

C2107 Tutorial 1 (Intro & Encryption)

School of Computing, NUS

January 20, 2021

Remark: Not all questions would be covered by the instructors during tutorial.

1. Alice was the Web administrator of the company VIC¹. A malicious attacker sent an email to Alice. The email instructed Alice to click on a link so as to login to the HR's system to view a report. In the email, information on the "sender" indicated it was from the HR manager in VIC. Alice wrongly believed that the email was indeed sent by the manager and followed the instructions. In doing so, she revealed her password to the attacker. Using Alice's password, the attacker logged-in to the web-server, and invoked many processes. As a result, the server was overloaded.

With respect to the security requirements mentioned in the lecture (confidentiality, integrity, authentication, availability, etc), discussed what aspects of security were compromised.

2. Suppose it takes 512 clock cycles to test whether a 64-bit cryptographic key is correct, when given a 64-bit plaintext and the corresponding ciphertext.

- (a) How long does it take to exhaustively check all the keys using a 4 GHz (single-core) processor?
- (b) How long does it take on a cluster of 1024 servers, each with a quad-core 4GHz processor.

(Hint: For simplicity, you can take 1 year $\approx 2^{25}$ seconds. We follow the notations where $1K = 2^{10}$, $1M = 2^{20}$, $1G = 2^{30}$.)

3. Suppose it takes 512 clock cycles to test whether a 36-bit cryptographic key is correct, when given a plaintext m and the corresponding ciphertext c . Length of plaintext irrelevant in this question.

How long does it take to exhaustively check all the keys using a 4GHz (single-core) processor? Using exhaustive search, is it then possible to crack a ciphertext and obtain its plaintext in *realtime*, say within 0.1 second?

A walkie-talkie system *Secure Walkie Talkie* (rSWT)¹ secures its communication using symmetric keys encryption. rSWT uses two encryption schemes, AES block cipher, and another fast stream cipher developed by the company called FAST. The cipher FAST is really fast, but its key length is only 32 bits.

¹Companies are fictional.

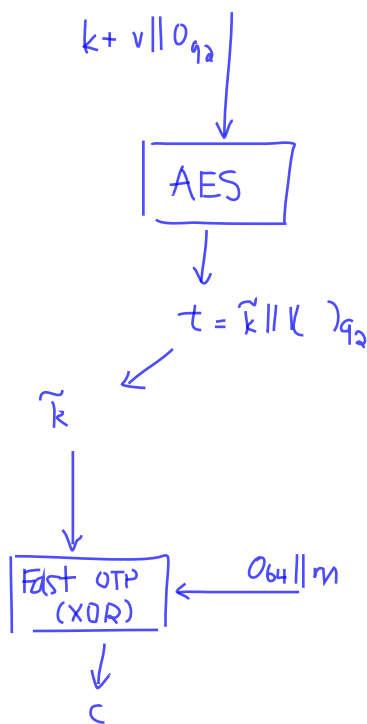
$$2^{41} = \frac{2^{64} \times 2^7}{2^{32}}$$

$$\frac{2^{64} \times 2^7}{2^{32} \times 2^{12}} = \frac{2^{41}}{2^{44}} = 2^{-3}$$

$$\frac{2^{36} \times 2^7}{2^{32}} = 2^{11}$$

$$2^{10} \times 2^4 \times 2^{32}$$

$$4 \times 2^{30} = 2^{32}$$



During installation, the user enters k , a 128-bit *master key*, into each walkie-talkie. After installation, when a walkie-talkie wants to send a plaintext m to another, the sent signal is computed in the following ways:

- A 36-bit v is randomly chosen.
- Computes $t = \text{AES}_{\text{enc}}(k, v)$, where AES_{enc} is encryption of AES block cipher (without mode of operation) and v is padded with zeros.
- Obtains \tilde{k} , which is the first 36 leading bits of t . This \tilde{k} is called the *temporary key*.
- Computes $c = \text{FAST}_{\text{enc}}(\tilde{k}, 0_{64} || m)$, where 0_{64} is a string of 64 zeros, $||$ is string concatenation, and FAST_{enc} is the deterministic encryption of FAST. Note that c does not contain initial value.
- Sends $(v || c)$ over the air.

We consider attackers who can eavesdrop the ciphertexts (both v and c) transmitted over the air.


We know that 36-bit is too short and the key can be broken, but, as calculated before, it would take very long time. In their marketing efforts, rSWT claims that 36-bit is sufficient for realtime applications. This is what appeared in their advertisement:

“36-bit is sufficient. By the time the message is maliciously decrypted, the message becomes useless”.

\tilde{k}	$\text{FAST}_e(\tilde{k}, 0_{64})$
00...00	$c_0 = \dots$
00...01	$c_1 = \dots$
\vdots	
\tilde{k}_{36}	$c_{2^{36}}$

Now, you want to design a hand-held device that is able to crack and obtain the plaintext *realtime*. Specifically, when given the v and c , the device should derive the 36-bit session key readily within 0.1 second. The hand-held device can have computing resource comparable to a laptop. Suggest a way to get the temporary key.

(Hint: Use storage to help. Here, we assume that the hand-held device can hold a large, say 1TB, of pre-computed table whereby the key can be looked up. We use the notations where $1K = 2^{10}$, $1M = 2^{20}$, $1G = 2^{30}$.)

- Lecture 1 mentioned that Winzip encrypts the **compressed file**. Why it is meaningless to carry out the two operations  the other way, that is, encrypts the file, and then compresses the encrypted file?

(Hint: Consider the effectiveness of compression on “random” sequences, and a **requirement of encryption scheme**.)

- Bob encrypted a music mp3 file using Winzip, which employs the 256-bit key AES. He chose a 6-digit number as password. Winzip generated the 256-bit AES key from the 6-digit password using a (deterministic) function, say SHA1.

Alice obtained the ciphertext. Alice also knew that Bob used a 6-digit password and knew how Winzip generated the AES key.

- (a) Give a 256-bit string, can Alice determine whether this string was indeed the correct AES key?
 - (b) How many guesses did Alice really need in order to get the `mn3` file?
6. Compare Symantec Internet Security Threat Report in 2019 and 2009. Discuss what are new and what remain over 10 years. (Open-ended discussions. No right or wrong. The link for 2009 in lecture note is broken. Try <http://www.securityprivacyandthelaw.com/uploads/file/symantec%202009.pdf>)
7. Find out more about these terminologies:
- (a) *Cryptology, Cryptanalysis, Cryptography,*
 - (b) *NSA, NIST, cryptography, backdoor, Key Escrow, Decryption order*
- Find out more about the following well-known persons in cryptography: *Whitfield Diffie, Ron Rivest, Alice, Bob, Eve, Mallory and Trent.*
- (optional) Can NSA break AES? Can NSA by-pass cryptography?

Hands-on Exercise: Linux Set-Up

A Linux system would be required for many higher level security module, and likely for our CS2107 assignments. Likely that you are using Windows or MacOS. Hence, you should know how to set up a **Linux host** using VM (Virtual Machine).

An **Ubuntu desktop** is recommended since it is user friendly enough even for new users. Please use a recent Ubuntu version, such as Ubuntu 16.04.5 LTS (Xenial Xerus), which is available from: <http://releases.ubuntu.com/16.04/>. A 32-bit PC (i386) desktop image is sufficient.

For Windows/MacOS, you can use **VirtualBox** or **VMWare** to create an **Ubuntu VM**. You can follow the steps described in: <https://www.lifewire.com/run-ubuntu-within-windows-virtualbox-2202098>. Simply use the “NAT” or “bridge adapter” connection/networking mode for your VM, so that it can access the Internet.

It is also expected that you have rudimentary proficiency in using a Linux system. You can read the tutorial given at: <https://www.digitalocean.com/community/tutorials/an-introduction-to-the-linux-terminal>. However, more knowledge might be needed, and it is expected that you do some self-exploration. You may want to refer to this freely-downloadable good book on Linux: “**The Linux Command Line**”, which is available from <http://linuxcommand.org/tlcl.php>.

If we indeed require Linux for the assignment, there would be open consultation session after assignment is released. Nonetheless, do self exploration now in setting up your Linux system. Setting up test environment in fact is a “skillset” required for security professionals.