



DayBlock – Security Report

An internal security audit, written up for everyone

October 2026 · by René · dayblock.de

The essentials at a glance

- In October 2026 I checked DayBlock completely for security vulnerabilities – from sign-in and encryption to every single interface.
- Result: no critical vulnerability. There was no way to reach encrypted content, passwords or other people's accounts.
- Three places could be more careful – I improved all three while the audit was still running.
- This report is an internal audit and does not replace the independent external review. That one remains mandatory before the public launch.



Why this report

DayBlock is a day planner, and a day planner knows a lot about you: when you see the doctor, who you meet, how you felt on a Tuesday evening. That is why everything you write into DayBlock is encrypted on your device before it leaves it. The server only stores gibberish.

A promise like that is worthless if nobody checks it. So I attacked DayBlock systematically – the way someone with bad intentions would – and I am writing down here what came out of it. Only what is already fixed gets published: if I found an open vulnerability, I would close it first and report on it afterwards.

This report is written for everyone, not just for experts. Where a technical term is needed, I explain it. If you want to go deeper, the full threat model is a separate article in the DayBlock magazine.



What was checked

The audit went once through everything between your browser and the database:

- **Sign-in and account** the sign-in method where your password never leaves your device; sessions and their cookies; recovery with the 24 words; signing out and deleting the account.
- **Encryption** the keys created on your device; the wrapping of the main key; the encryption of every record; the question of whether the server really never sees plain text.
- **Sync between devices** whether anything can be injected, swapped or resurrected while the encrypted envelopes travel back and forth.
- **Groups and sharing** private groups whose key only lives in the invitation link; public groups; the report function.

- **Calendar subscriptions** whether fetching external calendars can be abused as a bridge into the internal network (a well-known class of attack that I played through deliberately).
- **Magazine and admin area** whether articles can smuggle foreign code into the page; whether the admin access is tight.
- **Emails, limits, headers** confirmation and recovery emails; brakes against password guessing; the security headers the browser evaluates.
- **Libraries** all third-party libraries in use, automatically checked for known vulnerabilities.

3 The result

I found no critical vulnerability. There was no way to reach encrypted content, intercept passwords, take over other people's accounts or turn the server into a tool against others.

What I did find were three places where DayBlock could be more careful than it was. In my assessment none of them was exploitable – they were missing seatbelts, not open doors. I improved all three while the audit was still running. They are listed in the next section, because that is exactly what this report is for.

4 Three improvements, applied immediately

1. The browser now insists on an encrypted connection, permanently.

DayBlock was always reachable only over an encrypted connection. What is new is a header that makes your browser remember this for a whole year: it will not even try to open an unencrypted connection any more – even if someone on the same Wi-Fi tried to redirect it there. Technical term: HSTS.

2. Sender information gets less trust.

Requests reach DayBlock through an upstream relay that tells the server which address a request comes from. The server now trusts this information exactly one hop far. Before, in an unlucky setup, someone could have attached a made-up sender address and tricked the brakes that prevent password guessing. The relay in use discards such made-up information itself; but security should not hang on a single point.

3. A side service of the test environment gets the strict headers too.

A small service that only exists in the internal test environment ran without the security headers that the website and app have had for a long time. It was no risk to user data – the test environment only holds toy data – but consistency is the better habit.

5 What withstood the test

An audit does not just find weaknesses; it also confirms strengths. These points held up:

- **Your password stays with you.** It never leaves your device – not even when you sign in. The server can neither see it nor write it down. Even someone who steals the database cannot simply try passwords, because that would require another secret that is not stored there.

- **Encrypted records cannot be swapped.** Every record is cryptographically bound to its own identifier. Anyone reshuffling envelopes in the database produces error messages, not wrong entries.
- **The calendar fetch is no back door.** DayBlock checks on every fetch that the target address is public – after redirects too, and even when the name lookup is being tricked.
- **Articles cannot inject code.** Magazine texts are defused character by character before they become a web page; links are limited to harmless forms.
- **The database takes no commands from outside.** All access is built so that input always stays data and never becomes a command.
- **Foreign pages cannot act in your name.** Modifying requests need a marker that only the real app can set; your session never travels along to other sites.
- **The libraries are clean.** The automatic check of all third-party libraries reports no known vulnerability.

6 Honest limits

Even a good audit does not change these points – you should know them:

- **The server delivers the code.** The fundamental problem of every encrypted web app: whoever controls the server could deliver altered code that reads along. That is why I advertise “encrypted on your device” – not “I can technically never read your data”. Planned: verifiable builds with published checksums, public app code, and later apps with built-in code.
- **Two-factor sign-in is still missing.** Until it arrives, a short or reused password remains the biggest risk – and only you can keep that small.
- **Who is registered can be guessed.** As with almost every service, trying to create an account with someone else’s email address reveals whether one already exists. Recovery and resending the confirmation deliberately answer neutrally instead.
- **If you lose both password and recovery key, your data is gone.** Nobody can bring it back – not even me. That is the price of real encryption, and DayBlock tells you so clearly when you sign up.

7 How the audit was done – and what this report is not

This audit was an internal one: I ran it myself, systematically and with the help of AI analysis tools – every interface, every input check, every cryptographic operation against DayBlock’s own specification and against known classes of attack. That is honest diligence, but it is not the same as a second pair of independent eyes.

So this holds: before the public launch, independent professionals will review DayBlock – encryption, sign-in and sync. This report does not replace that review; it prepares for it. The external report will be published as well.

Deliberately not in this report: server addresses, operating settings, and anything that would be a still-open vulnerability. Such details are only added after a fix.

8 What comes next

The next steps are planned in this order – planned means planned, not done:

- **Verifiable builds** every delivered version can be rebuilt exactly from the source code, and the checksums are published.
- **Two-factor sign-in** via authenticator app or passkey.
- **Public app code** so that the promise “the server never sees your content” can be verified by anyone.
- **External security review** with a published report, mandatory before the public launch.

9 Found a vulnerability?

Then please write to hallo@dayblock.de with the subject “Security”. Describe what you found and how it can be reproduced. I will answer, fix the issue as fast as I can – and will gladly credit you by name afterwards if you want that. Please only test with your own account and do not access other people’s data.

More about how DayBlock is built: the “threat model” article in the magazine on dayblock.de and the page “What René sees”. This report is dated October 2026; every future audit gets a new, dated report.

DayBlock · Security report October 2026 · October 2026 · by René · dayblock.de