Security changelog

Ask about this page

Fixed in ClickHouse v25.1.5.5, 2025-01-05

CVE-2025-1385

When the library bridge feature is enabled, the clickhouse-library-bridge exposes an HTTP API on localhost. This allows clickhouse-server to dynamically load a library from a specified path and execute it in an isolated process. Combined with the ClickHouse table engine functionality that permits file uploads to specific directories, a misconfigured server can be exploited by an attacker with privileges to access both table engines, allowing them to execute arbitrary code on the ClickHouse server.

Fix has been pushed to the following open-source versions: v24.3.18.6, v24.8.14.27, v24.11.5.34, v24.12.5.65, v25.1.5.5

ClickHouse Cloud is unaffected by this vulnerability.

Credits: Arseniy Dugin

Fixed in ClickHouse v24.5, 2024-08-01

CVE-2024-6873

It is possible to redirect the execution flow of the ClickHouse server process from an unauthenticated vector by sending a specially crafted request to the ClickHouse server native interface. This redirection is limited to what is available within a 256-byte range of memory at the time of execution. This vulnerability was identified through our Bugbounty program and

v24.4.2.141-stable, v24.3.1.1705, v24.0.1.4425-stable

ClickHouse Cloud uses different versioning and a fix for this vulnerability was applied to all instances running v24.2 onward.

Credits: malacupa (Independent researcher)

Fixed in ClickHouse v24.1, 2024-01-30

CVE-2024-22412

When toggling between user roles while using ClickHouse with query cache enabled, there is a risk of obtaining inaccurate data. ClickHouse advises users with vulnerable versions of ClickHouse not to use the query cache when their application dynamically switches between various roles.

Fix has been pushed to the following open-source versions: v24.1.1.2048, v24.1.8.22-stable, v23.12.6.19-stable, v23.8.12.13-lts, v23.3.22.3-lts

ClickHouse Cloud uses different versioning and a fix for this vulnerability was applied at v24.0.2.54535.

Credits: Evan Johnson and Alan Braithwaite from Runreveal team - More information can be found on their blog post.

Fixed in ClickHouse v23.10.5.20, 2023-11-26

CVE-2023-47118

A heap buffer overflow vulnerability affecting the native interface running by default on port 9000/tcp. An attacker, by triggering a bug in the T64 compression codec, can cause the

v23.3.16.7

ClickHouse Cloud uses different versioning and a fix for this vulnerability was applied at v23.9.2.47475.

Credits: malacupa (Independent researcher)

CVE-2023-48298

An integer underflow vulnerability in the FPC compressions codec. An attacker can use it to cause the ClickHouse server process to crash. This vulnerability can be exploited without the need to authenticate.

Fix has been pushed to the following open-source versions: v23.10.4.25, v23.9.5.29, v23.8.7.24, v23.3.17.13.

ClickHouse Cloud uses different versioning and a fix for this vulnerability was applied at v23.9.2.47475.

Credits: malacupa (Independent researcher)

CVE-2023-48704

A heap buffer overflow vulnerability affecting the native interface running by default on port 9000/tcp. An attacker, by triggering a bug in the Gorilla codec, can cause the ClickHouse server process to crash. This vulnerability can be exploited without the need to authenticate.

Fix has been pushed to the following open-source versions: v23.10.5.20, v23.9.6.20, v23.8.8.20, v23.3.18.15.

ClickHouse Cloud uses different versioning and a fix for this vulnerability was applied at v23.9.2.47551.

Credits: malacupa (Independent researcher)

A heap buffer overflow issue was discovered in ClickHouse server. A malicious user with ability to load data into ClickHouse server could crash the ClickHouse server by inserting a malformed CapnProto object.

Fix has been pushed to version 22.9.1.2603, 22.8.2.11, 22.7.4.16, 22.6.6.16, 22.3.12.19

Credits: Kiojj (independent researcher)

CVE-2022-44010

A heap buffer overflow issue was discovered in ClickHouse server. An attacker could send a specially crafted HTTP request to the HTTP Endpoint (listening on port 8123 by default), causing a heap-based buffer overflow that crashes the ClickHouse server process. This attack does not require authentication.

Fix has been pushed to version 22.9.1.2603, 22.8.2.11, 22.7.4.16, 22.6.6.16, 22.3.12.19

Credits: Kiojj (independent researcher)

Fixed in ClickHouse 21.10.2.15, 2021-10-18

CVE-2021-43304

Heap buffer overflow in ClickHouse's LZ4 compression codec when parsing a malicious query. There is no verification that the copy operations in the LZ4::decompressImpl loop and especially the arbitrary copy operation `wildCopy<copy_amount>(op, ip, copy_end)`, don't exceed the destination buffer's limits.

Credits: JFrog Security Research Team

CVE-2021-43305

vulnerable copy operation is in a different wildCopy call.

Credits: JFrog Security Research Team

CVE-2021-42387

Heap out-of-bounds read in ClickHouse's LZ4 compression codec when parsing a malicious query. As part of the LZ4::decompressImpl() loop, a 16-bit unsigned user-supplied value ('offset') is read from the compressed data. The offset is later used in the length of a copy operation, without checking the upper bounds of the source of the copy operation.

Credits: JFrog Security Research Team

CVE-2021-42388

Heap out-of-bounds read in ClickHouse's LZ4 compression codec when parsing a malicious query. As part of the LZ4::decompressImpl() loop, a 16-bit unsigned user-supplied value ('offset') is read from the compressed data. The offset is later used in the length of a copy operation, without checking the lower bounds of the source of the copy operation.

Credits: JFrog Security Research Team

CVE-2021-42389

Divide-by-zero in ClickHouse's Delta compression codec when parsing a malicious query. The first byte of the compressed buffer is used in a modulo operation without being checked for 0.

Credits: JFrog Security Research Team

CVE-2021-42390

Credits: JFrog Security Research Team

CVE-2021-42391

Divide-by-zero in ClickHouse's Gorilla compression codec when parsing a malicious query. The first byte of the compressed buffer is used in a modulo operation without being checked for 0.

Credits: JFrog Security Research Team

Fixed in ClickHouse 21.4.3.21, 2021-04-12

CVE-2021-25263

An attacker that has CREATE DICTIONARY privilege, can read arbitary file outside permitted directory.

Fix has been pushed to versions 20.8.18.32-lts, 21.1.9.41-stable, 21.2.9.41-stable, 21.3.6.55-lts, 21.4.3.21-stable and later.

Credits: Vyacheslav Egoshin

Fixed in ClickHouse Release 19.14.3.3, 2019-09-10

CVE-2019-15024

An attacker that has write access to ZooKeeper and who can run a custom server available from the network where ClickHouse runs, can create a custom-built malicious server that will act as a ClickHouse replica and register it in ZooKeeper. When another replica will fetch data part from the malicious replica, it can force clickhouse-server to write to arbitrary path on filesystem.

An OOB read, OOB write and integer underflow in decompression algorithms can be used to achieve RCE or DoS via native protocol.

Credits: Eldar Zaitov of Yandex Information Security Team

CVE-2019-16536

Stack overflow leading to DoS can be triggered by a malicious authenticated client.

Credits: Eldar Zaitov of Yandex Information Security Team

Fixed in ClickHouse Release 19.13.6.1, 2019-09-20

CVE-2019-18657

Table function `url` had the vulnerability allowed the attacker to inject arbitrary HTTP headers in the request.

Credits: Nikita Tikhomirov

Fixed in ClickHouse Release 18.12.13, 2018-09-10

CVE-2018-14672

Functions for loading CatBoost models allowed path traversal and reading arbitrary files through error messages.

Credits: Andrey Krasichkov of Yandex Information Security Team

Fixed in ClickHouse Release 18.10.3, 2018-08-13

CVE-2018-14671

Fixed in ClickHouse Release 1.1.54388, 2018-06-28

CVE-2018-14668

“remote” table function allowed arbitrary symbols in “user”, “password” and “default_database” fields which led to Cross Protocol Request Forgery Attacks.

Credits: Andrey Krasichkov of Yandex Information Security Team

Fixed in ClickHouse Release 1.1.54390, 2018-07-06

CVE-2018-14669

ClickHouse MySQL client had “LOAD DATA LOCAL INFILE” functionality enabled that allowed a malicious MySQL database read arbitrary files from the connected ClickHouse server.

Credits: Andrey Krasichkov and Evgeny Sidorov of Yandex Information Security Team

Fixed in ClickHouse Release 1.1.54131, 2017-01-10

CVE-2018-14670

Incorrect configuration in deb package could lead to the unauthorized use of the database.

Credits: the UK’s National Cyber Security Centre (NCSC)

Was this page helpful? Yes No Suggest edits Raise issue

ClickHouse Docs

Product

ClickHouse Cloud
Bring Your Own Cloud
Postgres managed by ClickHouse
Managed ClickStack
ClickHouse
ClickStack
Agentic Data Stack
ClickHouse Government
ClickHouse Keeper
ClickPipes
Integrations
chDB
Pricing

Resources

Documentation
Trust Center
Training
Support
Benchmarks
Use Cases
Videos
Demos
Presentations
Real-time Data Warehouse
Engineering Resources

Company

Blog
Our Story
Careers
Contact Us
Events
News
Media

Join our community

GitHub
Slack
LinkedIn
X
Bluesky
Telegram
Meetup

Comparisons

BigQuery
PostgreSQL
Redshift

ClickHouse Docs

Partners

AWS

Azure

ClickHouse

Star us on Github

© 2026 ClickHouse, Inc. HQ in the Bay Area, CA and Amsterdam, NL.

Trademark Privacy Security Legal Cookie Policy