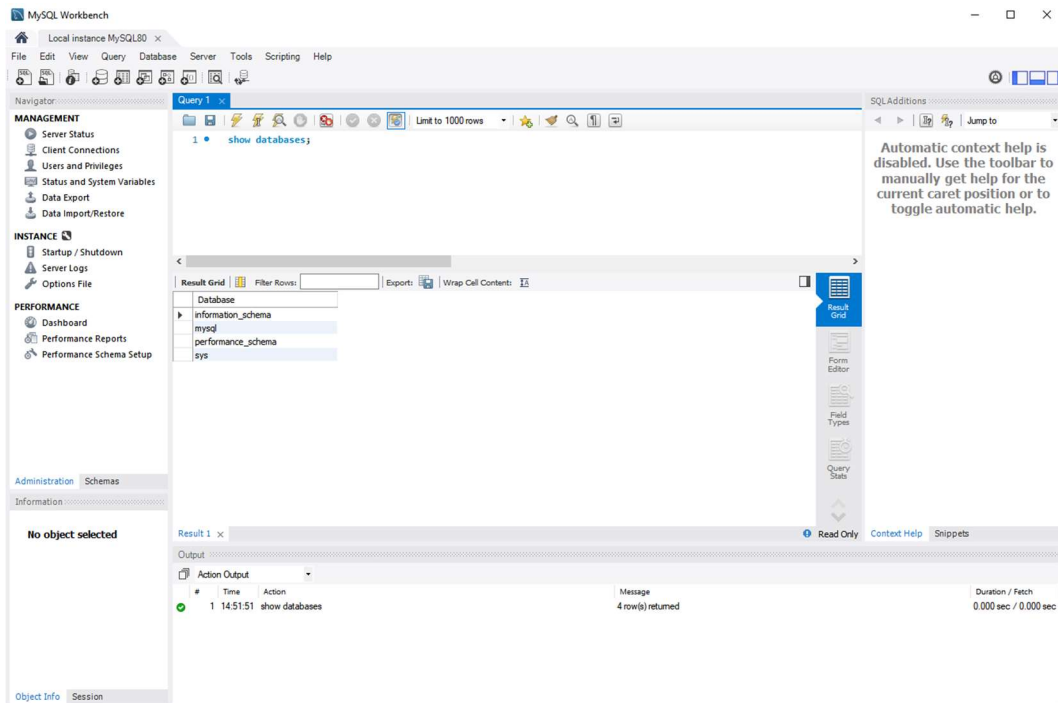


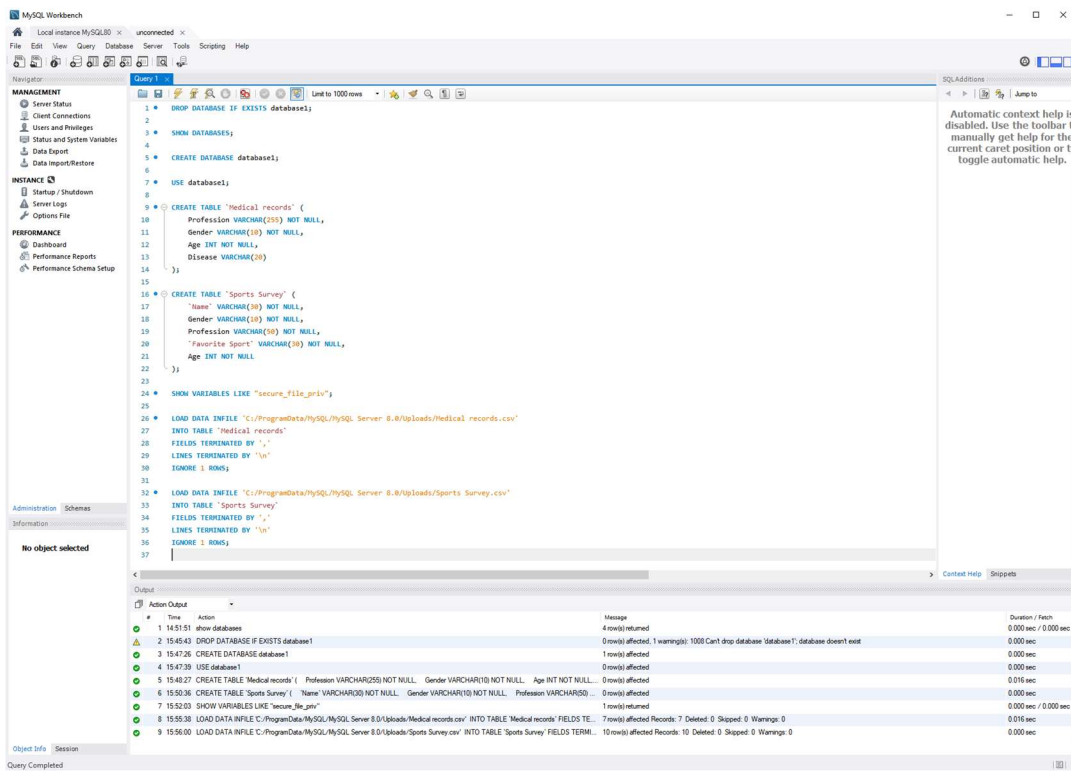
Lab 1 – Linkage Attack

Gunnar Yonker

MySQL Setup and Running:



Data Loaded:



Lab 1 – Linkage Attack

Gunnar Yonker

Examples of Manual Query Matches:

45 • `SELECT * FROM `Sports Survey` WHERE AGE = 29 AND Gender = 'Male' AND Profession = 'Artist';`
46

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	Name	Gender	Profession	Favorite Sport	Age
▶	Frank Sierra	Male	Artist	Football	29

45 • `SELECT * FROM `Sports Survey` WHERE AGE = 39 AND Gender = 'Female' AND Profession = 'Bank Manager';`
46

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	Name	Gender	Profession	Favorite Sport	Age
▶	Angie Griffith	Female	Bank Manager	Football	39

45 • `SELECT * FROM `Sports Survey` WHERE AGE = 27 AND Gender = 'Male' AND Profession = 'Mechanic';`
46

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	Name	Gender	Profession	Favorite Sport	Age
▶	Steve Schez	Male	Mechanic	Football	27

Manually searching for values that match between the Sports Survey and Medical record csv tables works in finding the matches as seen above. However, it is inefficient, especially as the datasets become larger.

Lab 1 – Linkage Attack

Gunnar Yonker

Linkage Attack That Automatically Finds Matches:

The screenshot displays a database management system interface. At the top, a SQL query is entered in a text area:

```
45 SELECT * FROM 'Sports Survey' WHERE AGE = 33 AND Gender = 'Male' AND Profession = 'Cashier';
46
47 SELECT * FROM 'Sports Survey' INNER JOIN 'Medical Records' ON
48 'Sports Survey'.Age = 'Medical records'.Age AND
49 'Sports Survey'.Gender = 'Medical records'.Gender AND
50 'Sports Survey'.Profession = 'Medical records'.Profession;
51
```

Below the query editor, a 'Result Grid' is shown with a table of data. The table has columns: Name, Gender, Profession, Favorite Sport, Age, Profession, Gender, Age, Disease. The data rows are:

Name	Gender	Profession	Favorite Sport	Age	Profession	Gender	Age	Disease
Angie Griffith	Female	Bank Manager	Football	39	Bank Manager	Female	39	Asthma
Frank Sierra	Male	Artist	Football	29	Artist	Male	29	Tuberculosis
Steve Sanchez	Male	Mechanic	Football	27	Mechanic	Male	27	Flu

At the bottom, an 'Action Output' pane shows a log of executed queries and their results. The log includes the following entries:

#	Time	Action	Message	Duration / Fetch
1	16:02:27	SELECT * FROM 'Sports Survey' WHERE AGE = 31 AND Gender = 'Male' AND Profession = 'Sales Executive' LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
2	16:03:03	SELECT * FROM 'Sports Survey' WHERE AGE = 35 AND Gender = 'Female' AND Profession = 'Teacher' LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
3	16:03:28	SELECT * FROM 'Sports Survey' WHERE AGE = 29 AND Gender = 'Male' AND Profession = 'Artist' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
4	16:06:00	SELECT * FROM 'Sports Survey' WHERE AGE = 39 AND Gender = 'Female' AND Profession = 'Bank Manager' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
5	16:06:26	SELECT * FROM 'Sports Survey' WHERE AGE = 31 AND Gender = 'Female' AND Profession = 'Sales Executive' LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
6	16:06:44	SELECT * FROM 'Sports Survey' WHERE AGE = 27 AND Gender = 'Male' AND Profession = 'Mechanic' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
7	16:07:05	SELECT * FROM 'Sports Survey' WHERE AGE = 33 AND Gender = 'Male' AND Profession = 'Cashier' LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
8	16:10:10	SELECT * FROM 'Sports Survey' INNER JOIN 'Medical Records' ON 'Sports Survey'.Age = 'Medical records'.Age LIMIT 0, 1000	9 row(s) returned	0.000 sec / 0.000 sec
9	16:14:05	SELECT * FROM 'Sports Survey' INNER JOIN 'Medical Records' ON 'Sports Survey'.Age = 'Medical records'.Age AND 'Sports Survey'.Gender = 'Medical records'.Gender LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

For this query, the two tables are put together using the INNER JOIN command so that matches can automatically be found based on certain criteria. The criteria that is used is looking for matching age, gender, and profession as those are the columns that both of the databases have in common. If the age, gender, and profession match between the two tables it is highly likely that they are attributes of the same person. This information can then be used to match the values and provide sensitive information using the general information from the Sports Survey. The three matches that were found in this linkage attack show that using the datasets and matching the values, I was able to find the disease and names for these three individuals:

Angie Griffith, Female, Bank Manager, Football, 39, Bank Manager, Female, 39, Asthma

Frank Sierra, Male, Artist, Football, 29, Artist, Male, 29, Tuberculosis

Steve Sanchez, Male, Mechanic, Football, 27, Mechanic, Male, 27, Flu

The information gained by executing these queries is the name and disease, sensitive information, of these individuals based on the matching attributes between the Sports Survey and Medical records tables.

Conclusion:

With the provided csv files of Sports Survey and Medical records, I was able to carry out an effective linkage attack that resulted in finding sensitive information of 3 individuals. The Sports Survey table contained names, ages, professions, and gender information which is not directly sensitive information. The Medical records table contained ages, gender, profession, and disease information of which the disease information is sensitive. Using a linkage attack, the tables could be joined and then queried based on matching criteria that the two tables shared. The matching criteria queried for was age, gender, and profession. These are attributes that if they matched, it is highly likely that the information associated with the matched attributes could be tied to a single individual. The queries resulted in 3 matches that allowed the sensitive information of Angie Griffith, Frank Sierra, and Steve Sacher to be discovered. This is a good example of a situation where information gathered in the Sports Survey that seemed harmless, was able to be used in a linkage attack in conjunction with the Medical records table to match the name and disease information to those previously mentioned individuals.

My important takeaway from this lab is that it is crucial to be conscious of what information you are giving out in surveys because something as simple as a sports survey in this situation was then used to figure out sensitive information from a table that provides some anonymity if it was the only table compromised. Just because anonymity is promised in one data collection, doesn't mean it is guaranteed because attacks such as the linkage attack practiced here proves that sensitive data can be uncovered using seemingly innocent data. Another takeaway for me personally is that I understand SQL a lot more now than I did before this lab. I didn't really have any knowledge of how to make queries and following along with the video and then making my own queries I felt like I learned a lot. I now also know what the back tick is and its use in SQL.