International Journal of Engineering Research and Development
e-ISSN: 2278-067X, **p**-ISSN: 2278-800X, www.ijerd.com
Volume 6, Issue 9 (April 2013), PP. 48-52

Robust Encryption and Decryption (RED) Component and Standard Encrypt Syntax (SES) For Web Services Security Architecture

Abhilash R[1], Saravanan E[2],Kalpana G[3]
[1] M.Tech,Information Security and Cyber Forensics, Dr.M.G.R. Educational and Research Institute University
[2] Asst.Professor, Dr.M.G.R. Educational and Research Institute University
[3] Asst.Professor, Dr.M.G.R. Educational and Research Institute University

Abstract:- With changes in technological landscape the threats and vulnerabilities in network are rising at an alarming rate. There are many complexities specific to, and inherent in Web services that further complicate their security. Numerous threats can compromise the confidentiality, integrity, or availability of a Web service or the back-end systems that a Web service might expose. This paper discusses re-modeling web services security architecture from security attacks such as network security attacks, E-Commerce attacks and man in the middle attacks. A detailed conceptual framework is proposed, based on how safely to send valuable packets to the destination without any security attacks. This framework is based on providing a Robust Encryption and Decryption (RED) technique for web browser and the web server and Standard Encrypt Syntax (SES) to web application.

Keywords:- Web Security, Threats and Vulnerabilities in web, Cryptography, Web Security Architecture

I. INTRODUCTION

Today, there is an ever-growing dependency on computer networks for business transactions. With the free flow of information and the availability of vast resources, users of enterprise networks have to understand all the possible threats. These threats[3,9] take many forms; but all result in loss of privacy to some degree and possibility of malicious destruction of information or resources, that can lead to large monetary losses.
Though many countermeasures are in place taken to provide secure internet, there are still some holes in the network which lead to severe security attacks[7]. This paper discusses the holes in the network and provides a detailed conceptual framework to avoid these threats and vulnerabilities.

II. SECURITY THREATS AND VULNERABILITIES IN WEB SERVICES

When we surf the Web, or send E-Mail, the communications between our computer and the server takes place via the data packet. It is the data packet that contains the information and the request for information that is sent from our computer to other computers over the Internet. The communication protocol which is used to govern the flow of data packets is called Transmission Control Protocol/Internet protocol or TCP/IP for short.

A. Eavesdropping

Eavesdropping, a network packet capturing application that collates and displays all packets in a user understandable format that passes in communication channel. Intruder understands and gain information about the packet.
A sample skype Trojan source snippet:

```
If (fPFWCircumventionStatus [mCircumventionType] (mGetBuffer, mProxyCon
REQUEST_TYPE_SYSTEMUP) != 0)
Sleep(NOCONNECTIVITYSLEEP);
else{
remoteStatusChecks(mGetBuffer);
mConnectivity=1;
}}
```

B. TCP Hijacking

Weakness in creating TCP connections will open way for new attacks such as TCP/IP hijacking. TCP/IP hijacking technique[4] uses spoofed packets to take over a connection between a victim and a host

machine. It is successfully done by predicting intial sequence number that gets exchanged between victim and host.

C. Session hijacking

Session tokens are passed between client and server when server authenticates a client during login. Attackers hijack these session tokens and use it to acquire unauthorized access to web server.

```
<?php
If(isset($_REQUEST[“cmd”]))
{
Sf=fopen(“stolen.txt”,”w+”);
fwrite(Sf,$_REQUEST[“cmd”].”<br>”);
fclose(sf);
}
```

D. Network sniffing

It is a technique of capturing network packets that are transferred in communication links. It captures all network packets and provides clear information about packets to intruder which results in attacks like eavesdropping, man-in-the-middle attack.

Sample code for incoming traffic on all ports

```
WSAIoct1(thisSocket, &inn, sizeof(inn), &out, sizeof(out))
```

III. EXISTING SYSTEM

A. SSL/TLS Certificate:

SSL[8] is an encryption technology that provides secure communication between client and server by encrypting packet information and link. It involves

- SSL Server Authentication
- SSL Client Authentication
- An Encrypted SSL Connection

It protects data passed between communication links from network attacks such as sniffing, spoofing, data tampering. These certificates are issued by third party companies.

B. Disadvantages of SSL

1. *SSL reduces site speed:* SSL can reduce the time it takes for data to travel between the user and website by two to ten times. Moreover it increases network traffic by 300 percent.
2. *SSL costs money:* The cost of SSL is more which looks very expensive for single one who maintains site.
3. *SSL isn't secure:* As day-by day many attacking techniques evolves, SSL fails to provide security.
4. *SSL regular renewal:* SSL certificates have to be renewed regularly and it fails if renewal is not done on time.
5. *Complex installation of SSL:* Installing process of SSL is very complex.

IV. COUNTERMEASURES: RE-MODELING WEB SERVICES SECURITY ARCHITECTURE

The key to effective Web services security[11] is to know the threats as described previously, understand the technical solutions for mitigating these threats, and then establish and follow a defined engineering process that takes security into consideration from the beginning and throughout the Web service life cycle. This process can be established in the following two steps:

1. Deployment of Robust Encryption and Decryption Entity [RED] for Browser and Web Server
2. Deployment of Standard Encrypt Syntax

A. Robust Encryption and Decryption Entity [RED] for Browser and Web Server

A detailed conceptual framework is proposed on how safely to send valuable packets to destination without any security attacks related to web services. This proposed framework is based on providing a common encryption and decryption component that should be deployed in both web browser and the web server.

1. *RED Component:* A set of common standard encryption algorithm such as RSA, DES could be grouped together as a component and be deployed in both web server and web browser. This could be done by adding

- Add-on service
- Updating the browser/server

Every Algorithm has its own unique id in RED.
For eg: RSA id=3341
Standard Encrypt Syntax in the code calls a particular algorithm from RED to encrypt or decrypt a page.

2. *Deployment of RED In Browser:* Communication between browser [10] and RED is given below.

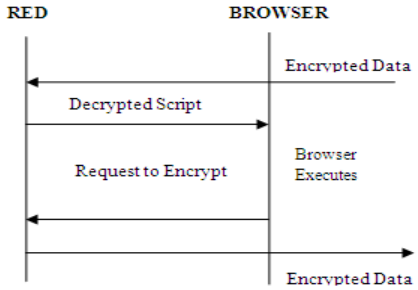

Fig 1.2: Communication between Browser and RED

3. *Deployment of RED in Web Server:* Communication between Server and RED, before and after a web page is uploaded in to server.

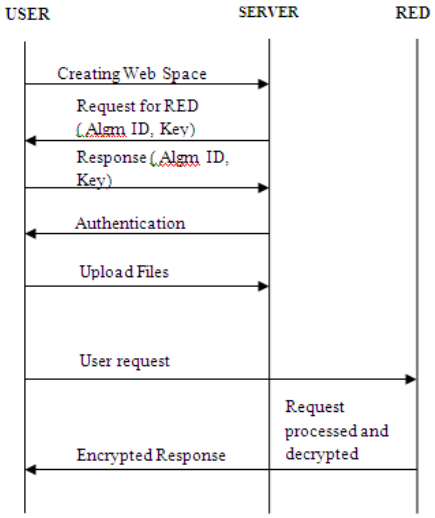

Fig 1.1: Communication between User, Server and RED

B. Deployment of SES

Standard Encrypt Syntax (SES) will be a common code embedded in any programming language to communicate with RED in browser and server.

1) *The syntax for accessing RED:*
<SES id=”Encryption Algorithm ID” name=”User defined Encryption name” key=”key_name”> </SES>
For every request been processed, RED encrypts or decrypts from the algorithm and key referred in SES tag.

2) *Algorithm with no key exchange:*
<SES id=”Encryption Algorithm ID” name=”User defined Encryption name” key=”no-key”> </SES>

3) *Algorithm with single key exchange:*
<SES id=”Encryption Algorithm ID” name=”User defined Encryption name” key=”key_name”> </SES>

4) *Algorithm with multiple key exchanges:*
Cryptographic algorithm with a multiple key[2] could be represented as,
<SES id=”Encryption Algorithm ID” name=”User defined Encryption name” key=”key_name1, key_name2,key_name3….”></SES>

5) *Public Key Algorithm:*
Public key Cryptographic algorithm[1] with private key and public key
<SES id=”Encryption Algorithm ID” name=”User defined Encryption name” publickey= ”key_name” private-key=”key_name”></SES>

C. Working of RED

The web developer after designing a webpage can use an encryption algorithm from RED which could be added as a tag in the webpage. While hosting the webpage to the server, the developer is intimated for choose of any cryptographic algorithm. Therefore, before a data is been send to communication channel, it encrypts with above mentioned algorithm and send.

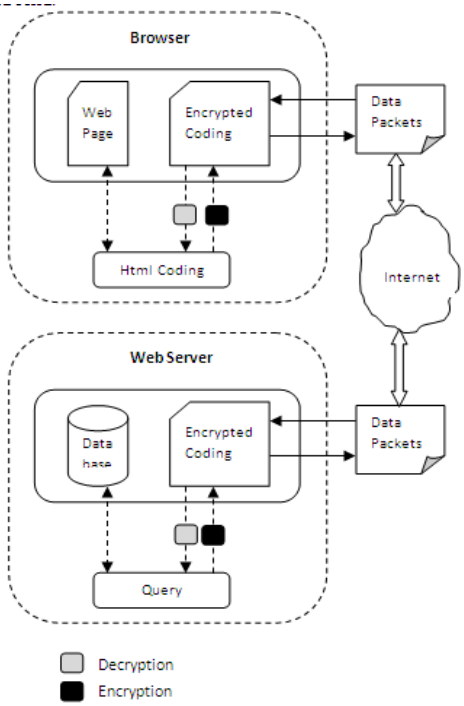

Fig 1.1: Robust Encryption and Decryption Component for Web Browser and Web Server

Similarly, when server response to a user, data is encrypted. Browser decrypts the content of the page by same cryptographic algorithm which is mentioned in the SES tag.

D. From Developer's Perspective:

There are certain options could be implemented from developer's perspective:

1) Determine a suitable encryption algorithm from RED

Developer can choose any algorithm [6] of his choice suitable for his webpage from RED. Developer need not be a cryptographer. Implementing a cryptographic algorithm is done by adding a small SES tag in to his web page.

2) Encrypt Priority fields

Encryption and Decryption technique can possibly place a high load on the server/client proportionate to the size of the webpage. Considering a webpage with content larger than 2mb, there is a possibility of time delay in loading the content of the webpage.

In order to reduce the time in loading and to provide more efficiency, the developer can encrypt a part of a webpage which he is going to give more priority. It may be a form field or a table of content. Encryption is done on priority basis. For instance consider an E-Commerce website. The Developer gives more attention on form fields and transaction links. Therefore he can encrypt that part of priority areas.

```
<body>
<div>…….</div>
<SES id=”2233” name=”Caesar” key=”no-key”>
<div> <form>…..</form></div>
</SES>
```

V. ADVANTAGES

1. Deploying RED component is done by simple updating or through add-ons.
2. Accessing RED is done by simple SES tag that is supported by RED.
3. Another advantage over sniffers is after the user types his username and password and clicks send, the username and passwords are encrypted and sent via packets. If an intruder tries to capture and view the packet

information, there will be no possibility of identifying the actual information. This ensures more security on networks.

4. Instead going for paid secure services, this framework can bring security to entire network.

VI. CONCLUSION

Due to enormous threats and vulnerabilities, there is a desperate need for some standards that should prevent and deter the attackers. Implementing this Robust Encryption and Decryption component and Standard Encrypt Syntax will add a higher level of security to the communication channel.

REFERENCES

[1] V. Shoup, "A proposal for an ISO standard for public key encryption", http://eprint.iacr.org/2001/112, 2001.
[2] S. S. Al-Riyami and K. G. Paterson. "Authenticated three party key agreement protocols from pairings." Cryptology e-Print Archive, http://eprint.iacr.org/2002/035/
[3] R. C. Summers, "Secure Computing" – Threats and Safeguards, McGraw-Hill, 1997
[4] CERT, "TCP SYN Flooding and IP Spoofing Attacks", Carnegie Mellon University, Sept. 1996
[5] Eastlake and P. Jones, US Secure Hash Algorithm 1 (SHA1), IETF RFC 3174, Sept. 2001.
[6] Choosing an encryption algorithm: technet.microsoft.com/en-us/library/ms345262.aspx
[7] Common types of network attacks: http://technet.microsoft.com/en-us/library/cc959354.aspx
[8] Understanding of digital certificates and SSL: http://www.entrust.net/ssl-resources/pdf/understanding_ssl.pdf
[9] Web security threats: http://www.gfi.com/whitepapers/GFI-Web_Based_Threats_v2_Whitepaper.pdf
[10] Browser security and privacy: http://www.securingthehuman.org/newsletters/ouch/issues/OUCH-201111_en.pdf
[11] Bhandari R, Suman U, "Generalized Framework for Secure Web Service Composition"