Risk-based Zero Trust Scale for Tactical Edge Network Environments

Saketh Poduvu, Sayed M. Saghaian N. E., Ekincan Ufuktepe, Alicia Esquivel Morel, Prasad Calyam
Department of Electrical Engineering and Computer Science, University of Missouri-Columbia
Email: {spf94, ssddd, euh46, ace6qv, calyamp}@missouri.edu

***Abstract*—Tactical edge network environments are critical to deploy applications in e.g., military, disaster response, and industrial manufacturing environments. Given the Denied, Disrupted, Intermittent, and Limited Impact (DDIL) nature of these environments, a resource-aware security approach is essential to address edge resource constraints and enable real-time decision-making. The Zero Trust (ZT) security paradigm can be used to enable strict access controls, continuous entity verification, and mitigation of unauthorized access, tampering, and data integrity issues. However, there is a need to transform ZT security principles that are typically developed for unconstrained data center environments with reliable networking and abundant computing power and are not suitable in a tactical edge network setting. In this paper, we propose a risk-based ZT scale approach that tailors security measures to scenario-associated risk levels, while having low resource overheads. Specifically, we devise a Bayesian Network (BN) model to evaluate communication request risk based on metrics indicating possible attacks. In addition, we formulate a ZT metric based on the evaluated risk, environmental constraints, and entity attributes resulting in an assigned grade reflecting these factors. Our performance evaluation methodology encompasses an object-oriented drone simulation involving the ZT metric used across diverse scenarios in a collaborative drone system. Our results demonstrate the effectiveness and adaptability of our risk-based ZT scale approach in ensuring secure and efficient operations within dynamic and resource-constrained tactical edge network environments.**

***Index Terms*—Zero Trust, Collaborative Drone Systems, Bayesian Networks, DDIL, Criticality**

I. Introduction

Tactical Edge Networks (TENs) [1], [2] hold immense significance across various critical applications, thanks to the advancement of 5G technology and innovations like Low Earth Orbit (LEO) satellite connectivity. These networks serve as the backbone for real-time decision-making and communication in complex and rapidly changing environments. In Collaborative Drone Systems (CDS), they facilitate remote control and data transfer, while in special operations, they enable effective coordination and situational awareness.

However, these networks operate within the constraints of Denied, Disrupted, Intermittent, and Limited Impact (DDIL) environments. These constraints encompass challenges such as low bandwidth, intermittent connectivity, and restricted power resources. These limitations underscore the pressing need for innovative solutions that can ensure consistent communication and streamlined operations, effectively addressing the unique resource challenges within TEN environments.

This material is based upon work supported by the National Science Foundation (NSF) under Award Numbers CNS-2243589, CNS-2243619 and CNS-1647182, and the National Security Agency (NSA) under Award Number H98230-21-1-0260. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the NSF, or NSA.

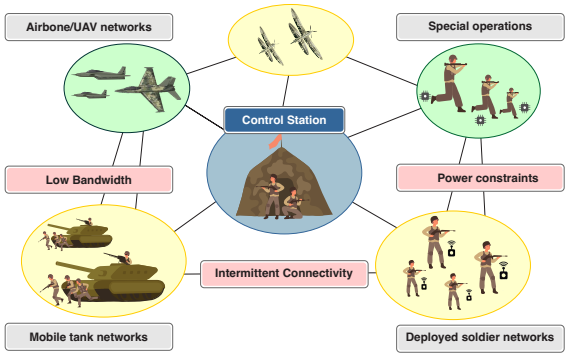


Figure 1: Tactical edge networks face several challenges, including low bandwidth, intermittent connectivity, and power constraints. However, they are essential for airbone/UAV, special operation soldiers, and mobile tanks to interact with each other to accomplish specific mission tasks.

Traditional practices of perimeter-based security [3] have assumptions that make enterprises vulnerable when put in the context of e.g., cloud adoption, remote work. Hence, there is a growing adoption of the Zero Trust (ZT) security paradigm [4], [5], [6] that shuns inherent trust assumptions in perimeter-based security. ZT security is particularly relevant in tactical edge networks with its rigorous access scrutiny, entity authentication, and risk evaluation. The ZT paradigm operates on the principle of "never trust, always verify" [4]. It offers an essential framework for improving the security of CDSs by challenging implicit trust in communication channels. In this approach, no entity within the system is inherently trusted, and every communication request undergoes rigorous verification and validation based on verified credentials, access controls, and other security measures. This reduces the attack surface, enhances threat detection and mitigation by network segmentation, enforces least privilege, and ensures continuous monitoring and validation of communication channels, maintaining trust throughout the operation.

However, while existing ZT security approaches suit enterprise networks in data centers assuming stable hardware and predictable network conditions, they encounter limitations in TENs due to their DDIL nature. The challenges posed by low bandwidth, intermittent connectivity, and resource constraints make complete ZT implementation impractical for

tasks such as those depicted in Figure 1. For example, tasks with computational nature could involve running a computer vision pipeline within a Docker container located at an edge node [7]. These require high availability of CPU, memory, and battery, and a good network throughput, which are otherwise used up by security processes in a full ZT implementation scenario. This leads to the need for tailored solutions that can navigate these unique conditions.

In this paper, we address the above issues and propose a novel risk-based ZT scale approach that tailors security measures to the inherent risk of each scenario in a TEN. This strategy aims to strike a balance between maintaining robust security and ensuring operational efficiency within the dynamic confines of TENs. This tailored approach involves employing a BN model to evaluate communication risks, forming the foundation for a refined ZT metric. This integrated metric takes into account factors such as risk assessment, resource availability, network strength, and entity attributes, including the severity and potential consequences of a breach. The resulting metric assigns a dynamic security grade that reflects the intricate interplay of these factors. To assess the effectiveness of our approach, we develop a drone simulation and implement the BN model. We also implement the ZT system to generate simulation data to input into the BN model. Further, we perfom tests of the ZT system using the generated data, and evaluate the resulting outcomes.

The remainder of this paper is organized as follows: Section II presents related work. Section III puts forth the overview of the ZT metric solution. Section IV explicates the calculation of risk in a communication request using a BN model. In Section V, we detail the parameters and formulation of the Risk-based ZT Scale. The setup of simulation data and a comprehensive analysis of its results are presented in Section VI. Finally, we conclude in Section VII.

II. Related Work

While the growth and significance of edge computing systems are undeniable, their increasing integration with physical infrastructure has exposed them to a host of security risks [8]. To ensure the continued success and safe deployment of edge computing systems, it is essential to develop a comprehensive security framework that addresses their unique challenges [9]. For instance, traditional security standards designed for enterprise networks are not equipped to handle the dynamic and resource-constrained nature of tactical edge networks, which can be vulnerable to cyber threats, unauthorized access, tampering, and data integrity and privacy concerns.

When implementing security measures for TEN environments, it is vital to consider practicality and efficiency, taking into account their resource limitations, intermittent connectivity, bandwidth constraints, and real-time decision-making demands [10]. Implementing an extensive ZT based security protocol for all communication requests can introduce substantial system overhead, potentially hindering the availability of resources required for core tasks such as drone video analytics. These tasks heavily rely on consistent access to network and computing resources [11], [12]. A one-size-fits-all security approach might prove impractical and resource-intensive, potentially impeding the seamless operation of these dynamic systems. Therefore, our novel approach introduces a risk-based ZT scale solution [9], which tailors security measures based on the risk levels obtained using BNs for a given scenario. The solution's implementation is intentionally centered around CDS, which is a prominent use-case within the realm of tactical edge network systems [13], [11]. These situational risk levels can be associated with various dynamic metrics that may indicate a potential breach of network security. By considering the communication and compute bandwidths, and the significance of the asset/drone being requested, we formulated a ZT grading system, which executes the right amount of security protocols [9] on the communication request before granting or denying it.

III. Zero Trust Scale Design

A. Solution Overview

Our proposed framework of the ZT Metric Solution is illustrated in Figure 2 given a context where a source drone initiates communication with a service or hardware on another drone. This communication is channeled through the ZT system, which evaluates the request's state and forwards the relevant metrics to a Bayesian model. Leveraging historical attack data, the Bayesian model calculates a risk score, and transmits back to the ZT system. Considering factors such as network speed, resource availability, and request criticality, the ZT system assigns a context-aware ZT grade to the request and enforces tailored security measures on the source drone for verification.

B. Risk Estimation using Bayesian Network Model

To study past intrusions and leverage such information, we use a Bayesian Network model [14] to probabilistically understand how likely it is for a communication request to be malicious. By customizing security protocols to match the inherent risks of different environments, our risk-based approach optimizes resource allocation, minimizing overhead while precisely aligning security measures with the unique characteristics of each edge system. This allows TENs to strike an optimal balance between their security requirements and operational efficiency, ensuring continued functionality and reliability in diverse and challenging contexts.

In the context of implementing the BN model, several key steps are involved. Initially, the mean values of each of the metrics for all the drones are computed, namely the distance from the control station, packet rate, and energy consumption. Subsequently, the deviations of these metrics pertaining to the requesting drone are calculated in relation to their respective means. These metric deviations then serve as essential input for the BN model. Within the BN framework, a sliding window subset consisting of the n closest rows is generated for each metric, facilitating a focused analysis. For each conceivable attack scenario, the model assesses the probability of an attack attempt by dividing the count of records where attack attempts occurred by the total number of records in the given subset. This probability computation process is repeated across all potential attack combinations, encompassing the various attack vectors. Then the probability of overall attack being successful is calculated, considering different combinations of attack successes. It uses conditional probabilities to model

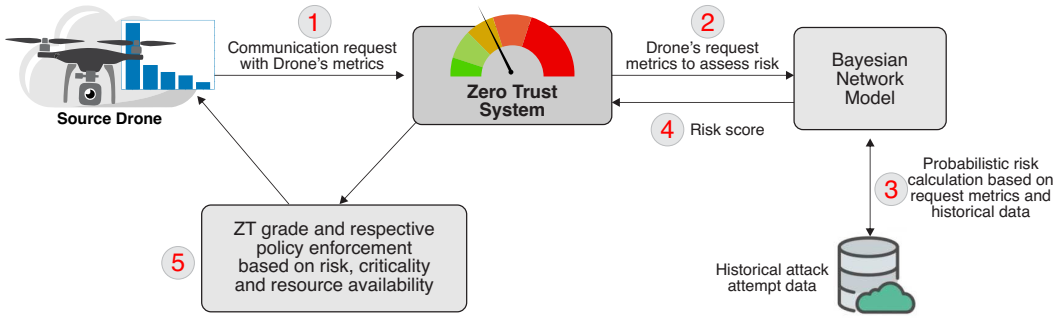


Figure 2: Pipeline architecture of the proposed solution.

how "Attack Success" depends on these conditions, and by summing up these probabilities over all possible scenarios, it gives us the overall risk score of the scenario.

C. Risk-Based Zero Trust Scale

To address the above challenges, we propose a risk-based ZT scale. The scale aims to quantify the level of ZT required for different communication requests based on their associated risks. By assessing the risk profile of each communication request and assigning a ZT metric, the scale classifies them into different ZT grades, allowing for proportionate enforcement of ZT. Hence, we obtain more efficient allocation of resources and targeted enforcement of security measures. Figure 2 illustrates the architecture of the ZT system and the flow of control for a communication request.

IV. Risk Quantification Methodology

In this study, we considered three major attack vectors: Flooder Attack [14], Faker Attack [14], and our novel idea of a third attack vector, the *Physical Capture Attack*.

1) Flooder Attack: A Flooder Attack involves a Denial-of-Service (DoS) attack that can be identified by an unusually high amount of traffic originating from a specific drone or service. Any abnormality in packet transfer rates symptomizes a Flooder attack.

2) Faker Attack: A Faker Attack occurs when a malicious drone pretends to have high levels of available energy to deceive the system into prioritizing itself for packet-forwarding over the intended drones. Abnormal energy consumption can serve as an indicator for detecting a Faker attack.

3) Physical Capture Attack: In CDSs, a physical capture attack refers to a situation where a drone deviates from its designated home territory and is seized or controlled by an unintended party during its flight operation. To ascertain the occurrence of a physical capture attack, we rely on the drone's location as a key indicator at the moment. If a drone is detected within enemy territory, it inevitably indicates a high probability that the enemy has taken possession of it. Although not absolute, the presence of a drone in an unusually distant location significantly suggests the likelihood of a capture situation. Other parameters can however be discovered and used as indicators while further theorizing the attack's mechanism. Figure 3 illustrates this attack scenario.

To evaluate the risk associated with potential attacks, we employ a BN Model [14]. The model encompasses four nodes,

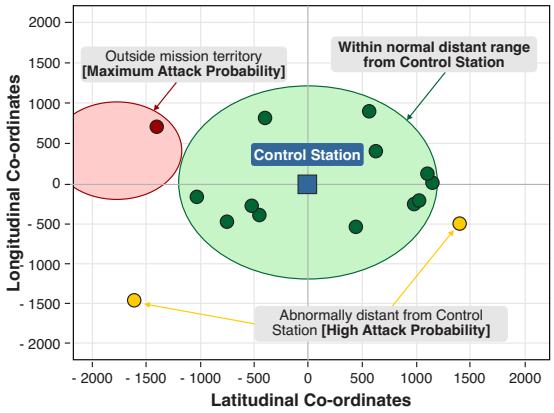


Figure 3: Physical Attack likelihoods based on proximity to control station on a co-ordinate plane.

three of which represent the probabilities of different attack vectors, while the fourth node indicates the likelihood of attack success. Each of these nodes have two states: *True (T)* and *False (F)*, which are indicative of whether a specific type of attack was attempted or not. The attack vector nodes are parents to the attack success node, whose "true" probability gives the chances of the attack being a success.

Historical data concerning attempted attacks, their corresponding behaviors, and the resulting outcomes are utilized as inputs to the model, facilitating an assessment of the attack's likelihood. The "Attack Success" node in the BN model is influenced by the attack vector nodes, resulting in a probabilistic inference referred to as the risk score. This score is derived from the analysis of detected attack activities within the network. By incorporating the probabilities of observing distinct attack patterns and their interrelationships, the model calculates the risk value.

Through the consideration of conditional probabilities linked to each attack type and their joint probabilities in positive scenarios of attack success, the model can estimate the likelihood of various attack scenarios. Additionally, it evaluates the overall risk if approval is granted under those specific behavioral circumstances of the requesting drone.

To calculate the risk based on the overall probability of the attack's success, denoted by Att, we perform summation by taking the product of conditional probabilities and joint probabilities for all scenarios when the attack success node

holds a "True" value.

$$P(Att = True) = \sum_{i,j,k \in \{True, False\}} [P(Att = True | FL = i, FR = j, PC = k) \\ P(FL = i) P(FR = j) P(PC = k)] \quad (1)$$

where FL, FR, PC indicate whether the node is a flooder, faker, or physical capture node, respectively. For a more comprehensive understanding of the aforementioned formalization, further details are available in the reference [14].

V. Risk-based Zero Trust Scale

A. Zero Trust Metric Calculation

To evaluate the Zero Trust metric, we propose to exploit three main factors: *Risk score*, *Compute Easer* parameter and *Criticality* of the assets/drones. Our proposed ZT metric is not a binomial value, rather it is a quantitative scale which is divided into multiple grades based on the value range. Thus, we define ZT Metric as:

$$\textit{ZT Metric} = \textit{Risk Score} \times \textit{Compute Easer} \times \textit{Criticality} \times 100 \quad (2)$$

To put the ZT quantity on a comprehensive scale rather than a decimal point value which is difficult to scale, we multiply the result by 100. While the highest ZT grade is preferable, due to the limitations in the network and processing power, it is not always achievable. The different grades of the model can be assigned to different assets and tasks [15] of a network depending on their risk [9]. The risk estimated from the BN is the preliminary input for our ZT metric. Moreover, by allowing for smoother flow of traffic for tasks that are not high risk, we create faster programs with less overhead. We also consider other criteria defined below, such as the criticality of the requested resource and the availability of resources to fine-tune the ZT metric.

a) Criticality: There arises a need to have proportionate allocation of resources to tasks based on how important the resource is. For instance, surveillance drones or communication relay drones are crucial drones that have dependency and control over a number of other drones, however we do not want intruders to gain access to these at any cost. On the other hand, there are trivial assets such as weather monitoring nodes that are not of much danger in the wrong hands because there is not a lot of control over the system that intruders can gain. Hence, based on the criticality of assets, we categorize them into tiers and postulate a criticality multiplier which raises or cuts down the risk. We have initially allocated these tiers to drones along with their corresponding criticalities, which can be adjusted to align with the specific preferences of our application cautiously. We do so because, a minute misconfiguration of these multipliers can drastically influence the overall ZT metric leading to inappropriate mapping of security measures. For our implementation, we defined four tiers - 1, 2, 3, 4 with criticalities 1.5, 1.25, 1, and 0.8, respectively.

b) Compute Easer: The other critical criteria to be addressed are the available compute and network bandwidths. It is important to conserve compute power of the PDP because it is detrimental for the system to be overloaded affecting availability. Hence, we introduce a conditional multiplier called Compute Easer, that is dynamically figured out based on CPU usage.

Underestimation of risk can result in severe consequences [16]. These consequences are even more drastic for highly critical assets (Tiers 1 and 2). To avoid under-estimating risk for highly critical assets, we set this factor to one for assets in Tiers 1 and 2. For less critical tasks (Tier 3 and 4), based on the usage and availability of CPU resources, a factor between 0.5 and 1 is assigned to reduce the ZT metric to a portion of the actual value. We increase this factor to one in a nonlinear fashion as the available CPU cycles are increased by using a *sigmoid* [17] function:

$$\text{Compute Easer} = \begin{cases} 1, & \text{if Criticality} \geq 1.25 \\ \frac{1}{1+e^{-\Delta}}, & \text{otherwise} \end{cases} \quad (3)$$

where, Δ is the available CPU cycles in billions.

Considering the aforementioned factors such as risk and various criteria, we devise a ZT metric to assess the level of ZT. The ZT metric ranges from 0 to 150 and can be visualized on a sliding scale with a descending value of ZT. This scale assigns tasks or requests into five grades, namely A-E, with grade A indicating the highest need for stringent ZT enforcement, while grade E corresponds to low-risk tasks that do not necessitate extensive ZT measures.

Specific ZT mechanisms are enforced based on the ZT grade of a connection request. For instance, a request falling under grade E undergoes no advanced ZT mechanisms, but some Open Authentication, Role-based Access Control, while all the communication is done cryptographically using digital signatures and public key encryption [18] for confidentiality. For grade D, we can assume the presence of an Intrusion Detection System for continuous monitoring, but with a low granularity. For grade C, we can assume a Multi-Factor Authentication using encrypted one-time pins for verification. Attribute-based Access Control and an increased granularity of continuous authentication can be added in grade B for context-aware authorization and proactive security. Further, we can assume Behavioral Analysis exploiting ML/AI, and Challenge-Response Protocols, which are compute and network-intensive for grade A, the highest. Based on the ZT grade a task falls into, we enforce these various security schemes to "never trust and always verify". However, the number of these grades and the security mechanisms that can be associated with each of them are completely configurable according to the nature and scale of the application.

B. ZT Enforcement Policy based on Network Resource Availability

We need to ensure that the communication among nodes in ZT enforcement remains efficient and does not excessively consume network bandwidth. To do so, we introduce the concept of a Minimum Speed Policy (MSP). This policy is designed to optimize ZT enforcement while considering factors such as network conditions, drone mobility, and bandwidth usage, allowing a balance between security and resource utilization. MSP is the minimum network speed required for effective ZT enforcement within a specified time frame. This speed threshold ensures that the ZT mechanisms can

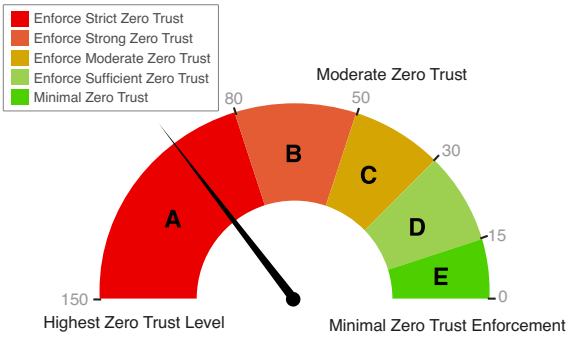


Figure 4: Risk-based ZT Scale with Grade Categorization. Highest Zero Trust Levels, Moderate Zero Trust and Minimal Zero Trust Enforcement.

be applied without compromising their effectiveness, even in situations where network connectivity might be weak due to drone movement or high communication traffic.

When a drone requests a connection, its current network speed is assessed. If the network speed exceeds the MSP, we proceed to enforce ZT mechanisms based on the assigned security grade. This makes sure that the appropriate level of security measures can be applied without overhead. In case the drone's network speed is at least 80% of the MSP, we choose to apply a slightly reduced grade of ZT enforcement, adaptively helping manage network bandwidth without compromising security significantly. If the drone's network speed falls below the 80% threshold of the MSP, it indicates poor network conditions. In this scenario, we generate a Deny Policy, preventing the drone from accessing the network. This proactive step ensures that only devices with a reasonable level of network connectivity can access the resources. To prevent a flood of access requests overwhelming the system, we introduce a *Cool-down period*. This is a standard waiting time imposed on denied nodes before they can make a new access request.

$$\text{ZT Enforcement By MSP} = \begin{cases} \text{Execute ZT with assigned grade,} \\ \quad \text{if Network Speed} \geq \text{MSP} \\ \text{Execute ZT with reduced grade,} \\ \quad \text{if Network Speed} \geq \text{MSP} \times 0.8 \\ \text{Generate Deny Policy,} \quad \text{otherwise} \end{cases} \tag{4}$$

VI. Performance Evaluation

In this section, we present the evaluation of the BN's performance based on metrics such as packet transmission rate, distance from the control station, and energy levels of drones. We also analyze the influence of key ZT parameters, namely Criticality, Compute Easer, and MSP, on security execution.

A. Simulation Setup

1) Drone Data Generation: We created drones as objects with properties including tier level, energy level (Joule), packet transmission rate (Mbps), and location co-ordinates (a co-ordinate pair on a square grid 4000 feet long and wide). This object-oriented approach allowed us to efficiently handle multiple drone instances and their respective characteristics throughout the simulation. The control station was configured with ZT scores of respective grades, tiers and their corresponding criticalities, forbidden (out-of-mission-scope) territory co-ordinates, network MSP standard requirements, and a snapshot of CPU availability which is adjusted for every communication request simulation.

2) Simulated Historical Attack Attempt Data: The historical attack attempt data of the CDS has the record of attack attempts, their corresponding deltas, i.e., how much the packet transmission rate, energy levels, and Euclidean distance of the drone from the control station deviated from the mean of those of the remaining drones in the system, the outcome of each of those attempted attacks, and whether the overall infiltration was successful or not.

To model these instances, we generated simulation data with 3 numerical columns: distance_delta, packet_rate_delta, and energy_delta and corresponding *attack attempted* boolean columns namely is_capture, is_flooder_attack, is_faker_attack, holding either a True or a False value. The values in the numerical columns represent the deviations of a drone's metrics from the mean of those of the remaining drones in the system. To assign truth values of the boolean columns to the data, we would need a proportionality function that would be inclined to having low attack probability when close to the mean of the metrics. As we move beyond the mean, it should indicate that attack attempts are more probable.

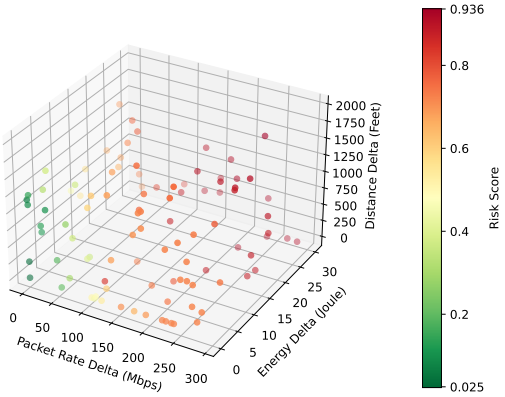
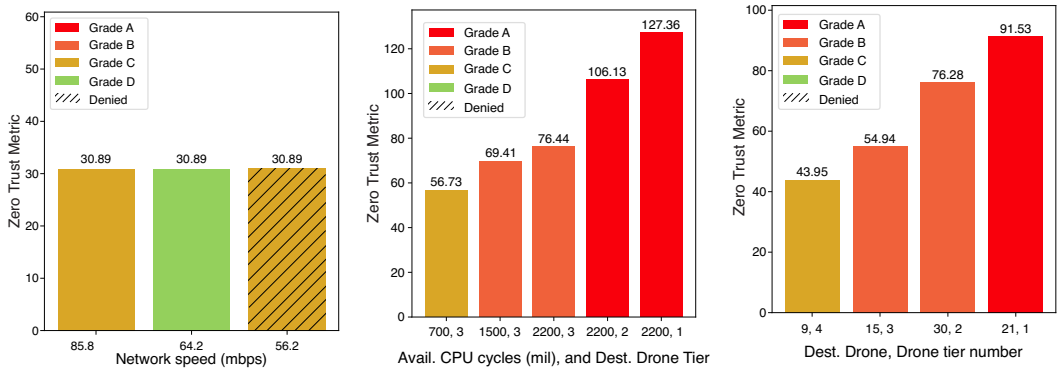


Figure 5: Simulation Results of Variance in Risk score from Bayesian Network based on risk metrics

3) Model Execution: The boolean columns, is_capture, is_flooder_attack, and is_faker_attack of the simulated data determine the probability of each type of attack based on the corresponding delta values. Based on the Equation (1), the probability of attack_success node being true was calculated and termed as risk. This risk value was then used as input for the ZT metric calculation. Based on the parameters *Criticality, Compute Easer and MSP*, risk metrics and corresponding risk grades were computed, allowing for proportionate execution of ZT mechanisms depending on the scenario.

By following this simulation setup, we aim to evaluate the effectiveness and scalability of our risk-based ZT metric for CDSs. In each successive simulation, the effects of the deviation (delta) values on the risk score and impact of our ZT parameters on the ZT metric and grade are analyzed. Several simple scenarios with tuned values of deltas and ZT parameters are used as settings to our simulation and the behavior of the ZT grade according to the scenarios is recorded.



(a) (b) (c)

Figure 6: Experiment scenarios showing: (a) Effects when ZT security policies are enforced fully or partially, or denied. (b) Effects of the Available CPU Resources on the ZT Metric and corresponding ZT grades. (c) Effects of the Criticality parameter on the ZT Score and corresponding ZT grades.

B. Experiment Results

In this section, we present our experiment results from our simulation analysis that are obtained by observing how the BN model risk quantification results vary based on different network activities and scenarios, followed by how the ZT scale is impacted with respect to the ZT parameters and the BN risk scores. In addition, we present the impact of the risk grades on the network performance observing throughput and latency variations over the simulation of a data transfer. Finally, we discuss the complexity analysis of the proposed approach.

1) Risk Quantification Results with Bayesian Network: We conducted tests on the BN using various sizes of drone simulation data and attack attempt data. In Figure 5, we present the results for one specific snapshot of drone data. For this snapshot, various simulations were run with a wide window of deviations of distance from GCS, packet rate, and energy were used as judging metrics of attack probability. The sensitivity of the probabilistic outcome from the BN was found to be very volatile, adjusting itself to minute changes in the criterion metrics. These means were consistent across all the simulations, allowing us to analyze the impact of different metrics and their respective deviations (deltas) from the mean on the risk score. This trend of the model can be observed in the 3D Scatter plot shown in Figure 5.

2) Impacts of Zero Trust Parameters on the Grades: To study the influence of network speed and Minimum Speed Policy (MSP) constraint on the overall ZT grade, simulations were done on a drone with varying network speeds while keeping all other parameters such as the requested target done, available CPU cycles, and the drone's metrics constant. The obtained results of these simulations can be seen in Figure 6(a). It can be observed from the results of the first drone in Figure 6 how under the same ZT metric, network speed enforced the original ZT grade, denied the request, or enforced a lowered grade of ZT. For the original ZT grade 'C', when there was sufficient network speed (i.e., 85.8 Mbps) the ZT system enforced grade 'C' security policies. When the speed was slightly reduced to 64.2 Mbps, the ZT system enforced a lowered grade 'D' of security policies because of only 80% of MSP requirements were being satisfied. Further lowered to 56.2 Mbps, a deny policy was issued because - at least the minimum MSP standards of network speed were not being met. These results show how the MSP parameter helps reduce the load of ZT enforcement in poor network connectivity scenarios, without compromising security of the CDS.

To analyze how the availability of CPU resources affect the compute easer, and in turn the overall ZT metric and grade, simulations under the same risk metrics and network speed were done while changing the target drone being requested and the available CPU cycles as illustrated in Figure 6(b). The first two simulations show that the compute easer does not really have an effect when the criticality of the requested drone is 1.25 or above. This ensures that the CPU shortage does not become an excuse to gain access to critical resources. The ZT metric only decreased slightly because of criticality's effect. In the other three simulations, it can be seen how compute easer can lower the overall ZT metric and grade to prevent CPU overload by high enforcement of ZT. For the same risk, the ZT grade reduced from grade 'A' to grade 'C' with available CPU cycles falling from 2.2 billion to 700 million.

Further, as depicted in Figure 6(c), with the same risk metrics and all other conditions, we perform simulations of requests to drones belonging to various tiers. The simulation results from Figure 6 show how the strength of ZT enforcement with the same risk parameters can vary based on the seriousness of the drone. The final ZT grade was proportionately lowered from grade 'A' to grade 'C' with varying criticalities of the target drones.

VII. Conclusion

In this paper, we have addressed the problem of applying data center-oriented ZT solutions that typically demand large overheads in computation (e.g., traffic flows encryption, anomaly detection using deep learning), storage (e.g., to store activity logs at per-device/per-user granularity) and network resources (e.g., threat intelligence sharing) that are not generally available in tactical edge networks. Specifically, we have developed and evaluated a novel TEN resource-aware, risk-based ZT grading system for an exemplar collaborative drone system setting using a BN model for risk calculation.

Future work can incorporate more attack vectors with their corresponding metrics, defining ZT security mechanisms based on the respective ZT grades.

REFERENCES

[1] A. Al-Ansi, A. M. Al-Ansi, A. Muthanna, I. A. Elgendy, and A. Koucheryavy, "Survey on intelligence edge computing in 6g: Characteristics, challenges, potential use cases, and market drivers," *Future Internet*, vol. 13, no. 5, p. 118, 2021.

[2] G. Cirincione and D. Verma, "Federated machine learning for multi-domain operations at the tactical edge," in *Artificial intelligence and machine learning for multi-domain operations applications*, vol. 11006, pp. 29–48, SPIE, 2019.

[3] M. J. Arata, *Perimeter security*. Mcgraw-hill, 2006.

[4] V. Stafford, "Zero trust architecture," *NIST special publication*, vol. 800, p. 207, 2020.

[5] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero trust architecture," tech. rep., National Institute of Standards and Technology, 2020.

[6] N. F. Syed, S. W. Shah, A. Shaghaghi, A. Anwar, Z. Baig, and R. Doss, "Zero trust architecture (zta): A comprehensive survey," *IEEE Access*, vol. 10, pp. 57143–57179, 2022.

[7] R. Gargees, B. Morago, R. Pelapur, D. Chemodanov, P. Calyam, Z. Oraibi, Y. Duan, G. Seetharaman, and K. Palaniappan, "Incident-supporting visual cloud computing utilizing software-defined networking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 1, pp. 182–197, 2016.

[8] B. B. Madan, M. Banik, and D. Bein, "Securing unmanned autonomous systems from cyber threats," *The Journal of Defense Modeling and Simulation*, vol. 16, no. 2, pp. 119–136, 2019.

[9] D. W. R. Simpson, "Toward a zero trust metric," *Procedia Computer Science*, vol. 204, pp. 123–130, 2022. International Conference on Industry Sciences and Computer Science Innovation.

[10] K. Chan, E. Graves, K. Marcus, T. Moore, J. Perazzone, L. Scott, A. Swami, A. Toth, G. Verma, P. Yu, *et al.*, "Context-aware networking and cybersecurity for resilient networking (summary technical report, oct 2017-sep 2020)," 2022.

[11] A. E. Morel, D. Kavzak Ufuktepe, R. Ignatowicz, A. Riddle, C. Qu, P. Calyam, and K. Palaniappan, "Enhancing network-edge connectivity and computation security in drone video analytics," in *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1–12, 2020.

[12] A. E. Morel, P. Calyam, C. Qu, D. Gafurov, C. Wang, K. Thareja, A. Mandal, E. Lyons, M. Zink, G. Papadimitriou, and E. Deelman, "Network services management using programmable data planes for visual cloud computing," in *2023 International Conference on Computing, Networking and Communications (ICNC)*, pp. 130–136, 2023.

[13] A. E. Morel, C. Qu, P. Calyam, C. Wang, K. Thareja, A. Mandal, E. Lyons, M. Zink, G. Papadimitriou, E. Deelman, and E. Deelman, "Flynet: Drones on the horizon," *IEEE Internet Computing*, vol. 27, p. 35–43, may 2023.

[14] A. E. Morel, E. Ufuktepe, C. Grant, S. Elfrink, C. Qu, P. Calyam, and K. Palaniappan, "Trust quantification in a collaborative drone system with intelligence-driven edge routing," in *NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium*, pp. 1–7, 2023.

[15] A. Osman, A. Wasicek, S. Köpsell, and T. Strufe, "Transparent microsegmentation in smart home IoT networks," in *3rd USENIX Workshop on Hot Topics in Edge Computing (HotEdge 20)*, USENIX Association, June 2020.

[16] S. M. Saghaian NE, T. La Porta, T. Jaeger, Z. B. Celik, and P. McDaniel, "Mission-oriented security model, incorporating security risk, cost and payout," in *Security and Privacy in Communication Networks: 14th International Conference, SecureComm 2018, Singapore, Singapore, August 8-10, 2018, Proceedings, Part II*, pp. 192–212, Springer, 2018.

[17] L. Chen, B. Wang, X. Chen, X. Zhang, and D. Yang, "Utility-based resource allocation for mixed traffic in wireless networks," in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 91–96, 2011.

[18] R. Kaur and A. Kaur, "Digital signature," in *2012 International Conference on Computing Sciences*, pp. 295–301, 2012.