Security Precautions

Learning Intentions

By the end of this lesson learners will be able to:

- ❑ Describe what is meant by **Encryption**
- ❑ Discuss the purpose of **Digital Certificates** and **Digital Signatures**
- ❑ Describe the purpose of server side validation of form data

Encryption

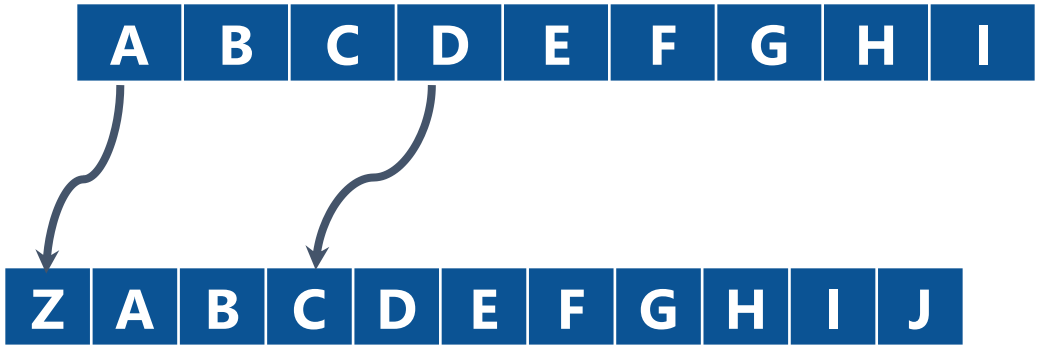
Encryption is when data is encoded into another form.

This means that even if data is intercepted then the data is meaningless until it is deciphered using a **key**.

How encryption works

Lets look at a basic cypher

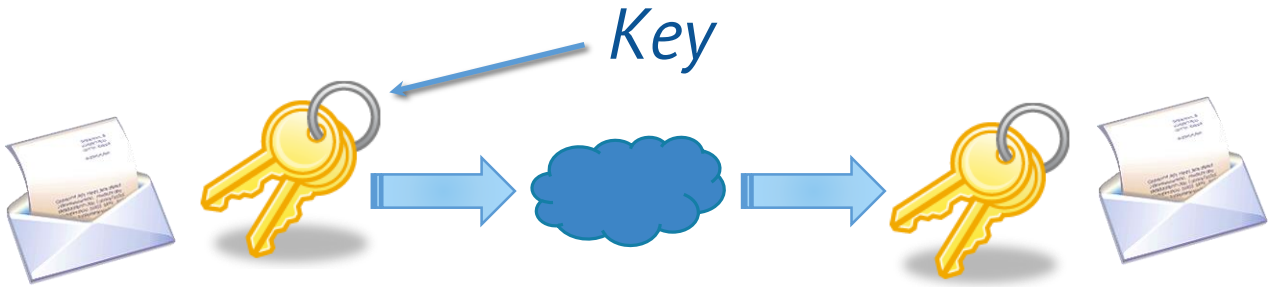
A Caesar(shift) cypher



So **DAD** would be **CZC**

The Key

The **key** is the secret to encryption



Encryption

There are two main methods to encrypt data
Symmetric Key - Secret Key
Asymmetric Key - Private key/public key

Symmetric Encryption

Data Encryption Standard (DES) was created in 1977

Used a 56 bit key

Approx. 72,000,000,000,000,000 combinations (2^{56})

Advanced Encryption Standard (AES)

Used by the US government

Offers 128,192 or 256 bit encryption

2^{256} combinations (1.15e+77)

RSA (Ron Rivest, Adi Shamir and Leonard Adleman)

1024 - 2048 bit

$2^{1,024} =$

179,769,313,486,231,590,772,931,...,304,835,356,329,624,224,137,216 (309 digits)

$2^{2,048} =$

32,317,006,071,311,007,300,714,...,193,555,853,611,059,596,230,656 (617 digits)

Encryption and Digital Rights Management

CSS (Content Scramble System) encryption was used as the original **D**igital **R**ights **M**anagement system on **DVD's** in 1996

- Used a **40 bit** cipher
- Compromised in 1999

Pros and Cons – Symmetric Encryption

For

- ✓ The key doesn't have to be sent with the message
- ✓ The system is usually more straight forward/faster

Against

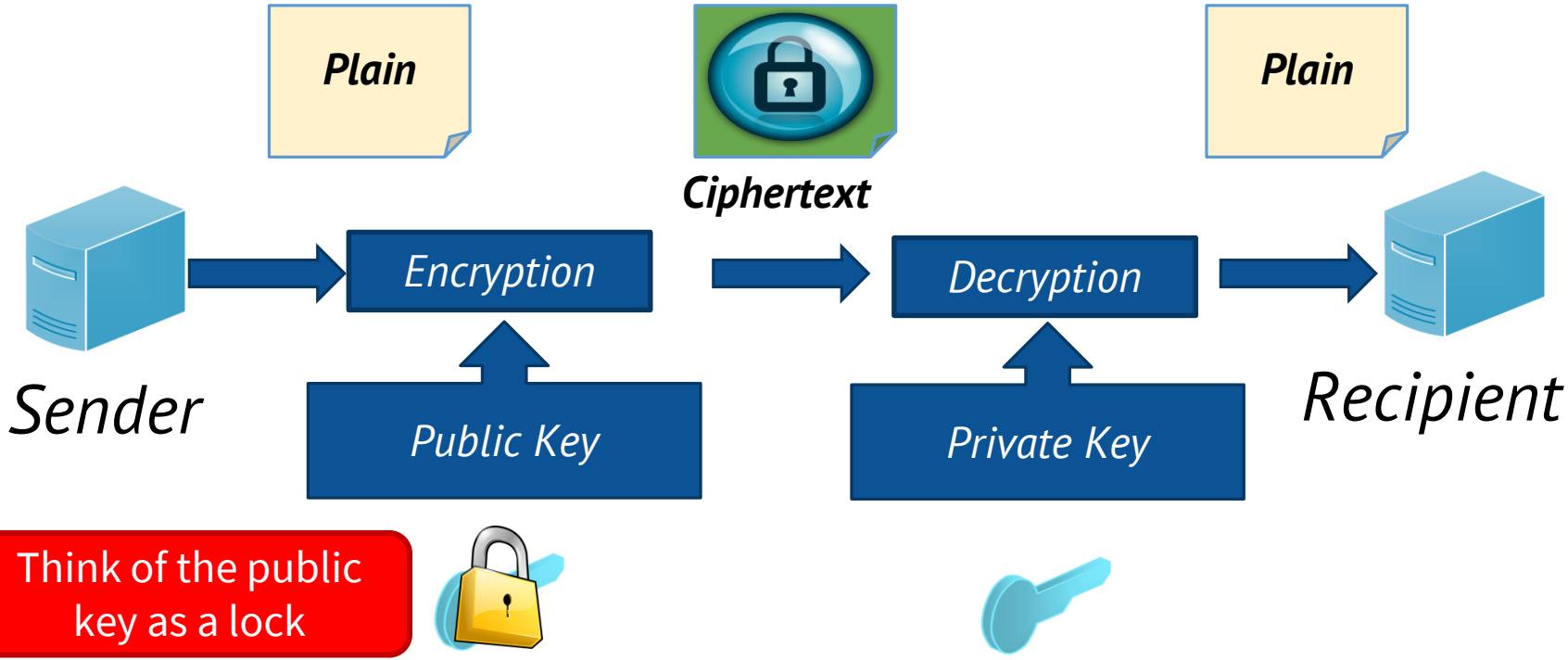
- ✗ The key has to be installed with the receiver before transmission
- ✗ If the key is compromised both sent AND received messages can be decyphered

Asymmetric Key

Asymmetric Key or public key encryption is when there are **two keys**.
A **public key** is made freely available to anyone who might want to **send** you a message.

A **second**, **private key** is kept secret and used to decrypt received mesages

Asymmetric Encryption



Think of the public key as a lock

Which key and when?

The **sender** uses the **public key** to:
Encrypt any **messages to recipient**

The **recipient** uses the **private key** to:
Decrypt any **messages from sender**

Pros and Cons – Asymmetric Encryption

For

- ✓ The Private key never needs to be distributed
- ✓ Can be used to implement digital signatures

Against

- ✗ Is slower than symmetric encryption.
- ✗ It requires far more processing power to encrypt and decrypt the content of the message.

How do you get the keys?

To use **asymmetric encryption** then there has to be a way to distribute the public keys.

And to **verify the authenticity of the sender**

Digital Certificates are a useful tool for this.

Public Key Encryption & Malware

Some **malware** (ransomware) such as **CryptoLocker** will encrypt the contents of infected machines

It uses RSA encryption

Will only decrypt the drive once payment has been made

The key is held on a private server

Further reading:

https://traxarmstrong.com/free-tool-remove-cryptorbit-ransomware/#gsc.tab=0

Basic RSA Encryption Example

To encrypt a message such as HELLO, we will first turn it into numbers

H	E	L	L	O
8	5	12	12	15

Our public key is 33,3

This consists of two prime numbers

RSA Encryption Example

Our public key is 33,3

So we raise the number to the power of **3**

Original	H	E	L	L	O
Original	8	5	12	12	15
^3	512	125	1,728	1,728	3,375

RSA Encryption Example

Our public key is 33,3

Then we will perform our number mod 33 (the remainder after dividing by 33)

Original	H	E	L	L	O
Original	8	5	12	12	15
^3	512	125	1,728	1,728	3,375
%33	17	26	12	12	9

This is our ciphertext

RSA Decryption Example

Our private key is 33,7

We will raise our number to the power of 7

Original	H	E	L	L	O
Original	8	5	12	12	15
Cipher	17	26	12	12	9
^7	410338673	8031810176	35831808	35831808	4782969

RSA Decryption Example

Our private key is 33,7

Then we will perform our number mod 33 (the remainder after dividing by 11)

Original	H	E	L	L	O
Original	8	5	12	12	15
Cipher	8	26	12	12	9
^7	410338673	8031810176	35831808	35831808	4782969
%33	8	5	12	12	15

WannaCry (2017) Ransomware Sample

How strong is the encryption?

The **Key size** is the size in bits of the key used in the algorithm that encrypts the data

Assume that a 40 bit key has been cracked

May have been brute forced

Then an example 128 bit key may only actually provide 88 bits of encryption

Cracking Encryption

512 bit keys were broken in **1999**

Encryption has even been compromised by listening to the sound the CPU makes.

It was conducted with one of the co-inventors of the RSA algorithm

Further reading:

https://www.extremetech.com/extreme/173108-researchers-crack-the-worlds-toughest-encryption-by-listening-to-the-tiny-sounds-made-by-your-computers-cpu

Unbreakable codes?

There is a lot of research going on into **Quantum cryptography**

- Relies on physics and not maths
- Data is transmitted using light

The main theory behind this is that when the light is intercepted then it is changed

- This uses Heisenberg's uncertainty principle

Identifying yourself

In real life if you are proving your identity you can use multiple forms of ID

- Passport
- Drivers License

How can prove the identity of senders of emails/providers of websites?

Proving your identity

A **digital certificate** is the digital version of a passport or driving license
They are issued by a **central certification authority**
Many digital certificates conform to the X.509 standard.

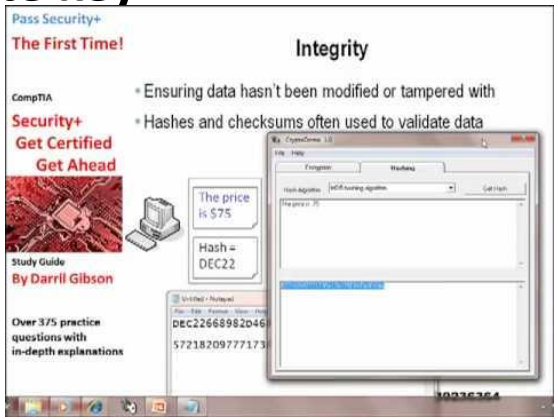
Can contain the following information

- ❏ Public Key
- ❏ Owners Name
- ❏ Expiration and Issuer

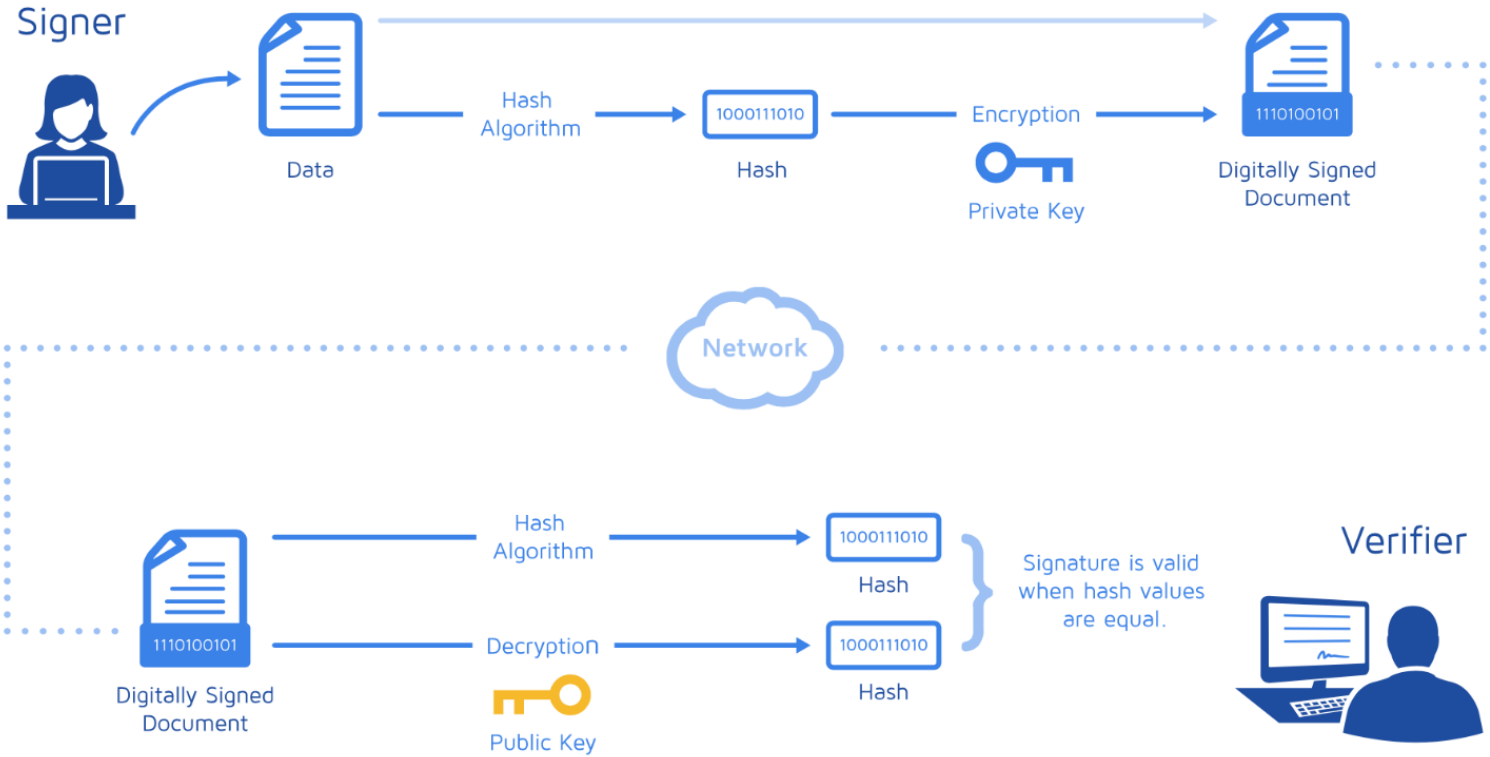
Digital Signatures

A **digital signature** is a method of ensuring that a message is authentic (unaltered)

1. You obtain a **message hash**
   A mathematical summary of the contents of the messag
2. You can **encrypt** this **message hash** with your **private key**
3. This **'signature'** is attached to a message



Digital Signatures



https://www.docusign.co.uk/how-it-works/electronic-signature/digital-signature/digital-signature-faq

How long to crack a digital certificate?

https://www.digicert.com/TimeTravel/

Hashing

In Feb 2017 - Google broke the SHA1 hashing system

Further reading: https://security.googleblog.com/2017/02/announcing-first-sha1-collision.html

The computations involved in this was::

- Nine quintillion (9,223,372,036,854,775,808) SHA1 computations in total
- 6,500 years of CPU computation to complete the attack first phase
- 110 years of GPU computation to complete the second phase

Moving on from SHA1

MD5 (1992) produces 128 bit outputs

SHA-0 (1993) produces 160 bit outputs

SHA-1 (1995) produces 160 bit outputs

SHA-2 (2001) produces 224-512 bit outputs

Further reading:

https://en.wikipedia.org/wiki/Secure_Hash_Algorithms

http://borjournals.com/a/index.php/jecas/article/viewFile/1807/1127

At the other side

1. The recipient has to apply the same mathematical hash of the received message
2. The encrypted message hash is then decrypted
3. If both of the hashes match then the message is valid and authentic

MD5 Hash Generator:

http://www.md5.cz