AEGIS: A Special-Purpose Computer Network For Strategic Cyber Defense

by

Francis Mendoza

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Masters of Science

Approved September 2024 by the
Graduate Supervisory Committee:

Aviral Shrivastava, Co-Chair,
Hokeun Kim, Co-Chair
Hwisoo So
Chanhee Lee

ARIZONA STATE UNIVERSITY

December 2024

ABSTRACT

Defending large-scale networks of Connected Autonomous Vehicles (CAVs) and Smart Cities presents unique challenges due to constraints in resources, interoperability, and device diversity. This complexity is compounded by the inherent asymmetry in cyber warfare, where attackers can exploit vulnerabilities with significantly lower resource investment than defenders require.

To address this critical issue, AEGIS (Asynchronous Evidence-Based Guardianship for Integrated Security) is introduced as a government-controlled, special-purpose network designed specifically to enhance the security of CAV and Smart City infrastructure meeting very topology requirements in a very particular configuration. AEGIS operates by imposing high and exponentially increasing computational costs on malicious actions, effectively serving as a deterrent against cyber threats. The system integrates adaptive defense mechanisms, including dynamic subnet reconfiguration, a modified proof-of-work protocol optimized for CAV & Smart City lightweight devices, and a decentralized threat detection system. These features collectively mitigate the asymmetric cost advantage that attackers typically enjoy, requiring them to expend substantially more resources and face statistically unfavorable conditions to successfully execute attacks on this novel network.

Evaluation results demonstrate that AEGIS significantly reduces the probability of successful Byzantine faults while improving performance and energy efficiency by 99.998% and 99.999%, respectively, compared to traditional mechanisms that are incompatible with CAV and Smart City environments. By effectively raising the computational costs associated with attacks and reducing the likelihood of successful exploits, AEGIS provides a novel and robust framework for safeguarding critical infrastructure specifically in the realms of CAVs and Smart Cities.

DEDICATION

To my parents, who sacrificed everything by starting over in a foreign land, for the chance of creating an auspicious future for themselves, me, and the rest of my family. To my siblings Faith, Felicity, and Farenzo: thank you for helping your Kuya for a change with your continuous encouragement and support.

And lastly to my beloved, Brunelle. Without you, I wouldn't be here and none of this would be possible. Thanks for inspiring me to chase my dreams. I love you.

ACKNOWLEDGMENTS

I would like to give special thanks to my longtime co-author, Edward Andert, for being my research partner-in-crime for several years on our many other projects as I finished my MSc degree. His expertise in Autonomous Vehicles was vital for the initial V2V and V2X network hardware for our deployment, compatiblity with SUMO, as well as the wealth of sensor data for consensus. I would also like to thank my colleague, Collins Munyendo, for his thorough edits and illuminating advice about paper presentation. Lastly, I would like to thank my co-chairs, Dr. Hokeun Kim and Dr. Aviral Shrivastava, for their continued mentorship and guidance on paper formatting.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Chapter 1

INTRODUCTION

Cyberwarfare remains a significant threat to essential services and infrastructure. The integration of cyber-physical systems (CPSs) has opened new avenues for cyber attacks with potentially catastrophic consequences. High-profile incidents, such as the GhostStripe attack on Autonomous Vehicles to artificially engineer traffic accidents, demonstrated the profound capability of cyberwarfare to endanger the physical safety of others and cause real-world damage Petit and Shladover 2014. Similarly, the Florida Water Treatment plant hack Cervini, Rubin, and Watkins 2022, an attack that attempted to inject dangerous levels of sodium hydroxide to poison the local water supply, highlighted the vulnerabilities of municipal smart infrastructure to cyber threats, raising national security concerns Perlroth 2021. Furthermore, large numbers of contemporary Internet-of-Things (IoT) devices and compelling use-cases (such as Smart Cities and fleets of Connected Autonomous Vehicles) present themselves as attractive targets due to their emerging, untested nature, and their ability to enable access, supply bandwidth, and other resources to fuel further attacks. These attacks underscore not only the susceptibility of critical infrastructure to cyber threats but also the extensive societal and economic impacts they pose.

The increasing sophistication of cyber threats against CPSs highlights the need for advanced security solutions tailored to their unique challenges Singh et al. 2020. The asymmetry in cyberwarfare, where attacking requires fewer resources and less information than defending, complicates the development of effective defenses ALSABARY 2017. This problem is exacerbated in CPS due to integration of computational and

physical processes, making CPS high-value targets that are often more vulnerable. Traditional countermeasures that are often effective for IT networks are inadequate for CPS environments due to real-time operational demands and heterogeneous nature.

Existing security frameworks for IoT and related technologies highlight these inadequacies, and CPS shows limited scalability and adaptability to evolving threats Radanliev et al. 2018. Furthermore, CPS are especially vulnerable because they present a broad attack surface and lack robust interoperability and standardization needed for a cohesive security posture in large-scale, heterogeneous networks Humayed et al. 2017. Although mechanisms in networks like Bitcoin and those derived from Adam Back's hashcash Alviano 2023 offer solutions to certain asymmetries in cyberwarfare, primarily at the network level against threats like DDoS attacks, they are not suitable for CPS due to the assumption of a homogeneous network coupled with intensive power requirements. Prior-art blockchain-based defense solutions either aren't paradigmically centered on tackling the asymmetry problem in the context of IoT/CPS either Huang et al. 2019; Lee and Kim 2021; Cybenko and Hallman 2021 or are only focused on a specific subproblem, but not holistic defense posture Purohit et al. 2020; Rot and Blaicke 2019; Liang et al. 2022. Currently, there is no comprehensive security approach that effectively addresses both the inherent asymmetry of cyberwarfare and is optimized to protect uniquely vulnerable, high-value CPS networks Lu, Xu, and Yi 2013.

In this paper, we propose 𝔸𝔼𝔾𝕀𝕊 (Asynchronous Evidence-based Guardianship for Integrated Security): a novel special-purpose computer network engineered to meet the multifaceted security demands and aforementioned challenges of CPS networks while mitigating the asymmetric advantage of attackers. 𝔸𝔼𝔾𝕀𝕊 is an application-layer overlay network designed to mitigate the asymmetry traditionally seen in cyber

defense primarily by exponentially increasing the computational cost of cyberattacks via punitive fees, requiring several vulnerabilities to be exploited in parallel to achieve any meaningful result under unfavorable and statistically unlikely conditions. This more tightly-constrains the number of attempts to attack the network by imposing a universal cost of computational resources (for example, in Bitcoin or kilowatt/hours) in response to any malicious activity, in lieu of the more relaxed resource constraint of the number of proxies an attacker has to launch attacks. Participants are permanently banned if they are unable to pay the cost. 𝔸𝔼𝔾𝕀𝕊 possesses several layers for an adaptive "defense-in-depth" cybersecurity doctrine to disrupt, deny, and deceive opportunistic adversaries. This makes the cost-benefit ratio of launching cyber attacks severely unfavorable, thus acting as an excellent strategic deterrent. Furthermore, 𝔸𝔼𝔾𝕀𝕊 is uniquely designed for heterogeneous, safety-critical, large-scale networks of CPS and IoT devices and their operational requirements.

Specifically, 𝔸𝔼𝔾𝕀𝕊 presents the following key contributions:

- Proposes a dynamic, entropy-based form of Moving-Target Defense (*Entropically Randomized Interconnected Subnets*/ERIS) that significantly reduces the probability of a Byzantine fault succeeding to **near-zero**.
- Introduces a decentralized threat identification system (*Autonomous Threat Handling and Engagement Network Application* or ATHENA) that swiftly and reliably detects threats, leveraging a performant consensus mechanism, retroactive audits, and efficient gossip to minimize false positives.
- Develops a high-attrition defense strategy that imposes severe resource costs on attackers as a deterrent, thus making attacks prohibitively expensive and fundamentally unsustainable.

By implementing a defense mechanism that significantly increases the cost and

complexity of cyberattacks for IoT and CPS networks, 𝔸𝔼𝔾𝕀𝕊 effectively deters opportunistic attackers. This approach is crucial in sectors like defense and national security, where the integrity and availability of systems are paramount. The system exemplifies a shift towards more symmetric cyber defense strategies by providing a scalable, effective, and legally compliant method for governments and militaries to secure heterogeneous networks against increasingly sophisticated threats. During our testing, 𝔸𝔼𝔾𝕀𝕊 was shown to drastically reduce the probability of a Byzantine fault succeeding to **near-zero**, while optimizing performance by **99.998%** and energy cost by **99.999%** in normal operation compared to prior-art measures such as Hashcash, whilst rapidly scaling costs in response to malicious behavior. 𝔸𝔼𝔾𝕀𝕊 also possesses high resilience, with a **99.2%** recovery rate.

Chapter 2

BACKGROUND AND RELATED WORK

2.1 Cyberwarfare

Contemporary cyberwarfare is characterized by a significant asymmetry between attackers and defenders Geers 2010 ALSABARY 2017 Todd 2009. This imbalance stems from the inherent advantage attackers possess: they need to identify and exploit only a single vulnerability to succeed Cárdenas et al. 2008. In contrast, defenders must protect all potential points of vulnerability, a task that is both exhaustive and resource-intensive Rid and Buchanan 2015. This disparity is exacerbated by the rapid evolution of attack methodologies for increased lethality and sophistication. Cyber defense difficulty is further exacerbated by needing uninterrupted services while also protecting against an ever-expanding threat landscape, thus making it more challenging and resource-intensive than that of the attacker. This imbalance creates a scenario where the cost and effort of launching attacks are significantly lower than defense Rid and Buchanan 2015. Furthermore, the attacker's only real upper limit in attempts to attack a defender is the number of exploits they find for systems-level attacks and the number, role, and positioning of proxies for network-level attacks; the attacker is never truly eliminated, with the defenders only able to patch the exploit and pursue recourse through limited legal means, which are in most cases functionally impotent and do not deter attacks from continuing in most cases.

2.2 Security For Cyber-Physical Systems

Based on the high value use-cases, large attack surface, and unique threat model that encompass networks of CPS and IoT devices, the Cyber-Physical Systems (CPS) security domain has been shaped by the need to address complex, heterogeneous environments Rajhans et al. 2014. Early works in CPS security focused on the interoperability, game theory, and technical security challenges posed by the heterogeneous nature of these systems, setting the foundation for understanding the need for adaptive and multifaceted security approaches that can accommodate such a network Song, Fink, and Jeschke 2017. Prior-art that focuses on hardware-based security is an integral part of a strong security posture, but the majority of these are not applicable to the 20 billion IoT devices currently in circulation, as retrofits are complex, expensive, and practically infeasible given the incentives of manufacturers and other parties in the technology stack Wu, Sun, and Chen 2016. Prior-art software-based security solutions are better suited to address heterogeneity and scalability problems Humayed et al. 2017, varying significantly in focus from low-level systems to application-layer software Ruan and Hori 2012, but the overwhelming majority are still within the same paradigm of asymmetric cyber defense that is maladaptive to the defender. Papers like Hashcash Back et al. 2002 first exemplified a preliminary paradigm shift to make cyber defense symmetric, but Hashcash is not designed for the context of CPS and IoT. Regardless, application-layer software is one of the most sensible choices for an interoperable security solution.

2.3 Intrusion Detection and Threat Intelligence

Heuristic-based intrusion detection Bazrafshan et al. 2013 remains a staple for identifying and mitigating threats in any computer, including CPS. Studies in this domain have explored the efficacy of using diverse sets of heuristics to detect anomalies and potential security breaches, emphasizing the need for adaptable and context-aware security mechanisms Keshk et al. 2019. These approaches have been instrumental in moving beyond traditional static security solutions and are critical in offering more flexibility and responsiveness in rapidly changing CPS environments. Such intrusion detection systems are usually centrally hosted, which presents a critical point of failure in the rest of the network. This is not as applicable in a CPS environment, where decentralized peer-to-peer networks between devices are far more common Rajhans et al. 2014. There are, however, multiple prior-art works utilizing Blockchain/Distributed Ledger Technology for disseminating threat intelligence, but these aren't connected to a broader security system, much less being optimized for the performance and topology requirements of IoT/CPS Ma et al. 2023 Gong and Lee 2020 Chatziamanetoglou, Rantos, et al. 2023.

2.4 Decentralization In Distributed Systems

The decentralization of networks in CPS is another key research area, driven by the need to eliminate single points of failure and enhance resilience Cassottana et al. 2023. Notable studies have highlighted the advantages of decentralized architectures, including improved scalability, fault tolerance, and resistance to targeted attacks Wang, Xuan, and Zhao 2003. However, there is an inherent strict performance tradeoff

compared to centralized topologies. Due to the critical nature of performance for CPS, 𝔸𝔼𝔾𝕀𝕊 integrates a hybrid topology, inclusive of a decentralized peer-to-peer network layer in the fog layer, with partial centralization in the edge layer below to quickly disseminate messages.

Due to decentralization, the consistency in such a distributed system must be obtained through consensus Mullender 1990. Hence, consensus mechanisms optimized for performance within CPS have been pivotal. A subset of research delved into the adaptation of BFT protocols for large, heterogeneous networks due to their performant nature Moniz, Neves, and Correia 2012, underscoring their potential in maintaining consistent and reliable network states under adversarial conditions Bodkhe et al. 2020. However, environmental interference and physical attacks, which are part of the larger attack surface, prevent strong assertions or guarantees for synchronous consensus. Asynchronous consensus is more resilient and reliable, but no explicit guarantees can be made regarding time-to-finality; only the number of steps.

2.5 Punitive Cost Functions

Lastly, while heavily underexplored, preliminary prior art exists in proposing punitive measures to deter malicious activities in cyberwarfare. While contemporary measures such as legal recourse are applicable on less-sophisticated attackers within the jurisdiction of the victim Galliott 2016, they are functionally impotent against sophisticated attackers outside both their geopolitical and legal sphere of influence, especially in the case of Advanced Persistent Threats (APTs) engaging in classical cyberwarfare. Hashcash Back et al. 2002 proposed the use of escalating cost structures as a means to impose computational penalties on attackers, effectively disincentivizing

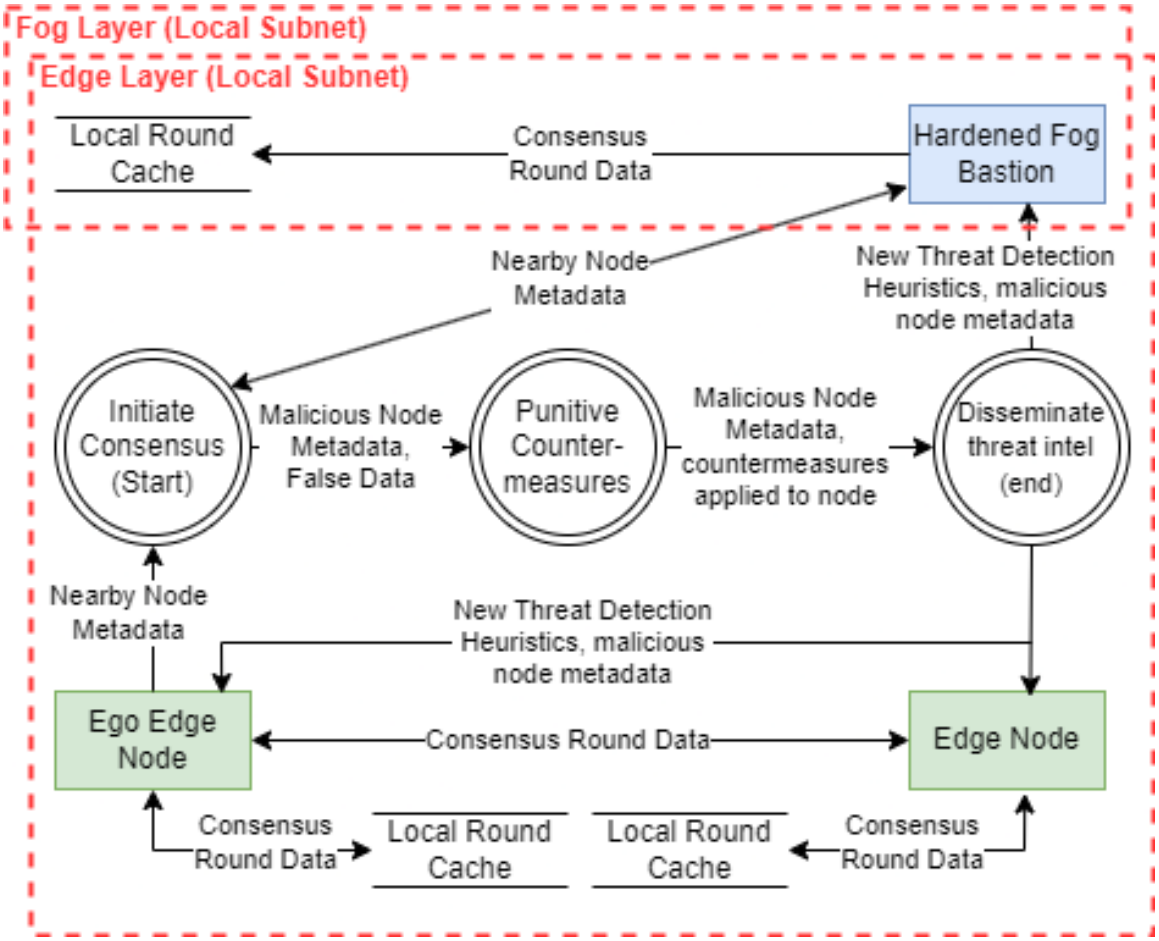

Figure 1. Threat model specific to a local subnet

sustained malicious actions Robinson, Jones, and Janicke 2015. However, Hashcash is not suitable for the unique network topology and resource constraints of CPS. Furthermore, Hashcash was designed to protected against DDoS attacks, but not other attack types whose nature doesn't emphasize overwhelming through volume (ie: Zip Bombs, Worms, etc.). Other cost-asymmetric countermeasures also exist, but these are limited in scope and are not, by themselves, a holistic application-layer security system that protect against different types of attacks Blocki and Datta 2016.

Chapter 3

OUR APPROACH

3.1 Threat Model

The AEGIS framework is engineered to anticipate and mitigate a broad spectrum of cyber threats targeting CPS, from standard lone wolf hackers to sophisticated entities akin to Advanced Persistent Threats (APTs) with extensive resources and advanced cyberwarfare capabilities. While AEGIS is robust against many forms of cyber attacks, it recognizes specific attacker capabilities and vulnerabilities that require rigorous defense mechanisms. The threat model in Figure 1 is for a subnet, with Figure 2 representative of other subnets interacting across the cloud, fog, and edge layers.

Attacker Capabilities: We assume the attackers can:

- Conduct network-based attacks aiming to disrupt, intercept, delay, or manipulate the communication between nodes in the system's layers: edge, fog, and cloud nodes.
- Employ sophisticated malware and phishing tactics to gain unauthorized access to system components or sensitive data.
- Execute various forms of denial-of-service (DoS) attacks aimed at degrading system performance or availability.
- Exploit vulnerabilities within the system's components or communication protocols, barring zero-day exploits which are considered out-of-scope.
- Can compromise an individual hardened fog-layer bastion RSU with irrationally

high resource expenditure, but not a majority. Furthermore, if an RSU is compromised, the edge devices under it are assumed to be compromised as well (given the lower security capabilities of edge devices compared to the RSU). However, the attackers *cannot* breach any hardened cloud layer device.

𝔸𝔼𝔾𝕀𝕊 operates through an initial registration and authentication phase and then switches between scenarios between normal operation and when threats are detected, as outlined in Algorithm 1. Initially, the Subscription Algorithm referenced in Algorithm 2 facilitates the secure registration and integration of devices into 𝔸𝔼𝔾𝕀𝕊's network, ensuring that each device is properly authenticated and aligned with the network's security protocols.

Algorithm 1: 𝔸𝔼𝔾𝕀𝕊 General Algorithm

Initial Registration, Authentication, and Topology Assignment: Subscription Algorithm applied.
Consensus & Operation: For each subnet $\mathcal{S}_j$, operations proceed as follows:
  Consensus Initiation: $\text{Ego}_i \in \mathcal{S}_j$ initiates with $Consensus(\mathcal{S}_j)$.
  Subnet Formation: Nodes in $\mathcal{S}_j$ are determined by $Proximity(\text{Ego}_i, RSU) + Flocking(\mathcal{D})$.
  Protocols Applied:
    Handshake, Proactive Defense, and Dual-Consensus Algorithms: $Handshake(\mathcal{S}_j) + PD(\mathcal{S}_j) + DC(\mathcal{S}_j)$.
    Heartbeat applied post-consensus: $Heartbeat(\mathcal{S}_j)$.
Threat Response: On detecting a threat $\mathcal{T}_k$ by Targeting Service:
  Threat Identification & Relay: $TS(\mathcal{D}_i) \rightarrow DC(\mathcal{T}_k)$.
  Defensive Actions: Apply $ReactiveDefense(\mathcal{D}_i, \mathcal{T}_k)$.
  Information Dissemination: $TS \rightleftharpoons Info(\mathcal{T}_k, \mathcal{D}_i)$.
  Network Stabilization: Return to normal operation with updated protocols if necessary.

Following this, 𝔸𝔼𝔾𝕀𝕊 employs a consensus-based operational model for each subnet. Consensus can be run on any data, but is typically used to check the versioning hashes of the AEGIS client and other system software, as well as the chain-

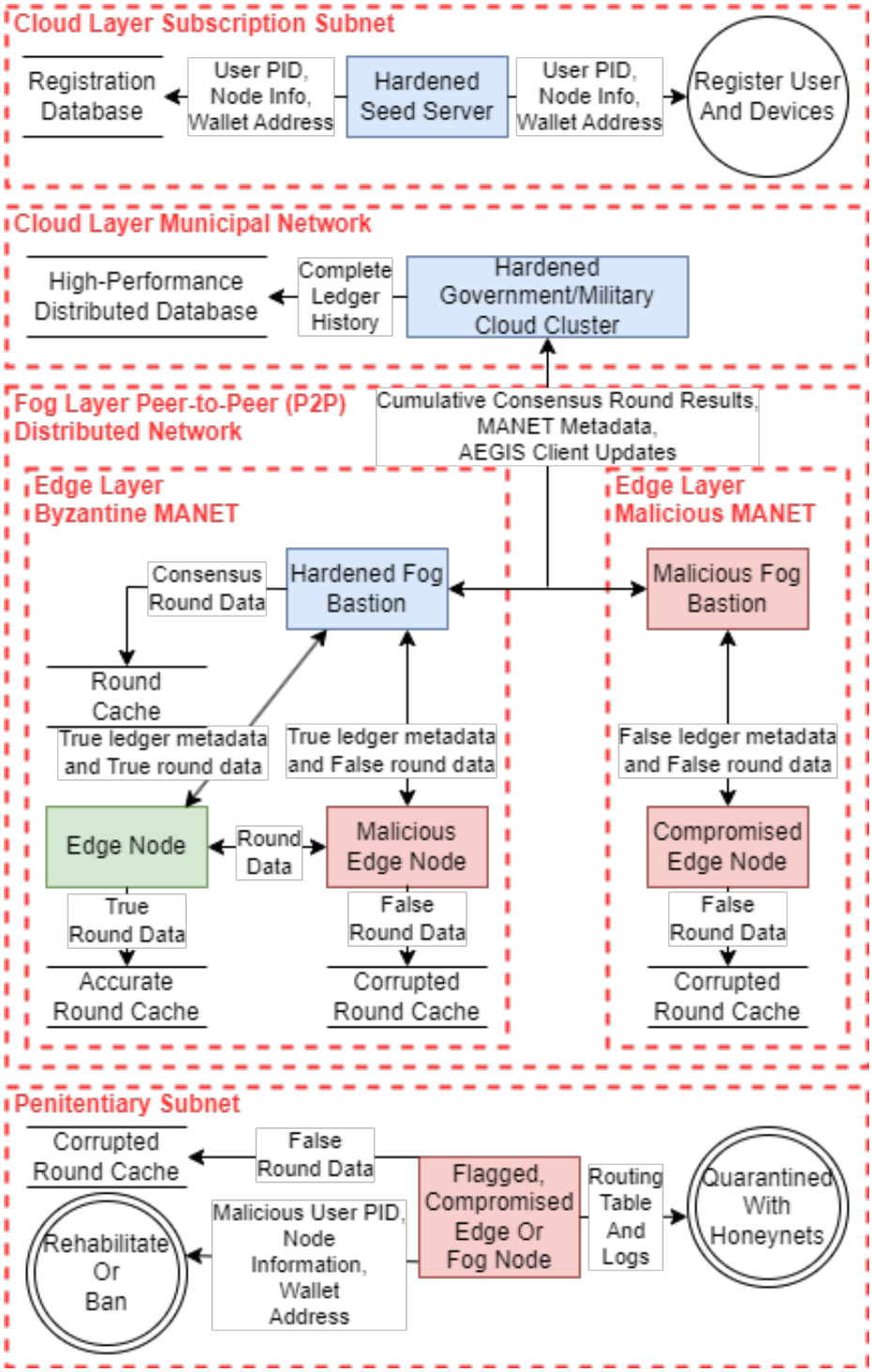

Figure 2. Threat model across subnets in various layers

Algorithm 2: $\mathbb{AEGIS}$ Subscription Algorithm

Subscription Initiation: Initiate$(\mathcal{U}, \mathcal{D})$ where $\mathcal{U}$ is the user and $\mathcal{D}$ is the set of devices.
Credential Submission & Linkage:
    VerifyCredentials$(\mathcal{U})$ and LinkWallet$(\mathcal{W}_{\text{user}}, \mathcal{A}_{\text{AEGIS}})$.
Device Attestation & Authentication:
    $\forall d_i \in \mathcal{D}$, Attest$(d_i, C_i, S_i)$ and Authenticate$(d_i, \mathcal{N}, \mathcal{K}_{\text{priv},i})$.
Security Assessment & Topology Assignment:
    $\forall d_i \in \mathcal{D}, A_i = \text{Assess}(C_i, S_i)$.
    TopologyAssignment$(\mathcal{D}, A_i, P_{i,j})$ based on A_i and proximity $P_{i,j}$ to $\mathcal{S}_j$.
Authentication & Finalization:
    Finalize$(\mathcal{D}, \mathcal{K}_{\text{crypto},i}, \mathcal{C}_{\text{cert},i})$.
    $\mathcal{U}$ and $\mathcal{D}$ are now active within AEGIS.

of-custody as to the ownership of devices, their communications with other devices (whether benign or malicious), and their movement across the network. It is initiated by an ego node, in which subnet topology is then dynamically calculated based on physical proximity via flocking algorithms. The Handshake Protocol is then applied to check the nodes' eligiblity to join the subnet. In Proactive Defense, ERIS is then utilized to generate expirable security groups to all subnet nodes. Dual-Consensus is then applied to continuously monitor the network state. Post-consensus, the hardened fog bastion (RSU) round cache updates on consensus results, while the Heartbeat protocol is used to maintain network integrity and coherence by disseminating round data.

When a potential threat is detected by the autonomous Targeting Service, $\mathbb{AEGIS}$ transitions into a threat response mode. This involves the identification and relay of threat information, the application of "Reactive Defense" punitive measures on malicious nodes, and the dissemination of attack and malicious node metadata across the network. Finally, $\mathbb{AEGIS}$ seeks to stabilize the network, the identities of the malicious nodes and their associated owners, deciding whether to rehabilitate or ban

them, and then returning to a standard operational state. A hierarchical finite state machine describing 𝔸𝔼𝔾𝕀𝕊's general operation can be found as Figure 3.

We acknowledge limitation of the 𝔸𝔼𝔾𝕀𝕊 design that its vulnerable to partitioning attacks, where attackers can isolate a subnet and gain control over its data, compromising network integrity and confidentiality. Notably, supply-chain attacks, zero-day exploits at the protocol level, and attackers with unlimited resources are out of scope for this threat model. Our threat model assumes attackers cannot breach the hardened cloud layer, which is protected by robust physical security and authentication measures.

3.2 Network Topology

Due to the two-fold performance and security requirements of 𝔸𝔼𝔾𝕀𝕊, a hybrid topology was designed, akin to a "forest of trees" outlined in Figure 4. AEGIS uses a private cloud model to severely limit the attack surface, whilst being composed of hardened, cross-layer subnets so that microsegmentation Basta et al. 2022 prevents attackers' lateral movement. There are three layers to the 𝔸𝔼𝔾𝕀𝕊 network:

- **Cloud Layer:** Modified Proof-Of-Work computations and critical data exist here in the government or military-controlled hardened cloud data center. Completely homogeneous.
- **Fog Layer:** Numerous Hardened Roadside Units (RSUs) acting as bastion nodes for local edge subnets are geospatially dispersed, providing modified Proof-Of-Work computations, low latency, and routing to other adjacent subnets. Largely homogeneous.

- **Edge Layer:** Most chaotic. Very vulnerable to attack if isolated. Organized as a set of both Wireless and Mobile Ad-Hoc Networks (WANETs/MANETs). Completely heterogeneous.

Functionally, a decentralized peer-to-peer network exists at the fog layer among Bastion RSUs to facilitate data sharing. However, due to the nature of the distribution of RSUs at the Fog Layer, there is at least one RSU in the transmission vicinity of edge layer WANETs and MANETs. The devices at such WANET/MANET form a star topology to the hardened RSU, which acts as a bastion host for the edge subnet. This enables fast gossip from ad-hoc consensus instances at the edge to the RSU as a witness, while also enabling the hardened RSU bastion to manage and protect the edge devices for the duration of the subnet lifetime. As outlined in Figure 4, a cloud layer subscription subnet is publicly queryable for prospective users who wish to register, who must bind IP and wallet addresses as collateral. Penitentiary subnets are made ad-hoc at both fog and edge layers for compromised devices, their connections to the rest of the $\mathbb{AEGIS}$ network secretly isolated by quarantine honeynets.

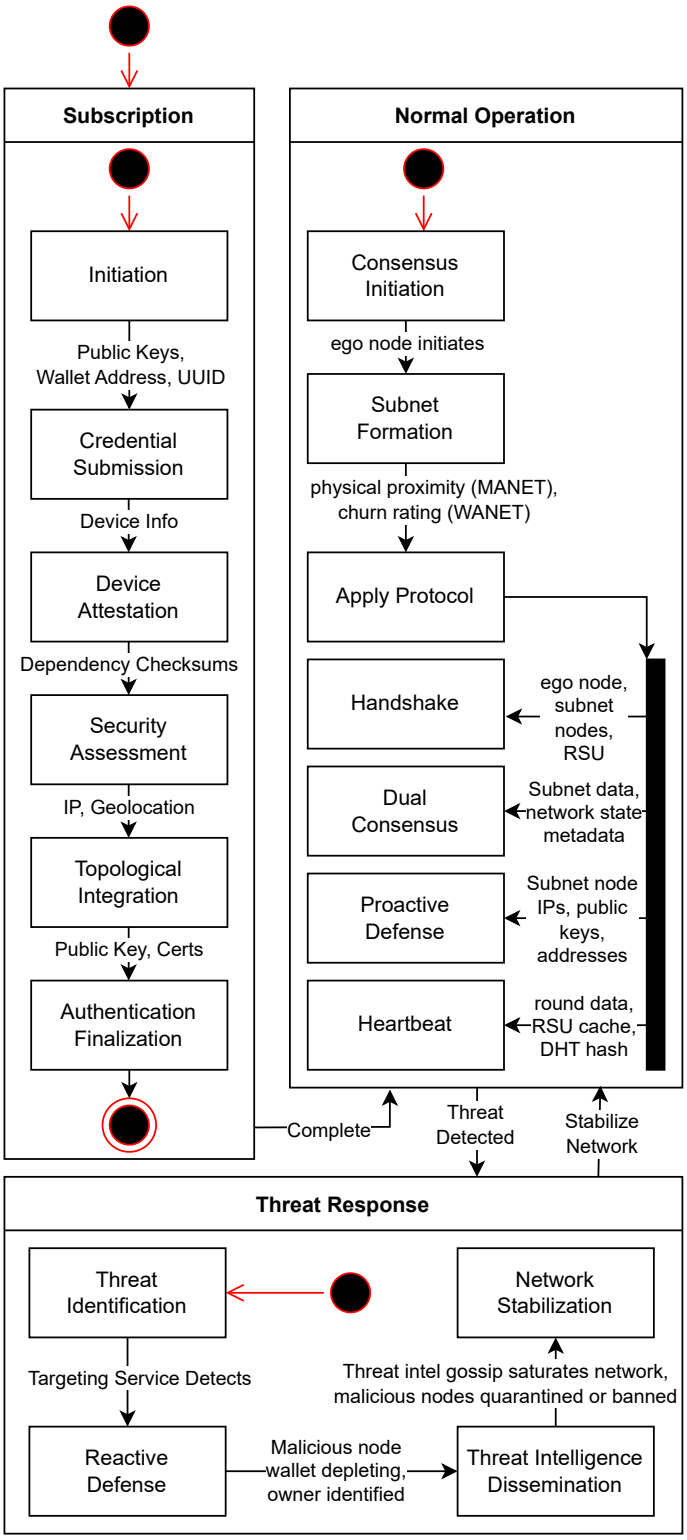

Figure 3. Hierarchical Finite State Machine of the AEGIS general algorithm for high-level operational flow

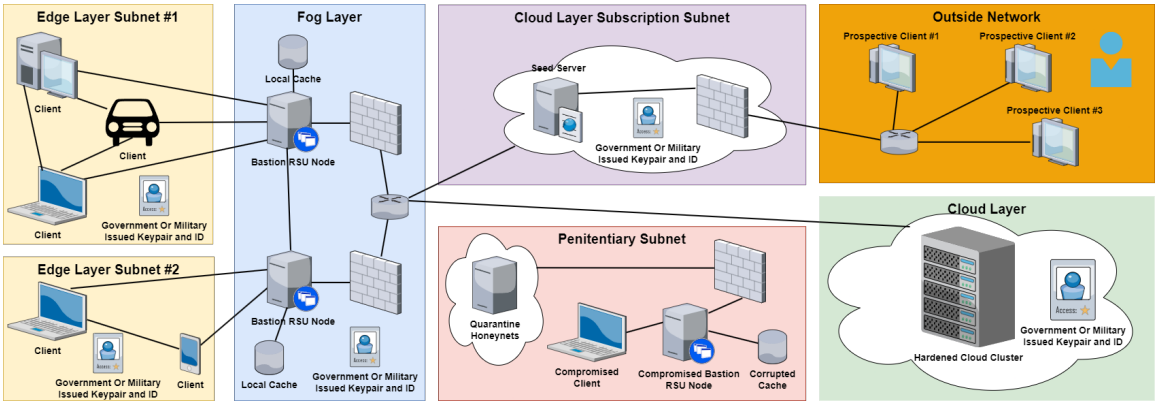

Figure 4. AEGIS topology diagram with different responsibilities and capabilities.

Chapter 4

CORE CONTRIBUTIONS

4.1 Entropically Randomized Integrated Subnets

Entropically Randomized Integrated Subnets (ERIS) harnesses randomness within the $\mathbb{AEGIS}$ to make reconnaissance and the subsequent execution of successful attacks difficult. By integrating the principles of dynamic allocation, entropy maximization, and obfuscation, ERIS complicates the attack vectors, requiring attackers to concurrently gain control over both the target Mobile Ad-Hoc Network (MANET) and the associated expirable security groups in both the MANET and Wireless Ad-Hoc Network (WANET) part of the subnet, which can be composed of non-moving devices that can still join or leave the ad-hoc network (virtual churn). The topology and groupings are dynamically assigned upon subnet creation and exists for a limited time before the subnet dissipates due to node churn. ERIS works as such:

- **Dual-Layer Fault Tolerance Breach Requirement**: ERIS requires attackers to breach the fault tolerance limit of both the transient MANET and the randomly assigned expirable security groups. This dual-layered approach significantly elevates the complexity and resources required for an attack to be successful.
- **Identity Obscuration and Honey Network Integration**: Through temporary IPs and addresses, ERIS masks the identities of nodes within the expirable virtual groups, further obscuring the targets. The inclusion of nodes from a

honey network into these groups serves to mislead potential attackers, making reconnaissance difficult to reduce the likelihood of attack.

- **Churn and Its Implications**: ERIS capitalizes on churn—the phenomenon of nodes joining and leaving the network—for effective Moving-Target Defense in the hybrid subnet composed of a physical MANET and virtual WANET. In the MANET, churn presents an operational window of approximately **10-30 seconds**, making reconnaissance and attack planning challenging without total control over the subnet. Churn within the security groups is randomized, rendering predictive strategies futile.

To enhance the technical understanding of ERIS, we introduce a specific algorithm for subnet configuration, alongside a mathematical model to quantify the entropy within the system.

4.1.0.1 Subnet Configuration Algorithm

This algorithm dynamically allocates nodes to different subnets based on entropy maximization principles to ensure robust network defense and manageability.

Algorithm 3: Dynamic Subnet Allocation for ERIS

```
Initialize network node list N and subnet list S
Define maximum subnet size maxSize
for each node n ∈ N do
    Calculate potential subnets based on proximity and current entropy
    Assign node n to subnet s ∈ S that maximizes entropy
    if size of subnet s exceeds maxSize then
        Trigger reconfiguration for subnet s
    end if
end for
return Updated subnet list S
```

4.1.0.2 Entropy Calculation Formula

To measure the entropy and thus the unpredictability introduced by ERIS, we use the following entropy measure:

$$H(S) = -\sum_{i=1}^{k} p_i \log p_i \tag{4.1}$$

where $H(S)$ is the entropy of subnet configuration S, p_i represents the proportion of nodes in the i-th subnet, and k is the total number of subnets. This entropy measure helps in evaluating how well the subnets are configured to prevent predictability in node assignments, thus enhancing network security.

4.1.0.3 Network Reconfiguration Trigger

The network reconfiguration is triggered based on a threshold entropy value that ensures optimal distribution of nodes and maximizes network resilience:

$$\text{Trigger Reconfiguration if } H(S) < H_{\text{threshold}} \tag{4.2}$$

These models and algorithms form the core mechanisms by which ERIS dynamically manages and secures the network, leveraging entropy to create a robust defense against potential cyber threats.

The robustness of ERIS against cyber-attacks is quantified by modeling the near-zero probability of successfully breaching the network, considering both the dynamics of churn and the necessity for dual-layered control.

Given variables:

- $P_M(t)$: Probability of controlling the MANET within time limit (t).

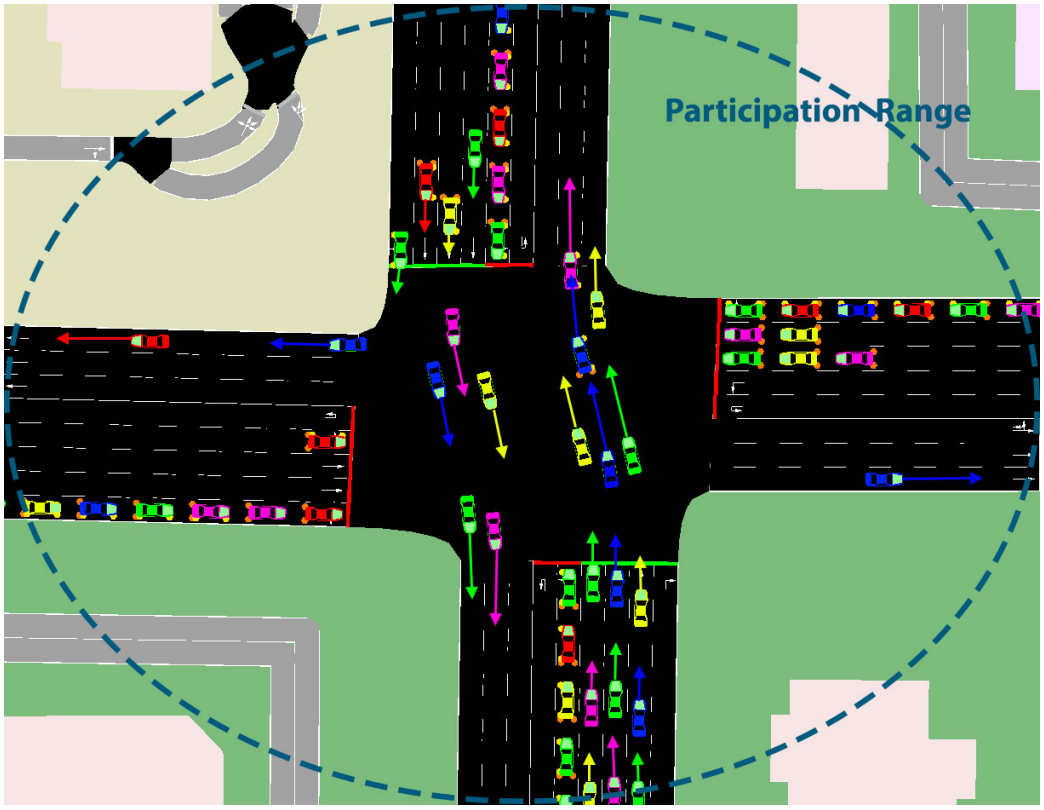

Figure 5. This shows a group of Connected Autonomous Vehicles (CAVs) that has been segmented into randomized subnets of size=10 by ERIS, where the vehicles are colored based on their respective subnet. Velocity vectors of the vehicles are depicted showing the physical churn the participation area is about to undergo with multiple vehicles entering and exiting within the next second. Both of these physical and virtual (ERIS) churn mechanisms work together to make the likelihood of the conditions for a fault to succeed near zero.

- $P_{V_i}(t)$: Probability of controlling the i^{th} security group within time limit (t).
- $P_{C_{ph}}(t)$: Probability of retaining control in the MANET despite churn to make a successful attack within time (t).
- $P_{C_{v_i}}(t)$: Probability of retaining control in the i^{th} security group despite churn to make a successful attack within time (t).

The comprehensive success probability is expressed below in equation 4.3:

$$P_s(t) = P_M(t) \cdot P_{C_{ph}}(t) \cdot \prod_{i=1}^{l} \left(P_{V_i}(t) \cdot P_{C_{v_i}}(t) \right) \tag{4.3}$$

Assume:

- $P_M(t) = 0.05$, $P_{C_{ph}}(t) = 0.30$
- For $l = 2$ security groups: $P_{V_1}(t) = 0.04$, $P_{C_{v_1}}(t) = 0.45$; $P_{V_2}(t) = 0.03$, $P_{C_{v_2}}(t) = 0.55$.

Therefore:

$$P_s(t) = 0.05 \times 0.30 \times (0.04 \times 0.45) \times (0.03 \times 0.55) \approx 4.46 \times 10^{-6} \tag{4.4}$$

This sample calculation shown in the above equation 4.4 proves that, even under conservative estimates, the probability of successfully compromising ERIS remains exceedingly low (0.000446% in this case). The compounded effect of needing to control and hold both network groups amidst churn requires attackers to overcome multiple high-improbability hurdles within a limited time frame, rendering the attack's success probability near-zero.

4.2 Autonomous Threat Handling and Engagement Network Application

The AEGIS architecture integrates a pivotal subsystem, the Autonomous Threat Handling and Engagement Network Application (ATHENA). This decentralized subsystem uses heuristic-based malware detection and efficient gossip of critical updates for swift and accurate threat detection and mitigation. The operation of this system is characterized by:

- **Decentralized Detection and Response**: By using a resilient, distributed architecture, this system ensures comprehensive monitoring and threat identification capabilities across the network, while enabling a swift collective response to threats.
- **Heuristic Multimodality for Enhanced Accuracy**: Employing a diverse array of heuristic analyses allows the system to accurately distinguish between benign and malicious activities. This multimodal approach significantly reduces the likelihood of false positives to maintain operational integrity and minimize unnecessary disruptions. We are not claiming to have made any new heuristics (though the system utilizes prior-art detection heuristics and can accommodate new ones), but rather engineered a way to detect threats and disseminate threat intelligence quickly in a decentralized and resilient manner.
- **Dynamic Response Protocols**: Through the implementation of advanced protocols such as the Handshake, Heartbeat, and Dual-Consensus Protocols in algorithms 4, 5, and 6 respectively, the system dynamically updates its defense strategies based on current threat intelligence.

Three core algorithms of the $\mathbb{AEGIS}$ system (Handshake, Heartbeat, and Dual Consensus) form the backbone of ATHENA's fast and resilient threat identification and response across the network. The Handshake algorithm in Algorithm 4 ensures that only nodes with validated service codes can form or join subnets, effectively preventing unauthorized access. The Heartbeat protocol in Algorithm 5 continuously monitors the network, updating node statuses and disseminating critical threat intelligence quickly across nodes, which is crucial for maintaining network integrity in real-time. Finally, the Dual Consensus protocol in Algorithm 6 employs both immediate local consensus through BOSCO for quick reactions and blockchain-backed validations for

long-term accuracy, ensuring decisions are both rapid and reliable Song and Renesse 2008.

Algorithm 4: Handshake Algorithm in 𝔸𝔼𝔾𝕀𝕊

Input: ego_node, $nearby\_nodes[]$, rsu
$ego\_hash \leftarrow \text{hash}(targeting\_service\_code(ego\_node))$
$rsu\_hash \leftarrow \text{hash}(targeting\_service\_code(rsu))$
if $ego\_hash \neq rsu\_hash$ **then**
    Abort handshake. ego_node cannot form or join a MANET under this rsu.
else
    $candidate\_nodes[] \leftarrow \emptyset$
    for all $node \in nearby\_nodes[]$ **do**
        $node\_hash \leftarrow \text{hash}(targeting\_service\_code(node))$
        if $node\_hash = rsu\_hash$ **then**
            $candidate\_nodes[]$.add($node$)
        end if
    end for
    if size($candidate\_nodes[]$) > predefined_threshold **then**
        Form MANET with ego_node and $candidate\_nodes[]$
    else
        Output failure to form MANET
    end if
end if
Output: Formed MANET or failure indication.

𝔸𝔼𝔾𝕀𝕊 employs an enhanced consensus mechanism tailored for Cyber-Physical Systems (CPS), integrating Byzantine One Shot Consensus (BOSCO) as a wrapper with a modified proof-of-work as the underlying consensus mechanism designed to factor in churn, the unique network topology of CPS, and the lightweight form factors for compatibility while maintaining respectable security. This is because traditional proof-of-work is ill-suited for CPS, while BOSCO by itself requires an underlying consensus mechanism that satisfies the properties of Agreement, Unanimity, Validity, and Termination Song and Renesse 2008. BOSCO's optimization allows rapid state validation across nodes with varying computational capabilities, ensuring performant consensus even the complex, dynamic subnet topology generated by ERIS. Additionally,

Algorithm 5: Heartbeat Protocol in $\mathbb{AEGIS}$

Input: final_round_data, local_RSU_cache, kademlia_DHT
malicious_nodes, *honest_nodes*, *nonresponsive_nodes* ← ExtractNodes(final_round_data)
for node ∈ malicious_nodes **do**
    UpdateCacheAndDHT(node, local_RSU_cache, kademlia_DHT, 'malicious')
    behavior_type ← IdentifyMaliciousBehavior(node)
    if Not PreviouslyRecorded(behavior_type) **then**
        DisseminateThreatIntelligence(behavior_type)
        UpdateTargetingServiceWithHeuristic(behavior_type)
    end if
end for
for node ∈ honest_nodes **do**
    UpdateCacheAndDHT(node, local_RSU_cache, kademlia_DHT, 'honest')
end for
for node ∈ nonresponsive_nodes **do**
    UpdateCacheAndDHT(node, local_RSU_cache, kademlia_DHT, 'nonresponsive')
end for
Output: Updated kademlia_DHT and local_RSU_cache, Dissemination of new threat intelligence (if applicable)

Algorithm 6: Dual Consensus Protocol in $\mathbb{AEGIS}$

Input: subnet_data, network_state_components
violating_nodes ← BOSCO(subnet_data, network_state_components)
for node ∈ violating_nodes **do**
    if TargetingService(node) == "malicious" **then**
        ExponentialSlidingCost(node)
        MedusaStunlock(node)
    end if
end for
validated_txns ← ProofOfWork(subnet_data)
for txn ∈ validated_txns **do**
    Add txn to subnet transaction pool
end for
Output: Updated node statuses in the subnet, Validated transactions for the subnet, Nodes flagged as malicious by TargetingService

𝔸𝔼𝔾𝕀𝕊 introduces a mechanism for managing suspected threats through immediate and retroactive measures. Once a threat is detected, the system dynamically adjusts punitive costs tailored to the nature of the attack: employing exponentially sliding costs for volume-based attacks like DDoS, or activating slashing conditions for non-volume attacks, such as a zip bomb or Trojan horse. Concurrently, flagged nodes are unknowingly isolated within a penitentiary subnet, while their connections are instead routed to honeynets in order to blunt the attacker's offensive capabilities away from real targets. The dual consensus protocol, detailed in algorithm 6, allows for rapid local actions against threats while a subsequent blockchain-based validation process, which includes up to 60 minutes of confirmation and a sysadmin review, ensures accurate final adjudications. Erroneously targeted nodes, if cleared, are promptly exonerated, refunded overcharges, and reintegrated into the network. Confirmed malicious nodes and their associated owner are banned from the network.

4.3 Adaptive, High-Attrition Defense

The "Adaptive, High-Attrition Defense Mechanism" within the 𝔸𝔼𝔾𝕀𝕊 network is specifically designed to ensure the network's resilience against advanced and persistent cyber threats. This mechanism expands upon concepts derived from Adam Back's Hashcash protocol and similar initiatives like Microsoft's "Penny Black" project. Unlike traditional approaches that focus primarily on passive defense, 𝔸𝔼𝔾𝕀𝕊 employs a "Reactive Defense" strategy that makes the launch of attacks prohibitively expensive for attackers, requiring substantial computational or financial resources that scale with the intensity of the attack.

As part of the protocol and conditions for joining, the 𝔸𝔼𝔾𝕀𝕊 network enforces

a mandatory cost for each message sent on the network. Non-malicious nodes are automatically refunded the message fee, while malicious nodes are immediately penalized based on the type of attack they attempted. Every node has an associated multisignature Bitcoin wallet tied to the device and the authenticated device owner. However, the keys are not directly hosted on the device, but are represented in a light client fashion akin to Electrum. Upon a node being flagged as malicious by the targeting service and subsequently punished by Reactive Defense, they are placed into an isolated subnet awaiting the verdict after six sufficient confirmations by the Bitcoin network, where they are either permanently booted from the network or are refunded their full amount and allowed to rejoin.

𝔸𝔼𝔾𝕀𝕊 enables this high resource attrition for attackers by imposing a set of Hashcash-inspired cost functions on malicious actors Back et al. 2002. Upon the targeting service detecting malicious activity, two reactive defense measures are currently utilized based on the severity and type of threat:

- **Exponentially Sliding Punitive Cost Function**: Designed to counter throughput of messages beyond the rate limit, such as in DDoS attacks, by increasing the message fee.
- **Punitive Slashing Condition**: Designed to counter abnormal single messages, such as payloads of unusual size with malware (such as a Trojan) attached through collateral.

𝔸𝔼𝔾𝕀𝕊 provides an effective deterrent and cost-prohibitive defense posture to both opportunistic and strategic attackers through a series of adaptive cost functions tailored to the network's heterogeneous and dynamic environment. This includes devices ranging from high-capacity cloud servers to low-power IoT devices, each contributing differently to the network's security posture. The defense mechanism

adapts dynamically to current network conditions, such as node churn and transaction rates, adjusting the difficulty of cryptographic challenges accordingly. This adaptability ensures that the defense mechanism remains effective even as network conditions change rapidly, which is typical in environments with high churn. The equations detailing our adapted proof-of-work and punitive cost functions for our unique operational requirements are detailed below:

4.3.0.1 Dynamic Difficulty Adjustment

A modified version of the classical dynamic difficulty adjustment formula is designed to adapt the mining difficulty based on the rate of transactions and the current network load to ensure computational feasibility for IoT devices:

$$D(t) = D_0 \cdot \left(1 + \alpha \left(\frac{\overline{\lambda}(t)}{\lambda_{\text{ref}}}\right)\right) \tag{4.5}$$

where:

- $D(t)$: Difficulty at time t.
- D_0: Base difficulty.
- α: Adjustment factor, which scales the difficulty based on network conditions.
- $\overline{\lambda}(t)$: Average transaction rate at time t.
- λ_{ref}: Reference transaction rate for normal operation.

4.3.0.2 Churn Factor for Dynamic Difficulty

A new formula where the churn factor adjusts the difficulty in response to the rate of node churn in the network, reducing the difficulty to accommodate sudden drops in

network participation, as long as the network size stays within a sufficient range to be sufficiently resilient against byzantine faults:

$$\text{ChurnFactor}(t) = \exp\left(-\beta \cdot \left|\frac{d|\mathcal{N}_i(t)|}{dt}\right|\right) \tag{4.6}$$

where:

- β: Sensitivity parameter that modulates the effect of churn.
- $\mathcal{N}_i(t)$: Number of active nodes in the network at time t.

4.3.0.3 Stair-Stepping Difficulty Levels

A modified version of the classic stair-stepping algorithm provides more gradual changes in difficulty to prevent large fluctuations and maintain stability:

$$D(t + \Delta t) = D(t) \cdot \left(1 + \gamma \cdot \text{Sign}\left(\frac{\Delta\overline{\lambda}}{\Delta t}\right) \cdot \text{ChurnFactor}(t)\right) \tag{4.7}$$

where:

- Δt: Time increment for difficulty adjustment.
- γ: Step size for difficulty adjustment.
- $\Delta\overline{\lambda}$: Change in the average transaction rate.

4.3.0.4 Probabilistic and Bounded Cost Functions

This modified function accounting for churn ensures that the computational cost remains within a feasible range while still being probabilistic:

$$P(t) = \frac{1}{1 + \exp\left(-\xi\left(\overline{\lambda}(t) - \lambda_{\text{target}}\right)\right)} \tag{4.8}$$

$$\text{Cost}(t) = \text{BaseCost} \cdot \left(1 - \frac{\text{ChurnFactor}(t)}{\theta}\right) \tag{4.9}$$

where:

- $P(t)$: Probabilistic cost function at time t.
- ξ: Factor controlling the sensitivity to deviations from the target rate λ_{target}.
- θ: Normalization factor to ensure the cost stays within bounds.

Chapter 5

SECURITY ANALYSIS

5.1 Countermeasures

𝔸𝔼𝔾𝕀𝕊 employs a multi-layered defense-in-depth strategy to counter a variety of attacks. Table 1 summarizes specific attacks according to the MITRE ATT&CK Framework Strom et al. 2018 and the corresponding countermeasures implemented by 𝔸𝔼𝔾𝕀𝕊.

As a permissioned overlay network, 𝔸𝔼𝔾𝕀𝕊 requires all peers to register their IPs and link their nodes to their Bitcoin wallet as collateral. Each message incurs a nominal cost (expressible in Satoshis), refunded to honest nodes but forfeited by malicious actors, with penalties escalating upon malicious activity. This cost mechanism imposes a hard limit on the number of attempts an attacker can make, as insufficient funds result in a permanent ban from the network. To protect edge nodes, 𝔸𝔼𝔾𝕀𝕊 utilizes hardened fog-layer RSUs as bastion nodes, preventing unauthorized external communications. Connections within subnets are modeled as Kleinberg small-world networks in the worst-case to maintain connectivity under adverse conditions, while purely peer-to-peer in the subnet in the best case. Nodes generate one-time IP addresses for each session for anonymity. The brief operational lifespan of subnets (15 to 30 seconds), combined with the randomness introduced by ERIS for Moving-Target Defense Cai et al. 2016, makes exploitation unlikely and costly. Honeynets are interspersed within 𝔸𝔼𝔾𝕀𝕊 to quarantine and isolate compromised subnets to mitigate damage.

Table 1. AEGIS Countermeasures Against Specific Attacks

Type of Attack (MITRE ATTACK ID)	**AEGIS Countermeasure**
Eavesdropping (T1430)	Uses AES-256 encryption at rest and TLS 1.3 in transit with mutual authentication to ensure data confidentiality.
Sybil Attacks (T1098)	Requires cryptographic staking tied to device identity; high resource costs deter fake identities.
Man-in-the-Middle Attacks (T1557)	Utilizes TLS 1.3 with mutual authentication; dual-consensus detects anomalies; ERIS reduces predictability.
Replay Attacks (T1003)	Implements time-stamped messages and nonces; dual-consensus validates freshness; ATHENA monitors patterns.
Message Tampering (T1565)	Uses digital signatures and integrity checks; consensus mechanisms detect alterations; ATHENA responds.
Wormhole Attacks (T1430)	ERIS's dynamic subnet formation hinders wormholes; ATHENA detects routing anomalies.
Blackhole Attacks (T1499)	Dual-consensus identifies malicious nodes; ATHENA quarantines them; reroutes communications.
Jamming Attacks (T1495)	Detects communication disruptions; devices switch frequencies or use alternatives when possible.
Spoofing Attacks (T1556)	Employs PKI with RSA 2048-bit encryption and device certificates to prevent impersonation.
DoS and DDoS Attacks (T1498)	Adaptive rate limiting and resource metering; high-attrition defense increases attackers' costs.
Routing Attacks (T1592)	ERIS prevents routing manipulation; dual-consensus validates routing; ATHENA detects anomalies.
Side-Channel Attacks (T1407)	Implements constant-time cryptography; isolates sensitive operations; hardware security modules used.

However, 𝔸𝔼𝔾𝕀𝕊 acknowledges limitations in defending against certain scenarios, such as sophisticated supply chain attacks that compromise hardware or software integrity before deployment. To mitigate such risks, 𝔸𝔼𝔾𝕀𝕊 incorporates redundancy and rate limiting, and utilizes ERIS to enhance system resilience through multiple, overlapping security groups for verification. This layered approach ensures operational continuity even under attack. 𝔸𝔼𝔾𝕀𝕊 also has very little protection against zero-day vulnerabilities at the protocol level besides rapid patch dissemination through ATHENA.

5.2 Proofs of Correctness

The general AEGIS protocol that powers the network is composed of four subprotocols that occur in chronological order, which include the Subscription, Handshake, Dual Consensus, and the Heartbeat Protocols. The Alloy analyzer is a formal verification model-checking tool that is used to prove the correctness of each subprotocol, represented in the Alloy language Jackson 2019. For example, Figure 17 in the appendix contains a snippet of the Alloy model of the Heartbeat protocol. In the main AEGIS protocol and every subprotocol modeled through Alloy, no counterexamples designating insecurity were found, proving its correctness.

Chapter 6

EVALUATION

6.1 Experimental Setup

To evaluate 𝔸𝔼𝔾𝕀𝕊 within a realistic Cyber-Physical System (CPS) environment, we combined the simulation of urban mobility (SUMO) and the Cisco Packet Tracer, as depicted in Figure 6. This setup aimed to replicate a smart city scenario featuring vehicular Mobile Ad-hoc Networks (MANETs) and a broad array of connected devices, using SUMO to simulate the dynamics of Connected Autonomous Vehicles (CAVs) and an ns3 module for broader smart city network components. These simulations generated extensive sensor data from CAV interactions and movements, integrating vehicular and other smart city devices into a unified network model.

?In implementing the 𝔸𝔼𝔾𝕀𝕊 system, we integrated SUMO sensor data into the 𝔸𝔼𝔾𝕀𝕊 network built in Python, managed via Docker and Kubernetes, to simulate digital twins of each network node. Given the limitations of Cisco Packet Tracer, this integration of passing SUMO sensor data into it to perform calculations was crucial. The experimental network included node wallets, modified Kademlia distributed hash tables, and communication protocols, operating across a network of Raspberry Pi 4's and a GPU Rig, and small-scale connected Autonomous Vehicles engineered from retrofitted RC cars, as shown in Figures 7 and 8. This arrangement tested our adaptive, high-attrition defense mechanisms, providing insights into the real-world resource costs associated with our security strategies, thereby demonstrating 𝔸𝔼𝔾𝕀𝕊's functionality in managing security across a heterogeneous and dynamic CPS network.

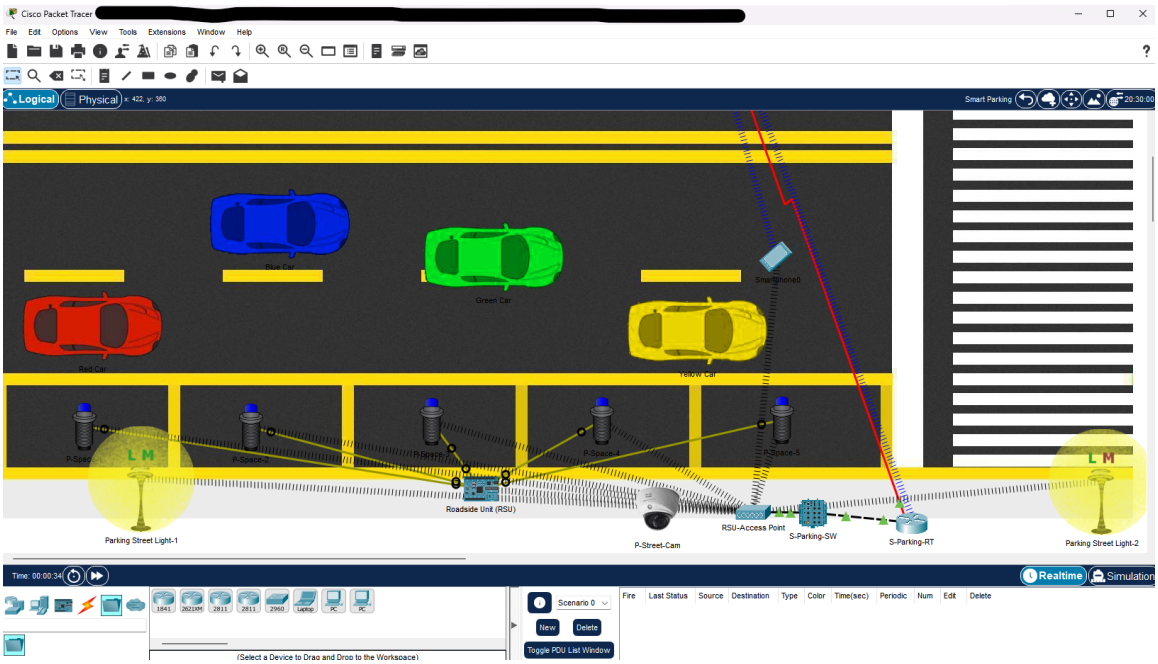

Figure 6. AEGIS subnet setup in Cisco Packet Tracer For high-scale, heterogeneous network emulation.

6.2 Result: AEGIS Has High Resilience

Our experiments show that Entropically Randomized Interconnected Subnets (ERIS) within the AEGIS framework have markedly enhanced the network's resilience. ERIS increases network resilience and lowers the probability of a byzantine fault occurring. In Figure 9, analyzing resilience over multiple test scenarios revealed:

- **Total Failures versus Total Recoveries:** The network experienced 369 failures, with ERIS enabling 366 recoveries, demonstrating a **99.2% recovery rate**. This underscores ERIS's role in rapid system recovery from rare disruptions, with disruptions only succeeding in scenarios where the attacker has committed unreasonably high resources and got lucky across multiple, statistically independent prerequisites.

Figure 7. Cluster of Raspberry Pi 4's used to model the fog layer (left) and GPU rig used to model the cloud layer (right).

Figure 8. Bastion fog-layer RSU and local edge-layer CAV modeling heterogeneous devices in a combined setting.

- **Mean Time to Recovery (MTTR):** The average MTTR was **9.73 seconds per subnet**, highlighting ERIS's efficiency in minimizing system downtime and ensuring continuity.

Key mechanisms by which ERIS enhances resilience include:

1. **Dynamic Subnet Reconfiguration:** ERIS continually adapts the network topology in response to threats and failures, mitigating the risk of exploits that leverage static configurations.

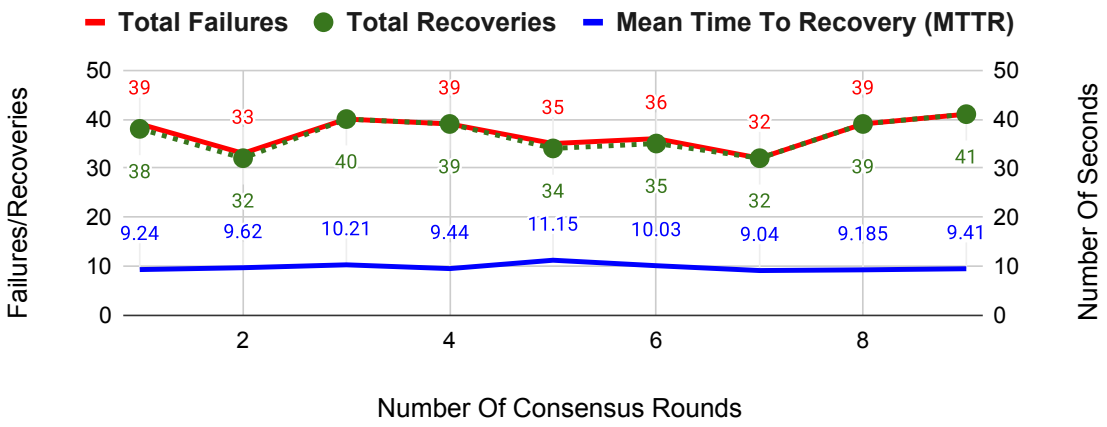

Figure 9. Resiliency and Recovery metrics over various consensus rounds utilizing ERIS.

2. **Entropy Maximization:** By randomizing network connections, ERIS complicates attacker efforts to predict or map network behaviors, denying critical intelligence.
3. **Fault Tolerance Integration:** ERIS's design inherently incorporates fault tolerance, preventing single points of failure from undermining the network.

ERIS improves AEGIS's security posture by embedding resilience into the network architecture. Our experimental data corroborate the theoretical advantages of ERIS and confirms its practical impact on maintaining high resilience levels. Through dynamic, entropy-enhanced networking, ERIS ensures the robustness and agility of the AEGIS framework against various cyber threats.

6.3 Result: AEGIS Quickly Detects Attackers

The dual-consensus model of the AEGIS network, incorporating both Byzantine One Shot Consensus (BOSCO) and Proof of Work (PoW) elements, was utilized

Table 2. Comparative Analysis of Time-to-Finality

Consensus Mechanism	Time-to-Finality (Seconds)
AEGIS	0.3 to 5 seconds
Honey Badger BFT	1 to 3 seconds
IOTA	10 seconds
Hashgraph	3 to 5 seconds

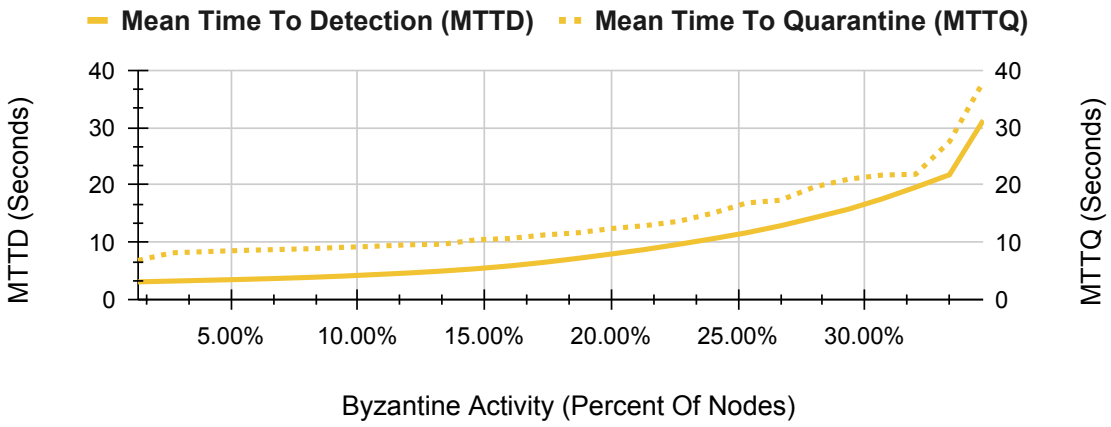

Figure 10. As the number of byzantine nodes is scaled in a 50 node network, AEGIS Mean Time To Detection (MTTD) and Mean Time To Quarantine (MTTQ) of the network increases, however the network remains effective at removing threats until the 33% byzantine fault tolerance (3f + 1) threshold.

to detect and disseminate threat detection information and post-incident threat intelligence. Specifically, the BOSCO component of this model has demonstrated exceptional performance in achieving time-to-finality, a critical metric in consensus mechanisms that measures the speed at which a network can reach a consensus on its state. The experimental data in Figure 10 revealed that BOSCO achieved time-to-finality within a remarkably swift range of **0.3 to 5 seconds**, depending on the network's scale, latency, and level of Byzantine activity. Comparative analysis of time-to-finality between consensus mechanisms compatible with IoT/CPS is highlighted in Table 2.

Table 3. Comparative Analysis of 𝔸𝔼𝔾𝕀𝕊 vs. Hashcash.

Metric	𝔸𝔼𝔾𝕀𝕊	Hashcash	% Diff.	Ratio
Attempts	15.35	1.48M	-99.999%	1.04×10^{-5}
Elapsed Time (s)	1.33×10^{-5}	0.964	-99.999%	1.38×10^{-5}
Hashpower Util.	594.38	1.54M	-99.961%	3.86×10^{-4}
Kilowatt-Hours (kWh)	1.84×10^{-11}	1.34×10^{-6}	-99.999%	1.38×10^{-5}
Cost (USD, $0.13/kWh)	2.40×10^{-12}	1.74×10^{-7}	-99.999%	1.38×10^{-5}
Cost (BTC)	5.99×10^{-17}	4.35×10^{-12}	-99.999%	1.38×10^{-5}

6.4 Result: 𝔸𝔼𝔾𝕀𝕊 Is Optimized For CPS and Imposes Severe Costs On Attackers

Our comprehensive evaluations of the 𝔸𝔼𝔾𝕀𝕊 network's adaptive, high-attrition defense mechanism, utilizing novel adaptations of cost functions inspired by Adam Back's hashcash Back et al. 2002, demonstrate significant enhancements in performance and resource utilization for compatibility with IoT and CPS networks. This approach dynamically adjusts the computational and economic costs imposed on nodes based on their behavior, based on whether they are flagged as honest or malicious; the former only optimal operating costs with the latter bearing exponentially cost-prohibitive ones. The experimental results are as follows:

- **Comparative Performance to Hashcash:** As outlined in Table 3, 𝔸𝔼𝔾𝕀𝕊 is optimized for IoT & CPS, with a **99.998%** time reduction and **99.999%** energy cost reduction compared to Hashcash in normal, non-Byzantine operation.
- **Adaptation to Attack Behavior**: During tests, nodes that started with high message rates (simulating a DDOS attack) adjusted their message sending rates in response to increased costs, demonstrating a drop in message traffic by **over 90%** within seconds of punitive cost adjustments.

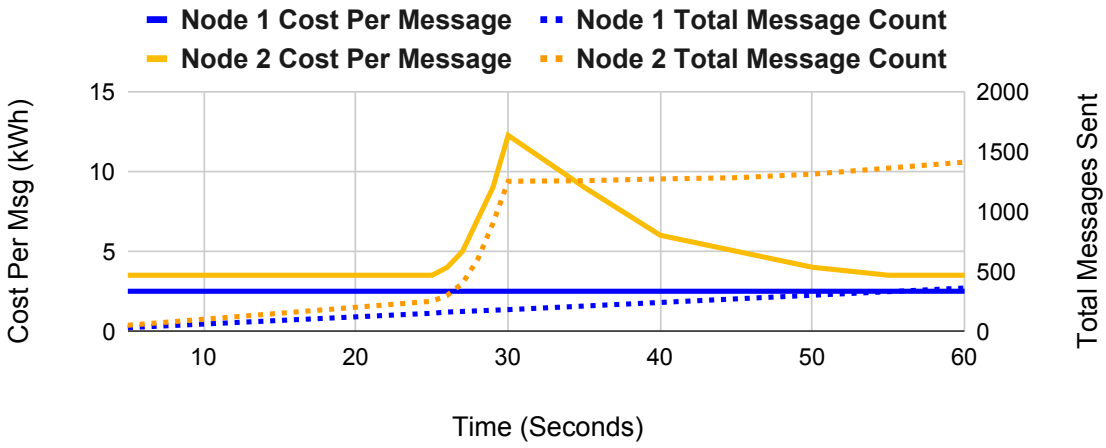

Figure 11. This figure depicts a situation where a node accidentally begins what looks like a Distributed Denial of Service (DDOS) attack, but then realizes the mistake and ceases the action. We can see that AEGIS reacts quickly by increasing the cost per message for the node, until it rolls back the changes after the node is compliant with the rules again.

Furthermore, the empirical data shown in Figures 11, 12, and 13 provide compelling evidence of AEGIS's ability to achieve rigorous cyber defense. AEGIS's strategy of imposing escalating costs on malicious activities forces attackers to expend resources at a rate that mirrors the defensive efforts of the network, thereby making the attacker's efforts economically unfeasible. This cost escalation is a direct result of AEGIS's dynamically adjusting the resource demands on nodes based on their behavior, thereby imposing a prohibitive financial and computational burden on attackers.

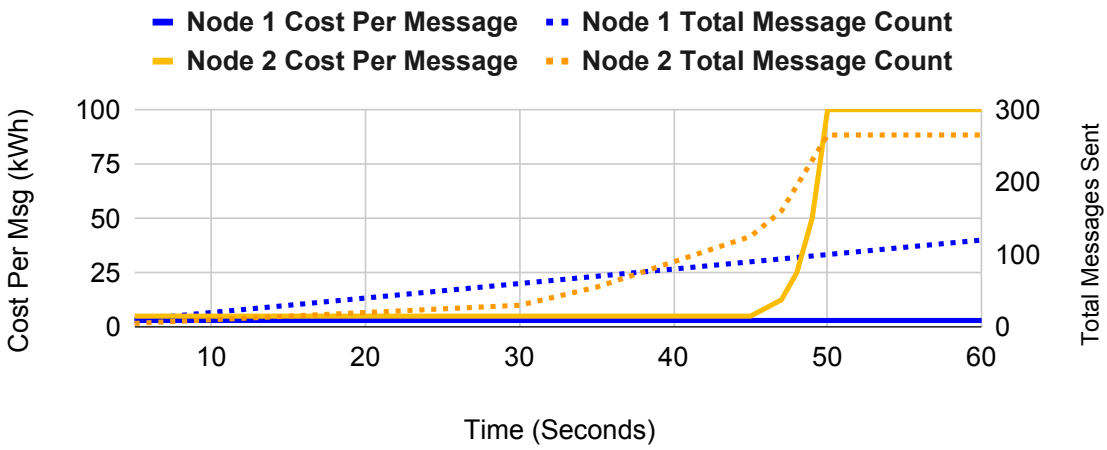

Figure 12. This graph depicts a scenario where a node actually attempts to perform a DDOS attack, and is quarantined. The node at first behaves, but then begins a DDOS attack at the 35-second mark. AEGIS reacts by increasing the message cost until, eventually, the node can no longer afford to send a message and ceases traffic.

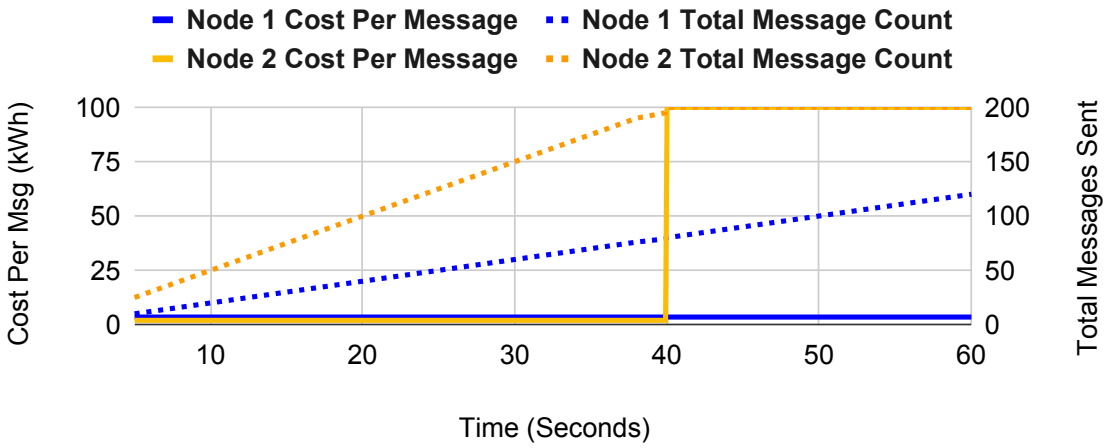

Figure 13. This graph depicts a scenario where a node attempts a zip bomb, and gets caught. At the 35 second mark, after playing normal for a long time, the node attempts to send the zip bomb but is caught by the heuristics within the MANET, and due to AEGIS node costs to send a message are increased such that the node is effectively quarantined in the network.

Chapter 7

NOVELTY AND LIMITATIONS OF AEGIS

AEGIS represents a transformative advance in network security, specifically within CPSs involving connected devices like autonomous vehicles and smart city IoT frameworks. Its novelty stems from utilizing the intrinsic dynamics of these systems to bolster security measures. Here is a summary of how AEGIS enhances network security:

- **Entropically Randomized Integrated Subnets (ERIS)**: Unlike prior-art solutions that have not harnessed churn, AEGIS leverages physical MANET and virtual WANET churn, inherently unpredictable & dynamic subnet creation & destruction, and limited subnet lifespan as a form of Moving-Target Defense (MTD). By dynamically reconfiguring subnets and masking node identities, it greatly reduces attack predictability and employs entropy maximization to severely reduce useful reconnaissance and the probability of a successful byzantine fault.
- **Autonomous Threat Handling and Engagement Network Application (ATHENA)**: This system introduces a novel dual-consensus model that significantly enhances threat detection and response. It uniquely combines rapid autonomous decisions at the local level with robust blockchain-based confirmations, drastically reducing response times and minimizing the incidence of false positives, thereby maintaining a highly adaptive and resilient defense mechanism.
- **Adaptive High-Attrition Defense Mechanism**: AEGIS implements an adaptive cost function optimized for complex, heterogeneous IoT and CPS

networks. It adjusts costs based on network behavior and threat levels, being optimal for honest lightweight nodes but imposing prohibitive costs on attackers and deterring prolonged malicious activities by making them economically and operationally unsustainable.

Generally, AEGIS introduces novel changes to CAV & Smart City cyber defense by harnessing randomness for a unique form of Moving-Target Defense and adaptive, high-cost deterrence. This integrated approach secures large-scale, heterogeneous networks prone to dynamic changes. The system's capacity to seamlessly integrate these elements represents more optimal approaches over traditional methods, which often struggle to adapt to and effectively secure rapidly evolving network environments.

Furthermore, while AEGIS marks a significant advancement in securing CAVs & Smart Cities, we acknowledge the potential for further enhancements and its limitations. Future developments could focus on further variations in responsibilities for different classes of devices by capability and enhancing its threat prediction capabilities, or more lightweight consensus mechanisms with a higher fault tolerance. In addition, AEGIS is a special-purpose network designed with specific assumptions regarding CPS network topology in more modernized, IoT-centric use-cases, but may not be applicable for antiquated, legacy SCADA systems with no contemporary internet connectivity or command-and-control, or ability for a retrofit.

Chapter 8

CONCLUSION

The development and evaluation of AEGIS marks a significant advancement in the field of Cyber-Physical System (CPS) security, specifically for CAVs & Smart Cities. Throughout this paper, we have detailed the approach of AEGIS, focusing on its dual-consensus model, decentralized and heuristic-based targeting service, and unique design optimized to the large-scale and heterogeneous nature of CAV & Smart City CPS networks. The experimental results, obtained from a sophisticated simulation setup using SUMO, Cisco Packet Simulator, Alloy Analyzer, and the AEGIS implementation in Python, reinforce the effectiveness of AEGIS in a realistic smart city scenario, demonstrating its robustness and scalability in managing the security of complex CPS environments. In conclusion, AEGIS represents a comprehensive, scalable, and effective approach to securing public-sector CAV & Smart City CPS networks while enabling rigorous cyber defense compared to prior-art solutions. Its design and successful validation pave the way for future research and development, with the potential to significantly enhance the security posture of CAV & Smart City CPS networks.

REFERENCES

ALSABARY, Bahaa Adnan. 2017. "Cyber Wars: Asymmetry in Threat." *Uluslararası Kriz ve Siyaset Araştırmaları Dergisi* 1 (2): 11–32.

Alviano, Mario. 2023. "Hashcash Tree, a Data Structure to Mitigate Denial-of-Service Attacks." *Algorithms* 16 (10): 462.

Back, Adam, et al. 2002. "Hashcash-a denial of service counter-measure."

Basta, Nardine, Muhammad Ikram, Mohamed Ali Kaafar, and Andy Walker. 2022. "Towards a zero-trust micro-segmentation network security strategy: an evaluation framework." In *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium,* 1–7. IEEE.

Bazrafshan, Zahra, Hashem Hashemi, Seyed Mehdi Hazrati Fard, and Ali Hamzeh. 2013. "A survey on heuristic malware detection techniques." In *The 5th Conference on Information and Knowledge Technology,* 113–120. IEEE.

Blocki, Jeremiah, and Anupam Datta. 2016. "CASH: A cost asymmetric secure hash algorithm for optimal password protection." In *2016 IEEE 29th Computer Security Foundations Symposium (CSF),* 371–386. IEEE.

Bodkhe, Umesh, Dhyey Mehta, Sudeep Tanwar, Pronaya Bhattacharya, Pradeep Kumar Singh, and Wei-Chiang Hong. 2020. "A survey on decentralized consensus mechanisms for cyber physical systems." *IEEE Access* 8:54371–54401.

Cai, Gui-lin, Bao-sheng Wang, Wei Hu, and Tian-zuo Wang. 2016. "Moving target defense: state of the art and characteristics." *Frontiers of Information Technology & Electronic Engineering* 17 (11): 1122–1153.

Cárdenas, Alvaro A, Tanya Roosta, Gelareh Taban, and Shankar Sastry. 2008. "Cyber security basic defenses and attack trends." *Homeland Security Technology Challenges,* 73–101.

Cassottana, Beatrice, Muhammad M Roomi, Daisuke Mashima, and Giovanni Sansavini. 2023. "Resilience analysis of cyber-physical systems: A review of models and methods." *Risk Analysis.*

Cervini, James, Aviel Rubin, and Lanier Watkins. 2022. "Don't drink the cyber: Extrapolating the possibilities of Oldsmar's water treatment cyberattack." In *International conference on cyber warfare and security,* 17:19–25. 1. Academic Conferences International Limited.

Chatziamanetoglou, Dimitrios, Konstantinos Rantos, et al. 2023. "Blockchain-Based Cyber Threat Intelligence Sharing Using Proof-of-Quality Consensus." *Security and Communication Networks* 2023.

Cybenko, George, and Roger Hallman. 2021. "Resilient Distributed Adaptive Cyber-Defense Using Blockchain." *Game Theory and Machine Learning for Cyber Security,* 485–498.

Galliott, Jai. 2016. "Cyber warfare, asymmetry, and responsibility: Considerations for defence theorem." In *Handbook of research on civil society and national security in the era of cyber warfare,* 1–21. IGI Global.

Geers, Kenneth. 2010. "The challenge of cyber attack deterrence." *Computer Law & Security Review* 26 (3): 298–303.

Gong, Seonghyeon, and Changhoon Lee. 2020. "Blocis: blockchain-based cyber threat intelligence sharing framework for sybil-resistance." *Electronics* 9 (3): 521.

Huang, Junqin, Linghe Kong, Guihai Chen, Long Cheng, Kaishun Wu, and Xue Liu. 2019. "B-IoT: Blockchain driven Internet of Things with credit-based consensus mechanism." In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS),* 1348–1357. IEEE.

Humayed, Abdulmalik, Jingqiang Lin, Fengjun Li, and Bo Luo. 2017. "Cyber-physical systems security—A survey." *IEEE Internet of Things Journal* 4 (6): 1802–1831.

Jackson, Daniel. 2019. "Alloy: a language and tool for exploring software designs." *Communications of the ACM* 62 (9): 66–76.

Keshk, Marwa, Elena Sitnikova, Nour Moustafa, Jiankun Hu, and Ibrahim Khalil. 2019. "An integrated framework for privacy-preserving based anomaly detection for cyber-physical systems." *IEEE Transactions on Sustainable Computing* 6 (1): 66–79.

Lee, Suhyeon, and Seungjoo Kim. 2021. "Blockchain as a cyber defense: opportunities, applications, and challenges." *Ieee Access* 10:2602–2618.

Liang, Hao, Li Zhu, F Richard Yu, and Xuan Wang. 2022. "A cross-layer defense method for blockchain empowered CBTC systems against data tampering attacks." *IEEE Transactions on Intelligent Transportation Systems* 24 (1): 501–515.

Lu, Wenlian, Shouhuai Xu, and Xinlei Yi. 2013. "Optimizing active cyber defense." In *Decision and Game Theory for Security: 4th International Conference, GameSec*

2013, Fort Worth, TX, USA, November 11-12, 2013. Proceedings 4, 206–225. Springer.

Ma, Xingbang, Dongsheng Yu, Yanhui Du, Lanting Li, Wenkai Ni, and Haibin Lv. 2023. "A Blockchain-Based Incentive Mechanism for Sharing Cyber Threat Intelligence." *Electronics* 12 (11): 2454.

Moniz, Henrique, Nuno F Neves, and Miguel Correia. 2012. "Byzantine fault-tolerant consensus in wireless ad hoc networks." *IEEE Transactions on Mobile Computing* 12 (12): 2441–2454.

Mullender, Sape. 1990. *Distributed systems.* ACM.

Perlroth, Nicole. 2021. *This is how they tell me the world ends: The cyberweapons arms race.* Bloomsbury Publishing USA.

Petit, Jonathan, and Steven E Shladover. 2014. "Potential cyberattacks on automated vehicles." *IEEE Transactions on Intelligent transportation systems* 16 (2): 546–556.

Purohit, Soumya, Prasad Calyam, Songjie Wang, RajaniKanth Yempalla, and Justin Varghese. 2020. "Defensechain: Consortium blockchain for cyber threat intelligence sharing and defense." In *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS),* 112–119. IEEE.

Radanliev, Petar, Dave De Roure, Jason RC Nurse, Razvan Nicolescu, Michael Huth, Stacy Cannady, and Rafael Mantilla Montalvo. 2018. "Integration of cyber security frameworks, models and approaches for building design principles for the internet-of-things in industry 4.0." In *Living in the Internet of Things: Cybersecurity of the IoT-2018,* 1–6. IET.

Rajhans, Akshay, Ajinkya Bhave, Ivan Ruchkin, Bruce H Krogh, David Garlan, André Platzer, and Bradley Schmerl. 2014. "Supporting heterogeneity in cyber-physical systems architectures." *IEEE Transactions on Automatic Control* 59 (12): 3178–3193.

Rid, Thomas, and Ben Buchanan. 2015. "Attributing cyber attacks." *Journal of Strategic Studies* 38 (1-2): 4–37.

Robinson, Michael, Kevin Jones, and Helge Janicke. 2015. "Cyber warfare: Issues and challenges." *Computers & security* 49:70–94.

Rot, Artur, and Bartosz Blaicke. 2019. "Blockchain's future role in cybersecurity. analysis of defensive and offensive potential leveraging blockchain-based platforms." In *2019 9th International Conference on Advanced Computer Information Technologies (ACIT),* 447–451. IEEE.

Ruan, Na, and Yoshiaki Hori. 2012. "DoS attack-tolerant TESLA-based broadcast authentication protocol in Internet of Things." In *2012 International Conference on Selected Topics in Mobile and Wireless Networking,* 60–65. IEEE.

Singh, Debabrata, Bibudhendu Pati, Chhabi Rani Panigrahi, and Shrabanee Swagatika. 2020. "Security issues in IoT and their countermeasures in smart city applications." In *Advanced Computing and Intelligent Engineering: Proceedings of ICACIE 2018, Volume 2,* 301–313. Springer.

Song, Houbing, Glenn A Fink, and Sabina Jeschke. 2017. *Security and privacy in cyber-physical systems: foundations, principles, and applications.* John Wiley & Sons.

Song, Yee Jiun, and Robbert van Renesse. 2008. "Bosco: One-step byzantine asynchronous consensus." In *Distributed Computing: 22nd International Symposium, DISC 2008, Arcachon, France, September 22-24, 2008. Proceedings 22,* 438–450. Springer.

Strom, Blake E, Andy Applebaum, Doug P Miller, Kathryn C Nickels, Adam G Pennington, and Cody B Thomas. 2018. "Mitre att&ck: Design and philosophy." In *Technical report.* The MITRE Corporation.

Todd, Graham H. 2009. "Armed attack in cyberspace: deterring asymmetric warfare with an asymmetric definition." *AFL Rev.* 64:65.

Wang, Shenyquan, Dong Xuan, and Wei Zhao. 2003. "On resilience of structured peer-to-peer systems." In *GLOBECOM'03. IEEE Global Telecommunications Conference (IEEE Cat. No. 03CH37489),* 7:3851–3856. IEEE.

Wu, Guangyu, Jian Sun, and Jie Chen. 2016. "A survey on the security of cyber-physical systems." *Control Theory and Technology* 14:2–10.

APPENDIX A

ALLOY IMPLEMENTATIONS

A.1 AEGIS Subscription Alloy Implementation

The Alloy model depicted in the Figure 14 provides a formal verification framework for a Subscription Protocol. This protocol ensures that each user's device maintains at least one credential and is linked to a user-specific wallet, thereby enhancing security. The predicates such as LinkVerify and FinalizeDevices ensure that devices are properly linked to users and are active within the system. This formal approach using Alloy helps in identifying potential security flaws and ensures that the system adheres to specified security standards.

A.2 AEGIS Handshake Alloy Implementation

The AEGIS Handshake algorithm, as depicted in Figure 15, is designed to ensure secure communication within Mobile Ad-hoc Networks (MANETs) through a formal verification model. The handshake predicate (hShake) checks if the egoNode's hash matches the RSU's hash, filtering nodes based on this criterion and a predefined threshold for network formation. The formMANET predicate further ensures that all nodes within a MANET share the same targeting service code as the associated RSU, thereby asserting the validity of the network formation. This model is crucial for demonstrating the security properties of the handshake protocol in diverse and dynamic network environments.

A.3 AEGIS Dual Consensus Alloy Implementation

The Alloy model presented in Figure 16 outlines the Dual-Consensus protocol, which integrates elements of both BOSCO and Proof of Work (PoW) consensus mechanisms to enhance network security and efficiency. The model defines entities such as nodes, wallets, and transactions within a subnet. Nodes are characterized by their transaction sets and a Boolean status indicating whether they are malicious or nonresponsive. Transactions are simply marked valid or invalid. The BOSCO Consensus predicate focuses on identifying and flagging nodes with invalid transactions, applying abstract constraints to manage these nodes. Concurrently, the ProofOfWork predicate ensures all transactions are valid and differentiates node handling based on their malicious status, applying penalties or refunds through abstract mechanisms. This dual approach allows the protocol to robustly handle network discrepancies

and maintain integrity, as demonstrated through formal verification within the Alloy framework.

A.4 AEGIS Heartbeat Alloy Implementation

The Alloy model depicted in Figure 17 outlines the Heartbeat protocol, which is designed to enhance security in network communications. This model categorizes nodes into three distinct statuses: Honest, Malicious, and Nonresponsive, each represented by signatures extending a base 'Status' signature. The 'DualConsensusRound' signature captures the state of the network in each round, distinguishing between malicious, honest, and nonresponsive nodes. The model ensures that each node is uniquely categorized in each round, preventing overlap between categories. Furthermore, the model describes mechanisms for updating local RSU caches and Kademlia DHTs based on the consensus of node statuses, ensuring that all nodes are consistently accounted for across updates. By formalizing these processes, the model provides a robust framework for verifying the integrity and security of the Heartbeat protocol through formal methods.

```
module AEGISSubscription
sig User {
  devices: set Device,
  wallet: one Wallet
}

sig Device {
  credentials: some Credential,  // Each device must have at least one credential
  securityAssessment: one SecurityAssessment,
  topology: one Topology,
  isActive: Bool // Boolean to indicate if the device is active
}

sig Wallet {
  linkedUser: one User
}

sig Credential, SecurityAssessment, Topology {}

// Introducing a simple Boolean type for clarity
sig Bool {
  state: Int
}

// Ensure every active device must have credentials
fact DeviceCredentials {
  all d: Device | d.isActive.state = 1 implies some d.credentials
}

// Ensure each device with credentials is active
fact EnsureActiveDevices {
  all d: Device | some d.credentials implies d.isActive.state = 1
}

// Link wallet to user and verify credentials
pred LinkVerify[u: User, w: Wallet] {
  w.linkedUser = u
  all d: u.devices | some d.credentials
}

// Finalize devices in the system
pred FinalizeDevices[u: User] {
  all d: u.devices | d.isActive.state = 1
}

// Check the completeness of the protocol setup
pred CompleteProtocol {
  all u: User |
    all d: u.devices | d.isActive.state = 1 and
    some w: Wallet | w.linkedUser = u
}

// Run the model to find an instance where all conditions hold
run CompleteProtocol for 4 but 5 Device, 3 User, 3 Wallet
```

Figure 14. A snippet of an Alloy model of the Subscription protocol, proving security through formal verification

```
module AEGISHandshake
// Define Node, RSU, and Hash
sig Node {
    targetingServiceCode: one String,
    hashValue: one Hash
}
sig RSU {
    targetingServiceCode: one String,
    hashValue: one Hash
}
sig Hash {}
// Define a MANET which consists of Nodes and an associated RSU
sig MANET {
    nodes: set Node,
    associatedRSU: one RSU
}
// Define a threshold for forming a MANET
let threshold = 3 // This can be adjusted as needed

// Define the relations and constraints for the handshake protocol
pred hShake(egoNode: Node, nearbyNodes: set Node, rsu: RSU) {
    // Check if egoNode's hash matches the RSU's hash
    egoNode.hashValue != rsu.hashValue implies no MANET
    else {
        // Filter candidate nodes
        let candidateNodes = { n: nearbyNodes |
            n.hashValue = rsu.hashValue } |

        // Check if the number of candidate nodes exceeds
        // the threshold
        #candidateNodes > threshold implies some MANET
        else no MANET
    }
}
// Predicate to form a MANET only with nodes having
// the same targeting service code as the RSU
pred formMANET[manet: MANET] {
    all n: manet.nodes |  n.targetingServiceCode =
        manet.associatedRSU.targetingServiceCode
}
// Assert that all nodes in a MANET have the same
// targeting service code as the RSU
assert ValidMANETFormation {
    all manet: MANET | formMANET[manet]
}
// Check the assertion
check ValidMANETFormation for 4
// Define a run command for simulation
run hShake for 10
```

Figure 15. A snippet of an Alloy model of the Handshake protocol, proving security through formal verification

```
module AEGISDualConsensus
// Define basic entities
sig Node {
    wallet: one Wallet,
    transactions: set Transaction,
    isMalicious: one Boolean,
    isNonresponsive: one Boolean
}

sig Wallet {
    balance: Int
}

sig Transaction {
    isValid: one Boolean
}

sig Subnet {
    nodes: set Node,
    transactions: set Transaction
}

// Define the Boolean signature and its atoms
abstract sig Boolean {}
one sig True, False extends Boolean {}

// Define a predicate for the BOSCO consensus process
pred BOSCOConsensus(subnet: Subnet) {
    // Identify and flag violating nodes
    all n: subnet.nodes |
        n.isMalicious = if some t: n.transactions |
t.isValid = False then True else False
    // Apply MedusaStunlock to flagged nodes
    // Represented abstractly as a constraint here
}

// Define a predicate for the PoW process and transaction handling
pred ProofOfWork(subnet: Subnet) {
    // Validate transactions
    all t: subnet.transactions | t.isValid = True
    // Handle transactions for each node
    all n: subnet.nodes | {
        n.isMalicious = True implies handleMaliciousNode[n]
        n.isMalicious = False implies handleHonestNode[n]
    }
}

// Define a predicate for handling malicious nodes
pred handleMaliciousNode(node: Node) {
    // Apply ExponentiallySlidingCost to the malicious node's wallet
    // Abstract representation of the cost mechanism
}

// Define a predicate for handling honest nodes
pred handleHonestNode(node: Node) {
    // Refund fee to the honest node's wallet
    // Abstract representation of the refund mechanism
}

// Define a predicate for the overall DualConsensus protocol
pred DualConsensus(subnet: Subnet) {
    BOSCOConsensus[subnet]
    ProofOfWork[subnet]
}

run DualConsensus for 5 // Run the DualConsensus protocol
```


Figure 16. A snippet of an Alloy model of the Dual-Consensus protocol, proving security through formal verification

```
module AEGISHeartbeat
// Definitions of basic entities
sig Node {
    status: one Status,
    behaviorType: lone BehaviorType
}

abstract sig Status {}
sig Honest, Malicious, Nonresponsive extends Status {}

sig BehaviorType {}

// Represents the DualConsensus round data
sig DualConsensusRound {
    maliciousNodes: set Node,
    honestNodes: set Node,
    nonresponsiveNodes: set Node
}

// Local RSU Cache and Kademlia DHT
sig LocalRSUCache {
    nodes: set Node
}

sig KademliaDHT {
    nodes: set Node
}

// Updating nodes based on DualConsensus round
fact updateNodes {
    all dc: DualConsensusRound | {
        dc.maliciousNodes.status = Malicious
        dc.honestNodes.status = Honest
        dc.nonresponsiveNodes.status = Nonresponsive
        // Each node is in exactly one category per round
        no (dc.maliciousNodes & dc.honestNodes)
        no (dc.maliciousNodes & dc.nonresponsiveNodes)
        no (dc.honestNodes & dc.nonresponsiveNodes)
    }
}

// Process for updating caches and DHTs
pred updateCachesAndDHTs [dc: DualConsensusRound,
cache: LocalRSUCache, dht: KademliaDHT] {
    cache.nodes = dc.maliciousNodes + dc.honestNodes
    + dc.nonresponsiveNodes
    dht.nodes = cache.nodes
}

// Check if all nodes are accounted for after an update
assert AllNodesUpdated {
    all dc: DualConsensusRound, cache: LocalRSUCache,
    dht: KademliaDHT | updateCachesAndDHTs[dc, cache, dht] =>
    (cache.nodes = dc.maliciousNodes +
    dc.honestNodes + dc.nonresponsiveNodes) and
    (dht.nodes = cache.nodes)
}

// Run command for checking the assertion
check AllNodesUpdated for 4
```

Figure 17. A snippet of an Alloy model of the Heartbeat protocol, proving security through formal verification