Name:

Computer Security I: Homework 3

Due September 18, 2009 on compass:

1. Consider using AES-192 in cipher block chaining mode to encrypt plaintext m:
a.   What are the data elements must be communicated to the peer to ensure he can decrypt the ciphertext?
b.   What information must be kept private?
c.   If an error is introduced in the plaintext, how much information is lost?
d.   If an error is introduced in the ciphertext, how much information is lost?

2. Let the function f be a four-stage NLFSR be $f(r_0...r_{n-1}) = (r_0$ and $r_1)$ or r_3 and the initial value of the register is 1001. Derive the initial sequence and cycle size.

3. This question involves working with AES and DES encryption. Use the "openssl" utility that comes with the OpenSSL distribution to perform the following operations. OpenSSL is installed on the csil-linux machines which all members of the class should be able to access. csil-linux-ts1.cs.uiuc.edu and csil-linux-ts2.cs.uiuc.edu are configured for remote terminal access. "openssl" should be on your path. "openssl help" and then "openssl <keyword> help" will recursively give you more information about valid options.

    OpenSSL also runs on Windows platforms. I can provide a built OpenSSL tree from a Windows XP platform (should probably work on Vista too) on request.

    http://www.openssl.org/docs/apps/openssl.html is the root man page for the openssl tool.

    a. Fetch an encrypted file and a key file from http://www.cs.uiuc.edu/class/fa09/cs461/assignments/hw3-enc.bin and http://www.cs.uiuc.edu/class/fa09/cs461/assignments/hw3-key128.hex. Decrypt the file using 128 bit AES in ECB mode. Look at the "openssl enc" command. It should result in a plain English file. Submit the resulting plaintext. Note: the current version of openssl seems to insist on an IV file for all modes of AES. Just use all 0's for the initialization vector.
    b. Select an ascii text file to encrypt using AES in CBC mode. Submit the encrypted file, key file, size of key, IV, and any additional information we will need to decrypt the file. Do not include your binary file in the word document. Rather attach at least the binary file (i.e. the encrypted file) as a separate file in compass.
    c. Use the "speed" option for the "openssl" command to compare the performance of AES in ECB mode using key lengths 128, 192, and 256 bits, DES in ECB mode, and triple DES (DES in EDE mode). The "speed options

Name:

reports how many bytes are processed in a fixed amount of type. Report your results in bytes per second as reported by the "openssl speed" command.

Extra information on AES/DES implementations and invocations

The information in this section is not required to successfully complete your homework. Rather it is here if you wish to work with AES or DES a bit more.

Looking inside AES

If you are interested in looking at the inside of a C++ AES implementation, look at the references library at http://www.cs.uiuc.edu/class/fa07/cs461/aes-files.zip. This library includes a test AES program. It also includes a makeKey program which creates a key of the specified length using the rand() pseudo-random function.

Using Encryption in your programs

In question 3 we are using the crypto library of the OpenSSL implementation through the standard openSSL utility. You can write programs that directly invoke the crypto library API's to build your own encrypting applications. An example client program that uses AES 256 to encrypt and decrypt a string is posted at http://www.cs.uiuc.edu/class/fa08/cs461/example_crypt.cpp

On the csil-linux machines the program is compiled by "g++ -o test example_crypt.cpp -l crypto" and invoked with no arguments. The program is currently hard-coded to use aes_256 in CBC mode. There are some man pages that define the OpenSSL crypto messages.

.