

Article

Integrating Cyber Threat Intelligence into Email Gateway Architectures for Phishing and Ransomware Detection and Prevention

Mustafa Kara ¹, Hasan Burus ²  and Hasan Hüseyin Balık ^{3,*} 

¹ Department of Computer Engineering, Turkish Air Force Academy, National Defence University, Istanbul 34149, Türkiye; mustafa.kara@msu.edu.tr

² Department of Computer Engineering, Atatürk Strategic Studies and Graduate Institute, National Defence University, Istanbul 34334, Türkiye; hasan@burus.tr

³ Department of Engineering and Natural Sciences, Istanbul Atlas University, Istanbul 34408, Türkiye

* Correspondence: hasan.balik@atlas.edu.tr

Abstract

Email remains a major Initial Access vector for phishing and ransomware attacks, requiring security mechanisms that can combine multiple sources of threat evidence before malicious content reaches end users. Modern commercial email gateways already incorporate antivirus, reputation-based filtering, email authentication, and vendor-provided threat-intelligence mechanisms. However, their correlation logic, scoring criteria, decision thresholds, and automated response workflows are often proprietary or insufficiently documented. Moreover, transparent evaluations of the incremental benefit provided by external CTI enrichment and unified Indicator of Compromise (IoC) and Indicator of Attack (IoA) correlation remain limited. To address this gap, this study presents and experimentally evaluates a transparent CTI-supported email gateway workflow that integrates external intelligence feeds, IoC/IoA correlation, contextual threat scoring, predefined decision thresholds, and automated response actions. The proposed configuration was compared with a commercial baseline using the same vendor-native security controls in an isolated VMware ESXi-based environment. The evaluation included 1000 emails comprising 500 benign emails, 200 emails containing ransomware attachments, and 300 emails containing phishing URLs. The CTI-supported configuration achieved an overall detection rate of 98.8%, compared with 64.2% for the commercial baseline. Phishing URL and ransomware attachment detection rates reached 99.3% and 98.0%, respectively, while the false positive rate decreased from 4.5% to 1.2%. In the evaluated scenarios, the CTI-supported architecture interrupted the modeled attack chain at the Initial Access stage, demonstrating the measurable benefit of external CTI enrichment and unified correlation beyond the gateway's native controls.



Academic Editor: Hung-Yu Chien

Received: 1 July 2026

Revised: 16 August 2026

Accepted: 20 August 2026

Published: 24 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: cyber threat intelligence; email security; phishing detection; ransomware

1. Introduction

Today, the widespread use of technology and the Internet has also brought security risks for systems that are positioned at the center of digital life [1,2]. With the global proliferation of the Internet, especially since the early 2000s, cybersecurity has ceased to be merely a technical field of research and has become one of the fundamental components of daily life and organizational operations [3,4]. Along with this transformation, cybersecurity

practices, which were initially carried out without standards, have over time gained a more systematic structure through national and international frameworks [3]. In particular, the widespread use of digital communication infrastructures has caused cybersecurity measures to assume a more critical role over communication systems [5]. In this context, email systems are one of the fundamental tools of individual and corporate communication and hold a critical position in the digital ecosystem due to their wide area of use [6,7]. However, this widespread usage has made email an important initial vector for threat actors. Today, many cyber-attacks are initiated via email. Phishing, credential harvesting, and ransomware attacks stand out in this context [6–8]. In addition, methods such as the reuse of compromised accounts and the derivation of identity information from open-source data increase the impact of attacks [7–9].

Organizations generally resort to basic measures such as spam filters, simple signature checks, and user awareness training in order to ensure email security [6,10,11]. These approaches often remain insufficient against complex and advanced attacks [6,8,12]. Signature-based and rule-based systems show limited success, especially in detecting zero-day attacks, polymorphic malware, and dynamically changing phishing campaigns, since they rely on previously defined patterns [13–15]. In addition, the production of false positives by traditional filtering mechanisms may lead to the quarantine of critical business emails and disruption of business processes. The need for security teams to continuously check quarantine folders and manually whitelist safe messages also increases the operational load [12].

The increasing complexity of the cyber threat environment has caused not only system vulnerabilities but also human psychology to be targeted. Today, threat actors use more advanced structures that combine technical deception with social engineering. Generative artificial intelligence tools facilitate the production of phishing emails that do not contain grammatical errors and have high persuasive power [16,17]. The ability of LLM-based attack content to bypass traditional spam filters at a high rate has made the limitations of existing defense mechanisms more visible [8]. This situation increases the need for contextual and multi-layered analysis mechanisms that can predict the attack before it occurs, instead of filtering based only on content.

At this point, Cyber Threat Intelligence (CTI) stands out as a critical approach in email security. CTI is based not only on the collection of raw data, but also on processing data related to network telemetry, log records, Indicators of Compromise (IoCs), Indicators of Attack (IoA), and threat behaviors and transforming them into applicable and actionable intelligence [4,18]. In the literature, the main strength of CTI lies not in the passive reporting of indicators, but in making sense of threat actors' motivations, Tactics, Techniques, and Procedures (TTPs), and targeting methods [4,5]. Bringing together SIEM/EDR outputs and incident response findings obtained from internal sources with external threat reports, open-source data, and security bulletins increases the value of CTI applications [6]. When applied to email security, this approach creates a stronger defense ground by considering not only email content, but also multidimensional data such as the sender's global reputation, domain history, IP relationships, and behavioral patterns [7].

In the literature, studies examining the relationship between email security and CTI focus on increasing detection performance. In studies comparing traditional machine learning and deep learning approaches, accuracy rates above 98% have been achieved [19,20]. It has been shown that content-dependent and content-independent methods (for example, URL-based analyses) are effective and that language-independent detection is possible [21]. In addition, it has been reported that ensemble methods (especially stacking) provide superiority over many individual models with accuracy above 99% and high F1 scores [6,22,23]. Nevertheless, most existing artificial intelligence-based studies

focus on content classification, which is the process of classifying emails into categories such as spam, phishing, or benign by analyzing only the text, header, and content features of emails. In addition, these studies do not sufficiently evaluate the integrated effects on the email gateway. This situation may allow attackers to easily manipulate email content and bypass detection mechanisms.

This study targets the relevant gap and examines the integration of CTI data into corporate email gateway architecture and threat blocking performance. A structure supported by IoC/IoA is compared with a system that does not include CTI. The study focuses on phishing and ransomware attack chains. It addresses scenarios that start with email and include lateral movement and data encryption stages. In this scope, the effect of IP reputation, domain age, hash, and behavioral data at the gateway level is analyzed. However, some technical and operational limitations should also be considered when evaluating the results of the research. The study was carried out in a controlled laboratory environment, on virtual machines and an isolated network topology. It ensured secure analysis, but partially reflects the complexity of real-world network traffic, user diversity, and operational variability. In addition, the attack scenarios and phishing techniques used are based on threat intelligence data through simulation. User behaviors were simulated for technical measurement purposes. The instant reactions of real users to social engineering attacks and the dynamic effects of awareness training were excluded from the evaluation.

CTI integration is already employed in modern secure email gateways and operational security platforms through mechanisms such as threat intelligence feeds, URL and domain reputation, malicious IP and file-hash matching, sandboxing, and behavioral analysis. Therefore, this study does not claim that the integration of CTI into email gateway architectures is itself a novel concept. Instead, its contribution lies in presenting a transparent and reproducible engineering workflow that explicitly defines authentication verification, IoC and IoA correlation, contextual threat scoring, decision thresholds, and automated gateway response actions. Unlike proprietary commercial implementations, the proposed workflow enables the role of heterogeneous intelligence sources in gateway-level decisions to be independently examined under controlled phishing and ransomware scenarios.

Main contributions of the study:

- A transparent and reproducible CTI-driven email gateway workflow is presented, explicitly defining authentication verification, IoC/IoA correlation, threat scoring, decision thresholds, and gateway response actions.
- The operational contribution of CTI integration is evaluated by comparing CTI-supported and conventional gateway configurations under identical phishing and ransomware scenarios.
- The proposed framework experimentally demonstrates how heterogeneous intelligence sources contribute to interrupting email-based attack chains at the Initial Access stage.

In the remainder of the study, Section 2 reviews the literature, Section 3 presents CTI, IoC, and IoA concepts, Section 4 explains the methodology and experimental setup, Section 5 reports the results, Section 6 discusses the findings, and Section 7 concludes the study.

2. Literature Review

The use of CTI in email security has become an important research area, especially in the detection of phishing and malicious content. Studies in the literature focus on the integration of CTI data into automated detection systems, methods that provide high accuracy and low false positives, and real-time defense requirements. When the approaches

in the literature are examined, it is observed that machine learning and deep learning-based classification methods predominantly stand out.

One of the significant contributions in this field was presented by Korkmaz [19]. In the study, conducted on a dataset consisting of 17,538 emails, traditional machine learning methods and deep learning techniques were compared using TF-IDF-based feature extraction. It is noteworthy that traditional methods such as Naive Bayes, XGBoost, and Random Forest demonstrated higher performance compared to more complex deep learning architectures such as Bi-LSTM and GRU. The fact that traditional models achieved an accuracy of 98.52%, while deep learning models reached 97.71%, shows that computationally expensive methods are not always superior in processing CTI data. This situation is considered an important finding that shifts the research direction toward task-appropriate and cost-effective solutions rather than more complex models.

In terms of deep learning-based approaches, the method proposed by Alhogail and Alsabih [20] is also noteworthy. The researchers developed a classifier that combines Graph Convolutional Networks (GCN) with Natural Language Processing (NLP) techniques in the analysis of email texts. The model, achieving 98.2% accuracy and a 0.015 false positive rate, demonstrates that text mining-based deep learning approaches provide a strong alternative in CTI processes. In addition, the low false positive rate stands out as a critical performance indicator in terms of not disrupting legitimate traffic in security operations.

In the literature, it is observed that not only model selection but also data representation and feature selection are decisive for performance. Bari et al. [16] combined ANOVA and Mutual Information-based feature selection methods with a stacking ensemble architecture. The system was tested on three datasets, achieving 98.17% accuracy and a 1.31% false positive rate, which shows that meaningful feature extraction increases detection success rather than directly processing raw data. Similarly, the stacking ensemble model proposed by Connolly and Atlam [22] achieved superior performance compared to individual models, with 99.53% accuracy and a 0.9955 F1 score. Such studies reveal that ensemble methods can achieve high accuracy, especially due to their ability to combine different data sources.

On the other hand, some studies show that content-independent approaches can also be effective instead of content-dependent analysis. Sahingoz et al. [21] proposed a language-independent and real-time detection mechanism through analyses performed on URL strings. The Random Forest model achieving 97.98% accuracy shows that even only URL structural features can be a strong CTI source. This approach provides advantages especially in terms of user privacy and processing cost. However, it is emphasized in the literature that although content-based methods provide contextual richness, they involve higher computational cost and privacy risks, whereas URL and structural analyses are faster but provide limited contextual information [16].

Approaches for the classification of malicious emails have not been limited to binary classification, and the need for more detailed threat analysis has also been revealed. Jáñez-Martino et al. [24] performed a multi-class analysis by dividing emails into 11 different threat categories. With the use of OCR, the detection of hidden text and image-based spam techniques has been made possible, and an accuracy rate of 96.4% was achieved with the combination of TF-IDF and Logistic Regression. This study shows that computationally efficient methods can also produce effective results instead of more complex models.

Figure 1 summarizes these studies by comparing their detection focus and reported performance accuracy, showing that high detection rates are achieved across different CTI-based email security approaches.

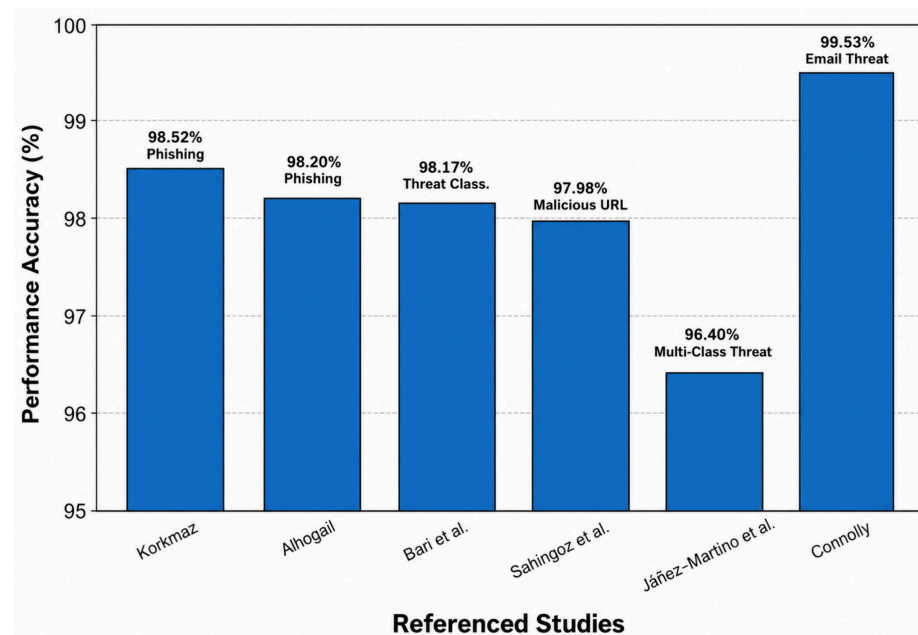


Figure 1. Comparative detection focus and performance of existing email security studies [16,19–22,24].

Although the reviewed studies report high detection performance, their strengths and limitations differ considerably. Traditional machine-learning approaches provide competitive accuracy with relatively low computational complexity, whereas deep-learning and ensemble methods can improve feature representation and classification performance at the cost of increased computational requirements. URL-based approaches offer advantages in terms of processing efficiency and privacy but provide less contextual information than content- and behavior-based methods. Similarly, multi-class classification approaches provide more detailed threat categorization but remain primarily focused on dataset-level classification rather than operational gateway decisions. Thus, the literature demonstrates a trade-off between detection accuracy, computational complexity, contextual awareness, and operational applicability.

When the studies in the literature are evaluated together, criteria such as high accuracy, low false positive rates, real-time analysis, computational cost, and feature selection emerge as key considerations in detecting email-based threats. However, most existing studies primarily focus on classifying emails as benign, phishing, or malicious based on text, header, URL, or attachment features and evaluate performance using dataset-level metrics. Although these approaches are valuable for algorithmic comparison, they provide limited insight into how CTI is operationally integrated into corporate email gateway decision processes, whether malicious content reaches the end user, and at which stage the attack chain is interrupted. In contrast, this study proposes and experimentally evaluates a CTI-enabled email gateway architecture that correlates IoCs, IoAs, authentication results, URL and attachment analyses, and contextual threat intelligence to support delivery, quarantine, and blocking decisions. Thus, the study addresses an architectural and operational research gap beyond conventional email classification approaches.

3. IoC- and IoA-Based Cyber Threat Intelligence Approaches and Their Applications in Operational Security

CTI is the process of collecting, analyzing, and transforming data into intelligence in order to make sense of and prevent cyber threats targeting the digital assets of organizations [4,5]. In the CTI process, two basic metrics are used for the identification and classification of threats: Indicators of Compromise (IoC) and Indicators of Attack (IoA).

IoC refers to indicators that serve as digital evidence confirming that access to a network has been obtained or that a security breach has occurred. IoC is defined as indicators obtained as a result of analyses carried out after a cyber attack and representing the digital traces of the attack. Within the scope of email security, the most commonly used IoC types include hash values belonging to malicious files, IP addresses connected to command-and-control (Command and Control, C2) infrastructure, and URL structures used in phishing attacks [11]. In the detection of phishing attacks, filter-based feature selection processes for IoC data such as URL and IP reputation increase detection accuracy [16].

IoA refers to indicators focusing on the threat actor's purpose and the methods applied, namely TTP. While IoCs mostly reveal the technical traces obtained after an attack, the IoA approach focuses on the occurrence process and behavioral patterns of the attack. In this context, while IoCs answer the question "what happened?", IoAs answer the questions "how did it happen?" and "why did it happen?". The IoA approach enables the detection of an ongoing cyber-attack before the threat actor reaches the target [25]. Especially in artificial intelligence-supported phishing attacks, semantic analysis of email content, sender behavior patterns, and unusual communication patterns are among the indicators evaluated within the scope of IoA.

One of the most critical roles of CTI in operational security systems is transforming the outputs of malware analysis performed in isolated environments into immediately actionable intelligence for email gateways and endpoint defense mechanisms. In traditional analysis processes, static file hashes (SHA-256), C2 server IP addresses, and malicious domain names obtained through reverse engineering or dynamic analysis are, on their own, only raw data. In the "processing and analysis" stages of the CTI lifecycle, these raw data are interpreted and transformed into indicators that reflect the threat actor's tactics, techniques, and procedures (TTPs) and can be consumed by operational systems. Through this intelligence, which is fed instantly to operational systems via standard protocols such as STIX/TAXII, the security architecture gains the ability to disrupt the attack chain at the "Initial Access" stage by identifying not only a specific malware sample but also the associated infrastructure of the phishing campaign.

Figure 2 illustrates the comparative roles of IoA and IoC in CTI. IoA supports pre-attack behavioral detection by identifying attacker intent and tactical anomalies. IoC enables post-compromise identification through static artifacts such as hashes, URLs, and IP addresses. Both are integrated through the CTI engine for correlation, real-time detection, and pre-delivery quarantine.

As shown in Figure 3, the process of cyber threat intelligence transforming from raw data into meaningful intelligence takes place within a systematic lifecycle. The continuous and uninterrupted operation of the cyber threat intelligence lifecycle is of critical importance for effective cyber defense [26]. The CTI lifecycle generally consists of data collection, processing, analysis, and dissemination stages.

In the data collection stage, raw data are obtained from open-source intelligence (OSINT), social media platforms, technical telemetry data, and dark web sources. The automatic extraction of cyber threat intelligence from social media data is important for early warning systems [27]. Data obtained from illegal online marketplaces can be used in predicting ransomware operations [9]. Making the collected data meaningful takes place in the processing stage. At this stage, raw data are filtered, cleaned, and made suitable for analysis. Machine learning techniques play a critical role in intelligence-oriented data processing processes in terms of classifying and correlating large datasets [28]. Interpreting processed data and obtaining meaningful inferences about threats takes place in the analysis stage. At this stage, human analysts or automated analysis systems examine the behavior patterns of threat actors and reveal attack scenarios. The transfer of the obtained intelligence

to security infrastructures is carried out in the dissemination stage. In this process, the generated intelligence is transmitted to security devices (Firewall, Secure Email Gateway, etc.) and operational teams (Security Operations Center—SOC). Performing intelligence sharing through automated protocols increases resilience against cyber attacks across the system and network [18].

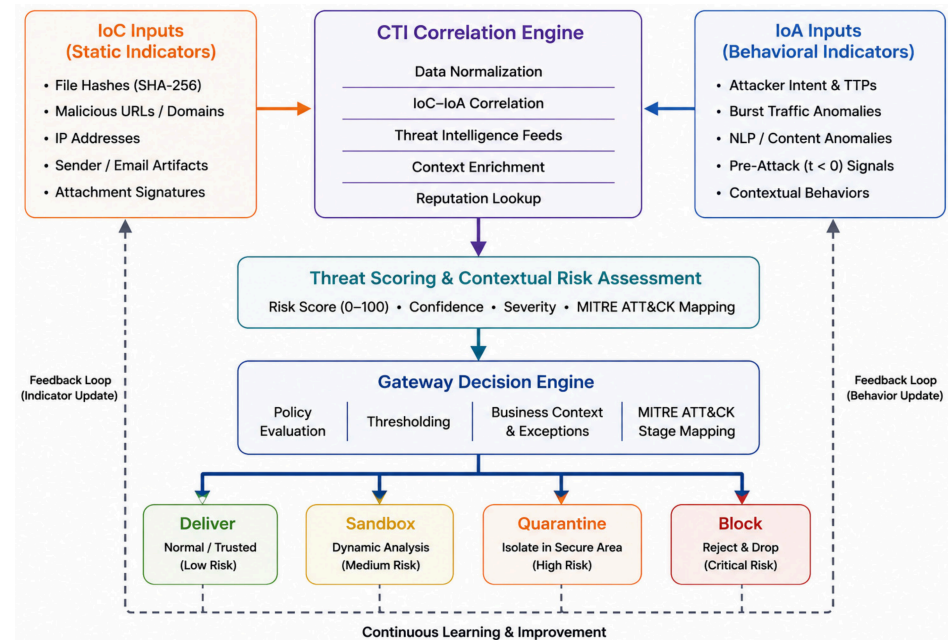


Figure 2. CTI-driven email gateway workflow with IoC/IoA correlation and response actions.



Figure 3. Cyber threat intelligence lifecycle from data collection to operational dissemination.

4. Methodology

In the first scenario of the experimental study, the email gateway product operates in an integrated manner with cyber threat intelligence. This architecture evaluates incoming emails within a multi-layered analysis process by using threat intelligence obtained at strategic, operational, and tactical levels in accordance with the methodology. In the analysis process, domain name and IP address reputation checks, known malicious hash values, signature-based detection mechanisms, and behavioral analysis methods are used simultaneously. In this way, the cyber-attack chain is interrupted at the “Initial Access” stage, and the cyber-attack is prevented from negatively affecting the corporate network and systems.

In the second scenario of the experimental study, an architecture in which cyber threat intelligence methodologies are not applied or are applied at a limited level in the email gateway product is addressed. This configuration uses the vendor-native security capabilities of the commercial email gateway but does not include external CTI enrichment, unified IoC/IoA correlation, or contextual threat scoring. This situation makes it difficult to detect advanced attack techniques used by current threat actors. Therefore, malicious email attachments and links can reach end-user systems, and risks dependent on user awareness emerge. When the malicious email reaches the end user's mailbox and the user executes the malicious content, the malware infects the endpoint and communicates with the C2 infrastructure. Then, the threat actor can perform lateral movement on the network surface by using the infected device. As a result of this process, not only end-user devices but also servers, network devices, and embedded systems can be infected by the malware.

After the malware successfully infiltrates the system, the attack progresses to the "Execution", "Persistence", and "Privilege Escalation" stages according to the MITRE ATT&CK framework. In line with the privileges obtained, the attacking software spreads to other systems within the corporate network by using lateral movement techniques. In this process, spread within the network is achieved by exploiting unpatched security vulnerabilities. In the final stage, the ransomware starts the data encryption process. The data encryption process can affect many systems, especially critical servers and user computers. The data on these devices are protected in terms of data confidentiality by using encryption algorithms. Threat actors offer to provide the decryption key to the target organization in exchange for a ransom payment. If the ransom is not paid, they threaten that the data will be published publicly, and in some cases, the data may be sold to third parties.

4.1. Simulation Environment and Performance Evaluation Criteria

In this section, a controlled simulation environment was established in order to analyze the effect of CTI integration on email security and to compare two different architectures. The experimental infrastructure was configured on the VMware ESXi virtualization platform in a laboratory environment completely isolated from external networks. The main purpose of preferring an isolated environment is to ensure that malware and phishing scenarios are tested safely without harming real systems and to guarantee the reproducibility of experimental results. The simulation architecture was divided into two logical zones as "Attacker" (Red Team) and "Target/Victim" (Blue Team) in a way that represents attack and defense behaviors. The software components, versions, and functional roles used in the experimental environment are summarized in Table 1.

These components established an isolated architecture integrating a commercial email gateway with open-source CTI platforms. The experimental architecture combined a commercial email security gateway with open-source CTI platforms. Symantec Messaging Gateway provided vendor-native email security controls, while OpenCTI and MISP were used to collect, normalize, enrich, and correlate external threat intelligence. This configuration enabled incoming emails to be evaluated using both native gateway controls and external IoC/IoA evidence.

In the simulation topology, two different email gateway configurations were tested: Baseline and Secured. In the experimental evaluation, Symantec Messaging Gateway (SMG) was used as the commercial email security gateway. In the baseline configuration, the vendor-provided security capabilities used in the study were enabled and configured according to the recommended security settings. These capabilities included anti-spam filtering, antivirus scanning, SPF, DKIM, and DMARC verification, the vendor's proprietary URL reputation service, global bad-sender IP feeds, and industry-specific threat-blocking

policies. Therefore, the baseline represents a commercial-like email security configuration commonly deployed in enterprise environments rather than a system relying solely on static signatures and keyword filtering. No external CTI platform or custom correlation layer was connected to the baseline configuration. The CTI-supported configuration retained the same vendor-native controls and additionally incorporated external IoC data from MISP and OpenCTI, including malicious URLs, IP addresses, domains, and file hashes, together with IoA-based behavioral analysis and unified IoC/IoA correlation. Consequently, the comparison evaluates the added value of external CTI enrichment, contextual correlation, and CTI-driven gateway decisions beyond the native capabilities of the commercial gateway.

Table 1. Software components used in the experimental environment.

Component	Software/Product	Version	Role in the Study
Email security gateway	Symantec Messaging Gateway	10.9.0	SMTP filtering, anti-spam, antivirus, URL reputation, and SPF/DKIM/DMARC verification
CTI platform	OpenCTI	6.4	Contextual enrichment and IoC/IoA correlation
Threat-sharing platform	MISP	2.5.30	IoC collection, management, and sharing
Virtualization platform	VMware ESXi	7.0	Isolated experimental infrastructure
Server operating system	Ubuntu Server	24.04	Hosting of CTI and integration services
Email server	Microsoft Exchange Server	2022	Simulation of corporate email infrastructure
Attack platform	Kali Linux	2024.1	Phishing and malicious-email simulation

In this study, standard performance metrics widely used in the literature were employed to quantitatively evaluate both systems. It is stated that the success of email security systems should be measured not only by their ability to detect threats, but also by their ability not to block legitimate traffic [16].

Figure 4 comparatively shows the behaviors of the CTI-supported email security architecture and the standard architecture under the same attack scenario. In the CTI-Enabled System section, when the malicious email sent by the attacker over the Internet reaches the CTI-supported gateway, it is matched in real time with IoC and IoA indicators and is blocked before entering the internal network. The gateway rejects the malicious content before it reaches the user mailbox by using a rule base updated through CTI feeds. Thus, end-user computers and servers are kept out of the attack chain, and the threat is prevented from accessing the internal network. In the commercial baseline system section, the same malicious email passes through the standard gateway without CTI integration without being detected and reaches the end user. With the execution of malicious content, ransomware becomes active on the endpoint and encrypts the data on the user system. Then, lateral movement and propagation mechanisms are activated, causing the malware to reach corporate servers and lead to data loss. This scenario shows that a single detection deficiency at the gateway level can put the entire corporate infrastructure at risk.

The fundamental difference between the two architectures emerges at the stage at which the threat is blocked. In the CTI-supported architecture, defense is carried out at the gateway level, and the malicious email content is blocked before reaching the internal network. In contrast, in the standard architecture, since the threat cannot be detected at the gateway, it infiltrates the internal network, and the attack chain continues at the endpoint level. This causes the attack to affect not only end-user systems but also the entire corporate infrastructure through lateral movement.

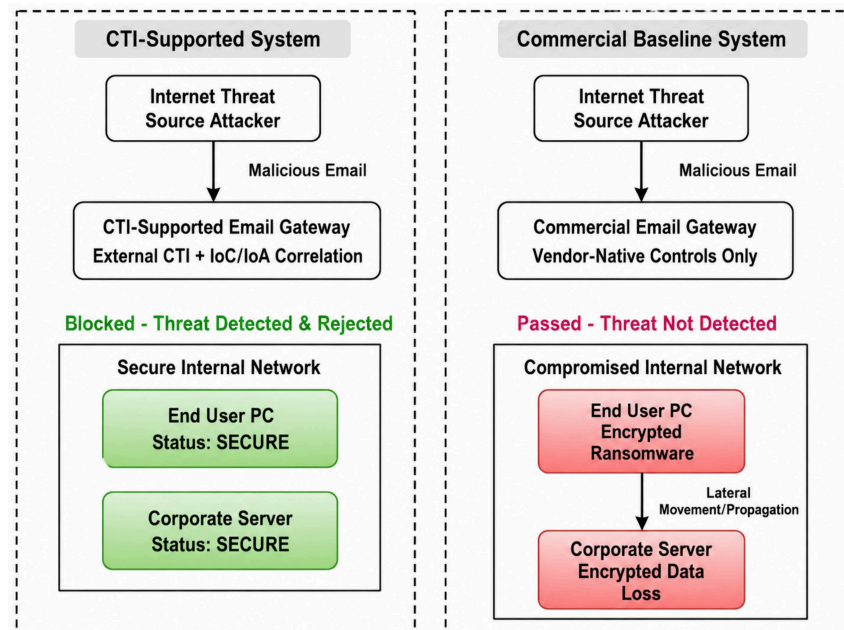


Figure 4. Comparative attack progression in commercial baseline and CTI-supported architectures.

Within the scope of performance evaluation, three fundamental metrics widely used in the literature were considered:

- Accuracy;
- Detection Rate (Recall);
- False Positive Rate (FPR).

Accuracy refers to the rate at which the system correctly classifies malicious and benign content within the total email traffic. Detection Rate (Recall) indicates the proportion of successfully detected emails among the total malicious emails sent to the system and is a critical performance indicator, especially in ransomware scenarios. False Positive Rate (FPR) refers to the rate at which legitimate emails are incorrectly classified as malicious and is considered an important metric in terms of operational continuity.

4.2. Comparative Gateway Configurations and CTI Decision Workflow

The comparative analysis model established to test the main hypothesis of the study examines how the same attack vectors produce results under two different defense architectures. In this context, malicious emails containing phishing links and ransomware were tested on both the CTI-supported secure architecture and commercial baseline configuration. Thus, the study comparatively evaluated the stage at which each architecture detected, blocked, or allowed the attack to progress into the internal network.

4.2.1. Scenario 1—Cyber Threat Intelligence-Supported Secure Architecture

In this scenario, the email gateway is configured in accordance with cyber threat intelligence methodologies. The system analyzes each incoming email by comparing it with IoC and IoA indicators obtained from global threat databases. The main purpose of this architecture is to detect malicious email content before it reaches the user and to disrupt the attack chain at the “Initial Access” stage.

4.2.2. Scenario 2—Commercial Baseline System

In this scenario, SMG 10.9.0 was operated using its vendor-native security capabilities and recommended configuration settings. The enabled controls included anti-spam filtering, antivirus scanning, SPF, DKIM, and DMARC verification, the vendor’s proprietary

URL reputation service, global bad-sender IP feeds, and industry-specific threat-blocking policies. Therefore, the baseline represents a commercial enterprise email security gateway rather than a system relying solely on static signatures and keyword filtering.

The CTI-supported configuration retained all of these vendor-native controls. In addition, it incorporated external URL, IP, domain, and file-hash IoCs through MISP and OpenCTI, IoA-based behavioral analysis, contextual threat scoring, and unified IoC/IoA correlation. Thus, both configurations shared the same native commercial security mechanisms, while the CTI-supported system introduced an additional external intelligence and correlation layer. This design enables the experimental comparison to isolate the operational contribution of external CTI enrichment and contextual IoC/IoA correlation beyond the native capabilities of the commercial gateway.

4.2.3. CTI-Enabled Email Gateway Decision Workflow

In the CTI-supported email gateway architecture, the email analysis process was designed to progress from low-cost verification mechanisms to more computationally intensive contextual and behavioral analyses. After an incoming email is received via SMTP, the gateway first extracts the sender IP address, sender domain, Reply-To field, email headers, embedded URLs, attachments, and authentication information. Subsequently, SPF, DKIM, and DMARC verification are performed to validate sender identity, domain authorization, and message integrity. Following authentication, the sender IP address, domain name, URLs, and available attachment hash values are compared with IoCs obtained from multiple CTI sources, including MISP, OpenCTI, OSINT, and commercial threat intelligence feeds. If a verified high-confidence IoC match is identified, the email is immediately quarantined or blocked without requiring further analysis. Otherwise, the gateway continues with URL and attachment inspection. URL analysis evaluates domain reputation, typosquatting patterns, redirect chains, shortened URL usage, and phishing page characteristics. Attachment analysis examines file type, antivirus detection results, cryptographic hash values, embedded macros, executable content, and sandbox execution behavior.

The outputs generated during the previous stages are then forwarded to the IoA evaluation module, where behavioral indicators are correlated to identify potential attack patterns. For example, combinations of SPF or DMARC failures, spoofed domains resembling legitimate corporate domains, suspicious redirect chains, abnormal email transmission behavior, and PowerShell or command-line execution attempts observed during sandbox analysis collectively increase the attack probability. Finally, all technical and behavioral findings are integrated within the CTI correlation engine, where contextual information such as indicator confidence, source reliability, threat relevance, and indicator recency are incorporated into the overall threat assessment.

This sequential processing order was intentionally adopted to optimize both detection performance and computational efficiency. Authentication mechanisms (SPF, DKIM, and DMARC) provide an early trust assessment with minimal processing overhead, while IoC matching rapidly identifies previously known threats. URL and attachment analyses require additional computational resources and therefore are performed only when necessary. IoA evaluation is executed after these stages because it relies on the contextual evidence collected throughout the entire inspection process. Consequently, the final decision is not based on a single security indicator but on the correlation of multiple independent evidence sources, reducing false positives while improving the detection of sophisticated phishing and ransomware attacks.

The cyber threat intelligence data used in this study were obtained from four main source categories: open-source intelligence, commercial threat intelligence feeds, threat

intelligence platforms, and internal organizational security logs. IoC data such as IP addresses, domain names, URLs, and file hash values obtained from these sources were used in sender reputation analysis, URL inspection, and malicious attachment detection processes performed at the email gateway. The CTI data sources used in this study, the data they provide, and their usage in the proposed architecture are summarized in Table 2.

Table 2. CTI data sources used in this study.

Data Source	Provided Data	Usage in This Study
OSINT threat feeds	Malicious URLs, domains, IP addresses, and file hash values	Detection of known phishing and malware indicators
Commercial threat feeds	IP/domain reputation, confidence score, and threat classification	Verification and enrichment of indicators
MISP	IoCs, events, campaigns, and tag information	Collection, management, and sharing of indicators
OpenCTI	IoC relationships, threat actor, campaign, and ATT&CK context	Correlation and contextual analysis
Sandbox and antivirus	File hash values and behavioral analysis results	Verification of malicious attachments
Email gateway and SIEM logs	Sender IP, URL, header, attachment, and event records	Correlation of external intelligence with organizational data

Within the scope of open-source intelligence, community-based threat feeds containing malicious URLs, phishing domains, command-and-control addresses, and malware indicators were used. Commercial threat intelligence feeds provided additional contextual information such as domain and IP reputation, malware classification, and indicator confidence scores. Internal organizational sources included email gateway logs, DNS logs, proxy logs, SIEM records, EDR alerts, and sandbox outputs. These internal records were used to compare externally obtained indicators with observations within the organizational environment. MISP was used to collect, tag, manage, and share IoC data in relation to events and campaigns. OpenCTI was used to associate indicators obtained from different sources with threat actors, malware families, campaigns, and MITRE ATT&CK techniques.

In this context, MISP and OpenCTI were not considered the original producers of all indicators; rather, they were used as CTI platforms where indicators obtained from different threat feeds were collected, normalized, enriched, and correlated. IoC data were ingested into the CTI platforms through STIX/TAXII, APIs, or structured file transfers. Before being forwarded to the email gateway, the collected indicators were processed through deduplication, format normalization, timestamp validation, and source verification. Each indicator was evaluated according to recency, confidence score, relevance, and source reliability. Outdated or low-confidence indicators were not directly used for blocking decisions; instead, they were treated as supporting indicators requiring additional analysis. At the email gateway, the sender IP address and domain name were compared with reputation and blacklist records available in the CTI platforms. URLs embedded in emails were matched with malicious link and phishing indicators, while hash values extracted from attachments were compared with malware records. The matching results were then correlated with SPF, DKIM, and DMARC checks, URL analysis, sandbox outputs, and IoA evaluations, and were incorporated into the final delivery, secondary inspection, quarantine, or blocking decision.

CTI feeds were automatically synchronized according to the intervals shown in Table 3. Newly received indicators were normalized, deduplicated, timestamp-validated, and enriched within OpenCTI and MISP before being transferred to the gateway integration layer. Vendor reputation information was queried through the native SMG infrastruc-

ture, whereas external IP, URL, domain, and file-hash indicators were obtained through STIX/TAXII and REST API connections.

Table 3. CTI feeds, data types, integration methods, and update intervals used in the proposed email security architecture..

Feed	Source	Data Type	Integration Method	Update Interval
Vendor reputation	Broadcom/Symantec	URL, IP, and domain reputation	Native integration	Real time
MISP feed	MISP	IoCs	STIX/TAXII	Every hour
OpenCTI connector	OpenCTI	IoCs and IoA context	REST API	Every hour
AbuseIPDB	OSINT	Malicious IP addresses	REST API	Daily
URLhaus	OSINT	Malicious URLs	REST API	Hourly
PhishTank	OSINT	Phishing URLs	REST API	Hourly
MalwareBazaar	OSINT	SHA-256 file hashes	REST API	Daily

At the email gateway, IoCs such as malicious IP addresses, domain names, URLs, and file hash values were evaluated together with indicator confidence, recency, source reliability, and outputs from other security controls. For each email, a threat score between 0 and 100 was calculated, and the corresponding security action was selected according to predefined decision thresholds. These thresholds and the associated gateway actions are summarized in Table 4.

Table 4. Decision thresholds and security actions according to IoC detections.

Detection Condition	Threat Score/Condition	Security Action
Verified malicious file hash match	Critical IoC or source confidence $\geq 90\%$	The email is directly blocked and the attachment is quarantined.
Verified phishing URL or malicious domain match	Critical IoC or source confidence $\geq 90\%$	URL access is blocked, the email is quarantined, and the proxy/firewall blocklist is updated.
Known malicious sender IP address	High-confidence IoC and additional risk indicator present	The SMTP session is rejected or the message is quarantined.
Multiple suspicious indicators	Score 70–100	The email is blocked or quarantined. 70–89: Quarantine. 90–100: Block.
Unconfirmed but risky findings	Score 40–69	Email delivery is suspended and the message is sent to sandbox or analyst review.
Low-risk or contextual indicator	Score 20–39	The email is delivered with a warning label and a monitoring log is generated.
No malicious match	Score 0–19	The email is delivered normally to the user mailbox.

If a critical IoC match obtained from a verified and highly reliable source was detected, the email was blocked or quarantined without waiting for the total threat score. For example, if the hash value of an attachment matched a verified ransomware sample, the email was considered sufficiently malicious to be blocked before reaching the user mailbox. Similarly, a verified phishing URL or malicious domain match triggered URL blocking, email quarantine, and updating of the proxy or firewall blocklist. In contrast, the presence of only a low-reputation IP address or a newly registered domain did not automatically result in blocking; instead, it was correlated with other findings.

The final decision was not based solely on a single IoC match. SPF, DKIM, and DMARC results, URL and domain reputation, attachment analysis, sandbox outputs, and IoA-based behavioral indicators were evaluated within a common correlation process. For example, a low-reputation domain alone may indicate a moderate level of risk; however,

when DMARC failure, typosquatting, and a fake login page are detected in the same email, the total threat score may exceed the blocking threshold.

The operational effectiveness of the CTI-supported architecture depends on the freshness of threat intelligence data. Therefore, the threat intelligence data used in this study were updated throughout the experimental period. Community-based OSINT feeds and commercial threat intelligence feeds were synchronized periodically based on their availability. Internal security logs, including email gateway, SIEM, EDR, DNS, proxy, and sandbox logs, were processed immediately after collection. Newly received indicators were validated before being integrated into the CTI platform. This validation included deduplication, format normalization, timestamp verification, and source reliability assessment. Indicator recency was also considered as a decision parameter because the operational value of threat intelligence decreases over time. Indicators that exceeded their operational validity period or had low confidence scores were not directly used for blocking decisions. Instead, they were treated as supporting contextual evidence. These indicators required additional correlation with authentication results, behavioral analysis, or other IoCs before affecting the final gateway decision. This update strategy prevents outdated indicators from increasing false positives and allows newly emerging threats to be included in the detection process when updated intelligence becomes available.

In the proposed workflow, CTI was used in three roles: pre-filtering, decision support, and final enforcement. In the pre-filtering stage, sender IP addresses, domains, URLs, and attachment hash values were compared with known IoC records. In the decision support stage, CTI outputs were correlated with SPF, DKIM, DMARC results, URL analysis, attachment inspection, sandbox outputs, and IoA-based behavioral indicators. In the final enforcement stage, only verified and high-confidence critical IoCs, such as confirmed ransomware hashes, phishing URLs, or malicious C2 domains, directly triggered blocking or quarantine. For all other cases, the final action was determined according to the combined threat score and predefined decision thresholds. The input variables used by the CTI-enabled gateway decision workflow are summarized in Table 5. These variables represent the email content, authentication results, extracted IoC candidates, and external threat intelligence records processed by Algorithm 1.

Table 5. Input variables used in Algorithm 1.

Variable	Description	Data Type	Source
E	Incoming email	SMTP message	SMG
H_{dr}	Email headers	Header fields	SMTP message
IPs	Sender and relay IP addresses	IoC candidates	SMTP headers
Ds	Sender and embedded domains	IoC candidates	Headers and message body
Us	Embedded URLs	IoC candidates	Message body
As	Email attachments	MIME objects	MIME parser
SPF	SPF verification result	Boolean/status	DNS verification
DKIM	DKIM verification result	Boolean/status	Signature verification
DMARC	DMARC verification result	Boolean/status	DNS and policy verification
CTI_Feeds	External threat intelligence records	STIX/TAXII objects	OpenCTI and MISP

File hashes were calculated from the attachments represented by As using SHA-256 and compared with antivirus, sandbox, MISP, OpenCTI, and MalwareBazaar records. Therefore, the decision process incorporated both message-level technical evidence and external threat intelligence rather than relying solely on email content.

Algorithm 1 summarizes the decision-making workflow implemented by the CTI-enabled email gateway. The algorithm illustrates how incoming emails are processed through authentication verification, IoC matching, IoA-based behavioral analysis, and CTI

correlation. Based on the computed threat score and critical threat indicators, the gateway determines whether the email should be delivered, forwarded for sandbox analysis, quarantined, or blocked.

Algorithm 1: CTI-Enabled Email Gateway Decision Function

Input: E, H, IPs, Ds, Us, As, SPF, DKIM, DMARC, CTI_Feeds

Output: Decision $\in$ {Deliver, Sandbox, Quarantine, Block}

```

1: receive incoming email E;
2: extract Hdr, IPs, Ds, Us, As from E;
3: AuthRisk  $\leftarrow$  CheckAuthentication(SPF, DKIM, DMARC, H);
4: HeaderRisk  $\leftarrow$  AnalyzeHeader(H);
5: IoCScore  $\leftarrow$  MatchIoC(IPs, Ds, Us, As, CTI_Feeds);
6: AttachRisk  $\leftarrow$  AnalyzeAttachment(As);
7: IoAScore  $\leftarrow$  AnalyzeBehavior(E);
8: CTIContext  $\leftarrow$  CorrelateCTI(IoCScore, IoAScore, CTI_Feeds);
9: S(E)  $\leftarrow$   $w_H \cdot \text{HeaderRisk} + w_D \cdot \text{IPDomainRisk} + w_U \cdot \text{URLRisk} + w_A \cdot \text{AttachRisk} +$ 
 $w_B \cdot \text{IoAScore} + w_C \cdot \text{CTIContext}$ ;
10: if CriticalIoCMatch(E) == true then
11:   Decision  $\leftarrow$  Block;
12: else if HighRiskIoABehavior(E) == true then
13:   Decision  $\leftarrow$  Quarantine;
14: else if S(E)  $\geq T_3$  then // ( $T_3 = 90$ : Block)
15:   Decision  $\leftarrow$  Block;
16: else if S(E)  $\geq T_2$  then // ( $T_2 = 70$ : Quarantine)
17:   Decision  $\leftarrow$  Quarantine;
18: else if S(E)  $\geq T_1$  then // ( $T_1 = 40$ : Sandbox or analyst review)
19:   Decision  $\leftarrow$  Sandbox;
20: else
21:   Decision  $\leftarrow$  Deliver;
22: end if
23: send Decision, S(E), and matched indicators to SIEM/SOAR;
24: update CTI engine using analyst feedback;
25: return Decision;
```

$$S(E) = 100(w_H H + w_D D + w_U U + w_A A + w_B B + w_C C) \quad (1)$$

where $S(E)$ denotes the overall threat score of the incoming email; H represents the header and authentication risk; D denotes the sender IP/domain reputation score; U represents the URL risk score; A denotes the attachment risk score; B represents the IoA-based behavioral risk score; C denotes the CTI contextual confidence score; and w_H, w_D, w_U, w_A, w_B , and w_C represent the relative weights assigned to each security component in the decision-making process. The decision thresholds T_1, T_2 , and T_3 are used to classify incoming emails as Deliver, Sandbox, Quarantine, or Block, respectively. All component scores were normalized to the range $[0, 1]$ before calculating the overall threat score. In Equation (1), H represents header and authentication risk, D represents sender IP and domain reputation risk, U represents URL risk, A represents attachment risk, including file-hash matching, antivirus inspection, and sandbox results, B represents IoA-based behavioral risk, and C represents CTI contextual risk based on indicator confidence, source reliability, relevance, and recency.

Within the CTI contextual component C , indicator confidence, source reliability, relevance, and recency were represented on normalized $[0, 1]$ scales and incorporated through the predefined rule-based correlation process. These contextual factors were not learned or optimized from the evaluation dataset; rather, they were used to enrich and contextualize the evidence generated by the other gateway components.

The weights used in the experimental configuration were defined as $w_H = 0.15$, $w_D = 0.15$, $w_U = 0.20$, $w_A = 0.25$, $w_B = 0.15$, and $w_C = 0.10$, with their sum equal to 1.0. Attachment and URL risks were assigned relatively higher weights because they provide more direct evidence of malicious payload delivery and phishing activity. The weights were fixed before the experiments based on operational severity, evidence reliability, and preliminary observations, and were not optimized using the test dataset. Therefore, the proposed scoring model represents a deterministic and reproducible engineering mechanism rather than an adaptive or learning-based algorithm.

For example, assume that an incoming email produces the normalized scores $H = 0.7$, $D = 0.8$, $U = 0.9$, $A = 1.0$, $B = 0.6$, and $C = 0.9$. The overall threat score is calculated as follows:

$$S(E) = 100[0.15(0.7) + 0.15(0.8) + 0.20(0.9) + 0.25(1.0) + 0.15(0.6) + 0.10(0.9)] = 83.5 \quad (2)$$

According to the predefined thresholds, this score results in a Quarantine decision. However, a verified critical IoC, such as a confirmed ransomware file hash or phishing URL, triggers immediate blocking independently of the overall threat score.

4.3. Attack Lifecycle Modeling and Defense Analysis

Within the scope of this study, the progression of the cyber-attack was modeled to cover the processes from the threat actor's discovery of the target system to the final impact stage. The attack lifecycle was addressed within a phased structure in order to analyze how the attack evolves at different stages and at which points it can be prevented. This structure was divided into four main phases in a way that is consistent with the Cyber Kill Chain and the MITRE ATT&CK framework in the literature.

First, in the Reconnaissance phase, the preliminary process in which the threat actor analyzes the infrastructure of the target organization and prepares the phishing scenario is addressed. Then, the Initial Access stage represents the critical point where the malicious email is delivered to the target system and CTI-supported gateways aim to block the attack. Threats that cannot be detected at this stage proceed to the Execution and Lateral Movement phase, become active at the endpoint, and spread to other systems through vulnerabilities within the network.

In the final stage, the Impact phase, the attack reaches its objective, rendering corporate data inaccessible and directly disrupting operational continuity. Through this modeling approach, it becomes possible to comparatively evaluate at which stage of the attack chain the CTI-supported architecture and the standard security architecture are effective.

4.3.1. Pre-Attack (Reconnaissance)

In this stage, preliminary reconnaissance activities are carried out by the threat actor about the target organization. Using open-source information, social network data, or other online sources, the email addresses of target users and potential attack targets are identified.

4.3.2. Initial Access

The threat actor sends an email containing malicious content to the target user. The email may include malicious attachments or phishing links. Cyber threat intelligence applications can prevent malicious emails from reaching the system at this stage through analysis and reputation checks.

4.3.3. Execution and Lateral Movement

In systems with a standard security architecture, if the malicious email reaches the user, the endpoint system becomes infected with the execution of the malicious content. After the infection, the threat actor attempts to gain access to other systems within the network by using lateral movement techniques and expands the scope of the attack by exploiting existing security vulnerabilities.

4.3.4. Impact

In the final stage of the attack chain, ransomware encrypts the data on systems, rendering them inaccessible. As a result of this process, operational disruptions may occur in critical systems, and corporate data may come under the control of the threat actor.

4.4. Dataset Construction, Anonymization, and Test Scenarios

The dataset used in this study was constructed from security events recorded by SMG during real-world corporate email security operations between January 2025 and January 2026. The samples were obtained from legitimate and malicious email traffic observed across multiple organizations operating in different sectors. Due to confidentiality agreements and organizational information-security policies, the identities of the organizations were not disclosed. The dataset was structured as follows:

- Benign emails: 500;
- Emails containing malicious attachments: 200;
- Emails containing phishing links: 300.

The 200 emails containing malicious attachments represented the ransomware-related subset of the dataset. Benign emails included typical corporate communication patterns, such as order notifications, invoices, quotations, meeting messages, internal announcements, payment-related correspondence, and routine business communications. These emails had successfully passed the relevant security controls and contained no validated IoC or IoA evidence.

The phishing subset included credential-harvesting pages, spoofed sender identities, typo squatting domains, shortened or redirected URLs, cloud-storage-based delivery links, and invoice-, quotation-, or payment-themed social-engineering messages. Since the objective of the study was to evaluate gateway-level protection rather than compare individual phishing categories, these samples were collectively evaluated as phishing emails. Ransomware-related emails were identified using antivirus verdicts, sandbox execution behavior, SHA-256 hash matching, attachment analysis, and vendor- or CTI-based threat validation. Family-level ransomware labels were retained only when attribution was consistently confirmed by multiple reliable security sources. Samples without sufficiently reliable family-level attribution were conservatively classified as ransomware-related emails rather than being assigned to a specific ransomware family.

Before the experimental analysis, all organizational and personal identifiers that were not required for security evaluation were removed, masked, or pseudonymized. These included sender and recipient email addresses, usernames, organization names, internal domain names, internal IP addresses, file paths, telephone numbers, customer information, and other personally identifiable content. In contrast, security-relevant technical

attributes were preserved within the controlled environment, including email-header structure, SPF, DKIM, and DMARC results, URL characteristics, MIME types, attachment extensions, SHA-256 hash values, SMTP routing attributes, sandbox verdicts, and IoC/IoA-related labels.

The selection criteria were based on relevance to email-based Initial Access techniques, the presence of observable IoCs such as URLs, domains, IP addresses, or file hash values, compatibility with CTI validation, and safe processing in the VMware ESXi-based laboratory environment. Duplicate samples, corrupted attachments, incomplete headers, outdated indicators, and samples that could not be independently validated were excluded from the dataset.

Sample labels were not determined solely from SMG detection results. Email events were cross-validated using records obtained from network security gateways, firewalls, proxy systems, DNS logs, antivirus engines, sandbox analyses, and threat intelligence sources. Phishing samples were validated through URL and domain reputation analysis, SMTP-header examination, sender-impersonation indicators, and verification of fraudulent credential-harvesting pages. Ransomware-related samples were validated using antivirus detections, sandbox behavior, SHA-256 records, and threat intelligence evidence.

To avoid evaluation bias and data leakage, the phishing and ransomware-related samples were selected and anonymized independently from the IoC records used by the CTI-supported gateway. Their indicators were evaluated against CTI sources only during the gateway analysis process. The CTI feeds used during the experiments were treated as external intelligence sources rather than as the source from which the test emails were obtained. Therefore, the evaluation did not rely on pre-labeling emails using the same CTI indicators that were subsequently employed for detection. No test email was included or excluded based on whether its indicators were already present in the CTI feeds. CTI repositories were not modified, supplemented, or retrained using the labels or indicators of the evaluation samples.

During testing, malicious emails were evaluated at runtime through the same workflow applied to all incoming messages, including SPF, DKIM, and DMARC verification, IoC matching, URL analysis, attachment analysis, sandbox inspection, and IoA-based behavioral correlation. Benign and malicious emails were processed under the same gateway conditions, and the commercial baseline and CTI-supported systems were tested using identical email sets. Thus, the observed performance difference reflects the added contribution of external CTI enrichment and contextual IoC/IoA correlation rather than prior knowledge of the test samples.

In the experimental study, Kali Linux was used to represent the threat actor, and SMTP relay tools were employed to send the anonymized malicious email samples to the simulated Exchange Server infrastructure. Both high-volume burst transmissions and low-and-slow delivery patterns were used to represent different email attack behaviors. All malicious attachments and related activities were executed exclusively within the isolated VMware ESXi environment.

It is acknowledged that the dataset consisting of 1000 emails is limited in volume. However, targeted phishing and ransomware campaigns are frequently conducted through narrow-scope and strategic deliveries directed at selected users or organizations rather than through large-volume email batches. Therefore, the main objective of this study is not to train a statistical classification model on a large-scale dataset, but to demonstrate how CTI integration contributes to operational defense at the email gateway level and supports the interruption of the attack chain at an early stage.

The complete raw dataset cannot be publicly released because it was derived from real corporate email security operations and remains subject to confidentiality, privacy,

licensing, and security restrictions. Nevertheless, the manuscript provides the class distribution, collection period, represented threat categories, sample-selection and exclusion criteria, anonymization procedure, validation workflow, gateway configurations, and scoring parameters required to support partial methodological reproducibility. Accordingly, the findings should be interpreted as a controlled proof-of-concept evaluation of CTI integration under equivalent experimental conditions rather than as universally generalizable performance estimates for all enterprise email environments.

5. Experimental Results

In the simulation scenario, an order-confirmation-themed email designed to deceive the recipient was used. The file downloaded through the cloud-storage link embedded in the email contained malicious content. The performance of the CTI-supported email gateway architecture and commercial baseline configuration was compared under these attack scenarios using quantitative and qualitative metrics. The experiments were conducted in a controlled virtualization environment completely isolated from the external network. Sandbox-based isolation was applied in order to prevent malware scenarios from harming real systems [23].

The experimental environment was deployed within an isolated virtualized network, and all components were operated under controlled conditions. VMware ESXi 7.0 platform was used as the virtualization infrastructure. In order to control network communication between virtual machines and to isolate the experimental environment from external networks, an isolated vSwitch architecture was established. The Red Team machine representing the threat actor was configured on the Kali Linux 2024.1 operating system, and 4 vCPU and 8 GB RAM were allocated as system resources. In the execution of attack scenarios, the GoPhish v0.12.1 was used for managing phishing campaigns, while the Metasploit Framework v6.5.1 was used for vulnerability exploitation (exploit delivery) processes. Additionally, custom Python 3.11.7 scripts were developed for scenario automation and data processing. In order to simulate the email infrastructure, Microsoft Exchange Server was installed on Windows Server 2022 and configured to represent corporate email communication. The machines representing the target systems consist of end-user devices and servers running Windows 10 and Windows 11 operating systems. In order to create a realistic attack environment, security patches were not applied on some systems, allowing potential vulnerabilities to be observed in a controlled manner.

The sandbox analysis environment was implemented using Windows Sandbox within the isolated VMware ESXi 7.0 infrastructure and isolated vSwitch network. The host virtual machine was allocated 4 vCPUs and 8 GB RAM, and the sandbox was operated without direct access to external networks. During each analysis session, process creation, command-line activity, file-system and registry modifications, and network connection attempts were monitored and recorded. After each sample analysis, the Windows Sandbox session was closed and relaunched to provide a clean execution environment and prevent cross-sample contamination. GoPhish, Metasploit Framework, SMTP relay tools, and custom Python scripts were operated within the same isolated experimental infrastructure. GoPhish was used for phishing campaign management and email scheduling, Metasploit Framework supported controlled exploit-delivery scenarios, and the Python scripts were used for message transmission, scenario automation, indicator extraction, and log processing. Executable malicious payloads were not publicly distributed due to security restrictions.

5.1. Centralized Monitoring of Email Security Events and Automated Response System

Figure 5 presents the centralized management panel showing threat intelligence logs and content filtering results maintained on the email security system. In the table structure, each row represents a different email event, and time information, sender address, subject line, evaluation result (verdict), and applied actions are systematically listed.

Time (AST)	From	To	Original Subject	Verdict	Actions Taken
Wed, Feb 11, 2026 12:51:55 PM	aaliyeva@grantagro.az		Hello, material order...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Wed, Feb 11, 2026 12:48:18 PM	btc@benlimetal.com		Hello, parts file...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Wed, Feb 11, 2026 12:21:35 PM	info@sakaryamagazalari...		Hello, product list...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Wed, Feb 11, 2026 11:40:14 AM	support@savenily.com		Hello, material list...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Tue, Feb 10, 2026 04:03:45 PM	sercan@kuntotomasyon.com		Hello, material order...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Tue, Feb 10, 2026 03:47:57 PM	hi@lisanson.com		Hello, product list...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Tue, Feb 10, 2026 02:28:56 PM	info@gensismuhendislik...		Hello, product order...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Tue, Feb 10, 2026 02:26:18 PM	dsayid@maarifschools.ed...		Hello, parts order...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine
Tue, Feb 10, 2026 02:01:59 PM	aecne@iproject.com.pe		Hello, material order...	Content Filtering	• Send notification • Modify subject line • Add annotation • Spam Quarantine

All 9 messages quarantined via Content Filtering. Recipient addresses redacted for privacy.

Figure 5. Content filtering mechanism results for suspicious email traffic, email gateway quarantine log (10–11 February 2026).

In the examples shown in the figure, it is observed that emails identified as malicious are blocked by the “Content Filtering” mechanism, and accordingly, security actions such as sending notifications, modifying the subject line, adding annotations, and quarantining are automatically applied. This demonstrates that the CTI-supported system not only detects threats but also automates operational response processes. Additionally, the timestamps and process logs displayed in the panel indicate that threats are analyzed and blocked in real time. Such centralized monitoring and response mechanisms enable indicators such as malicious domain names, IP addresses, and file hash values to be continuously updated and integrated into security systems, allowing threats to be stopped before reaching the user.

5.2. Detection and Blocking of Malicious Email Content Before Reaching the User Mailbox

Mean Time to Detect (MTTD) in email security is one of the critical indicators that determine the impact of an attack. In the CTI-supported system, the sent email is stopped

at the gateway level and analyzed in real time with CTI feeds. The header information of the email, the reputation of the URLs it contains, and the hash values of attached files are matched with IoC and IoA records in up-to-date threat intelligence databases. As a result of the matching, emails with a high threat score are isolated at the network level before being delivered to the user and are directed to a secure quarantine area that can only be accessed by authorized security experts.

As shown in Figure 6, order-themed emails from different senders with the “[SPF Error]” label in the subject lines are listed in the “Spam Message Quarantine” section. The SPF validation error is an important authentication violation indicating that the sender domain may be spoofed.

Spam Message Quarantine (Active)

Search Filters
Use filters to search the Spam Quarantine for sidelined messages. In any field, you may use an asterisk (*) to perform a wildcard search.

To: From: Time range: Past day
Subject: merhabalar Message ID:

QUARANTINE ENTRIES

Displaying: 1–10 of 10 entries | Entries per page: 500

To	From	Subject	Date	Submission Status
[Redacted]	PAXIZE AKPINAR aaliyeva@grantaqro.az	[SPF Error] Merhabalar malzeme siparis listesini kontrol...	12:51:59 PM	Not submitted
[Redacted]	FATOS ASLANER btc@benlimetal.com	[SPF Error] Merhabalar parca dosyasi...	12:48:20 PM	Not submitted
[Redacted]	ALPARSLAN info@sakaryamagazalari.com	[SPF Error] Merhabalar urun listemiz...	12:21:37 PM	Not submitted
[Redacted]	TUFAN DEMIRTAS support@sevenlily.com	[SPF Error] Merhabalar malzeme liste...	11:40:17 AM	Not submitted
[Redacted]	EROL KARADAS sercan@kuntotomasyon.com	[SPF Error] Merhabalar malzeme siparis...	Feb. 10, 2026	Not submitted
[Redacted]	DERYA YALINKILIS hil@lisanson.com	[SPF Error] Merhabalar urun listemiz...	Feb. 10, 2026	Not submitted
[Redacted]	EBRU DEMIRTAS christian.rivera@ttindustrial.com	[SPF Error] Merhabalar urun dosyamiz...	Feb. 10, 2026	Not submitted
[Redacted]	KENAN chrisid@maarifschools.edu.krd	[SPF Error] Merhabalar parca siparis...	Feb. 10, 2026	Not submitted
[Redacted]	ESRA AKSOY aeche@iproject.com.pe	[SPF] Sender Verification Issue — Merhabalar...	Feb. 10, 2026	Not submitted
[Redacted]	DEFNE ALPARSLAN tz_tbillignstar.com	Merhabalar urun listesini kontrol eder...	Feb. 10, 2026	Not submitted

Figure 6. Gateway-level quarantine of suspicious emails before mailbox delivery.

In addition, the marking of messages as “Not submitted” in the “Submission Status” field shows that these emails were blocked at the gateway level without ever being delivered to the user mailbox. These emails, sent from different domains with similar content, indicate a distributed phishing campaign and demonstrate that the CTI-supported system can detect such coordinated threats at an early stage. Thus, the attack chain is interrupted at the “Initial Access” stage, significantly reducing the risk of user-driven erroneous clicks or malicious file downloads.

In the cyber threat intelligence-supported architecture, incoming emails are analyzed as soon as they reach the gateway based on sender IP address, URL content, domain reputation, and attachment indicators. Messages evaluated as having a high threat score are rejected or quarantined, thus preventing malicious content from reaching the user mailbox. In contrast, in systems without CTI integration, malicious emails can reach the user mailbox, and the detection process is often delayed until after the user clicks the related link or executes the malicious file. This delay allows the threat actor to achieve the “Initial Access” objective and enables the attack chain to progress to subsequent stages.

Figure 7 shows the Message Audit Logs screen maintained on the email security system. These logs reveal that emails reaching the system are analyzed in terms of

sender information, timestamp, authentication results (Sender Policy Framework—SPF, DomainKeys Identified Mail—DKIM, Domain-based Message Authentication, Reporting and Conformance—DMARC), and applied security policies. The log outputs also include details regarding message size, attachment information, and triggered filter or policy rules. Such audit logs provide an important data source for examining suspicious email activities, supporting incident response processes, and conducting forensic analysis of email-based attacks.

Message Audit Logs

MESSAGE DATA	
ID	0a621921-ab3517000001f037-26-696c5138b2c8
Message-ID	<20260211125154.qk3ezuvzq897737b67-dc43-416e-83ed-bab4ff856841>
Accepted From	178.██.██.██ (Logical IP: 178.██.██.██)
TLS Cipher	ecdhe-rsa-aes256-gcm-sha384
Scanners	SMG-03
Time Accepted	Wednesday, Feb 11, 2026 — 12:51:56 PM AST
Direction	Inbound
Sender	aaliyeva@grantaqro.az
Auth. Username	(none)
Original Subject	Hello, could you please check the material order list?
Message Size	1 KB (translated — original: “merhabalar malzeme siparisini check edebilirmisiniz”)
Identified Attach.	None
Suspect Attach.	None

AUTHENTICATION RESULTS	
DMARC	Details
SPF Result	softfail spf=softfail smtp.mailfrom=aaliyeva@grantaqro.az smtp.helo=forward400a.mail.yandex.net
DKIM	Details
SENDERID	Details

VERDICT & POLICY ACTIONS				
VERDICT	FILTER POLICY	POLICY GROUP	DETAILS	MSG PART
Content Filtering Violation — SPF Validation Softfail Modify subject & strip attachments	SPF validation softfail: modify subject line with "[SPF – Sender Verification Issue]" and strip specific attachments	com.tr	None	—
Content Filtering Violation	_routing_rule_08.01.2021	com.tr	None	envelope — recipient

Figure 7. Detailed message audit log for phishing message (ID: 0a621921...) intercepted on 11 February 2026. SPF returned softfail; two content filtering policies triggered: subject-line tagging and routing rule enforcement. Sender relayed via Yandex—consistent with spoofed-identity phishing TTPs.

5.3. Attack Blocking Success

In the experimental study conducted on the original dataset consisting of 1000 emails, the obtained results demonstrate that CTI integration provides a significant improvement in detecting malicious email content. As summarized in Table 6, the CTI-supported system

achieved an Accuracy of 98.6%, Precision of 98.4%, Recall (Detection Rate) of 98.8%, and an F1-score of 98.6%, whereas the commercial baseline system achieved 79.8% Accuracy, 93.3% Precision, 64.2% Recall, and 76.1% F1-score. These results indicate that integrating CTI into the email gateway substantially improves both detection capability and overall classification performance.

Table 6. Attack blocking performance of the models.

Metric	Commercial Baseline System	CTI-Supported System	Performance Difference
Accuracy	79.8%	98.6%	+18.8%
Precision	93.3%	98.4%	+5.1%
Recall (Detection Rate)	64.2%	98.8%	+34.6%
F1-score	76.1%	98.6%	+22.5%
Phishing URL Detection	58.0%	99.3%	+41.3%
Ransomware Attachment Detection	73.5%	98.0%	+24.5%
False Positive Rate (FPR)	4.5%	1.2%	−3.3%

In phishing URL detection, commercial baseline system achieved a detection rate of 58.0%, whereas the CTI-supported system reached 99.3%. This result demonstrates that URL reputation analysis, domain intelligence, and continuously updated CTI feeds play a critical role in identifying and blocking phishing links before they reach end users. Similarly, in detecting ransomware-related malicious attachments, the detection rate increased from 73.5% in commercial baseline system to 98.0% in the CTI-supported architecture. The CTI-supported system also outperformed the baseline architecture in terms of false positive reduction. The False Positive Rate (FPR) decreased from 4.5% to 1.2%, indicating that CTI integration not only improves the detection of malicious emails but also significantly reduces the probability of legitimate emails being incorrectly classified as malicious. Consequently, operational overhead caused by unnecessary quarantine actions is minimized while maintaining a high level of detection accuracy.

In commercial baseline system, false positives mainly occurred because legitimate business emails containing keywords such as “invoice”, “payment”, “quotation”, or “order” were incorrectly classified as suspicious by static spam and keyword-based filtering rules. In addition, some legitimate emails sent from newly registered domains or external suppliers were incorrectly flagged due to limited sender reputation information. Since commercial baseline system did not correlate these indicators with contextual CTI data, sender history, domain reputation, or authentication results, weak indicators could incorrectly trigger quarantine decisions. False negatives in commercial baseline system were primarily caused by dynamic phishing URLs, shortened links, newly registered domains, and ransomware attachments whose hash values were modified through packing or padding techniques. Although the commercial baseline incorporated vendor-native reputation, authentication, antivirus, and filtering capabilities, it did not correlate these outputs with external IoCs, IoA patterns, and contextual CTI evidence within a unified decision mechanism.

In the CTI-supported architecture, false positives were reduced because suspicious indicators were not evaluated in isolation. Instead, SPF, DKIM, and DMARC results, sender IP/domain reputation, URL intelligence, attachment analysis, sandbox outputs, and IoA-based behavioral indicators were correlated before the final decision was made. For example, a newly registered domain did not automatically lead to blocking unless it was also associated with additional risk indicators such as DMARC failure, typosquatting, suspicious redirects, or malicious attachment behavior. This contextual correlation reduced unnecessary quarantine actions for legitimate emails. False negatives in the CTI-supported system were mostly associated with indicators that had not yet appeared in CTI feeds, very recent phishing domains with insufficient reputation history, and payloads that did not

exhibit malicious behavior during sandbox execution. However, CTI integration reduced the impact of such cases by using multiple evidence sources rather than relying only on static signatures.

5.4. Mapping of Attacks with MITRE ATT&CK Tactics and Techniques

The defense mechanisms were evaluated not only in terms of overall detection performance but also according to their position within the attack lifecycle. The implemented attack scenarios were mapped to the MITRE ATT&CK framework to determine the stages at which the commercial baseline configuration and the CTI-supported architecture responded to the modeled attacks.

Figure 8 comparatively presents the stage-based defense outcomes of the two architectures. In the evaluated scenario, the CTI-supported system detected and blocked the tested malicious email at the Initial Access stage, represented by T1566.001 (Spearphishing Attachment) and T1566.002 (Spearphishing Link). Consequently, the subsequent Execution, Command and Control, and Impact stages were not reached in this scenario. In contrast, the tested malicious email was not detected at the Initial Access stage by the commercial baseline configuration, allowing the modeled attack to progress to Execution (T1204.002—User Execution), Command and Control (T1071.001—Web Protocols), and Impact (T1486—Data Encrypted for Impact). Figure 8 therefore distinguishes between threats that were blocked, attacks that progressed, and stages that were not reached. The findings indicate that, under the evaluated scenarios, the CTI-supported architecture improved early-stage threat detection and interrupted the observed attack-chain progression before subsequent attack phases were reached.

In the second phase of the simulation, the commercial baseline configuration did not stop the tested attack at the Initial Access stage, and the modeled attack subsequently progressed through later phases. The cloud-based file-sharing link contained in the email reached the user's mailbox, and the malicious content was executed through user interaction. Following the Execution stage, the threat actor initiated internal network reconnaissance and lateral movement activities through the compromised endpoint. During the Lateral Movement stage, access to file servers was achieved by exploiting vulnerabilities in unpatched SMB services (T1021—Remote Services). Previous studies have also reported that systems operating without CTI feeds may have limited ability to identify such internal network activity as anomalous and to intervene in the progression of multi-stage attacks [9].

In the final Impact stage, ransomware encrypted data on the server systems and generated ransom notes. Thus, within this controlled scenario, the modeled attack progressed beyond the endpoint and affected additional systems in the corporate test environment. These findings suggest that vendor-native security controls alone may be insufficient against some dynamic and multi-stage email-based attacks. In contrast, the CTI-supported configuration provided an additional contextual defense layer that, in the evaluated scenario, interrupted the attack chain at the Initial Access stage [25].

MITRE ATT&CK stage-based defense outcomes		
Attack stage / technique	Architecture	
	Commercial baseline system	CTI-supported system
	Initial Access T1566.001	Blocked
	Initial Access T1566.002	Blocked
	Execution T1204.002	Not reached
	Command and Control T1071.001	Not reached
	Impact T1486	Not reached

■ Blocked
 ■ Not detected / progressed
 ■ Not reached

Figure 8. MITRE ATT&CK stage-based defense outcomes for the commercial baseline and CTI-supported systems.

6. Discussion

The simulation results were evaluated in terms of attack blocking success, the operational impact of the false positive rate, business continuity, and cost–benefit balance. The findings show that email security architecture using CTI provides a more adaptive and contextual defense model compared to traditional static control mechanisms.

6.1. Comparison of Attack Blocking Success Rates and False Positive/False Negative Analysis

The simulation results showed that the CTI-supported system reached a total detection rate of 98.8%. Commercial baseline system operating under the same scenarios remained at a detection rate of 64.2%. The observed 34.6% performance difference is primarily attributed to the additional external CTI enrichment and unified contextual correlation mechanisms available in the CTI-supported configuration. While the commercial baseline already incorporates vendor-native capabilities such as antivirus scanning, URL reputation, authentication controls, and threat-blocking policies, it does not integrate external IoCs from MISP and OpenCTI or correlate them with IoA-based behavioral evidence and contextual threat information within a unified decision mechanism.

The ransomware samples used in the tests were structurally altered in each transmission by adding meaningless data (junk data/padding) to the structure without disrupting the main functionality of the malicious code or by modifying packer routines. Since even a small modification to a file result in a different SHA-256 hash value, some modified samples were not identified by hash-based or signature-oriented controls available in the commercial baseline when no corresponding known indicator was available. This situation demonstrates the limitation of hash-based detection mechanisms, especially against polymorphic or previously unsigned malicious content.

The architecture using threat intelligence applied a contextual decision mechanism beyond static signature matching. Even if the file signature was unknown, the command-

and-control (C2) infrastructure that the malware attempted to communicate with was analyzed, and IP addresses with poor reputation in global threat databases were detected and blocked at the connection stage. This approach contributed to interrupting the attack chain at an early stage.

A significant difference was also observed between the two architectures in terms of false positive rates. Commercial baseline system incorrectly quarantined some legitimate business emails containing expressions such as “invoice” or “quotation” and produced a 4.5% false positive rate. High false positive rates can cause alert fatigue in security teams and increase the risk of overlooking real threats.

In contrast, the CTI-supported system more accurately distinguished between legitimate senders and spoofed domains by jointly analyzing the age of the sender domain, historical behavior records, IP/domain reputation, and threat intelligence data. As a result, the false positive rate decreased to 1.2%, and the operational workload was reduced.

The performance improvement achieved through CTI integration can be attributed to the use of external and continuously updated threat intelligence together with behavioral and contextual evidence. While the commercial baseline relies on vendor-native antivirus, reputation, authentication, and filtering mechanisms, the CTI-supported architecture additionally correlates current IoCs obtained through OpenCTI and MISP with IoA-based behavioral indicators. Consequently, malicious URLs, domains, IP addresses, or file hashes that may not generate sufficient evidence through the native gateway controls alone can be identified when additional CTI and behavioral evidence is available. This multi-source correlation makes the decision process more contextual and explains the higher detection performance observed in the experiments.

Nevertheless, several failure conditions remained. Attacks involving newly registered domains, malicious URLs not yet represented in external intelligence sources, or previously unseen file hashes may not be directly identified through IoC-based matching. In such cases, IoA-based behavioral analysis provides an additional layer of evidence; however, completely new or zero-day threats may still remain undetected when both intelligence and behavioral evidence are insufficient. These cases demonstrate that CTI enrichment reduces dependence on known indicators but does not eliminate detection failures.

False-positive cases were mainly associated with messages involving recently registered but legitimate domains, URL-shortening services, and trusted file-sharing platforms. Such messages may receive elevated risk scores when sufficient reputation history is unavailable or when their characteristics partially resemble malicious activity. However, the multi-criteria decision mechanism combines this evidence with SPF, DKIM, and DMARC results and behavioral indicators, which contributed to reducing the false positive rate. Overall, the findings indicate that the main benefit of CTI integration is not only improved detection performance but also a more contextual decision process based on multiple complementary sources of evidence.

6.2. The Impact of Intelligence-Based Defense on Organizational Continuity

In the second scenario, the commercial baseline system could not stop the ransomware before it reached the end-user device. After the execution of the malicious content, the file encryption process started, and disruptions occurred in some business processes. Meland et al. [29] emphasize that the primary cost in ransomware attacks arises not only from data recovery processes but also from service disruption and business loss.

In contrast, in the CTI-supported architecture, the attack was stopped at the gateway level, and the malicious link was blocked before reaching the user’s mailbox. Thus, in the evaluated scenarios, the attack chain was interrupted at the Initial Access stage, and no malicious-code execution or resulting data loss was observed within the internal network.

When evaluated in terms of organizational continuity, preventing the attack at an early stage ensured that systems maintained their normal operational state and significantly reduced the risk of operational disruption.

6.3. The Role of Early Detection in Preventing Lateral Movement

One of the highest risks in the attack chain is the spread of the threat within the internal network after the initial infection. After compromising the first system, the threat actor seeks access to other systems, and this process is defined as Lateral Movement in the MITRE ATT&CK model. In commercial baseline architecture, since the initial infection could not be detected at an early stage, the threat actor attempted to access other systems through privilege escalation and internal network reconnaissance activities. During this process, access to the file server was achieved and corporate data were encrypted. In the CTI-supported system, however, the attack was blocked before reaching this stage. Since the phishing email did not reach the user and the malware could not be executed, the initial infection did not occur, and accordingly, no internal network propagation was observed during the evaluated experiments. This finding demonstrates that early detection is a critical defense point in preventing lateral movement.

6.4. Cost and Benefit Evaluation in Terms of Security Investment

The results obtained show that CTI integration should be evaluated not only from a technical perspective but also from an economic perspective. Although threat intelligence integration requires certain infrastructure, licensing, and subscription costs, these costs are predictable and plannable. In contrast, the cost of a successful ransomware attack has a much broader impact, including data recovery, system reinstallation, potential ransom payments, service disruption, legal penalties, and operational losses. Regulations such as KVKK and GDPR may impose additional notification, compliance, and sanction obligations on organizations in the event of a data breach. In addition, loss of corporate reputation is an important risk factor that is often not directly measurable but has long-term effects.

Simulation results show that even preventing a single successful attack at an early stage can offset the security investment made for CTI integration in a short time. Therefore, threat intelligence integration should be evaluated not only as an additional security feature, but also as a preventive and strategic investment in terms of corporate risk management.

Although the experimental results demonstrate the operational benefits of CTI integration, their direct transferability to real corporate environments should be interpreted with caution. Production email infrastructures involve substantially higher traffic volumes, heterogeneous user behaviors, organization-specific security policies, diverse gateway configurations, and continuously changing threat intelligence sources. These factors may affect detection performance, processing latency, and false positive rates. Therefore, the results primarily demonstrate the comparative value of CTI integration under controlled and equivalent conditions, while large-scale deployment studies are required to confirm performance across different organizational environments.

The reported performance metrics are based on a single controlled evaluation using the same set of 1000 emails for both gateway configurations. Repeated experimental runs, confidence intervals, and statistical significance tests were not conducted in the present study. Therefore, the reported percentages should be interpreted as point estimates under the specified experimental conditions rather than as population-level performance estimates.

6.5. Operational Challenges and Deployment Considerations

Although the CTI-supported architecture demonstrated improved detection performance in the controlled experimental environment, its deployment in real-world enterprise

infrastructures introduces several operational considerations, particularly CTI feed quality, maintenance requirements, scalability, and infrastructure capacity. The effectiveness of CTI-based detection depends strongly on the accuracy, freshness, reliability, and relevance of the intelligence sources. Outdated or low-confidence indicators may increase false-positive decisions, whereas newly emerging indicators that have not yet been incorporated into intelligence repositories may contribute to false negatives. Therefore, CTI sources should be periodically updated, indicator confidence and source reliability should be considered during correlation, and obsolete indicators should be appropriately managed.

CTI integration also introduces continuous operational requirements beyond the initial deployment. Threat feeds, IoC repositories, connectors, APIs, and correlation services require monitoring and maintenance. Centralized platforms such as OpenCTI and MISP can simplify intelligence aggregation and management; however, organizations must still consider the administrative effort and infrastructure costs associated with maintaining these services. Scalability is another important consideration in high-volume enterprise email environments. Increasing message volume and the number of CTI queries may increase processing and resource requirements. Techniques such as indicator caching, asynchronous processing, load balancing, and distributed CTI services may therefore be required in large-scale deployments. These mechanisms were not independently benchmarked in the present study and should be evaluated under production-scale workloads.

7. Conclusions

This study experimentally examined the effect of integrating CTI methodologies into corporate email gateway architectures on the detection and prevention of email-based cyber-attacks. In simulations carried out in an isolated virtualization environment, the CTI-supported architecture and a commercial baseline configuration operating with vendor-native security controls but without external CTI enrichment and unified IoC/IoA correlation were compared under the same attack scenarios.

The experimental results showed that the CTI-supported system reached 98.8% in the overall detection rate, while the commercial baseline system remained at 64.2%. The success rate increased from 58.0% to 99.3% in phishing URL detection, and from 73.5% to 98.0% in ransomware attachment detection. In addition, the false positive rate decreased from 4.5% to 1.2%. These results show that CTI integration provides higher detection success and a lower false alarm rate, especially against polymorphic malware and dynamic threats.

In the evaluated scenarios, the CTI-supported architecture interrupted the observed attack chain at the Initial Access stage, blocked most malicious content before delivery, and reduced the risk of internal network propagation. The obtained results reveal that CTI integration is not only a technical security feature, but also a strategic defense mechanism in terms of corporate risk management and business continuity.

The findings of this study suggest that CTI-based architectures can improve early-stage detection by jointly evaluating IP reputation, domain history, IoC indicators, and behavioral analysis data. In the evaluated scenarios, this approach contributed to reducing the risk of malicious emails reaching user mailboxes and may help reduce data loss, operational disruption, and financial losses associated with ransomware attacks.

Considering the increasing complexity of corporate email threat environments, the findings suggest that CTI integration can play an important role in modern email security architectures. However, further validation in larger-scale production environments is required before broader operational conclusions can be drawn.

Future work will test the proposed architecture on larger datasets containing different threat categories, validate it in different corporate network topologies, evaluate its per-

formance in real-time production environments, and conduct repeated experiments with confidence intervals and statistical significance analysis.

Author Contributions: Conceptualization, M.K. and H.B.; methodology, M.K. and H.B.; software, M.K.; validation, M.K., H.B. and H.H.B.; formal analysis, M.K. and H.B.; investigation, M.K. and H.B.; resources, M.K.; data curation, M.K.; writing—original draft preparation, M.K. and H.B.; writing—review and editing, M.K., H.B. and H.H.B.; visualization, M.K.; supervision, M.K. and H.H.B.; project administration, M.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All datasets used are available online with open access.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ildırım, K.Y.; Demir, M.E.; Keles, T.; Yıldız, A.M.; Dogan, S.; Tuncer, T. A YARA-Based Approach for Detecting Cyber Security Attack Types. *Firat Univ. J. Exp. Comput. Eng.* **2023**, *2*, 55–68. [\[CrossRef\]](#)
2. Ye, Y.; Li, T.; Adjeroh, D.; Iyengar, S.S. A Survey on Malware Detection Using Data Mining Techniques. *ACM Comput. Surv.* **2017**, *50*, 1–40. [\[CrossRef\]](#)
3. Mirtsch, M.; Pohlisch, J.; Blind, K. Certification as a Compensation Mechanism for Weak Regulation? Exploring the Diffusion of the International Standard ISO/IEC 27001 for Information Security Management. *Comput. Secur.* **2026**, *162*, 104774. [\[CrossRef\]](#)
4. Ainslie, S.; Thompson, D.; Maynard, S.; Ahmad, A. Cyber-Threat Intelligence for Security Decision-Making: A Review and Research Agenda for Practice. *Comput. Secur.* **2023**, *132*, 103352. [\[CrossRef\]](#)
5. Santos, P.; Abreu, R.; Reis, M.J.C.S.; Serôdio, C.; Branco, F. A Systematic Review of Cyber Threat Intelligence: The Effectiveness of Technologies, Strategies, and Collaborations in Combating Modern Threats. *Sensors* **2025**, *25*, 4272. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Doshi, J.; Parmar, K.; Sanghavi, R.; Shekhar, N. A Comprehensive Dual-Layer Architecture for Phishing and Spam Email Detection. *Comput. Secur.* **2023**, *133*, 103378. [\[CrossRef\]](#)
7. Almutairi, A.; Kang, B.J.; Alhashimy, N. Business Email Compromise: A Systematic Review of Understanding, Detection, and Challenges. *Comput. Secur.* **2025**, *158*, 104630. [\[CrossRef\]](#)
8. Heiding, F.; Schneier, B.; Vishwanath, A.; Bernstein, J.; Park, P.S. Devising and Detecting Phishing Emails Using Large Language Models. *IEEE Access* **2024**, *12*, 42131–42146. [\[CrossRef\]](#)
9. Yazdanjue, N.; Rakhshaninejad, M.; Yazdanjoui, H.; Niemelä, M.S.; Chen, F.; Gandomi, A.H. Cyber Threat Management Using Semi-Supervised Ensemble Learning and Enhanced Interior Search Algorithm: Applications for Illicit Marketplace Classification in Deep/Dark Web and Social Platforms. *Ann. Oper. Res.* **2025**. [\[CrossRef\]](#)
10. Sharabov, M.; Tsochev, G.; Gancheva, V.; Tasheva, A. Filtering and Detection of Real-Time Spam Mail Based on a Bayesian Approach in University Networks. *Electronics* **2024**, *13*, 374. [\[CrossRef\]](#)
11. Prümmer, J.; van Steen, T.; van den Berg, B. A Systematic Review of Current Cybersecurity Training Methods. *Comput. Secur.* **2024**, *136*, 103585. [\[CrossRef\]](#)
12. Alhogail, A.; Alsabih, A. Evaluating Spam Filters and Stylometric Detection of AI-Generated Phishing Emails. *Expert Syst. Appl.* **2025**, *276*, 127044. [\[CrossRef\]](#)
13. Mauri, L.; Damiani, E. Hardening Behavioral Classifiers against Polymorphic Malware: An Ensemble Approach Based on Minority Report. *Inf. Sci.* **2025**, *689*, 121499. [\[CrossRef\]](#)
14. Alani, M.M.; Mashatan, A.; Miri, A. XMal: A Lightweight Memory-Based Explainable Obfuscated-Malware Detector. *Comput. Secur.* **2023**, *133*, 103409. [\[CrossRef\]](#)
15. Brezinski, K.; Ferens, K. Metamorphic Malware and Obfuscation: A Survey of Techniques, Variants, and Generation Kits. *Secur. Commun. Netw.* **2023**, *2023*, 8227751. [\[CrossRef\]](#)
16. Bari, N.; Saleem, T.; Shah, M.; Algarni, A.; Patel, A.; Ullah, I. A Filter-Based Feature Selection Framework to Detect Phishing URLs Using Stacking Ensemble Machine Learning. *Comput. Model. Eng. Sci.* **2025**, *145*, 1167–1187. [\[CrossRef\]](#)
17. Eze, C.S.; Shamir, L. Analysis and Prevention of AI-Based Phishing Email Attacks. *Electronics* **2024**, *13*, 1839. [\[CrossRef\]](#)
18. Abraham, C.; Bélanger, F.; Daultrey, S. Promoting Research on Cyber Threat Intelligence Sharing in Ecosystems. *J. Cybersecur.* **2025**, *11*, tyaf016. [\[CrossRef\]](#)
19. Korkmaz, Y. Fraud E-Mail Detection Using Intelligent Algorithms: Comparison of Traditional Approaches with Deep Learning Techniques. *Inf. Process. Manag.* **2026**, *63*, 104416. [\[CrossRef\]](#)
20. Alhogail, A.; Alsabih, A. Applying Machine Learning and Natural Language Processing to Detect Phishing Email. *Comput. Secur.* **2021**, *110*, 102414. [\[CrossRef\]](#)

21. Sahingoz, O.K.; Buber, E.; Demir, O.; Diri, B. Machine Learning Based Phishing Detection from URLs. *Expert Syst. Appl.* **2019**, *117*, 345–357. [[CrossRef](#)]
22. Connolly, A.; Atlam, H.F. Effective Ensemble Learning Phishing Detection System Using Hybrid Feature Selection. *J. Netw. Comput. Appl.* **2025**, *242*, 104251. [[CrossRef](#)]
23. Aljabri, M.; Alhaidari, F.; Albuainain, A.; Alrashidi, S.; Alansari, J.; Alqahtani, W.; Alshaya, J. Ransomware Detection Based on Machine Learning Using Memory Features. *Egypt. Inform. J.* **2024**, *25*, 100445. [[CrossRef](#)]
24. Jáñez-Martino, F.; Alaiz-Rodríguez, R.; González-Castro, V.; Fidalgo, E.; Alegre, E. Classifying Spam Emails Using Agglomerative Hierarchical Clustering and a Topic-Based Approach. *Appl. Soft Comput.* **2023**, *139*, 110226. [[CrossRef](#)]
25. Tounsi, W.; Rais, H. A Survey on Technical Threat Intelligence in the Age of Sophisticated Cyber Attacks. *Comput. Secur.* **2018**, *72*, 212–233. [[CrossRef](#)]
26. Wagner, T.D.; Mahbub, K.; Palomar, E.; Abdallah, A.E. Cyber Threat Intelligence Sharing: Survey and Research Directions. *Comput. Secur.* **2019**, *87*, 101589. [[CrossRef](#)]
27. Zhao, J.; Yan, Q.; Li, J.; Shao, M.; He, Z.; Li, B. TIMiner: Automatically Extracting and Analyzing Categorized Cyber Threat Intelligence from Social Data. *Comput. Secur.* **2020**, *95*, 101867. [[CrossRef](#)]
28. Naseer, I. Machine Learning Applications in Cyber Threat Intelligence: A Comprehensive Review. *Asian Bull. Big Data Manag.* **2024**, *3*, 190–200. [[CrossRef](#)]
29. Meland, P.H.; Bayoumy, Y.F.F.; Sindre, G. The Ransomware-as-a-Service Economy within the Darknet. *Comput. Secur.* **2020**, *92*, 101762. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.