ISSN:0975-9646

Keshav Jain / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (6) , 2014, 7904-7908

Intrusion Detection through Session Hijacking

Keshav Jain

Chaudhary Devilal University,
Sirsa, Haryana, Inida.

Abstract:-The security of web applications has become increasingly important and a secure web environment has become a high priority for e-businesses communities. Online transaction of high sensitive corporate information and its security has becomes more difficult due to increase in online traffic. The latest technologies like Ajax (Asynchronous JavaScript) and emergence of Web 2 have complicated the security problem. The problem of security becomes more severe because of the open threats from hackers to corporate secrets, financial information and medical resources that exist on Web sites. Rapidly increase in online business and sharing of information are more prone to attacks and more curial to protect these applications from hackers. Application level attacks especially Session Hijacking (Web attack) and Buffer-overflow are two of the most common security vulnerabilities that plague web applications today. So to improve the results in these conditions, a proposed concept of session hijacking and prevention from attacks has introduced in this research paper. As stated above we are proposing this technique with using Wamp Server which is implemented in PHP language, it totally works on simple lines, so its processing is very fast and it also take very less memory to store databases.

INTRODUCTION:-

Intrusion-An Overview (Axelsson, 2000)

Intrusion may be defined as the potential possibility of a attempt to:-access information, manipulate information, render a system unreliable or unusable

Attacks can be classified on the basis of two criteria, depending on whether or not an attacker is normally authorized to use the computer system, and whether or not a user of the computer system is authorized to use a particular resources in the system. Hence we can broadly divide these into three intrusion classes

External penetrators - those who are not authorized use of system

Internal penetrators – those who are authorized use of the system, but are not authorized access to the data, program, or resource accessed.

Misfeasors – authorized users of the system and resources accessed, who misuse their privileges

Various attack method are used by attackers to compromise network security. We can broadly group them into a few categories-Information gathering, Unauthorized access, Disclosure Information, Denial of service

As work environment becomes more interconnected and exposed, service providers will need increasingly to rely on a wide range of anti-intrusion techniques. List here are six approaches

1. Prevention: reduces the probability of a successful attack.
2. Preemption: strikes offensively against likely threat agents prior to an intrusion attempt to lessen the likelihood of a particular intrusion occurring later.
3. Deterrence: deters the initiation or continuation of an intrusion attempt by increasing the necessary effort for an attack to succeed, increasing the risk associated with the attack, and/or devaluing the perceived gain that would come with success.
4. Deflection: leads an intruder to believe that he has succeeded in an intrusion attempts, where instead he has been attracted or shunted off to where harm is minimized. Setting up " honey pots", decoy systems that appear as vulnerable targets and lure the attacker, is one such attempt.
5. Countermeasures : actively and autonomously counter an intrusion as it is being at-tempted. This includes dropping routes which lead to the source machine, launching a counter attack, etc
6. Detection: discriminate intrusion attempts and intrusion preparation from normal activity and alerts the authorities.

Session Hijacking

History:- Session hijacking was not possible with early versions of HTTP.HTTP protocol versions 0.8 and 0.9 lacked cookies and other features necessary for session hijacking. Version 0.9beta of Mosaic Netscape, released on October 13, 1994, supported cookies.Early versions of HTTP 1.0 did have some security weaknesses relating to session hijacking, but they were difficult to exploit due to the vagaries of most early HTTP 1.0 servers and browsers. As HTTP 1.0 has been designated as a fallback for HTTP 1.1 since the early 2000s—and as HTTP 1.0 servers are all essentially HTTP 1.1 servers the session hijacking problem has evolved into a nearly permanent security risk.The introduction of super cookies and other features with the modernized HTTP 1.1 has allowed for the hijacking problem to become an ongoing security problem. Web server and browser state machine standardization has contributed to this ongoing security problem.

Method:-There are four main methods used to perpetrate a session hijack. These are:

- Session fixation, where the attacker sets a user's session id to one known to him.
- Session sidejacking, where the attacker uses packet sniffing to read network traffic between two parties to steal the session cookie.
- Alternatively, an attacker with physical access can simply attempt to steal the session key by, for example, obtaining the file or memory contents of

the appropriate part of either the user's computer or the server.

- Cross-site scripting, where the attacker tricks the user's computer into running code which is treated as trustworthy because it appears to belong to the server, allowing the attacker to obtain a copy of the cookie or perform other operations.

Prevention:-Methods to prevent session hijacking include:

- Encryption of the data traffic passed between the parties; in particular the session key, though ideally all traffic for the entire session by using SSL/TLS.
- Use of a long random number or string as the session key.
- Regenerating the session id after a successful login.
- Some services make secondary checks against the identity of the user.
- Alternatively, some services will change the value of the cookie with each and every request.
- Users may also wish to log out of websites whenever they are finished using them .(Burgers,2013 et al)

Intrusion Detection system:-Our system is based on session hijacking sometimes also known as cookie hijacking is the exploitation of a valid computer session—sometimes also called a *session key*—to gain unauthorized access to information or services in a computer system. In particular, it is used to refer to the theft of a magic cookie used to authenticate a user to a remote server. It has particular relevance to web developers, as the HTTP cookies used to maintain a session on many web sites can be easily stolen by an attacker using an intermediary computer or with access to the saved cookies on the victim's computer**.** How we can detect the hacker at local host and how detection and prevention take place at network host.

PREVIOUS WORK:-

The field of intrusion detection and network security has been around since late 1980s There are many IDSs currently available, ranging from commercial products to unprofitable ones. Some of are mentions as follows:-

Genetic algorithms were originally introduced in the field of computational biology. Since then, they have been applied in various fields with promising results. Fairly recently, researchers have tried to integrate these algorithms with IDS. The REGAL System (Neri,2000) is a concept learning system based on a distributed genetic algorithm that learns First Order Logic multi-modal concept descriptions.

Lu and Traore (Lu,2004) used historical network dataset using GP to derive a set of Classification . They used support-confidence framework as the fitness function and accurately classified several network intrusions. But their use of genetic programming made the implementation procedure very difficult and also for training procedure more data and time is required.

Lee et al. introduced data mining approaches for detecting intrusions in (Lee, 1998 et al), and (Lee,1999 et al). Data mining approaches for intrusion detection include association rules and frequent episodes, which are based on building classifiers by discovering relevant patterns of program and user behavior. Association rules and frequent episodes are used to learn the record patterns that describe user behavior. These methods can deal with symbolic data, and the features can be defined in the form of packet and connection details. However, mining of features is limited to entry level of the packet and requires the number of records to be large and sparsely populated; otherwise, they tend to produce a large number of rules that increase the complexity of the system (Kshirsagar).

Data clustering methods such as the k-means and the fuzzy c-means have also been applied extensively for intrusion detection. (Portnoy, 2001 et al) One of the main drawbacks of the clustering technique is that it is based on calculating numeric distance between the observations, and hence, the observations must be numeric. Observations with symbolic features cannot be easily determine.

Decision trees have also been used for intrusion detection (Reddyl, 2004 et al) . The decision trees select the best features for each decision node during the construction of the tree based on some well-defined criteria.

SYSTEM ARCHITECTURE

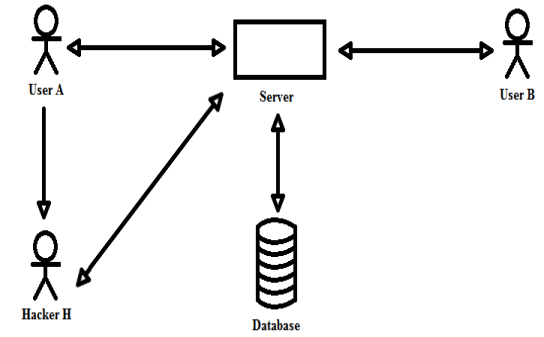


Figure: 4.4 System Architecture

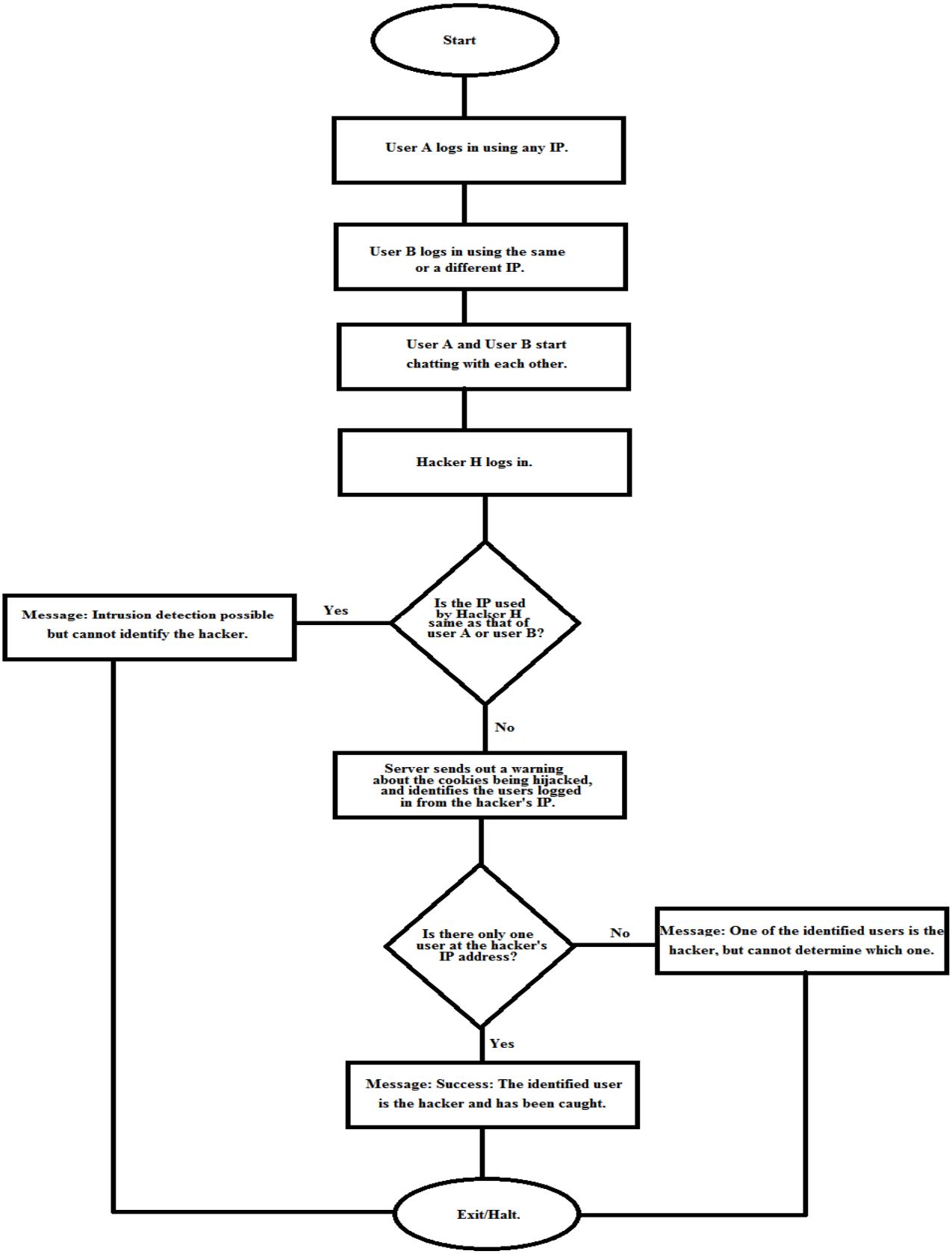
Client-Server Roles:-The *client–server* characteristic describes the relationship of cooperating programs in an application. The server component provides a function or service to one or many clients, which initiate requests for such service.

Client-Server Communication:-Clients and servers exchange messages in a request-response messaging pattern: The client sends a request, and the server returns a response. (Dustar,2005 et al)

Hacker:-Hacker someone who seeks and exploits weaknesses in a computer system or computer network.

Database:-A database is an organized collection of data.

Algorithm:



Flow Chart of Proposed System

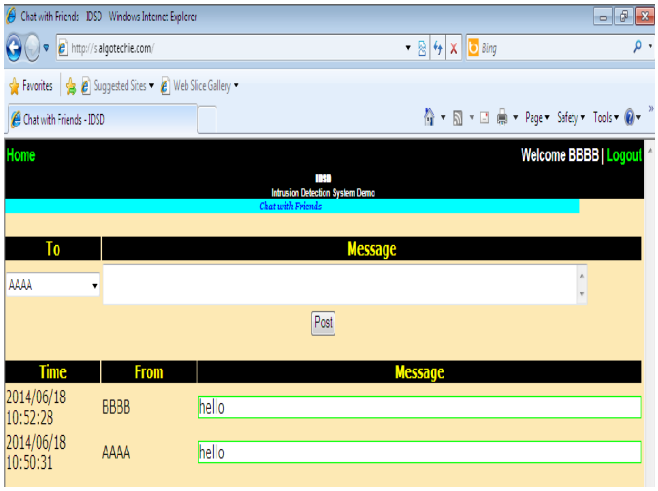
IMPLEMENTATION

A Communicate with B

First of all two users start chating with each other by Signin. At that time a session will create between two user as A and B.

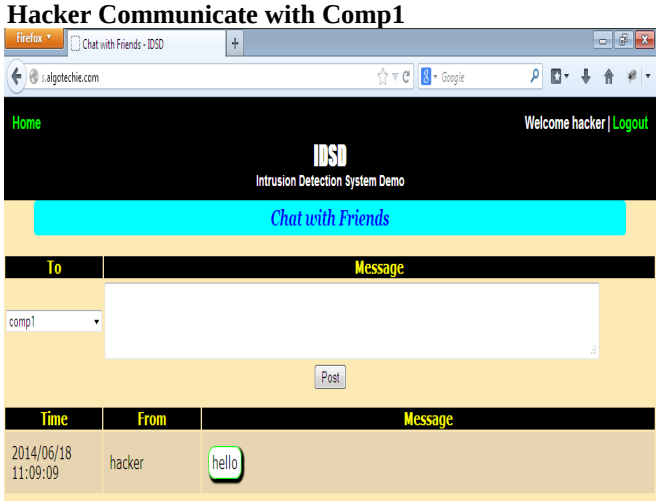
A Communication with B

B Communicate with A



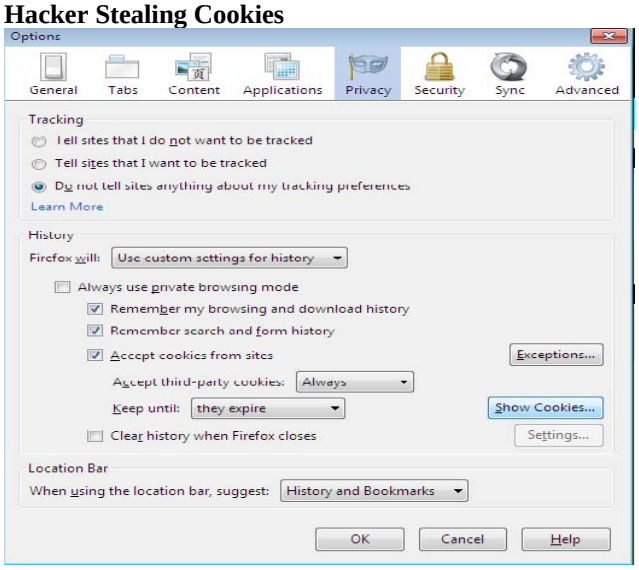
B communicate with A

Hacker Communicate with Comp1



Hacker communicate with comp1

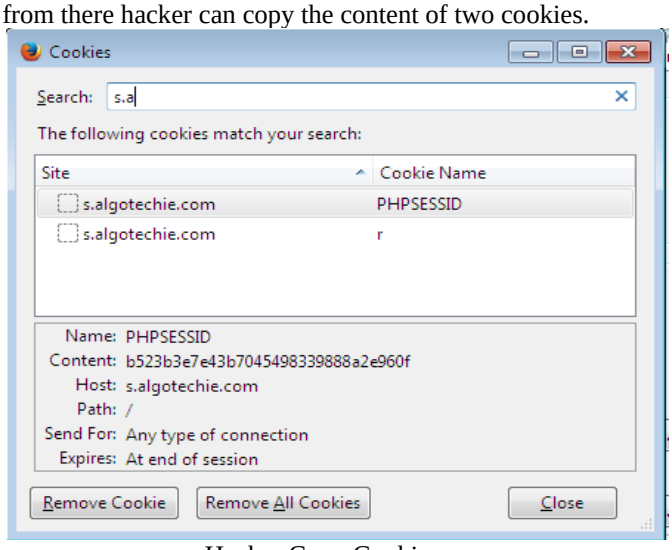
Hacker Stealing Cookies



Hacker Stealing Cookies

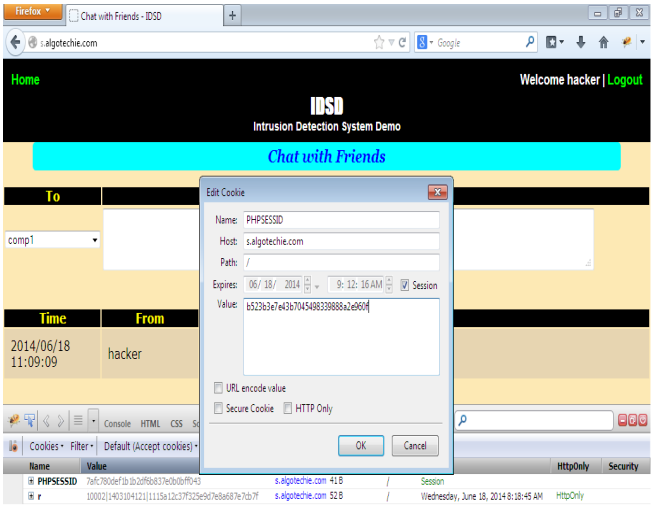
Hacker Copy Cookies

On click on show cookies a menu Cookies as shown in Figure 4.10 will be open and by entering Domain name in search option two cookies PHPSESSID and r will open from there hacker can copy the content of two cookies.



Hacker Copy Cookies

Hacker Paste Cookies



Hacker Paste Cookies

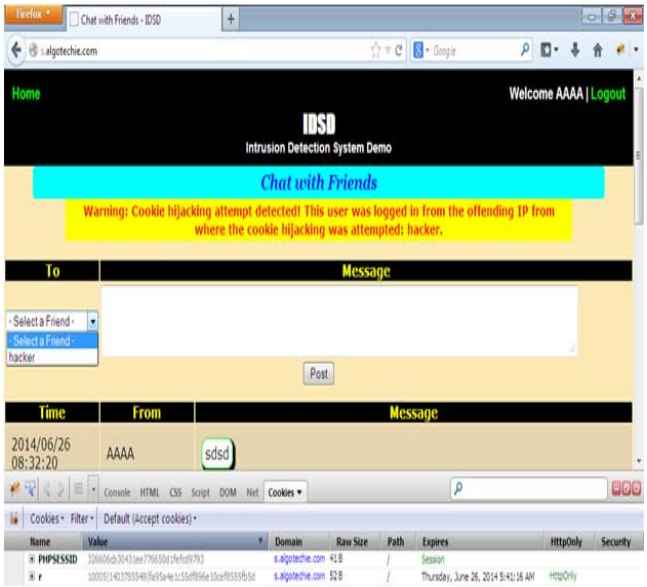
Hacker Account Convert in Real User

Hacker Account Convert in Real User

Event Generated By Hacker

Event Generated By Hacker

Prevention of all Account from Hacker



Prevention of all account from Hacker account

CONCLUSION:-

It contain the knowledge of the session based attacks, the protocol, the data and the target application that would enhance detection capability as well as result in efficient working of intrusion detection system.

Detection is possible in session based hijacking with same IP address but prevention is not possible So, to prevent intrusion, we use network based server to generate different IP address to each and every user. So, by using domain name server we can block unauthorized access.

Future work:-

Currently, the system is unable to detect the intruding computer if its IP is the same as that of the user whose cookies are being hijacked.

Currently, the proposed system is unable to sort out the exact hacker if there are multiple users on the hacker's IP. The proposed system only lists all the users logged in from the hacker's IP at the time of the attack, but does not identify one of them as the hacker.

REFERENCE

1. (Axelsson, 2000) Stefan Axelsson(2000). Intrusion detection system: A survey and taxonomy,
2. (Burgers, 2013 et al) Burgers, Willem; Roel Verdult, Marko van Eekelen (2013). Prevent Session Hijacking by Binding the Session to the Cryptographic Network Credentials *Proceedings of the 18th Nordic Conference on Secure IT Systems*
3. (Neri, 2000) Neri, F.(2000) Comparing local search with respect to genetic evolution to detect intrusion in computer networks *Congress on Evolutionary Computation (CEC00)* IEEE Press, 16-19 July, 2000.
4. (Lu, 2004) W. Lu, I. Traore(2004) Detecting New Forms of Network Intrusion Using Genetic Programming *Computational Intelligence, Blackwell Publishing, Malden.*
5. (Lee, 1999 et al) W. Lee, S. Stolfo, and K. Mok(1999) A Data Mining Framework for Building Intrusion Detection Model *IEEE Symp. Security and Privacy (SP "99)*
6. (Portnoy, 2001 et al) L. Portnoy, E. Eskin, and S. Stolfo(2001) Intrusion Detection with Unlabeled Data Using Clustering *ACM Workshop Data Mining Applied to Security (DMSA)*
7. (Reddyl, 2004 et al) Y.B. Reddyl, R. Guha(2004) Intrusion Detection using Data Mining Techniques *Artificial Intelligence and Applications*
8. (Kshirsagar) D.D.Kshirsagar Data Mining based Intrusion Detection System (DM-IDS) *International Conference*