securitum

Security Report

SUBJECT

DuckDuckGo VPN No-Log Policy Audit

DATE

15.10.2025 – 19.01.2026

LOCATION

Gdansk, Poland
Cracow, Poland

AUTHORS

Maciej Szymczak
Marek Rzepecki

VERSION

1.0

securitum

Executive Summary

This report summarizes the findings of an independent security assessment conducted by Securitum, commissioned by DuckDuckGo. The primary objective of this engagement was to validate that DuckDuckGo's VPN server infrastructure strictly adheres to its publicly stated No-Logs policy. This policy asserts that no user activity, connection timestamps, or identifying metadata is logged or stored on the egress infrastructure, thereby protecting user privacy and anonymity.

To achieve this objective, Securitum dispatched two senior security consultants to engage directly with the DuckDuckGo Engineering Team. The assessment occurred between October 15, 2025, and January 18, 2026, constituting twelve person-days of focused technical evaluation. Throughout the engagement, Securitum auditors worked with DuckDuckGo's senior engineers responsible for the infrastructure.

The assessment involved a deep-dive technical inspection, source code review of proprietary components, and live system analysis to verify that the deployed environment contains no mechanisms capable of collecting or retaining user-identifiable data.

Engagement Scope and Methodology

The assessment was designed to provide a whitebox/clearbox validation of DuckDuckGo VPN's privacy claims. Unlike with a more limited closed box audit, Securitum's methodology combined documentation review, source code analysis, process evaluation, and direct, hands-on inspection of production systems.

It should be noted that throughout the engagement, the DuckDuckGo engineering team provided full transparency, granting auditors access to private source code repositories, architecture diagrams, and an indirect (managed by the DDG Team) root-level access to live production servers.

The scope of the audit covered:

1. The Controller Infrastructure - hosted in Azure, responsible for authentication and peer orchestration.
2. The Egress Infrastructure - bare-metal servers hosted by third-party providers (i3D, DataPacket) running the WireGuard protocol.
3. The "Wedge" Software - the proprietary daemon responsible for managing VPN interfaces.

Audit Activities

The following activities were performed to achieve the engagement's objectives:

Documentation Review and Technical Interviews

Analysis of the "DuckDuckGo" architecture, including the separation of the Subscription API from the VPN Controller. This was supplemented by in-depth discussions with the Site Reliability Engineering (SRE) and Backend teams to understand data flows and failover mechanisms.

Source Code and Configuration Analysis

Unlike standard *black-box* audits, this engagement included a review of the source code for the "Wedge" internal VPN software and the "Scam Blocker" implementation. Auditors verified that the logic governing traffic handling does not contain instructions to log user activity.

Live System Inspection

Direct, hands-on examination of production egress servers. Securitum auditors independently and randomly selected production servers for verification. The analysis focused on file system integrity and running processes to identify any mechanisms capable of retaining user data.

Change and Deployment Process Review

Evaluation of the operational security practices governing configuration changes via Ansible and Chef. This verified that processes are in place to prevent the unauthorized introduction of logging mechanisms.

Key Areas of Investigation

The audit was structured to answer critical questions derived directly from the assertions made in the No-Logs policy:

1. User Activity Tracking - verification that user activity is not tracked or logged on production egress servers.
2. Metadata Logging - verification that connection metadata, such as DNS traffic, is not logged.
3. Network Traffic Inspection - verification that user network traffic is not inspected or logged.
4. Service Monitoring - verification that services a user connects to are not monitored.
5. Server Configurations - verification that the VPN uses only dedicated, bare-metal servers.
6. Policy Uniformity - verification that the No-Logs policy is applied uniformly across all regions.
7. Change Management - verification of a formal process with dual control for logging configurations.
8. Configuration Files - verification that active configuration files have no logging enabled.
9. User Authentication - verification that authentication tokens are separated from VPN usage data.

Contents

Change history

Document date	Version	Change description
20.03.2026	1.0	The final version of the report.

Detailed Findings

Detailed Findings

This section presents a comprehensive analysis of the privacy and security claims made by DuckDuckGo. The findings below are the result of a direct inspection of DuckDuckGo's production infrastructure, source code, and operational procedures.

DuckDuckGo VPN does not track or log user activity on its egress servers.

Status: Confirmed.

Securitum confirmed that DuckDuckGo does not track or log user activity on its egress servers. Auditors performed a forensic review of random live egress servers. This included searching for monitoring processes. No evidence of activity tracking was found. The verification of the entire connection process, from authentication to traffic redirection, demonstrated a design explicitly focused on stateless forwarding rather than data retention.

DuckDuckGo does not log user-attributable connection metadata, such as DNS traffic.

Status: Confirmed.

User-attributable connection metadata is not logged. The audit confirmed that DuckDuckGo utilizes an internal DNS Resolver for DNS queries on egress servers. While the system uses a caching mechanism to improve performance, the Time-to-Live (TTL) is set to a standard 24 hours (86400 seconds), after which data is purged.

Auditors verified that this cache exists in memory and is not written to persistent logs that could be analyzed post-mortem. Searches for raw log files on the egress servers yielded no metadata that could be linked to individual users or their specific activities. The risk of correlating a domain to a specific user is assessed as negligible due to the high volume of shared traffic and the lack of user identifiers in the DNS requests.

DuckDuckGo VPN does not inspect or log user network traffic on its VPN servers.

Status: Confirmed.

The audit confirmed that DuckDuckGo VPN does not inspect or log the payload of network traffic. The solution is fully WireGuard-based, and the server-side software stack includes custom builds of Nginx. Technical analysis of these Nginx configurations demonstrated that they are manually hardened and stripped of standard logging directives.

Specific attention was paid to the "Scam Blocker" feature. It was verified that threat detection primarily occurs locally on the client device. When server-side verification is required, the client sends only a partial 4-character hash prefix of the domain. This architectural decision ensures that DuckDuckGo servers cannot reconstruct the full URL or associate the request with a specific user's browsing history.

Information about services (websites, servers) a user connects to is not monitored or logged.

Status: Confirmed.

DuckDuckGo does not log or monitor the specific services, websites, or servers that users connect to. The infrastructure acts as a transparent conduit. Securitum recommended enhanced file integrity to actively detect unauthorized configuration changes and regular rebuild and cleanup procedures to enforce a minimal footprint. DuckDuckGo acknowledged this finding and has already implemented stricter monitoring and cleanup procedures.

DuckDuckGo VPN only uses dedicated servers which are not shared with any other businesses or service providers.

Status: Confirmed.

Auditors validated that the egress infrastructure consists of bare-metal servers hosted by third-party providers (i3D, DataPacket) that are dedicated solely to DuckDuckGo. These servers are not virtualized slices shared with other tenants. This dedicated tenancy ensures complete control over the kernel and system-level configurations, preventing external interference or side-channel data leakage from other customers of the hosting provider.

The No-Logs policy is applied uniformly across all servers and geographic regions.

Status: Confirmed.

It has been verified that a consistent server configuration is deployed across DuckDuckGo's global infrastructure. Securitum auditors analyzed the Ansible and Chef playbooks used for server deployment, confirming that every egress server—whether in the US, Europe, or elsewhere—receives an identical, hardened configuration image. Auditors manually compared the active state of servers from different geographic regions and found zero deviation in privacy settings or logging configurations.

DuckDuckGo enforces a formal Change Management process that requires dual control for changes to log-related configurations.

Status: Confirmed.

A formal Change Management process is enforced. Changes to the infrastructure code require a Pull Request (PR) review and approval within GitHub before merging. Furthermore, the deployment of the critical "Wedge" software is managed via GitHub Actions pipelines that mandate explicit approval from a restricted `deploy` group. This ensures that no single engineer can unilaterally alter logging configurations or push unapproved code to the production fleet. Auditors noted that the group of users with potential write access to repositories was broad. It was recommended to enforce stricter multi-factor authentication requirements and cross-checks for repository actions to mitigate potential insider threats.

Active VPN configuration files do not have logging directives enabled.

Status: Confirmed.

A direct, observed review of the server-side configuration files confirmed that no logging directives are enabled. Auditors inspected the configurations for WireGuard interfaces, the DNS Resolver, and system-level daemons (`systemd`/`journald`). No parameters were found that would enable persistent logging of user traffic, connection metadata, or IP addresses.

DuckDuckGo VPN and Subscription APIs use separate authentication tokens to authorize accounts that are not connected to an individual user or their VPN connection.

Status: Confirmed.

The system is architected to explicitly separate user authentication from VPN usage. The audit analyzed the "Registration Controller" flow:

1. Token Separation - the user authenticates with a Subscription API to obtain a signed token. This token is legally and technically separate from the user's identity.
2. Connection Setup - the client presents this token to the Controller (hosted in Azure). The Controller assigns the client to an egress server based on load and location.
3. Ephemeral Data - the Controller passes only the client's ephemeral WireGuard public key to the egress server.

This process does not create or retain a persistent database mapping a specific user account to a specific egress server or connection timestamp. Once the session ends, the ephemeral state is cleared.

Conclusion

Securitum has completed its independent third-party assessment of the DuckDuckGo VPN infrastructure and its adherence to its publicly stated No-Logs policy. The engagement involved a direct review of production systems, source code, and architectural documentation.

The technical evidence reviewed showed no instances of user activity logging, connection metadata storage, or network traffic inspection that would contradict the No-Logs policy. Furthermore, the architecture's strict separation of the authentication layer from the data plane provides strong privacy assurance.

Based on these findings, Securitum attests that the DuckDuckGo VPN service, as configured at the time of the audit, fully complies with the privacy commitments outlined in its No-Logs policy.