Course : Information Systems Management

Course Code :ITB7321

Name :Yufei Liu

Id :16441834

Assignment :Case Study 2

Tutor :Nick Wilkinson

Due Date : Monday 26th September 9.00 am

Word Count :1810

(Excluding Cover, Footnotes and References)

Introduction

Before the advent of Internet, criminals usually had to dig out people’s garbage or intercept their communicated mails to sniff their private information. Almost all information now that can be digitalized and available online, criminals can take advantage of the Internet to steal people’s identities, hack into their accounts, phish them into revealing the information, or infect their computers with malware. Furthermore, via Internet, such convenient tool allows attackers easily to obtain the malicious software to launch attacks, a well-known tool set on Linux based Operation System named “Kali Linux”. In this OS, there are integrated all different usages out-of-box hacking tools from exploitation to vulnerability analysis, wireless attacks to Web application attacks. This makes launching a hack attack easier than before. In this report, two kinds of most used attack method will be introduced, as well as their countermeasures people can take when they happen.

DDoS

DDoS is a short term of Distributed Denial of Service, which is a kind of attack attempting to take an online service or resource down by overwhelming it with network traffic coming from loads of other sources. They can target a wide range of significant resources, from the press to bank websites, and compose a major challenge to the right for people who can publish or query important information. There are several reports showing that attackers could easily purchase a week-long DDoS attack from black market at the cost of $150 and according the statistic, meanwhile, 1/3 of being attacked downtime are caused by DDoS (Goncharov, 2012). How does DDoS work? First, attackers need to build a network consisting of a lot of infected computers, typically by Trojan horse, and these computers compose a network called ‘botnets’. Two words, robot and network, build up the term botnet which usually relevant with a negative or malicious connotation. A botnet is a network that can be coordinated their actions by a remote master computer that sends command and control to the rest of infected computers. Once infected, these computers can be taken over remotely without their owners’ awareness, and being used as an army to launch DDoS attack against specific target. During the attack, botnets can generate tremendous floods of network traffic to overwhelm and cripple a target to make its service or resources unavailable. Floods can be generated in many ways, such as sending more requests than a server can handle or letting computers send the garbage data to use up the target’s bandwidth so that the true service users cannot easily or completely unable to access to the resources. An attack can go extreme to be too big to max out a country’s international cable capacity. To shed some light on this, there are four types of common kinds of DDoS attacks:

* TCP Connection attacks, attackers can use up available connection to infrastructure devices such as load-balancers, firewalls and application servers to make the infrastructure no spare resource to handle normal customers’ requests.
* Volumetric Attacks, this type of attack is attempting to consume the bandwidth and cause the network congestion.
* Fragmentation Attacks, these attempt to send a flood of TCP or UDP fragments to a target, cripple the victim’s ability to re-assemble the streams and reducing performance, this often happen to those online media websites which heavily rely on data streaming.
* Application Attacks, these attempt to overwhelm a certain aspect of application or service with a very effective and few attacking machines, this like a surgical attack.

To enlarge the effect of the attacks, there are two kinds of amplification methods attackers may use:

* DNS reflection, by counterfeiting a victim’s IP to mock DNS server, an attack can send small requests to DNS server and ask it to send the victim a large reply.
* Chargen Reflection, most computers and connected printers support an old-fashioned testing service called Chargen, which can be used as a way for amplifying attacks similar to DNS reflection. (ideas, 2013)

One of the most sensational attack in these years is between GitHub, the largest public code repository in the world and Baidu, Chinese largest search engine. According to the report from GitHub, this attack is the largest DDoS attack in the site’s history. Researchers analysed the root cause of this attack is caused by a malicious JavaScript injected into the traffic of China-based search engine Baidu, which redirected the traffic to two targeted GitHub pages, similarly to the two amplification methods described above. The attack lead to the consequence that these two pages ,even the whole website is inaccessible from China mainland for almost 1 week (Goodin, 2015).

To minimized the effects caused by DDoS, there are some useful suggestions for fighting DDoS attacks. Basically, this attack is to exhaust the website with bandwidth to slowdown the accessibility. A simple solution is to overprovision bandwidth, which means to have more bandwidth available to the web server than you ever think you are likely to need. By doing so, you can accommodate sudden and unexpected surges in traffic. The next step is to call your ISP or hosting provider, tell them you are under attack and call for help. Keep emergency contacts for your ISP or hosting provider readily available, so you can response to this quickly. Besides the inner part, another important response to a DDoS attack is how you will communicate the problem to customers. DDoS attacks could last as long as 24 hours, and good communication can ensure that the cost to your business is minimized while you survival under attack (Rubens, 2013).

Injection flaws

Injections can be identified as many types: SQL, Hibernate Query Language, LDAP, XPath and many more. Injection often occurs the command or query consists of user input data. Attackers can trick the interpreter into carrying out an unintended command by inputting certain crafted data. Sometimes, injection flaws enable attackers to access or modify any arbitrary data. In the worst case, these flaws allow an attacker to completely not only compromise the application, but also the underlying systems, even penetrate secured environments. Injection flows, especially SQL injection, unfortunately are very common in web applications. Take SQL Injection as example, this kind of attack allows attackers to spoof identity, tamper with existing data such as invaliding transactions, modifying balances, destroying the data or becoming administration of the database server. According to the technic statistics, PHP and ASP applications tend to be more vulnerable to SQL Injection due to the popularity of old interfaces. Relying on the feature of language interfaces, JavaEE and ASP.NET applications are less likely to have such issue. Generally, SQL Injection error happens when data comes into a system from an untrusted source or the data dynamically constructs a SQL query. A victim system could be compromised in following ways:

* Confidentiality: Losing confidentiality is a common problem with SQL Injection vulnerabilities as SQL databases hold sensitive data.
* Authentication: Attackers may possibly access to a system, which has a poor SQL commands to check username and password, without having previous knowledge of others password.
* Authorization: it may change the information of authorization through a successful SQL Injection attack if such information existing in a SQL database.
* Integrity: Changing or deletion sensitive information will cause data integrity violation with a SQL Injection attack.

Here are two examples to demonstrate how SQL Injection works:

Assume we have a system using SQL to query user stuff with specific item name, the SQL can be written like this

SELECT \* FROM items

WHERE owner =

AND itemname = ;

An attacker named “wiley” attempting to inject a crafted string, name' OR 'a'='a', in SQL to bypass the check in items table, then it construct a query in underlying system like this

SELECT \* FROM items

WHERE owner = 'wiley'

AND itemname = 'name' OR 'a'='a';

With an additional condition OR ‘a’=’a’, the query becomes

SELECT \* FROM items;

This simple injection allows attackers to return all items without the checks that if items belong to the real authenticated user.

Here is another more vicious example, in this example, attacker crafts a sub string as "name'); DELETE FROM items; SELECT \* FROM items WHERE 'a'='a"

then the query becomes the following three queries, but the middle one will delete all items table entries.

SELECT \* FROM items

WHERE owner = 'hacker'

AND itemname = 'name';

DELETE FROM items;

SELECT \* FROM items WHERE 'a'='a';

How to prevent SQL Injection attacks? First of all, websites must filter all users’ input. Specifically, user data should be filtered for context, for example, email address, phone numbers should be filtered to allow only the characters allowed in a phone number. Second, to setup a web application firewall which comes with a sophisticated and ever-evolving set of rules to filter potentially dangerous web requests. Third, limit database privileges by context which means creating multiple database user accounts with minimum levels of privilege for their usage environment. By adopting these three tips, it will provide a high degree of SQL Injection prevention. (Cohen, 2010)

Crimes Act 1961

To deal with an increasing number of cyber-crime, New Zealand government legislated a series of law to confine the boundary of cyber-crime in Crimes Act 1961 section 249 to 252 in detail. At the interpretation section, 248, it defined what access, authorization and computer system are. 249 to 252 list out 4 types of cyber-crime and give the punishments accordingly. For example, anyone who is liable to imprisonment for a term from 2 to 10 years for the cyber-crime. The two attack methods described above apparently violate section 250 in Crimes Act 196, “Damaging or interfering with computer system”. In a situation where cyber-crime is likely danger to life as a result, a criminal could be sentenced to up most 10 years. In general, anyone who is liable to imprisonment for a term not exceeding 7 years for intentionally or recklessly, and without authorization, knowing that he or she is not authorized, or being reckless as to whether or not he or she is authorized, damages, deletes, modifies, or interferes with or impairs any data or software in any computer system or causes any computer system to fail or deny service to any authorized users. Obviously, DDoS, comes with the purpose that interfere and impair a websites’ computer system and SQL Injection can cause data’s damages, deletes and modifies (1961, 2003). Strict law is essential, however, according to the character of cyber criminals, and this becomes more difficult as cyber-crime continues to grow catching the often requires the collaboration of many people. No matter who is victim or cyber expert, company or government. Everyone has the obligation to countermeasure to cyber-crime. Such as software developers are also needed to work with law enforcement by helping analyse cyber-crime and prevent attack. Companies and government officials can also attend security conferences to try and recruit cyber defender, meanwhile, security experts teach people who are in charge of system how to protect their systems from being hacked.

Conclusion

To sum up, from individual perspective, taking measures to secure your own computer and protect your personal information, not only are you preventing cyber-attack from phishing your identity, but protecting others by preventing your computer from becoming part of botnet as well; from government law enforcement agencies, besides the legitimation, other measures also can be launched to counter attack cyber-crime, like encouraging people to report cyber-crimes and recruiting cyber defenders.

# Bibliography

1961, C. A. (2003). *Damaging or interfering with computer system*. Retrieved from Crimes Act 1961: http://www.legislation.govt.nz/act/public/1961/0043/latest/DLM330425.html

Cohen, G. (2010). *SQL INJECTION ANYWHERE.* HACKITCS.

Goncharov, M. (2012). *Russian Underground 101.* Russia: Trend Micro Incorporated Research Paper.

Goodin, D. (2015, 4 1). *Massive denial-of-service attack on GitHub tied to Chinese government*. Retrieved from arstechnica: http://arstechnica.com/security/2015/03/massive-denial-of-service-attack-on-github-tied-to-chinese-government/

ideas, G. (2013). *What is a DDoS Attack*. Retrieved from http://www.digitalattackmap.com/understanding-ddos/

Rubens, P. (2013, 4 30). *Tips for Fighting DDoS Attacks*. Retrieved from eSecurity Planet: http://www.esecurityplanet.com/network-security/5-tips-for-fighting-ddos-attacks.html