

Simulation Analysis

A Zero-Trust QKD System Based on Trust Matrix

Yancheng Zhou

1 Introduction

This project develops a Python-based simulation framework to evaluate a **Zero-Trust Quantum Key Distribution (ZT-QKD)** network. The model reproduces dynamic multipath key transmission under adversarial conditions through sequential phases of path selection, trust update, and security evaluation.

A random network topology is generated as an adjacency matrix, where each node represents a relay with a dynamic trust value. Two matrices define the system state: a **trust matrix** indicating node reliability and an **attack matrix** marking nodes compromised by the adversary. In each iteration, the simulator constructs multiple transmission paths using a modified Dijkstra-based algorithm with a correlation factor λ_{cor} to ensure route diversity. Node trust values are then adjusted through reward, penalty, and recovery rules while simulating probabilistic attack transfers between connected nodes. Finally, the model evaluates network resilience by computing the **Attack Success Rate (ASR)**, determined by the number of compromised nodes and the minimum trust along each path.

By integrating spatiotemporal diversification and moving-target defense, the simulation quantifies how adaptive trust management and dynamic routing enhance the robustness of QKD networks against evolving threats, providing a practical benchmark for comparing traditional and zero-trust multipath schemes.

2 Methodology

The simulation framework is implemented in Python to emulate the dynamic behavior of a Zero-Trust Quantum Key Distribution (ZT-QKD) network. The model follows a modular design consisting of five computational layers: network initialization, path selection, trust dynamics, attack propagation, and security evaluation. Each component is parameterized by matrices, vectors, and coefficients that together describe the spatiotemporal evolution of trust and compromise across the network.

2.1 Network Initialization

The network topology is represented by an adjacency matrix $A \in \{0, 1\}^{N \times N}$, where $A_{ij} = 1$ indicates a bidirectional link between node i and node j . A random graph is generated with

a fixed connection probability to ensure both the source node (n_0) and destination node (n_{N-1}) are connected.

A diagonal **attack matrix** M is used to mark compromised nodes, with $M_{ii} = 1$ denoting that node i has been occupied by the adversary (Eve). The proportion of such nodes defines the **attack pervasiveness (AP)**.

A **trust matrix** T is initialized from the same topology, where diagonal entries T_{ii} represent the initial trust value of each node, typically close to one, and non-diagonal entries inherit the connectivity from A . This matrix evolves dynamically over time to reflect node behavior and recovery.

2.2 Path Selection and Correlation Control

At each simulation epoch, the framework constructs a weighted graph using the current trust matrix T . Each edge weight is computed as the negative logarithm of the average trust between two connected nodes:

$$w_{ij} = -\ln\left(\frac{T_{ii} + T_{jj}}{2} + \epsilon\right),$$

where ϵ prevents divergence for high trust values.

A modified Dijkstra-based search is used to enumerate feasible paths between the source and destination nodes. The multiplicative path security is defined as the product of the trust values of all nodes on the path:

$$P(S_i) = \prod_{j \in S_i} T_{jj}.$$

To avoid selecting overlapping routes, the **path correlation** between two paths S_i and S_j is quantified as

$$\text{Cor}(S_i, S_j) = \frac{|E_i \cap E_j|}{|E_i \cup E_j|},$$

where E_i is the set of edges in path S_i . The average correlation among candidate paths determines a dynamic threshold $\text{TH}_{\text{Cor}} = \lambda_{\text{cor}} \cdot \overline{\text{Cor}}$, which limits redundancy and enforces spatial diversity.

2.3 Trust Dynamics and Attack Propagation

During each iteration, the model updates node trust values according to local events. For node i at time t , the trust update rule is given by

$$T_{ii}(t+1) = \begin{cases} T_{ii}(t) + \beta_r(1 - T_{ii}(t)), & \text{if rewarded,} \\ T_{ii}(t) - \beta_p T_{ii}(t), & \text{if penalized,} \\ T_{ii}(t) + \gamma_t(1 - T_{ii}(t)), & \text{time-based recovery.} \end{cases}$$

Here, β_r , β_p , and γ_t are the reward, penalty, and temporal recovery coefficients, respectively. Attack mobility is simulated through the **Eve transfer factor** $f_{\text{Eve.transfer}}$, which controls the probability that an existing compromised node transmits the attack to a neighboring node in the topology matrix A .

2.4 Security Evaluation

After updating trust and attack states, the simulator evaluates whether a given transmission round is compromised. For each selected path, the model checks two criteria:

$$(i) \ n_{\text{occ}} \geq n_{\text{th}}, \quad (ii) \ \min(T_{jj}) < T_{\text{th}},$$

where n_{occ} is the number of occupied nodes on the path, n_{th} is the occupation threshold, and T_{th} is the trust threshold. If either condition holds, the transmission is marked as an **attack success**. Across all iterations, the **Attack Success Rate (ASR)** is calculated as

$$\text{ASR} = \frac{N_{\text{success}}}{N_{\text{total}}},$$

representing the probability that the adversary successfully compromises the network during multipath key exchange.

2.5 Parameter Configuration

All parameters are user-defined to enable sensitivity analysis. The attack pervasiveness (AP) sets the initial compromise level; λ_{cor} controls route diversity; β_r , β_p , and γ_t govern trust dynamics; and $f_{\text{Eve_transfer}}$ determines spatial propagation intensity. By varying these coefficients, the framework can reproduce different zero-trust network behaviors and compare them quantitatively with traditional multipath QKD schemes.

3 Data

This section presents the parameters and numerical results obtained from the Python-based simulation of the Zero-Trust Quantum Key Distribution (ZT-QKD) network. The parameters define the structural configuration and behavioral coefficients of the model, while the results illustrate how the Attack Success Rate (ASR) evolves with different attack pervasiveness (AP) and numbers of multipath transmissions.

3.1 Simulation Parameters

Table 1 lists the fixed parameters used in the simulation. The network consists of ten nodes with a random connectivity probability of 0.4. The system evolves through 1000 iterations after a 100-step thermalization phase. Trust dynamics are governed by the reward (β_r), penalty (β_p), and recovery (γ_t) coefficients, while λ_{cor} controls path correlation and $f_{\text{Eve_transfer}}$ determines the likelihood of attack transfer between neighboring nodes. The trust and occupation thresholds define the conditions under which a transmission is considered compromised.

Table 1: Simulation parameters used in the ZT-QKD model.

Parameter	Symbol / Function	Value / Range
Number of nodes	N	10
Connection probability	p_{conn}	0.4
Number of candidate paths	n_{paths}	8
Correlation coefficient	λ_{cor}	0.5
Reward coefficient	β_r	0.1
Penalty coefficient	β_p	0.3
Recovery coefficient	γ_t	0.005
Eve transfer factor	$f_{\text{Eve_transfer}}$	0.03
Occupation threshold	n_{th}	1
Trust threshold	T_{th}	0.7
Number of iterations	N_{iter}	10^3
Thermalization steps	N_{therm}	100
Multipath range	n_{multi}	2–6 (step 2)
Attack pervasiveness range	AP	0.0–0.8 (step 0.2)

Table 2: Attack Success Rate (ASR) under different attack pervasiveness (AP) and multipath configurations.

Number of Multipaths	AP	ASR
2	0.40	0.803
2	0.60	0.968
2	0.80	1.000
4	0.00	0.000
4	0.20	0.388
4	0.40	0.753
4	0.60	0.935
4	0.80	0.991
6	0.00	0.000
6	0.20	0.476
6	0.40	0.736
6	0.60	0.909
6	0.80	0.996

3.2 Simulation Results

Table 2 reports the Attack Success Rate (ASR) measured under varying attack pervasiveness (AP) and numbers of active multipaths. Each value represents the proportion of compromised transmissions during steady-state iterations.

As shown in Table 2, the ASR increases monotonically with higher AP values, reflecting the rising proportion of compromised nodes in the network. At low attack levels ($AP \leq 0.2$), the number of active transmission paths produces only a minor improvement in overall

security, since most nodes remain unoccupied and redundant routes rarely differ in exposure. However, as AP increases, the influence of path diversity becomes progressively significant. Configurations with more multipaths exhibit a slower growth in ASR, indicating that distributing key exchange across several disjoint routes enhances resilience against moderate to high pervasiveness. When AP approaches 0.8, however, nearly all nodes are compromised, and the advantage of additional paths diminishes, resulting in ASR values converging toward one. These results demonstrate that Zero-Trust multipath routing effectively mitigates risk under medium-level attacks but cannot fully prevent network compromise under extreme conditions.

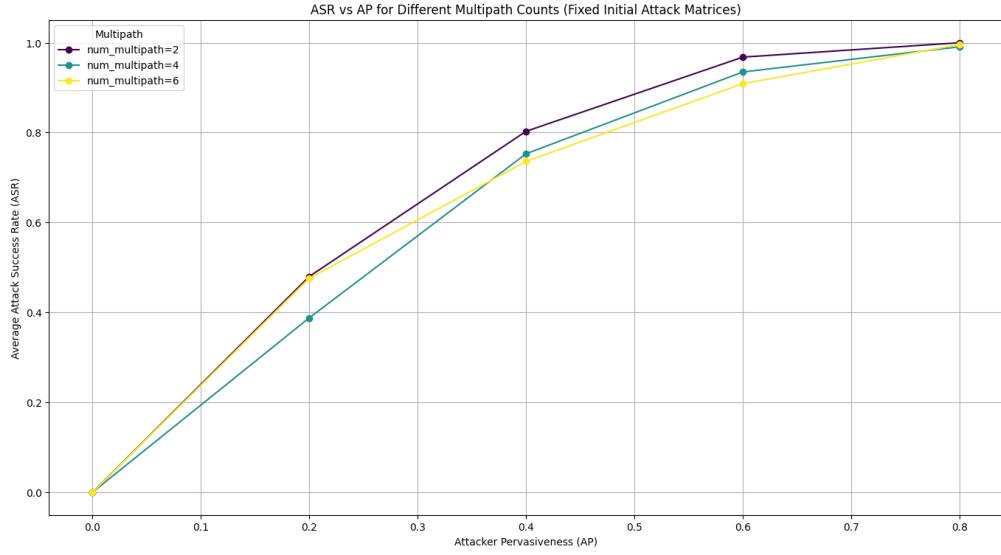


Figure 1: ASR vs. AP (2, 4, 6 paths are used as multipath QKD setting)

Figure 1 visualizes the relationship between attack pervasiveness and system vulnerability. The curves demonstrate that ASR increases almost linearly with AP at low multipath levels, while configurations with more active routes exhibit a delayed onset of failure. This trend highlights the protective effect of multipath diversification, which distributes risk across independent channels and slows the propagation of attacks through the network.

4 Future Work

Although the current simulation effectively models the dynamic behavior of a Zero-Trust Multipath QKD network, several extensions are envisioned to enhance its adaptability and theoretical alignment with the reference framework.

4.1 Dynamic Path Correlation Control

In the present model, the correlation factor λ_{cor} remains static throughout all iterations. Future work will transform this constant into a **self-adaptive parameter** that dynamically

adjusts according to the average real-time correlation among candidate paths. This modification would allow the routing algorithm to automatically respond to fluctuations in path overlap and network stability, improving the realism of the multipath diversification process.

4.2 Neighbor-Aware Trust Updating

Currently, trust evolution depends solely on each node's own observable behavior. Future implementations will incorporate the **observables of neighboring nodes** into the trust update mechanism, enabling the trust of one node to be influenced by the reliability or compromise state of adjacent nodes. Such neighbor-coupled trust adaptation would better capture the interdependence of nodes in realistic quantum communication networks.

4.3 Enhancing Eve's Ability

The current simulation simplifies the attack propagation process by allowing Eve's influence to transfer to only one node per iteration. As a future enhancement, a **probabilistic multi-node propagation model** could be introduced to simulate correlated attacks across spatially adjacent clusters. This extension would provide a more accurate representation of distributed adversarial behavior and enable the study of cascading compromise phenomena in large-scale QKD networks.

5 References

1. Chen, L.-Q., Chen, J.-Q., Chen, Q.-Y., & Zhao, Y.-L. (2023). A quantum key distribution routing scheme for hybrid-trusted QKD network system. *Quantum Information Processing*, 22(75). <https://doi.org/10.1007/s11128-022-03825-x>
2. Conrad, A., Isaac, S., Cochran, R., Sanchez-Rosales, D., Wilens, B., Gutha, A., Rezaei, T., Gauthier, D. J., & Kwiat, P. (2021). Drone-based quantum key distribution: QKD. *Proceedings of SPIE*, 11678, 116780X. <https://doi.org/10.1117/12.2582376>
3. Ghourab, E. M., Azab, M., & Graanin, D. (2025). A quantum key distribution routing scheme for a zero-trust QKD network system: A moving target defense approach. *Big Data and Cognitive Computing*, 9(76). <https://doi.org/10.3390/bdcc9040076>
4. Gisin, N., Fasel, S., Kraus, B., Zbinden, H., & Ribordy, G. (2006). Trojan-horse attacks on quantum-key-distribution systems. *Physical Review A*, 73(2), 022320. <https://doi.org/10.1103/PhysRevA.73.022320>
5. Han, Q., Yu, L., Zheng, W., Cheng, N., & Niu, X. (2014). A novel QKD network routing algorithm based on optical-path-switching. *Journal of Information Hiding and Multimedia Signal Processing*, 5(1), 13–19.
6. Lin, J., Jiang, Q., Zhang, W., Lin, Z., & Du, X. (2024). Quantum-enhanced zero trust security: Evolution, implementation, and application. In *2024 International Conference on Quantum Communications, Networking, and Computing (QCNC)*.
7. Suhail, M., & Kaif, M. (2023). Quantum hacking: Challenges and countermeasures. *International Journal for Multidisciplinary Research*, 5(5), 1–12. E-ISSN: 2582-2160

8. Zhou, Y. (2025). *Notes on Simulation Script for Zero-Trust QKD*. Unpublished technical notes.