

CERTCC / cveClient Public[Code](#) [Issues](#) 3 [Pull requests](#) 2 [Actions](#) [Projects](#) [Security and quality](#)

# Harden API key storage and encryption key management #39

Merged[sei-vsarvepalli](#) merged 2 commits into [CERTCC:main](#) from[jgamblin:security/credential-hard...](#)  2 days ago[Conversation](#) 3[Commits](#) 2[Checks](#) 0[Files changed](#) 2[jgamblin](#) commented [3 days ago](#)Contributor

## Summary

Hey Vijay! Following up on the security items from our email thread — this PR addresses the credential storage and encryption concerns (Findings 2 and 3 from my report).

Three changes:

- Don't store the API key in plaintext during login.** Previously the key was written to `localStorage/sessionStorage` immediately at login, before `enable_encryption()` finished its async work. Now the `key` field is skipped during the initial form storage loop — `enable_encryption()` already handles storing the encrypted version once it's ready.
- Store RSA private key as non-extractable `CryptoKey`.** In `save_key()`, the private key is now re-imported with `extractable: false` before being stored in IndexedDB. This means even if script accesses IndexedDB, it can't export the private key as JWK. Backward compatible — `import_key()` handles both legacy JWK entries and new `CryptoKey` entries gracefully.
- Add error handling for encryption script load failure.** If `encrypt-storage.js` fails to load (network issue, CSP, etc.), a `.fail()` handler now stores the key in `sessionStorage` only (not persistent `localStorage`) and warns the user. Previously the plaintext key would silently persist in `localStorage` forever.

## Test plan

- Login with "Keep me logged in" checked — verify key is NOT in localStorage immediately after clicking login (check DevTools > Application > Local Storage)
- After ~2-3 seconds, verify the encrypted key (data URI) appears in localStorage
- Login without "Keep me logged in" — verify key goes to sessionStorage after encryption
- If you have existing stored keys from before this change, verify auto-login still works (backward compat)
- Verify encryption toggle still works from the UI

Generated with [Claude Code](#)



**jgamblin** force-pushed the `security/credential-hardening` branch from `1a6868f` to `0e71872`  
3 days ago Compare



**sei-vsarvepalli** requested changes 2 days ago  
[View reviewed changes](#)



**sei-vsarvepalli** left a comment

Collaborator

In these cases, it will be good to update the version of each file. Eventually we would want to send an SBOM that captures all the components, say from the package.json.

Thanks

encrypt-storage.js Outdated



**sei-vsarvepalli** 2 days ago







Collaborator

Suggested change

```
-  
+ const encrypt_storage_version = "1.1.15";
```




**jgamblin** and others added 2 commits 2 days ago

-   [fix: prevent plaintext API key storage and harden encryption keys](#) ... [627c521](#)
-   [chore: bump version to 1.0.25 \(cveInterface\) and 1.1.15 \(encrypt-stor...](#) ... [5c98009](#)
-   **jgamblin** [force-pushed](#) the `security/credential-hardening` branch from `0e71872` to `5c98009` [2 days ago](#) Compare

**jgamblin** commented [2 days ago](#) Contributor Author

Hey Vijay, bumped the versions as requested — cveInterface.js to 1.0.25 and encrypt-storage.js to 1.1.15. Good to go for another look when you get a chance!



 **sei-vsarvepalli** approved these changes [2 days ago](#)

[View reviewed changes](#)

  **sei-vsarvepalli** merged commit `d518ff0` into `CERTCC:main` [2 days ago](#)

[Sign up for free](#) to join this conversation on GitHub. Already have an account? [Sign in to comment](#)

**Reviewers**

 **sei-vsarvepalli** 

**Assignees**

No one assigned

**Labels**

None yet

**Projects**

None yet

### Milestone

No milestone

---

### Development

Successfully merging this pull request may close these issues.

None yet

---

### 2 participants

