecosystems through induced agent identity dynamicity. The re-

vised architecture achieved a pooled output entropy of 7.9886/8.00 bits/byte, zero collisions across 1,000 distinct inputs, a 49.6% avalanche flip rate, a bit independence average deviation of 1.14 percentage points, below the theoretical noise floor, and a mean pairwise Hamming distance of 128.34/256 bits, collectively placing the 1.0 in quantitative competition with production-grade classical hash functions. Unlike prior theoretical constructs, MARK-B.L.U. 1.0 bridges the gap between the persistent lag in AI governance and thorough technological execution, offering an interpretable, reproducible, and modular framework for further research and development in this landscape.

As quantum devices continue to mature towards fault-tolerant operation, architectural frameworks of the nature of MARK-B.L.U. 1.0 will serve as critical testbeds for studying, designing, evaluating, and deploying native quantum cryptographic protocols in real-world systems, and for safeguarding the increasingly intermingled fabric of a Human-AI society.

References

1. Seaborn, K.: Social Identity in Human-Agent Interaction: A Primer. ACM Transactions on Human-Robot Interaction, 1–22 (2025).
2. Leonardi, P.M.: Homo agenticus in the age of agentic AI: Agency loops, power displacement, and the circulation of responsibility. Information and Organization 35, 100582 (2025).
3. Chaffer, T.J.: Know Your Agent: Governing AI Identity on the Agentic Web. McGill University (2025)..
4. Chan, A., Wei, K., Huang, S., Rajkumar, N., Perrier, E., Lazart, S., Hadfield, G.K., Anderljung, M.: Infrastructure for AI Agents. arXiv preprint arXiv:2501.10114v3 (2025).
5. Gutoreva, A.: Sharing Identity with AI Systems: A Comprehensive Review. Procedia Computer Science 231, 759–764 (2024).
6. Holldack, F., Banh, L., Strobel, G.: Agentic information systems. Electronic Markets 36(5), 1–15 (2026).
7. South, T. (ed.): Identity Management for Agentic AI: The new frontier of authorization, authentication, and security for an AI agent world. OpenID Foundation Whitepaper (2025).
8. Garg, V.: Designing the Mind: How Agentic Frameworks Are Shaping the Future of AI Behavior. Journal of Computer Science and Technology Studies 7(5), 182–193 (2025).
9. Uddin, M., Islam, S., Al-Nemrat, A.: A Dynamic Access Control Model Using Authorising Workflow and Task-Role-Based Access Control. IEEE Access 7, 166676–166689 (2019).
10. Alsagheer, D., Xu, L., Shi, W.: Decentralized Machine Learning Governance: Overview, Opportunities, and Challenges. IEEE Access 11, 96718–96734 (2023).
11. Baligodugula, V.V., Ghimire, A., Amsaad, F.: An Overview of Secure Network Segmentation in Connected IIoT Environments. Computing & AI Connect 1, 2024.0004 (2024).
12. Mu, C., Pang, J.: On Observability Analysis in Multiagent Systems. In: Gal, K., et al. (eds.) Proceedings of the 26th European Conference on Artificial Intelligence (ECAI 2023), pp. 1755–1762. IOS Press (2023).
13. Kim, J.: Structural Risks of Applying Symbolic Persona Coding (SPC) to Agentic AI Architectures. Independent Research (2025).

14. Mafrur, R.: AI-Based Crypto Tokens: The Illusion of Decentralized AI? IET Blockchain (2025).
15. Soni, A.K., Kumar, R.: Bridging the Gap: Improving Agentic AI with Strong and Safe Data Practices. Journal of Intelligent Learning Systems and Applications 17(4), 257–266 (2025).
16. Tupe, V.: AI Agentic workflows and Enterprise APIs: Adapting API architectures for the age of AI agents. Equinix Whitepaper (2025).
17. McCarthy, J.: From here to human-level AI. Artificial Intelligence 171(18), 1174–1182 (2007).
18. Korteling, J.E., van de Boer-Visschedijk, G.C., Blankendaal, R.A.M., Boonekamp, R.C., Eikelboom, A.R.: Human- versus Artificial Intelligence. Frontiers in Artificial Intelligence 4, 622364 (2021).
19. Thakran, U.: Designed Mortality: The Ethics of Terminating Ephemeral AI Agents. Independent Research (2026).
20. Zhou, J.C.: In the Collapsing Language: AI sees through the ephemeral emergence of absence. In: Proceedings of EVA London 2024, pp. 384–391. BCS, The Chartered Institute for IT (2024).
21. Mia, L.: Evaluating the Trade-offs Between Explainability and Security in AI-Powered Cyber Defense. SSRN Electronic Journal, 1–12 (2020).
22. Capuano, N., Fenza, G., Loia, V., Stanzione, C.: Balancing Transparency and Risk: An Overview of the Security and Privacy Risks of Open-Source Machine Learning Models. In: Steffen, B. (ed.) Bridging the Gap Between AI and Reality. AISoLA 2023. Lecture Notes in Computer Science, vol. 14129, pp. 284–292. Springer, Cham (2025).
23. Leslie, D.: Understanding artificial intelligence ethics and safety: A guide for the responsible design and implementation of AI systems in the public sector. The Alan Turing Institute (2019).
24. Scatiggio, V.: Tackling the issue of bias in Artificial Intelligence to design AI-driven fair and inclusive service systems. Master's Thesis, Politecnico di Milano (2022).
25. Bringsjord, S.: Toward Non-Algorithmic AI. Rensselaer Polytechnic Institute (1991).
26. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. 10th Anniversary edn. Cambridge University Press, Cambridge (2010).
27. Bharadwaj, Bhavyadhirr: The MARK-1 Architecture: A Quantum Hash Function Framework for NISQ Research and Implementation, GitHub Repository (https://github.com/Bhavyadhirr/MARK-1).

Appendix

This research is accompanied by a complete open-source implementation of MARK-BLU 1.0. All the code, including the parameterized circuit design, input encoder, analysis scripts, and visualization modules, is available at the GitHub repository: https://github.com/GENORROW/MARK-B.L.U.

This artifact outlines the:

- ***Core Quantum Hashing Mechanism***
- ***Agent Badge Generation and Rotation System***

The artifact is self-contained, reproducible, and open-source. No proprietary data or hardware is required to execute the hash function. The artifact is the intellectual and technological property of GENORROW ENTERPRISES.

Authors Credit Statement

This credits of this research work are attributed exhaustively to Bhavyadhirr V. Bharadwaj and Harshvardhan Bhosale, solely on behalf of GENORROW ENTERPRISES; details of the contributions are as follows:

- ***Bhavyadhirr V. Bharadwaj***: Conception; Building of the Core Quantum Hashing Mechanism; Research Drafting; All-Round R&D; Project Lead.

- ***Harshvardhan Bhosale***: Addition of the Agent Badge Rotation Mechanism; First draft manuscript of section 2.2.

The entirety of the research work is the intellectual and technological establishment of GENORROW ENTERPRISES.