

Forensic Reconstruction of Web-Based Social Media Account Takeover via Credential Phishing

Zulki Yanto Rivai¹, Anton Yudhana², Imam Riadi³

¹Master Program of Informatics, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

²Department of Electrical Engineering, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

³Department of Information System, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

Received:

July 12, 2026

Revised:

September 18, 2026

Accepted:

September 27, 2026

Published:

October 24, 2026

Corresponding Author:

Author Name*:

Anton Yudhana

Email*:

eyudhana@ee.uad.ac.id

DOI:

10.63158/journalisi.v8i5.1968

© 2026 Journal of Information Systems and Informatics. This open access article is distributed under a (CC-BY License)



Credential phishing against browser-authenticated accounts can produce fragmented digital traces across browser storage, phishing infrastructure, network traffic, and account activity records. This study aims to reconstruct a Discord Web account takeover caused by credential phishing through multi-source forensic correlation. Evidence was obtained from Chrome browser artifacts, phishing server logs, Wireshark network captures, and documentation of attacker-side account activity within a controlled experimental scenario. The forensic analysis recovered 10 of 12 expected artifacts, resulting in an Artifact Recovery Rate (ARR) of 83.3%, while 4 of 6 incident stages were supported by at least two evidence items or source layers, producing an Evidence Corroboration Score (ECS) of 66.7%. The reconstructed timeline covers six stages of the incident, from the delivery of a phishing link to the confirmation of account access. The study introduces ARR and ECS as quantitative measures for evaluating artifact recovery and cross-source evidence corroboration. The findings further demonstrate that browser artifacts alone are insufficient to establish a complete account takeover sequence, and that correlation among browser, server, network, and account activity evidence is necessary to construct a coherent and verifiable forensic timeline. These results provide a practical basis for incident responders and forensic examiners to correlate heterogeneous evidence sources when investigating browser-mediated credential phishing.

Keywords: credential phishing; Discord Web; forensic reconstruction; multi-source correlation; browser forensics

1. INTRODUCTION

Web-based social media platforms increasingly rely on browser-based authentication, making them susceptible to credential phishing attacks that result in account takeover. Among these platforms, Discord is widely used for community interaction, gaming, education, and professional communication [1], [2], [3], with more than 373.6 million registered users worldwide as of 2025 [4]. Discord Web enables account access directly through a browser without a dedicated desktop application, making browser-based authentication a critical access mechanism [5]. This browser-mediated authentication and session mechanism also creates opportunities for credential phishing, in which users are redirected to fake login pages that mimic legitimate authentication interfaces [6], [7], [8]. In Discord Web environments, submitted credentials can be reused by an attacker to access the Victim's account through the official platform, resulting in account takeover.

The forensic reconstruction of a Discord Web account takeover is challenging because relevant digital evidence is distributed across multiple sources, including browser storage, phishing server logs, network traffic captures, and account activity records [9], [10], [11]. In the controlled scenario used in this study, the phishing domain `dĩiscord.online` was represented in browser and network artifacts using its Punycode form, `xn--dscord-wwaa.online`. This IDN-based visual spoofing can make the phishing domain appear similar to the legitimate Discord domain and may deceive users into entering credentials on a fraudulent login page. Browser artifacts may indicate access to a phishing page or retain traces associated with form input, while server logs and network captures can provide independent evidence of request paths, domain resolution, IP communication, timestamps, and TLS SNI associated with the phishing infrastructure [12], [13]. Consequently, individual artifacts cannot be interpreted in isolation because the reconstruction of an account takeover requires chronological correlation among evidence sources to establish a coherent sequence from phishing link delivery and credential input to subsequent unauthorized account access.

Previous research has established the forensic value of browser and web application artifacts in reconstructing user activity. Gupta et al. (2023) demonstrated that Discord Web activity on Google Chrome leaves recoverable browser artifacts, while Davis et al. (2022) identified Discord and Slack traces from volatile memory. Raza et al. (2024) mapped

browser artifacts across different stages of the browser lifecycle, and Wang et al. (2022) demonstrated that V8 memory forensics can recover traces associated with web applications. Other browser forensic studies have shown that artifacts such as History, Cookies, Cache, Local Storage, Session Storage, and Web Data can provide evidence for reconstructing user activity [9], [10], [14], [15]. These studies demonstrate that browser environments preserve diverse traces relevant to forensic examination, but the presence of individual artifacts does not necessarily establish the causal relationship between phishing activity and account takeover.

The importance of examining multiple artifacts and evidence sources is further supported by broader browser-based forensic research. Paligu and Varol (2022) demonstrated that forensic examination of browser-based social media activity on Instagram requires analysis of IndexedDB persistent storage in addition to conventional browser databases, indicating that platform-specific storage mechanisms require tailored multi-artifact analysis. Alotibi et al. (2024) showed that no single forensic tool consistently recovers all categories of browser artifacts, reinforcing the need for multi-tool and multi-source examination strategies. Similarly, Rzaeva et al. (2026) proposed an intelligent browser history forensics approach that correlates web activity logs, credentials, and behavioral profiles, demonstrating the value of cross-artifact correlation in reconstructing user activity. These findings indicate that the completeness of browser-based forensic reconstruction depends not only on artifact recovery but also on the correlation of evidence obtained from different artifacts and sources.

From the phishing perspective, previous studies have mainly examined web spoofing, phishing URLs, phishing behavior, and mechanisms for credential theft [6], [7], [8]. Although these studies provide an important foundation for understanding phishing attacks, identifying a phishing page or URL alone is insufficient to reconstruct an account takeover incident. Browser-side evidence needs to be correlated with server-side requests, network traffic, and account activity records to establish the relationship between credential input and subsequent unauthorized access. Server logs can provide evidence regarding request paths and access metadata, while network evidence such as DNS resolution, IP communication, timestamps, and TLS SNI can independently support the verification of communication with phishing infrastructure [12], [13]. The positioning of the present study relative to previous research is summarized in Table 1.

Table 1. Comparison of Previous Studies

Study	Scope	Main Contribution	Limitation
(Gupta et al., 2023); (Davis et al., 2022)	Discord forensics	Identified Discord artifacts from browser storage and volatile memory	Does not reconstruct credential phishing-based account takeover
(Raza et al., 2024); (Wang et al., 2022)	Browser and web application forensics	Mapped browser lifecycle artifacts and web application traces	Does not correlate browser, server, and network evidence
(Ahmed et al., 2023), (Asiri et al., 2023), (Jain & Gupta, 2022)	Phishing URLs and web spoofing	Explained phishing behavior and detection mechanisms	Detection-oriented rather than reconstruction-oriented
(Yudhana et al., 2022), (Komosny, 2023)	Network traffic and timestamp evidence	Supports validation of DNS, IP communication, and time-based evidence	Not specific to Discord Web phishing reconstruction
Proposed study	Discord Web credential phishing	Reconstructs account takeover using browser, server, network, and account activity correlation; evaluates reconstruction completeness and corroboration using ARR and ECS	–

As shown in Table 1, previous studies have addressed Discord artifacts, browser and web application traces, phishing mechanisms, and network-based evidence separately. A few researchers have focused on Discord artifacts, browser and web application traces, phishing mechanisms, or network-based evidence separately. However, limited attention has been given to integrating browser artifacts, phishing server logs, network traffic captures, and account activity records into a single causally ordered timeline for reconstructing a Discord Web account takeover. Previous studies have also not explicitly

evaluated the completeness of recovered evidence and the degree of cross-source corroboration supporting reconstructed incident stages. Therefore, this research intends to address these limitations through a multi-source forensic reconstruction approach that correlates heterogeneous evidence sources to establish the sequence of credential phishing and subsequent account takeover. The objectives of this research are to reconstruct the Discord Web account takeover timeline, evaluate the completeness of recovered forensic artifacts, and assess the degree of cross-source evidence corroboration using the Artifact Recovery Rate (ARR) and Evidence Corroboration Score (ECS).

To address this gap, this study reconstructs a Discord Web account takeover caused by credential phishing through multi-source forensic correlation. A controlled experimental scenario is used to generate evidence from browser, server, network, and account activity sources. ISO/IEC 27037 is used as a supporting reference for evidence handling to ensure that digital artifacts are appropriately preserved before examination, while the analytical focus remains on artifact recovery, evidence corroboration, and timeline reconstruction [16]. The proposed approach integrates browser artifacts, phishing server logs, network traffic captures, and account activity records to establish a causally ordered sequence of the phishing and account takeover process.

The main contributions of this study are threefold. First, it demonstrates a multi-source forensic reconstruction approach that integrates browser artifacts, phishing server logs, network traffic captures, and account activity records into a causally ordered account takeover timeline. Second, it identifies Web Data autofill traces and Autopsy session artifacts as important indicators of credential input when server-side logs and saved browser passwords do not record the submitted data. Third, it introduces two reconstruction-oriented indicators, namely Artifact Recovery Rate (ARR) and Evidence Corroboration Score (ECS), to evaluate artifact completeness and the degree of cross-source support for reconstructed incident stages. This approach provides a basis for investigating browser-authenticated platforms exposed to credential phishing and demonstrates how multi-source evidence correlation can strengthen forensic reconstruction of account takeover incidents.

2. METHODS

2.1. Experimental Scenario and Evidence Sources

This study employed an experimental digital forensic approach to reconstruct a Discord Web account takeover caused by credential phishing. The experiment was conducted in a controlled environment that included a victim workstation, an attacker workstation, a simulated phishing page, phishing server records, network captures, and a separate forensic workstation. This design was used to generate evidence across browser, server, network, and account activity sources, which were then correlated into a reconstructed incident timeline. No real victims, real credentials, or uncontrolled third-party infrastructure were involved. Figure 1 illustrates the experimental scenario and the multi-source evidence correlation design used in this study.

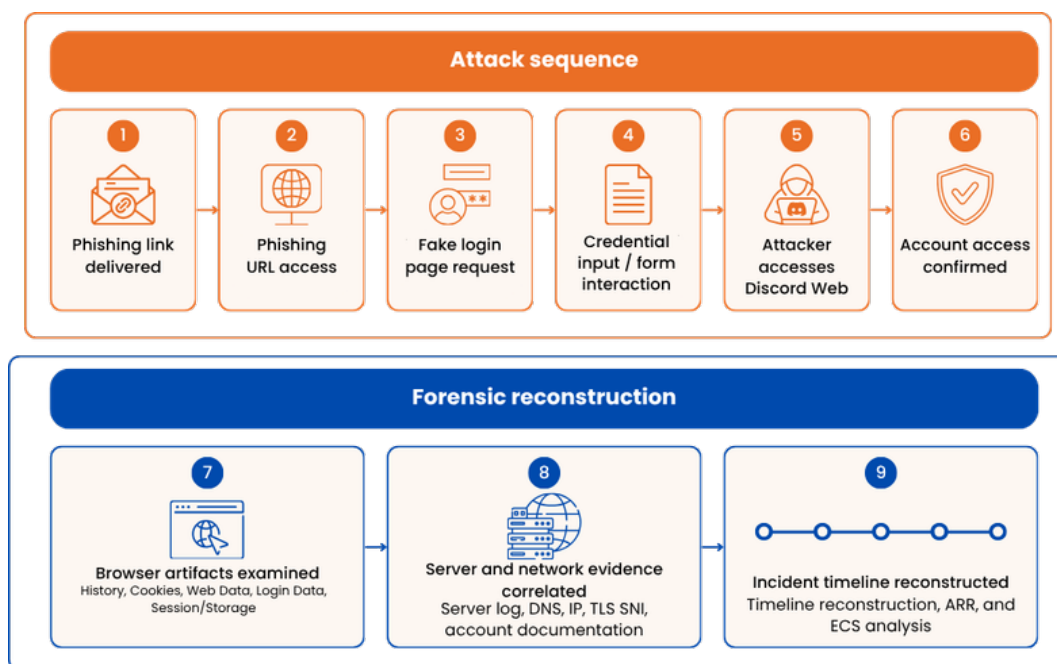


Figure 1. Credential Phishing and Multi-Source Forensic Reconstruction Scenario on Discord Web

As shown in Figure 1, the scenario covers two phases: the phishing sequence (stages 1-6) and the forensic reconstruction process (stages 7-9). The simulated phishing domain `discord.online` appeared in browser and network artifacts as `xn--discord-wwaa`, online due to IDN representation [17]. Browser-side examination focused on History, Cookies, Web Data, Login Data, Cache, Local Storage, and Session Storage. Server-side examination

used the /login.html access record, while network examination used Wireshark PCAP to validate DNS resolution, IP communication, TCP/TLS communication, and TLS SNI [13]. Table 2 summarizes the experimental environment, evidence sources, forensic tools, and analysis strategy used in this study.

Table 2. Experimental Environment and Forensic Tools

Component	Specification / Role
Victim environment	Windows 11 Pro workstation with Google Chrome browser
Attacker environment	Browser-based Discord Web access and simulated phishing page control
Phishing domain	dìiscord.online, represented as xn--dscord-wwaa.online
Phishing server evidence	Access log containing /login.html request, timestamp, IP address, referrer, and user-agent
Browser acquisition	FTK Imager for Chrome browser profile acquisition
Browser database examination	DB Browser for SQLite for History, Cookies, Web Data, and Login Data
File and artifact examination	Autopsy for cache, local storage, session storage, and keyword-based traces
Network examination	Wireshark for DNS, IP communication, TCP/TLS, and TLS SNI
Timeline reconstruction	Manual correlation of browser, server, network, and account activity evidence
Data analysis strategy	Artifact Recovery Rate, Evidence Corroboration Score, and timeline consistency analysis

As shown in Table 2, each tool was assigned according to its forensic role. FTK Imager was used for browser profile acquisition, DB Browser for SQLite and Autopsy were used for browser artifact examination, and Wireshark was used to validate communication with the phishing infrastructure [18], [19]. ISO/IEC 27037 was operationalized during evidence handling by first identifying and documenting the relevant evidence sources before acquisition. FTK Imager was used to acquire the Chrome browser profile as the forensic working copy, while the acquired data were kept separate from the original evidence source and subsequently examined using DB Browser for SQLite and Autopsy.

The acquisition activity was documented using evidence identifiers and acquisition records to maintain traceability between the source evidence, the acquired copy, and the subsequent examination activities. Examination was conducted on the acquired copy rather than directly on the original source, while the evidence-handling records were maintained throughout the process to support chain-of-custody continuity. These procedures were applied to preserve the integrity and traceability of the evidence before forensic examination, while the analytical focus remained on artifact recovery, evidence corroboration, and timeline reconstruction [16], [20].

2.2. Data Analysis Strategy

Evidence analysis followed three complementary procedures timeline reconstruction, artifact recovery analysis, and evidence corroboration analysis consistent with forensic event reconstruction frameworks (Bhandari & Jusas., 2020; Breitinger et al., 2025; Pathirana et al., 2026; Studiawan et al., 2026). For timeline reconstruction, timestamps extracted from Google Chrome SQLite databases were interpreted as Chrome Time values, represented as microseconds elapsed since 1601-01-01 00:00:00 UTC. Each Chrome Time value was converted to Unix time by dividing the value by 1,000,000 and subtracting 11,644,473,600 seconds, which accounts for the difference between the Chrome and Unix epochs. The resulting timestamp was interpreted in UTC and then converted to the local experiment time zone (UTC+07:00) to obtain a human-readable date and time. The normalized timestamps were then correlated with phishing server log and Wireshark timestamps using a common temporal reference for incident timeline reconstruction. Artifact recovery analysis was used to evaluate the completeness of recovered evidence by comparing recovered artifacts with expected artifacts. This approach is consistent with browser forensic studies that assess the availability of artifacts and the effectiveness of extraction across forensic tools and browsing environments [14], [25]. In this study, artifact completeness was expressed as the ARR at Equation 1:

$$ARR = \frac{\text{Recovered Artifacts}}{\text{Expected Artifacts}} \times 100\% \quad (1)$$

Evidence corroboration analysis was used to evaluate whether each reconstructed incident stage was supported by more than one independent evidence item or source layer [23], [26]. This measure was expressed as the ECS at Equation 2:

$$ECS = \frac{\text{Incident Stages Supported by } \geq 2 \text{ Evidence Items}}{\text{Total Incident Stages}} \times 100\% \quad (2)$$

ARR and ECS were used as operational reconstruction indicators in this study, not as standardized statistical measures. Together, timeline reconstruction, ARR, and ECS enabled evaluation of temporal consistency, artifact completeness, and cross-source support within the controlled forensic case-study design.

3. RESULTS AND DISCUSSION

3.1. Browser Artifact Findings

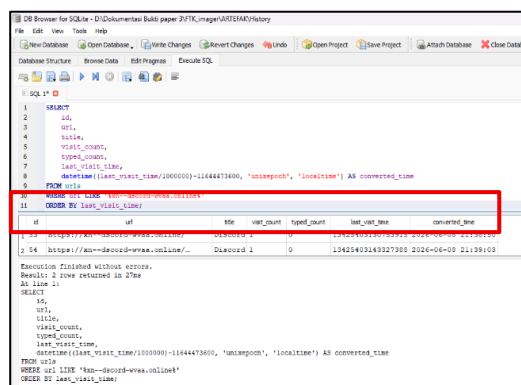
Browser artifact examination was conducted to identify victim-side traces related to phishing URL access, fake login interaction, credential input indicators, and Discord Web session activity. The recovered artifacts are summarized in Table 3.

Table 3. Browser Artifact Findings from the Victim Device

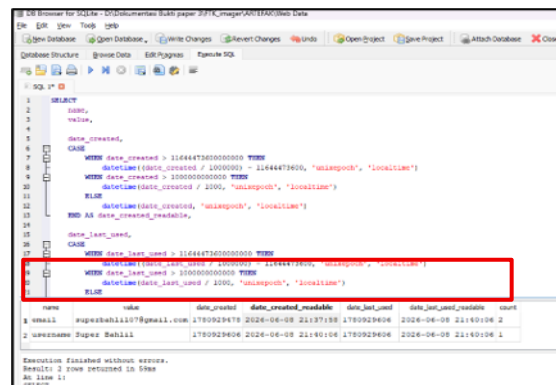
Artifact	Source / Location	Finding	Timestamp	Reconstruction Meaning
History	Browser History DB	https://xn--dscord-wwaa.online/discord.com,	08/06/2026 21:38	Victim accessed the phishing URL
Cookies	Cookies DB	.discordapp.com, .discord.gg, .discordapp.net	08/06/2026 21:38	Discord Web session traces were present
Cookies	Cookies DB	No phishing-domain cookie found	10/06/2026 13:00	Phishing page did not create a persistent cookie
Web Data	Autofill table	Dummy email and username	08/06/2026 21:40	Victim input was retained as autofill data
Login Data	Logins table	No saved login found	10/06/2026 19:37	Credential was not stored by the browser password manager

Artifact	Source / Location	Finding	Timestamp	Reconstruction Meaning
Autopsy Session	Session / Session	Phishing domain, login.html, dummy email, and username	10/06/2026 20:56	Session traces supported fake login interaction
Cache / Local Storage	Autopsy browser artifacts	Discord-related cache and local storage traces	10/06/2026 20:56	Browser storage supported Discord Web activity

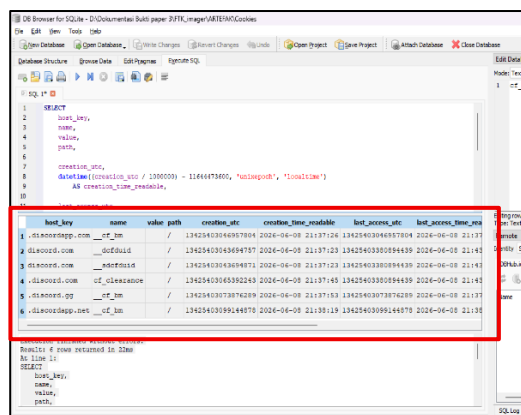
As shown in Table 3, browser artifacts confirmed access to the phishing URL, Discord Web session traces, and credential input indicators, but, on their own, did not establish the complete account takeover sequence. Representative browser evidence is shown in Figure 2.



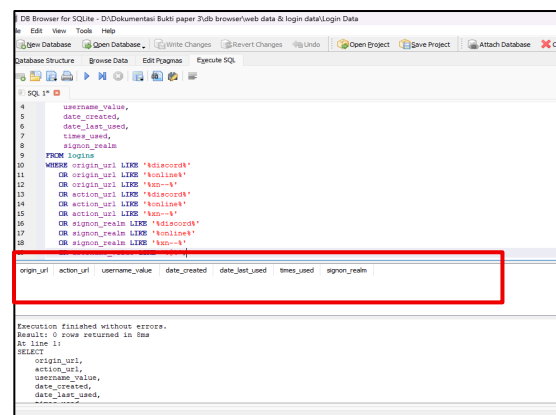
(a) Browser History showing phishing URL access



(b) Web Data showing dummy email and username



(c) Cookies showing Discord-related domains



(d) Login Data showing no saved credential

Figure 2. Browser Database Artifact Evidence Identified from the Victim Device

Figure 2 presents four browser-side findings: phishing URL access in History, dummy input retained in Web Data, Discord-related cookies, and the absence of saved credentials in Login Data. These findings indicate that credential input was supported by autofill and session-related artifacts rather than by Chrome's password database. Therefore, browser evidence provided the initial basis for reconstruction but required server, network, and account activity evidence to validate the subsequent stages of the incident.

The absence of a saved credential in Chrome's Login Data should be interpreted separately from the presence of dummy email and username values in Web Data. Login Data represents credentials that have been processed by the browser's password-management mechanism, whereas Web Data can retain form-related information used by browser autofill. Therefore, the presence of autofill values does not imply that the password was stored in Chrome's password database. In this experiment, the absence of a Login Data record indicates that no saved password credential was recovered from the browser password database, while the Web Data finding provides evidence that form-related input was retained. This combination is consistent with a credential submission process in which the input was used on the phishing page without resulting in a recoverable saved-password record. However, the browser artifacts alone cannot establish the exact reason why Chrome did not create a password record, because password-saving behavior depends on the browser's password-management heuristics and user interaction.

3.2. Server and Network Evidence Correlation

Server-side and network evidence were examined to validate the browser findings through independent sources. The phishing server log confirmed the /login.html request, while Wireshark verified DNS resolution, IP communication, TCP/TLS communication, and TLS SNI for the phishing domain. Table 4 summarizes the evidence used to link victim-side browser activity with the phishing infrastructure.

Table 4. Evidence Collection and Acquisition Records

Evidence	Source	Value Found	Reconstruction Meaning
Phishing page request	Server access log	GET /login.html, referrer https://xn--dscord-wwaa.online/	Confirms fake login page access

Evidence	Source	Value Found	Reconstruction Meaning
Client access metadata	Server access log	08/Jun/2026 21:39:03 +0700; GET /login.html	Records client request details
DNS request	Wireshark PCAP	xn--dscord-wwaa.online	Confirms phishing domain resolution
IP communication	Wireshark PCAP	103.196.155.224:443	Confirms HTTPS communication with phishing server
TLS SNI	Wireshark PCAP	xn--dscord-wwaa.online	Verifies encrypted connection target
Discord service traffic	Wireshark PCAP	discord.com, cdn.discordapp.com, gateway.discord.gg	Indicates Discord Web activity

As shown in Table 4, the server and network findings corroborated the browser evidence from outside the browser storage layer. The server record confirmed access to the fake login page, while DNS, IP communication, and TLS SNI verified communication with the phishing infrastructure. This cross-layer validation reduced dependence on browser artifacts alone and strengthened the reconstructed account takeover sequence.

Network evidence provided an independent layer of attribution because the HTTPS payload itself was encrypted and therefore did not directly expose the contents of the phishing form. In this condition, TLS Server Name Indication (SNI) provided observable metadata indicating the hostname targeted during the TLS connection. The appearance of xn--dscord-wwaa.online in the PCAP therefore strengthened the attribution of the encrypted connection to the phishing infrastructure identified in the browser evidence. The Punycode representation was particularly relevant because the visually deceptive domain dīscord.online was represented at the protocol level as xn--dscord-wwaa.online. The combination of browser history, DNS resolution, IP communication, and TLS SNI therefore provided stronger domain attribution than browser history alone.

3.3. Timeline Reconstruction and Evidence Corroboration

Confirmed events from browser artifacts, phishing server records, Wireshark PCAP

findings, and account activity documentation were arranged chronologically to assess whether the evidence supported a coherent account takeover sequence. Figure 3 shows the corroboration profile of each incident stage based on its supporting evidence items or source layers.

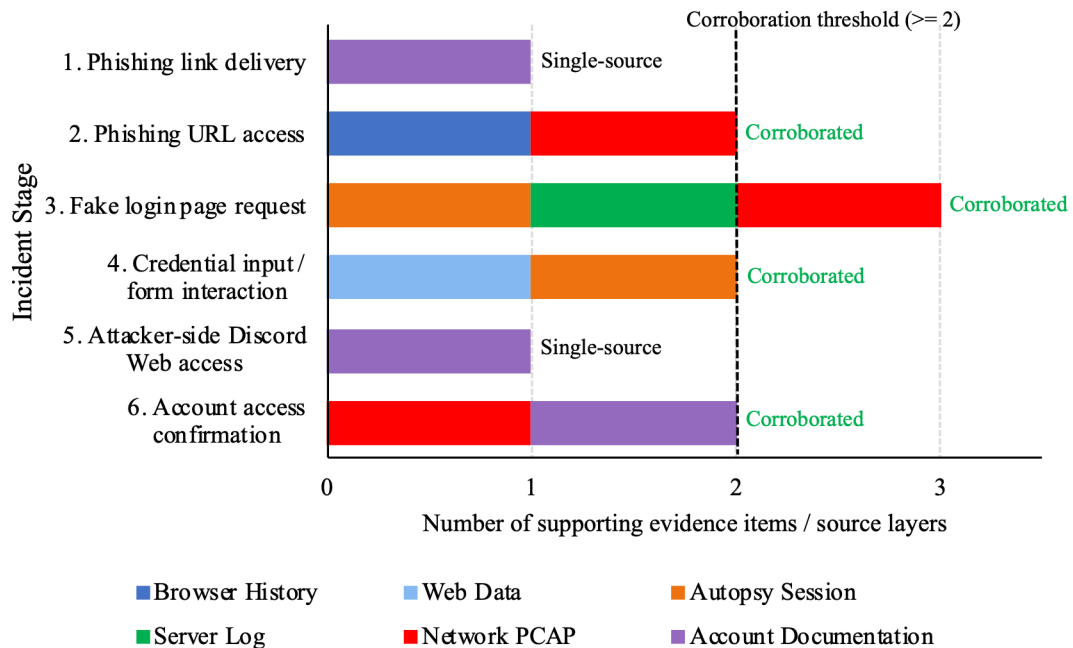


Figure 3. Evidence Corroboration by Incident Stage in Reconstructed Discord Web Account Takeover

As shown in Figure 3, the reconstructed incident stages were supported by varying levels of evidential strength. Phishing URL access, fake login page requests, credential input or form interactions, and account access confirmation were each supported by at least 2 evidence items or source layers. In contrast, phishing link delivery and attacker-side Discord Web access remained single-source stages because they relied solely on account documentation. The strongest corroboration was observed at the stage of the fake login page request, supported by Autopsy session traces, server log evidence, and network PCAP findings. Based on these results, the Evidence Corroboration Score was calculated as $ECS = (4/6) \times 100 = 66.7\%$. This result indicates that most stages in the reconstructed account takeover sequence were supported by cross-source or cross-artifact evidence rather than by a single artifact category.

The ECS of 66.7% indicates that four of the six reconstructed incident stages were supported by at least two evidence items or source layers, while two stages remained

single-source. The first stage, phishing link delivery, could not be independently corroborated because the available evidence did not preserve a second source showing how the phishing link was delivered to the victim. Likewise, the fifth stage, attacker-side Discord Web access, relied on account activity documentation without an independent victim-side or network evidence source directly establishing the attacker's access event. These gaps illustrate an inherent boundary of endpoint-centered forensic reconstruction: evidence generated outside the examined endpoint may not be preserved in the victim environment. Consequently, the absence of corroboration for these stages should not be interpreted as evidence that the events did not occur, but rather as a limitation in the availability and preservation of independent evidence needed to validate those stages.

3.4. Artifact Recovery and Source Contribution Analysis

Artifact recovery was evaluated to measure the completeness of the reconstructed evidence across browser, server, network, and account activity layers. Figure 4 presents a combined view of expected, recovered, and source-contribution artifacts across the evidence layers examined in this study.

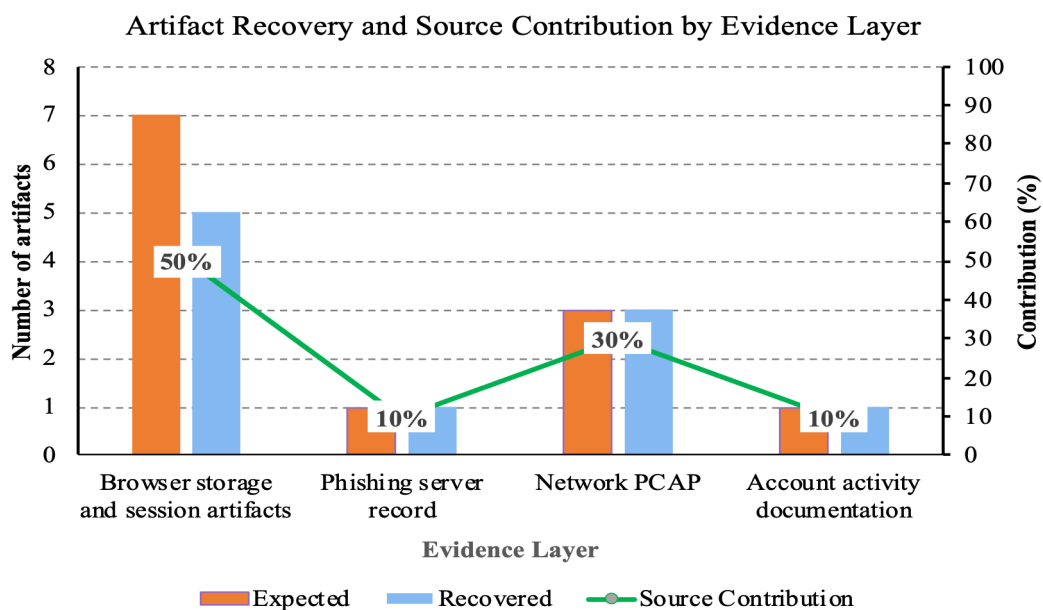


Figure 4. Artifact Recovery and Source Contribution by Evidence Layer

As shown in Figure 4, the browser storage and session artifact layer produced the most recovered artifacts, with five of the seven expected items recovered. Network PCAP recovered all three expected items, while the phishing server record and account activity documentation each recovered one expected item. Based on the recovered evidence, 10

of 12 expected artifact items were recovered. Therefore, the Artifact Recovery Rate was calculated as $ARR = (10/12) \times 100\% = 83.3\%$.

The two unrecovered artifacts were the phishing-domain cookie and the saved browser credential. These negative findings were not treated as failed evidence; rather, they provided a forensic context for interpreting the credential phishing sequence. The absence of a phishing-domain cookie indicates that the fake login page did not create a persistent browser-side cookie. In contrast, the absence of saved browser credentials shows that Chrome did not store the submitted input in the password database. Consequently, credential input was supported primarily by Web Data autofill traces and Autopsy session artifacts.

Source contribution analysis showed that browser storage and session artifacts accounted for 50% of all recovered artifacts. Network PCAP contributed 30%, while phishing server records and account activity documentation each contributed 10%. Although browser evidence provided the largest artifact yield, it did not independently establish the complete account takeover sequence. Browser History confirmed access to a phishing URL, while Web Data and Autopsy session artifacts supported credential entry or form interaction. Server and network evidence validated request activity and communication with the phishing infrastructure, while account activity documentation supported attacker-side access and account confirmation.

3.5. Threats to Validity and Technical Limitations

This study has several technical limitations that may affect the completeness of the forensic reconstruction. First, volatile RAM acquisition was not performed, so transient evidence such as active session states or other memory-resident traces was not preserved. Second, the experiment was conducted using a standard browsing session and did not evaluate private browsing modes such as Incognito, which may affect the availability and persistence of browser artifacts. Third, anti-forensic actions such as browser history deletion, profile removal, or log wiping were not simulated. Such conditions may reduce artifact availability and limit cross-source corroboration. Therefore, the reported ARR and ECS values should be interpreted as case-specific indicators for the controlled experimental environment rather than as universal measures of browser-mediated phishing investigations. Further validation under volatile-

memory, private-browsing, and anti-forensic conditions is required to assess the robustness of the reconstruction approach.

3.6. Incident Response and Forensic Governance Implications

The negative findings also provide diagnostic value for incident response. The absence of a saved browser credential in Login Data and the absence of a phishing-domain cookie indicate that the recovered evidence did not originate from persistent browser password or cookie storage. Instead, the presence of form-related values in Web Data and session traces in Autopsy provides stronger support for an interaction in which credential data were entered into the phishing page and subsequently transmitted to the phishing infrastructure. Therefore, negative findings should not be treated solely as missing evidence, but should be evaluated together with positive findings to determine the most plausible evidence-generation process.

From an incident response perspective, the findings demonstrate the need to integrate endpoint telemetry with network and server-side evidence. Browser history and local artifacts may establish user interaction with a phishing domain, but they may be incomplete when browser history is cleared, profiles are removed, or session information is no longer available. Network DNS and TLS metadata and phishing server request records can provide independent evidence of communication with the phishing infrastructure and therefore help maintain the continuity of the incident timeline. For enterprise security operations centers (SOCs), correlating endpoint, DNS, proxy, and server telemetry can reduce dependence on a single evidence source during phishing investigations.

For forensic examiners and incident responders, the reconstructed evidence model suggests three practical measures. First, automated cross-source timeline correlation can be used to align browser, network, server, and account activity timestamps during incident triage. Second, phishing-resistant authentication mechanisms such as FIDO2/WebAuthn can reduce the risk associated with credential-based phishing because authentication is not solely dependent on reusable passwords. Third, ARR and ECS can be used as supplementary reporting indicators to describe artifact recovery completeness and the degree of cross-source corroboration, provided that their definitions and calculation rules are stated explicitly for each investigation.

4. CONCLUSION

This study reconstructed a web-based social media account takeover, specifically a Discord Web credential phishing incident, by correlating browser artifacts, phishing server logs, network traffic captures, and account activity records into a causally ordered incident timeline. The reconstruction demonstrated that no single evidence source was sufficient to establish the complete account takeover sequence and that multi-source correlation was necessary to connect the incident stages with verifiable evidence. The analysis recovered 10 of 12 expected artifacts, resulting in an Artifact Recovery Rate (ARR) of 83.3%, while 4 of 6 reconstructed incident stages were supported by at least two evidence items or source layers, producing an Evidence Corroboration Score (ECS) of 66.7%. Browser storage and session artifacts provided the largest artifact contribution, while network PCAP, phishing server records, and account activity documentation supplied independent evidence for communication, request activity, and attacker-side account access. The absence of a phishing-domain cookie and saved browser credential also provided useful forensic context by indicating that the recovered evidence was not derived from persistent browser-side cookie or password storage. These findings demonstrate that multi-source correlation is essential for reconstructing browser-mediated credential phishing incidents, particularly when individual evidence sources are incomplete. From an operational perspective, the findings support cross-source timeline correlation during incident response and the use of phishing-resistant authentication mechanisms such as FIDO2/WebAuthn. ARR and ECS may also be used as supplementary indicators for reporting artifact completeness and cross-source corroboration when their definitions are explicitly stated. However, because the study used a controlled experimental environment without volatile-memory acquisition, private-browsing conditions, or simulated anti-forensic activity, the findings should be further validated under broader forensic conditions.

ACKNOWLEDGMENT

The author expresses gratitude to the Directorate of Research and Community Service, Ministry of Higher Education, Science, and Technology of the Republic of Indonesia, for the financial support provided through the Master's Thesis Research scheme with contract number 181/KPA/C4/KPT/2026.

REFERENCES

- [1] G. Kaur *et al.*, "Social Media in the Digital Age: A Comprehensive Review of Impacts, Challenges and Cybercrime," *Engineering Proceedings*, vol. 62, no. 1, 2024, doi: 10.3390/engproc2024062006.
- [2] M. F. Safitra, M. Lubis, and H. Fakhurroja, "Counterattacking Cyber Threats: A Framework for the Future of Cybersecurity," *Sustainability*, vol. 15, no. 18, p. 13369, Sep. 2023, doi: 10.3390/su151813369.
- [3] Š. Grigaliūnas, R. Brūzgienė, and A. Venčkauskas, "Ontology-Driven Digital Profiling For Identification and Linking Evidence Across Social Media Platform," *IEEE Access*, vol. 11, pp. 111672–111691, 2023, doi: 10.1109/ACCESS.2023.3322162.
- [4] "Discord Users by Country 2026." Accessed: Jun. 11, 2026. [Online]. Available: <https://worldpopulationreview.com/country-rankings/discord-users-by-country>
- [5] F. Iqbal, M. Motylinski, and A. MacDermott, "Discord Server Forensics: Analysis and Extraction of Digital Evidence," in *2021 11th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, IEEE, Apr. 2021, pp. 1–8. doi: 10.1109/NTMS49979.2021.9432654.
- [6] M. Ahmed *et al.*, "PhishCatcher: Client-Side Defense Against Web Spoofing Attacks Using Machine Learning," *IEEE Access*, vol. 11, pp. 61249–61263, 2023, doi: 10.1109/ACCESS.2023.3287226.
- [7] S. Asiri, Y. Xiao, S. Alzahrani, S. Li, and T. Li, "A Survey of Intelligent Detection Designs of HTML URL Phishing Attacks," *IEEE Access*, vol. 11, pp. 6421–6443, 2023, doi: 10.1109/ACCESS.2023.3237798.
- [8] A. K. Jain and B. B. Gupta, "A survey of phishing attack techniques, defence mechanisms and open research challenges," *Enterp. Inf. Syst.*, vol. 16, no. 4, pp. 527–565, Apr. 2022, doi: 10.1080/17517575.2021.1896786.
- [9] S. Berham and S. Morris, "A critical comparison of Brave browser and Google Chrome forensic artefacts," *Journal of Digital Forensics, Security and Law*, vol. 17, no. 1, Art no. 4, 2022, doi: 10.15394/jdfsl.2022.1752.
- [10] P. Kumari, K. Shukla, and K. Modi, "Reconstructing user activity through browser forensics by examining techniques and ethical considerations," *Journal of Digital Security and Forensics*, vol. 2, no. 2, pp. 111–123, Dec. 2025, doi: 10.29121/digisecforensics.v2.i2.2025.67.

- [11] A. Raza, M. Hussain, H. Tahir, M. Zeeshan, M. A. Raja, and K.-H. Jung, "Forensic analysis of web browsers lifecycle: A case study," *Journal of Information Security and Applications*, vol. 85, p. 103839, Sep. 2024, doi: 10.1016/j.jisa.2024.103839.
- [12] D. Komosny, "Evidential value of country location evidence obtained from IP address geolocation," *PeerJ Comput. Sci.*, vol. 9, p. e1305, Mar. 2023, doi: 10.7717/peerj-cs.1305.
- [13] A. Yudhana, I. Riadi, and S. Suharti, "Network Forensics Against Volumetric-Based Distributed Denial of Service Attacks on Cloud and the Edge Computing," *International Journal of Safety and Security Engineering*, vol. 12, no. 5, pp. 577–588, Oct. 2022, doi: 10.18280/ijssse.120505.
- [14] R. R. Chand, J. Kumar, A. Prasad, and S. Chand, "Effectiveness of Forensic Tools in Extracting Web Browser Artifacts," in *2025 International Conference on Sustainable Technology and Engineering (i-COSTE)*, IEEE, Dec. 2025, pp. 1–5. doi: 10.1109/i-COSTE68047.2025.11467264.
- [15] L. Rzayeva, A. Zhetpisbayeva, A. Batkuldin, N. Nyssanov, A. Ryzhova, and F. Saeed, "An Intelligent Browser History Forensics Method for Automated Analysis of Web Activity Logs, Credentials, and User Behavioral Profiles," *Algorithms*, vol. 19, no. 1, p. 75, Jan. 2026, doi: 10.3390/a19010075.
- [16] Z. Morić, V. Dakić, and I. Ogrizek Biškupić, "An Empirical Assessment of Digital Forensic Process Reliability Using Integrated ISO/IEC 27037 and 27041 Standards," *Journal of Cybersecurity and Privacy*, vol. 6, no. 2, p. 57, Apr. 2026, doi: 10.3390/jcp6020057.
- [17] A. M. Almuhaideb *et al.*, "Homoglyph Attack Detection Model Using Machine Learning and Hash Function," *Journal of Sensor and Actuator Networks*, vol. 11, no. 3, p. 54, Sep. 2022, doi: 10.3390/jsan11030054.
- [18] R. R. Chand, N. A. Sharma, and M. A. Kabir, "Advancing Web Browser Forensics: Critical Evaluation of Emerging Tools and Techniques," *SN Comput. Sci.*, vol. 6, no. 4, Art. No. 355, Apr. 2025, doi: 10.1007/s42979-025-03921-6.
- [19] M. S. Javed *et al.*, "Analyzing Tor Browser Artifacts for Enhanced Web Forensics, Anonymity, Cybersecurity, and Privacy in Windows-Based Systems," *Information*, vol. 15, no. 8, p. 495, Aug. 2024, doi: 10.3390/info15080495.
- [20] A. Yudhana, Z. Y. Rivai, and I. Riadi, "Assesing Digital Evidence Availability in Discord Phishing using ISO/IEC 27037 and Anti-Forensics Analysis," *International Journal of*

- Advances in Data and Information Systems*, vol. 7, no. 1, pp. 402–413, Apr. 2026, doi: 10.59395/ijadis.v7i1.1518.
- [21] S. Bhandari and V. Jusas, "An Abstraction Based Approach for Reconstruction of TimeLine in Digital Forensics," *Symmetry (Basel)*, vol. 12, no. 1, p. 104, Jan. 2020, doi: 10.3390/sym12010104.
 - [22] F. Breitingner, H. Studiawan, and C. Hargreaves, "SoK: Timeline based event reconstruction for digital forensics: Terminology, methodology, and current challenges," *Forensic Science International: Digital Investigation*, vol. 53, p. 301932, Jul. 2025, doi: 10.1016/j.fsidi.2025.301932.
 - [23] S. Pathirana, R. Dushmantha, C. Jayarathna, B. Thilakarathna, A. Senarathna, and A. Wijesooriya, "Next-Gen Mobile Forensics Tool: Timeline Reconstruction from Android Digital Evidence," in *2026 8th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (ICHORA)*, IEEE, May 2026, pp. 1–10. doi: 10.1109/ICHORA69329.2026.11536997.
 - [24] H. Studiawan, I. M. T. Pardede, A. P. Sayakkara, B. A. Pratomo, and A. M. Shiddiqi, "Event Reconstruction From Forensic Timeline with Formal Event Modeling," in *2026 International Conference on AI Innovations and Industry (ICAIII)*, IEEE, Apr. 2026, pp. 1–6. doi: 10.1109/ICAIII69475.2026.11521234.
 - [25] A. M. Alotibi, S. Y. Altaleedi, T. Zia, and E. U. H. Qazi, "Examining the Behavior of Web Browsers Using Popular Forensic Tools," *International Journal of Digital Crime and Forensics*, vol. 16, no. 1, pp. 1–22, Jul. 2024, doi: 10.4018/IJDCF.349218.
 - [26] R. Adila, H. Studiawan, and F. Breitingner, "Forensic Event Reconstruction With Process Mining," *IEEE Access*, vol. 14, pp. 18964–18985, 2026, doi: 10.1109/ACCESS.2026.3660165.