**Relatório Autenticação EA-975 Grupo 2**

Henrique Roberto da Cunha Junior - 174638 | Leonardo Rodrigues Marques - 178610

No código do exemplo fornecido pelo Professor, nós tivemos um pequeno trecho de Login. A verificação do usuário e senha não é funcional e efetiva, pois o trecho que verifica a validade das credenciais está *hardcoded* no código. Esse método de autenticação não é recomendado, pois além de se tratar de uma falha de segurança grave, os dados não são buscados de fontes externas ou gerados em tempo de execução. Portanto, para modificá-los, é necessário editar o código-fonte e compilar o executável, o que é muito dispendioso.

No código fornecido abaixo, é possível ler explicitamente as credenciais de validação.

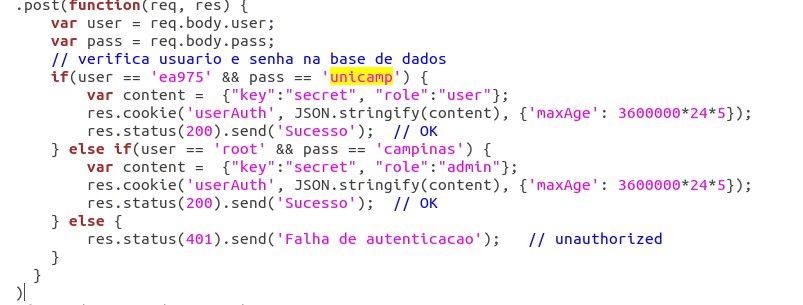


Figura 1: arquivo app.js com método hardcoded de autenticação.

Para solucionar esse problema, é necessário que a credenciais de autenticação sejam recuperadas de um banco de dados. No caso da Framework MEAN, deve-se utilizar uma função de requisição que consulte o usuário e senha do MongoDB. Portanto, duas edições de código nos arquivos app.js e models/mongo2.js e uma alteração no banco de dados foram realizadas. No arquivo models/mongo2.js, definiu-se um novo tipo de estrutura Schema e criou-se um módulo de exportação com conn2 para dar acesso externo ao caminho /usuarios contendo as credenciais de autenticação.

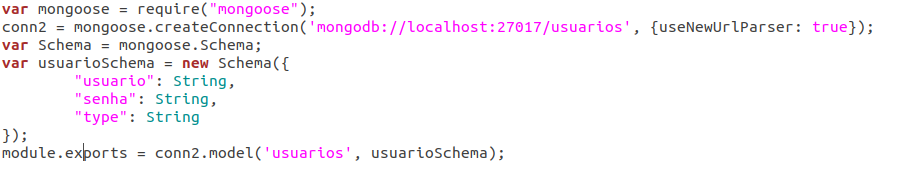


Figura 2: arquivo mongo2.js.

No arquivo app.js, a função .post foi reestruturada. A requisição findOne() foi usada para acessar o banco de dados MongoDB e consultar a existência do usuário e sua permissão(admin ou padrão).

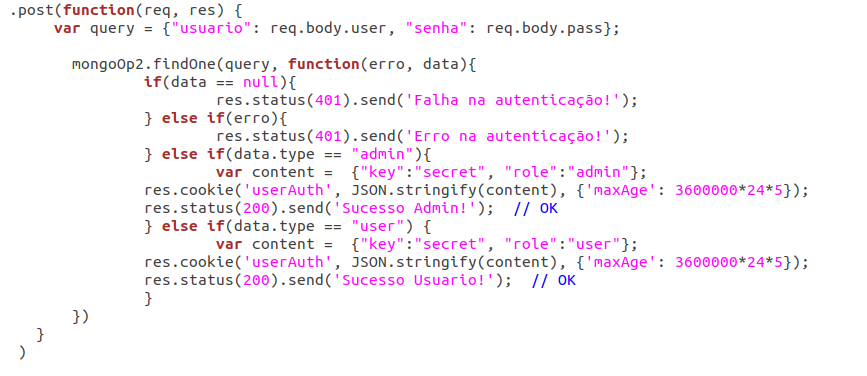


Figura 3: arquivo app.js com consulta no banco de dados MongoDB.

No MongoDB, foi criado um banco usuarios com collections usuarios para armazenar os dois usuários e suas credenciais, como é possível observar na figura.

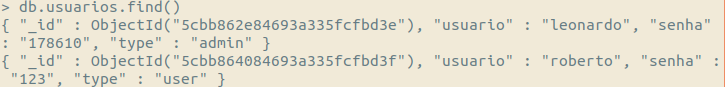
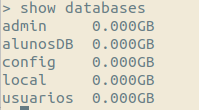


Figura 4: banco usuarios com 2 usuários adicionados.