2/14 Introduction to Network Security

Two presentations today because of last Thursday's snow day

Presentation #1: Apple vs FBI Case

- Started with San Bernardino shooting & FBI wanted to access one of the shooters iPhones for evidence
- The iCloud is not secure
    - Apple originally wanted the FBI to connect to the iCloud via a known and recently used Wi-Fi network of the shooter but the FBI messed up and got locked out, making their only option hacking into the iPhone
- 4 Major Options in This Case
    - Apple helps the FBI: no this is bad practice and sets a legal precedent
    - FBI uses iCloud: no messed up
    - FBI finds password from user: no the shooters were killed
    - FBI hacks in iPhone: only remaining option
- The FBI and Apple ended up engaging in a long and lengthy court case where the FBI wanted Apple to build in a backdoor to the iOS, with warrants and various other issues but Apple continued to refuse
- Eventually the FBI did hack the iPhone
    - Unsure how, FBI have not shared
    - Most likely methodology was that the FBI put the phone in DFU mode (which restores the OS) and reverted the iPhone to an earlier model
        - This is significant because the iPhone was running iOS8 which set a limit on the number of failed password attempts allowed before wiping the device
        - Previous iOS7 allowed for this mode to be disabled and the FBI could have brute force hacked in while keeping the data's encryption/integrity
- A brief overview of iOS hardware and the difference between complete and complete until first authentication was covered
    - iOS8 is complete which is more secure
    - Other versions of iOS were complete until first authentication

Presentation #2: Equation Group Breach and EXTRABACON

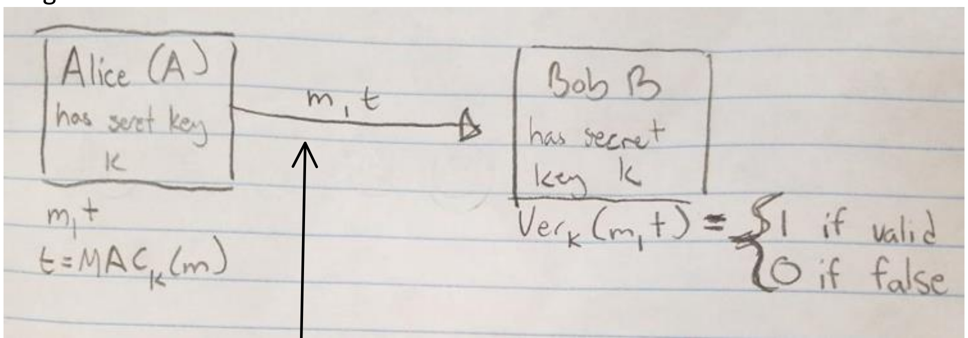
- Recently, various hacks and exploits from a secure hacking group were stolen and put up for sale. The victims were the Equation Group and the shadow brokers/attackers are unkown
- The Equation Group is an group that exploits various hacks
    - It is theorized to be an analysist side group of cryptographers working with the NSA
- There are several theories as to the identities of the Shadow Brokers
    - The people selling the information
    - Snowdin thinks it could be Russia
    - Other people think it may be an inside leak
- EXTRABACON was a small snippet of the breach released to prove the authenticity of the offered information
    - If a hack/exploit aimed at Enterprise/NSP routers
    - Why target routers?
        - All devices or many devices can be connected to a router which is itself either behind or merged with a firewall. If an attacked can gain access to a router then they can control your data without issue
    - Exploits ssh connections
- There have been Updates made to make the released attack more difficult - but otherwise the situation has not been resolved

CLASS NOTES:

Last class we learned about message authentication codes or MACs

- MACs grant integrity but do not encrypt the message

Diagram of a Basic MAC:

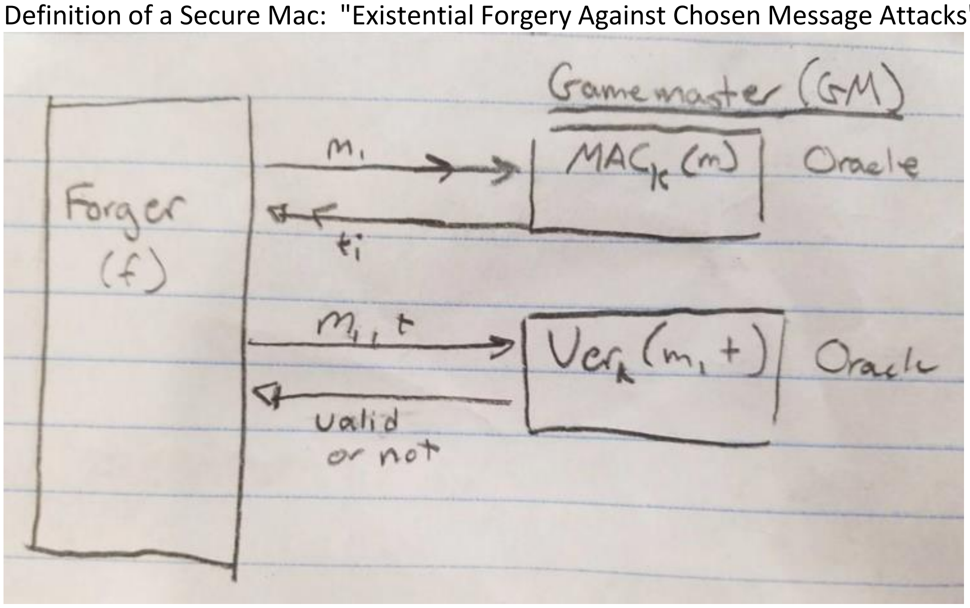


This works even if the adversary is intercepting and sitting where the black box is. He could try to modify the message but because he does not know the secret key it would be invalid.

Given a mac scheme, say MD5(k||m) how do you guarantee it is secure

- MD5 is a bad MAC because it is vulnerable to length extension attacks

Definition of a Secure Mac: "Existential Forgery Against Chosen Message Attacks"



- The forger can send and verify messages (m) and tags (t) of his choice
- To win the forget must construct m*, t* that are deemed valid
    - The forger cannot query for the tag of m*
- The forget is allowed to choose and construct messages however he wants because it makes this game easier for him && more accurately models the real world

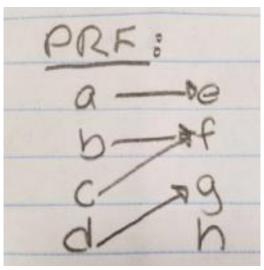
Q: Can the GMs Oracle's have overlapping keys? Are the MACs PRG (Pseudo Random Generators?)
A: It depends; the theoretical game above implements whatever MAC scheme is being tested for security.

Let's review how a length extension attack for an MD5 could be carried out for the above game (thus proving MD5 not secure):
F -> m -> GM's MAC Oracle
GM's MAC Oracle -> t -> F
F adds m + m' = m*
F derives t* from m and t using length extension attack (see Lab 01)
F -> m*, t* - GM's Ver Oracle
F WINS!

HMACS are good MACs that prevent length extension attacks, a sample HMAC for MD5 would be:
$HMAC - MD5 = MD5(k \oplus ipad\ || MD5(k \oplus opad || m))$

- HMAC's are not vulnerable to length extension attacks
- How does HMAC provide extra protection?
  - It requires the secret key k a second time to make a valid tag and because the forger never knows the key he cannot construct a valid tag from the information given

Given a MAC with PRF (Pseudo Random Function) $PRF_k(m) = t$ is a good MAC

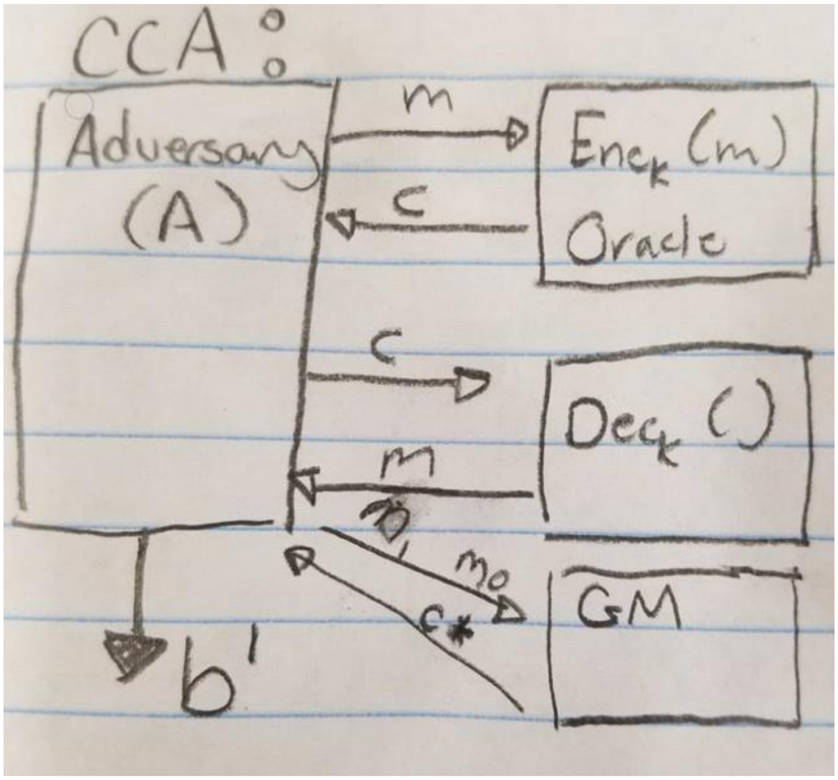


A PRF has single and consistent output but does not have to be deterministic (of used for a MAC)

MAC's in Practice:
Why do we need to MAC everything on the internet?

- MAC's provide CCA level security

Below is a CCA security game (see Lab 02)



- To win CCA the Adversary (A) must choose random bit b such that c* = $Enc_k(m_b)$
- Adversary A can send and decrypt any message they want except for c*
- A CPA scheme is a CCA scheme without the Decryption oracle
- CPA and CCA are not encryption schemes but are instead security definitions

To be CCA secure on the internet you must also MAC the message.
- Let Enc & Dec be CCA secure encryption (e.g. CBC mode)
- Let MAC & Ver be a proven secure message authentication code
- Add MAC to the Encryption Oracle and Ver to the Decryption Oracle

Adversary A can now send messages m and receive c, t back but the Decryption oracle is now useless because it will always output a fail. Why?
- A cannot generate tags and thus cannot send c,t to the Dec Oracle - the adversary can only send a known c which means that without a valid tag the Decryption Oracle will always return false to the adversary and is thus useless.

Anytime you encrypt something you want to MAC it
- Integrity and prevents CCA attacks (aka everything on the internet has MAC)