Apply filters to SQL queries

Project description

As a security professional for a large organization, I have to investigate security issues to help keep my organization's system secure. I've recently discovered there are some potential security issues with login attempts and employee machines. I will look through data in `employees` and `log_in_attempts` tables, using SQL filters to investigate the potential security issues.

Retrieve after hours failed login attempts

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

- `SELECT` and `*` will display all columns in the table
- `FROM log_in_attempts` will display columns from this specific table
- `WHERE login_time > '18:00' AND success = 'FALSE';` will display rows only with login attempts after 6pm and failed login attempts
- `AND` is an operator that allows multiple conditions in a query to be combined, and will display records where all conditions are true

This is the result:

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	aestrada	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	aestrada	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

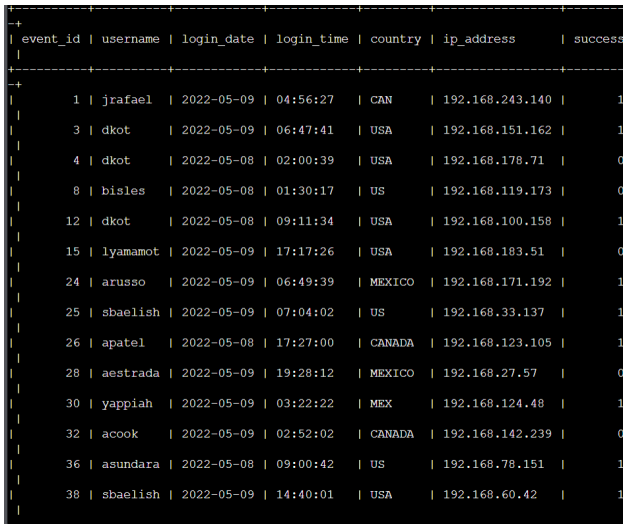
19 rows in set, 1 warning (0.286 sec)

Retrieve login attempts on specific dates

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

- A suspicious event occurred on September 5th, 2022 so I ant to look at all login attempts on that date and the date before
- `WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';` will display all of the login attempts on the date of the suspicious event and the day preceding that
- `OR` is an operator that allows multiple conditions in a query to be combined, and will display records where at least one of the conditions is true

This is the result:



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
163	tmitchel	2022-05-08	09:21:16	MEX	192.168.119.29	0
165	jreckley	2022-05-08	15:28:43	MEXICO	192.168.34.193	0
168	jlansky	2022-05-08	13:25:42	USA	192.168.210.94	1
169	alevitsk	2022-05-08	08:10:43	CANADA	192.168.210.228	0
170	sbaelish	2022-05-09	16:43:18	USA	192.168.65.113	0
172	mabadi	2022-05-08	08:06:50	US	192.168.180.41	1
178	sgilmore	2022-05-08	12:27:22	CAN	192.168.52.216	0
184	alevitsk	2022-05-08	03:09:48	CAN	192.168.33.70	0
186	bisles	2022-05-09	04:29:17	USA	192.168.40.72	0
187	arusso	2022-05-09	00:36:26	MEX	192.168.77.137	0
189	nmason	2022-05-08	05:37:24	CANADA	192.168.168.117	1
190	jsoto	2022-05-09	05:09:21	USA	192.168.25.60	0
191	cjackson	2022-05-08	06:46:07	CANADA	192.168.7.187	0
193	lrodriqu	2022-05-08	07:11:29	US	192.168.125.240	0
197	jsoto	2022-05-08	09:05:09	US	192.168.36.21	0

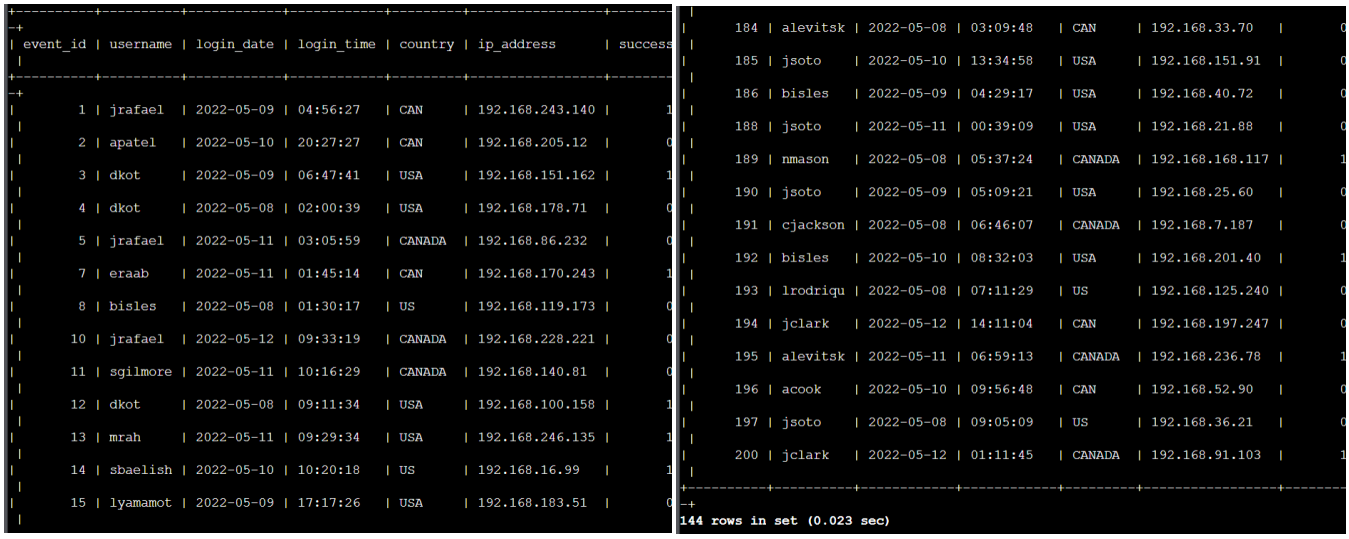
75 rows in set (0.075 sec)

Retrieve login attempts outside of Mexico

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

- It was determined that the suspicious activity with login attempts occurred outside of Mexico, so to narrow it down we want to look at all login attempts that happened from countries that are not Mexico
- `WHERE NOT country LIKE 'MEX%';` will display all country names in the country column that are not Mexico
- `NOT` is an operator that negates a condition

This is the result:



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
184	alevitsk	2022-05-08	03:09:48	CAN	192.168.33.70	0
185	jsoto	2022-05-10	13:34:58	USA	192.168.151.91	0
186	bisles	2022-05-09	04:29:17	USA	192.168.40.72	0
188	jsoto	2022-05-11	00:39:09	USA	192.168.21.88	0
189	nmason	2022-05-08	05:37:24	CANADA	192.168.168.117	1
190	jsoto	2022-05-09	05:09:21	USA	192.168.25.60	0
191	cjackson	2022-05-08	06:46:07	CANADA	192.168.7.187	0
192	bisles	2022-05-10	08:32:03	USA	192.168.201.40	1
193	lrodriqu	2022-05-08	07:11:29	US	192.168.125.240	0
194	jclark	2022-05-12	14:11:04	CAN	192.168.197.247	0
195	alevitsk	2022-05-11	06:59:13	CANADA	192.168.236.78	1
196	acook	2022-05-10	09:56:48	CAN	192.168.52.90	0
197	jsoto	2022-05-08	09:05:09	US	192.168.36.21	0
200	jclark	2022-05-12	01:11:45	CANADA	192.168.91.103	1

144 rows in set (0.023 sec)

Retrieve employees in Marketing

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

- My team wants to perform security updates on employee machines from the Marketing department that are located in offices inside the East building, so I need to filter through the employees table to find these specific employees
- `WHERE department = 'Marketing' AND office LIKE 'East%';` will display only employees from the Marketing department and in the East building office
- `LIKE` is an operator that searches for a specified pattern in a column

This is the result:

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	k8651965m233	rgosh	Marketing	East-157
1103	NULL	randerss	Marketing	East-460
1156	a184b775c707	dellery	Marketing	East-417
1163	h679i515j339	cwilliam	Marketing	East-216

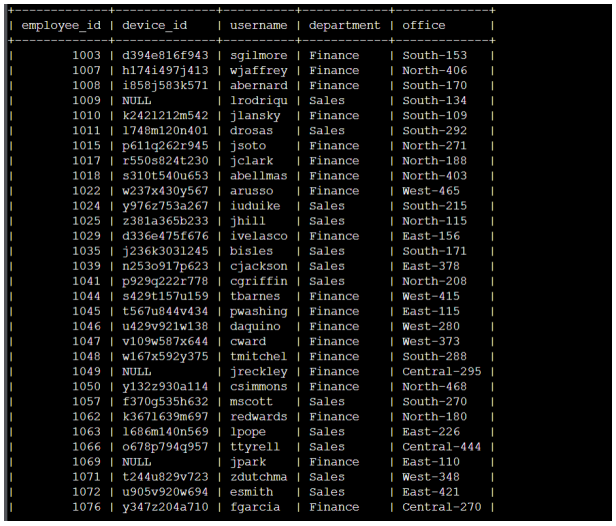
7 rows in set (0.001 sec)

Retrieve employees in Finance or Sales

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

- Now, I need to perform security updates for employee machines in Finance or Sales departments
- `WHERE department = 'Finance' OR department = 'Sales';` will display all employees present in these two departments

This is the result:



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	k2421212m542	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	j236k3031245	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	w167x592y375	tmitchel	Finance	South-288
1049	NULL	jreckley	Finance	Central-295
1050	y132z930a114	csimmons	Finance	North-468
1057	f370g535h632	mscott	Sales	South-270
1062	k3671639m697	redwards	Finance	North-180
1063	l686m140n569	lpope	Sales	East-226
1066	o678p794q957	ttyrell	Sales	Central-444
1069	NULL	jpark	Finance	East-110
1071	t244u829v723	zdutchma	Sales	West-348
1072	u905v920w694	esmith	Sales	East-421
1076	y347z204a710	fgarcia	Finance	Central-270
1098	u671v146w618	tarchamb	Sales	North-423
1099	v283w690x104	anaser	Finance	West-357
1105	b551c837d758	kmei	Finance	Central-232
1107	d168e758f876	akajwara	Sales	North-471
1109	f229g533h679	nlocklea	Sales	East-196
1110	g567h376i314	pchaudhu	Sales	Central-428
1111	h835i179j862	jlee	Sales	West-309
1116	m272n572o874	nzhao	Sales	South-100
1117	n683o758p820	dahmad	Sales	West-405
1118	o305p208q337	jpark3	Sales	South-329
1119	p164q780r999	omubarak	Sales	West-409
1121	r628s557t397	mrojas	Sales	East-288
1122	s103t952u851	btorres	Finance	West-319
1130	a317b635c465	tsnow	Sales	Central-451
1136	g299h520i457	jhawes	Finance	West-416
1138	i671j355k725	sromero	Finance	South-329
1142	m674n127o823	lsilva	Finance	East-440
1144	NULL	erobinso	Finance	Central-266
1147	r454s225t299	tvega	Finance	West-177
1148	s328t505u907	dharvey	Finance	South-181
1159	d881e710f732	jshen	Finance	East-193
1164	i682j513k442	fsmeltz	Finance	North-163
1169	NULL	mmitchel	Sales	Central-250
1174	s371t911u987	eortiz	Finance	North-428
1175	t959u687v394	jclark2	Finance	North-194
1176	u849v569w521	nliu	Sales	West-220
1181	z803a233b718	sessa	Finance	South-207
1185	d790e839f461	revens	Sales	North-330
1186	e281f433g404	sacosta	Sales	North-460
1187	f963g637h851	bbode	Finance	East-351
1188	g164h566i795	noshiro	Finance	West-252
1195	n516o853p957	orainier	Finance	East-346

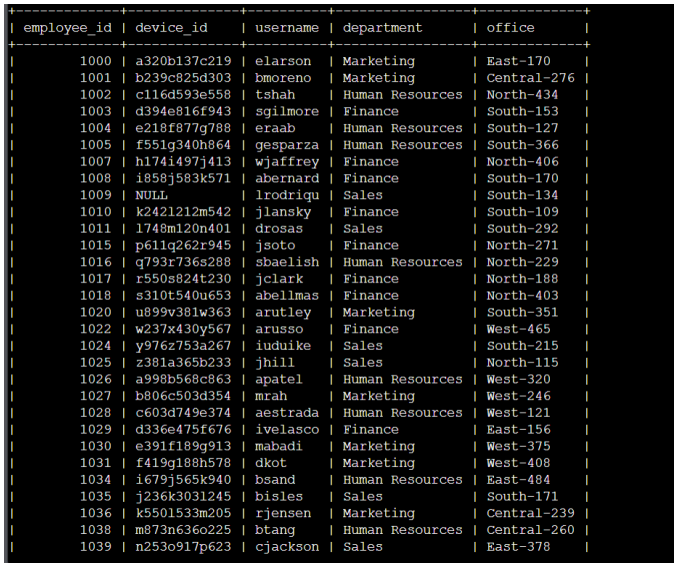
71 rows in set (0.201 sec)

Retrieve all employees not in IT

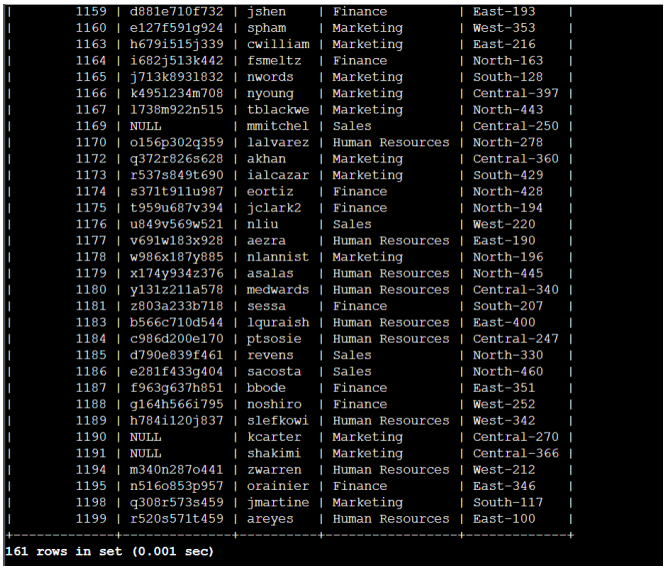
```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE NOT department = 'Information Technology';
```

- Lastly, I need to make one final update on employee machines, but the Information Technology department already received it
- `WHERE department = 'Finance' OR department = 'Sales';` will display all employees present in all departments except for the Information Technology department

This is the result:



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	k2421212m542	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	iuduike	Sales	South-215
1025	z381a365b233	jhill	Sales	North-115
1026	a998b568c863	apatel	Human Resources	West-320
1027	b806c503d354	mrah	Marketing	West-246
1028	c603d749e374	aestrada	Human Resources	West-121
1029	d336e475f676	ivelasco	Finance	East-156
1030	e391f189g913	mabadi	Marketing	West-375
1031	f419g188h578	dkot	Marketing	West-408
1034	i679j565k940	bsand	Human Resources	East-484
1035	j236k3031245	bisles	Sales	South-171
1036	k5501533m205	rjensen	Marketing	Central-239
1038	m873n636o225	btang	Human Resources	Central-260
1039	n253o917p623	cjackson	Sales	East-378



1159	d881e710f732	jshen	Finance	East-193
1160	e127f591g924	spham	Marketing	West-353
1163	h679i515j339	cwilliam	Marketing	East-216
1164	i682j513k442	fsmeltz	Finance	North-163
1165	j713k8931832	nwords	Marketing	South-128
1166	k4951234m708	nyoung	Marketing	Central-397
1167	l738m922n515	tblackwe	Marketing	North-443
1169	NULL	mmitchel	Sales	Central-250
1170	o156p302q359	lalvarez	Human Resources	North-278
1172	q372r826s628	akhan	Marketing	Central-360
1173	r537s849t690	ialcazar	Marketing	South-429
1174	s371t911u987	eortiz	Finance	North-428
1175	t959u687v394	jclark2	Finance	North-194
1176	u849v569w521	nliu	Sales	West-220
1177	v691w183x928	aezra	Human Resources	East-190
1178	w986x187y885	nlannist	Marketing	North-196
1179	x174y934z376	asalas	Human Resources	North-445
1180	y131z211a578	medwards	Human Resources	Central-340
1181	z803a233b718	sessa	Finance	South-207
1183	b566c710d544	lquraish	Human Resources	East-400
1184	c986d200e170	ptsosie	Human Resources	Central-247
1185	d790e839f461	revens	Sales	North-330
1186	e281f433g404	sacosta	Sales	North-460
1187	f963g637h851	bbode	Finance	East-351
1188	g164h566i795	noshiro	Finance	West-252
1189	h784i120j837	slefkowi	Human Resources	West-342
1190	NULL	kcarter	Marketing	Central-270
1191	NULL	shakimi	Marketing	Central-366
1194	m340n287o441	zwarren	Human Resources	West-212
1195	n516o853p957	orainier	Finance	East-346
1198	q308r573s459	jmartine	Marketing	South-117
1199	r520s571t459	areyes	Human Resources	East-100

161 rows in set (0.001 sec)

Summary

First, I collected data about login attempts related to suspicious activity in our organization's system. Next, I collected data about employees in different departments, so I could make the proper security updates for their machines to help respond to the aforementioned suspicious activity. This was done using SQL.