Quantifying Resilience of Cyber-Physical Systems to Zero-Day Threats: A Security Twin-Based What-If Analysis Framework

Fabrizio Baiardi

Dipartimento di Informatica, Università di Pisa, Italy. E-mail: fabrizio.baiardi@unipi.it

Vincenzo Sammartino

Dipartimento di Informatica, Università di Pisa, Italy. E-mail: vincenzo.sammartino@phd.unipi.it

Reliability and risk assessment methodologies for cyber-physical systems that heavily rely on historical failure data and public vulnerability databases are increasingly ineffective against ”zero-day” threats—unknown vulnerabilities for which no data or signatures exist. This paper proposes to quantify the system resilience against these vulnerabilities through a *what-if* analysis based on a stochastic simulation framework using a security twin. A security twin enriches a digital twin with information to discover the possible actions for a threat actor in an intrusion, effectively generating the search space for these intrusions. To assess the system resilience, our *what-if* analysis systematically injects hypothetical zero-day vulnerabilities into the system modules. Only the security twin is affected by the injection, while the cyber-physical system is unaffected. Our framework runs extensive Monte Carlo simulations using the security twin to discover possible intrusions. Then, we measure the resulting degradation of resilience using metrics such as Mean Time to Compromise and Intrusion Success Percentage. Our results show that the topological centrality of the target of the injection is a more significant predictor of systemic failure than the intrinsic vulnerability severity. If the zero-day compromises internal ”pivot points” of the system, the time-to-compromise reduces by up to 88% compared to perimeter breaches. These findings provide a quantitative basis for optimizing network segmentation and enhance resilience.

Keywords: Cyber-Physical Systems, Security Twin, Zero-Day Vulnerability, Resilience Engineering, Probabilistic Risk Assessment, Attack Graphs, Monte Carlo Simulation.

1. Introduction

The convergence of information technology(IT) and operational technology (OT) into Cyber-Physical Systems (CPS) has dissolved the traditional ”air gap,” exposing industrial environments to sophisticated digital threats. If, on one hand, the adoption of CPSs improves efficiency, on the other it creates a complex attack surface where intrusions of threat actors can lead to catastrophic physical failures.

Estimating the resilience of CPSs is challenging for risk assessment and management strategies that rely on historical data on failures, intrusions and enumerated vulnerability databases, such as the Common Vulnerabilities and Exposures (CVE) database The MITRE Corporation (2024). The conventional risk equation, $R = P \times C$ (Risk equals Probability times Consequence), assumes P can be estimated from statistical distributions of attacks or known intrusion patterns. However, this approach falters when addressing ”zero-day” threats—vulnerabilities that are unknown to both the vendor and the defender, but potentially known and weaponised by threat actors Roncone and et al. (2024).

The ”Zero-Day Paradox” for an unknown vulnerability arises because, while it can be exploited, historical data cannot model its probability of occurrence, and standard vulnerability scanners cannot detect it. As a result, traditional risk models can optimize defenses against known threats but leave the system architecturally fragile against novel vectors. In the context of high-consequence CPS, ignoring these low-probability, high-impact events is unacceptable. The question for resilience engineering thus shifts from *”What is the probability of an intrusion?”* to *”How does the system behave if an unknown vector compromises a critical component?”* Wang et al. (2014).

Proc. of the 36th European Safety and Reliability Conference
Edited by José C. Matos, Paulo B. Lourenco, Michael Beer, Daniel Oliveira and Edoardo Patelli
Copyright © 2026 by ESREL2026 Organizers. *Published by* Research Publishing
ISBN: 981-973-0000-00-0 :: doi: 10.3850/981-973-0000-00-0_output

To address this, we propose a framework that merges a *what-if* analysis and dynamic stress testing using a **security twin** (ST). Unlike generic digital twins focused on operational performance, a ST is an enriched inventory that integrates asset inventory, network topology, physical dependencies, configurations, ACLs, and vulnerability states to support rigorous *what-if* analysis Baiardi et al. (2024, 2025). A ST makes it possible to simulate intrusions where an intrusion is a sequence of actions by a threat actor that tries to reach a goal, i.e. to control some asset. The information in the ST defines the intrusion *state space*, that is the actions the actor can execute in an intrusion. Each action defines a state transition in this space. *What-if* analyses update information in the ST to model alternative configurations of the CPS. Then, the analysis runs simulations of a threat actor to discover if the changes enable new intrusions. The usage of a ST avoids any disturbance to the CPS. Hence, our approach does not attempt to predict *when* a zero-day occurs. Instead, it systematically injects hypothetical zero-day vulnerabilities into the ST to assess the resulting impact and measure systemic resilience of the CPS.

This framework defines a proactive tool to quantify the "blast radius" of zero-day vulnerabilities and validate architectural resilience strategies, such as Zero Trust segmentation, before real-world incidents occur Baiardi and Sammartino (2025).

The contribution of this work is threefold. First, we formalize the ST as a structural model defining valid state transitions. Second, we propose a **Zero-Day Injection Methodology** that treats unknown vulnerabilities as concrete enabling conditions for worst-case scenario evaluation. Finally, we provide empirical evidence that the **topological centrality** of an asset is a better risk predictor than intrinsic vulnerability severity. This defines a new metric for prioritizing hardening efforts.

2. Related Work

The problem of securing a CPS has been addressed through various methodological lenses, ranging from graph theory to reliability engineering. This section reviews the evolution of these approaches, highlighting the transition from static risk assessment to dynamic resilience modeling, and frames the concept of ST within the current literature.

2.1. *Probabilistic Risk Assessment and Attack Graphs*

The formalization of intrusions in complex networks has traditionally relied on Attack Graphs (AGs). Seminal work by Sheyner et al. (2002) established the foundation for automated AG generation, allowing analysts to map all possible intrusions to reach a target. These models have been enhanced with Bayesian networks to compute the probability of node compromise based on the exploitability of vulnerabilities.

However, traditional AGs suffer from two significant limitations when applied to CPS. First, they are predominantly static; they rely on fixed probability scores derived from the Common Vulnerability Scoring System (CVSS), which often fails to reflect the temporal dynamics of an attack or the specific configuration of the target environment Baiardi et al. (2026c). Second, they typically assume a "closed world" scenario where all vulnerabilities are known *a priori*. This makes them ill-suited for modeling zero-day threats, where the probability of exploitation also depends on the capability and information of a threat actor.

2.2. *CPS Resilience Engineering*

As the focus of safety engineering shifts from preventing failure (reliability) to sustaining operations under stress (resilience), new frameworks have emerged. Zio (2018) and Fang and Zio (2019); Baiardi et al. (2025) have defined resilience metrics for critical infrastructures, emphasizing component interdependencies and how perturbations in the cyber domain can cascade into disruptions.

Despite these advances, there remains a gap in the integration of intelligent adversary models into resilience frameworks. Most models treat cyber-attacks as stochastic failure events or focus on the physical consequences and neglect the intelligent behaviour of the threat agents that determine the success of intrusions. Instead, we model an adver-

sary not as a random noise generator, but as an agent that applies a strategy to select a sequence of actions to navigate the target CPS, and build an intrusion that impacts the CPS resilience.

2.3. *From Digital to Security Twins*

The digital twin (DT) paradigm has revolutionized predictive maintenance and operational optimization. The safety domain, Maio et al. (2013), has shown the efficacy of DTs combined with statistical methods for fault detection in nuclear components. However, the application of DTs to cybersecurity is a nascent field Baiardi and Sammartino (2026a); Sammartino et al. (2025). In this field, a ST generalizes the notion of DT by enriching the twin with information on vulnerabilities and attacks to discover the intrusions of a threat actor in a target system. Current literature on STs mainly focuses on intrusion detection systems, using the twin to mirror network traffic and detect anomalies in real-time. There is a paucity of research using STs for *predictive what-if* analysis. Existing simulation-based approaches often lack the granular fidelity to model specific vulnerability injections (such as zero-days) or fail to account for the hybrid IT/OT topology of industrial environments Sammartino (2025).

Our research bridges this gap by proposing a ST that acts as a generative model for valid sequences of actions in intrusions. By injecting deterministic zero-day vulnerabilities into the twin, we generalize the probabilistic estimation of known risks to the stress test of architectural resilience when facing unknown Baiardi et al. (2026a) vulnerabilities.

3. Methodology

To quantify resilience against unknown vulnerabilities, we propose a computational framework based on discrete-step adversary simulation. The methodology distinguishes the static infrastructure model (the ST) from the dynamic execution model (the Intrusion Simulation).

3.1. *Formalizing the Security Twin*

We define the state of an intrusion as the pair of information and access rights a threat actor has acquired by its previous actions in an intrusion. Each action of the actor results in a state transition. In this perspective, we see the ST as a directed, attribute-rich graph $\mathcal{G} = (V, E, \Lambda)$ that essentially defines which actions an actor can execute in a state where it owns some information and some access rights. In this way, we generate the state space in which an intrusion occurs as a function of the possible actions of an actor and the resulting state transitions. More formally,

- $V = \{v_1, v_2, \ldots, v_n\}$ is the set of vertices representing cyber-physical assets (e.g., Engineering Workstations, PLCs, Historians). Each vertex v_i is paired with a state vector $\mathbf{s}_i = \langle \text{Zone}, \text{Crit}, \text{Vuln} \rangle$, where Vuln is the set of asset vulnerabilities (known or zero-day).
- $E \subseteq V \times V$ is the set of directed edges representing logical reachability. An edge e_{ij} exists if and only if the network configuration (firewall rules, routing tables) permits traffic flow between the assets.
- $\Lambda : E \to [0, 1]$ is a weighting function representing the *exploitability probability* inherent to the infrastructure's state.

3.2. *Modeling the Zero-Day Perturbation*

Our *what-if* framework models a zero-day vulnerability as a topological perturbation function $\Phi(\mathcal{G}, v_{target})$ that maps the graph $\mathcal{G}$ into $\mathcal{G}'$. Unlike known vulnerabilities, where exploit success is probabilistic ($P < 1$), we model a zero-day exploit under the "Assume Breach" paradigm. If an attacker possesses a zero-day for asset v_{target}, the corresponding state transition the ST describes is deterministic, as it is always successful provided that the attacker reaches a state where it can exploit the zero-day.

Mathematically, we inject a vulnerability z_0 into the set $\text{Vuln}(v_{target})$ with the following properties:

$$P(\text{exploit}|z_0) \approx 1.0, \quad \text{Severity}(z_0) = 10.0 \quad (1)$$

This injection creates a "gravity well" in the ST: any potential intrusion where the actor controls some neighbors of v_{target} now has a high-probability next valid transition that controls v_{target} itself.

3.3. *Stochastic Adversary Simulation*

As previously mentioned, we assume the actor has a strategy that selects which action it executes in a state. This describes a rational actor who prioritizes high-value targets and reliable exploits and distinguishes the set of possible transitions the ST defines from the actual ones the actor selects. Hence, the ST defines which moves, i.e. actions, can happen while the strategy decides which should happen. We model a strategy via a utility function that selects the next action and its target v_{next} from the valid options the ST provides in the current state:

$$v_{next} = \arg \max_{v \in \mathcal{N}(C_t)} (P(C_t \to v) \times \text{Value}(v)) \tag{2}$$

3.3.1. *Pathfinding Strategy*

We have specified and built a simulation engine that emulates a threat agent $\mathcal{A}$ traversing the intrusion state space Baiardi et al. (2026b). The simulation works in discrete time steps t. At any given step, the engine considers the set of compromised nodes $C_t \subseteq V$ and it determines the set of possible state transitions and $\mathcal{N}(C_t)$, the reachable neighbors nodes $\mathcal{A}$ can move to. Then, the engine applies the strategy of $\mathcal{A}$ to select the action to execute and the resulting transition Baiardi and Sammartino (2026b).

3.3.2. *Time-Domain Stochasticity*

A critical component of our methodology is time modeling because usually there are constraints on the time of an intrusion. We model the time an action requires, T_{comp}, as a random variable drawn from a Log-Normal distribution:

$$T_{comp}(v, e) \sim \text{LogNorm}(\mu, \sigma^2) \tag{3}$$

It is important to emphasize that the distribution parameters are not arbitrary but are functionally dependent on the edge weight $\Lambda(e)$ in the ST. We model the mean time μ as inversely correlated to the probability of success:

$$\mu \propto \frac{1}{\Lambda(e)} \tag{4}$$

This dependency captures that vulnerabilities with lower exploitability scores require longer execution times for trial-and-error. In contrast, high-probability exploits—and injected zero-day vulnerabilities where $\Lambda(e) \approx 1.0$—result in minimal μ, simulating a near-instantaneous compromise. This ensures that the MTTC metric reflects not just the number of times an action is repeated before it is successful, but also the overall complexity of the action.

3.4. *Resilience Metrics*

To compute the statistics to assess resiliency, we apply a Monte Carlo method that executes N intrusion simulations ($N = 10^4$) for each *what-if* analysis. In this way, we derive two primary metrics:

- **Intrusion Success Percentage (IS%):** The percentage of simulation runs where the agent successfully reaches the goal, a target in V_{OT} (Operational Technology zone).
- **Mean Time to Compromise (MTTC):** The expected value of the time to breach the target in successful runs:

$$MTTC = \mathbb{E}[T_{total}|\text{Success}] = \frac{1}{|S|} \sum_{k \in S} \tau_k \tag{5}$$

  where S is the set of successful intrusions and τ_k is the time of the $k-th$ intrusion.

These metrics quantify the "grace period" available to the defender before an impact occurs.

4. Experimental Setup

To validate our framework, we built the ST of a medium-sized CPS. The experimental design aims to investigate the "vulnerability location", the position of the asset where the vulnerability is injected, to measure its specific contribution to systemic risk.

4.1. *The Security Twin Architecture*

We assume that the target CSP follows the Purdue Enterprise Reference Architecture, the industry standard for segmenting ICS/OT networks. The CPS topology includes $|V| = 54$ nodes and $|E| = 138$ edges, structured into four hierarchically segmented zones:

- **Level 4 (Enterprise Zone):** The business-critical assets, including the Domain Controller (DC), Enterprise Resource Planning (ERP) systems, and 20 generic User Workstations. It has broad internet connectivity but restricted access to the OT layers.
- **Level 3.5 (DMZ):** A demilitarized zone acting as a buffer. It contains internet-facing services: a Web Server, a Mail Gateway, and a VPN Concentrator.
- **Level 3 (Operations Zone):** The critical management layer that hosts the SCADA Server, the Data Historian, and Engineering Workstations (EWS). Traffic to Level 4 is strictly filtered (e.g., only the Historian can push data to the ERP).
- **Level 1 (Control Zone):** The physical control layer with Programmable Logic Controllers (PLCs) and Remote Terminal Units (RTUs) that are the V_{target} for the adversary.

The ST is populated with a realistic vulnerability landscape derived from the National Vulnerability Database (NVD), simulating a typical "patch lag" scenario where non-critical legacy vulnerabilities exist in the OT environment due to maintenance constraints.

4.2. *Statistical Robustness*

To evaluate how injected vulnerabilities affect resilience, we employ a dynamic statistical convergence protocol to ensure the robustness of the Monte Carlo rescontinues running simulations until the number of distinct intrusions the simulations discover stabilizes. This ensures that the derived metrics (IS% and MTTC) are statistically significant and that the state space of potential intrusions has been adequately explored. To this purpose, the engine computes the running mean μ and standard deviation σ of the metrics at regular intervals. In our experiments, the simulation ends when the width of the 95% confidence interval for the mean falls within a 5% relative error margin, but these values can be tuned to the target CPS. The stopping condition is formally defined as:

$$\frac{1.96 \cdot \sigma}{\sqrt{N}} \leq 0.05 \cdot \mu \tag{6}$$

where N is the current number of simulations.

4.3. *The What-If Scenarios*

We defined four distinct scenarios to stress-test the CPS. The **Baseline** represents the control group, while the subsequent three scenarios inject a single zero-day vulnerability (CVSS 10.0) at different topological depths.

4.3.1. *Baseline Scenario (Current Posture)*

No zero-day vulnerabilities are injected. The adversary must rely solely on existing, known misconfiguration or unpatched legacy vulnerabilities to reach its target. This establishes the reference MTTC.

4.3.2. *Scenario A: Perimeter Breach*

The zero-day is injected into the **Public Web Server** in the DMZ to simulate an intrusion where an attacker exploits a novel vulnerability in the web application stack to gain an initial foothold.

4.3.3. *Scenario B: Internal Pivot*

The zero-day is injected into the **Central File Server**, CFS, in the Enterprise Zone. This asset is chosen for its high centrality degree as it is accessible by most workstations and has administrative paths to backup systems. This scenario simulates an intrusion enabled by a supply chain attack or by a compromised insider credential that grants immediate control over a core infrastructure server.

4.3.4. *Scenario C: Endpoint Compromise*

The zero-day is injected into a randomly selected **User Workstation** at Level 4. This simulates a client-side attack, such as a zero-click browser exploit delivered via a phishing campaign. While the vulnerability is critical, the asset itself has lower centrality than a server.

5. Results and Analysis

The output of the Monte Carlo method provides a quantitative assessment of resilience under the influence of unknown threats. By aggregating the Monte Carlo simulations according to the statistical criterion in Section 4.2, we deduced the IS% and the MTTC. The results are summarized in Table 1.

Table 1. Resilience Metrics Across Scenarios.

Scenario	IS%	MTTC (h)	Δ MTTC
Baseline	5.2%	72.5	-
Scenario A	45.8%	15.2	-79.0%
Scenario B	**62.3%**	**8.9**	**-87.7%**
Scenario C	38.5%	25.4	-64.9%

5.1. *Baseline Resilience*

The simulations confirm the system robustness in the Baseline scenario. The low IS% (5.2%) and high MTTC (72.5 hours) show that the existing network segmentation and patch management are effective against adversaries relying on known CVEs. The actors are forced to chain multiple low-probability exploits, significantly increasing the likelihood of detection or failure before reaching the Control Zone (Level 1).

5.2. *The Impact of Zero-Day Location*

The introduction of a zero-day vulnerability fundamentally shifts the resilience equilibrium, but the magnitude of this shift is highly sensitive to the location where the injection occurs.

5.2.1. *The Criticality of Internal Pivots*

Scenario B (Internal Pivot) represents the absolute worst-case. Although the severity of the injected vulnerability is the same as in Scenarios A and C, the IS% surges to 62.3% and the MTTC collapses to just 8.9 hours. This is due to the *betweenness centrality* of the CFS that acts as a bridge between the Enterprise Zone and the management interfaces of the OT zone. The compromise via a zero-day eliminates the ”security depth” the DMZ provides and grants the adversary a high-speed lateral path that bypasses perimeter monitoring.

5.2.2. *Perimeter vs. Endpoint Resilience*

In spite of a large degradation, (IS% 45.8%), Scenario A (Perimeter) is more resilient than Scenario B. This suggests that the DMZ architecture still imposes a ”topological tax” on the attacker, who still has to avoid internal firewalls to reach the OT core even when the main defense (the Web Server) is bypassed by a zero-day.

Scenario C (Endpoint) yields the highest MTTC among the zero-day scenarios (25.4h). While the initial compromise is instantaneous, the workstation’s low *Out-Degree* and restricted access rights force additional, time-consuming discovery and privilege escalation steps by the attacker. This confirms that workstation-level zero-days, while dangerous, offer a larger window for incident response than server-side exploits.

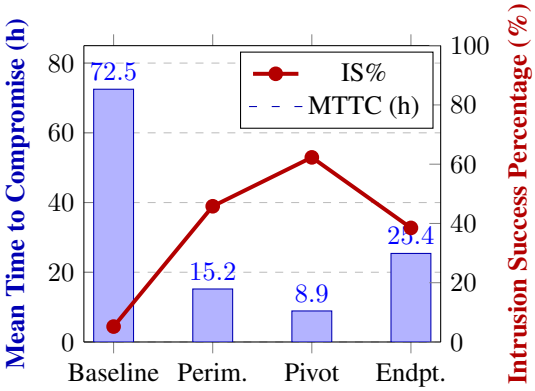

Fig. 1. Dual-axis comparison of Resilience Metrics. The blue bars represent the MTTC (left axis), while the red line tracks the IS% (right axis). Note the inverse correlation in Scenario B.

Fig. 1 highlights the inverse correlation between the time available for defense and the percentage of adversary success. The **Baseline** scenario exhibits a ”healthy” resilience profile: a tall bar (high MTTC) coupled with a low data point (low IS%).

5.3. *Distribution of Attack Paths*

A detailed analysis of simulations reveals that T_{total} has the largest variance in Scenario C. This shows that as far as endpoint compromise, resilience is ”path-dependent”: if the attacker strategy luckily selects the workstation of a privileged user (e.g., a SCADA admin), the MTTC drops sharply. In Scenario B, the variance is minimal, indicating a deterministic-like collapse of resilience regardless of the simulation.

5.4. *Sensitivity to Network Segmentation*

An interesting result is the non-linear relationship between segmentation and MTTC. By re-running a subset of simulations with weakened firewall rules, we observed that the MTTC in Scenario B dropped by an additional 40%, while Scenario A remained relatively stable. This suggests that internal segmentation is the primary resilience factor against unknown vectors.

6. Discussion

The simulation results offer several insights into the resilience of CPSs. Beyond raw metrics, these results suggest a topological approach to defense.

6.1. *Re-evaluating the Risk Equation*

Our experiments confirm the fragility of the equation $Risk = Likelihood \times Impact$ when applied to zero-day threats because the *Likelihood* term is dominated by epistemic uncertainty and cannot be reliably estimated. Our findings suggest that resilience engineering should pivot from probabilistic estimation of threats to the deterministic analysis of *topological exposure*. As evidenced by the differences between Scenario A and Scenario B, systemic risk is not a function of the vulnerability severity because it was constant at CVSS 10.0. The important parameter is the vulnerability location and mainly the *betweenness centrality* of the compromised asset. Hence "Criticality" in a CPS should not merely depend on the physical function of an asset (e.g., controlling a valve), but on its structural capacity to bridge segmented network zones.

6.2. *The Strategic Value of "Pivot Points"*

The identification of the CFS as a catastrophic failure point (Scenario B) highlights the concept of "Pivot Points"—assets that, while not physically critical themselves, serve as force multipliers for an adversary. In many CPSs, these assets are often overlooked because they reside in the IT or Enterprise layers (Level 4). However, our *what-if* analysis reveals that they may act as high-speed conduits into the OT domain. This provides further support for the adoption of Zero Trust principles. Specifically, it argues for the micro-segmentation of internal "trusted" traffic. If the CFS in Scenario B had been subjected to east-west traffic inspection, the MTTC would have likely converged towards the higher values in Scenario C.

6.3. *Operationalizing MTTC for Defense*

From an operational perspective, the MTTC is a crucial design metric. In resilience engineering, total prevention of failure is often impossible; the goal is to ensure that the system's "Time-to-Failure" exceeds the "Time-to-Recover" (or in this case, "Time-to-Detect"). The baseline MTTC of 72.5 hours suggests that, for known threats, standard security response times are adequate. However, the collapse to 8.9 hours in Scenario B creates a "race condition" that most human-driven response teams cannot win. This supports the argument for automated response mechanisms in CPSs. The system should autonomously isolate a high-centrality node as soon as it exhibits anomalous behavior, as the window for human decision-making is critically compressed.

6.4. *Limitations and Model Constraints*

While the ST supports a high-fidelity simulation, some simplifications were adopted. First, we assume a "rational agent" optimizing for speed and probability. This does not account for psychological factors or deceptive "low-and-slow" strategies typical of state-sponsored actors, which might intentionally increase the MTTC to slow down detection. Second, our framework assumes static defensive actions as it neglects the feasibility of patching or isolating nodes during an intrusion. Future work will incorporate multi-agent Reinforcement Learning to simulate a dynamic interaction between attacker and defender.

7. Conclusion

The convergence of IT and OT has rendered CPSs susceptible to zero-day exploits that most reliability models cannot quantify. This paper proposes a dynamic assessment framework centred on a *what-if* analysis that fully exploits a ST by us-

ing adversary simulations as a stress test method against the unknown.

Our research yields three main conclusions. First, the **topological location** of a vulnerability is a far more accurate predictor of systemic risk than the intrinsic severity of the vulnerability itself. As demonstrated, a zero-day in a high-centrality internal server (Scenario B) resulted in an 88% reduction in MTTC Second, the concept of **"Pivot Points"** supports a prioritized defense because assets bridging the IT/OT divide act as force multipliers for attackers. By identifying and hardening these nodes, we achieve a higher Return on Investment (ROI) for resilience than by adopting indiscriminate patching. Third, the use of **MTTC** as a design metric provides a tangible benchmark for resilience. It transforms abstract risk probabilities into a time-domain value that can be directly compared against incident response capabilities.

Future research will integrate Multi-Agent Reinforcement Learning to model dynamic adversarial interactions, allowing us to simulate not just the intrusion, but the automated adaptive response of the defender in real-time.

References

Baiardi, F., S. Ruggieri, and V. Sammartino (2024). Anticipating Disasters through a Security Twin. In *SPRINGER OPTIMIZATION AND ITS APPLICATIONS - ARES*.

Baiardi, F., S. Ruggieri, and V. Sammartino (2025). Ai-enabled cybersecurity using synthetic data. In *2025 IEEE PerCom*, pp. 140.

Baiardi, F. and V. Sammartino (2025). A quantitative framework for the validation of twin-based cyber defense. In *I3M*.

Baiardi, F. and V. Sammartino (2026a). From digital twins to ai agents: A synthetic data paradigm for next-generation cybersecurity. In *Artificial Intelligence in Cybersecurity: Unlocking the Power of Large Language Models*. CRC Press.

Baiardi, F. and V. Sammartino (2026b). Simulation-powered cybersecurity: Real-time risk assessment via non-intrusive security twin. *The Journal of Supercomputing*.

Baiardi, F., V. Sammartino, and S. Ruggieri (2025). Notline: A non-intrusive automated platform to build a digital twin. In *29th DS-RT*.

Baiardi, F., V. Sammartino, and S. Ruggieri (2026a). Enriching security twins with behavioral performance modeling and vm detection. In *ISRERM 2026*.

Baiardi, F., V. Sammartino, and S. Ruggieri (2026b). Evaluating adversary strategies through a security twin. In *IEEE PerCom*.

Baiardi, F., V. Sammartino, and S. Ruggieri (2026c). Quantifying the impact of cvss score ordering and attack paths. In *GOODTECHS 2026*.

Fang, Y. and E. Zio (2019). An adaptive robust framework for the optimization of the resilience of critical infrastructure networks against malicious attacks. *EJOR 276*(3), 1119–1133.

Maio, F. D., P. Baraldi, and E. Zio (2013). Fault detection in nuclear power plants components by a combination of statistical methods. *IEEE Transactions on Reliability 62*(4), 833–845.

Roncone, G. and et al. (2024). Apt44: Unearthing sandworm. Technical report, Mandiant.

Sammartino, V. (2025). A framework for proactive cyber-resilience: Non-intrusive modeling for autonomous defense. In *DS-RT 2025*.

Sammartino, V., F. Baiardi, and S. Ruggieri (2025). A Security Twin to Defeat Intrusions in Cyber Physical Systems. In *ESREL SRA-E 2025*.

Sheyner, O., J. Haines, S. Jha, R. Lippmann, and J. M. Wing (2002). Automated generation and analysis of attack graphs. In *IEEE Symposium on Security and Privacy*.

The MITRE Corporation (2024). Cve.

Wang, L., S. Jajodia, A. Singhal, P. Cheng, and S. Noel (2014). k-zero day safety: A network security metric for measuring the risk of unknown vulnerabilities. *IEEE Trans. on Dependable and Secure Computing 11*(1), 30–44.

Zio, E. (2018). The future of risk assessment. *Reliability Engineering & System Safety 177*, 176–190.