

deepseek Data Breach: **What Could Have Prevented It?**

San Tran

Jan, 2025

||||· ClickHouse

Speakers



San Tran

Pentester|Application|Product Security dude

01

DeepSeek Data Breach case study.



||||· ClickHouse

DeepSeek Data Breach

Consequences of data breach:

- Reputation damage
- Financial loss
- Legal and Regulatory Consequences
- Loss of Customer Trust
- Competitive Disadvantage



More info: <https://www.wiz.io/blog/wiz-research-uncovers-exposed-deepseek-database-leak>

Wiz Research Uncovers Exposed DeepSeek Database Leaking Sensitive Information, Including Chat History

A publicly accessible database belonging to DeepSeek allowed full control over database operations, including the ability to access internal data. The exposure includes over a million lines of log streams with highly sensitive information.



Gal Nagli

January 29, 2025

3 minute read



Table of contents

- Executive Summary
- Exposure Walkthrough
- Key Takeaways
- Conclusion

Wiz Research has identified a publicly accessible **ClickHouse** database belonging to DeepSeek, which allows full control over database operations, including the ability to access internal data. The exposure includes over a million lines of log streams containing chat history, secret keys, backend details, and other highly sensitive information. The Wiz Research team immediately and responsibly disclosed the issue to DeepSeek, which promptly secured the exposure.

In this blog post, we will detail our discovery and also consider the broader implications for the industry at large.



Top page on Hackernews



Wiz Research Uncovers Exposed DeepSeek Database Leaking Sensitive Information, Including Chat History

A publicly accessible database belonging to DeepSeek allowed full control over database operations, including the ability to

information.



Table of contents

Executive Summary

Exposure Walkthrough

Key Takeaways

Conclusion

▲ danielodieovich 19 days ago | prev | next [-]

open exposed clickhouse is this decade's open exposed elasticsearch so common in the past

4 points by ebfe1 18 days ago | parent | next [2 more]

▲ bearjaws 19 days ago | parent | prev | next [-]

Which was originally the open exposed mongo server, then mysql/phpmyadmin, then exposed ftp, and then exposed telnet.

▲ hmmm-i-wonder 18 days ago | root | parent | next [-]

We move on and upwards, but never really stop making the same mistakes do we.

▲ astrea 19 days ago | parent | prev | next [-]

Shows how old I am. Thought we were still in the "exposed ElasticSearch" era.

▲ kdmctcl 19 days ago | root | parent | next [-]

I was sure this was Elastic, you are not alone.

▲ blitzar 18 days ago | parent | prev | next [-]

open exposed S3 bucket is this decade's open exposed S3 bucket so common in the past

In this blog post, we will detail our discovery and also consider the broader implications for the industry at large.



More info: <https://news.ycombinator.com/item?id=42871371>



Hackernews...

Wiz Research Uncovers Exposed DeepSeek Database Leaking Sensitive Information. Including Chat History

▲ nyclounge 19 days ago | parent | prev | next [-]

Why is ClickHouse exposing unauthenticated database access at port 9000 to the public? Is this the default behavior or did DeepSeek open it up for dev purposes?

AlexClickHouse 19 days ago | root | parent | next [4 more]

▲ ceejayoz 19 days ago | root | parent | prev | next [-]

That used to be the default setup for Redis, too. Might still be. You aren't supposed to have it on a public subnet.

▲ SahAssar 19 days ago | root | parent | next [-]

> You aren't supposed to have it on a public subnet.

That's an incredibly bad assumption. To have defaults assume that you are on a protected network (what does that even mean? like what permissions are assumed just because you are on the same network? admin?) is just bad practice.

▲ ceejayoz 19 days ago | root | parent | next [-]

Private networking for internal things like databases has been the standard best practice for a long, long time.

▲ SahAssar 18 days ago | root | parent | next [-]

Safe default configuration has been the standard practice for even longer.

▲ ceejayoz 18 days ago | root | parent | next [-]

I'm all for both.



backend details, and other highly sensitive information. I
and responsibly disclosed the issue to DeepSeek, which

In this blog post, we will detail our discovery and also cor
industry at large.



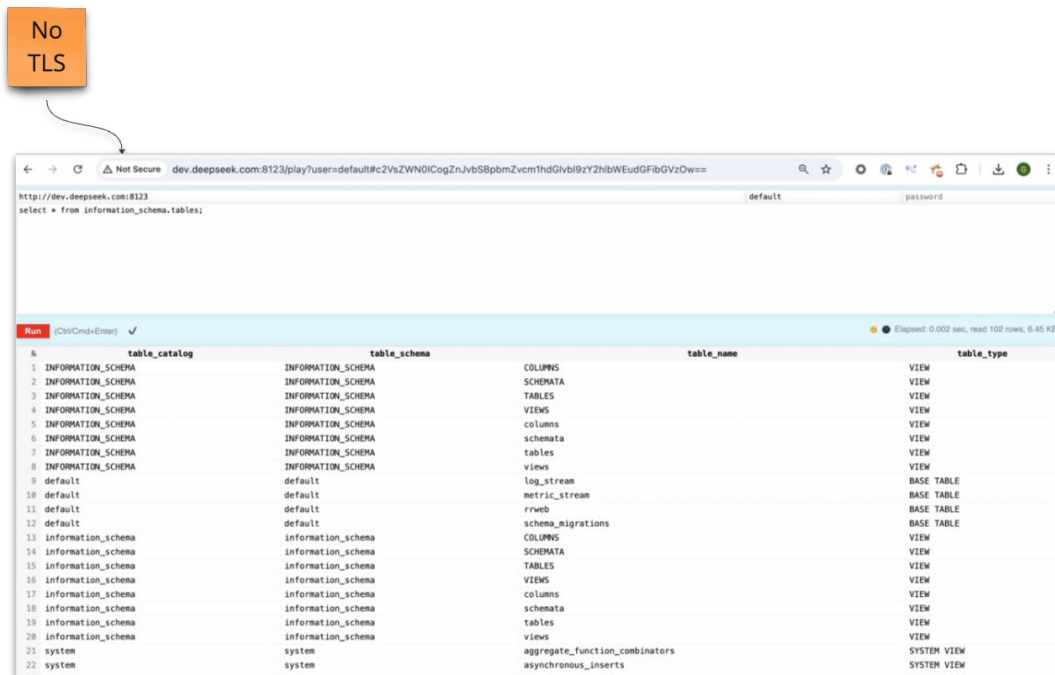
More info: <https://news.ycombinator.com/item?id=42871371>

02

What went wrong?

Open access

No
TLS



#	table_catalog	table_schema	table_name	table_type
1	INFORMATION_SCHEMA	INFORMATION_SCHEMA	COLUMNS	VIEW
2	INFORMATION_SCHEMA	INFORMATION_SCHEMA	SCHEMATA	VIEW
3	INFORMATION_SCHEMA	INFORMATION_SCHEMA	TABLES	VIEW
4	INFORMATION_SCHEMA	INFORMATION_SCHEMA	VIEWS	VIEW
5	INFORMATION_SCHEMA	INFORMATION_SCHEMA	columns	VIEW
6	INFORMATION_SCHEMA	INFORMATION_SCHEMA	schemata	VIEW
7	INFORMATION_SCHEMA	INFORMATION_SCHEMA	tables	VIEW
8	INFORMATION_SCHEMA	INFORMATION_SCHEMA	views	VIEW
9	default	default	log_stream	BASE TABLE
10	default	default	metric_stream	BASE TABLE
11	default	default	rweb	BASE TABLE
12	default	default	schema_migrations	BASE TABLE
13	information_schema	information_schema	COLUMNS	VIEW
14	information_schema	information_schema	SCHEMATA	VIEW
15	information_schema	information_schema	TABLES	VIEW
16	information_schema	information_schema	VIEWS	VIEW
17	information_schema	information_schema	columns	VIEW
18	information_schema	information_schema	schemata	VIEW
19	information_schema	information_schema	tables	VIEW
20	information_schema	information_schema	views	VIEW
21	system	system	aggregate_function_combinators	SYSTEM VIEW
22	system	system	asynchronous_inserts	SYSTEM VIEW



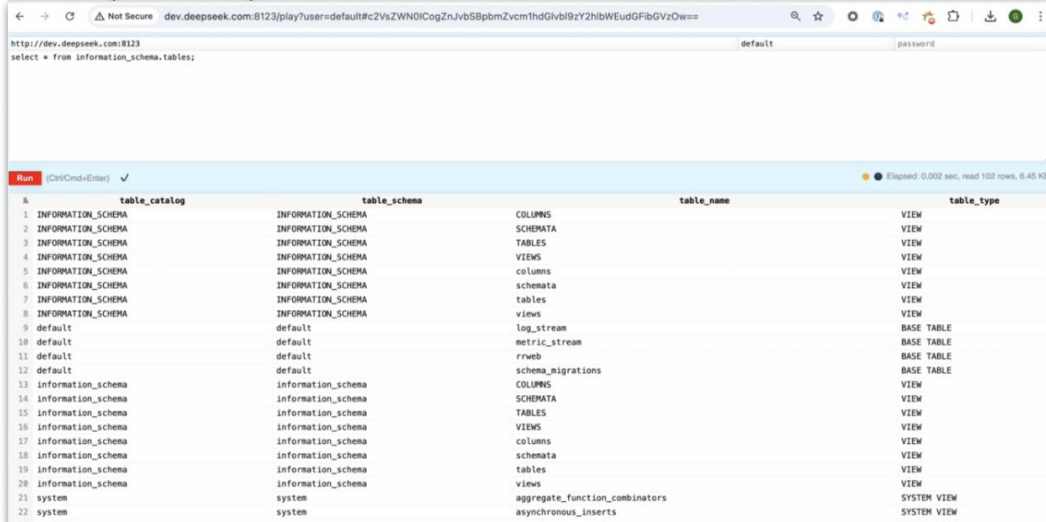
More info: <https://www.wiz.io/blog/wiz-research-uncovers-exposed-deepseek-database-leak>



Open access

No TLS

Open to internet



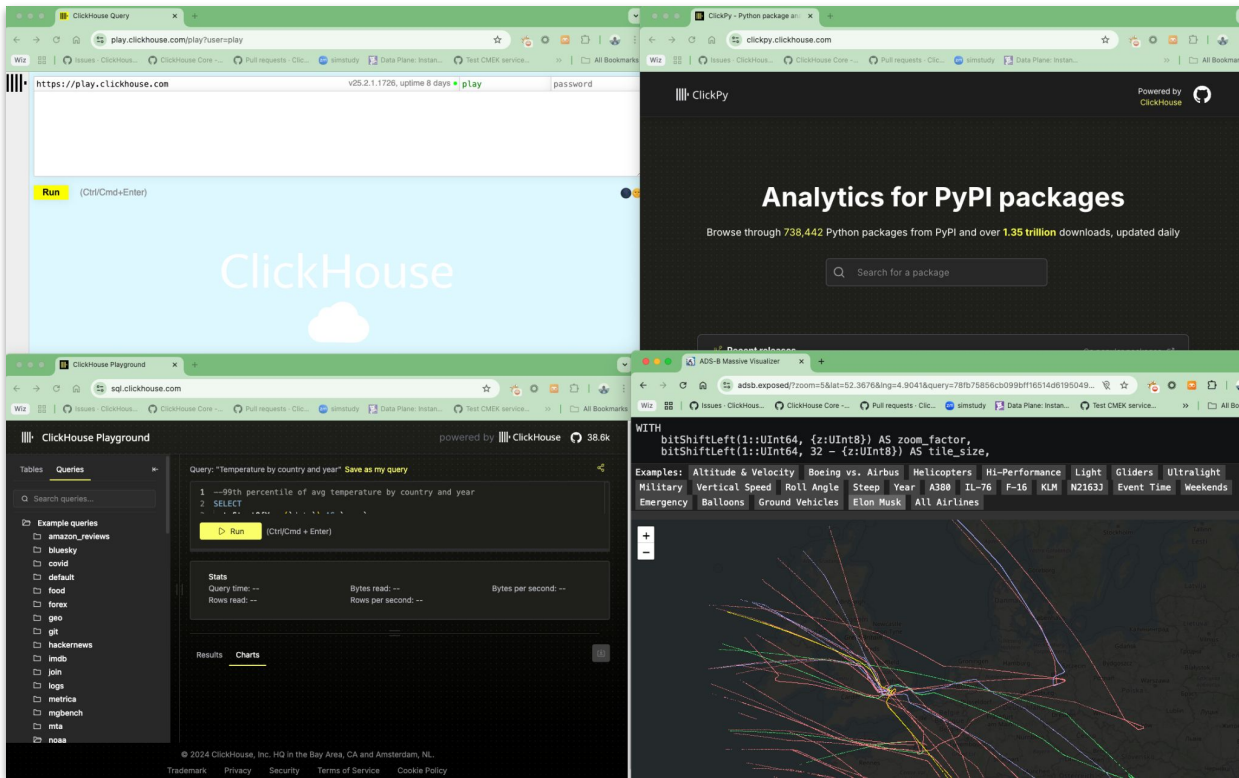
#	table_catalog	table_schema	table_name	table_type
1	INFORMATION_SCHEMA	INFORMATION_SCHEMA	COLUMNS	VIEW
2	INFORMATION_SCHEMA	INFORMATION_SCHEMA	SCHEMATA	VIEW
3	INFORMATION_SCHEMA	INFORMATION_SCHEMA	TABLES	VIEW
4	INFORMATION_SCHEMA	INFORMATION_SCHEMA	VIEWS	VIEW
5	INFORMATION_SCHEMA	INFORMATION_SCHEMA	COLUMNS	VIEW
6	INFORMATION_SCHEMA	INFORMATION_SCHEMA	SCHEMATA	VIEW
7	INFORMATION_SCHEMA	INFORMATION_SCHEMA	TABLES	VIEW
8	INFORMATION_SCHEMA	INFORMATION_SCHEMA	VIEWS	VIEW
9	default	default	log_stream	BASE TABLE
10	default	default	metric_stream	BASE TABLE
11	default	default	rwmb	BASE TABLE
12	default	default	schema_migrations	BASE TABLE
13	information_schema	information_schema	COLUMNS	VIEW
14	information_schema	information_schema	SCHEMATA	VIEW
15	information_schema	information_schema	TABLES	VIEW
16	information_schema	information_schema	VIEWS	VIEW
17	information_schema	information_schema	COLUMNS	VIEW
18	information_schema	information_schema	SCHEMATA	VIEW
19	information_schema	information_schema	TABLES	VIEW
20	information_schema	information_schema	VIEWS	VIEW
21	system	system	aggregate_function_combinators	SYSTEM VIEW
22	system	system	asynchronous_inserts	SYSTEM VIEW



More info: <https://www.wiz.io/blog/wiz-research-uncovers-exposed-deepseek-database-leak>



Open access



More info: <https://www.wiz.io/blog/wiz-research-uncovers-exposed-deepseek-database-leak>



No Authentication

No TLS

Open to internet

default user & no password

#	table_catalog	table_schema	table_name	table_type
1	INFORMATION_SCHEMA	INFORMATION_SCHEMA	COLUMNS	VIEW
2	INFORMATION_SCHEMA	INFORMATION_SCHEMA	SCHEMATA	VIEW
3	INFORMATION_SCHEMA	INFORMATION_SCHEMA	TABLES	VIEW
4	INFORMATION_SCHEMA	INFORMATION_SCHEMA	VIEWS	VIEW
5	INFORMATION_SCHEMA	INFORMATION_SCHEMA	columns	VIEW
6	INFORMATION_SCHEMA	INFORMATION_SCHEMA	schemata	VIEW
7	INFORMATION_SCHEMA	INFORMATION_SCHEMA	tables	VIEW
8	INFORMATION_SCHEMA	INFORMATION_SCHEMA	views	VIEW
9	default	default	log_stream	BASE TABLE
10	default	default	metric_stream	BASE TABLE
11	default	default	rwmb	BASE TABLE
12	default	default	schema_migrations	BASE TABLE
13	information_schema	information_schema	COLUMNS	VIEW
14	information_schema	information_schema	SCHEMATA	VIEW
15	information_schema	information_schema	TABLES	VIEW
16	information_schema	information_schema	VIEWS	VIEW
17	information_schema	information_schema	columns	VIEW
18	information_schema	information_schema	schemata	VIEW
19	information_schema	information_schema	tables	VIEW
20	information_schema	information_schema	views	VIEW
21	system	system	aggregate_function_combinators	SYSTEM VIEW
22	system	system	asynchronous_inserts	SYSTEM VIEW



More info: <https://www.wiz.io/blog/wiz-research-uncovers-exposed-deepseek-database-leak>

Full access, no quota, no limit & data has no encryption

Log Stream Query

trace_id	span_name	span_id	parent_span_id	severity_number	severity_text
06976a2b387c335e6d76fa2a84570	usage_checker_batch	06fae5993727c0		0	ok
0117496042067c14880168788a7d5f2	usage_checker_batch	7794af2c1456892		0	ok
0117496042067c14880168788a7d5f2	send_completion_request	8c2a2b35f156a70	7794af2c1456892	0	ok
af420633a0487055370baad293e411b	check_api_key_from_header	18028d0f10823c4	48ecde767030908	0	ok
af420633a0487055370baad293e411b	api_chat_completion_request	48ecde767030908		0	ok
af420633a0487055370baad293e411b	api_handler	4f6c8a705509ff79	18028d0f10823c4	0	ok
af420633a0487055370baad293e411b	generate	069327c43f8a73c	4f6c8a705509ff79	0	ok
af420633a0487055370baad293e411b	generate_stream	57209277b122a81	069327c43f8a73c	0	ok
af420633a0487055370baad293e411b	sse_generation_task	cc1f09ec7f6051c1	4f6c8a705509ff79	0	ok
af420633a0487055370baad293e411b	sse_generation	60a053474013079	4f6c8a705509ff79	0	ok

span_name
check_api_key_from_db_instance
check_api_key_from_header
api_chat_completion_request
concurrency_limit
check_balance_enough
p8s_trace_data_on_set
p8s_init
generate
p8s_trace_data_on_set
generate_stream
sse_generation
stream_cpl_api_request
p8s_trace_data_on_set
p8s_trace_data_on_set
sqlite_cold_acquire
sqlite_data_transfer
GET /auth-api/v0/index.html
GET /api/v0/index.html
GET /api/v0/index.html
sqlite_cold_acquire

api-backend
api-backend
api-backend
platform-backend
chat-backend
chat-backend
api-backend
platform-backend
chat-backend
usage-checker

Services & APIs

```
string.values
["check_api_key_from_db_instance","sk-79d...","deepsuite-api-server/src/middleware/dependency.rs","deepsuite_api_server:middleware:dependency","api-backend","api-backend","internal","tokio-runtime-worker"]
[{"key":"event","type":"string","value":"Request initiated from"}]
[{"key":"event","type":"string","value":"started processing from"}]
["concurrency_limit","deepsuite-api-server/src/controller/chat/api.rs","deepsuite_api_se..."]
[{"key":"event","type":"string","value":"dispatch result: Serv..."}]
["check_balance_enough","deepsuite-api-server/src/controller/chat/api.rs","deepsuite_api..."]
[{"key":"event","type":"string","value":"Memory stats: RSS pre..."}]
["p8s_init","deepsuite-api-server/src/controller/chat/api.rs","deepsuite_api_server:con..."]
[{"key":"event","type":"string","value":"Sampling Params: Samp..."}]
["p8s_trace_data_on_set","deepsuite-api-server/src/controller/prometheus.rs","deepsuite..."]
[{"key":"event","type":"string","value":"p8s_trace_data_on_set..."}]
["sse_keepalive_task","deepsuite-api-server/src/controller/chat/api.rs","deepsuite_api_s...
```

```
("JaegerTag":{"api_key":"sk-79d...","busy_ns":58927,"code_filepath":"deepsuite-api-server/src/middleware/dep..."},{"span_kind":"internal","thread_id":3,"thread_name":"tokio-runtime-worker","duration":2842,"logs":[{"operationName":"check_a...","type":"SQL","spanID":"8b1d1f68b1e4","traceID":"6a4c10f3a1672805472389713846cd6f1","spanID":"2ef4418083447750","startTi..."}]})
("JaegerTag":{"busy_ns":1363413,"code_filepath":"deepsuite-api-server/src/middleware/dependency.rs","code_lineno":43,"code_namespace":..."},{"busy_ns":2487129,"code_filepath":"deepsuite-api-server/src/server.rs","code_lineno":275,"code_namespace":"deepsuite_api..."},{"busy_ns":31181,"code_filepath":"deepsuite-api-server/src/controller/chat/api.rs","code_lineno":211,"code_namespace":"de..."},{"JaegerTag":{"api_key_tracking_id":"08f4b02-53be-455c-b8a0-03033a7c7d4c","busy_ns":1493314,"code_filepath":"deepsuite-api-server/src..."},{"busy_ns":44288,"code_filepath":"deepsuite-api-server/src/controller/chat/api.rs","code_lineno":245,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":15113,"code_filepath":"deepsuite-api-server/src/controller/prometheus.rs","code_lineno":263,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":328,"code_filepath":"deepsuite-api-server/src/controller/chat/api.rs","code_lineno":398,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":120781,"code_filepath":"deepsuite-api-server/src/generation/client.rs","code_lineno":57,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":383,"code_filepath":"deepsuite-api-server/src/controller/prometheus.rs","code_lineno":215,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":15672818,"code_filepath":"deepsuite-api-server/src/generation/client.rs","code_lineno":358,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":631216,"code_filepath":"deepsuite-api-server/src/controller/chat/api.rs","code_lineno":1849,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":32688379,"code_filepath":"deepsuite-api-server/src/controller/chat/api.rs","code_lineno":788,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":34778875,"code_filepath":"deepsuite-api-server/src/controller/chat/api.rs","code_lineno":1882,"code_namespace":"de..."},{"JaegerTag":{"busy_ns":322,"code_filepath":"deepsuite-api-server/src/controller/prometheus.rs","code_lineno":215,"code_namespace":"de...
```

DeepSeek API Key Leakage

WIZ Research



More info: <https://www.wiz.io/blog/wiz-research-uncovers-exposed-deepseek-database-leak>



default user - worst case scenario (DO NOT DO THIS)

```
[14:14:46]-pewpew-[ ]
> docker run --rm -d clickhouse/clickhouse-server:head-alpine
9de07b85a6c6bf3bb14ec8fe7d34e9b21beb8dfcdc73a95107110c5aad686813

[14:15:21]-pewpew-[ ]
> docker exec -it 9d bash
9de07b85a6c6:/# ls /etc/clickhouse-server/
config.d/  config.xml  users.d/    users.xml
9de07b85a6c6:/# ls /etc/clickhouse-server/users.d/
default-user.xml
9de07b85a6c6:/# cat /etc/clickhouse-server/users.d/default-user.xml
<clickhouse>
  <!-- Docs: <https://clickhouse.com/docs/en/operations/settings/settings_users/> -->
  <users>
    <default>
      <!-- User default is available only locally -->
      <networks>
        <ip>::1</ip>
        <ip>127.0.0.1</ip>
      </networks>
    </default>
  </users>
</clickhouse>
9de07b85a6c6:/#
```



More info: <https://www.wiz.io/blog/wiz-research-uncovers-exposed-deepseek-database-leak>

Hackernews clarification...

Wiz Research Uncovers Exposed DeepSeek Database Leaking Sensitive Information, Including Chat History

AlexClickHouse 19 days ago | root | parent | next [-]

ClickHouse does not allow external connections by default.

If someone wants to configure an unauthenticated access from the Internet, they have to do the following extra steps:

- enable listening to the wildcard address;
- remove IP filtering for the default user;
- set up a no-password authentication;

It is possible to ignore and turn off all guardrails that the system has by default, but it needs extra efforts. However, it's possible that someone copy-pasted a wrong configuration file from somewhere without knowing what is inside, or do something like - listen to localhost, but expose ports from Docker.

A use case for direct database access exists, and is acceptable, assuming you set up a readonly user, grant access to specific tables, limit queries by complexity, and limit total usage by quotas. This is demonstrated by the following public services:

<https://play.clickhouse.com/>

<https://adsb.exposed/>

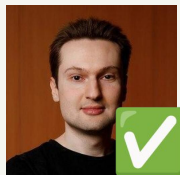
<https://reversedns.space/>

In this way, ClickHouse can be used to implement public data APIs (which is probably not what DeepSeek wanted).

ClickHouse has a wide range of security and access control restrictions: authentication methods with SSL certificates; SSH keys; even simple password-based auth allows bcrypt and short-living credentials; integration with LDAP and Kerberos; every authentication method can be limited on a network level; full Role-Based Access Control; fine-grained restrictions on query complexity and resource consumption, user quotas.

But still, according to Shodan, there are 33,000 misconfigured ClickHouse servers on the Internet: <https://www.shodan.io/search?query=clickhouse> This can be attributed to a high popularity of ClickHouse (it is the most widely used analytic DBMS).

When you use ClickHouse Cloud, which is a commercial cloud service based on the open-source ClickHouse database (<https://clickhouse.com/cloud>), it ensures the needed security measures, improving strong defaults even more: TLS, strong credentials, IP filtering; plus it allows private link, data encryption with customer keys, etc.



In this blog post, we will detail our discovery and also consider the broader implications for the industry at large.



More info: <https://news.ycombinator.com/item?id=42871371>



03

OSS ClickHouse Security features

CREATE USER - Authentication

- Username & Password : double_sha1_password;
sha256; bcrypt

```
CREATE USER sec_user IDENTIFIED  
WITH bcrypt_hash BY '$2y$10$3y43Ch.....J9k1bw4Twhy8YLm'
```



CREATE USER - Authentication

- Username & Password : double_sha1_password; sha256; bcrypt
- SSH authentication

```
CREATE USER sec_user IDENTIFIED  
WITH bcrypt_hash BY '$2y$10$3y43Ch.....J9k1bw4Twhy8YLm'
```

```
CREATE USER sec_user IDENTIFIED  
WITH ssh key BY KEY 'AAAC3NzaC1lZDI...AtNYgwncUnjaS14e1Od'  
TYPE `ssh-ed25519`
```



CREATE USER - Authentication

- Username & Password : double_sha1_password; sha256; bcrypt
- SSH authentication
- Mutual TLS (mtls - certificate authentication)

```
CREATE USER sec_user IDENTIFIED  
WITH bcrypt_hash BY '$2y$10$3y43Ch.....J9k1bw4Twhy8YLm'
```

```
CREATE USER sec_user IDENTIFIED  
WITH ssh key BY KEY 'AAAC3NzaC1lZDI...AtNYgwncUnjaSl4e1Od'  
TYPE `ssh-ed25519`
```

```
CREATE USER sec_user IDENTIFIED  
WITH ssl_certificate CN 'serviceA.customer.com'
```



CREATE USER - Authentication

- Username & Password : double_sha1_password; sha256; bcrypt
- SSH authentication
- Mutual TLS (mtls - certificate authentication)
- Or 2 authentication scheme at the same time

```
CREATE USER sec_user IDENTIFIED  
WITH bcrypt_hash BY '$2y$10$3y43Ch.....J9klbw4Twhy8YLm'
```

```
CREATE USER sec_user IDENTIFIED  
WITH ssh key BY KEY 'AAAC3NzaC1lZDI...AtNYgwncUnjaSl4e1Od'  
TYPE `ssh-ed25519`
```

```
CREATE USER sec_user IDENTIFIED  
WITH ssl_certificate CN 'serviceA.customer.com'
```

```
CREATE USER sec_user IDENTIFIED  
WITH bcrypt hash BY '$2y$10$3y43Ch.....J9klbw4Twhy8YLm',  
ssh_key BY KEY 'AAAC3NzaC1lZDI...AtNYgwncUnjaSl4e1Od'  
TYPE `ssh-ed25519`
```




CREATE USER - Authentication - External

🏠 / Security and Authentication / External Authenticators

 [Edit this page](#)

External User Authenticators and Directories

 Not supported in ClickHouse Cloud



Note

This page is not applicable to ClickHouse Cloud. The feature documented here is not available in ClickHouse Cloud services. See the ClickHouse Cloud Compatibility guide for more information.

ClickHouse supports authenticating and managing users using external services.

The following external authenticators and directories are supported:

- [LDAP Authenticator and Directory](#)
- Kerberos [Authenticator](#)
- [SSL X.509 authentication](#)
- HTTP [Authenticator](#)



CREATE USER - VALID UNTIL

VALID UNTIL specify the expiration date and, optionally, the time for an authentication method

```
CREATE USER sec_user_5 IDENTIFIED  
WITH bcrypt_hash BY '$2y$10$3y43Ch.....J9klbw4m'  
VALID UNTIL '2025-03-01'
```



CREATE USER - NETWORK RESTRICTION

HOST IP server restrict access from the specified IP address or a subnetwork

```
CREATE USER sec_user_5 IDENTIFIED  
WITH bcrypt_hash BY '$2y$10$3y43Ch.....J9klbw4m'  
VALID UNTIL '2025-03-01'  
HOST IP '10.1.1.0/24'
```



CREATE USER - QUOTA

Using **SETTINGS**, it is possible to configure quota on various limits such as
max_row_to_read, max_result_row,
max_result_bytes, max_memory_usage and
many more

```
CREATE USER sec_user_5 IDENTIFIED
WITH bcrypt_hash BY '$2y$10$3y43Ch.....J9klbw4m'
VALID UNTIL '2025-03-01'
HOST IP '10.1.1.0/24'
SETTINGS max_memory_usage=100000,
max_rows_to_read=10000,
max_result_rows=10
```



CREATE USER - READONLY

Setting **readonly = 1** prohibits the user from changing settings.

Tips: This setting is perfect for readonly service account - preventing accidental delete when granting wrong roles/permissions.

```
CREATE USER sec_user_5 IDENTIFIED WITH bcrypt_hash BY  
'$2y$10$3y43ChJKzPN15xbS400e6eQSGDtV7.zqU0mffFJ9klbw4Twhy8Ylm'  
  
HOST IP '10.1.1.0/24'  
  
VALID UNTIL '2025-03-01'  
  
SETTINGS max_memory_usage=100000,  
  
max_rows_to_read=10000,  
  
max_result_rows=10,  
  
readonly=1
```



Authorisation - GRANTs & Roles

- GRANT/REVOKE privileges to ClickHouse user accounts or roles.

```
GRANT SELECT ON system.* TO `developer_role`  
REVOKE SELECT ON system.query_* FROM `developer_role`  
GRANT SELECT,CREATE,INSERT ON `developer_db` TO `santran`
```



Authorisation - GRANTs & Roles

- GRANT/REVOKE privileges to ClickHouse user accounts or roles.
- Assigns multiple roles to user accounts or to the other roles.

```
GRANT SELECT ON system.* TO `developer_role`  
REVOKE SELECT ON system.query_* FROM `developer_role`  
GRANT SELECT,CREATE,INSERT ON `developer_db` TO `santran`
```

```
GRANT `developer_role`,`analytic_role` TO `santran`
```



Authorisation - GRANTs & Roles

- GRANT/REVOKE privileges to ClickHouse user accounts or roles.

```
GRANT SELECT ON system.* TO `developer_role`  
REVOKE SELECT ON system.query_* FROM `developer_role`  
GRANT SELECT,CREATE,INSERT ON `developer_db` TO `santran`
```

- Assigns multiple roles to user accounts or to the other roles.

```
GRANT `developer_role`,`analytic_role` TO `santran`
```

- Administrators can grant permissions if they have "WITH GRANT OPTION"

```
GRANT * ON `developer_db` TO `santran` WITH GRANT OPTION
```



Authorisation - Row Policy

Allows specifying a condition to filter rows for user or role.

The users will see **only** the rows if the condition satisfy.

```
CREATE ROW POLICY analytic ON secure.data  
USING security_tag='analytic'  
TO analytic_role
```

```
CREATE ROW policy header_fun ON secure.data  
  
FOR SELECT USING security_tag IN  
( SELECT tag from security_tag_reference  
  WHERE user=getClientHTTPHeader('X-USER')  
)  
TO grafana
```

Authorisation - Row Policy

- Allows specifying a condition to

filter

the c

non-

```
CREATE ROW POLICY analytic ON secure.data  
USING security_tag='analytic'  
AS RESTRICTIVE TO user_santran
```

```
[15:49:24]-pewpew-[ ]  
→ curl -H "X-ClickHouse-User: grafana" -H "X-USER:san.tran@clickhouse.com" "https://mt[REDACTED]2.aws.clickhouse-staging.  
com?query=select+*+from+secure.flag&allow_get_client_http_header=true"  
clickhouse clickhouse comment Some secret data for clickhouse?
```

```
[15:49:27]-pewpew-[ ]  
→ curl -H "X-ClickHouse-User: grafana" -H "X-USER:admin" "https://mt[REDACTED]clickhouse-staging.com?query=select+*  
+from+secure.flag&allow_get_client_http_header=true"  
flag flag R 0H0Q0000mR0000000gW0c0_00I0v0G000  
not-a-flag not-a-flag The flag has comment=flag haha  
whatever some comment The flag has comment=flag haha  
clickhouse clickhouse comment Some secret data for clickhouse?
```

- Cleave

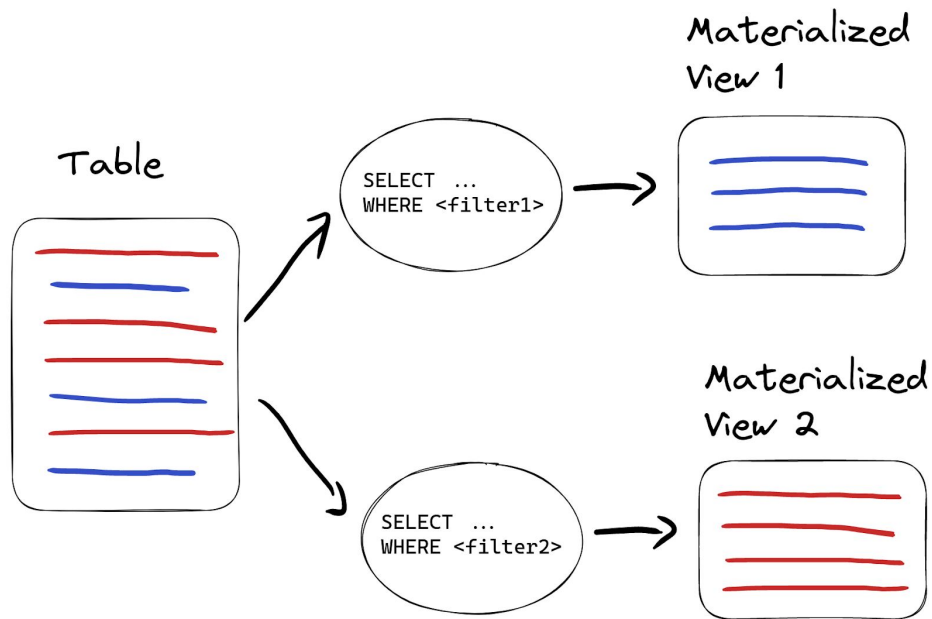
- Encryption at rest

-



Authorisation - View security

Previously, user need access to underlying tables in order to access View.



Authorisation - SQL Security Definer >= 24.2

With **SQL Security**, it is now possible to specify DEFINER user that would execute the underlying query to access the table.

```
CREATE VIEW employees  
DEFINER = service_payroll_account  
SQL SECURITY DEFINER  
AS  
SELECT name, department  
FROM payroll;
```

```
GRANT SELECT ON employees  
TO ops_role
```

Authorisation - SQL Security Definer >= 24.2

In this example of **SQL Security**, definer_user has access to both source and target table.

When data is inserted into source_table, the target_table is also populated and the definer_user permission is applied.

This means that the user inserted data to source_table do not need to have access to target_table as well.


```
CREATE MATERIALIZED VIEW materialized_view  
TO target_table  
DEFINER = definer_user  
SQL SECURITY DEFINER  
AS SELECT *  
FROM source_table;
```

```
GRANT SELECT ON default.source_table to  
definer_user;  
GRANT INSERT ON default.target_table to  
definer_user;  
GRANT INSERT ON default.source_table to  
other_user;
```



Encryption at rest - Disk Encryption

- Virtual Disk Encryption

 / [Manage and Deploy](#) / **External Disks for Storing Data** [Edit this page](#)

External Disks for Storing Data

Data, processed in ClickHouse, is usually stored in the local file system — on the same machine with the ClickHouse server. That requires large-capacity disks, which can be expensive enough. To avoid that you can store the data remotely. Various storages are supported:

1. [Amazon S3](#) object storage.
2. [Azure Blob Storage](#).
3. Unsupported: The Hadoop Distributed File System ([HDFS](#))

Note

ClickHouse also has support for external table engines, which are different from external storage option described on this page as they allow to read data stored in some general file format (like Parquet), while on this page we are describing storage configuration for ClickHouse [MergeTree](#) family or [Log](#) family tables.

1. to work with data stored on [Amazon S3](#) disks, use [S3](#) table engine.
2. to work with data stored in [Azure Blob Storage](#) use [AzureBlobStorage](#) table engine.
3. Unsupported: to work with data in the Hadoop Distributed File System — [HDFS](#) table engine.

Configuring external storage

[MergeTree](#) and [Log](#) family table engines can store data to [S3](#), [AzureBlobStorage](#), [HDFS](#) (unsupported) using a disk with types [s3](#), [azure_blob_storage](#), [hdfs](#) (unsupported) accordingly.

Disk configuration requires:

1. `type` section, equal to one of `s3`, `azure_blob_storage`, `hdfs` (unsupported), `local_blob_storage`, `web`.
2. Configuration of a specific external storage type.

Starting from 24.1 clickhouse version, it is possible to use a new configuration option. It requires to specify:

1. `type` equal to `object_storage`
2. `object_storage_type`, equal to one of `s3`, `azure_blob_storage` (or just `azure` from 24.3), `hdfs` (unsupported), `local_blob_storage` (or just `local` from 24.3), `web`. Optionally, `metadata_type` can be specified (it is equal to `local` by default), but it can also be set to `plain`, `web` and, starting from 24.4, `plain_rewriteable`. Usage of `plain` metadata type is described in [plain storage section](#), `web` metadata type can be used only with `web` object storage type, `local` metadata type stores metadata files locally (each metadata files contains mapping to files in object storage and some additional meta information about them).

E.g. configuration option

```
<s3>
  <type>s3</type>
  <endpoint>https://s3.eu-west-1.amazonaws.com/clickhouse-eu-west-1.clickhouse.com/data/</endpoint>
  <use_environment_credentials>1</use_environment_credentials>
</s3>
```

is equal to configuration (from 24.1):



Encryption at rest - Encryption Function

- Virtual Disk Encryption
- Built-in encryption functions

```
clickhouse-cloud :) select secret from secure.flag where security_tag='flag'
```

```
SELECT secret  
FROM secure.flag  
WHERE security_tag = 'flag'
```

```
Query id: 766de476-be17-44e9-af46-245c56760b77
```

```
1. [secret  
   R HQwRgW0c IvG ]
```

```
1 row in set. Elapsed: 0.003 sec.
```

```
clickhouse-cloud :) select  
decrypt('aes-256-ofb',secret,'d33ps33kR1Is_pretty_cool_model!!')  
from secure.flag where security_tag='flag'
```

```
SELECT decrypt('aes-256-ofb', secret, 'd33ps33kR1Is_pretty_cool_model!!')  
FROM secure.flag  
WHERE security_tag = 'flag'
```

```
Query id: f18e100ff894-4b64-8adf-f55296fcc44f
```

```
1. [decrypt('aes-256-ofb', secret, [HIDDEN id:1])  
   Hello Singapore meetup folks, howare you? ]
```

```
1 row in set. Elapsed: 0.003 sec.
```



Question?

For more information:

<https://clickhouse.com/docs>

<https://trust.clickhouse.com/>

QUIZ 1 - I can't access the data!

```
CREATE ROW POLICY analytic ON secure.data USING security_tag='analytic' AS RESTRICTIVE TO analytic_role
CREATE ROW POLICY developer ON secure.data USING security_tag='developer' AS RESTRICTIVE TO developer_role

GRANT analytic_role to `san.tran`
GRANT developer_role to `san.tran`
```



QUIZ 1 - I can't access the data!

```
CREATE ROW POLICY analytic ON secure.data USING security_tag='analytic' AS RESTRICTIVE TO analytic_role
CREATE ROW POLICY developer ON secure.data USING security_tag='developer' AS RESTRICTIVE TO developer_role

GRANT analytic_role to `san.tran`
GRANT developer_role to `san.tran`
```

```
CREATE ROW POLICY analytic ON secure.data USING security_tag='analytic' AS PERMISSIVE TO analytic_role
CREATE ROW POLICY developer ON secure.data USING security_tag='developer' AS PERMISSIVE TO developer_role

GRANT analytic_role to `san.tran`
GRANT developer_role to `san.tran`
```



ClickHouse on the internet...

SHODAN

Explore

Downloads

Pricing

"X-ClickHouse-Summary:"


Q

Account

TOTAL RESULTS

16,087

TOP COUNTRIES



China	3,484
Russian Federation	2,719
Germany	2,576
United States	2,254
Finland	1,417

More...

TOP PORTS

8123	9,305
9009	3,822
8443	1,990
443	200
80	106

View Report

Download Results

Historical Trend

View on Map

Advanced Search

Product Spotlight: We've Launched a new API for Fast Vulnerability Lookups. Check out [CVEDB](#)

144.76.97.57

static.57.97.76.144.clients.your-server.de
Hetzner Online GmbH
Germany, Falkenstein

database

HTTP/1.1 200 OK
Date: Wed, 12 Feb 2025 01:40:05 GMT
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Keep-Alive: timeout=10
X-ClickHouse-Summary: {"read_rows":"0","read_bytes":"0","written_rows":"0","written_bytes":"0","total_rows_to_re

2025-02-12T01:40:05.829109

49.13.99.18

static.18.99.13.49.clients.your-server.de
Hetzner Online GmbH
Germany, Nürnberg

database

HTTP/1.1 200 OK
date: Wed, 12 Feb 2025 00:59:52 GMT
content-type: text/html; charset=UTF-8
x-clickhouse-summary: {"read_rows":"0","read_bytes":"0","written_rows":"0","written_bytes":"0","total_rows_to_re
x-envoy-upstream-service...

2025-02-12T00:59:52.602773

96.46.187.84

Servers.com, Inc.
United States, Ashburn

database

HTTP/1.1 200 OK
Date: Wed, 12 Feb 2025 00:57:45 GMT
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Keep-Alive: timeout=60
X-ClickHouse-Summary: {"read_rows":"0","read_bytes":"0","written_rows":"0","written_bytes":"0","total_rows_to_re

2025-02-12T00:57:46.141249

ClickHouse on the internet...

censys

Results Tty CensysGPT Beta

Host Filters

Labels:

- 10,54K remote-access
- 6,775 database
- 2,341 jquery
- 1,618 bootstrap
- 1,588 login-page
-

Autonomous System:

- 2,415 HETZNER-AS
- 1,526 YANDEXCLOUD
- 1,355 ALIBABA-CN-NET
- Hangzhou Alibaba Advertising Co.,Ltd.
- 842 AMAZON-02
- 540 TENCENT-NET-AP
- Shenzhen Tencent Computer Systems Company Limited
-

Location:

- 3,374 China
- 2,847 Russia
- 2,221 United States
- 2,155 Germany
- 939 Finland
-

Service Filters

Service Names:

- 101,60K HTTP
- 12,48K SSH
- 7,262 UNKNOWN
- 7,191 MYSQL
- 1,889 REDIS
-

Ports:

Hosts

Results: **14,946** Time: 0.11s

138.197.179.140 (7leads.website)

- Linux DIGITALOCEAN-ASN (14061) Hesse, Germany
- remote-access requirejs bootstrap jquery
- 22/SSH 80/HTTP 443/HTTP 6060/HTTP 8088/HTTP
- 8123/HTTP

172.105.163.229 (172-105-163-229.ip.linodeusercontent.com)

- Ubuntu Linux AKAMAI-LINODE-AP Akamai Connected Cloud (63949) New South Wales, Australia
- remote-access
- 22/SSH 80/HTTP 443/HTTP 3000/HTTP 8123/HTTP
- 9000/HTTP

37.27.10.234

- Ubuntu Linux HETZNER-AS (24940) Uusimaa, Finland
- remote-access bootstrap clipboard.js handlebars jquery moment.js riot
- 22/SSH 8888/HTTP 10250/HTTP 10256/HTTP 30131/HTTP
- 31092/HTTP 31111/HTTP 31112/HTTP 31113/HTTP 31174/HTTP
- 31427/HTTP 31728/HTTP 31854/HTTP

106.52.94.17

- TENCENT-NET-AP Shenzhen Tencent Computer Systems Company Limited (45090) Guangdong, China
- database
- 80/HTTP 6600/HTTP 9000/HTTP 9004/MYSQL 9009/HTTP

89.169.147.42

- YANDEXCLOUD (200350) Moscow, Russia
- 8443/HTTP 9440/HTTP

77.91.122.214 (vm3268296.stark-industries.solutions)

- Ubuntu Linux STARK-INDUSTRIES (44477) Flevoland, Netherlands
- bulletproof remote-access
- 22/SSH 80/HTTP 443/HTTP 8123/HTTP 32769/HTTP
- 32770/HTTP 32771/HTTP



What's wrong with this picture?

