Naif Arab University for Security Sciences
Journal of Information Security & Cybercrimes Research
مجلة بحوث أمن المعلومات والجرائم السيبرانية
https://journals.nauss.edu.sa/index.php/JISCR

JISCR

CrowdStrike Causes Global Microsoft Outage: A Case Study

Reef E. Alsowaigh
Independent Researcher, Saudi Arabia, Dammam

Received 15 Oct. 2024; Accepted 10 Jun. 2025; Available Online 26 Jun. 2025

Abstract

In today's world, reliance on technology is rapidly growing across critical sectors such as business, banking, healthcare, and education. While technology enhances convenience and efficiency in daily activities, its failure can lead to significant disruptions. A notable global incident caused by a fault in CrowdStrike software disrupted the availability aspect of the Confidentiality, Integrity, and Availability (CIA) triad in cybersecurity, impacting Microsoft Windows users. The issue stemmed from the Falcon sensor, a faulty update that triggered the Blue Screen of Death (BSOD) due to a mismatch in parameters within the sensor code and the Inter-Process Communication (IPC) Template Type. To resolve the problem, CrowdStrike implemented runtime array bounds checks in the Content Interpreter function and validated input parameters to ensure system stability. These corrective measures aimed to prevent similar incidents and restore normal functionality for affected users. This paper introduces a case study that provides an overview of CrowdStrike, examines the incident in detail, identifies the root cause, outlines the remediation techniques employed, and highlights key lessons learned. It emphasizes the importance of effective incident response strategies and the use of canary testing to mitigate the impact of future technological failures.

I. Introduction

Technology has become an integral part of our daily lives, fundamentally transforming how we connect, work, and live. Global communication, online shopping, and making reservations have become easier and faster because of technology. Recently, in July 2024, a global technology outage impacted approximately 8.5 million Windows devices, disrupting critical services across multiple industries. This widespread disruption was caused by a fault in the enterprise software provided by CrowdStrike. CrowdStrike is a third-party provider of cybersecurity services, offering software solutions designed to protect its clients from cyber threats. CrowdStrike provides services such as threat intelligence, endpoint protection, and incident response to ensure the security and reliability of its clients' systems worldwide. One of

Keywords Blue screen of death (BSOD), CrowdStrike, cyber security, Falcon sensor, incident response plan (IRP)

Production and hosting by NAUSS

*Corresponding author: Reef E. Alsowaigh
reef_alsowaigh@hotmail.com

doi: 10.26735/QHDD4798

its key products, Falcon software, is widely used by organizations to secure their systems against cyberattacks [1]. This incident underscores the critical importance of securing the software supply chain, as any vulnerabilities or failures can expose organizations to significant risks [2]. Despite these assurances, a faulty software update in the CrowdStrike Falcon sensor led to an outage that triggered the Blue Screen of Death (BSOD) for Microsoft Windows users [3]. The BSOD is a critical error screen that appears when Windows encounters a severe system failure, as shown in Fig 1. This incident highlights the inherent risks in the software supply chain and the potential for disruptions to compromise system reliability. In cybersecurity, the CIA triad: Confidentiality, Integrity, and Availability, provides a comprehensive framework for assessing and managing security risks. These three principles are essential for minimizing risks to data, maintaining trust, and ensuring operational resilience. However, the increasing reliance on technology introduces significant challenges, particularly in maintaining the availability of systems and data. Availability refers to the ability to provide timely and reliable access to systems and data whenever required. The CrowdStrike incident compromised the availability of the CIA triad, causing a widespread outage that disrupted system access across numerous industries. The impact of this outage was far-reaching, affecting sectors such as airlines, airports, hotels, banks, manufacturing, healthcare, gas stations, the stock market, and many others [1]. Despite the severity of the incident, there is a notable lack of case studies analyzing the CrowdStrike outage, leaving many questions unanswered and limiting deeper insights into the root cause and implications of the event. This gap highlights the need for detailed analysis and research to address these questions. The study's primary contributions aim to fill that gap by providing a thorough examination of the incident's root cause, comparing it to similar cases in software supply chain security, and proposing effective solutions to prevent such occurrences in the future. By addressing these aspects, this study contributes to a better understanding of the risks associated with software supply chain vulnerabilities and offers actionable recommendations to enhance cybersecurity resilience. This paper is organized into seven key sections to provide a comprehensive analysis of the CrowdStrike incident and its broader implications. The introduction outlines the incident within the context of cybersecurity, emphasizing the importance of securing the software supply chain. The Background section explains the role of the Falcon sensor and its operation within operating system architecture. The Analysis & Discussion section delves into technical details, the root cause of the incident, findings, phishing campaigns, comparisons with similar incidents, and financial implications. The Solutions & Remediation section highlights the importance of software deployment testing and details various techniques used in the process. The Methodology section describes how the study was conducted to assess its validity and reliability. The Lessons Learned & Recommendations section identifies the key lessons learned from the CrowdStrike incident and provides actionable recommendations for organizations. Finally, the conclusion summarizes the key findings and proposes preventive measures to mitigate future incidents.

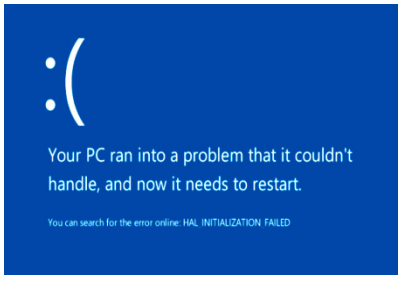

Fig. 1. Blue Screen Of Death (BSOD)[1]

II. Background

The Falcon sensor is an Endpoint Detection and Response (EDR) software installed on client devices operating at the kernel driver level to provide cyberattack response services and threat intelligence, ensuring the protection of endpoint users. The Falcon sensor monitors critical system activities such as process and thread creation, as well as file operations like saving, deleting, and modifying. By observing these activities, it can detect and block suspicious actions, offering robust protection against potential threats. In

the Windows operating system, hardware devices and software applications communicate through drivers, categorized into two types: kernel mode and user mode. Applications in kernel mode have the highest level of access and can directly interact with critical system resources. However, if a kernel-mode application encounters an error, it can crash the entire operating system. In contrast, user-mode applications operate with restricted access and must make system calls to interact with system resources, limiting the impact of crashes to only the specific application involved [4][5]. The C:/Windows/System32/drivers directory contains essential system files, including the Falcon sensor, which operates in kernel mode and has deep access to system resources [6]. To address emerging threats, CrowdStrike regularly releases patches for the Falcon sensor. However, a faulty software update caused a system crash and a BSOD due to its kernel integration. Fig 2 illustrates applications like Google Chrome, PDF Reader, Microsoft PowerPoint, and Microsoft Word functioning in user mode. These applications must request access to system resources from the kernel. For example, when saving a Word document, the application sends a system call to the kernel for permission to write to the disk. In contrast, the Falcon sensor operates in kernel mode, allowing direct access to system resources without needing to make such requests.

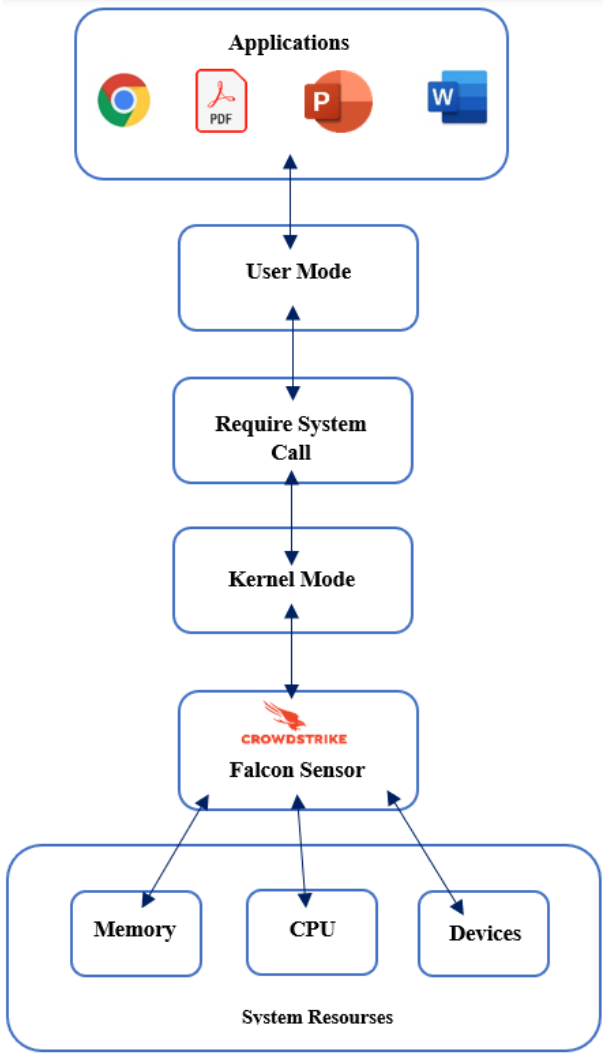

Fig. 2. User Mode and Kernel Mode

III. Analysis & Discussion

As part of the analysis of the CrowdStrike incident, this section provides a comprehensive examination of the technical aspects. It includes detailed explanations of technical terms, an exploration of the root cause of the incident, clarification of key findings, and a demonstration of how attackers activated phishing campaigns. Additionally, it offers a comparison with similar occurrences and discusses the resulting downtime and financial implications.

A. Technical Terms

Several technical terms must be defined to understand the CrowdStrike incident fully, such as Falcon Sensor, Channel File, Rapid Response Content, Content Interpreter, Template Type, Template Instance, Inter-Process Communication (IPC), IPC Template Type, and Named Pipes.

1) Falcon Sensor

The Falcon sensor is software executed locally on the client's device. It monitors user activities and takes preventive measures to block malicious actions. There is constant communication between the Falcon sensor and the CrowdStrike cloud. The Falcon sensor sends **telemetry** to the CrowdStrike cloud, which consists of data collected from client devices to analyze user system activities and determine whether they are legitimate or suspicious. In contrast, the CrowdStrike cloud sends **content** to the Falcon sensor and provides updates on the

latest threats, which can assist in detecting and responding to new threats on client devices [7][8].

2) Channel File

The Channel File is an instruction manual containing configuration information, including allowlists and blocklists, to guide the sensor's proper operation. This file is stored on the client device and can be updated dynamically without user intervention. The CrowdStrike Cloud periodically sends content updates through administrators based on threat intelligence and changes in user behavior [7].

3) Rapid Response Content

The Rapid Response Content is used to gather telemetry from client devices. CrowdStrike sends channel file updates through this rapid response content, which is designed to quickly enhance the security system [7].

4) Content Interpreter

The Content Interpreter is a component of the Falcon sensor that interprets and translates the channel file content from Rapid Response Content. It functions similarly to a compiler in coding [7].

5) Template Type

The Template types, like blueprints or pre-defined forms, are written in code to respond to threat behaviors. To simplify, each channel file is associated with a specific template type [7].

6) Template Instance

Each Template Instance is associated with a specific template type, which is a sequence of instructions for the sensor to recognize a specific threat behavior, enabling it to detect and prevent that behavior [7].

7) Inter-Process Communication (IPC)

Inter-process communication refers to the methods and mechanisms provided by an operating system that enable multiple processes to communicate and share data and memory [7].

8) IPC Template Type

The IPC template type is a pre-defined form used for inter-process communication to detect and identify threats. It monitors unauthorized or unusual communications between different processes [7].

9) Named Pipes

The Named pipes are an example of IPC that allows different software applications to communicate within the same computer [7].

B. Technical Analysis

Many organizations rely on third-party security services, such as CrowdStrike, to safeguard their digital infrastructure. The Falcon sensor operates in kernel mode, granting deep access to system resources on the client device to protect against malicious activities and threats. The CrowdStrike Cloud dynamically sends updates to the Falcon sensor without user intervention. However, a faulty update impacted the kernel level, which is the core of the Windows operating system. The Falcon sensor is interconnected with the sensor content found in Rapid Response Content, which collects telemetry data and identifies indicators of threat behavior. This integration enhances the system's ability to detect and respond to potential threats effectively. Channel Files are delivered to the sensor through Rapid Response Content and interpreted by the Content Interpreter. Each Channel File is associated with specific Template Types. CrowdStrike released a new configuration update for Channel File 291 and established a new Template Type, the IPC Template Type, to detect malicious misuse of Named Pipes. In this IPC Template Type, 21 input parameters are defined; however, when the Content Interpreter processes the Channel File 291 Template Instance content, the integration code invokes only 20 inputs. This mismatch in parameters caused an issue when the Content Interpreter attempted to access the 21st value, resulting in an out-of-bounds memory read issue and leading to a BSOD (Blue Screen of Death) loop, an endless crash cycle that caused a global outage affecting Windows 10, Windows 11, and various Windows Server versions. Since CrowdStrike enabled automatic updates without gradual rollout, all systems were affected simultaneously [9][10]. Fig 3 illustrates the architecture of the Falcon sensor and summarizes the CrowdStrike incident.

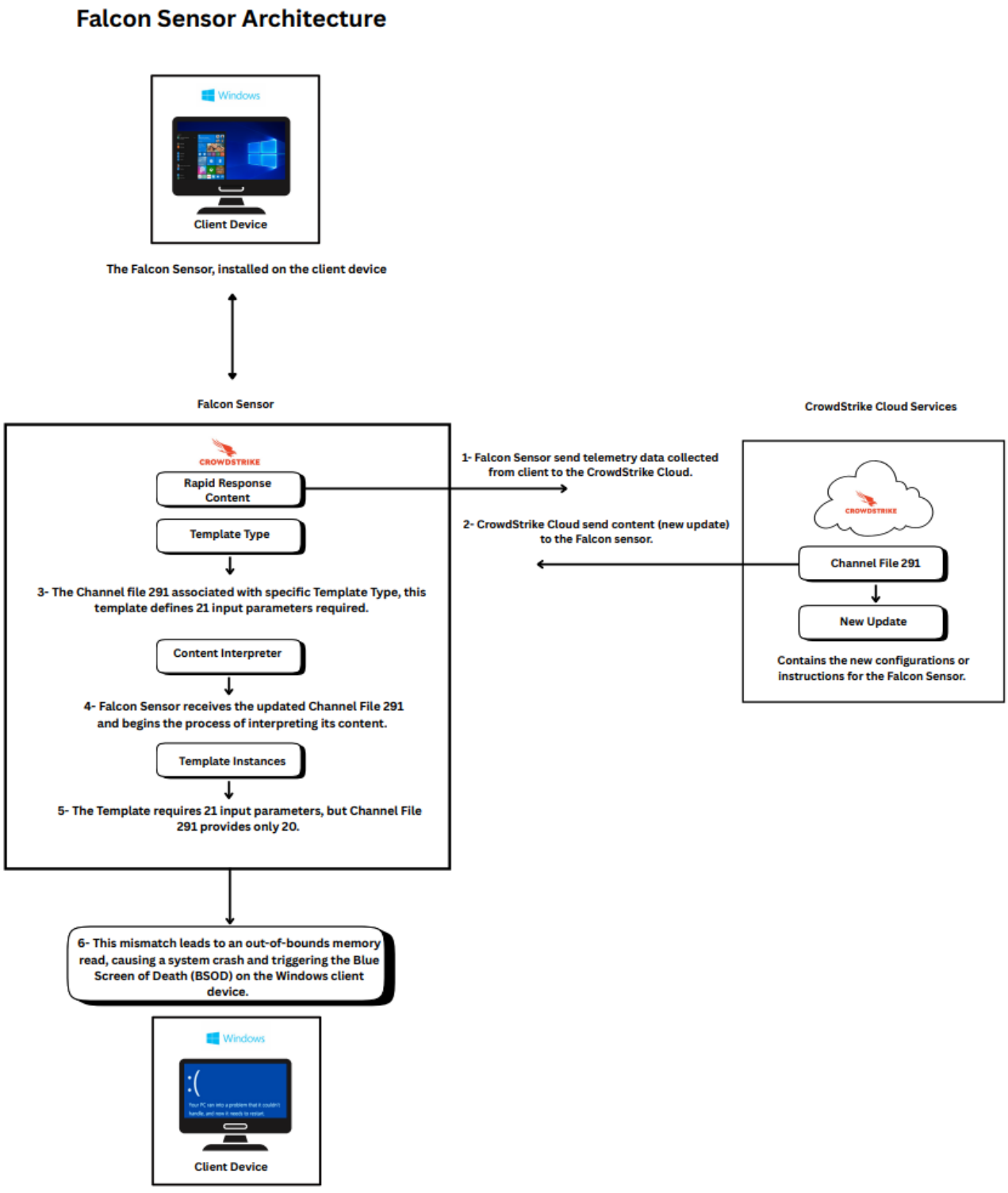

Fig. 3. Falcon Sensor Architecture

C. Findings and Mitigations

CrowdStrike clarified that the sensor compile process failed to verify the number of fields in the IPC Template Type, and the Content Interpreter lacked runtime array bounds checks. During the sensor compile phase, the code is typically reviewed to ensure it is precise, efficient, and capable of protecting systems against potential threats. The primary issue arose in Channel File 291, where the sensor code specified 20 input sources, while the IPC Template Type defined 21 inputs. This mismatch went undetected during development, highlighting a gap in the quality assurance (QA) testing process. Additionally, the Content Interpreter did not include a runtime array bounds check, allowing attempts to access non-existent array inputs, which could lead to system instability or failure. To address these issues, CrowdStrike released a patch that introduced a validation mechanism to ensure the number of inputs in the template type is verified during the compile process, along with adding a runtime bounds check to the Content Interpreter function [9]. The initial lack of validation stemmed from an insufficient software testing process, suggesting that the incident could have been avoided with more thorough testing and validation procedures. Implementing various validation methods such as fuzz testing, regression testing, formal verification, and canary testing could significantly enhance the overall testing process. Fuzz testing, for instance, improves the reliability and security of software updates by injecting malformed or random inputs to identify abnormal behaviors, while regression testing ensures that applications function correctly after code changes. Formal verification uses mathematical methods to confirm that software meets specified requirements, and canary testing rolls out new versions to a small user group to catch issues early, preventing larger impacts [11][12].

D. Phishing Campaigns

The phishing campaigns achieved high success rates, as global panic enabled the attackers to leverage the situation to trick CrowdStrike clients. Attackers are exploiting the outage incident through a phishing campaign. They impersonate CrowdStrike in fraudulent emails, falsely claiming to provide troubleshooting assistance for the outage. To deceive clients, the attackers employ various techniques, including voice scams, malicious emails with harmful links, typosquatting domains, and ZIP files containing malware or malicious scripts, all designed to target CrowdStrike clients [13]. Additionally, masquerading attacks involve manipulating malicious files and programs by altering their metadata and renaming them with legitimate names to appear trusted. To effectively detect the masquerading attack, it is recommended to use file integrity monitoring (FIM). This technology identifies unauthorized changes to files, programs, directories, and systems [14].

The real-world cases of malware infections:

1- Stealer Macro Malware:

Macro malware is stealthily injected into Microsoft Office files. The attackers, impersonating Microsoft, created a fake recovery manual in a Word document that contained hidden stealer macro malware designed to steal CrowdStrike client information. They delivered this malicious document via email attachments or as ZIP files to their victims. Once activated, the stealer malware terminates all running browser processes and proceeds to collect sensitive credentials, including login data and cookies. Its goal is to extract confidential information from the victim's device. After stealing the data, the malware saves it in a text file within the temporary %TMP% folder. Finally, it transmits this text file to the attackers through a command and control server [15][16] see Fig. 4.

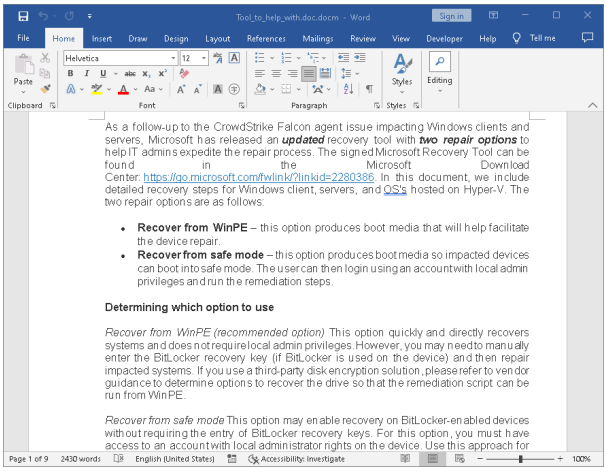

Fig. 4. Malicious Word attachment in the phishing email [17]

2- Data Wiping Malware new:

The attacker is exploiting the CrowdStrike outage by sending a phishing email with an attached malicious PDF file that claims to offer a solution to the issue. This PDF as illustrated in Fig. 5.contains a harmful link, and once the victim clicks it, a ZIP file containing wiper malware is downloaded. The wiper malware then executes its payload, wiping the data by overwriting files with zero bytes, effectively destroying the stored information on the device [16][17].

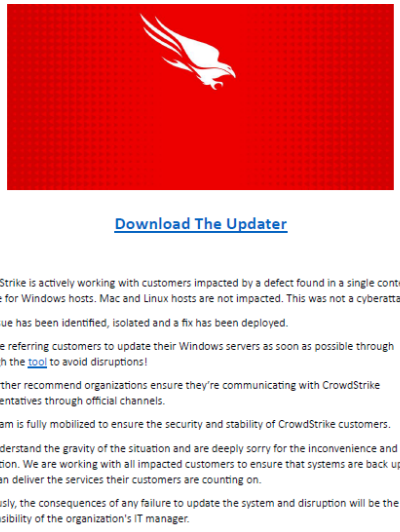

Fig. 5. Malicious PDF attachment in the phishing email [17]

E. Financial And Operational Impacts

IT disruptions caused significant financial losses and operational impacts across several sectors, including aviation, healthcare, business, government, and banking. The outage obstructed operational efficiency in all these sectors, leading to delays in services. For instance, airlines experienced flight delays and cancellations, with some passengers receiving handwritten boarding passes due to system failures. In healthcare, providers encountered similar challenges; medical staff could not access electronic health records, medical histories, and treatment plans, which posed risks to patient safety. Additionally, many businesses experienced service interruptions, resulting in financial losses and decreased productivity. The government sector, interconnected with departments such as transport authorities, emergency response networks, and customs systems, is particularly vulnerable; any disruption in this area can affect all services. Furthermore, banking faced challenges due to disruptions in critical services like transaction processing, user account access, and delayed wire transfers, leading to further financial losses and operational impacts [18][19]. The economic loss due to downtime varies depending on the number of affected clients and the duration of the outage. While the precise amount of losses from the CrowdStrike outage is not publicly available, Fortune magazine publishes an annual list of the 500 largest companies based on their total revenue. The incident affected over 25% of Fortune 500 companies. The most impacted sectors are airlines, with an impact rate of 100%; banking, with an impact rate of 76%; and healthcare, with an impact rate of 75%. The estimated financial impact is reported to be $5.4 billion, according to the Parametrix report. As a result of the outage incident, CrowdStrike's reputation suffered, leading to a significant drop in its stock price [20].

F. Saudi Arabia's Situation During the Incident

The CrowdStrike incident outage had a significant global impact; however, the Kingdom of Saudi Arabia was less affected, according to the Saudi National Cybersecurity Authority (NCA) [21]. This was largely due to legislation issued by the NCA that prevents data transfer outside the geographical boundaries of the Kingdom. This legislation conflicts with CrowdStrike's operations, as the company relies on sending information from user devices to its main center in the United States [22]. Additionally, the Saudi Central Bank (SAMA) confirmed that all its systems, including banking and national payment systems, remained secure. SAMA stated that it regularly reviews and updates its precautionary measures to maintain the efficiency and resilience of its business continuity plan and banking systems, thereby ensuring high operational efficiency [23]. During the incident, airports activated the joint operations room to implement appropriate plans for maintaining operational continuity [24].

G. Comparison With Similar Incidents

TABLE I
COMPARISON TABLE

Aspect	CrowdStrike	SolarWinds	Kaseya
Company Overview	CrowdStrike is a provider of cloud-native cybersecurity solutions company aimed at protecting endpoints.	SolarWinds is an IT infrastructure management software company that specializes in monitoring network devices and traffic.	Kaseya provides IT management and monitoring solutions that enable remote monitoring and management of clients through its Virtual System Administrator (VSA) platform.
Platform	Falcon Platform	Orion Platform	VSA Platform
Type of Incident	Accidental global IT outage due to faulty update	Compromised software update	Zero-day vulnerability in Kaseya VSA software
Type of Cyber Attack	Not a cyber attack	Supply chain attack	Ransomware supply chain attack
Cause	Faulty update to the Falcon Sensor	Malicious update contains a backdoor known as SUNBURST injected by attackers into SolarWinds Orion software	Leveraging a zero-day vulnerability in Kaseya VSA software allowed unauthorized access to VSA servers, leading to the distribution of ransomware to all connected client systems.
Year	2024	2020	2021
Downtime Duration	A few days	Months	Several days to weeks
Estimated Economic Impact	$5.4 billion	$100 billion	More than $1 billion
Supply Chain Vulnerability	Internal software defect	Compromise of the software development lifecycle	Zero-day vulnerability in VSA software
SDLC or QA Failure	QA testing failure	SDLC failure	QA testing failure

Table I compare three significant software supply chain incidents involving CrowdStrike, SolarWinds, and Kaseya, outlining aspects such as incident types, platforms, causes, and impacts. In 2024, CrowdStrike faced a global IT incident due to an internal software defect in an update to the Falcon Sensor. This issue arose from a failure in quality assurance (QA) testing, which did not detect the defect before deployment. As a result, the company experienced several days of downtime and an estimated loss of $5.4 billion in revenue. Although this incident was not a direct cyberattack, it indirectly highlighted significant risks within the supply chain [19][25]. In contrast, SolarWinds experienced a more explicit attack in 2020, exploiting a compromised Software Development Lifecycle (SDLC). Insufficient security measures enabled attackers to target the build environment, modifying the source code by injecting the SUNBURST backdoor into the Orion software, which went undetected. This malware was distributed to clients through a legitimate update, allowing the theft of sensitive data and resulting in impacts that lasted for months, with an estimated loss of $100 billion in revenue [26].

Furthermore, the 2021 Kaseya ransomware attack resulted from inadequate security testing and QA failures, which permitted a zero-day vulnerability to remain undetected in production. Attackers exploited this flaw in Kaseya's Virtual System Administrator (VSA) software, bypassing authentication to gain unauthorized access to VSA servers. The ransomware was distributed to clients, leading to several days or even weeks of downtime, ransom payments, and estimated losses exceeding $1 billion [27]. While these incidents illustrate common vulnerabilities in the software supply chain, the QA testing failures in CrowdStrike and Kaseya, although similar, led to different consequences: an accidental outage for CrowdStrike and a ransomware attack for Kaseya. In contrast, the SolarWinds incident primarily resulted from a compromise in the SDLC rather than QA testing shortcomings. Overall, vulnerabilities within the supply chain and dependencies on third-party software pose significant risks in cybersecurity. By learning from these incidents and past failures, organizations can strengthen their defenses against future threats. Adopting secure development practices, ensuring software compliance, managing third-party dependencies, implementing zero-trust architecture, and enforcing stringent identity and access management are crucial steps [28]. These measures are essential for securing the software development process and enhancing QA practices to mitigate risks associated with supply chain vulnerabilities and maintain trust in software and services.

IV. Solutions & Remediation Techniques

Software deployment testing is a critical practice that involves thoroughly testing new software updates before their release to ensure applications are free from bugs or other potential issues. A variety of techniques are employed in this process, including sandbox testing [29], rollback system techniques [30], virtual desktop infrastructure (VDI) [31], blue-green deployment, and canary deployment. By implementing software deployment testing, organizations can proactively address potential issues in future updates and maintain the availability and reliability of their applications. Fig. 6 provides an overview of each technique and its concept.

Table II provides a summary of various techniques, including their advantages, disadvantages, and associated costs, that can be utilized

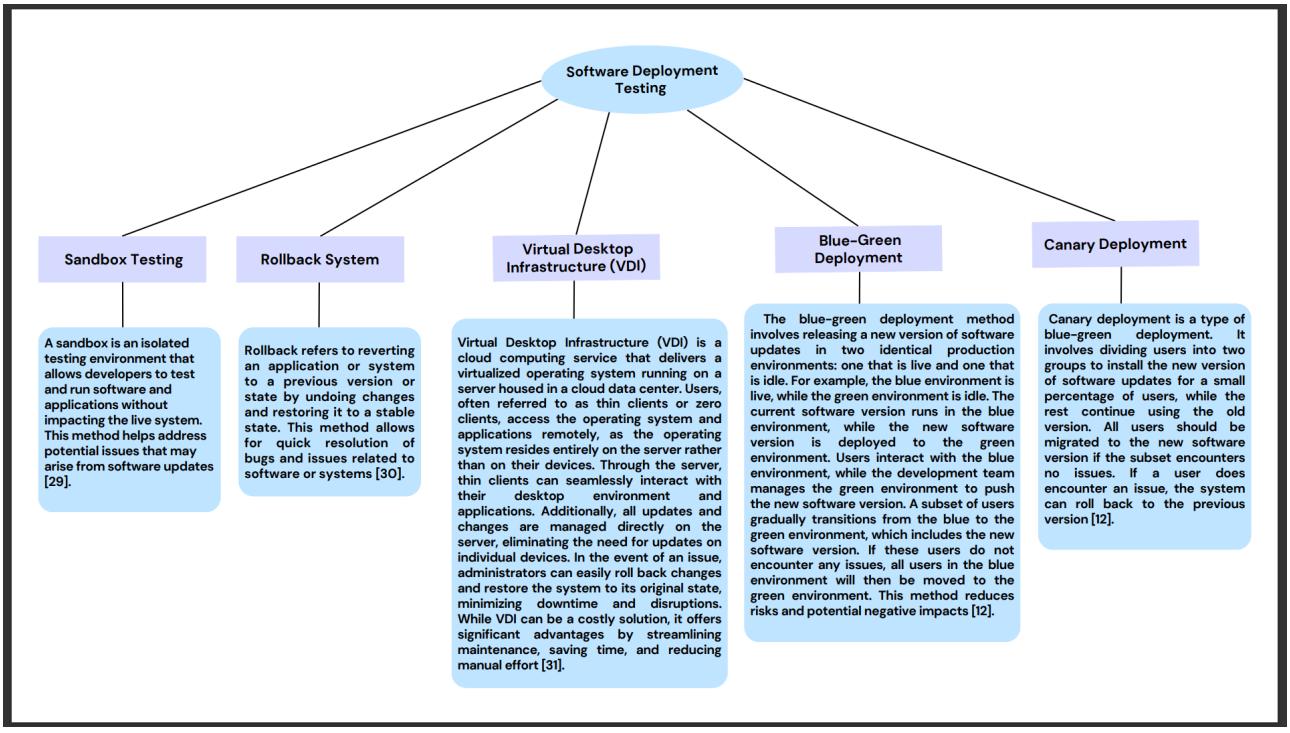

Fig. 6. Conceptual diagram of Software deployment testing techniques

to mitigate the risks and impacts of issues arising from new software updates. Organizations can select the most appropriate technique based on their budget and specific requirements. According to CrowdStrike, following an incident, they plan to implement canary testing [9]. This method involves dividing users into two groups as shown in Fig. 7 95% will continue using the existing version of the software, while 5% will receive the new software update. Canary testing minimizes risk by initially releasing

the new version to a small subset of users, allowing potential issues to be identified and addressed before the update is rolled out to the broader user base. This approach enhances system stability by detecting and resolving problems early. The testing process further involves gradually introducing new features and closely monitoring their performance. If the subset of users encounters no issues, the remaining users are seamlessly migrated to the updated version. However, if any problems occur, the system can quickly roll back to the previous version to avoid disruptions [12].

TABLE II
SUMMARY TABLE

Technique	Advantages	Disadvantages	Cost
Sandbox Testing	- Testing the new version of the software in a safe environment	Complexity	Moderate
Rollback System	- Enhanced disaster recovery	Risk of data consistency issues	Moderate
VDI	- Control the infrastructure remotely - Scalability	Latency issues that impact the user experience	High
Blue-Green Deployment	- Allows quick rollbacks if issues arise - Minimizes risks - Reduce downtime	Requires two environments which will increase the cost	High
Canary Deployment	- Allows quick rollbacks if issues arise - Minimizes risks - Reduce downtime	Complexity	Moderate (Lower than Blue-Green deployment)

V. METHODOLOGY

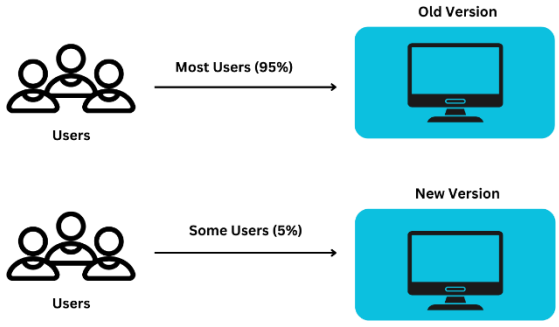

Fig 7. Canary Testing Deployment

The methodology section explains the process of conducting the case study, detailing the analysis approaches and data collection methods. This ensures that readers gain a clear understanding of the research process and can assess the study's reliability. The notable lack of case studies on the recent CrowdStrike incident leaves many questions unanswered, making it a compelling and timely subject for investigation.

A. Analysis Approach

This study uses both qualitative and quantitative approaches to analyze the CrowdStrike incident, allowing for a deeper understanding of the event.

1. Qualitative approach:

The qualitative approach involves analyzing textual data, emphasizing technical details to provide an in-depth explanation of complex events, such as the CrowdStrike outage incident.

2. Quantitative approach:

The quantitative approach focuses on analyzing numerical data to conduct statistical analyses, providing insights into the financial impacts and operational downtime caused by the CrowdStrike incident.

B. Data Collection Sources

Data collection is the process of acquiring data from different sources. To conduct the case study, a variety of primary and secondary data sources were included. These sources were analyzed to explain the CrowdStrike incident in detail and answer the readers' questions.

1) Primary sources:

Data collected for a specific purpose originated from original sources created by individuals who directly experienced the event. This data consists of original documents, not interpretations or summaries from other sources. In this case study, primary sources include technical reports, official blogs, press releases, and social media posts from authoritative organizations. This diverse range of materials enhances the reliability and depth of the analysis.

- **Technical Reports:**
  These documents provide an in-depth analysis of specific technical topics. This study utilizes the CrowdStrike Root Cause Analysis (RCA) report, which is a type of technical report.
- **Official Blogs:**
  The official CrowdStrike blog provides direct, original information and analysis of the incident.
- **Press Releases:**
  An official statement issued by organizations to announce or share information about new events, specifically the CrowdStrike incident.
- **Official Social Media Posts:**
  Official accounts on the X platform, including CrowdStrike, Microsoft, airports, banks, and more, disseminate news, updates, and announcements related to the CrowdStrike incident outage.

2) Secondary sources:

Data is derived from the interpretation, analysis, and summarization of primary sources and is generated by individuals other than the original source. In this case study, secondary sources encompass journal papers, news articles, white papers, and blogs. Leveraging these secondary sources facilitates diverse interpretations of primary data, offering a more comprehensive understanding of the topic. This approach enhances the analysis by incorporating various perspectives and insights.

- **Journal Papers:**

  A wide range of journal articles, including research papers and review articles are utilized to provide detailed insights into the CrowdStrike outage incident.

- **News Articles:**
  Journalists analyze and interpret information from various sources to provide clarity and detailed insights into the incident through news articles.
- **White Papers:**
  A document that does not present original research findings directly but instead provides a detailed report interpreting and discussing information from various sources about the CrowdStrike incident.
- **Blogs:**

  Some general cybersecurity blogs are utilized, offering information about the outage incident based on insights from other sources.

C. Selection Criteria

A diverse range of sources was selected based on key criteria, including relevance, timeliness, and credibility, to ensure the reliability, accuracy, and pertinence of the data regarding the CrowdStrike outage incident.

- **Relevance:**
  The selection was based on sources relating directly to CrowdStrike incident outage data, including root causes, technical analysis, key findings, impacts, and consequences to ensure data reliability.
- **Timeliness:**
  The sources were selected based on recent publications to ensure the data is accurate, reliable, and not outdated.
- **Credibility:**

  The data was collected from official credible sources, such as CrowdStrike,

Microsoft, and published research papers, to ensure data trustworthiness.

VI. Lessons learned & recommendation

The lessons learned from the CrowdStrike incident underscore the critical importance of effective software supply chain security management, rigorous software deployment testing to ensure software quality and system reliability, and the implementation of robust business continuity planning (BCP) and incident response plans to minimize disruptions to business operations. Preparing a BCP and an incident response plan in advance enhances organizational resilience and helps maintain system availability during crises. Furthermore, maintaining up-to-date snapshots and backups is essential for ensuring business continuity, as best practices in managing these resources enable organizations to recover quickly from incidents and sustain operational stability.

VII. Conclusion

The CrowdStrike incident had a global impact, causing widespread disruptions across various sectors, businesses, and services. The outage was traced to a faulty software update in the Falcon sensor, which resulted in a Blue Screen of Death (BSOD) for Microsoft Windows users. The root cause was identified as a mismatch between parameters in the sensor code and the IPC Template Type. To address the issue, CrowdStrike implemented a runtime array bounds check in the Content Interpreter function to retrieve inputs, as well as checks to validate the number of inputs in the Template Type. Additionally, various techniques were discussed to enhance response times and mitigate risks in the future. To prevent similar technological failures, the adoption of canary testing and effective incident response strategies is essential.

Funding

The author of this article did not receive any particular grant from any public, commercial, or not-for-profit funding agency.

Conflict of interest

The author declares that there is no conflict of interest.

References

[1] D. Weston, “Helping our customers through the CrowdStrike outage - The Official Microsoft Blog.” Accessed: Sep. 17, 2024. [Online]. Available: https://blogs.microsoft.com/blog/2024/07/20/helping-our-customers-through-the-crowdstrike-outage/

[2] T. Singla, D. Anandayuvaraj, K. G. Kalu, T. R. Schorlemmer, and J. C. Davis, “An Empirical Study on Using Large Language Models to Analyze Software Supply Chain Security Failures,” vol. 1, Aug. 2023, doi: 10.1145/3560835.3564556.

[3] “Technical Details: Falcon Update for Windows Hosts | CrowdStrike.” CrowdStrike Blog. Accessed: Sep. 12, 2024. [Online]. Available: https://www.crowdstrike.com/en-us/blog/falcon-update-for-windows-hosts-technical-details/

[4] “User Mode and Kernel Mode - Windows drivers | Microsoft Learn.” Accessed: Sep. 14, 2024. [Online]. Available: https://learn.microsoft.com/en-us/windows-hardware/drivers/gettingstarted/user-mode-and-kernel-mode

[5] N. Premakanthan, “Analysis of the CrowdStrike Software Update Failure.” Accessed: Apr. 03, 2025. [Online]. Available: https://www.researchgate.net/publication/383084159_Analysis_of_the_CrowdStrike_Software_Update_Failure

[6] “File System Redirector - Win32 apps | Microsoft Learn.” Accessed: Sep. 14, 2024. [Online]. Available: https://learn.microsoft.com/en-us/windows/win32/winprog64/file-system-redirector

[7] “Glossary of Terms.” CrowdStrike. Accessed: Sep. 26, 2024. [Online]. Available: https://www.crowdstrike.com/wp-content/uploads/2024/07/GlossaryOFTerms.pdf

[8] P. McCormack, “The 2024 CrowdStrike Incident - Simply Explained.” Accessed: Sep. 29, 2024. [Online]. Available: https://www.linkedin.com/pulse/2024-crowdstrike-incident-simply-explained-patrick-mccormack-jdxqc

[9] “External Technical Root Cause Analysis — Channel File 291,” Aug. 2024. Accessed: Oct. 03, 2024. [Online]. Available: https://www.crowdstrike.com/wp-content/uploads/2024/08/Channel-File-291-Incident-Root-Cause-Analysis-08.06.2024.pdf

[10] "CrowdStrike Global Outage." Marco Blog. Accessed: Apr. 02, 2025. [Online]. Available: https://www.marconet.com/blog/crowdstrike-global-outage

[11] M. de Rosa, "CAN Bus Security Analysis: a Fuzzing Approach," 2024, [Online]. Available: https://webthesis.biblio.polito.it/30899/

[12] V. Vidyasagar, "Blue-Green and Canary Deployments in DevOps: A Comparative Study," vol. 15, no. 01, pp. 1047–1063, 2024, doi: 10.5281/ZENODO.15483641.

[13] "Falcon Sensor Issue Likely Used to Target CrowdStrike Customers," CrowdStrike Blog. Accessed: Nov. 18, 2024. [Online]. Available: https://www.crowdstrike.com/en-us/blog/falcon-sensor-issue-use-to-target-crowdstrike-customers/

[14] "Masquerading, Technique T1036 - Enterprise | MITRE ATT&CK®." Accessed: Nov. 18, 2024. [Online]. Available: https://attack.mitre.org/techniques/T1036/

[15] "Threat Actor Uses Fake Recovery Manual to Deliver Unidentified Stealer," CrowdStrike Blog. Accessed: Apr. 05, 2025. [Online]. Available: https://www.crowdstrike.com/en-us/blog/fake-recovery-manual-used-to-deliver-unidentified-stealer/

[16] L. Mathur, V. Chole, and A. Karnik, "The Scam Strikes Back: Exploiting the CrowdStrike Outage | McAfee Blog." Accessed: Apr. 05, 2025. [Online]. Available: https://www.mcafee.com/blogs/other-blogs/mcafee-labs/the-scam-strikes-back-exploiting-the-crowdstrike-outage/

[17] "Find Threats Exploiting CrowdStrike Outage with TI Lookup - ANY.RUN's Cybersecurity Blog." Accessed: Apr. 05, 2025. [Online]. Available: https://any.run/cybersecurity-blog/crowdstrike-outage-abuse/

[18] O. Ogundipe and T. Aweto, "The shaky foundation of global technology: A case study of the 2024 CrowdStrike outage," International Journal of Multidisciplinary Research and Growth Evaluation, vol. 5, no. 5, pp. 106–108, 2024, Accessed: Apr 28, 2025. [Online]. Available: www.allmultidisciplinaryjournal.com

[19] Dr. A. S. George, "When Trust Fails: Examining Systemic Risk in the Digital Economy from the 2024 CrowdStrike Outage," Partners Universal Multidisciplinary Research Journal, vol. 1, no. 2, pp. 134–152, Jul. 2024, doi: 10.5281/ZENODO.12828222.

[20] "CrowdStrike's Impact on the Fortune 500 An Impact Analysis," 2024. Accessed: May 06, 2025. [Online]. Available: https://www.parametrixinsurance.com/reports-white-papers/crowdstrikes-impact-on-the-fortune-500

[21] "Press Release from the National Cybersecurity Authority | NCA." Accessed: May 07, 2025. [Online]. Available: https://nca.gov.sa/ar/news/1370/

[22] "Regulation on Personal Data Transfer outside the Geographical Boundaries of the Kingdom of Saudi Arabia," Jul. 2023. Accessed: May 07, 2025. [Online]. Available: https://www.crowdstrike.com/wp-content/uploads/2023/09/Saudi-Arabia-Data-Transfer-Regulation-Comments.pdf

[23] "SAMA Confirms Safety of Payment Systems and Banking Systems in Saudi Arabia." Accessed: May 07, 2025. [Online]. Available: https://www.sama.gov.sa/en-us/news/pages/news-1035.aspx

[24] "How Jeddah Airports interacted with the global technical systems crisis?" Jul. 19, 2024. Accessed: May 07, 2025. [Online]. Available: https://x.com/JedcoKSA/status/1814333783924973671

[25] H. İş, "Evaluating and Mitigating Cybersecurity Threats from System Update Vulnerabilities through the CrowdStrike Case," European Journal of Technique (EJT), vol. 14, no. 2, pp. 182–188, Dec. 2024, doi: 10.36222/EJT.1564440.

[26] H. Ghanbari, K. Koskinen, and Y. Wei, "From SolarWinds to Kaseya: The rise of supply chain attacks in a digital world," Journal of Information Technology Teaching Cases, 2024, doi: 10.1177/20438869241299823/SUPPL_FILE/SJ-PDF-1-TTC-10.1177_20438869241299823.PDF.

[27] S. Bhunia, M. Blackert, H. Deal, A. DePero, and A. Patra, "Analyzing the 2021 Kaseya Ransomware Attack: Combined Spearphishing Through SonicWall SSLVPN Vulnerability," IET Information Security, vol. 2025, no. 1, p. 1655307, Jan. 2025, doi: 10.1049/ISE2/1655307.

[28] A. Akinsola and A. Akinde, "Enhancing Software Supply Chain Resilience: Strategy For Mitigating Software Supply Chain Security Risks And Ensuring Security Continuity In Development Lifecycle," International Journal on Soft Computing, vol. 15, no. 1/2, pp. 01–18, Jul. 2024, doi: 10.5121/ijsc.2024.15201.

[29] L. Koycheva and A. VandenBroek, "Sandbox innovation: Potentials and impacts," Practicing Anthropology, vol. 46, no. 1, pp. 36–45, Jan. 2024. doi:10.1080/08884552.2024.2307293

[30] H.-N. Nguyen, T.-T. Nguyen, T.-N. N. Thi, M.-D. Tran, and B.-H. Tran, "Proposed methods T O rollback a failed update of IOT devices," International Journal of Engineering and Advanced Technology, vol. 11, no. 2,

pp. 55–62, Dec. 2021. doi:10.35940/ijeat.b3297.1211221

[31] A. A. Ibrahim, D. Kliazovich, P. Bouvry, and A. Oleksiak, "Virtual Desktop Infrastructures: Architecture, survey and Green Aspects proof of concept," 2016 Seventh International Green and Sustainable Computing Conference (IGSC), 2016. doi:10.1109/igcc.2016.7892624