International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering
Vol. 2, No. 2, 2015, pp. 234-248.
ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

A Survey of the Solutions to Detect and Deal with File Injection Attacks in Web Sites through Access to Web Server Shared Resources

Yaghoob Noroozi Ayeblo[a] , Ahmad Faraahi[b]

[a] *M.Sc. Student of Computer- software, Department of Computer, College of Engineering, Bandar Abbas Branch, Islamic Azad University, Bandar Abbas, Iran.*

[b] *Payam Noor University, Faculty Member of Computer and Information Technology, Tehran, Iran*.

Abstract

Web servers are the basis to respond to users' requirements using Hypertext Transfer Protocols (HTTPs), but can cause irreversible problems in case of insecurity. Therefore, security needs should be met in web servers so as to achieve key elements in security, namely authentication, permission, confidentiality, integrity, auditing, and availability. Using vulnerabilities, hackers threaten and attack websites and web servers. A threat can potentially lead to a catastrophe that can damage an organization's resources and use the system's vulnerabilities after a successful attack. Nowadays, different types of attacks are launched on web servers. The present study aimed to investigate file injection attacks using sharing resources on web servers. These attacks damage web servers using weak points and insufficient validation in web applications and web servers' infrastructures. On the other hand, weak points can help identify vulnerabilities and improve web servers' security. The results of this research in a real environment, Shows that vulnerabilities caused by this attack could have a heavier consequences than other attacks and ways to prevent file injection attacks the provided.

Keywords: File injection, Web server, Sharing resources, Vulnerability, Security.

Introduction:

Security is one of the critical issues in web which is designed to protect the organization`s assets. These assets may include a webpage or data bank for customers. It should be considered that security is a path not destination, it means that, in parallel with analysis of infrastructures and programs, risks and treats they arise should also be recognized at the same time. A process is consecutive and it can never be definitely guaranteed. Overall, the computer programs should be installed and used in a way to be safe from any kind of risk or threat. This issue is especially crucial the web programs and the reason is that the programs which are utilized more by public, is more exposed to risk. Applying various methods, infiltrators try to damage web servers and web site under divergent excuses. Each attack is performed with a certain algorithm; however, security experts have introduced a huge number of recognition and protection methods to world in order to minimize these attacks [9].
In this survey, initially, one of the attacks, "injection file", is introduced and by mutual sources in web server, and later the necessary strategies in order to protect and defend the attacks shall be presented. In order to provide strategy to defeat file injection attacks, first the attacks should be recognized, in order to fulfill this step, it is preferred to perform the attack, however, finding some attacks such as file injection by software method is a bit complex. While, performing this attack, its weak points and errors will be easily revealed. Vulnerable points can be identified in each step of performing the attack and as a result, necessary strategies in order to resolve it.

2- Literature review

Attentions to network security have risen since 1970, while the internet was applied by experts in commercial and business department. In 1993, the internet popularity was stabilized for the public and especially the businessmen who were optimist about reduction in exchange charges. It was considered as a tool which can be applied in the field of services and goods. This essay is directed in a way to recognize insecurity and vulnerability in a network, especially electronic websites and shall propose the protection approaches as a necessity of the research [8].

In article of "*Applying Vulnerable Data of Computer Systems In Order To Improve Network Security*", the solution to optimize network security to ban some of the ports which were under attack was investigated. This research introduced a method for improving the network security which included networks management, vulnerability scan, risk assessment, manual control and getting aware of the event [10].

In "*Inclusion Local and Remote File*" by Hubczyk, Domanski & Domanska in 2012, vulnerability of remote and local files, was under investigation in details. It has been proven that these security defects may lead to a variety of problems, including overloading the server or revealing customers' inaccessible files.

In a research in 2012, "*Preventing Automatic Code Injection for Web Software*", it was stated that a method based on integration of some layers in order to prevent code injection in web application programs is proposed. It is not essential to add or modify web explorer. The validity of this method is proven by this that the customers` complier can be proven in abstract [11].

According to a research in 2014, "*PHP Security*", websites attacks include SQL, file, code, orders, XML, HTML5 injections was fully discussed and solutions has been proposal [3].

3- Hypothesis

1- Applying suggested strategies in order to recognize and defend the injection file attacks can be efficient and optimized. They can recognize, control and prevent file injection attacks on websites.
2- There is a significant correlation between file injection attacks and accessing to sharing resources of web server.

4- Purposes

The main purpose of this essay is to increase security level of websites in Iran by assessing problems and vulnerable points including file or code injection, recognizing the security gaps, providing strategies and measures in order to minimize and improve the vulnerability. In the end, the necessary suggestions to recognize the defects and preventing the website vulnerability shall be provided.

5- Method

The essay is practical and consists of two parts. The first part is referring to library and studying scientific essays and research, technical reports, books, thesis, research projects and available technical and scientific contexts in the field of computer attacks to the Web server. University professors' knowledge, information technology experts in organizations and cyber police experts mastered in data exchange and network security information can assist a great deal in collecting data. In the second part, the real implementation attacks as the proposed method to recognize and confronting to this type of vulnerabilities discussed. Statistical Population is global websites and the sample space includes number of website. The output can also be displayed as percentage.

6- A brief history of injection attacks in web servers

SQL injection attack was first identified in 1998; however, a huge number of news regarding this attack arises daily. According to a record by MITRE website in 2006, SQL attack was 13.6%, XSS attack was 18.5% and so called PHP-Include attack was 13.1% which are all a sub branch of injection attacks. Accruing to a record of MITRE in 2011, SQL injection attack, command injection in operating system, buffer overloading and XSS were ranked 1 to 4 respectively.

In addition, according to website of "*Emergency management centre and coordination in computer events of communication and information technology organization*"[*], china and USA are still the main source of SQL injection attacks. In the first three months of 2011, china ranked the first by hosting 50% of SQL injection attacks and USA ranked the second by 25%. Ukraine owed the third position, while Iran was fourth, 13% and 5% respectively. Rest of the countries supported less than 2% of the attacks.

In the 2014 year, *Imperva* website revealed an injection attack on management website of Word Press and stated that remote file inserting and SQL injection posses the highest rate of attacks. As depicted in figure 1, PHP and ASP were among the vulnerable programs.

[*] www.certcc.ir

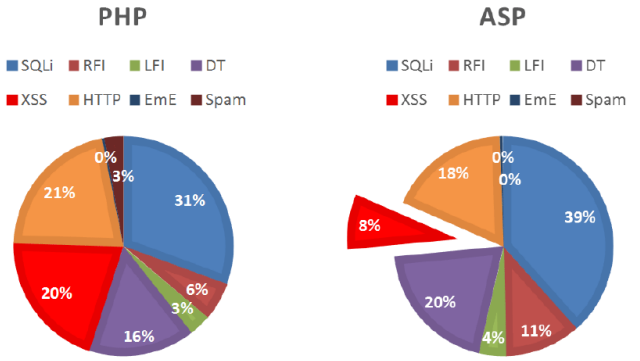


Figure (1) Comparison of injection attacks on ASP & PHP [7]

Based on the *Imperva* records in 2013, as shown in figure 2 and according to figure 3 in 2014, these programs had influence on web practical programs and penetration difference has been assessed.

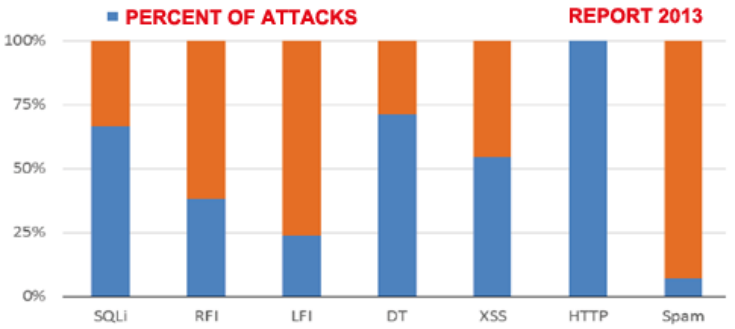


Figure (2) Bar chart comparison of injection attacks on web servers [7]

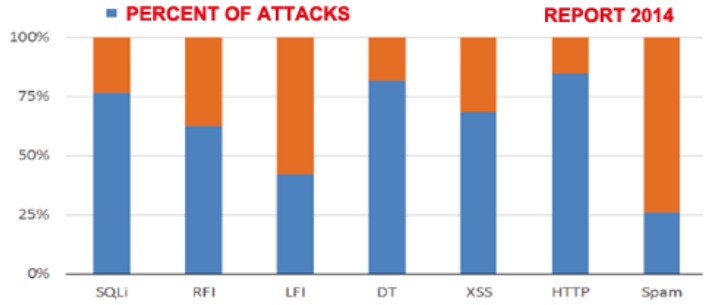


Figure (3) Comparison bar chart of injection attacks on web servers [7]

In February 2015, the comparative diagram by *Hackmageddon*, depicts attack motivation as crime and spy, respectively. Later, all types of attacks to websites and web servers were investigated and it was revealed that injection attacks were at the highest point. SQL injection attack and remote file injection, respectively allocated the highest level of penetration.

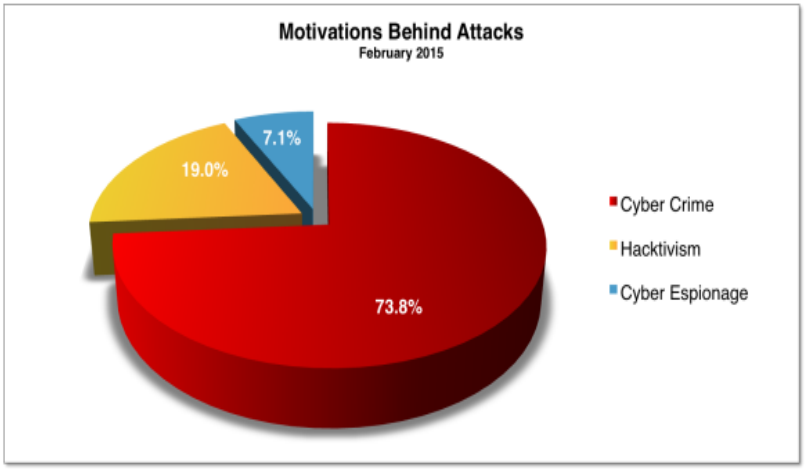


Figure (4) Comparison Chart motivated attacks [6]

7- Types of injection attacks:

1-7- Files injection attacks

It is among invalid access and penetration of a web site by a hacker. All following attacks are sub branches of injection file which start their attack by vulnerability and then penetrating the web server. It is a type of attack with similar application that occurs by injection or entering file to web server and is among invalid access. Whenever, there is a form on a website for entering information, it is potential for injection attack. Attackers download these files directly or indirectly on victim computer by running different websites and web based attacks [13].

2-7- code injection attacks

Code injection attacks are referred to a type of attack in which a code is inserted in destination point and it shall be translated and performed by the program. This attack, generally, occurs when there is no correct validation on input and output data. In this injection, valid characters (standard phrases and conditions), format and shape of data and the number of sufficient input data is important. Technical damage in code injection attack can lead to losing or damaging data, problem in response or inaccessibility. Sometimes, controlling the server will be possible, totally. One of the most prevalent code injection attacks is SQL injection which is targeted at data base interpreter [3].

3-7- command injection attacks

Command injection is an attack which is aimed at performing desirable commands on a host operating system on which a vulnerable program is installed. Command injection is possible to occur, while a program sends the data that a user enters, to the command line or system shell without validation. These data includes forms, cookies, HTTP headers, etc. Systematic commands sent by the hacker, will be performed by the accessibility level of the vulnerable program. The number of command injection attacks is so high, due to improper data validation. Code injection is distinct from command injection, in the former the hacker is limited to programming language and language operation which is inserted in. if a hacker inserts a PHP code in the program and the code be performed, it will be still limited to tasks which PHP can do. However, command injection utilizes available code source in order to perform command on system. In injection code attack, the hacker adds or substitutes his own code for the main code and his code is performed by the program. In injection code, the hacker improves program`s default operation and it does not necessarily need system commands [3].

4-7- LDAP[†] injection attacks

It is an attack which is used to exploit web programs that make commands and phrases based on user's information. While the program cannot validate user`s input properly, the hacker may change LDAP phrases by local proxy. The result of this attack leads to more accessibility to perform an inquiry without validation and a change in LDAP three contents and performing other commands of this type. Other improved methods presented in SQL, is also simulation in LDAP injection [11].

5-7- RFI[‡] attacks

Inserting a remote file is a type of vulnerability which is usually found in websites. This allows an attacker to access to a file remotely via a script in server, so this attack is injection type and its vulnerability is due to enter of invalid which has the following consequences [5]:

- ✓ Performing code on web server.
- ✓ Performing code in the position of receiver services including java script which may lead to (XSS)[§] attacks.
- ✓ Denial of Service.

6-7- LFI[**] attacks

It provides a local accessibility for hacker to view website`s target files. The defects of LFI is mainly in programmers errors in utilizing functions such as include, require, etc. and may lead to reading critical files such as configuration settings. Vulnerability of remote local injection files is categorized as practical web vulnerabilities, because they are independent of scripts which run in server [5].

7-7- SQL injection attacks

SQL injection attack is the simplest and most prevalent method of vulnerability which is installed via practical web-based data by the attackers. Utilizing simple SQL commands such as choosing, placing, deleting and updating crucial damaging attacks and performing vulnerable codes in web`s practical programs, SQL real code was created. While the attacker gain his aim, he shall succeed to access critical data, change secure data and perform data and may even ruin all the practical programs. SQL injection attacks are so beneficial for attackers, because they can access business issues, bank accounts, credit card numbers, etc. The security points in practical web programs, usually requires users identity confirmation. There are different types of SQL injection attack which are initiated by applying so many real parameters mistaken by the system [2].

8- Web server

Web server is able to respond to web explorer and sends its requested page. Web pages are placed on web server based on a certain structure and with a unique name. Several pages with separate structures may be placed on a web server [4].

1-8- Necessity of security in web server

Internet security is a sub branch of computer security which is involved particularly in internet and internet explorer security. There is also a more general level of network security. Web server's security is

---

[†] Light Weight Directory Access Protocol
[‡] Remote File Inclusion
[§] Cross Site Scripting
[**] local File Inclusion

its sub branch and is one of the most important cases, due to client's connections. Its operation is based on finding anti internet attacks algorithms and approaches. Web servers include basics of network issues, and it is because all clients` requested input and output data are originated from web. Initially the network security and later on the web server security should be taken into account by network manager. Internet is an unsecure channel for exchanging data, due to the huge number of users who listens to channels with no valid. Injection attack is a good sample for this case [15].

9- Vulnerability

In recent years, many violating attacks to servers have occurred. These vulnerabilities mainly included operating systems, practical software and server computer hardware, improper accessibility settings to systems or users abuse. As a result of vulnerability, access to system shall be possible, it can be manipulated, turned off or even get it out of security strategies. Because vulnerability has distinct reasons, its definition and categorization is very complex.

System vulnerability is as a result of deficiency in software, hardware design or even wrong settings. Vulnerability leads to unpredicted outcomes including decreases system operation or damaging system data. In some cases, critical data may be copied or robbed, which are all the results of systems vulnerability [13].

10- Shared resources and its security

1-10- shared resources

Any resource which can be put on a network as shared and is accessible for users through rules, group policies, valid shared accesses is defined as shared resource, including folders, drives, printers etc. in this way, operation system tasks are as follows:

- Files and folders management.
- Optimizing resources and preventing their wasting.
- Releasing resources.
- Managing queues and timing to use resources.
- Creating security.

2-10- shared resources security

Having entered the network, security system allows user to access proper and valid resources, set by network manager. Users have passwords, however resources have access permit. In other words, each resource is secured by a firewall. This wall has several layers or doors and users can go through them and access resources. Some doors provide more accessibility for users. The manager is responsible to determine which users are allowed to pass which doors. For instance, a door may allow the user to access fully, on the other hand, may solely let the user to access reading. Each resource or shared file is saved with a list of users, groups or related permits [1].

11- Check file injection method through access to shared resource

Note: the case study is a real vulnerable website by the address of http://www.xxxxxxx.net with IP of 62.xxx.xxx.203 which codes output is based on this website.

1-11- public shared resources of operating system:

Public shared resources on a window the based transaction and information processing with shared hardware resources, the maximum has all paths the following default that they will see in the following table1:

Table 1- Windows public shared resources and share type them

share the type	public shared resources of windows	Description
Hidden	Drive Letter$	Shared root partitions.
Hidden	ADMIN$	Shared resources for the use and management of the remote computer are use.
Hidden	IPC$	Resource that pipeline for communication between applications on a shared network and cannot be removed.
obvious	NETLOGON	In a network, resource that on the domain controller used.
obvious	SYSVOL	In a network, resource that on the domain controller used.
Hidden	PRINT$	Resources that to use remote printers on a network used.
Hidden	FAX$	Shared resources on a server that users send and receive faxes through a fax.

12- Suggested method

In this part, our suggested method shall be explained in several steps. Initially, internet address should be transformed to IP, open ports related to host shared resources should be defined and finally raveling shared resources paths of web server should be reviewed via open ports. Accessing to shared resources is so crucial, because there is the possibility of defining access levels with consideration of general and specific security IDs. Accepting or declining access levels is based on access lists control. If the web server is vulnerable, inserting file or data will be possible, so approaches to defend these attacks should be checked and provided.

1-12- Obtain the Internet address:

The first step in a server`s vulnerability to access of shared resources is to obtain internet address for *IPv4-base* protocol by referring to public or root *DNS*.

2-12- Check open ports associated with shared resources:

The second step is assessing open ports related to share resources of host website via *WAN* and evaluating reaction of host web server to open port access via internet.
Certain *TCP* ports for assessing vulnerability via *WAN* are 135, 139, and 445. By analyzing port *TCP* by applying *Nmap* or by sending telnet packet on these ports via *CMD*, it can be checked if the server is open or not and also responds or not to receiving *TCP* packets.

```
>Nmap -p 0-65535 62.xxx.xxx.203
Starting Nmap 6.40 at 2015-02-02 15:33 Iran Standard Time
Nmap scan report for 62.xxx.xxx.203
Host is up (0.38s latency).
PORT STATE SERVICE
25/tcp open smtp
80/tcp open http
135/tcp open msrpc
139/tcp open netbios-ssn
445/tcp open microsoft-ds
3389/tcp openms-wbt-server
49154/tcp open unknown
```

Figure (5) Check open ports associated with shared resources with Nmap command

3-12- Investigate of revelation shared resources paths:

The third step is to assess and reveal shared resources paths by related open ports and net command in *CMD*. Revealed paths includes are public and private shared addresses is the under the protocol Netbios-SSN on *TCP/ IP* connection. In the following code using the Net view command sample web server shared resources we will see:

```
>net view \\62.xxx.xxx.203
Shared resources at \\62.xxx.xxx.203
Share Name Type Used as Comment
ADMIN$ Disk Remote Admin
C$ Disk Default share
D$ Disk Default share
E$ Disk Default share
IPC$ IPC Remote IPC
Wwwroot Disk (UNC)
The command completed successfully.
```

Figure (6) Investigate and revelation shared resources paths on the web server

Shared resources of sample web server includes five default shared resources (ADMIN$, C$, D$, IPC$, E$) and one special resource with a name wwwroot.

4-12- Feasibility and determine of access levels to the various routes to Web server shared resources

In the fourth step, we will review possibility of accessibility level in different paths of shared resources in a web server via ports revealed by general and specific SIDs. It should be noted that while a *SID* is created for a user or group, it will never be possible to use this security data for any other user or group. Allocating or denying resource accessibility is done by operating systems, based on *ACL* (accessibility control list).

5-12- Security models based on ACL

While an operator asks for operation in *ACL* base security model, operating system initially checks ACL in order to confirm the validity of requested operation. Key point in defining any *ACL* based security model is determining the way of accessibility to editing control list; it means which users and processes access to *ACL*. *ACL* models may refer to a series of object and also to individual organizations.

In general, control and management of resource accessibility is based on shared permits and *NTFS* occurs by *LSASS*. In addition, shared permit is the priority in permits and then followed by *NTFS*.
In fact, when a user (generally: security core with SID) requests to access to the computer resources via operating system, access token of that user (including SID user) shall be sent to LSASS in operating system. After being compared with defined accessibility levels in related ACL, the certain permit shall be issues. Continuing target server security analysis, now we will review possibility and determination of access levels in different shared resources of web server via raveled ports. For this reason, in past cacls tool was applied in order to change file access, nevertheless, after Vista windows, icacls command is also added to enable us to change file and folder permits, view the accesses and edit them in case of necessity. In CMD, each shared resources of web server to investigate the following:

```
>icacls \\62.xxx.xxx.203\ADMIN$
\\62.xxx.xxx.203\ADMIN$: Access is denied.
Successfully processed 0 files; Failed processing 1 files
>icacls \\62.xxx.xxx.203\C$
\\62.xxx.xxx.203\C$: Access is denied.
Successfully processed 0 files; Failed processing 1 files
>icacls \\62.xxx.xxx.203\D$
\\62.xxx.xxx.203\D$: Access is denied.
Successfully processed 0 files; Failed processing 1 files
>icacls \\62.xxx.xxx.203\E$
\\62.xxx.xxx.203\E$: Access is denied.
Successfully processed 0 files; Failed processing 1 files
>icacls \\62.xxx.xxx.203\IPC$
\\62.xxx.xxx.203\IPC$: The request is not supported.
Successfully processed 0 files; Failed processing 1 files
>icacls \\62.xxx.xxx.203\wwwroot
\\62.xxx.xxx.203\wwwroot NT AUTHORITY\SYSTEM:(OI)(CI)(F)
 BUILTIN\Administrators:(OI)(CI)(F)
 S-1-5-21-734115787-4212597339-557516564-1000:(OI)(CI)(F)
 Everyone:(OI)(CI)(F)
Successfully processed 1 files; Failed processing 0 files
```

Figure (7) Investigate the feasibility and determine of access levels to the various routes to Web server shared resources

As it was recognized, access to shared resources of web server is only possible for specific path of \\62.xxx.xxx.203\\wwwroot. Its accessibility defines the limit for general and exclusive security cores (finally general & exclusive SID) especially general security core of every one (accessing to this share resource by LSASS is provided regardless of user`s security identity).

6-12- Data injection vulnerability to the web server directory:
Fifth step, In this step the last address (\\62.xxx.xxx.203\\wwwroot) content is under survey in target server and a sample data shall be injected to web server. Then its frequency is going to be checked through a web protocol (for example http) by a browser. To view the path contents dir /a command is applied in order to show the contents of any attribute (operating systems do not show hidden or systemic contents by default, therefore, any contents of any attribute is shown in this path. A file`s attribute and change is defined by attrib command and to receive more information, *attrib /*? May be deployed).

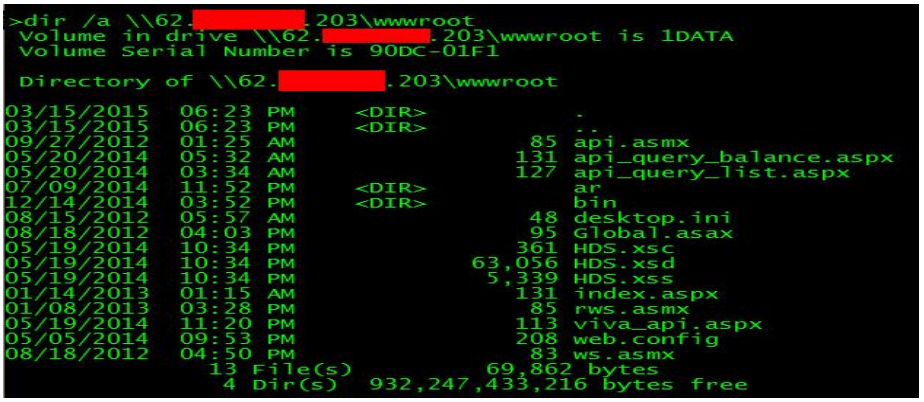


Figure (8) View the contents of the directory target using the dir command

In continues For example, select of ar directory and using the icacls command examine the access level.

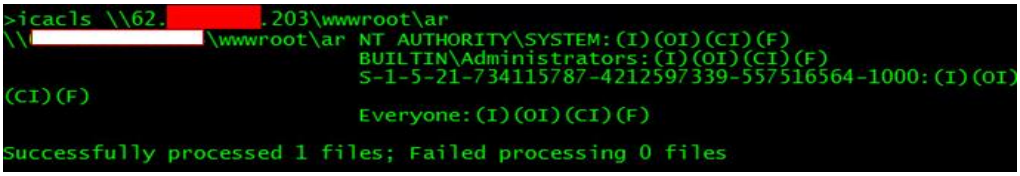


Figure (9) View access and authorization control using the icacls command

Therefore, accessing *ar* directory contents has full permission, due to the defined permits that they inherit from their parent, i.e. Wwwroot, moreover, *ar* path is equal to http://www.xxxxxx.net/sms/ar. Therefore, any file with any size or suffix may be injected. The following figure depicts injected file of sample.txt via web is depicted:



Figure (10) File injection with any extension to the target system

In addition, a shell with the ability of change, delete, add, asses and view access and control permits, group and heritage status, file name and size to destination may be injected and access all web server data. This shell is generally scripted by *PHP* and has different types. It is uploaded on server and should support main protocols in the network.
The proposed method mentioned above, our summarized in Figure 11:

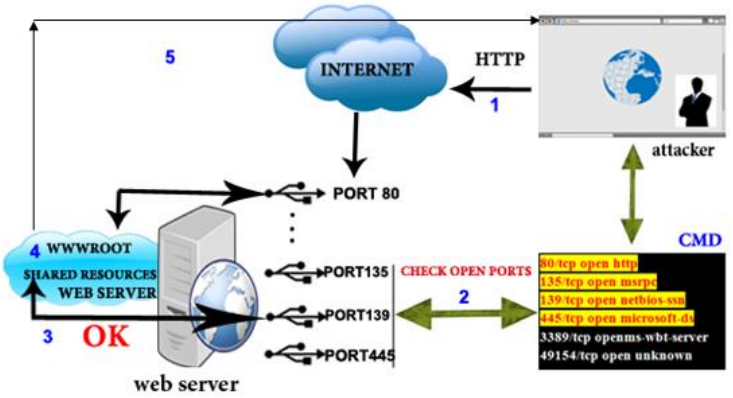


Figure (11) The implementation of architecture of proposed method the file injection

Attacker or penetrator send his request by his browser through *HTTP* protocol and can use share resources in web server. The penetrator checks opening of crucial ports via *CMD* command. In case these ports are open, accessing to share resources is possible via a port like 139; consequently, proper response will be send to the browser.

13- Evaluation criteria for assessing file injection attacks

While penetrating the system, file injection attacks may lead to error and failure. In diagram 12, there are some web servers as target and we will asses them in real environment, following the suggested method. The main criteria of this experiment with equal value of 20% are as follows:

1. Success in achieving IP the target site.
2. Obtaining open ports associated with shared resources on the host of website.
3. Revelation shared resources paths on web server via open ports.
4. Investigate the feasibility and determine of access levels to the various routes to Web server shared resources.
5. The ability of data inject to directory on the web server.

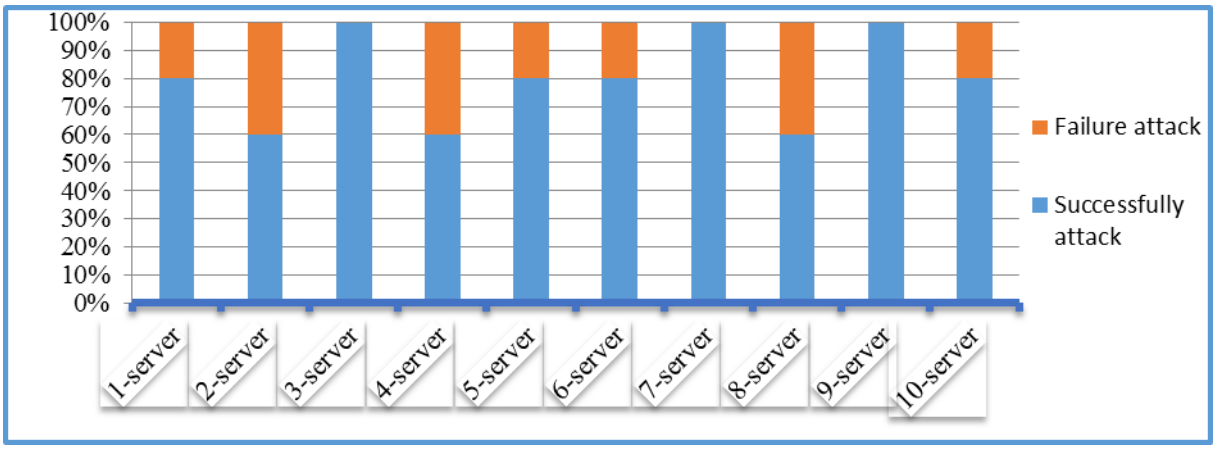


Figure (12) Diagram of success percentage file injection attack on some servers

14- Problems of data injection attacks for changing active process flow:

1) **Confidentiality:** most of injection attacks include crucial and beneficial information.
2) **Authentication:** in some cases, authentication control of injection code, may lead to remote vulnerability.
3) **Access control:** injection attacks are for control and change of a certain process, sometimes lead to desired code performance.
4) **Integrity:** data injection attack may lead to the whole data loss, in order to control data reading and writing has been done by injection.
5) **Accountability:** most of injection control code measurements are as hosts.

The main issue to consider is that all injection problems are mutual in this point that, all process performing are legal by sending code and file through share data channels [6].

15- Solutions to prevent file injection vulnerability via share resources access:

1-15- Investigate the need for a Web server resource sharing, during and after the implementation Web services:

In confronting this vulnerability, first it should be checked that whether sharing web server resources are basically conscious (by network manager) or it has been done unconsciously (by a software).
If it is conscious, in next step the purpose and requirements should be analyzed, for instance, some services exit in a network that need share resources only when they run, and then this need is fulfilled (for example, sharing installed files on a software in network). Obviously, in these cases network manager should disconnect exclusive share resource after run, or refer to security measurements in case of utilizing this source regularly.

However, if this has occurred unconsciously by software, there will be two modes:

1- Sharing a resource properly by web software (for example, creating and running a file storage server via web). In this case, need of software of to this resource shall be checked and security measurements in case of utilizing this source regularly. But if these share resources have been created just while running and is useless now, access will be stopped.
2- Sharing a resource with a bad ware (specially a spyware). There are several types of bad wares that provide spying through penetrating a computer (a server) and sharing its exclusive files. Therefore, first the bad ware should be removed and then access to share resources should be stopped.

2-15- Management and centralized shared resources of the web server:

In order to manage easier and secure more, share resources security boundaries can be reinforced by concentrating security cores and preventing resource diversity. One method to meet this issue is to run domain networks (client_server) rather than work group networks (peer to peer).

3-15- Investigate and correct the access levels for public and exclusive SID Web server:

In this step, security levels of security core of a web server to share resources paths should be reviewed and be modified if necessary. If access levels in *ACLs* are defined properly, security shall be deployed by running firewall. Obviously, permit definition for security cores including General *SID* is more crucial and risky, because they are free from systemic characters and are common in operating systems. Therefore, setting share permits and also *NTFS* permits should include security cores as less as possible. In other words, permits in *ACL* of a share resource, should be more exclusive (security cores with exclusive SID).

4-15- Implementation firewall:

Final step to immunize server to vulnerability is utilizing security wall (for instance firewall) and exact definition of server resources access. This stage can be independent but needs more complex configuration and setting. One of the preliminary points in running this stage of immunizing is the correct position of security walls in network which is between internet and web server by default. In figure 13, firewall is used properly.

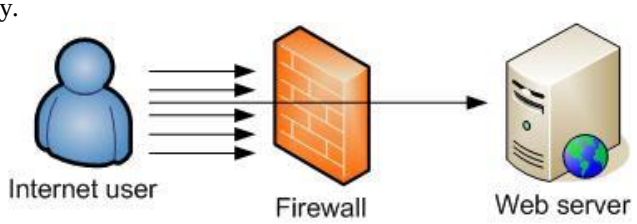


Figure (13) The proper use of a firewall between the user and the Web server [14]

This optimization in security wall configuration and setting is sufficient for web server security (and internal or local web server), due to accessing to web server through the internet.
However, access security should also be provided by local network.
In this regard, another security wall which is between web server and local network may be used. In this case, it is called inside firewall and the other security wall (between server and internet) is called outside firewall.

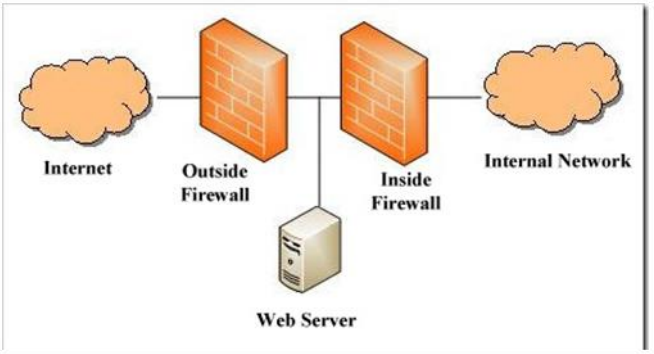


Figure (14) The correct configuration of internal and external firewall to the Web server [12]

Therefore, by deploying above mentioned security points and installing security walls with correct configuration and setting, web server (a server network) can provide service with a suitable security.

16- conclusion:

In this research, suggested file injection attack was performed in several stages in order to present strategies for preventing vulnerabilities. This method is a type of innovative attack that can penetrate target web servers with different techniques. Criteria of a good attack are observed in file injection attack, to some extent. Heritage in directories and permits in access control lists is also of high attention. After checking features of this injection file attack in this research, it is concluded that in comparison with other injection attacks, this one needs a high knowledge for penetration and damage. Other attacks may be performing easily. On the other hand, file injection attacks has a great damage in scale and recognizing these attacks is less than rest of attacks for users. Strong methods and immunizing presented web server

and finally, approaches and strategies to prevent these attacks by share resources in web server were presented in this research.

References:

Davari_dolatabadi, m. (2011). "Network Security Engineering", The second volume, the publishing tradition Kosar,(first edition), pp. 303-360.

Cherry, D. (2015). "Securing SQL Server: Protecting your Database from Attackers". (Third Edition), pp. 265-291.

Cornutt, Ch. (2014). "Securing PHP: The Usual Suspects". Lean Publishing, PP. 2-15.

Edge, Ch., Barker, W., Hunter, B., Sullivan, G. (2010). "File System Permissions: Enterprise Mac Security". pp. 79-111.

Hubczyk, M., Domanski, A., Domanska, J. (2012). "Local and Remote File Inclusion: Internet – Technical Developments and Applications 2". Berlin, Heidelberg, PP. 189-200.

Hackmageddon. (2015). "Cyber Attacks Statistics". Retrieved 13/04/2015, from http://www.hackmageddon.com/2015/04/13/march-2015-cyber-attacks-statistics.

Imperva. (2014), "Web Application Attack Report". Retrieved 10/03/2014 from http://www.imperva.com/docs/HII_Web_Application_Attack_Report_Ed4.

Jang-Jaccard, J., Nepal, S. (2013). "A survey of emerging threats in cyber security". Journal of Computer and System Sciences.

Kizza, J. (2013). "A Guide to Computer Network Security(The Computer Communications and Networks)". (Second Edition), pp. 107-132.

Lai, Y., Hsia, P. (2007). "Using the vulnerability information of computer systems to improve the network security". Computer Communications, PP. 2033-2036.

Luo, Zh., Rezk, T., Serrano, M. (2012). "Automated Code Injection Prevention for Web Applications". Berlin, Heidelberg, PP. 186-201.

MSDN. (2015). "firewall". Retrieved 18/06/2015 from Http://blogs.msdn.com.

Patil, S., Agrawal, N. (2015). "Web Security Attacks and Injection-A Survey". International Journal of Advancements in Research & Technology, Vol. 4, Issue 2, PP. 62-65.

Secforce. (2015). "Firewall Assessments". Retrieved 08/06/2015 from
https://www.secforce.com/services/firewall-assessments.html.

Williams, B., Chuvakin, A. (2015). "PCI Compliance: Vulnerability management". (Fourth Edition), PP. 161–196.