Section 2: Cryptography

18. Cryptography Basics

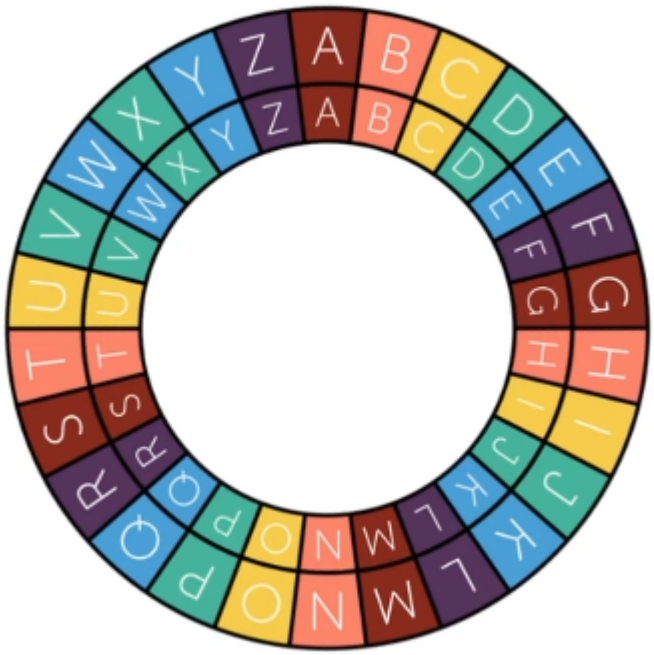
Obfuscation: To take something that makes sense and hide it so it does not make sense to the outside observer.

`-Diffusion`
`-Confusion`

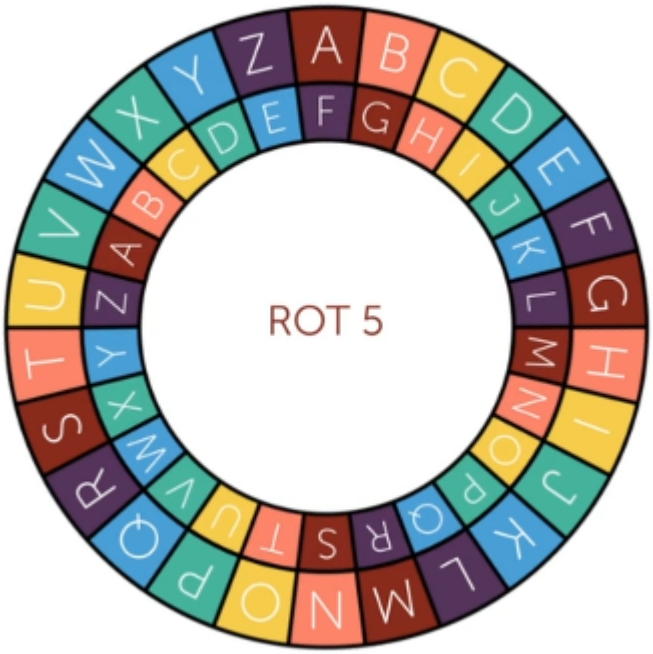
Encryption/Decryption -> Hiding something and bringing it back

Caesar Cipher:

`Decoder ring:`



`Substitution: Replacing a letter with a different letter:`




```
Rotating twice = ROT2
Rotating Three times = ROT3

The Ceaser Cipher is easy to crack.
```

Crytanalysis: Breaking encrypted codes

Vigenere Cipher

PLAINTEXT

0	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
4	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
13	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
14	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
15	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
25	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Use a Key to encrypt the plain text:

FACE

F A CEFACE FA CEFA

We Attack at Dawn

F A CEFACE FA CEFA
We Attack at Dawn
B E CX

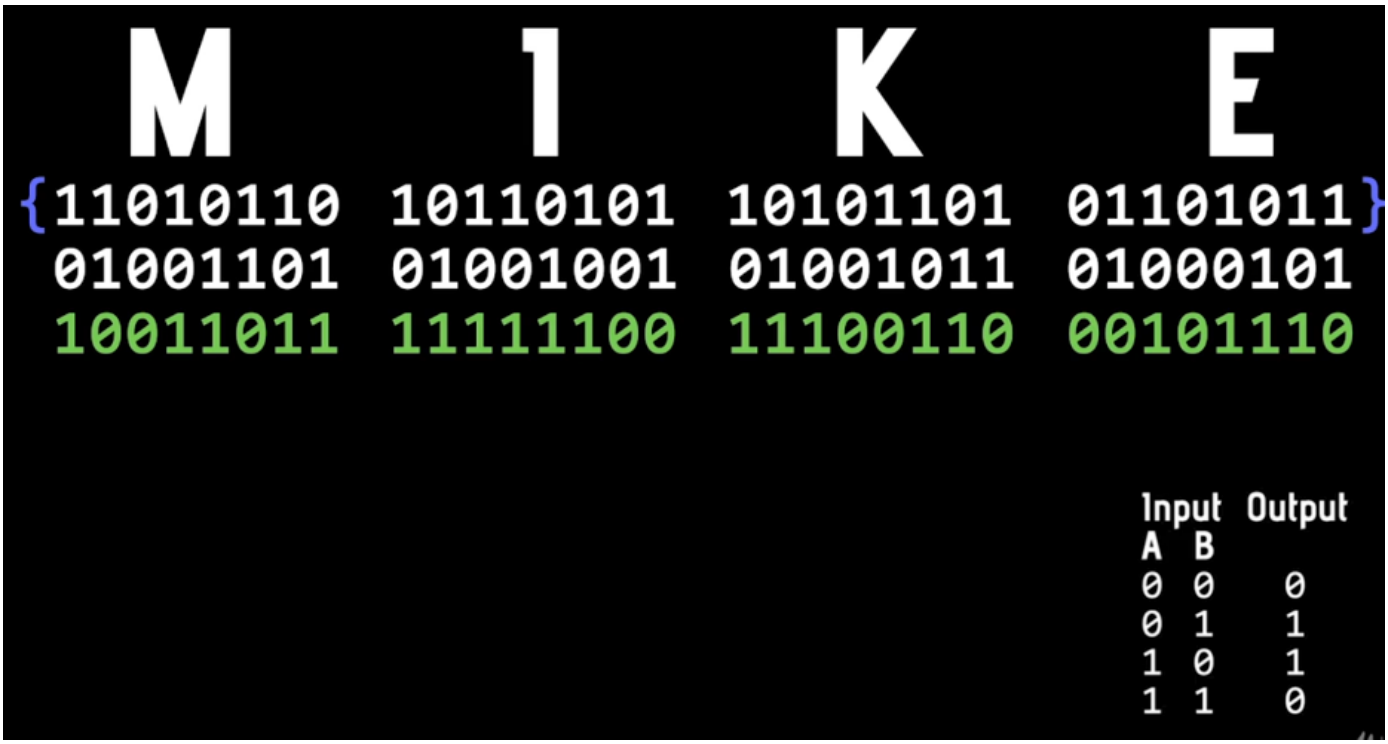
0	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
4	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
13	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
14	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
15	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
25	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Classic cryptography components

1. Algorithm
2. Key for encryption

Vigenere works great for letters of the alphabet but it doesn't do well with photos, sql databaseses, etc.
So, HOW DO WE ENCRYPT BINARY?????? We need an algorithm for binary data.

Example:
"Exclusive OR, "XOR"



Kerckhoff's Principle: As long as you don't know what the key is to the encryption you can understand the algorithm completely.
*All the algorithms available are open source.

Where are we going to be encrypting or decrypting data?

Types:

1. Data at rest: ex. something stored on a drive
2. Data in transit: ex. VoIP or a text
3. Data in process: ex. Data sitting in RAM or CPU

Quick Review:

-Cryptography is the practice of disguising information in a way that looks random
-The caesar cipher is one of the earliest known and simplest ciphers
-The Vigenere cipher employs the Caesar cipher as on element of the encryption process

---

19.Cryptographic Methods

Symmetric Encryption: Taking something and using a key to encrypt something, then using the exact same key to decrypt that something

The Key is called a **"Session key"**
In-Band: Sending the key with the encrypted data
Out-of-band: Sending the key NOT with the encrypted data

**Symmetric encryption is the primary way we encrypt data.

Ephemeral Key
-Temporary
-Provices perfect forward secrecy

Asymmetrick Encryption
- Uses a key pair
1. Public key (Given to anyone), (only used to encrypt)
2. Private key (Kept by the person), (only used to decrypt)

```
- The problem with asymmetric encryption is slow and tedious.
- Asymmetric encryption is used to send a secure session key (once the
session key is traded we go back to symmetric encryption)
```

CrypoSystem is highly defined process that programmers can do that actually makes cryptography work in the IT world.

Quick Review:

- Ephemeral keys provide perfect forwrd secrecy due to the temporary nature of the key
- Asymmetric encryption is slow, but very useful in exchanging session keys
- Cryptosystems define key properties, communication requirements for the key exchange and the actions taken through encryption and decryption process.

---

20. Symmetric Cryptosystems

when encrypting you have to develop algorithms

The first generation used a symmetric key algorithm. (Same key for encrypting and decrypting)

What is a Block?

*A symmetric block of data is when you have some data and its separated into blocks. ex. The first block is encrypted and sent. Then the next. etc.

**(DES) Data Encryption Standard **: The first type of Symmetric Block encryption
1. Invented by IBM Primarily
2. First open standard

Over simplified Steps of DES:

1. Grab a 64 bit chunk of plain text from the data stream we wish to encrypt
2. Perform an Initial Permutation (Very Specific stirring of the data)
3. Drop the last 8 bits off the key
4. split the key into two 28 bit chunks
5. Grab the first 24 bits from each half
6. Put both 24 bit segments together to create a 48 bit "SubKey"
7. Perform a Feistel Function on the data
    1. Take the 64 bits of data and split it into two 32 bit halves
    2. Set one half to the side
    3. Expand the working half into a 48 bit chuck using an expansion function

4. Apply an XOR function using the subkey
5. Use S-boxes (Take in 64 bits and output 4 bits) - There are 8 different s-boxes and each one gives a different four bit output
6. Apply the 8 s-boxes of data to create a 32 bit output
7. Then a final permutation is done
8. Then the two 32 bit chucks are put back together but backwards
9. Repeat

Issues with DES:
1. Short Key (Allowed it to be exposed to different types of attacks)

Alernatives to DES:

1. Blowfish
2. Triple DES

When talking about symmetric block encryptions we are talking about three things more than anything:

1. Key Size
2. Number of Rounds in encryption process
3. Block size

Comparisions of DES, BlowFish, and Triple DES:

1. DES
   a. Block Cipher
   b. 64-bit Block Size
   c. 16 Rounds
   d. Key Size: 56 bit
2. Triple DES
   a. Block Cipher
   b. 64-bit Block size
   c. 16 Rounds
   d. Key Size: (Repeated 3 Times) (56x3) = 168 bit key
3. BlowFish
   a. 64-bit Block Size
   b. 16 rounds
   c. Key Size: (Variable) as low as 32 bit - to as high as 448 bits

-In the late 1990's there was a competition (conference) to create a new symmetric encryption type.
-An Algorithm called Raindoll was adopted and it turned into what we call (AES) Advanced Encryption Standard in the early 2000s
-It is in essence unhackable

4. AES
    a. Block Cipher
    b. 128-bit Block size
    c. Key sizes: 128, 192, or 256 bits
    d. Rounds: 10, 12, 14 depending on the key size

Streaming Ciphers: As each bit comes out it is psuedo randomly encrypted (one bit at a time)

Example: RC4

5. RC4
    a. Streaming Cipher
    b. 1 bit at a time
    c. 1 round
    d. key Size: 40-2048 bits

Have a rough understanding of these types of encryptions:
-They are symmetric
-They are Block ciphers with the exception of RC4

Quick Review:

-Triple DES (3DES) is the DES key size tripled
-AES is a US Government encryption standard supported by the NIST
-RC4 (Rivest Cipher 4) is a stream Cipher

---

21. Symmetric Block Modes

** The problem with Symmetric Block Encryption: With any form of Symmetric block if you are applying the same key and same block size, each time you apply the key it will give you the same output (aka. Patterns can appear). This is called (Electronic Code Book Mode (ECB))*

To fix this we DONT use ECB mode, instead we use BLOCK MODES to obvuscate the data better.

BLOCK MODES: (The basic function of all these block modes is to encrypt something, and then use that encryption to encrypt the next one. Like a chain)

```
1. Cipher Block Chaining (CBC)
        1. Adds an initialization vector that is the same size as each
block. Before we encrypt we are going to take the first block and do an XOR
against the initialization vector.
        2. Then Encrypt it the first block
        3. Then we keep another copy of the encrypted first block (This is
all Binary by the way) and the copy
        4. An XOR is used against the copy for the next block.

2. Cipher Feedback (CFB)
        1. Adds an initialization vector and encrypts it
```

```
        2. Then take the output of the encryption and XOR to the first Block
        3. Then we keep another copy of the encrypted first block (This is
all Binary by the way) and the copy
        4. An XOR is used against the copy for the next block.

3. Output Feedback
        1. Adds an intialization vector and encrypts it.
        2. Then take the output of the encryption and XOR to the first Block
        3. Keep using the same initialization vector for the next block, etc

4. Counter (CTR)
        1. NONCE Value is added, Plus a counter value that increments in
binary.
        2. Add the Nonce to the counter value and encrypt it
        3. Then take the first block of plain text and XOR with the added
nonce and counter value to create the First block of cipher text
        4. Then the cycle is repeated with a different counter value etc


Quick Review:

- For the exam remember that absolutely no one uses ECB anymore.
- ECB block modes will always output the same results with the same input
- A binary block is plain text converted into 16-bit, 64-bit, or 128-bit
binary ciphertext
- CBC, CFC, OFB, CTR Block modes use an initialization Vector, which ensures
the output block is uniquely different.
```

---

22. RSA CryptoSystems

Asymmetric encryption always consists of a key pair.
1. Public Key (I pass this out so that someone can encrypt info)
2. Private Key (I have the private key and I am the only one who can decrypt the info from my public key with my private key)

RSA (RIVEST, SHAMIR, AND ADLEMAN)
-An asymmetric algorithm

Prime Numbers - Numbers that are only divisible by themselves. example: 11
Semi Prime Numbers - Two prime numbers Multiplied by each other. example: 11x7 = 187

So, how do we factor 187? Okay, we'll how fast can we factor a larger semi prime number? example: 100,160,063
In RSA we start off with massive values that it takes FOREVER to factor them.
The process of creating that massive semi prime number is what it used to generate our key pairs

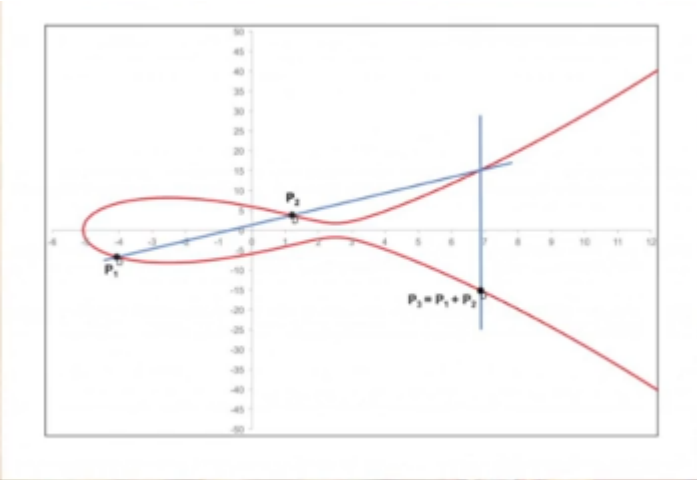
How an RSA Key Exchange happens.

```
1. Two parties generate their own key pairs and exchange them
                *** Keep in mind that if a third party grabs the public keys
it doesn't matter because she can only encrypt something and send it to one
of the other parties. well, this is a problem because what if the third
party pretends to be one of the primary 2 people exchanging information? RSA
includes Authenication protocols (e.g. Digital signatures etc.)
```

The NEWEST Methodology for asymmetric algorithms is called ECC (Elliptic Curve Cryptography):
- It can provide us very small keys that we can carry around that have the robustness of a large RSA key.
- Also Generating the keys and going through the process can be faster than RSA
- Based on an Elliptic curve formula:

$$y^2=x^3+ax+b$$

When it's plotted on a graph it will look like this:



So, you can plot a point on the curve and generate a key pair based on the plotted point.

Quick Review:

- Public Keys are paired with a private key (ie key pair) when using RSA Asymmetric Cryptosystems
- ECC can create a smaller key than RSA provides the same security with increased performance
- Each public key has a single private key, without the private key the information can not be decrypted

---

23.Diffe-Hellman

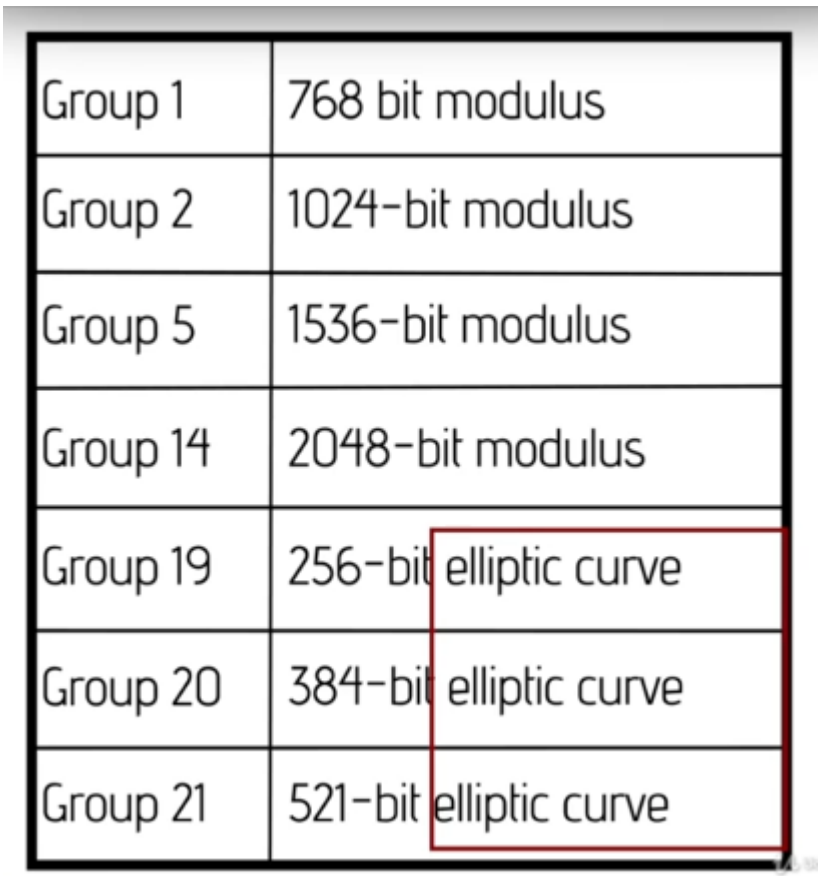
-Diffe-Hellman is an Asymmetric algorithm
-Only provides a methodology for both parties to come up with the same Session key
-Key exchange Protocol/Key agreement protocol

An Analogy using color:

-Two parties are trying to talk and they both need to generate a specific color.
(You can get a specific color out of mixing two colors of paint but it is extremely difficult to get the original paint colors out of the mixture.)
-The two parties have two different colors and they mix those colors together to derive a special color from them separately
-Then they send those colors to each other (exhange)
-Then separately they mix those colors together with their own private color to create another color and come up with the same color/Value

They do symmetric Encryption with the specific color/value they've created.

Diffie-Hellman Groups we're created to avoid crackable values. Elliptic Curve diffie-Hellman is now becoming popular because it is very tough to crack.



Group 1	768 bit modulus
Group 2	1024-bit modulus
Group 5	1536-bit modulus
Group 14	2048-bit modulus
Group 19	256-bit elliptic curve
Group 20	384-bit elliptic curve
Group 21	521-bit elliptic curve

Quick Review:

- Diffie-Hellman is an asymmetric algorithm often referred to as a key exchange agreement
- Diffie-Hellman groups help by defining the size or type of key structure to use
- Diffie-Hellman can have very large keys

---

24. PGP/GPG

PGP (Pretty good Privacy)
1. Invented by phil Zimmerman in 1991 (There was no encryption back then)
2. PGP was originally invented for email encryption
3. PGP Evolution: Today you can use it for lots of different things
4. HOW IT WORKS

1. Random key is generated by the encryptor
2. Encrypt the data using the key
3. then encrypt the key using the receivers public key
4. Decryption works the opposite direction and decrypt with the private key (Public Key, Private Key, and a Random Key) PGP works similar to using asymmetric encryption to echange an session key and then switch to symmetric encryption.

Public Key Infrastucture (PKI):

- Certificate Authority -> Intermediate Authority -> Individual Users (The way the internet works)

PKI/Web of Trust (was Failure):

- 1 Certificate that trusts another certificate and then that certificate trusts another certificate and you end up with a web of trust.
- Phil Zimmerman was the first to push Web of Trust first

PGP has changed a lot. Today there are three big players in PGP:

PGP Certificates:

1. symantec Corporation
    1. Only encrypts mass storage
        1. Signing and Disk encryption
        2. Works with Bitlocker
        3. Works with filevault
        4. Enterprise cloud solutions
        5. Not free

2. Open PGP
    1. Free
    2. Encrypted Email
    3. PKI Support
    4. S/MIME
    5. Works as a plugin in your current Email. (i.e. Outlook, etc)

    Encrypted Email:
    Proton Mail: Fully Encrypted (Uses open PGP)

3. GPG (GNU Privacy Guard) - Based on Open PGP
    1. Free Toolset
    2. File and Disk Encryption

Quick Review:

-PGP was originally used for email encryption
-PGP encrypts a message witht he public key; the message is decrypted with the private key
-PGP is based on open PGP

---

24. Hashing

-Hashing provides Integrity
-A Hash is a Fixed Size
-Hashes are one way (it is impossible to figure out the original value)
-Hashes are deterministic

Types of Hashes on SEC+

```
1. Message Digest 5 (MD5)
        1. Invented 1992 by Ron Rivest
        2. Uses 128 bit hash

2. SHA
        1. Developed by NIS
        2. SHA-1 (Earliest SHA)
                a. 160 bit Hash

MD5 and SHA-1 have the ability to create a collision. A collision means that you take two different types of data and generate the same Hash.
        -If you can force a hash to make occassional collisions you can figure out how the hash works.

So, MD5 and SHA-1 are not used today because of the vulnerability of collisions

3. SHA-2 (Different Types Based on the length of the Hash 'SHA-256 or SHA-512')

4. RIPEMD (RACE Integrity Primitives Evaluation Message Digest)
        1. Uncommon
        2. Open Standard
        3. BE AWARE FOR THE TEST THAT IT Comes in 128, 160, 256, 320 bit Versions
```

Memorize the Names and Sizes of the Hashes

Hashes are used everywhere.

Quick Review:

-Hashes are involved with password storae and encryption
-Hashes are one way, deterministic, and will produce the same results each time the source is hashed

-It doesn't matter how long the source data is, the hash will be the same exact size

---

25. HMAC

HMAC (Hash-Based Message Authentication Code)

Two computers that have gone through an encryption process and they are sharing the same key. A bad guy could get in the middle and mess it up. How do we know that we are still communiticating with the same person. This is where HMAC comes into play.

HMAC (Simplified) takes one individual packet and adds the key and makes a hash out of it. Great way to have more confidence that the information is coming from the right person.

www.feeformatter.com *HMAC Generator*

Quick Review:

-HMAC provies message integrity
-HMAC requires each side of the conversation to have the same key
-It is based on standard hashes (MD5, SHA-1, Etc)

27.Steganography

The process of taking data and hiding it in other data.

The number one way we do this is we take data and hide it within graphic images (Hiding things in the raw image text e.g. png files)

Tools to do this: "Image Steganography"

Quick Review:

-Steganography hides data within data
-Commonly used with graphic images
-Hidden data may or may not be encrypted

VERY COOL

---

28. Certificates of Trust

The problem with asymmetric encryption is the key exchange. How do you know the key actually belongs to the person or site you are visiting?

The way around this:

Digital Signature:
- When a website sends its public key, it also encrypts its webpage with its private key and sends a hash of that encryption with the public key.
With your public key you can encrypt the webpage and generate a hash and compare your hash with the hash that was sent to you to authenticate.

`- you get a 3rd party to generate their digital Sigature also`

Digital Certificate:
- My Public Key, My Digital Signature, and The Trusted 3rd Party Signature.

Three ways to do trust

```
1. Unsigned Certificate: Generate a certificate on your own     (No 3rd party voucher)
2. Web of Trust: Requires lots of maintenance (Never really taken off)
3. PKI (Public Key Infrastructure): Heirachial Method. (The way we do the internet)
        1. Certificate Authority (CA) ---> Intermediate Certificate Authorities ---> Users
```

Quick Review:

-Certificates include a public key and at least one digital signature
-Web of trust uses a web of mutually trusting peers
-Public Key infrastructure uses a hierarchial structure with root servers

---

29. Public Key Infrastructure (PKI)

1. Starts with "Certificate Authority (CA)"" (They have an unsigned root certificate)
2. Root Certificate Systems designate intermediary certificate authorities
3. intermediary certificate aurthorities give Certificates to the little guys (users)
4. These create a certification path

PKI is not a standard. However, there is PKCS.

PKCS has become the standard for PKI standard.

PKI is based on an old standard which is "X.509"

Certificates don't have a standardized format, but we do have the PKCS numbers.

PKCS-7: a way to store certficates as individual files
PKCS-12: a way to store the certificate and the private keys as a package

Read about thumb prints in certificates*

CRL (Certificate Revocation List) used as a check for integrity of certificates. Can take up to 24 hours to respond to a bad certificate (Can be slow).

OCSP (Online Certificate Status Protocl): Works similarly to CRL but pretty much works real time.

CRL is phasing out while OCSP is become the go-to

Quick Review:

What to know
- Know why we have certificates
- Know what a certificate looks like

- Know what a certification path is
- Things that are in the certificate
- Know PKCS

-PKCS-7: is a way to store certificates as individual files
-PKCS-12: is a way to store cerificates and private keys as a package
-OCSP (Online certification status protocol) is a more modern version of CRL (certification revocation list).

---

30. Cryptographic Attacks

- When you have a web server (Any Kind) you are going to have to have a list of usernames and passwords on that system server somewhere. These passwords are not stored in plain text. They are hashed.
- Password attacks are typically trying to hack hashes.
  1. You have to get to the Username and password lists first
  2. If the password is stored in a hash there is no way to reverse the hash. So, *Hashing attacks, are comparitive attacks, generated hashes vs. stored hashes.

  We are generating hashes and making a comparision. When we compare the right ones, then we can say we have the password.
- Generating hashes until we find the right one is called a *Brute Force Attack*
- Old brute force program called Cain and abel is used in the example. Hashcat is more modern.
- The longer the password the more difficult it is to crack with a brute force password.

COMPLEX PASSWORDS ARE USED TO MAKE CRYPTOGRAPHIC ATTACKS HARDER.

Dictionary Attack: Using a text file that contains dictionary words and mixes them up to create random passwords. You have to feed the tool a text file.

Rainbow-table attack: Pre-generated hashes in it and compares to all the pregenerated hashes. (These are massive) You can buy Rainbow table Hard drives with massive Rainbow tables in them.

SALT: (arbitrary value)

```
1. Example: Salt Value of 4
1. Password: TIMMY
2. Concatenate our salt value with the password: TIMMY4526
3. Then its hashed.

Salted Hash tables are VERY difficult to crack.
```

Key Stretching: Passphrase is taken with other values and generates a key that is complicated and difficult to crack.

Two types of Key Stretching for Wireless

1. Keystretching technique PBKDF2 Algorithm

2. Keystretching technique bcrypt

Proper Key Stretching in todays world is nearly uncrackable today.

Quick review:

- Passwords are usually stored in hash format making cracking very difficult
- More complex passwords make hacking much harder
- Salted passwords are a lot harder to crack

QUIZ 2

Question 1:

Which part of the cryptography method is publicly available information?

- ○ The keys
- ◉ The algorithm
- ○ None of the information is public knowledge
- ○ Only the public key pair

Question 2:

Which cipher did Mike refer to as his secret decoder ring?

- ◉ Caesar cipher
- ○ Vigenère cipher
- ○ Truth table
- ○ Binary XOR encryption

Question 3:

What are the two different types of cryptographic methods used for encryption?

- ○ Symmetric and hashing
- ◉ Symmetric and asymmetric
- ○ Blocking and streaming
- ○ Algorithms and keys

Question 4:

Which symmetric block encryption has a variable key size between 32-448 bits?

- ○ DES
- ◉ Blowfish
- ○ Triple DES
- ○ AES

Question 5:

Which of the following block modes are the most predictable because identical piece of plaintext will produce the same corresponding ciphertext?

- CTR
- OFB
- (•) ECB
- CBC
- CFC

Question 6:

RSA is an example of what type of encryption?

- Symmetric
- (•) Asymmetric
- Diffie-Hellman
- Block mode only

Question 7:

When using Diffie-Hellman key exchange, what is the one of the benefits?

- [x] Low overhead method where two parties need the same session key
- [ ] The key exchange has four keys, making it more secure
- [ ] The method is used in all asymmetric key exchanges
- [ ] The benefit is only to the 2nd party, who only needs to know a color

Question 8:

PGP has been used by what service?

- [ ] To generate digital signatures for Web pages
- [ ] PGP Corp is a certificate company
- [ ] Anti-malware
- [x] Mail encryption

Question 9:

Which of these hash algorithms is the oldest?

- [ ] SHA-1
- [ ] SHA-256
- [ ] SHA-512
- [x] MD5

Question 10:

What is the process of hiding one set of data within other data?

- Cryptography
- Stenography
- Oceanography
- Steganography

Question 11:

A digital signature is a:

- Certificate
- Hash
- Public key
- Private key

Question 12:

Which of the following is being replaced by Online Certificate Status Protocol (OCSP)?

- CRL
- PK1
- PKCS
- x.509