Article Type: Magazine

A Closed-Loop Architecture for DDoS-Resilient 6G Network Slicing

Amir Javadpour, *ICTFICIAL Oy, Espoo, Finland*

Forough Ja'fari, *Department of Computer Engineering, Sharif University of Technology, Tehran, Iran*

Tarik Taleb, *Electrical Engineering and Information Technology, Ruhr University Bochum, Bochum, Germany*

Abstract—*Future 6G networks will rely on software-defined networking, virtualization, and network slicing, but overloaded NFVI nodes may disrupt multiple co-hosted services. This article proposes a closed-loop DDoS-resilience architecture integrating moving target defense, risk-aware slice placement, and an isolated HoneySlice for deception-based intelligence. The framework defines attacker classes and trust boundaries, models placement and reconfiguration as a constrained Markov decision process, and applies a safety shield to reinforcement-learning actions. A reproducible synthetic evaluation with 10 NFVI nodes, 18 slices, and 100 seeds compares static, periodic, isolated, rule-based, and RL-based defenses. Relative to static placement, the RL controller reduces criticality-weighted outage damage by 50.4%, high-criticality violations by 55.8%, and successful attack slots by 44.5%, with a normalized reconfiguration cost of 0.038 per slot. The results demonstrate controlled feasibility rather than operational 6G validation.*

Keywords—6G security, network slicing, distributed denial-of-service, software-defined networking, moving target defense, cyber deception, reinforcement learning.

The transition from conventional mobile infrastructures to softwarized 5G and 6G environments has changed the way availability and security must be understood. Network slicing enables multiple logical services to coexist on shared physical resources [1, 2, 3], while software-defined networking (SDN) introduces centralized visibility and fine-grained control [4]. Network function virtualization (NFV) further supports this softwarized model by decoupling network functions from dedicated hardware and enabling flexible service deployment [5]. Together, these capabilities improve agility, service differentiation, and resource utilization. At the same time, they also create new dependencies between orchestration logic, hosting resources, and slice continuity. As a result, a successful distributed denial-of-service (DDoS) attack may no longer remain a localized outage; it can propagate across multiple co-hosted services and degrade the overall stability of the network core [6, 7, 8, 9].

This shift is especially important in 6G-oriented infrastructures, where the control plane is expected to support highly dynamic service composition, low-latency applications, and heterogeneous mission profiles. In such settings, defense mechanisms must do more than detect malicious traffic. They must continuously reason about target exposure, service colocation, failure propagation, and operational cost. Recent 6G visions also emphasize AI-native orchestration, dense heterogeneity, and service-specific performance requirements, which make security automation and closed-loop control increasingly important [10]. This article therefore views DDoS security not as a single detection problem but as a broader resilience problem for programmable infrastructure.

The central thesis is that DDoS resilience in SDN-orchestrated slicing requires three coordinated functions with non-overlapping operational roles: (1) *exposure reduction*, which invalidates reconnaissance before flooding becomes effective; (2) *damage containment*, which limits correlated service loss after an attack reaches an NFVI node; and (3) *deception-derived intelligence*, which improves the risk estimate used by the first two functions. The term *closed-loop coordinated defense* is used throughout the article instead of *cross-layer defense*, because the mechanisms are interdependent security functions rather than protocol-stack layers.

The article makes three concrete contributions. First, it defines an element-specific threat model and a precise trust boundary for external bots, compromised tenant endpoints or virtual network functions (VNFs), and untrusted data-plane telemetry. Second, it provides a constrained optimization and Markov decision process (MDP) formulation, including capacity, latency, migration-budget, and safety constraints. Third, it reports a reproducible controlled evaluation and analyzes both adversarial attacks against the RL decision process and cascading failure across tightly coupled defense functions.

Figure 1 summarizes how the shared-infrastructure DDoS problem motivates coordinated exposure reduction, damage containment, and deception-supported intelligence. Figure 2 presents the corresponding high-level architectural view; the formal trust assumptions, constraints, and controller logic that make this architecture operational are specified in the subsequent sections.

XXXX-XXX © 2026 IEEE
Digital Object Identifier 10.1109/XXX.0000.0000000

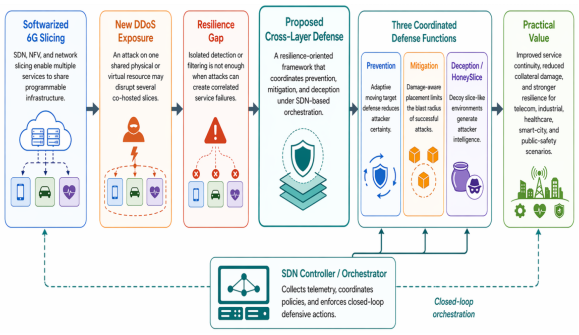


FIGURE 1. Conceptual roadmap of the article.

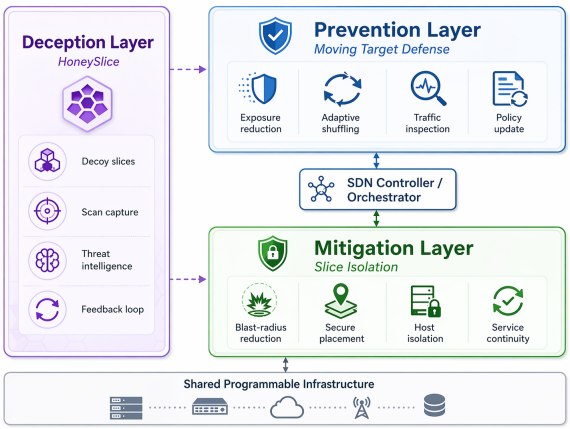


FIGURE 2. High-level coordinated defense architecture for securing SDN-based 6G network slicing against DDoS attacks.

Threat Model, Trust Boundary, and Problem Setting

The framework protects the programmable core and edge infrastructure of a slice-enabled 5G/6G system. Three attacker classes are considered. *A1: external bot-controlled endpoints* perform scanning and volumetric or application-layer flooding against public service endpoints. *A2: compromised tenant endpoints, containers, virtual machines, or service-facing VNFs* can generate attack traffic from inside the provider perimeter, exploit shared resources, or reveal placement information. *A3: compromised or faulty data-plane switches and telemetry exporters* can suppress, delay, or bias measurements used by the controller. These classes identify the malicious network elements explicitly rather than using the generic term *adversary*.

The trusted computing base comprises an authenticated and access-controlled SDN/NFV orchestration plane, an append-only policy and action log, and a quorum of monitored controller instances. NFVI compute nodes and switches are treated as potentially faulty and their telemetry is not accepted without source authentication, freshness checking, and cross-source agreement. Full compromise of the controller quorum, hypervisor root of trust, or cryptographic key-management service is outside the protection scope; such compromise requires platform-security mechanisms beyond DDoS resilience.

The principal protected assets are production slices, service endpoint identifiers such as virtual IP addresses, VNF instances, forwarding paths, and NFVI compute nodes. The phrase *compute host* or *NFVI node* is used when referring to the server that executes VNFs; *end host* is reserved for a traffic source or destination. The attacker may learn a stable service endpoint, infer a high-value NFVI node from repeated observations, and direct traffic toward that node. If it becomes saturated or unavailable, all slices mapped to it may experience correlated degradation. Network slicing therefore changes the attack blast radius: the attacker does not need to compromise every service independently if one shared resource carries several critical slices.

The architecture does not claim that endpoint mutation prevents every attack. Exposure reduction lowers the reliability of reconnaissance; damage containment limits service loss when exposure reduction fails; and HoneySlice interactions provide evidence that changes the confidence assigned to possible targets. These roles are complementary but operationally distinct.

Measurable Design Objectives

The earlier five-item list is consolidated into three measurable security objectives. Operational cost and SDN enforceability are retained as constraints rather than presented as independent security goals.

First, the architecture should *reduce target certainty*. Endpoint, route, or service-binding mutation should shorten the useful lifetime of reconnaissance information without causing uncontrolled churn. Second, it should *bound criticality-weighted damage*. Placement should limit the total criticality of slices that can fail together on one exposed NFVI node. Third, it should *extract reliable behavioral evidence*. HoneySlice observations should improve risk estimation only after source confidence and consistency checks.

All three objectives are subject to service-level, resource-capacity, latency, migration-budget, and controller-rate constraints. This formulation makes the trade-offs explicit and gives the orchestrator a basis for rejecting unsafe actions.

Coordinated Operational Functions and Intelligence

The proposed framework consists of two direct operational functions and one intelligence function. Exposure reduction and damage containment determine what the infrastructure changes; HoneySlice improves what the infrastructure knows. The SDN/NFV orchestrator maintains the slice-to-node mapping, resource and SLA state, and action history, while the safety shield converts a learned recommendation into an enforceable action or a safe no-op.

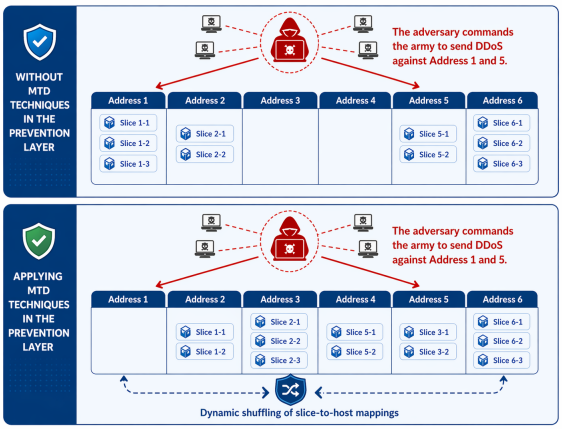


FIGURE 3. Exposure-reduction function: adaptive moving target defense reshapes target visibility and invalidates attacker reconnaissance.

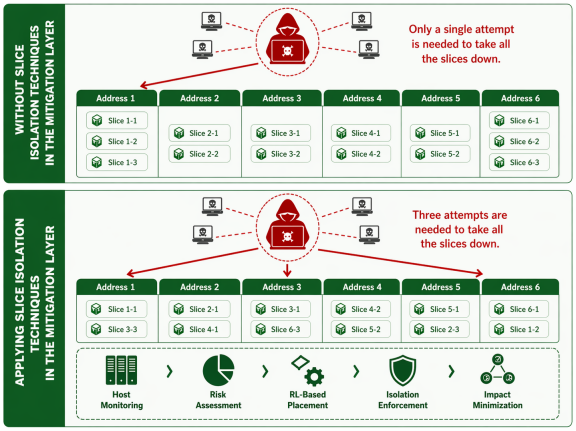


FIGURE 4. Damage-containment function: damage-aware placement reduces the number of slices affected by a successful attack against one NFVI node.

Exposure Reduction through Adaptive Moving Target Defense

Moving target defense (MTD) changes selected attacker-visible bindings—for example, a service virtual IP, forwarding path, ingress association, or VNF replica—so that reconnaissance becomes stale [16, 17]. Figure 3 illustrates this exposure-reduction mechanism. The action is preventive only in the limited sense that it decreases the probability of a correctly targeted flood; it does not guarantee that a flood will be stopped.

Frequent mutation may improve uncertainty but can also introduce control-plane load, transient path changes, state synchronization, and service interruption. The controller therefore selects mutation only when its estimated risk reduction exceeds the expected orchestration and SLA cost. Prior RL-driven MTD work provides a conceptual baseline [14, 13]; the present formulation adds slice criticality, shared-resource damage, and explicit safety constraints.

Damage Containment through Risk-Aware Slice Placement

Damage containment assumes that some floods reach a production endpoint and focuses on limiting their consequences. Its task is to map slices to NFVI nodes so that one overloaded node does not carry an excessive concentration of critical services. Figure 4 illustrates damage-aware placement and the reduction of simultaneous slice exposure when one hosting resource is attacked.

The mitigation objective is not simply to maximize utilization. It minimizes expected criticality-weighted damage while satisfying node capacity and service constraints. This turns slice placement into an adversarial virtual-network-embedding problem in which exposure and correlated failure complement conventional capacity and delay terms [15].

HoneySlice as an Isolated Deception and Evidence Function

A HoneySlice is a controlled, non-production slice-like environment that exposes plausible service metadata and decoy endpoints while remaining isolated from production slices. It has no route to production data stores, uses separate credentials and namespaces, and is subject to strict egress filtering. Its first role is diversion: some reconnaissance and flooding effort is directed toward a controlled target. Its second role is evidence generation: interactions are transformed into features such as scan sequence, source dispersion, protocol mix, inter-arrival behavior, target preference, and escalation timing.

The evidence path is explicitly separated from the enforcement path. HoneySlice observations are signed, timestamped, and assigned a confidence score. A single decoy event cannot directly trigger migration or mutation. The risk estimator combines decoy evidence with independent flow, utilization, and SLA telemetry; disagreement lowers confidence or causes the action shield to select a no-op. Figure 5 depicts the HoneySlice mechanism as a combination of controlled diversion and evidence generation rather than a generic monitoring function.

Formal System Model and Constrained Decision Process

Let the physical substrate be $G^P = (V^P, E^P)$, where $h \in V^P$ denotes an NFVI node with capacity C_h. Let $\mathcal{S}$ be the set of slices. Slice s has resource demand d_s, criticality $c_s > 0$, and maximum admissible latency $L_s^{\max}$. The binary variable $x_{sh}(t)$ equals one when slice s is placed on node h at slot t. The estimated probability that node h becomes an effective DDoS target is $p_h(t)$, derived from exposure, flow telemetry, and confidence-scored HoneySlice evidence.

The criticality-weighted blast radius of node h is

$$B_h(t) = \sum_{s \in \mathcal{S}} c_s x_{sh}(t), \tag{1}$$

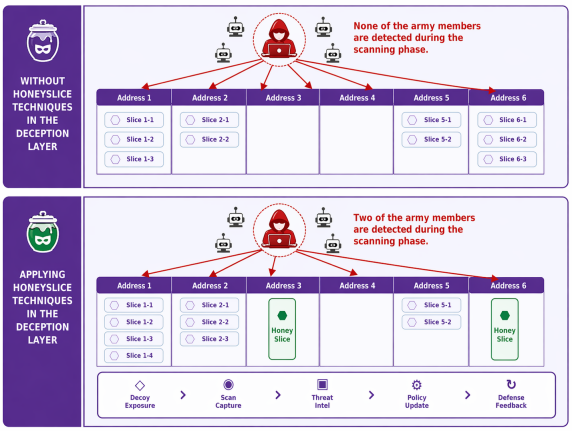


FIGURE 5. HoneySlice deception function: isolated decoys attract malicious interactions and generate intelligence for exposure reduction and damage containment.

and the expected one-slot outage damage is

$$D(t) = \sum_{h \in V^P} p_h(t) B_h(t). \tag{2}$$

A mutation action $m(t)$ changes one or more attacker-visible bindings and incurs cost $C_{\mathrm{MTD}}(t)$. A placement action incurs migration cost

$$C_{\mathrm{mig}}(t) = \sum_{s,h} \kappa_s \left| x_{sh}(t) - x_{sh}(t-1) \right|, \tag{3}$$

where κ_s reflects state size and service interruption sensitivity.

The controller minimizes discounted long-run damage and overhead,

$$\min_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{T} \gamma^t \left(\lambda_D D(t) + \lambda_M C_{\mathrm{mig}}(t) + \lambda_T C_{\mathrm{MTD}}(t) + \lambda_V V(t) \right) \right], \tag{4}$$

where $V(t)$ counts SLA violations and π is the control policy. The action must satisfy

$$\sum_{h} x_{sh}(t) = 1, \qquad \forall s, \tag{5}$$

$$\sum_{s} d_s x_{sh}(t) \le C_h, \qquad \forall h, \tag{6}$$

$$L_{sh} x_{sh}(t) \le L_s^{\max}, \qquad \forall s, h, \tag{7}$$

$$C_{\mathrm{mig}}(t) \le M^{\max}, \quad N_{\mathrm{mut}}(t) \le U^{\max}, \tag{8}$$

where $M^{\max}$ and $U^{\max}$ limit migration and mutation intensity.

The MDP state is

$$s_t = \{\mathbf{u}_t, \mathbf{x}_t, \hat{\mathbf{p}}_t, \mathbf{q}_t, \mathbf{v}_t, \mathbf{a}_{t-1}\}, \tag{9}$$

containing node utilization, placement, estimated exposure, telemetry confidence, SLA state, and recent actions. The action a_t is selected from no-op, endpoint/path mutation, risk-aware replacement, combined mutation and replacement, or HoneySlice rotation. The reward is the negative of the bracketed cost in Eq. (4), with a bounded bonus for verified diversion. A safety shield constructs the feasible action set $\mathcal{A}_{\mathrm{safe}}(s_t)$ and replaces any infeasible learned action with the least-cost safe alternative. Thus, RL recommends an action but does not bypass capacity, SLA, or rate constraints.

Executable Closed-Loop Controller

Figure 6 provides the operational view of the coordinated framework. The executable cycle below formalizes the information-validation, decision, safety-checking, enforcement, and feedback steps represented conceptually in that figure.

Safe coordinated decision cycle. (1) Authenticate and time-align flow, node, VNF, SLA, and HoneySlice telemetry. (2) Reject stale sources and compute source-agreement and anomaly scores. (3) Update $\hat{p}_h(t)$ and confidence $q_h(t)$ for each NFVI node. (4) Construct the MDP state and obtain the policy recommendation. (5) Apply the safety shield, cooldown, migration budget, and controller-rate limits. (6) Enforce the approved action through SDN/NFV interfaces and append the state, recommendation, approved action, and outcome to the audit log.

Reproducible Controlled Evaluation

The evaluation is a synthetic feasibility study designed to test the internal trade-offs of the architecture; it is not presented as a measurement of an operational 6G network. The accompanying Python program generates the workload, trains the tabular Q-learning controller, evaluates all baselines under identical random seeds, and exports the reported metrics.

Scenario and training data

Each episode contains 10 NFVI nodes and 18 slices. Slice demands are sampled from $[0.035, 0.085]$ of normalized node capacity, and criticality takes values $\{1, 2, 3, 5\}$ with probabilities $\{0.40, 0.30, 0.20, 0.10\}$. Benign node load evolves between 0.35 and 0.82 with periodic and stochastic variation. Attack campaigns follow four explicit phases: idle, reconnaissance, flood, and recovery. Flood intensity follows a clipped log-normal distribution over $[0.35, 2.2]$ normalized load units. Reconnaissance updates the attacker's target belief toward high-load and high-criticality-concentration nodes. HoneySlice attractiveness determines a bounded diversion probability and can also produce an early warning during reconnaissance.

These distributions are transparent modeling choices. They avoid privacy and confidentiality issues because no subscriber, operator, or packet payload is used. For deployment-oriented training, the same generator should be calibrated with privacy-preserving aggregates from public DDoS traces, operator-approved digital-twin measurements, or federated statistics; evaluation should then reserve attacks and operating regimes not used for training.

The RL state discretizes criticality concentration, utilization pressure, verified risk indication, and action

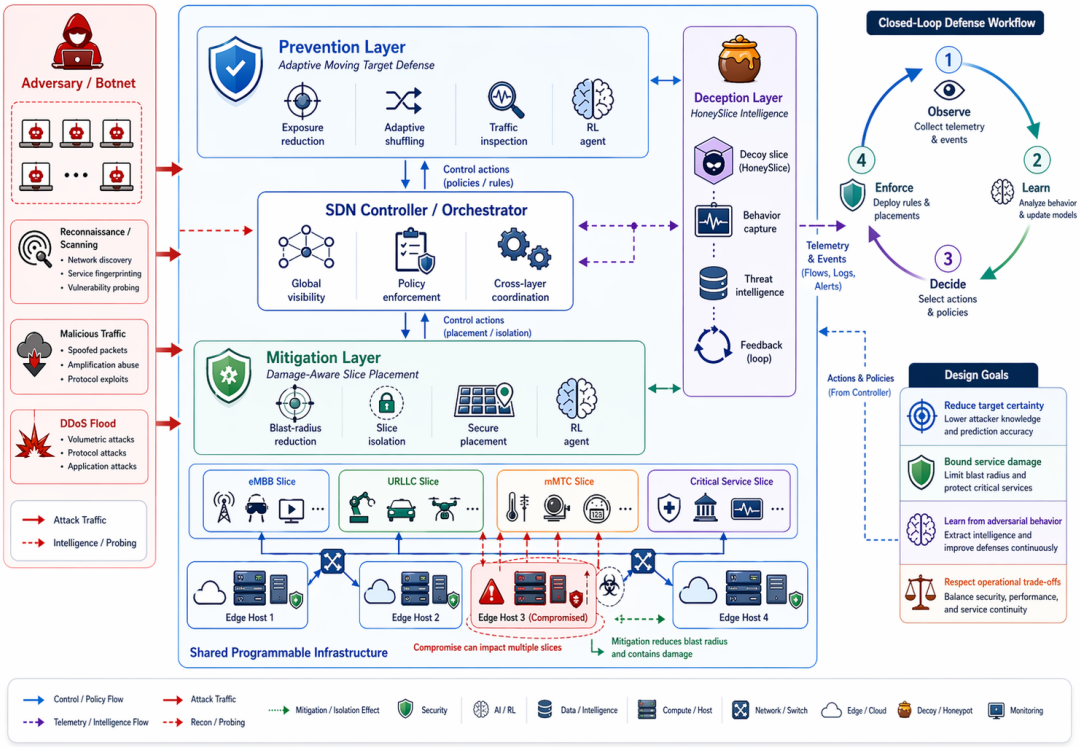


FIGURE 6. Operational view of the proposed coordinated defense framework.

cooldown. The five actions are no-op, MTD mutation, damage-aware replacement, both operations, and HoneySlice rotation. Q-learning uses learning rate 0.12, discount factor 0.94, and an exploration probability decaying from 0.35 to 0.03 over 900 independent training episodes of 100 slots. The final policy is evaluated without exploration over 100 independent seeds, each containing 180 slots.

Baselines and metrics

Five policies are compared: static capacity-first placement; periodic MTD; isolated MTD plus damage-aware placement without deception; a coordinated rule-based controller; and the coordinated RL controller. Metrics are mean criticality-weighted availability, weighted outage damage, high-criticality slice violations per slot, normalized reconfiguration cost, HoneySlice diversion ratio, and the fraction of slots in which a production node is successfully overloaded.

Interpretation

Table 1 shows that periodic MTD alone provides only a small improvement because mutation does not reduce criticality concentration and does not divert attack traffic. Damage-aware placement reduces outage damage and high-criticality violations but does not materially change the number of successful production-node attack slots. The coordinated policies improve both dimensions because HoneySlice evidence and diversion affect target risk while placement limits the damage of non-diverted attacks.

Relative to the static baseline, coordinated RL reduces weighted damage by 50.4%, high-criticality slice violations by 55.8%, and successful attack slots by 44.5%; availability increases by 0.98 percentage points. Compared with the coordinated rule-based controller, RL reduces weighted damage by 10.1% and normalized reconfiguration cost by 38.6%. The result supports the feasibility of joint decision making but does not establish general superiority in real networks. External validity still requires packet-level emulation, controller-in-the-loop experiments, unseen attack families, and operator-specific SLA models.

Security of the RL Decision Layer and Cascading-Failure Containment

The RL component creates an additional attack surface. Observation poisoning can falsify utilization or threat features; reward manipulation can make harmful actions appear beneficial; policy extraction can reveal the controller's response pattern; and adversarial timing can induce repeated migrations. The architecture therefore does not place the learned policy directly in the enforcement path. Telemetry is authenticated and compared across sources, confidence is reduced when sources disagree, actions are constrained by the safety shield, and a cooldown prevents

TABLE 1. Controlled evaluation over 100 independent seeds (mean ± standard deviation).

Policy	Availability (%)	Weighted damage (%)	Critical SLA violations/slot	Reconfiguration cost/slot	Diversion (%)	Successful attack slots (%)
Static	98.057 ± 0.545	1.943 ± 0.545	0.116 ± 0.050	0.000 ± 0.000	—	17.189 ± 4.591
Periodic MTD	98.140 ± 0.524	1.860 ± 0.524	0.108 ± 0.048	0.035 ± 0.000	—	17.167 ± 4.613
Isolated MTD + placement	98.292 ± 0.454	1.708 ± 0.454	0.086 ± 0.042	0.060 ± 0.003	—	17.133 ± 4.616
Coordinated rule-based	98.929 ± 0.313	1.071 ± 0.313	0.055 ± 0.028	0.062 ± 0.006	36.998 ± 9.435	10.761 ± 3.101
Coordinated RL	99.037 ± 0.290	0.963 ± 0.290	0.051 ± 0.028	0.038 ± 0.012	45.174 ± 9.207	9.539 ± 2.815

rapid oscillation. The controller also maintains a deterministic fallback policy that prioritizes service continuity and freezes nonessential reconfiguration when confidence falls below a threshold.

Tight coordination can also propagate failure. Misleading HoneySlice evidence could trigger unnecessary MTD, which could raise controller load and then cause placement churn. Four containment boundaries limit this cascade: (1) HoneySlice output is advisory and cannot directly invoke production actions; (2) each function has a rate and cost budget; (3) the safety shield evaluates the combined action against current SLA and capacity constraints; and (4) circuit-breaker logic disables the contributing signal or function when action failure, telemetry disagreement, or controller latency exceeds a threshold. Controller replication and an append-only decision log support recovery and post-incident analysis.

These safeguards do not eliminate adversarial ML risk, but they make the failure mode explicit and keep a compromised observation source or unstable policy from automatically becoming an infrastructure-wide actuation failure.

Industry Relevance and Representative Use Cases

The practical relevance of the proposed architecture becomes clearer when viewed from the perspective of operators, service providers, and industrial users that rely on shared programmable infrastructure. In conventional networks, DDoS defense is often treated mainly as a traffic-filtering or attack-detection problem. In sliced 6G environments, however, the implications are broader because multiple services may share the same physical, virtual, and edge resources. As a result, the disruption of one exposed NFVI node, orchestration component, or VNF can affect several logical services at the same time. Practical deployment therefore requires not only attack detection but also exposure reduction, damage containment, and intelligence-driven adaptation.

Figure 7 provides a visual overview of the application domains and connects each domain to exposure reduction, damage containment, and deception-derived intelligence.

The framework is especially relevant to telecom operators because network slicing is expected to support heterogeneous services with different latency, reliability, and availability requirements. A mobile-broadband slice, an industrial-control slice, and a public-safety slice may coexist on the same infrastructure, but they should not face the same level of operational risk. Exposure reduction changes selected service bindings before an attack becomes effective. Damage containment avoids unsafe concentration of critical slices. HoneySlice contributes early behavioral indicators before production services are directly affected.

Private 5G/6G deployments provide another application domain. Manufacturing plants, ports, hospitals, energy facilities, and smart campuses increasingly depend on wireless connectivity, edge computing, and VNFs. In these settings, availability is tightly linked to safety, operational continuity, and business impact. A short outage in a consumer service may be inconvenient, whereas the same outage in an industrial automation or emergency-response slice may be highly disruptive.

Across these use cases, the main benefit is the shift from isolated security functions toward coordinated resilience management. Exposure reduction is useful before a flood becomes effective, damage containment is essential when it is bypassed, and deception improves the information available to both. Real deployments must nevertheless calibrate the cost and safety constraints to the operator's topology, controller capacity, and service-level agreements.

Deployment Challenges and Research Directions

Control stability remains a primary challenge. Noisy risk estimates can cause unnecessary churn, and an incorrect reward can favor short-term attack reduction at the expense of long-term service continuity. Safe exploration, offline policy evaluation, action masking, and deterministic fallback policies should therefore precede live deployment.

Dataset realism requires a layered approach. Public benign and attack traces can provide reproducible traffic features; a provider digital twin can represent topology, slice criticality, migration, and controller delay; and privacy-preserving aggregates can calibrate rates without exposing subscriber or tenant data. Rare sophisticated attacks should be represented through scenario generation, red-team exercises, and held-out compositional attacks rather than by assuming that production observations alone are sufficient.

Operational overhead must also remain bounded. Monitoring, mutation, migration, deception maintenance, and policy recomputation consume compute and management bandwidth. Controller-in-the-loop validation should report decision latency, southbound message volume, migration interruption, false-action rate, and recovery time. Finally, the RL policy and every telemetry source should be included in the threat model, with poisoning, evasion, policy extraction, and coordinated cross-function failure evaluated explicitly.

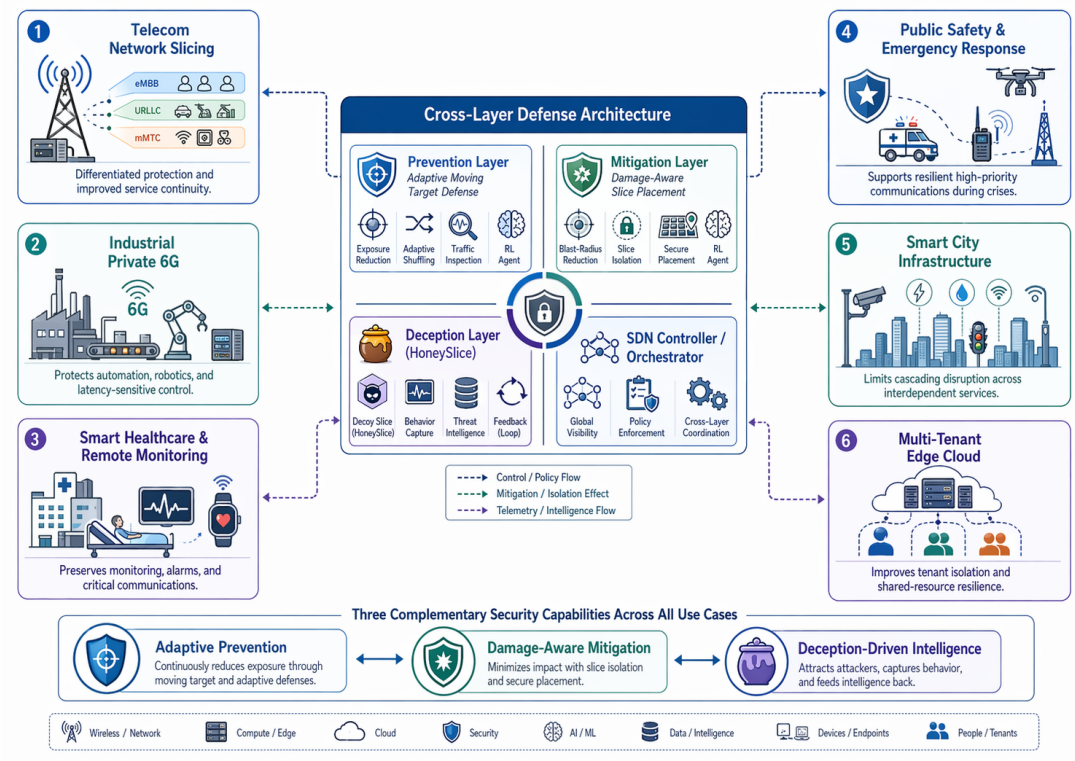


FIGURE 7. Representative industry use cases of the proposed coordinated defense architecture.

TABLE 2. Representative industry use cases and related European projects.

Use case	Main DDoS-related risk	Role of the proposed architecture	Practical value	Related project
Telecom slicing	A flood against a shared core or edge resource disrupts several co-hosted slices.	MTD reduces endpoint certainty, placement limits correlated failure, and HoneySlice improves early risk estimation.	Differentiated continuity protection.	SLICENET
Industrial private 6G	Automation and control slices are sensitive to delay, packet loss, and short outages.	Criticality-aware placement and bounded reconfiguration protect latency-sensitive services.	Reduced operational downtime.	5G CONNI
Smart healthcare	DDoS disrupts monitoring, alarm delivery, or edge analytics.	Stronger isolation for critical slices and decoy-based early warning.	Availability of safety-critical services.	5G-HEART
Public safety	Attackers target communications during crisis periods.	Higher protection priority and rapid constrained reconfiguration.	Crisis communication continuity.	5GASP
Smart city	Interdependent municipal services share edge resources.	Deception attracts reconnaissance and placement reduces correlated exposure.	Lower cascading disruption.	5GCITY
Multi-tenant edge	Tenants share compute, storage, and networking resources.	Risk-aware placement, endpoint mutation, and evidence validation.	Better tenant isolation.	SESAME

Conclusion

DDoS security in SDN-based 6G network slicing cannot be reduced to a single detection or filtering function. Shared NFVI nodes, programmable service bindings, and centralized orchestration create both a larger blast radius and new opportunities for coordinated resilience. This article has specified a closed-loop architecture in which moving target defense reduces reconnaissance reliability, risk-aware placement limits criticality-weighted damage, and an isolated HoneySlice supplies confidence-scored behavioral evidence.

The resulting framework combines an element-specific threat model, a constrained MDP, an enforceable controller cycle, and explicit safety mechanisms. In a reproducible synthetic stress test, coordinated RL reduced weighted outage damage by 50.4%, high-criticality slice violations by 55.8%, and successful attack slots by 44.5% relative to static placement. These results demonstrate internal feasibility and clarify the security–cost–continuity trade-off; they are not a substitute for operational evidence. The next validation stage should combine packet-level attack traces, SDN-controller and NFV-orchestrator prototypes, unseen multi-stage attacks, adversarial telemetry, and measured migration and control-plane overhead. The

central conclusion is therefore bounded but actionable: coordination can improve DDoS resilience only when intelligence is confidence-checked, learned actions are safety-constrained, and each defense function has an explicit containment boundary.

Acknowledgment

The research work is supported in part by the Federal Ministry of Research, Technology, and Space (BMFTR), Germany, through the Project 6GEM+ under Grant 16KIS2411; and in part by the European Union's Horizon Europe research and innovation programme under the 6G-Path project (Grant No. 101139172).

REFERENCES

1. J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 80–87, May 2017.
2. X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5G: Survey and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, May 2017.
3. I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2429–2453, 2018.
4. D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.
5. R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 236–262, 2016.
6. S. T. Zargar, J. Joshi, and D. Tipper, "A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013.
7. Q. Yan, F. R. Yu, Q. Gong, and J. Li, "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 602–622, 2016.
8. A. Javadpour, F. Ja'fari, T. Taleb, and C. Benza"id, "Reinforcement learning-based slice isolation against DDoS attacks in beyond 5G networks," *IEEE Transactions on Network and Service Management*, vol. 20, no. 3, pp. 3930–3946, 2023.
9. C. De Alwis, P. Porambage, K. Dev, T. R. Gadekallu, and M. Liyanage, "A survey on network slicing security: Attacks, challenges, solutions, and research directions," *IEEE Communications Surveys & Tutorials*, vol. 26, no. 1, pp. 534–570, 2024.
10. W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Network*, vol. 34, no. 3, pp. 134–142, May/Jun. 2020.
11. A. Javadpour, F. Ja'fari, T. Taleb, F. Turkmen, and C. Benza"id, "Beyond reinforcement learning for network security: A comprehensive survey and tutorial," *Journal of Information Security and Applications*, vol. 96, p. 104294, 2026.
12. Q. Liu, T. Han, and N. Ansari, "Learning-assisted secure end-to-end network slicing for cyber-physical systems," *IEEE Network*, vol. 34, no. 3, pp. 37–43, May/Jun. 2020.
13. A. Javadpour, F. Ja'fari, T. Taleb, M. Shojafar, and B. Yang, "SCEMA: An SDN-oriented cost-effective edge-based MTD approach," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 667–682, 2022.
14. S. Kim, S. Yoon, J.-H. Cho, D. S. Kim, T. J. Moore, F. Free-Nelson, and H. Lim, "DIVERGENCE: Deep reinforcement learning-based adaptive traffic inspection and moving target defense countermeasure framework," *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 4834–4846, Dec. 2022.
15. A. Javadpour, F. Ja'fari, T. Taleb, C. Benza"id, P. R. Tomas, L. Rosa, J. Proença, and L. Cordeiro, "A reinforcement learning approach to virtual network embedding problems in 5G networks," *IEEE Transactions on Network Science and Engineering*, 2026.
16. A. Javadpour, F. Ja'fari, T. Taleb, and C. Benza"id, "Moving target defense for DDoS mitigation with shuffling of critical edge(s) connections," *Journal of Information Security and Applications*, vol. 97, p. 104347, 2026.
17. A. Javadpour, F. Ja'fari, C. Benza"id, and T. Taleb, "An optimized reinforcement learning-based MTD mutation strategy for securing edge IoT against DDoS attack," *Journal of Information Security and Applications*, vol. 93, p. 104138, 2025.
18. A. Javadpour, F. Ja'fari, T. Taleb, M. Shojafar, and C. Benza"id, "A comprehensive survey on cyber deception techniques to improve honeypot performance," *Computers & Security*, vol. 140, p. 103792, 2024.