1- *Injection*

Pergunta 1.1

Colocando alguns nomes como Jones, Williams, Johnson entre outros não aparece resultados. No entanto, colocando a string Smith aparecem dois resultados.

Segundo o que aprendemos pela Experiência 1.1 é que passando uma tautologia ocorre uma SQL Injection. Assim, basta passar **Smith' or '1'='1**, porque quando estão a ver o last name, comparam 'Smith' or '1'='1'.

Pergunta 1.2

Usando os nomes lá apresentados como opções, Houston, New York, Seattle e Columbia apenas aparece um resultado.

No entanto seguindo a ideia da tautologia basta no código onde aparece statin=101 mudar para **station=101 or 1=1**.

Pergunta 1.3

Ao colocarmos o ID 101 a aplicação devolve os dados do empregado 101.

Para colocar um salário mais elevado basta fazer ID; UPDATE table_name SET column1=value, isto é, **101; UPDATE employee SET salary=90000**.

2- XSS

Pergunta 2.1

No formulário de compras o utilizador pode escolher a quantidade que quer de cada produtos apenas colocando essa quantidade no espaço 'Quantity'.

Os únicos campos com potencial para Reflected XSS seriam os "Enter your credit card number"ou "Enter your three digit acess code". Por tentativa e erro, descobrimos que ao colocar letras e dígitos à sorte no "Enter your credit card number"este não é verificado e por isso ele aceita qualquer coisa, no entanto ao fazer o mesmo para o "Enter your three digit acess code"ele queixa-se ao dizer que não são 3 dígitos. Com isto, descobrimos que se colocarmos a script <script> alert("SSA!!!")</script> aqui então temos que este campo tem potencial Reflected XSS.

Pergunta 2.2

No texto da mensagem podemos escrever apenas uma string, por exemplo, "Olá"e aparece-nos essa mensagem quando a vamos ler. No entanto se colocarmos algo do tipo "<script>A</script>"não nos aparece nada no corpo da mensagem. Se fizermos isto para o título acontece que nem sequer aparece essa mensagem para a ler. Assim, temos que o campo onde se escreve o corpo da mensagem tem potencial para Stored XSS.

Ao colocar a scritp <script language="javascript"type="text/javascript">alert(document.cookie);</script> no corpo da mensagem, quando se tenta ler o conteúdo da mensagem aparece uma janela em branco.

Um atacante pode beneficiar desta vulnerabilidade pois pode por outra pessoa que leia a mensagem a descarregar uma página indesejada ou um conteúdo indesejado.

3- Quebra na Autenticação

Pergunta 3.1

Ao usar o username "webgoat"e a cor "red"dá-nos a password deste utilzador, que é **"webgoat"**.

Normalmente, num sistema, exite o root, bin, admin, entre outros. Para comprometer o utilizador admin basta usar a cor "green".