bluewave-labs / **Checkmate** Public

Code | Issues 62 | Pull requests 6 | Discussions | Actions | Projects | Securit

Commit d4a6072

Browse files

ajhollid authored last week · 5 / 5 · Verified

```
Merge pull request #2160 from bluewave-labs/fix/insecure-user-creation

fix: insecure user creation
```

develop (#2160) 2 parents 7951e10 + b387eba commit d4a6072

Filter files...

- client/src
  - Components/TabPanels/Account
    - TeamPanel.jsx
  - Pages/Auth/Register
    - Register.jsx
- server
  - controllers
    - authController.js
  - db/models
    - InviteToken.js
  - validation
    - joi.js

5 files changed +34 -25 lines changed

Search within code

client/src/Components/TabPanels/Account/TeamPanel.jsx +2 -2

```
@@ -25,7 +25,7 @@ const TeamPanel = () => {
25  25
26  26          const [toInvite, setToInvite] = useState({
27  27                  email: "",
28      -           role: ["0"],
    28  +           role: ["user"],
```

```
29  29          });
30  30          const [data, setData] = useState([]);
31  31          const [members, setMembers] = useState([]);
    @@ -107,7 +107,7 @@ const TeamPanel = () => {
107 107
108 108          const handleChange = (event) => {
109 109                  const { value } = event.target;
110     -                const newEmail = value?.toLowerCase() || value
    110 +                const newEmail = value?.toLowerCase() || value;
111 111                  setToInvite((prev) => ({
112 112                          ...prev,
113 113                          email: newEmail,
```

client/src/Pages/Auth/Register/Register.jsx +4 -3

```
    @@ -148,8 +148,8 @@ const Register = ({ isSuperAdmin }) => {
148 148                                  try {
149 149                                          const res = await networkService.verifyInvitationToken(token);
150 150                                          const invite = res.data.data;
151     -                                        const { role, email, teamId } = invite;
152     -                                        setForm({ ...form, email, role, teamId });
    151 +                                        const { email } = invite;
    152 +                                        setForm({ ...form, email });
153 153                                  } catch (error) {
154 154                                          navigate("/register", { replace: true });
155 155                                  }
    @@ -259,7 +259,8 @@ const Register = ({ isSuperAdmin }) => {
259 259          const handleChange = (event) => {
260 260                  const { value, id } = event.target;
261 261                  const name = idMap[id];
262     -                const lowerCasedValue = name === idMap["register-email-input"]? value?.toLowerCase() || value : value
    262 +                const lowerCasedValue =
    263 +                        name === idMap["register-email-input"] ? value?.toLowerCase() || value : value;
263 264                  setForm((prev) => ({
264 265                          ...prev,
265 266                          [name]: lowerCasedValue,
```

server/controllers/authController.js +8 -6

```
@@ -57,9 +57,9 @@ class AuthController {
57 57              */
58 58             registerUser = async (req, res, next) => {
59 59                     try {
60    -                          if(req.body?.email){
   60 +                          if (req.body?.email) {
61 61                               req.body.email = req.body.email?.toLowerCase();
62    -                          }
   62 +                          }
63 63                           await registrationBodyValidation.validateAsync(req.body);
64 64                     } catch (error) {
65 65                           const validationError = handleValidationError(error, SERVICE_NAME);
@@ -68,11 +68,13 @@ class AuthController {
68 68                     }
69 69                     // Create a new user
70 70                     try {
71    -                          const { inviteToken } = req.body;
   71 +                          const user = req.body;
72 72                           // If superAdmin exists, a token should be attached to all further register requests
73 73                           const superAdminExists = await this.db.checkSuperadmin(req, res);
74 74                           if (superAdminExists) {
75    -                                await this.db.getInviteTokenAndDelete(inviteToken);
   75 +                                const invitedUser = await this.db.getInviteTokenAndDelete(user.inviteToken);
   76 +                                user.role = invitedUser.role;
   77 +                                user.teamId = invitedUser.teamId;
76 78                           } else {
77 79                                // This is the first account, create JWT secret to use if one is not supplied by env
78 80                                const jwtSecret = crypto.randomBytes(64).toString("hex");
@@ -133,8 +135,8 @@ class AuthController {
133 135              */
134 136             loginUser = async (req, res, next) => {
135 137                     try {
136    -                          if(req.body?.email){
137    -                          req.body.email = req.body.email?.toLowerCase();
   138 +                          if (req.body?.email) {
   139 +                                req.body.email = req.body.email?.toLowerCase();
```

```
138  140                          }
139  141                          await loginValidation.validateAsync(req.body);
140  142                      } catch (error) {
```

server/db/models/InviteToken.js +1

```
     @@ -15,6 +15,7 @@ const InviteTokenSchema = mongoose.Schema(
15   15                       role: {
16   16                               type: Array,
17   17                               required: true,
     18  +                            default: ["user"],
18   19                       },
19   20                       token: {
20   21                               type: String,
```

server/validation/joi.js +19 -14

```
     @@ -63,11 +63,7 @@ const registrationBodyValidation = joi.object({
63   63                       }),
64   64               password: joi.string().min(8).required().pattern(passwordPattern),
65   65               profileImage: joi.any(),
66        -          role: joi
67        -                  .array()
68        -                  .items(joi.string().valid("superadmin", "admin", "user", "demo"))
69        -                  .min(1)
70        -                  .required(),
     66  +          role: joi.array().items(joi.string().valid("superadmin", "admin", "user",
         "demo")),
71   67               teamId: joi.string().allow("").required(),
72   68               inviteToken: joi.string().allow("").required(),
73   69       });
     @@ -83,7 +79,6 @@ const editUserBodyValidation = joi.object({
83   79               newPassword: joi.string().min(8).pattern(passwordPattern),
84   80               password: joi.string().min(8).pattern(passwordPattern),
85   81               deleteProfileImage: joi.boolean(),
86        -          role: joi.array(),
87   82       });
88   83
89   84       const recoveryValidation = joi.object({
     @@ -305,7 +300,18 @@ const getChecksParamValidation = joi.object({
305  300      });
306  301
```

```
307  302    const getChecksQueryValidation = joi.object({
308         -         type: joi.string().valid("http", "ping", "pagespeed", "hardware", "docker",
               "port", "distributed_http", "distributed_test"),
       303  +         type: joi
       304  +                 .string()
       305  +                 .valid(
       306  +                         "http",
       307  +                         "ping",
       308  +                         "pagespeed",
       309  +                         "hardware",
       310  +                         "docker",
       311  +                         "port",
       312  +                         "distributed_http",
       313  +                         "distributed_test"
       314  +                 ),
309    315            sortOrder: joi.string().valid("asc", "desc"),
310    316            limit: joi.number(),
311    317            dateRange: joi.string().valid("recent", "hour", "day", "week", "month", "all"),
           @@ -578,16 +584,15 @@ const triggerNotificationBodyValidation = joi.object({
578    584
579    585    const createAnnouncementValidation = joi.object({
580    586            title: joi.string().required().messages({
581         -            'string.empty': 'Title cannot be empty',
582         -            'any.required': 'Title is required',
       587  +                 "string.empty": "Title cannot be empty",
       588  +                 "any.required": "Title is required",
583    589            }),
584    590            message: joi.string().required().messages({
585         -            'string.empty': 'Message cannot be empty',
586         -            'any.required': 'Message is required',
       591  +                 "string.empty": "Message cannot be empty",
       592  +                 "any.required": "Message is required",
587    593            }),
588    594            userId: joi.string().required(),
589         -   });
590         -
       595  + });
591    596
592    597    export {
593    598            roleValidatior,
           @@ -651,5 +656,5 @@ export {
651    656            imageValidation,
```

```
652  657            triggerNotificationBodyValidation,
653  658            webhookConfigValidation,
654       -         createAnnouncementValidation
     659  +         createAnnouncementValidation,
655  660      };
```

Comments 0

Please sign in to comment.