16/3/2017

KTMB
PENETRATION REPORT

PENTESTER: KASPAROV1996
EMAIL: APOPTOXIN4869@PROTONMAIL.COM

1.0 SUMMARY

This penetration test was conducted solely by the author without any intention to cause any damages to KTMB. All activities were conducted in a manner that simulated a malicious actor engaged in a targeted attack against KTMB with the goals of:

i)Identifying if a remote attacker could penetrate KTMB's defenses
ii)Determining the impact of a security breach on:
-Confidentiality of the company's private data
- Internal infrastructure and availability of KTMB's information systems

Efforts were placed on the identification and exploitation of security weaknesses that could allow a remote attacker to gain unauthorized access to organizational data. The attacks were conducted with the level of access that a general Internet user would have. The assessment was conducted in accordance with the recommendations outlined in NIST SP 800-1151 with all tests and actions being conducted under controlled conditions.

REF: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf

2.0 RESULT OF PENETRATION TEST

There was three domains owned by KTMB was tested. After a deep examination, there was two critical flaws was found. The first identified flaw is SQL- Injection and the second one is direct access to file upload link.

2.1 TESTED WEBSITE(S)

I) www.ktmb.com.my
II) www.intranet4.ktmb.com.my
III) www.intranet3.ktmb.com.my

2.2 BRIEF DEFINITION

SQL-INJECTION

SQL Injection (SQLi) refers to an injection attack wherein an attacker can execute malicious SQL statements (also commonly referred to as a malicious *payload*) that control a web application's database server (also commonly referred to as a *Relational Database Management System – RDBMS*). Since an SQL Injection vulnerability could possibly affect any website or web application that makes use of an SQL-based database, the vulnerability is one of the oldest, most prevalent and most dangerous of web application vulnerabilities.

By leveraging an SQL Injection vulnerability, given the right circumstances, an attacker can use it to bypass a web application's authentication and authorization mechanisms and retrieve the contents of an entire database. SQL Injection can also be used to add, modify and delete records in a database, affecting data integrity.To such an extent, SQL Injection can provide an attacker with unauthorized access to sensitive data including, customer data, personally identifiable information (PII), trade secrets, intellectual property and other sensitive information.

UNSECURE DIRECTORY LISITING

Properly controlling access to web content is crucial for running a secure web server. Directory traversal is an HTTP exploit which allows attackers to access restricted directories and execute commands outside of the web server’s root directory.

Web servers provide two main levels of security mechanisms

- Access Control Lists (ACLs)
- Root directory

An Access Control List is used in the authorization process. It is a list which the web server’s administrator uses to indicate which users or groups are able to access, modify or execute particular files on the server, as well as other access rights.

3.0 PROOF OF CONCEPT

3.0.1 SQL- INJECTION

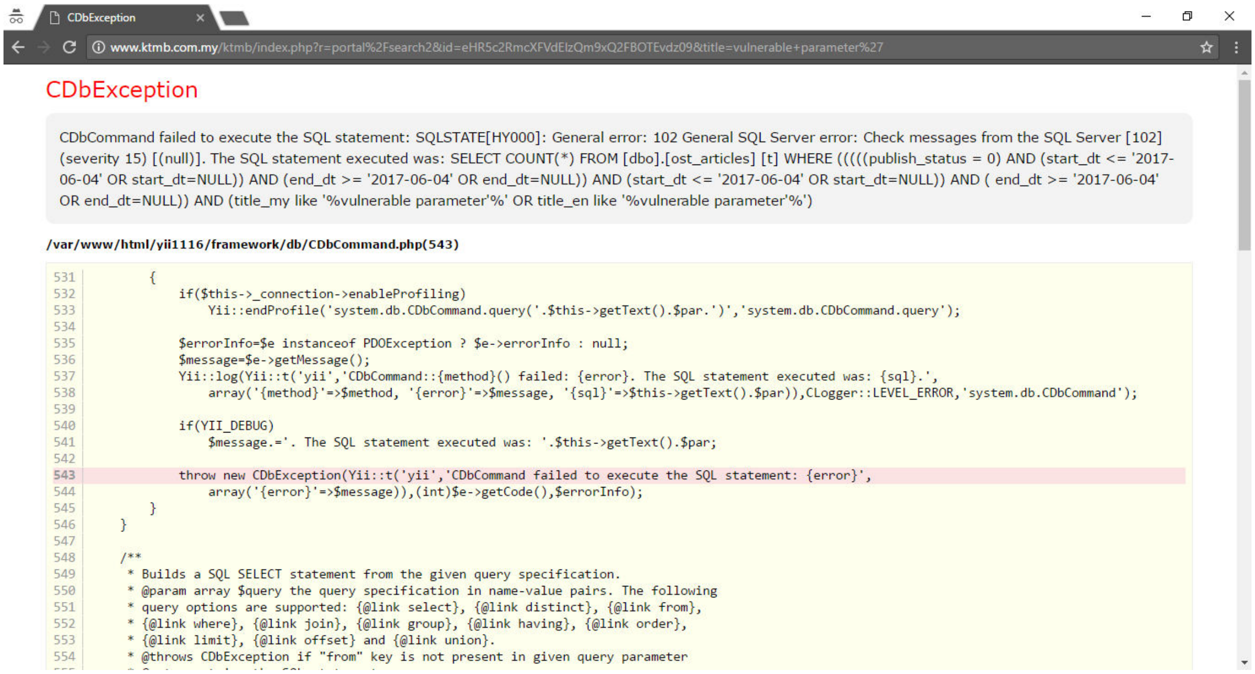
1.0

www.ktmb.com.my

SEVERITY = HIGH

http://www.ktmb.com.my/ktmb/index.php?r=portal%2Fsearch2&id=eHR5c2RmcXFVdElzQm9xQ2FBOTEvdz09&title=vulnerable+parameter

NOTICE THAT AFTER ADDING SINGLE QUOTE AFTER THE TEXT WILL CAUSE THE WEBPAGE ERROR.



www.ktmb.com.my/ktmb/index.php?r=portal%2Fsearch2&id=eHR5c2RmcXFVdElzQm9xQ2FBOTEvdz09&title=vulnerable+parameter%27

CDbException

CDbCommand failed to execute the SQL statement: SQLSTATE[HY000]: General error: 102 General SQL Server error: Check messages from the SQL Server [102] (severity 15) [(null)]. The SQL statement executed was: SELECT COUNT(*) FROM [dbo].[ost_articles] [t] WHERE (((((publish_status = 0) AND (start_dt <= '2017-06-04' OR start_dt=NULL)) AND (end_dt >= '2017-06-04' OR end_dt=NULL)) AND (start_dt <= '2017-06-04' OR start_dt=NULL)) AND (end_dt >= '2017-06-04' OR end_dt=NULL)) AND (title_my like '%vulnerable parameter'%' OR title_en like '%vulnerable parameter'%')

/var/www/html/yii1116/framework/db/CDbCommand.php(543)

```
        {
            if($this->_connection->enableProfiling)
                Yii::endProfile('system.db.CDbCommand.query('.$this->getText().$par.')','system.db.CDbCommand.query');

            $errorInfo=$e instanceof PDOException ? $e->errorInfo : null;
            $message=$e->getMessage();
            Yii::log(Yii::t('yii','CDbCommand::{method}() failed: {error}. The SQL statement executed was: {sql}.',
                array('{method}'=>$method, '{error}'=>$message, '{sql}'=>$this->getText().$par)),CLogger::LEVEL_ERROR,'system.db.CDbCommand');

            if(YII_DEBUG)
                $message.='. The SQL statement executed was: '.$this->getText().$par;

            throw new CDbException(Yii::t('yii','CDbCommand failed to execute the SQL statement: {error}',
                array('{error}'=>$message)),(int)$e->getCode(),$errorInfo);
        }
    }

    /**
     * Builds a SQL SELECT statement from the given query specification.
     * @param array $query the query specification in name-value pairs. The following
     * query options are supported: {@link select}, {@link distinct}, {@link from},
     * {@link where}, {@link join}, {@link group}, {@link having}, {@link order},
     * {@link limit}, {@link offset} and {@link union}.
     * @throws CDbException if "from" key is not present in given query parameter
```

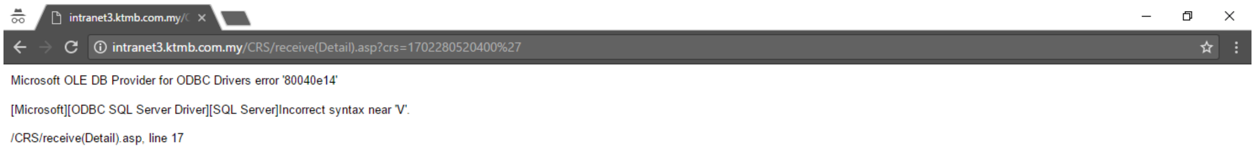
Indeed this is a SQL error that can lead to SQL Injection attack.

2.0

www.intranet3.ktmb.com.my

SEVERITY = HIGH

http://intranet3.ktmb.com.my/CRS/receive(Detail).asp?crs=1702280520400


intranet3.ktmb.com.my/(
intranet3.ktmb.com.my/CRS/receive(Detail).asp?crs=1702280520400%27

```
Microsoft OLE DB Provider for ODBC Drivers error '80040e14'

[Microsoft][ODBC SQL Server Driver][SQL Server]Incorrect syntax near 'V'.

/CRS/receive(Detail).asp, line 17
```

WEB SERVER: Windows 2000

WEB APPLICATION TECHNOLOGY: ASP.NET, ASP, Microsoft IIS 5.0

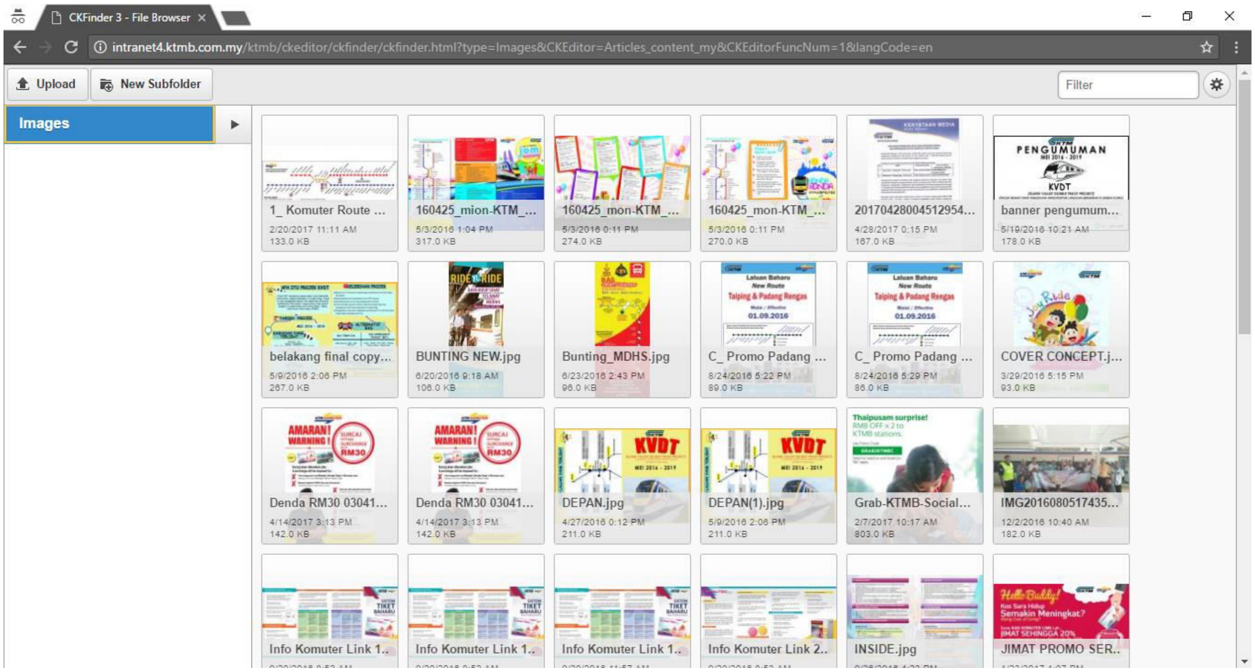
BACK-END DBMS: Microsoft SQL Server 2012

16/3/2017

UNSECURE DIRECTORY LISITING

3.0

http://intranet4.ktmb.com.my/ktmb/ckeditor/ckfinder/ckfinder.html?type=Images&CKEditor=Articles_content_my&CKEditorFuncNum=1&langCode=en



intranet4.ktmb.com.my/ktmb/ckeditor/ckfinder/ckfinder.html?type=Images&CKEditor=Articles_content_my&CKEditorFuncNum=1&langCode=en

1_ Komuter Route ...
2/20/2017 11:11 AM
133.0 KB

160425_mion-KTM_...
5/3/2016 1:04 PM
317.0 KB

160425_mon-KTM_...
5/3/2016 0:11 PM
274.0 KB

160425_mon-KTM_...
5/3/2016 0:11 PM
270.0 KB

20170428004512954...
4/28/2017 0:15 PM
167.0 KB

banner pengumum...
5/19/2016 10:21 AM
178.0 KB

belakang final copy...
5/9/2016 2:06 PM
267.0 KB

BUNTING NEW.jpg
6/20/2016 9:18 AM
106.0 KB

Bunting_MDHS.jpg
6/23/2016 2:43 PM
96.0 KB

C_ Promo Padang ...
8/24/2016 5:22 PM
89.0 KB

C_ Promo Padang ...
8/24/2016 5:29 PM
86.0 KB

COVER CONCEPT.j...
3/29/2016 5:15 PM
93.0 KB

Denda RM30 03041...
4/14/2017 3:13 PM
142.0 KB

Denda RM30 03041...
4/14/2017 3:13 PM
142.0 KB

DEPAN.jpg
4/27/2016 0:12 PM
211.0 KB

DEPAN(1).jpg
5/9/2016 2:06 PM
211.0 KB

Grab-KTMB-Social...
2/7/2017 10:17 AM
803.0 KB

IMG2016080517435...
12/2/2016 10:40 AM
182.0 KB

Info Komuter Link 1..

Info Komuter Link 1..

Info Komuter Link 1..

Info Komuter Link 2..

INSIDE.jpg

JIMAT PROMO SER..

4.0 CONCLUSION

The impact is serious. It is highly recommend for you to fix this flaws immediately as it threatens your system. I’ll not responsible for the damage caused by this documentation.