12th INTERNATIONAL SCIENTIFIC CONFERENCE ON DEFENSIVE TECHNOLOGIES
OTEH 2026
Serbia, Tara, 21 - 24 October 2026

QUANTITATIVE ASSESSMENT OF ZERO TRUST ARCHITECTURE EFFECTIVENESS AGAINST ADVANCED CYBER ATTACKS IN A SMALL-SCALE EMULATED GOVERNMENT AND MILITARY INFORMATION SYSTEMS

BOBAN STOJANOVIĆ
Serbian Armed Forces, Belgrade, boban.stojanovic@vs.rs

MLADEN VEINOVIĆ
Singidunum University, Belgrade, mveinovic@singidunum.ac.rs

MARKO PREMOVIĆ
Serbian Armed Forces, Belgrade, marko.premovic@vs.rs

IGOR PETROVIĆ
Ministry of Defense, Belgrade, igor.petrovic@mod.gov.rs

Abstract: *In government and military networks, a successful intrusion does not necessarily depend on breaking through the network perimeter. APT actors may obtain valid credentials and use them to gain additional privileges and move between systems. We compared two security configurations in a small testbed containing 28 hosts. We used the same six MITRE ATT&CK techniques in both environments. For each run, we measured detection and response time, attacker dwell time, lateral movement, compromised hosts, and data exfiltration. We derived Attack Containment Efficiency (ACE) from the number of compromised hosts and used it as an additional containment measure. Detection in the traditional environment averaged 38.4 min, whereas in the Zero Trust environment it averaged 6.9 min. Response time fell from 71.8 to 14.2 min. Following the initial compromise, we observed an average of 12.4 successful lateral transitions per run in the traditional environment and 1.7 in the Zero Trust environment. The mean number of compromised hosts was 23 and 3, respectively, while successful data exfiltration decreased from 78% to 11%. Paired analysis showed significant differences between the configurations for all evaluated variables ($p < 0.001$). These results reflect the combined effect of the controls implemented in the Zero Trust-oriented configuration and should not be interpreted as the effect of any single security control.*

Keywords: *Zero Trust, MITRE ATT&CK, Military Systems, Attack Containment, Lateral Movement*

1. INTRODUCTION

Malware is not the only way into a military network. An attacker who obtains valid user credentials may already have the access needed to establish a foothold. From there, the account can be used to seek additional privileges, retain access, or reach other systems [1]. Identity-based attacks have also been observed in public-sector networks [2]. In a military setting, this kind of access can have wider consequences when operational or administrative systems supporting mission activities are affected [3].

Traditional network security largely focuses on protecting the boundary between trusted and untrusted networks [4]. This approach becomes less useful once an attacker has crossed that boundary. An attacker with internal access may attempt to reach other hosts and services from inside the network. NIST SP 800-207 therefore treats network location as insufficient grounds for trust and considers threats that may be present both inside and outside the traditional perimeter [5].

Zero Trust does not assume that a user or device can be trusted simply because it is already inside the network. Access decisions are instead based on identity and other available security information [4], [6]. An access request must therefore be evaluated without relying on network location as proof of trust. This approach is also reflected in NSA guidance, which presents Zero Trust as a means of maintaining authorized users' access to mission data while limiting an adversary's access when part of the network has been compromised [4].

This study was motivated by the lack of quantitative evidence on Zero Trust effectiveness in military networks. We built two matched test environments with the same host and application inventory but different security configurations, ran the same attacks against both, and compared the results.

1.1 Related Work

Kindervag [6] proposed Zero Trust as a different way of thinking about network security. Instead of assuming that systems behind the perimeter are trustworthy, access must be considered separately for each resource. NIST SP 800-207 [5] later described this idea in greater technical detail,

Corresponding author: Boban Stojanović;
E-mail: boban.stojanovic@vs.rs
DOI: 10.5937/OTEH26096S

specifying the components responsible for making and enforcing access decisions. Experimental studies have since examined micro-segmentation, including work based on network emulation and attack graphs.

Controlled attack simulations provide one way to examine how security controls behave during an intrusion. MITRE ATT&CK [7] provides the terminology for describing the adversary techniques used in such tests. Caldera can execute the same ATT&CK techniques in an emulated environment, allowing detection and response to be observed under controlled conditions [7]. Hutchins et al. [8] earlier structured an intrusion using the intrusion kill chain model, dividing an adversary campaign into stages.

Attack graphs have been used to examine how far an attacker can move through a segmented network [9]. Other researchers have tested micro-segmentation by reproducing attacks in controlled environments [10]. Most of this work, however, deals with enterprise networks. Military-oriented test environments are less frequently examined, and containment is rarely expressed through a metric such as ACE.

This study addresses that gap by evaluating Zero Trust Architecture against six MITRE ATT&CK techniques in a small-scale emulated military environment, using paired runs and a controlled experimental design. The selected post-compromise behaviors are also consistent with APT techniques summarized in the literature [11].

2. RESEARCH METHODOLOGY

2.1 Experimental Design

The experimental setup consisted of two matched environments running on a VMware vSphere cluster. The hardware and software inventory was kept the same in both cases: 25 Windows 10 endpoints, three Windows Server 2022 systems providing file, application, and database services, an Active Directory domain, and Splunk for centralized logging. This gave 28 hosts in each environment. The environments differed in the security configuration applied to them.

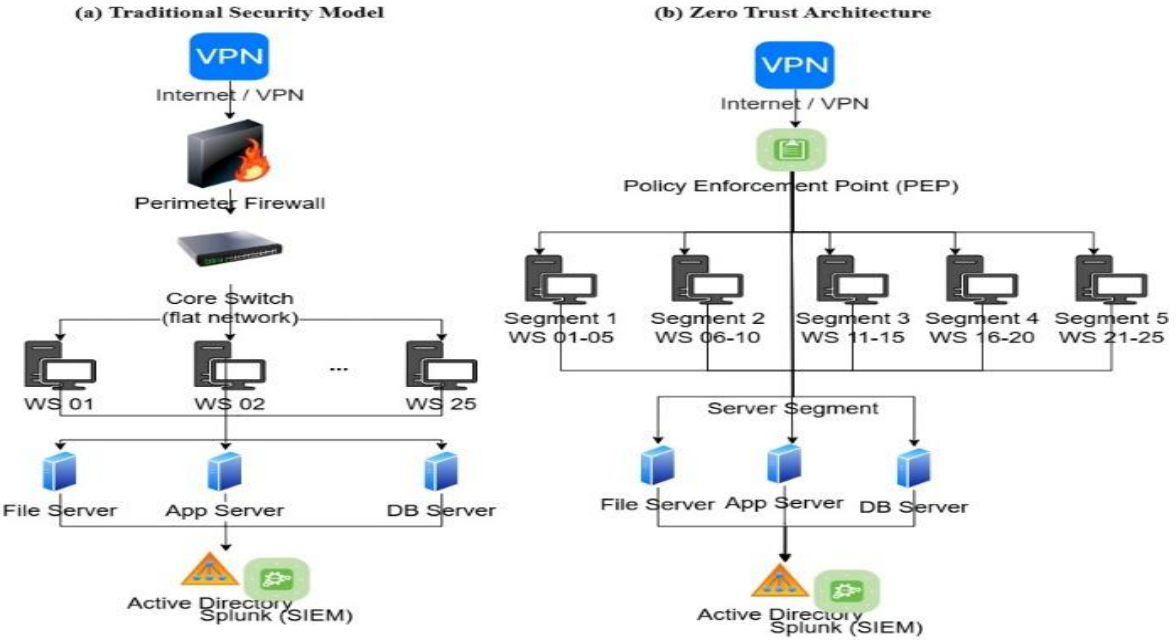


Figure 1. Topology of the two emulated environments: (a) Traditional Security Model with perimeter-only protection and implicit trust between hosts; (b) Zero Trust Architecture with five micro-segments, policy enforcement points (PEP), identity-aware access control, and least-privilege enforcement.

Environment A – Traditional Security Model: Flat internal network topology; perimeter firewall protection; VPN-based remote access; static authentication (password + MFA for external access); domain-based trust model.

Environment B – Zero Trust Architecture: We split the network into five segments and applied ABAC to control access between them. We included device posture in the access decision. We required re-authentication every 60 minutes and monitored user activity and active sessions between checks. Policy and user privileges limited access to resources [12].

2.2 Attack Simulation Scenarios

Six attack cases were executed in accordance with the MITRE ATT&CK Enterprise Framework [7]:

Table 1. MITRE ATT&CK Attack Cases

Attack Phase	Technique	Zero Trust Mitigation
Initial Access	T1078 - Valid Accounts	Periodic re-authentication + behavioral monitoring
Privilege Escalation	T1003 - Credential Dumping	Least-privilege access
Lateral Movement	T1550.002 - Pass-the-Hash	Network segmentation
Execution	T1059.001 - PowerShell	Application control
Lateral Movement	T1210 - Remote Service Exploit	Micro-segmentation
Propagation	T1570 - Lateral Tool Transfer	Traffic monitoring

An automated adversary emulation platform (MITRE Caldera) executed the attacks using predefined ability scripts mapped to the six ATT&CK techniques in Table 1. Each run executed a single technique against a freshly reverted snapshot. We reverted the environment to the baseline snapshot before each run and verified that Active Directory, the Splunk index, and the endpoint state were clean.

Both environments used the same Splunk-based SOC pipeline, with no changes to the correlation rules or alert thresholds. In Environment A, the analyst followed the standard response playbook and carried out the required actions manually. The same analyst handled the alerts in Environment B, but session termination and segment isolation could also be enforced automatically. Since the analyst knew which environment was being tested, the experiment was not blinded.

The first correlated alert recorded in Splunk marked the detection point for a run. We recorded containment after either isolating the compromised host or deactivating the account used in the attack. For LMS, we recorded each successful lateral transition during a run. DESR represented the share of the predefined canary files that successfully reached the external exfiltration listener.

We repeated each technique 20 times, for 120 runs per environment. We matched runs from the two environments by technique and initial access vector. Thus, both members of a pair started from the same access vector. This pairing allowed the two security configurations to be compared under equivalent starting conditions [10].

The security setup was not changed one control at a time. In Environment B, several controls were active simultaneously: network segmentation, ABAC, device-posture checks, application control, behavioral monitoring, periodic re-authentication, and policy enforcement. Environment A did not include these changes and retained the flat-network configuration. Consequently, the experiment compares the two implemented setups. It does not show which individual control drove a particular improvement. Answering that question would require additional tests in which the controls were introduced or removed separately.

2.3 Assessment Criteria

The following metrics were measured:

- **MTTD** – time from the initial compromise event to the first SOC alert.
- **MTTR** – time from the first SOC alert to the containment action.
- **ADT** – time from the initial compromise event to the eradication of the attacker's presence. ADT includes both the undetected and response periods.
- **LMS** – number of successful lateral transitions per run.
- **CH** – total number of systems compromised per run.
- **DESR** – percentage of the predefined canary file set successfully exfiltrated per run.

Note on MTTD and ADT. MTTD and ADT are not the same interval. MTTD ends at the first alert; ADT ends at eradication. In the traditional environment, the attacker remained active for 252 min because detection was late (38.4 min) and containment was slow (71.8 min); in the Zero Trust environment, ADT was 29 min because detection was fast (6.9 min) and containment was fast (14.2 min). The relationship ADT ≈ MTTD + MTTR + residual attacker activity is consistent with the recorded event sequence.

2.4 Attack Containment Efficiency (ACE)

ACE is a normalization of the compromised-host count to the interval [0, 1]:

$$ACE = 1 - \frac{CH}{TH} \quad (1)$$

where CH = number of compromised hosts and TH = total number of hosts (28). ACE values approaching 1 indicate stronger containment. ACE is a transformation of CH and TH already reported in Table 2.

2.5 Statistical Design

For every run in one environment, we selected the corresponding run in the other environment with the same ATT&CK technique and initial access vector. This produced 120 pairs for the analysis. We compared the values within these pairs using a paired t-test. We also checked the differences between paired observations with the Shapiro-Wilk test, and $p > 0.05$ for every variable. We obtained effect sizes directly from the paired-test statistic using Cohen's

$$d_z = t/\sqrt{n} \quad (2)$$

where $n = 120$ is the number of pairs. We interpreted the resulting $d_z$values using the benchmarks proposed by Sawilowsky [13].

3. EXPERIMENTAL RESULTS

Table 2. Security Performance Comparison (mean over 120 paired runs per environment)

Metric	Traditional	Zero Trust	Improvement
MTTD (min)	38.4	6.9	82.0% reduction
MTTR (min)	71.8	14.2	80.2% reduction
ADT (min)	252	29	88.5% reduction
Lateral Movements (count)	12.4	1.7	86.3% reduction
Compromised Hosts (count)	23	3	87.0% reduction
Data Exfiltration (%)	78	11	85.9% reduction

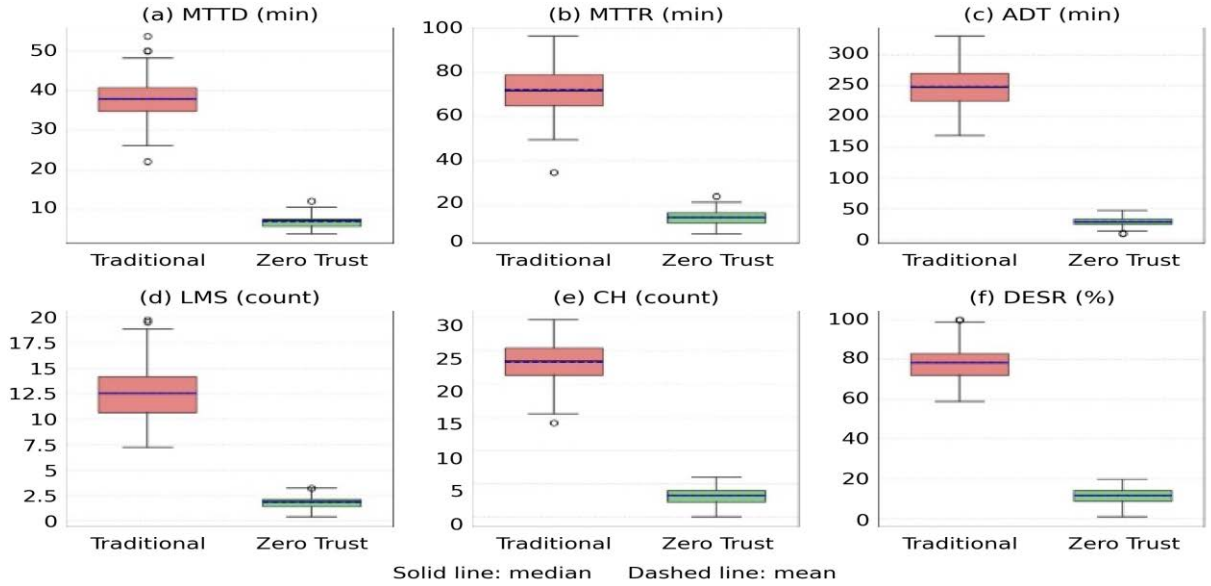


Figure 2. Comparison of the two environments over 120 paired runs. The plots present (a) MTTD, (b) MTTR, (c) ADT, (d) lateral movements, (e) compromised hosts, and (f) data exfiltration rate. The line inside each box marks the median, the box covers the middle 50% of observations, and the whiskers extend to 1.5 × IQR.

Attack detection took an average of 38.4 minutes in the traditional environment. With the controls used in Environment B, the corresponding time was 6.9 minutes. Response time followed the same pattern, decreasing from 71.8 to 14.2 minutes. This represents reductions of 82.0% and 80.2%, respectively. During the log review, we also found that containment generally started earlier in Environment B. Policy-violation alerts were one of the sources available to the analyst when deciding whether to initiate containment.

Dwell time showed the largest change. In the traditional setup, an attacker remained active for 252 min on average; in the Zero Trust setup, this value was 29 min (88.5% lower). A run in Environment A also produced 12.4 successful lateral transitions on average, compared with 1.7 in Environment B (86.3% lower). The mean number of affected hosts was 23 in the first environment and 3 in the second, corresponding to an 87.0% reduction. We observed the same direction of change for exfiltration: 78% of the predefined data set reached the external listener in Environment A, compared with 11% in Environment B. This is a 67-percentage-point difference, or an 85.9% relative reduction. Each comparison compared runs that shared the same ATT&CK technique and initial access vector.

4. STATISTICAL ANALYSIS

4.1 Paired-Samples t-Test

Table 3. Statistical significance results (paired runs, n = 120 per environment)

Variable	Traditional (mean ± SD)	Zero Trust (mean ± SD)	Mean diff.	SD(d)	t (df = 119)	p	Cohen's d_z
MTTD (min)	38.4 ± 6.2	6.9 ± 1.4	31.5	6.4	53.9	< 0.001	4.92
MTTR (min)	71.8 ± 11.5	14.2 ± 3.1	57.6	11.9	53.0	< 0.001	4.84
ADT (min)	252 ± 34	29 ± 7	223	34.7	70.4	< 0.001	6.43
LMS (count)	12.4 ± 2.8	1.7 ± 0.6	10.7	2.9	40.4	< 0.001	3.69
CH (count)	23 ± 3.1	3 ± 1.2	20.0	3.3	66.4	< 0.001	6.06
DESR (%)	78 ± 9	11 ± 4	67.0	9.9	74.1	< 0.001	6.77

The paired tests showed statistically significant differences between the two configurations ($p < 0.001$ for all evaluated variables). Under the null hypothesis of no paired mean difference, the reported p-values quantify how incompatible the observed test statistics are with that hypothesis. Cohen's d_z values derived from the paired t-statistics range from approximately 3.7 to 6.8, indicating large paired effects in the experimental dataset.

4.2 Pearson Correlation Analysis

We examined the relationship between detection time (MTTD) and attacker dwell time (ADT) across all runs. When we pooled all 240 runs, the correlation was $r = 0.87$ ($p < 0.001$). This pooled value partly reflects the large difference between the two environments rather than a within-environment relationship. When we analyzed each environment separately, the correlations were lower: $r = 0.42$ for the traditional setup and $r = 0.38$ for the Zero Trust setup. We therefore use the correlation results only to describe the observed data, not to infer causality.

5. CONTAINMENT ANALYSIS

5.1 Attack Containment Efficiency Calculation

Using the experimental data with TH = 28 (25 endpoints + 3 servers):

Traditional architecture:

$$ACE_{TRAD} = 1 - \frac{23}{28} = 0.179 \quad (3)$$

Zero Trust architecture:

$$ACE_{ZT} = 1 - \frac{3}{28} = 0.893 \quad (4)$$

ACE increased from 0.179 in Environment A to 0.893 in Environment B, giving a difference of 71.4 percentage points. This value comes directly from the compromised-host results in Table 2. We therefore use ACE only to express those results on a normalized scale, rather than treating it as a separate measurement.

6. DISCUSSION

The experimental results show that the Zero Trust-oriented configuration provided faster detection and containment and more effectively restricted post-compromise propagation than the traditional configuration under the evaluated ATT&CK scenarios.

Several security controls were active simultaneously in the Zero Trust environment, including micro-segmentation, ABAC, device-posture checks, application control, behavioral monitoring, periodic re-authentication, and policy enforcement. The measured improvement therefore cannot be assigned to any one of these controls. Our results describe the difference between the two configurations as they were tested.

We had to revise several rules while setting up the testbed. Some early rules restricted administrative tasks still necessary to operate the environment. After testing, we adjusted those rules so that the required tasks remained possible. This experience also showed a practical limitation of strict access control: security restrictions must be introduced without disrupting the administration of systems that need to remain available.

The 60-minute re-authentication requirement was likewise only one part of the setup: because a valid credential could still be abused between checks, behavioral monitoring, least-privilege restrictions, and session-termination mechanisms were used alongside it to identify suspicious activity and end a session when necessary.

Concrete Benefits for Military Systems:

- **Reduced Blast Radius:** Limiting the spread of a compromise can reduce the number of mission-supporting systems exposed after an attacker gains access.
- **Faster Incident Response:** Environment B combined the analyst's response workflow with automated session termination and segment isolation. The response-time comparison therefore reflects both the security controls and the response processes used in the testbed.
- **Operational Resilience:** In this setup, trust did not depend only on where a user or device was located in the network. Segmentation limited access between network areas, while identity policies and least-privilege rules restricted access to individual resources. Monitoring provided an additional check during active sessions. Together, these measures also allowed us to isolate an affected segment without necessarily disrupting services in other segments. This follows the basic idea of defense in depth [14].

7. LIMITATIONS OF THE STUDY

Our experiment compares the two environments as they were configured, not the effect of one security control in isolation. Several mechanisms used in Environment B were not present in Environment A, so we cannot assign the measured differences to any one of them. The experiment was also limited to 28 hosts and six ATT&CK techniques. We did not examine larger distributed networks, insider attacks, supply-chain compromise, DDIL operating conditions, or the performance cost of the implemented controls.

8. CONCLUSION

The two environments produced different results. In the 28-host testbed, Environment B detected attacks sooner and required less time to contain them. Attackers also completed fewer lateral transitions, reached fewer hosts, and exfiltrated less of the predefined test data. These results apply to Environment B as it was tested, with its security controls operating together. They do not show how much any one control contributed to the observed differences.

ACE expresses the compromised-host result on a normalized scale. Its value was 0.179 for Environment A and 0.893 for Environment B, a difference of 71.4 percentage points. The paired tests found significant differences for all variables included in the comparison. ATT&CK kept adversary actions organized consistently across runs and linked those actions to the observed defensive results.

These results come from testing all controls together. This experiment cannot determine which control produced which part of the difference. That would require another set of tests, changing one control at a time while leaving the rest of the setup unchanged.

A larger test environment would be a useful next step, particularly one that includes systems at more than one location. The attack set could also be expanded beyond the six ATT&CK techniques used here to cover insider activity, supply-chain compromise, and DDIL conditions. This experiment did not measure resource and performance costs, which should be included in further testing. Finally, testing the Zero Trust controls one at a time would help determine how much each control contributes to the observed results.

References

[1] VERIZON: 2024 Data Breach Investigations Report, Verizon Enterprise Solutions, 2024.

[2] CYBERSECURITY AND INFRASTRUCTURE SECURITY AGENCY (CISA): Zero Trust Maturity Model, Version 2.0, CISA, Washington, DC, 2023.

[3] U.S. DEPARTMENT OF DEFENSE: DoD Zero Trust Strategy, U.S. Department of Defense, Washington, DC, 2022.

[4] NATIONAL SECURITY AGENCY: Embracing a Zero Trust Security Model, Cybersecurity Information Sheet, NSA, 2021.

[5] ROSE, S., BORCHERT, O., MITCHELL, S., CONNELLY, S.: Zero Trust Architecture, NIST Special Publication 800-207, National Institute of Standards and Technology, Gaithersburg, MD, 2020, doi: 10.6028/NIST.SP.800-207.

[6] KINDERVAG, J.: Build Security Into Your Network's DNA: The Zero Trust Network Architecture, Forrester Research, Cambridge, MA, 2012.

[7] MITRE CORPORATION: MITRE ATT&CK Enterprise Matrix, 2026. [Online]. Available: https://attack.mitre.org.

[8] HUTCHINS, E. M., CLOPPERT, M. J., AMIN, R. M.: Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains, Leading Issues in Information Warfare & Security Research, Vol. 1, No. 1, 2011, pp. 80–106.

[9] GAIN, A., BARIK, M. S.: Augmenting Zero Trust Architecture with Temporal Attack Graph, 2025 IEEE 6th India Council International Subsections Conference (INDISCON), Rourkela, India, 2025, pp. 1–6.

[10] FRYDMAN, C., ZILBERMAN, A., KRIEF, R., SHOWGAN, A., MURILLO, A., MOTOYOSHI, S., SHABTAI, A., ELOVICI, Y., PUZIS, R.: COHORT: Collaborative Orchestration for Hardening via Offensive Replay on Emulated Topologies, arXiv preprint arXiv:2606.30479, 2026.

[11] ALSHAMRANI, A., MYNENI, S., CHOWDHARY, A., HUANG, D.: A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities, IEEE Communications Surveys & Tutorials, 21(2) (2019) 1851-1877.

[12] NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY: Security and Privacy Controls for Information Systems and Organizations, NIST Special Publication 800-53 Rev. 5, National Institute of Standards and Technology, Gaithersburg, MD, 2020, doi: 10.6028/NIST.SP.800-53r5.

[13] SAWILOWSKY, S. S.: New Effect Size Rules of Thumb, Journal of Modern Applied Statistical Methods, Vol. 8, No. 2, 2009, pp. 597–599, doi: 10.22237/jmasm/1257035100.

[14] STALLINGS, W., BROWN, L.: Computer Security: Principles and Practice, Pearson, Boston, MA, 2023.