ChurchCRM / **CRM** Public

Code | Issues 137 | Pull requests 23 | Discussions | Actions | Projects

Commit 28ea7a2

DawoudIO authored last week · 25 / 26 · Verified

```
security: block no-permission users + fix IDOR on person API (#8616)
```

master (#8616) · 7.2.1 7.2.0

1 parent a57cce5 commit 28ea7a2

8 files changed +227 -16 lines changed

Top

Filter files…

- cypress
  - data
    - seed.sql
  - e2e/ui/security
    - limited-access.spec.js
- src
  - ChurchCRM
    - Slim/Middleware
      - AuthMiddleware.php
    - model/ChurchCRM
      - User.php
    - Include
      - PageInit.php
    - api/routes/people
      - people-person.php
    - external

routes
system.php
templates
limited-access.php

8 files changed +227 -16 lines changed

Search within code

cypress/data/seed.sql

```
@@ -1966,7 +1966,7 @@ CREATE TABLE `user_usr` (
1966 1966  LOCK TABLES `user_usr` WRITE;
1967 1967  /*!40000 ALTER TABLE `user_usr` DISABLE KEYS */;
1968 1968  SET autocommit=0;
1969      - INSERT INTO `user_usr` VALUES
(1,'$2y$12$e3o8rmvWUYdgzUNB/AAMK.pRvT9rwsIZx4wYB0brOmVPB1UL.HA5S',0,'2026-04-
10 12:28:02',375,0,0,0,0,0,0,0,0,1,10,'skin-red',1,1,'2016-01-
01',23,1,'Admin','ajGwpy8Pdai22XDUpqjC5Ob04v0eG7EGgb4vz2bD2juT8YDmfM',0,NULL,
NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,0,NULL,NULL,NULL),
(3,'$2y$12$uWQcp6KU7C4JCTaVH.0hiekfpga36yBVhXG8.M/w9u3MmvHt/NWi2',0,'2025-11-
30 02:08:31',2,0,1,1,1,1,1,1,1,0,10,'skin-yellow-light',0,0,'2016-01-
01',26,0,'tony.wade@example.com','JZJApQ9XOnF7nvupWZlTWBRrqMtHE9eNcWBTUzEWGqL
4Sdqp6C',1,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,N
ULL),
(95,'$2y$12$7R0MqgyidzOqzXPrGbYdkO.y/decFpwSJM..fznzzvT4wiZqaJE4q',0,'2022-
12-29 21:01:30',0,0,1,1,0,0,0,0,0,0,10,'skin-blue',0,0,'2016-01-
01',26,0,'judith.matthews@example.com',NULL,0,NULL,NULL,NULL,NULL,NULL,NULL,N
ULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL),
(99,'$2y$12$OTZ5Y/hvCNVkfQqbQfNjRO/x2cBnx5O3yMJt4jXKw/yu/NjuxTHTK',1,'2025-
12-01 20:26:05',0,0,0,0,0,0,0,0,0,0,10,'skin-blue',0,0,'2016-01-
01',29,0,'amanda.black@example.com',NULL,1,NULL,NULL,NULL,NULL,NULL,NULL,NULL
,NULL,NULL,NULL,NULL,NULL,NULL,NULL);
     1969 + INSERT INTO `user_usr` VALUES
(1,'$2y$12$e3o8rmvWUYdgzUNB/AAMK.pRvT9rwsIZx4wYB0brOmVPB1UL.HA5S',0,'2026-04-
10 12:28:02',375,0,0,0,0,0,0,0,0,1,10,'skin-red',1,1,'2016-01-
01',23,1,'Admin','ajGwpy8Pdai22XDUpqjC5Ob04v0eG7EGgb4vz2bD2juT8YDmfM',0,NULL,
NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,0,NULL,NULL,NULL),
(3,'$2y$12$uWQcp6KU7C4JCTaVH.0hiekfpga36yBVhXG8.M/w9u3MmvHt/NWi2',0,'2025-11-
30 02:08:31',2,0,1,1,1,1,1,1,1,0,10,'skin-yellow-light',0,0,'2016-01-
01',26,0,'tony.wade@example.com','JZJApQ9XOnF7nvupWZlTWBRrqMtHE9eNcWBTUzEWGqL
```

```
4Sdqp6C',1,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,N
ULL),
(95,'$2y$12$7R0MqgyidzOqzXPrGbYdkO.y/decFpwSJM..fznzzvT4wiZqaJE4q',0,'2022-
12-29 21:01:30',0,0,1,1,0,0,0,0,0,0,10,'skin-blue',0,0,'2016-01-
01',26,0,'judith.matthews@example.com',NULL,0,NULL,NULL,NULL,NULL,NULL,NULL,N
ULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL),
(99,'$2y$12$OTZ5Y/hvCNVkfQqbQfNjRO/x2cBnx5O3yMJt4jXKw/yu/NjuxTHTK',1,'2025-
12-01 20:26:05',0,0,0,0,0,0,0,0,0,0,10,'skin-blue',0,0,'2016-01-
01',29,0,'amanda.black@example.com',NULL,1,NULL,NULL,NULL,NULL,NULL,NULL,NULL
,NULL,NULL,NULL,NULL,NULL,NULL,NULL),
(4,'$2y$12$e3o8rmvWUYdgzUNB/AAMK.pRvT9rwsIZx4wYB0brOmVPB1UL.HA5S',0,'2016-01-
01 00:00:00',0,0,0,0,0,0,0,0,0,0,10,'skin-blue',0,0,'2016-01-
01',26,0,'limited.user','limitedUserApiKeyForTesting1234567890123456 78',1,NUL
L,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL);
/*!40000 ALTER TABLE `user_usr` ENABLE KEYS */;
UNLOCK TABLES;
COMMIT;
```

...ress/e2e/ui/security/limited-access.spec.js

```
@@ -0,0 +1,85 @@
+ /// <reference types="cypress" />
+
+ /**
+  * Tests for limited-access users (EditSelf only, no admin permissions).
+  *
+  * Seed data: user "limited.user" (person ID 4, family 2) has
+  * usr_EditSelf=1 and all other permissions=0.
+  * Password: "changeme" (same as admin).
+  */
+ describe("Limited Access User", () => {
+     const limitedUser = "limited.user";
+     const limitedPassword = "changeme";
+
+     it("Login redirects to /external/limited-access", () => {
+         cy.clearCookies();
+         cy.visit("session/begin");
+         cy.get("input[name=User]").type(limitedUser);
+         cy.get("input[name=Password]").type(limitedPassword + "{enter}");
+
```

```
20 +         // Should end up on the limited access page, not the dashboard
21 +         cy.url({ timeout: 10000 }).should("include", "/external/limited-access");
22 +         cy.contains("Welcome");
23 +     });
24 +
25 +     it("Shows Verify Family Info button and Log Out button", () => {
26 +         cy.clearCookies();
27 +         cy.visit("session/begin");
28 +         cy.get("input[name=User]").type(limitedUser);
29 +         cy.get("input[name=Password]").type(limitedPassword + "{enter}");
30 +
31 +         cy.url({ timeout: 10000 }).should("include", "/external/limited-access");
32 +         cy.contains("Verify Family Info").should("exist");
33 +         cy.contains("Log Out").should("exist");
34 +     });
35 +
36 +     it("Verify Family Info link goes to /external/verify/{token}", () => {
37 +         cy.clearCookies();
38 +         cy.visit("session/begin");
39 +         cy.get("input[name=User]").type(limitedUser);
40 +         cy.get("input[name=Password]").type(limitedPassword + "{enter}");
41 +
42 +         cy.url({ timeout: 10000 }).should("include", "/external/limited-access");
43 +         cy.contains("Verify Family Info").click();
44 +         cy.url({ timeout: 10000 }).should("include", "/external/verify/");
45 +         // Verify page should show the family name (Campbell — person 4, family 1)
46 +         cy.get("body", { timeout: 10000 }).should("contain.text", "Campbell");
47 +     });
48 +
49 +     it("Log Out returns to login page", () => {
50 +         cy.clearCookies();
51 +         cy.visit("session/begin");
52 +         cy.get("input[name=User]").type(limitedUser);
53 +         cy.get("input[name=Password]").type(limitedPassword + "{enter}");
54 +
```

```
55 +         cy.url({ timeout: 10000 }).should("include", "/external/limited-access");
56 +         cy.contains("Log Out").click();
57 +         cy.url().should("include", "/session/begin");
58 +     });
59 +
60 +     it("Direct visit to /v2/dashboard redirects to limited-access", () => {
61 +         cy.clearCookies();
62 +         cy.visit("session/begin");
63 +         cy.get("input[name=User]").type(limitedUser);
64 +         cy.get("input[name=Password]").type(limitedPassword + "{enter}");
65 +
66 +         cy.url({ timeout: 10000 }).should("include", "/external/limited-access");
67 +
68 +         // Try to access an admin page directly
69 +         cy.visit("v2/dashboard", { failOnStatusCode: false });
70 +         cy.url().should("include", "/external/limited-access");
71 +     });
72 +
73 +     it("API call with limited user key returns 403", () => {
74 +         cy.apiRequest({
75 +             method: "GET",
76 +             url: "/api/person/1",
77 +             headers: {
78 +                 "x-api-key": "limitedUserApiKeyForTesting1234567890123456789",
79 +             },
80 +             failOnStatusCode: false,
81 +         }).then((resp) => {
82 +             expect(resp.status).to.eq(403);
83 +         });
84 +     });
85 + });
```

...hurchCRM/Slim/Middleware/AuthMiddleware.php

```
@@ -50,10 +50,31 @@ public function process(ServerRequestInterface $request, RequestHandlerInterface
50  50                  $logger->debug('API key authentication successful', [
51  51                      'path' => $request->getUri()->getPath()
52  52                  ]);
```

```
+
+                // Block users with no admin permissions from API access (GHSA-5w59-32c8-933v)
+                $apiUser = AuthenticationManager::getCurrentUser();
+                if ($apiUser->hasNoAdminPermissions()) {
+                    $response = new Response();
+                    $response->getBody()->write(json_encode(['error' => 'Account has limited permissions. Contact an administrator.']));
+                    return $response->withStatus(403)->withHeader('Content-Type', 'application/json');
+                }
            } elseif (AuthenticationManager::validateUserSessionIsActive(!$this->isPath($request, 'background'))) {
                // validate the user session; however, do not update tLastOperation if the requested path is "/background"
                // since /background operations do not connotate user activity.

+                // Block users with no admin permissions from MVC/API access (GHSA-5w59-32c8-933v)
+                $sessionUser = AuthenticationManager::getCurrentUser();
+                if ($sessionUser->hasNoAdminPermissions()) {
+                    if ($this->isBrowserRequest($request)) {
+                        $rootPath = SystemURLs::getRootPath();
+                        return (new Response())->withStatus(302)->withHeader('Location', $rootPath . '/external/limited-access');
+                    }
+                    // API request — return 403
+                    $response = new Response();
+                    $response->getBody()->write(json_encode(['error' => 'Account has limited permissions. Contact an administrator.']));
+                    return $response->withStatus(403)->withHeader('Content-Type', 'application/json');
+                }
+
                // User with an active browser session is still authenticated.
                // For browser requests (non-background), enforce any required redirect steps (e.g. forced password change).
                // Use a PSR-15 response redirect rather than calling ensureAuthentication() which exits via header().
```

```
@@ -118,4 +139,5 @@ private function redirectToLogin(ServerRequestInterface $request): ResponseInter
118 139
119 140             return $response->withStatus(302)->withHeader('Location', $redirectUrl);
120 141         }
    142 +
121 143     }
```

src/ChurchCRM/model/ChurchCRM/User.php

```
@@ -90,22 +90,25 @@ public function isEditSelfEnabled(): bool
90  90         }
91  91
92  92         /**
93      -      * Check if the user ONLY has EditSelf permission and no other functional permissions.
94      -      * These users need to be redirected to a self-service flow (e.g. family verify)
95      -      * rather than the full admin interface.
    93 +      * Check if the user lacks all functional admin permissions.
    94 +      * Users with no permissions (or only EditSelf) cannot use the admin interface
    95 +      * and should be redirected to a self-service flow or blocked.
96  96         *
97  97         * @see https://github.com/ChurchCRM/CRM/issues/8617
98  98         */
99      -    public function isEditSelfOnly(): bool
100     -    {
101     -        return $this->isEditSelf()
102     -            && !$this->isAdmin()
103     -            && !$this->isAddRecordsEnabled()
104     -            && !$this->isEditRecordsEnabled()
105     -            && !$this->isDeleteRecordsEnabled()
106     -            && !$this->isMenuOptionsEnabled()
107     -            && !$this->isManageGroupsEnabled()
108     -            && !$this->isFinanceEnabled();
    99 +    public function hasNoAdminPermissions(): bool
    100 +    {
    101 +        if ($this->isAdmin()) {
    102 +            return false;
```

```
+         }
+
+         return !$this->isAddRecords()
+             && !$this->isEditRecords()
+             && !$this->isDeleteRecords()
+             && !$this->isMenuOptions()
+             && !$this->isManageGroups()
+             && !$this->isFinance()
+             && !$this->isNotes();
      }

      /**
```

src/Include/PageInit.php

```
@@ -2,16 +2,25 @@

 use ChurchCRM\Authentication\AuthenticationManager;
 use ChurchCRM\dto\Cart;
+use ChurchCRM\dto\SystemURLs;
 use ChurchCRM\Service\PersonService;
 use ChurchCRM\Service\SystemService;
 use ChurchCRM\Utils\FunctionsUtils;
+use ChurchCRM\Utils\RedirectUtils;
+

 $personService = new PersonService();
 $systemService = new SystemService();

 // Basic security checks:
 if (empty($bSuppressSessionTests)) {  // This is used for the login page only.
     AuthenticationManager::ensureAuthentication();
+
+    // Block users with no admin permissions (GHSA-5w59-32c8-933v / #8617)
+    $currentUser = AuthenticationManager::getCurrentUser();
+    if ($currentUser->hasNoAdminPermissions()) {
+        RedirectUtils::redirect(SystemURLs::getRootPath() . '/external/limited-access');
+    }
 }
```

```
16  25
17  26      $sGlobalMessageClass = 'success';
```

src/api/routes/people/people-person.php

```
                @@ -207,11 +207,15 @@
207 207          *     @OA\Response(response=404, description="Person not found")
208 208          * )
209 209          */
210     -     // Get person by ID (requires EditRecords — prevents IDOR)
    210 +     // Get person by ID — IDOR check via canEditPerson (GHSA-5w59-32c8-933v)
211 211      $group->get('', function (Request $request, Response $response, array $args): Response {
212 212          $person = $request->getAttribute('person');
    213 +         $currentUser = AuthenticationManager::getCurrentUser();
    214 +         if (!$currentUser->canEditPerson((int) $person->getId(), (int) $person->getFamId())) {
    215 +             throw new HttpForbiddenException($request, gettext('You do not have permission to view this person'));
    216 +         }
213 217          return SlimUtils::renderStringJSON($response, $person->exportTo('JSON'));
214     -     })->add(new EditRecordsRoleAuthMiddleware());
    218 +     });
215 219
216 220      // Delete person
217 221      $group->delete('', function (Request $request, Response $response, array $args): Response {
```

src/external/routes/system.php

```
        •••   @@ -1,12 +1,49 @@
1   1     <?php
2   2
    3 + use ChurchCRM\Authentication\AuthenticationManager;
    4 + use ChurchCRM\dto\ChurchMetaData;
3   5     use ChurchCRM\dto\SystemURLs;
4   6     use ChurchCRM\Utils\VersionUtils;
5   7     use Slim\Routing\RouteCollectorProxy;
```

```
6   8   use Slim\Views\PhpRenderer;
7   9   use Psr\Http\Message\ResponseInterface as Response;
8   10  use Psr\Http\Message\ServerRequestInterface as Request;
9   11
    12  + // Limited access page for users with no admin permissions (GHSA-5w59-32c8-933v)
    13  + $app->get('/limited-access', function (Request $request, Response $response):
          Response {
    14  +     $renderer = new PhpRenderer(__DIR__ . '/../templates/');
    15  +     $userName = '';
    16  +     $churchName = ChurchMetaData::getChurchName();
    17  +     $verifyUrl = '';
    18  +
    19  +     // Try to get the user info from the active session
    20  +     try {
    21  +         if (AuthenticationManager::validateUserSessionIsActive(false)) {
    22  +             $user = AuthenticationManager::getCurrentUser();
    23  +             $person = $user->getPerson();
    24  +             $userName = $person ? ($person->getFirstName() . ' ' . $person-
          >getLastName()) : $user->getUserName();
    25  +
    26  +             // If user has a family, generate a verify link
    27  +             $familyId = $person ? $person->getFamId() : 0;
    28  +             if ($familyId > 0) {
    29  +                 $token = new \ChurchCRM\model\ChurchCRM\Token();
    30  +                 $token->build('verifyFamily', $familyId);
    31  +                 $token->save();
    32  +                 $verifyUrl = SystemURLs::getRootPath() . '/external/verify/' .
          $token->getToken();
    33  +             }
    34  +         }
    35  +     } catch (\Throwable $e) {
    36  +         // Session might be invalid — that's OK, just show the page without user
          info
    37  +     }
    38  +
    39  +     return $renderer->render($response, 'limited-access.php', [
    40  +         'sRootPath' => SystemURLs::getRootPath(),
    41  +         'userName' => $userName,
    42  +         'churchName' => $churchName,
    43  +         'verifyUrl' => $verifyUrl,
```

```
44  +     ]);
45  + });
46  +
10  47    $app->group('/system', function (RouteCollectorProxy $group): void {
11  48        $renderer = new PhpRenderer(__DIR__ . '/../templates/');
12  49
```

src/external/templates/limited-access.php

```
    @@ -0,0 +1,51 @@
1   + <?php
2   +
3   + use ChurchCRM\dto\ChurchMetaData;
4   + use ChurchCRM\dto\SystemURLs;
5   +
6   + $sPageTitle = gettext('My Account');
7   + $sBodyClass = 'page-auth page-login';
8   +
9   + require SystemURLs::getDocumentRoot() . '/Include/HeaderNotLoggedIn.php';
10  + ?>
11  +
12  + <div class="login-container">
13  +   <div class="login-wrapper">
14  +     <div class="login-form-section">
15  +       <div class="login-form-inner">
16  +         <!-- Header with Logo and Church Name -->
17  +         <div class="login-form-header">
18  +           <div class="login-header-logo">
19  +             <img src="<?= SystemURLs::getRootPath() ?>/Images/logo-churchcrm-
    350.jpg" alt="ChurchCRM" />
20  +           </div>
21  +           <h2 class="login-header-church-name"><?=
    htmlspecialchars(ChurchMetaData::getChurchName()) ?></h2>
22  +         </div>
23  +
24  +         <!-- Greeting -->
25  +         <div class="login-form-title">
26  +           <?php if (!empty($userName)): ?>
27  +           <h1><?= gettext('Welcome') ?>, <?= htmlspecialchars($userName) ?></h1>
28  +           <?php else: ?>
```

```
29 +           <h1><?= gettext('Welcome') ?></h1>
30 +           <?php endif; ?>
31 +           <p><?= gettext('You can review and verify your family information using the link below. If you need additional access, please contact your church administrator.') ?></p>
32 +         </div>
33 +
34 +         <!-- Action Buttons -->
35 +         <div class="d-grid gap-2 mb-3">
36 +           <?php if (!empty($verifyUrl)): ?>
37 +           <a href="<?= htmlspecialchars($verifyUrl) ?>" class="btn btn-primary btn-lg">
38 +             <i class="fa-solid fa-clipboard-check me-2"></i><?= gettext('Verify Family Info') ?>
39 +           </a>
40 +           <?php endif; ?>
41 +           <a href="<?= SystemURLs::getRootPath() ?>/session/end" class="btn btn-outline-secondary">
42 +             <i class="fa-solid fa-right-from-bracket me-2"></i><?= gettext('Log Out') ?>
43 +           </a>
44 +         </div>
45 +       </div>
46 +     </div>
47 +   </div>
48 + </div>
49 +
50 + <?php
51 + require SystemURLs::getDocumentRoot() . '/Include/FooterNotLoggedIn.php';
```

Comments 0

Please sign in to comment.