Homework #1

Oregon State University

ECE 478 Network Security

Spring 2016

Student:
Sam Quinn
Quinnsa@Oregonstate.edu

Professor:
Dr. Attila A Yavuz
Attila.Yavuz@oregonstate.edu

April 18, 2016

Oregon State
UNIVERSITY

College of Engineering

Contents

1 Question about basic concepts:

What are the main security/performance trade-offs for symmetric and asymmetric cryptography to be deployed in practice? Could you give real-life scenarios where symmetric and asymmetric crypto could be more suitable?

Symmetric and asymmetric cryptography are the two basic forms of encryption with each functioning differently than one another. Symmetric cryptography is faster but needs the secret key share between all parties before hand. Since there is one secret key that is shared between all parties before hand it all encryption as well as decryption is done with the same key, meaning that all parties use the same key. Symmetric is better when dealing with only trusted parties since you cannot establish authentication with only one secret key. Symmetric encryption is ideal for encrypting personal files where the secret key would not need to be shared.

Asymmetric cryptography dose not need the secret key to be shared before hand. Asymmetric encryption needs two keys, a private key and a public key. The public key may be shared with the presence of an eavesdropper with no security penalties. Asymmetric encryption keys do not need to be pre-shared with parties, and since every party has their own key pair it will also provide authentication. Asymmetric cryptography is ideal for applications where both parties have never spoken before, for example SSH.

OTPs are highly secure, but why we do not see them much in practice?

One time pad is one of the highest security since each bit of the plaintext is randomized, the random pattern used to in the OTP can only be used once or information can be extracted form the cipher text. The main problem with OTP is that the size of the key must be the size of the data, and a new key must be generated every message.

Does Kerckhoffs's principle contradict with the "secret algorithm" practice in military systems? Given sufficient financial capability, how could you incorporate Kerckhoffs's principle into such high-end systems?

The security systems used in the military are often classified until they come up with a new method in which the old systems are released to the public. Witholding the current system from the public introduces the property of security through obscurity since the adversary does not know the algorithims used. One way to introduce Kerckhoffs's principle in to military systems would be to hire a team of skilled cryptographers to study the security systems used thus giving them complete knowledge of the system minus the key. If the team are able to break the system then the only thing that is keeping the military security system secure is the obscurity which is very weak. However if the team is unable to break the system knowing everything but the key then the system is secure plus the added bonus of being obscure.

2 Data Encryption Standard (DES). DES has weak keys.

What is the difference between a weak key, a semi-weak key and a possible weak key?

The origin of weak, semi-weak, and possibly weak keys in DES start with how DES extracts each rounds encryption keys from the master 56bit key. The 56bit key is divided into 16 sub keys one for each cycle of the DES encryption. A weak key would be $\{0\}^{56}$ or $\{1\}^{56}$ where all of the bits are the same thus making the ratating of the keys for each cycle useless. A semi-weak key would be a key that would reverse the cycle that took place in the previous cycle. A key that repeats with a period of 2 meaning that the rotation of keys produces exactly two keys that will be each be used 8 times. Possibly weak keys are keys that repeat with a period of 4, outputing 4 unique keys that will all be used 4 times each. (http://search.proquest.com/openview/6cc5dfe1f7c352582f2b62aa741b47dd/1?pq-origsite=gscholar)

What is double DES? What kind of attack on double DES makes it useless?

Double DES does two rounds of DES encryption with two different keys, thus making the key 112 bits long. With a meet in the middle attack the added benifit of encrypting twice does not increas the security much. Meet in the middle attack. The effective security gain from the single DES 2^{56} is increased by one exponential to 2^{57} in double DES. Double DES is useless because an adversary could brute force the first round (single DES) and then just pass the output to the second round without the need for any brute forcing. (http://stephanemoore.com/pdf/meetinthemiddle.pdf)

What is triple DES? How many keys are use in triple DES process?

Triple DES uses three rounds of DES making the maximum security 168 bits. Triple DES can either use 2 or 3 keys, however the effective security due to the meet in the middle attack explained above is 2^{112}.

3 Block Cipher Design Principles and AES

What is the basic design technique, which is frequently used to construct modern symmetric ciphers?

Symmetric crpytography wil use techniques to obscure the data with confusion and diffusion. Confusion is mainly accomplished by substitutions, replacing original data with new data. Diffusion is mainly accomplished by premutations, where one bit changes will permute on to all the ciphertext data to further scramble the original input. Modern symmetric ciphers will alternate substitutions and permutations.

What are the main security properties achieved via this designed technique?

Confusion in a symmetric cipher will help prevent attacks linear attacks. Because the data is passed through a non-linear table often created by the secret key the data is translated in a non-linear way.

Diffusion will help prevent against patern analysis attacks. Since every language has traceable patterns in grammar, diffusion makes these patterns much harder to spot. If one bit is changed the entire ciphertext will be changed not just the coresponding bit within the ciphertext.

Given the example of AES, which functional steps enable achieving these properties? Please provide specific names of these operations and briefly explain how they are applied in AES (all answers are brief for this question)?

• Confusion: S-Box - Rijndael S-Box is a lookup matrix generated by deteriminng the multiplicative inverse of each byte from the original input.
• Diffusion: MixColumn - AES uses a MixColumn function on the Rijndael matrix that permutates any change with data through the use of XORs.

What are the benefits of the use of finite field arithmetic in AES?

4 Symmetric Encryption Modes

We have discussed various Symmetric Encryption Modes. Each of these modes satisfies certain properties, which can be an advantage or disadvantage for a given application. Construct a table providing a summary information about each mode and its corresponding properties. For example, each row of the table will be properties (e.g., parallel operation, ciphertext manipulation, pre-computation, etc., please see course notes for more properties) and columns are Modes (e.g., CBC, CTR). Each cell will take a value such as Yes, No, partially, high, low etc. according to given encryption mode and property.

5 Ciphertext manipulability

Ciphertext manipulability is generally considered as an undesirable property for Encryption Modes. However, for modes that operates in stream cipher fashion (discussed in class), by design, it is possible to flip bits in plaintext by flipping bits of ciphertext. Why is this possible? Is there a way to turn this (potentially) undesirable property into an advantage, describe how if there is one? Which extra cryptographic function (a group of functions discussed in the class) can be applied to the ciphertext so that the aforementioned advantage can be obtained without compromising the security?

Hints: (i) Consider noisy communication channels as application domain to exploit this feature. (ii) The extra cryptographic function will require annexing a small-constant tag to the ciphertext.

6 Encryption and Compression

Encryption (E) and Compression (C) are generally used together to achieve confidentiality and efficiency simultaneously. Given a message M, with which order function E and C must be applied? What are the reasons behind of this particular order?

7 Symmetric key Authentication

In symmetric key cryptography, it is desirable to achieve both confidentiality and authentication (also provides integrity) of the data. These properties can be achieved via Encryption (E) and Authentication Functions (A), respectively. What is the correct order of these operations? Lets assume the specific notation and order of these operations are as follows:

- **Authenticate-then-Encrypt (AtE)**
- **Encrypt-then-Authenticate (EtA)**
- **Encryption and Authentication (E&A) or the opposite way as (A&E)**

Discuss the security implications of these choices, which one is recommended and why?

Mention important crypto papers (at least one, cite it), in which the security of an important real-life protocol (Hint: the protocol that securely connects your VPN for each e-commerce transaction!) analyzed based on the above orders. Explain why this order matters a lot in practice? You may provide some discussions from these papers (please be brief, just hit on important points).

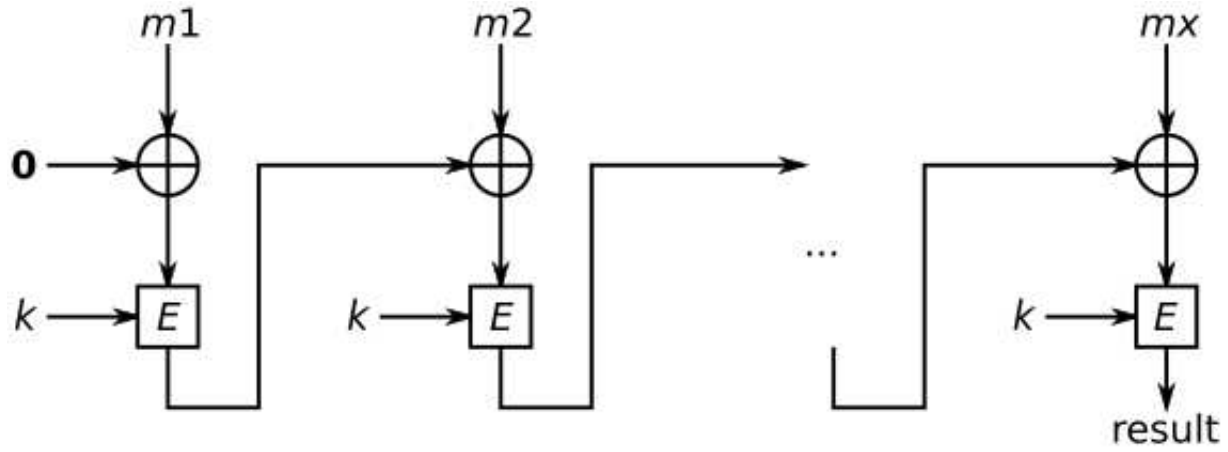
8 Cryptographic hash functions

Given a modern cryptographic hash function (e.g., SHA) with m-bit length output, what is the maximum security it can provide in terms of m (lets call it x-bit security)? A generalized proof for any given m-bit hash function is available to show why it can achieve at best x-bit security. Please describe this generalized proof (related birthday attack concept) that simply connects m-bit to x-bit.

9 Length extension

What is the hash length extension attack? Please describe by giving some specific real-life examples.

A length extension attack is an attack on HMACs with messages of varining length where an adversary may append new data to the authenticated message with the same signature.



CBC-MAC is suseptible to a lenghth extention attack when the messages are not restricted to a single length. CBC-MAC will take the input message and compute the CBC encryption of the message. With the idea of a MAC to shorten the input to a fixed size, CBC-MAC will only export the last block of the message into its signature. As described above the output from e will be $\otimes$ by mx until the last block. An adversary is able to attack this MAC if they take the *result* form the original MAC m and $\otimes m_1'$ with the *result*. This would make the *result* cancel out the *result*$\otimes$ that was in m_1'. After we break the *result* with m_1' we are able to append anything with in m' to the end of m with the same signature.

$$CBC\_MAC(k, m_1, ..., m_l) \rightarrow (m, t)$$
$$CBC\_MAC(k, m_1' \otimes t, ..., m_l') \rightarrow (m \| m', t')$$

10 Properties of cryptographic hash function

What are the essential properties that a cryptographic hash function must satisfy? What is a Random Oracle and how does it play a role in the security proofs in general (what is its relation with cryptographic hash functions?)

References