02

Mãos à obra: Alterando requisição para POST

Vamos mudar o formulário para que os parâmetros do email e da senha do usuário sejam enviadas no corpo da requisição através da requisição POST. Para isso vá até a jsp usuário e procure a tag referente ao formulário de login e troque o método da requisição do tipo GET para que seja do tipo POST

```
<form id="login-form" action="${s:mvcUrl('UC#login').build()}"
    method="post" role="form" style="display: block;">
```

Na sequência, vamos alterar o método login da classe **UsuarioController** para que o acesso ao endereço login seja feito somente com as requisições do tipo POST.

```
@RequestMapping(value="/login",method=RequestMethod.POST)
public String login(@ModelAttribute("usuario") Usuario usuario,
        RedirectAttributes redirect, Model model, HttpSession session) {

    Usuario usuarioRetornado = dao.procuraUsuario(usuario);
    model.addAttribute("usuario", usuarioRetornado);
    if (usuarioRetornado == null) {
        redirect.addFlashAttribute("mensagem", "Usuário não encontrado");
        return "redirect:/usuario";
    }

    session.setAttribute("usuario", usuarioRetornado);
    return "usuarioLogado";

}
```

Teste novamente a aplicação no Kali Linux e faça o login com o usuário do Alex (**E-mail:** alex@gmail.com, **senha:** 123). Confirme que agora os parâmetros do e-mail e da senha não estão presentes na URL.

Agora vamos fazer o teste do SQLMAP. Por padrão, o SQLMAP irá retornar os resultados do último teste realizado, remova os testes previamente realizados com o comando:

```
rm -r /root/.sqlmap/output/
```

Agora que os testes anteriores foram removidos, vamos fazer um novo teste com o SQLMAP com a requisição do tipo POST para vermos se ainda conseguimos obter todos os usuários da tabela usuario do banco owasp:

```
sqlmap -u "[url após login com requisição POST]" --dump -T usuario -D owasp --data="alex@gmail.
```

Qual foi o resultado? Ainda foi possível visualizar as entradas presentes na tabela Usuario?