

Aplicar filtros a consultas SQL

Descrição do projeto

Minha organização está trabalhando para tornar seu sistema mais seguro. É meu trabalho garantir que o sistema seja seguro, investigar todos os possíveis problemas de segurança e atualizar os computadores dos funcionários conforme necessário. As etapas a seguir fornecem exemplos de como usei SQL com filtros para executar tarefas relacionadas à segurança.

Recuperar após o expediente tentativas de login com falha

Houve um possível incidente de segurança que ocorreu após o horário comercial (após as 18:00). Todas as tentativas de login após o expediente que falharam precisam ser investigadas.

O código a seguir demonstra como criei uma consulta SQL para filtrar tentativas de login com falha que ocorreram após o horário comercial.

```
MariaDB [organization]> SELECT *
-> FROM log_in_attempts
-> WHERE login_time > '18:00' AND success = FALSE;
```

event_id	username	login_date	login_time	country	ip_address	success
2	apatel	2022-05-10	20:27:27	CAN	192.168.205.12	0
18	pwashing	2022-05-11	19:28:50	US	192.168.66.142	0
20	tshah	2022-05-12	18:56:36	MEXICO	192.168.109.50	0

A primeira parte da captura de tela é minha consulta e a segunda parte é uma parte da saída. Essa consulta filtra as tentativas de login com falha que ocorreram após as 18:00. Primeiro, comecei selecionando todos os dados da `tabela log_in_attempts`. Em seguida, usei uma `cláusula WHERE` com um operador `AND` para filtrar meus resultados para gerar apenas tentativas de login que ocorreram após as 18:00 e não foram bem-sucedidas. A primeira condição é `login_time > '18:00'`, que filtra as tentativas de login que ocorreram após as 18:00. A segunda condição é `success = FALSE`, que filtra as tentativas de login com falha.

Recupere tentativas de login em datas específicas

Um evento suspeito ocorreu em 09/05/2022. Qualquer atividade de login que aconteceu em 09/05/2022 ou no dia anterior precisa ser investigada.

O código a seguir demonstra como criei uma consulta SQL para filtrar tentativas de login que ocorreram em datas específicas.

```
MariaDB [organization]> SELECT *
-> FROM log_in_attempts
-> WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
```

event_id	username	login_date	login_time	country	ip_address	success
1	jrafael	2022-05-09	04:56:27	CAN	192.168.243.140	0
3	dkot	2022-05-09	06:47:41	USA	192.168.151.162	0
4	dkot	2022-05-08	02:00:39	USA	192.168.178.71	0

A primeira parte da captura de tela é minha consulta e a segunda parte é uma parte da saída. Essa consulta retorna todas as tentativas de logon que ocorreram em 09/05/2022 ou 08/05/2022. Primeiro, comecei selecionando todos os dados da tabela `log_in_attempts`. Em seguida, usei uma cláusula `WHERE` com um operador `OR` para filtrar meus resultados para gerar apenas tentativas de login que ocorreram em 2022-05-09 ou 2022-05-08. A primeira condição é `login_date = '2022-05-09'`, que filtra os logins em 2022-05-09. A segunda condição é `login_date = '2022-05-08'`, que filtra os logins em 2022-05-08.

Recuperar tentativas de login fora do México

Depois de investigar os dados da organização sobre tentativas de login, acredito que haja um problema com as tentativas de login que ocorreram fora do México. Essas tentativas de login devem ser investigadas.

O código a seguir demonstra como criei uma consulta SQL para filtrar tentativas de logon que ocorreram fora do México.

```
MariaDB [organization]> SELECT *
-> FROM log_in_attempts
-> WHERE NOT country LIKE 'MEX%';
```

event_id	username	login_date	login_time	country	ip_address	success
1	jrafael	2022-05-09	04:56:27	CAN	192.168.243.140	0
2	apatel	2022-05-10	20:27:27	CAN	192.168.205.12	0
3	dkot	2022-05-09	06:47:41	USA	192.168.151.162	0

A primeira parte da captura de tela é minha consulta e a segunda parte é uma parte da saída. Essa consulta retorna todas as tentativas de logon que ocorreram em países diferentes do México. Primeiro, comecei selecionando todos os dados da tabela `log_in_attempts`. Em seguida, usei uma cláusula `WHERE` com `NOT` para filtrar outros países além do México. Usei `LIKE` com `MEX%` como o padrão a ser correspondido porque o conjunto de dados representa o México como `MEX` e `MEXICO`. O sinal de porcentagem (%) representa qualquer número de caracteres não especificados quando usado com `LIKE`.

Recuperar funcionários no Marketing

Minha equipe deseja atualizar os computadores de determinados funcionários do departamento de Marketing. Para fazer isso, preciso obter informações sobre quais máquinas de funcionários atualizar.

O código a seguir demonstra como criei uma consulta SQL para filtrar máquinas de funcionários do departamento de Marketing no prédio Leste.

```
MariaDB [organization]> SELECT *  
  -> FROM employees  
  -> WHERE department = 'Marketing' AND office LIKE 'East%';
```

employee_id	device_id	username	department	office
1000	a320b137c219	elarson	Marketing	East-170
1052	a192b174c940	jdarosa	Marketing	East-195
1075	x573y883z772	fbautist	Marketing	East-267

A primeira parte da captura de tela é minha consulta e a segunda parte é uma parte da saída. Essa consulta retorna todos os funcionários do departamento de Marketing no prédio Leste. Primeiro, comecei selecionando todos os dados da tabela `employees`. Em seguida, usei uma cláusula `WHERE` com `AND` para filtrar os funcionários que trabalham no departamento de Marketing e no prédio Leste. Usei `LIKE` com `East%` como o padrão para correspondência porque os dados na coluna do `escritório` representam o prédio Leste com o número do escritório específico. A primeira condição é a parte `department = 'Marketing'`, que filtra os funcionários do departamento de marketing. A segunda condição é a parte do `escritório LIKE 'East%'`, que filtra os funcionários no prédio Leste.

Recuperar funcionários no Finance ou no Sales

As máquinas para funcionários nos departamentos financeiro e comercial também precisam ser atualizadas. Como é necessária uma atualização de segurança diferente, preciso obter informações sobre funcionários apenas desses dois departamentos.

O código a seguir demonstra como criei uma consulta SQL para filtrar máquinas de funcionários dos departamentos de Finanças ou Vendas.

```
MariaDB [organization]> SELECT *
-> FROM employees
-> WHERE department = 'Finance' OR department = 'Sales';
```

employee_id	device_id	username	department	office
1003	d394e816f943	sgilmore	Finance	South-153
1007	h174i497j413	wjaffrey	Finance	North-406
1008	i858j583k571	abernard	Finance	South-170

A primeira parte da captura de tela é minha consulta e a segunda parte é uma parte da saída. Essa consulta retorna todos os funcionários nos departamentos de Finanças e Vendas. Primeiro, comecei selecionando todos os dados da tabela `employees`. Em seguida, usei uma cláusula `WHERE` com `OR` para filtrar os funcionários que estão nos departamentos de Finanças e Vendas. Usei o operador `OR` em vez de `AND` porque quero todos os funcionários que estão em qualquer departamento. A primeira condição é `department = 'Finance'`, que filtra os funcionários do departamento financeiro. A segunda condição é `department = 'Sales'`, que filtra os funcionários do departamento de vendas.

Recuperar todos os funcionários que não estão em TI

Minha equipe precisa fazer mais uma atualização de segurança sobre os funcionários que não estão no departamento de Tecnologia da Informação. Para fazer a atualização, primeiro tenho que obter informações sobre esses funcionários.

A seguir, demonstramos como criei uma consulta SQL para filtrar máquinas de funcionários que não estão no departamento de Tecnologia da Informação.

```
MariaDB [organization]> SELECT *
-> FROM employees
-> WHERE NOT department = 'Information Technology';
```

employee_id	device_id	username	department	office
1000	a320b137c219	elarson	Marketing	East-170
1001	b239c825d303	bmoreno	Marketing	Central-276
1002	c116d593e558	tshah	Human Resources	North-434

A primeira parte da captura de tela é minha consulta e a segunda parte é uma parte da saída. A consulta retorna todos os funcionários que não estão no departamento de Tecnologia da Informação. Primeiro, comecei selecionando todos os dados da tabela `employees`. Em seguida, usei uma cláusula `WHERE` com `NOT` para filtrar os funcionários que não estão neste departamento.

Resumo

Apliquei filtros a consultas SQL para obter informações específicas sobre tentativas de login e máquinas de funcionários. Usei duas mesas diferentes, `log_in_attempts` e `funcionários`. Usei os operadores `AND`, `OR` e `NOT` para filtrar as informações específicas necessárias para cada tarefa. Também usei `LIKE` e o curinga do sinal de porcentagem (%) para filtrar padrões.