

# Apply filters to SQL queries

## Scenario

---

Review the scenario below. Then complete the step-by-step instructions.

You are a security professional at a large organization. Part of your job is to investigate security issues to help keep the system secure. You recently discovered some potential security issues that involve login attempts and employee machines.

Your task is to examine the organization's data in their **employees** and **log\_in\_attempts** tables. You'll need to use SQL filters to retrieve records from different datasets and investigate the potential security issues.

**Note:** This scenario involves the same queries as the ones the [Filter with AND, OR, and NOT](#) lab. You can revisit the lab to get screenshots to include in your portfolio document. If you choose, it's also possible to complete this activity without revisiting the lab by typing your queries in the template.

## Project description

I utilized SQL queries to investigate potential security issues related to login attempts and employee machines within a large organization. By analyzing data from the employees and log\_in\_attempts tables, I employed SQL filters to extract relevant records from different datasets. Through this process, I identified patterns and anomalies in login attempts that could indicate unauthorized access or suspicious activities, contributing to the organization's efforts to maintain a secure system environment. My findings and analysis were documented for reference and potential mitigation strategies.

Retrieve after hours failed login attempts

Query:

```
SELECT *  
FROM log_in_attempts  
WHERE login_time > '18:00' AND success = FALSE;
```

```
MariaDB [organization]>  
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
28	astrada	2022-05-09	19:28:12	MEXICO	192.168.27.57	0
34	drosas	2022-05-11	21:02:04	US	192.168.45.93	0
42	cgriffin	2022-05-09	23:04:05	US	192.168.4.157	0
52	cjackson	2022-05-10	22:07:07	CAN	192.168.58.57	0
69	wjaffrey	2022-05-11	19:55:15	USA	192.168.100.17	0
82	abernard	2022-05-12	23:38:46	MEX	192.168.234.49	0
87	apatel	2022-05-08	22:38:31	CANADA	192.168.132.153	0
96	ivelasco	2022-05-09	22:36:36	CAN	192.168.84.194	0
104	asundara	2022-05-11	18:38:07	US	192.168.96.200	0
107	bisles	2022-05-12	20:25:57	USA	192.168.116.187	0
111	astrada	2022-05-10	22:00:26	MEXICO	192.168.76.27	0
127	abellmas	2022-05-09	21:20:51	CANADA	192.168.70.122	0
131	bisles	2022-05-09	20:03:55	US	192.168.113.171	0
155	cgriffin	2022-05-12	22:18:42	USA	192.168.236.176	0
160	jclark	2022-05-10	20:49:00	CANADA	192.168.214.49	0
199	yappiah	2022-05-11	19:34:48	MEXICO	192.168.44.232	0

```
19 rows in set (0.002 sec)
```

This query retrieves all records from the "log\_in\_attempts" table where the "login\_time" is later than 6:00 PM (18:00) and the login attempt was unsuccessful (denoted by "success" being FALSE). It aims to identify and analyze login attempts that occurred in the evening hours and resulted in failures, potentially indicating suspicious or unauthorized activities during non-standard times.

Retrieve login attempts on specific dates

Query:

```
SELECT *  
FROM log_in_attempts  
WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
```

```
MariaDB [organization]>  
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	1
3	dkot	2022-05-09	06:47:41	USA	192.168.151.162	1
4	dkot	2022-05-08	02:00:39	USA	192.168.178.71	0
8	bisles	2022-05-08	01:30:17	US	192.168.119.173	0
12	dkot	2022-05-08	09:11:34	USA	192.168.100.158	1
15	lyamamot	2022-05-09	17:17:26	USA	192.168.183.51	0
24	arusso	2022-05-09	06:49:39	MEXICO	192.168.171.192	1
25	sbaelish	2022-05-09	07:04:02	US	192.168.33.137	1
26	apatel	2022-05-08	17:27:00	CANADA	192.168.123.105	1
28	aestrada	2022-05-09	19:28:12	MEXICO	192.168.27.57	0
30	yappiah	2022-05-09	03:22:22	MEX	192.168.124.48	1
32	acook	2022-05-09	02:52:02	CANADA	192.168.142.239	0
36	asundara	2022-05-08	09:00:42	US	192.168.78.151	1
38	sbaelish	2022-05-09	14:40:01	USA	192.168.60.42	1
39	yappiah	2022-05-09	07:56:40	MEXICO	192.168.57.115	1
42	cgriffin	2022-05-09	23:04:05	US	192.168.4.157	0
43	mcouliba	2022-05-08	02:35:34	CANADA	192.168.16.208	0
44	daquino	2022-05-08	07:02:35	CANADA	192.168.168.144	0

This SQL query retrieves all records from the "log\_in\_attempts" table where the "login\_date" matches either '2022-05-09' or '2022-05-08'. It aims to gather login attempt data for analysis specifically on these two dates, potentially to investigate any patterns or anomalies in the login activity that occurred on those days.

Retrieve login attempts outside of Mexico

Query:

```
SELECT *  
FROM log_in_attempts  
WHERE NOT country LIKE 'MEX%';
```

```
MariaDB [organization]>
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	1
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	1
4	dkot	2022-05-08	02:00:39	USA	192.168.178.71	0
5	jrafael	2022-05-11	03:05:59	CANADA	192.168.86.232	0
7	eraab	2022-05-11	01:45:14	CAN	192.168.170.243	1
8	bisles	2022-05-08	01:30:17	US	192.168.119.173	0
10	jrafael	2022-05-12	09:33:19	CANADA	192.168.228.221	0
11	sgilmore	2022-05-11	10:16:29	CANADA	192.168.140.81	0
12	dkot	2022-05-08	09:11:34	USA	192.168.100.158	1
13	mrah	2022-05-11	09:29:34	USA	192.168.246.135	1
14	sbaelish	2022-05-10	10:20:18	US	192.168.16.99	1
15	lyamamot	2022-05-09	17:17:26	USA	192.168.183.51	0
16	mcouliba	2022-05-11	06:44:22	CAN	192.168.172.189	1
17	pwashing	2022-05-11	02:33:02	USA	192.168.81.89	1
18	pwashing	2022-05-11	19:28:50	US	192.168.66.142	0
19	jhill	2022-05-12	13:09:04	US	192.168.142.245	1
21	iuduike	2022-05-11	17:50:00	US	192.168.131.147	1
25	sbaelish	2022-05-09	07:04:02	US	192.168.33.137	1
26	apatel	2022-05-08	17:27:00	CANADA	192.168.123.105	1

The query retrieves all records from the "log\_in\_attempts" table where the "country" column does not start with the abbreviation "MEX" (indicating Mexico). It filters out and excludes login attempts originating from Mexico, focusing on login data from other countries.

## Retrieve employees in Marketing

Query:

```
SELECT *
FROM employees
WHERE department = 'Marketing' AND office LIKE 'East%';
```

```
MariaDB [organization]>
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
1088	k865l965m233	rgosh	Marketing	East-157
1103	NULL	randerss	Marketing	East-460
1156	a184b775c707	dellery	Marketing	East-417
1163	h679i515j339	cwilliam	Marketing	East-216

This SQL query retrieves all records from the "employees" table where the employees belong to the "Marketing" department and are in offices with names that start with "East". The query is filtering and selecting employees who are both in the "Marketing" department and located in offices within the "East" region.

Retrieve employees in Finance or Sales

Query:

**SELECT \***

**FROM employees**

**WHERE department = 'Finance' OR department = 'Sales';**

```
MariaDB [organization]>
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
1009	NULL	lrodriqu	Sales	South-134
1010	k242l212m542	jlansky	Finance	South-109
1011	l748m120n401	drosas	Sales	South-292
1015	p611q262r945	jsoto	Finance	North-271
1017	r550s824t230	jclark	Finance	North-188
1018	s310t540u653	abellmas	Finance	North-403
1022	w237x430y567	arusso	Finance	West-465
1024	y976z753a267	iuduike	Sales	South-215
1025	z381a365b233	jhill	Sales	North-115
1029	d336e475f676	ivelasco	Finance	East-156
1035	j236k303l245	bisles	Sales	South-171
1039	n253o917p623	cjackson	Sales	East-378
1041	p929q222r778	cgriffin	Sales	North-208
1044	s429t157u159	tbarnes	Finance	West-415
1045	t567u844v434	pwashing	Finance	East-115
1046	u429v921w138	daquino	Finance	West-280
1047	v109w587x644	cward	Finance	West-373
1048	w157x502y275	twitkal	Finance	South-288

The query retrieves all records from the "employees" table where the department is either 'Finance' or 'Sales'. This means it will gather information about employees who work in either the Finance or Sales department, providing a list of employees associated with these specific departments.

Retrieve all employees not in IT

Query:

```
SELECT *  
FROM employees  
WHERE NOT department = 'Information Technology';
```

```
MariaDB [organization]>  
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
1003	d394e816f943	sgilmore	Finance	South-153
1004	e218f877g788	eraab	Human Resources	South-127
1005	f551g340h864	gesparza	Human Resources	South-366
1007	h174i497j413	wjaffrey	Finance	North-406
1008	i858j583k571	abernard	Finance	South-170
1009	NULL	lrodriqu	Sales	South-134
1010	k242l212m542	jlansky	Finance	South-109
1011	l748m120n401	drosas	Sales	South-292
1015	p611q262r945	jsoto	Finance	North-271
1016	q793r736s288	sbaelish	Human Resources	North-229
1017	r550s824t230	jclark	Finance	North-188
1018	s310t540u653	abellmas	Finance	North-403
1020	u899v381w363	arutley	Marketing	South-351
1022	w237x430y567	arusso	Finance	West-465
1024	y976z753a267	iuduke	Sales	South-215

The query retrieves all records from the "employees" table where the department is not labeled as "Information Technology." This SQL statement aims to gather data on employees who do not belong to the Information Technology department, potentially aiding in analyzing the composition and distribution of staff across various departments within the organization.

## Summary

I leveraged SQL queries to investigate potential security concerns involving login attempts and employee machines within a large organization. By filtering and analyzing data from the employees and log\_in\_attempts tables, I aimed to identify patterns of unauthorized access and anomalies in login behavior. Through careful examination of login records, cross-referencing with employee data, and collaborating with relevant stakeholders, I documented and presented my findings,

which included recommendations for enhancing security measures. The process involved querying employee information, scrutinizing suspicious login attempts, and assessing the relationship between login data and employee records. This endeavor contributed to the organization's efforts to bolster its system security and establish a proactive approach to identifying and addressing potential security threats.