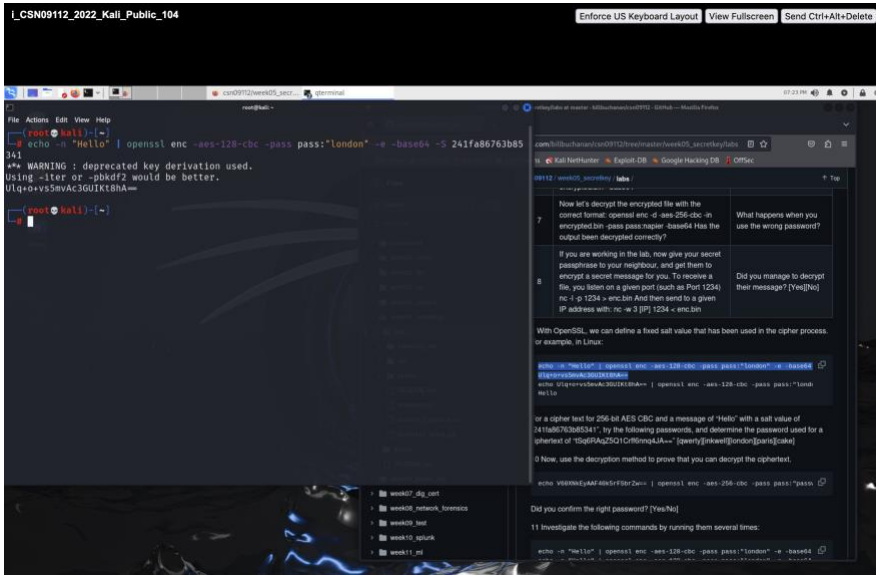
Lab 4: Symmetric Key and Hashing

Demo: http://youtu.be/3n2TMpHqE18

1 Symmetric Key

No	Description	Result
1	Log into vSoC 2, and select your Kali host on the public network. **You should open up the lab in your Kali instance, so that you can copy and paste from the document into your Kali console**. The link is here: https://github.com/billbuchanan/csn09112/tree/master/week05_secretkey/labs 	What is your IP address?

2	Use: `openssl list -cipher-commands` Use: `openssl version`	Outline five encryption methods that are supported: Outline the version of OpenSSL:
3	Using openssl and the command in the form: `openssl prime -hex 1111`	Check if the following are prime numbers: 42 [Yes][No] 1421 [Yes][No]
4	Now create a file named myfile.txt (either use nano or another editor). Next. encrypt with aes-256-cbc: `openssl enc -aes-256-cbc -in myfile.txt -out encrypted.bin -pbkdf2` and enter your password.	Use following command to view the output file: `cat encrypted.bin` Is it easy to write out or transmit the output: [Yes][No]
5	Now repeat the previous command and add the –base64 option: `openssl enc -aes-256-cbc -in myfile.txt -out encrypted.bin –base64 -pbkdf2`	Use following command to view the output file: `cat encrypted.bin` Is it easy to write out or transmit the output: [Yes][No]
6	Now repeat the previous command and observe the encrypted output: `openssl enc -aes-256-cbc -in myfile.txt -out encrypted.bin –base64 -pbkdf2`	Has the output changed from the run in 4? [Yes][No] Why has it changed?
7	Now let's decrypt the encrypted file with the correct format: `openssl enc -d -aes-256-cbc -in encrypted.bin -pass pass:napier -base64 -pbkdf2` Has the output been decrypted correctly?	Has the output been decrypted correctly? What happens when you use the wrong password?

8	If you are working in the lab, now give your secret passphrase to your neighbour, and get them to encrypt a secret message for you. To receive a file, you listen on a given port (such as Port 1234)
	
	`nc -l -p 1234 > enc.bin`
	
	And then send to a given IP address with:
	
	`nc -w 3 [IP] 1234 < enc.bin`
		Did you manage to decrypt their message?
		[Yes][No]

9	With OpenSSL, we can define a fixed salt value that has been used in the cipher process. For example, in Linux:
	
	`echo -n "Hello" | openssl enc -aes-128-cbc -pass pass:"london" -e -base64 -S 241fa86763b85341 -pbkdf2`
	
	and then decrypt:
	
	`echo 9Z+NtmCdQSpmRl+eZebFXQ== | openssl enc -aes-128-cbc -pass pass:"london" -d -base64 -S 241fa86763b85341 -pbkdf2`
	
	`Hello`
	For a ciphertext for 256-bit AES CBC and a message of "Hello" with a salt value of `241fa86763b85341`, try the following passwords, and determine the password used for a ciphertext of `tZCdiQE4L6QT+Dff82F5bw==`
		[qwerty]
		[inkwell]
		[london]
		[paris]
		[cake]

10	Now, use the decryption method to prove that you can decrypt the ciphertext. `echo tZCdiQE4L6QT+Dff82F5bw== | openssl enc -aes-256-cbc -pass pass:"password" -d -base64 -S 241fa86763b85341 -pbkdf2`	Did you confirm the right password? [Yes/No]

11	Investigate the following commands by running them several times:
	
	`echo -n "Hello" | openssl enc -aes-128-cbc -pass pass:"london" -e -base64 -S 241fa86763b85341 -pbkdf2`
	
	`echo -n "Hello" | openssl enc -aes-128-cbc -pass pass:"london" -e -base64 -salt -pbkdf2`
		What do you observe?
		
		Why do you think causes this (ask your tutor if you want some detail)?

12	We don't always need to use a file to save the cipher, too. With the following, we will encrypt the plaintext of "melon":
	
	`echo "melon" | openssl enc -e -aes-128-cbc -pass pass:stirling -base64 -pbkdf2`
	`U2FsdGVkX18cryB3vdNj+Tax1PGecO6ZOW2WL1LmdKQ=`
	
	and then we can decrypt with:
	
	`echo "U2FsdGVkX18cryB3vdNj+Tax1PGecO6ZOW2WL1LmdKQ=" | openssl enc -d -aes-128-cbc -pass pass:stirling -base64 -pbkdf2`
	
	`melon`
	
	Now crack the following cipher using a Scottish city as a password (the password is in lower case):
	
	`U2FsdGVkX1+7VpBGwevibQGgescaz5nsArtGLNqFaXk=`
	
	What is the fruit in the plaintext?
	
	Now try:
	
	`U2FsdGVkX18vpjgccu7VkPZrkncqADuy1kVKU9LbLec=`
	
	What is the fruit?
		Fruit:
		
		Fruit:

2 Hashing

http://youtu.be/Xvbk2nSzEPk

The current Hashcat version on Kali has problems with a lack of memory. To overcome this, install Hashcat 6.0.0. On Kali on your public network, first download Hashcat 6.0.0:

```
wget https://hashcat.net/files/hashcat-6.0.0.7z
```

Next unzip it into your home folder:

```
p7zip -d hashcat-6.0.0.7z
```

Then from your home folder, setup a link to Hashcat 6.0.0:

```
ln -s hashcat-6.0.0/hashcat.bin  hashcat
```

```
# ./hashcat --version
v6.0.0
```

No	Description	Result
1	Using: http://asecuritysite.com/encryption/md5 Match the hash signatures with their words ("Falkirk", "Edinburgh", "Glasgow" and "Stirling"). 03CF54D8CE19777B12732B8C50B3B66F D586293D554981ED611AB7B01316D2D5 48E935332AADEC763F2C82CDB4601A25 EE19033300A54DF2FA41DB9881B4B723	03CF5: Is it [Falkirk][Edinburgh][Glasgow][Stirling]? D5862: Is it [Falkirk][Edinburgh][Glasgow][Stirling]? 48E93: Is it [Falkirk][Edinburgh][Glasgow][Stirling]? EE190: Is it [Falkirk][Edinburgh][Glasgow][Stirling]?

2	Using: `http://asecuritysite.com/encryption/md5` Determine the number of hex characters in the following hash signatures.	MD5 hex chars: SHA-1 hex chars: SHA-256 hex chars: How does the number of hex characters relate to the length of the hash signature:

3	On Kali, for the following /etc/shadow file, determine the matching password:
	
	`bill:$apr1$waZS/8Tm$jDZmiZBct/c2hysERcZ3m1`
	`mike:$apr1$mKfrJquI$KxOCL9krmqhCuOSHKqp5Q0`
	`fred:$apr1$Jbe/hCIb$/k3A4kjpJyC06BUUaPRKs0`
	`ian:$apr1$OGyPhsLi$jTTzWOHNS4Cl5ZEoyFLjB.`
	`jane: $1$rqOIRBBN$R2pOQH9egTTVN1Nlst2U7.`
		The passwords are password, napier, inkwell and Ankle123.
		[Hint: openssl passwd -apr1 -salt ZaZS/8TF napier]
		
		Bill's password:
		
		Mike's password:
		
		Fred's password:
		
		Ian's password:
		
		Jane's password:

4	On Kali, download the following:
	
	`http://asecuritysite.com/files02.zip`
	
	and the files should have the following MD5 signatures:
	
	`MD5(1.txt)= 5d41402abc4b2a76b9719d911017c592`
	`MD5(2.txt)= 69faab6268350295550de7d587bc323d`
		Which file(s) have been modified:

	MD5(3.txt)= fea0f1f6fede90bd0a925b4194deac11 MD5(4.txt)= d89b56f81cd7b82856231e662429bcf2 Note: You can use **md5sum** to get the MD5 hash of the files.	
5	From Kali, download the following ZIP file: http://asecuritysite.com/letters.zip	View the letters. Are they different? Now determine the MD5 signature for them. What can you observe from the result?

3 Hashing Cracking (MD5)

No	Description	Result
1	On Kali, next create a words file (**words**) with the words of "napier", "password" "Ankle123" and "inkwell" Using hashcat crack the following MD5 signatures (hash1): -m 0 232DD5D7274E0D662F36C575A3BD634C 5F4DCC3B5AA765D61D8327DEB882CF99 6D5875265D1979BDAD1C8A8F383C5FF5 04013F78ACCFEC9B673005FC6F20698D Command used: hashcat -m 0 hash1 words Note: use the --show option to show your results	232DD...634C Is it [napier][password][Ankle123][inkwell]? 5F4DC...CF99 Is it [napier][password][Ankle123][inkwell]? 6D587...5FF5 Is it [napier][password][Ankle123][inkwell]? 04013...698D Is it [napier][password][Ankle123][inkwell]?

2	Using the method used in the first part of this tutorial, find crack the following for names of fruits (the fruits are all in lowercase): FE01D67A002DFA0F3AC084298142ECCD 1F3870BE274F6C49B3E31A0C6728957F 72B302BF297A228A75730123EFEF7C41 8893DC16B1B2534BAB7B03727145A2BB 889560D93572D538078CE1578567B91A	FE01D: 1F387: 72B30: 8893D: 88956:

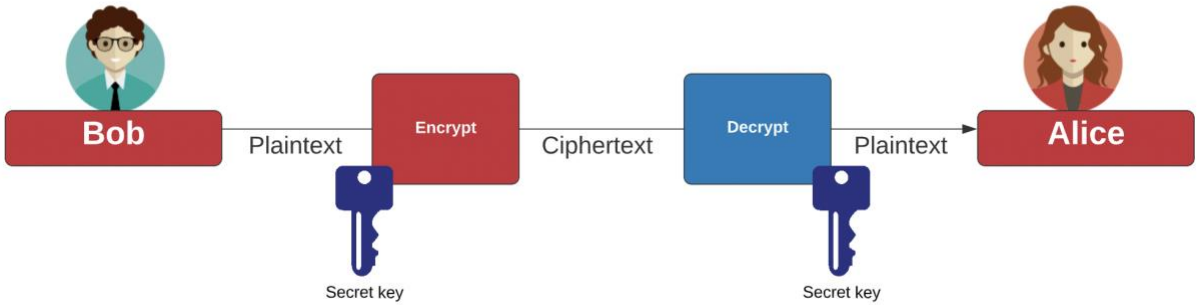
4 Hashing Cracking (LM Hash/Windows)

All of the passwords in this section are in lowercase.

No	Description	Result
1	On Kali, and using John the Ripper, and using a word list with the names of fruits, crack the following pwdump passwords: fred:500:E79E56A8E5C6F8FEAAD3B435B51404EE:5EBE7DFA074DA8EE8AEF1FAA2BBDE876::: bert:501:10EAF413723CBB15AAD3B435B51404EE:CA8E025E9893E8CE3D2CBF847FC56814:::	Fred: Bert:
2	On Kali, and using John the Ripper, the following pwdump passwords (they are names of major Scottish cities/towns): Admin:500:629E2BA1C0338CE0AAD3B435B51404EE:9408CB400B20ABA3DFEC054D2B6EE5A1::: fred:501:33E58ABB4D723E5EE72C57EF50F76A05:4DFC4E7AA65D71FD4E06D061871C05F2::: bert:502:BC2B6A869601E4D9AAD3B435B51404EE:2D8947D98F0B09A88DC9FCD6E546A711:::	Admin: Fred: Bert:
3	On Kali, and using John the Ripper, crack the following pwdump passwords (they are the names of animals): fred:500:5A8BB08EFF0D416AAAD3B435B51404EE:85A2ED1CA59D0479B1E3406972AB1928::: bert:501:C6E4266FEBEBD6A8AAD3B435B51404EE:0B9957E8BED733E0350C703AC1CDA822::: admin:502:333CB006680FAF0A417EAF50CFAC29C3:D2EDBC29463C40E76297119421D2A707:::	Fred: Bert: Admin:

5 AWS Cryptography

We are generally moving our security into the public cloud, and thus many of our keys are stored there. In AWS, we use KMS (Key Management System), and can create either symmetric keys or asymmetric keys (public keys). With symmetric key, Bob and Alice use the same encryption key to encrypt and decrypt:



Now complete the tutorial at:

https://asecuritysite.com/aws/lab03