

GENERATING RANDOMIZED KEY FOR CIPHER TEXT WHILE PROCESSING AJAX FUNCTION IN A WEBPAGE

P. Karthika

Department of CS&IT, Nadar Saraswathi College of Arts & Science, Theni, Tamil Nadu, (India)

ABSTRACT

Data security is critical for most businesses and even home computer users. Client information can be hard to replace and potentially dangerous if it falls into the wrong hands. Data lost due to disasters such as a flood or fire is crushing, but losing it to hackers or a malware infection can have much greater consequences. To increase the data security, we may go for cryptography technology. Cryptography is a science that applies complex mathematics and logic to design strong encryption methods. Achieving strong encryption, the hiding of data's meaning, also requires intuitive leaps that allow creative application of known or new methods. But intellectual hackers may find the key and break the message. So we must focus to give the critical logic based key production. By using RSA algorithm, we produce the randomized keys to open the cipher text when AJAX function revitalized in that web page. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script. Every time a page performs the AJAX function it will produce a new randomized key. It will fabricate the strapping security for the message.

Keywords: RSA Randomized Key Generation, Cryptography, AJAX

I. INTRODUCTION

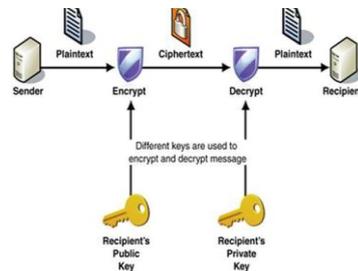
1.1 RSA

RSA is an algorithm used by modern computers to encrypt and decrypt messages. RSA involves a public key and private key. The public key can be known to everyone, it is used to encrypt messages. Messages encrypted using the public key can only be decrypted with the private key.

1.2 Crptography

Cryptography is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. Cryptography is closely related to the disciplines of cryptology and cryptanalysis. Cryptography includes techniques such as microdots, merging words with images, and other ways to hide information in storage or transit. Individuals who practice this field are known as cryptographers. Cryptography concerns itself with the following four objectives: Confidentiality, Integrity, Non-repudiation, Authentication. At a very high level, the RSA model uses prime numbers to create a public/private key set. Creation begins by selecting two extremely large prime numbers. They should be chosen at random and

of similar length. The two prime numbers are multiplied together. The product becomes the public key. The two factors become the private key.



1.3 AJAX

AJAX stands for Asynchronous JavaScript and XML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script. Ajax uses XHTML for content, CSS for presentation, along with Document Object Model and JavaScript for dynamic content display. Conventional web applications transmit information to and from the sever using synchronous requests. In the purest sense, the user would never know that anything was even transmitted to the server. XML is commonly used as the format for receiving server data, although any format, including plain text, can be used.

II METHODOLOGY

2.1 Proposed Work

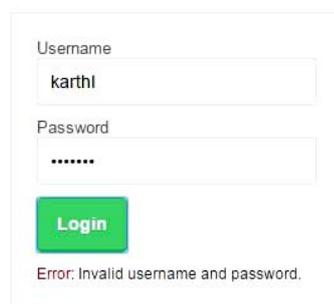
In existing system, secured message transfer performed by using different type of cryptography methodology. Every time we produce new methodology, Hackers may break the expertise and discover the confidential message. So we are in need to revolutionize our slant for creating key to decrypt the message. Here, we generate the randomized key for decryption.

```
public static void main(String[] args) throws Exception {
    // Subject to class loader constraints the Bouncy Castle Provider can be installed either dynamically or
    statically
    Security.addProvider(new org.bouncycastle.jce.provider.BouncyCastleProvider());

    byte[] input = "aa".getBytes();
    // To en/decrypt data without any padding an application may call
    Cipher cipher = Cipher.getInstance("RSA/None/NoPadding", "BC");
    SecureRandom random = new SecureRandom();
    KeyPairGenerator generator = KeyPairGenerator.getInstance("RSA", "BC");
    generator.initialize(256, random);
    KeyPair pair = generator.generateKeyPair();
    Key pubKey = pair.getPublic();
    Key privKey = pair.getPrivate();
    cipher.init(Cipher.ENCRYPT_MODE, pubKey, random);
    byte[] cipherText = cipher.doFinal(input);
    System.out.println("cipher: " + new String(cipherText));
    cipher.init(Cipher.DECRYPT_MODE, privKey);
```

```
byte[] plainText = cipher.doFinal(cipherText);
System.out.println("plain : " + new String(plainText));
}
}
```

Basic notion of this is, when we call this function it will erratically generate the key for the chipper text, but in our notion, the key will randomly generate when the AJAX page revitalized. When we apply the AJAX function in our page, it will refresh the division which one need to be changed instead of reload the total webpage. It will increase the user flexibility. That time we may call this RSA randomized key generation algorithm. Here, we add this methodology into the following dynamic web page.



- 1) Try Username: 9lessons and Password: 9lessons
- 2) Give wrong password and watch the shake effect.

```
$('#login').click(function()
{
var username=$('#username').val();
var password=$('#password').val();
var dataString='username='+username+'&password='+password;
if($.trim(username).length>0 && $.trim(password).length>0)
{
$.ajax({
type:"POST",
url:"ajaxLogin.php",
data:dataString,
cache:false,
beforeSend: function() { $("#login").val('Connecting...'); },
beforeSend: function(RSA),
success: function(data){
if(data)
{
$("#body").load("home.php").hide().fadeIn(1500).delay(6000);

```

```
//or
window.location.href="home.php";
}
else
{
//Shakeanimationeffect.
$('#box').shake();
$("#login").val('Login')
$("#error").html("<span style='color:#cc0000'>Error:</span> Invalid username andpassword. ");
}
}
});
```

When this program implement, consistently when the page laden for the incorrect login , it will achieve the AJAX exploit in the mean while randomized secret key will be generated.

III. CONCLUSION

To reduce the lacking of secured message transfer we may use the Randomized key generation combined with AJAX page loaded. It will engender the key often which is not easily traceable by the hackers and enhance the protected data transfer.

REFERENCES

- [1] Coutinho, S. C. The Mathematics of Ciphers: Number Theory and RSA Cryptography. Wellesley, MA: A K Peters, 1999.
- [2] Flannery, S. and Flannery, D. In Code: A Mathematical Journey. Profile Books, 2000.
- [3] Honsberger, R. Mathematical Gems III. Washington, DC: Math. Assoc. Amer., pp. 166-173, 1985.
- [4] RSA Laboratories. "TheRSAFactoringChallenge" <http://www.rsa.com/rsalabs/node.asp?id=2092>.
- [5] Meijer, A. R. "Groups, Factoring, and Cryptography." Math. Mag. **69**, 103-109, 1996.
- [6] Rivest, R. L. "Remarks on a Proposed Cryptanalytic Attack on the MIT Public-Key Cryptosystem." Cryptologia **2**, 62-65, 1978.
- [7] Ajax: The Complete Reference : Thomas Powell , McGraw Hill Professional, 2008.
- [8] <http://www.lynda.com/AJAX-training-tutorials/152-0.html>