Commit `9f50bb2`

new-usemame authored 2 days ago · ✕ 0 / 1 · Verified

```
fix(kobo_auth): close IDOR in generate_auth_token / delete_auth_token (#1303)
(#18)

The Kobo auth-token routes accepted any user_id from the URL and were
gated only by @user_login_required. Any authenticated user could:

  GET  /kobo_auth/generate_auth_token/<victim_user_id>
  POST /kobo_auth/deleteauthtoken/<victim_user_id>

…and either mint an auth token for the victim (impersonation: their Kobo
sync now flows through the attacker's session) or revoke their token
(soft-DoS).

Add an IDOR guard: only the user themselves OR an admin may operate on a
given user_id; otherwise abort(403). Mirrors the standard pattern used
in cps/admin.py for admin-only routes.

Reported upstream as crocodilestick/Calibre-Web-Automated#1303 (no
upstream fix at time of writing). The patch is identical for both routes
because the symmetry of the bug is the same.

Co-authored-by: new-usemame <248195428+new-usemame@users.noreply.github.com>
```

main (#18) · v4.0.12 ••• v4.0.7

1 parent 09bf581 commit 9f50bb2

1 file changed +12

Top

Filter files...

cps

kobo_auth.py

Search within code

cps/kobo_auth.py

```
@@ -70,6 +70,13 @@
70 70 @kobo_auth.route("/generate_auth_token/<int:user_id>")
71 71 @user_login_required
72 72 def generate_auth_token(user_id):
+ 73     # IDOR guard: only the user themselves OR an admin may mint a Kobo
+ 74     # auth token for a given user_id. Without this, any authenticated
+ 75     # user can request /kobo_auth/generate_auth_token/<other_user_id>
+ 76     # and walk away with a token that authorizes Kobo sync as that user.
+ 77     # Reported upstream as crocodilestick/Calibre-Web-Automated#1303.
+ 78     if current_user.id != user_id and not current_user.role_admin():
+ 79         abort(403)
73 80     warning = False
74 81     host_list = request.host.rsplit(':')
75 82     if len(host_list) == 1:
@@ -112,6 +119,11 @@ def generate_auth_token(user_id):
112 119 @kobo_auth.route("/deleteauthtoken/<int:user_id>", methods=["POST"])
113 120 @user_login_required
114 121 def delete_auth_token(user_id):
+ 122     # IDOR guard: same model as generate_auth_token. Without this any
+ 123     # authenticated user could revoke another user's Kobo auth token,
+ 124     # locking them out of Kobo sync (a softer DoS variant of #1303).
+ 125     if current_user.id != user_id and not current_user.role_admin():
+ 126         abort(403)
115 127     # Invalidate any previously generated Kobo Auth token for this user
116 128     ub.session.query(ub.RemoteAuthToken).filter(ub.RemoteAuthToken.user_id ==
    user_id)\
117 129         .filter(ub.RemoteAuthToken.token_type==1).delete()
```

Comments 0

Please sign in to comment.