CT255
Introduction to Cybersecurity

Module Information

Semester 1

Dr. Michael Schukat, 2019-22

About me

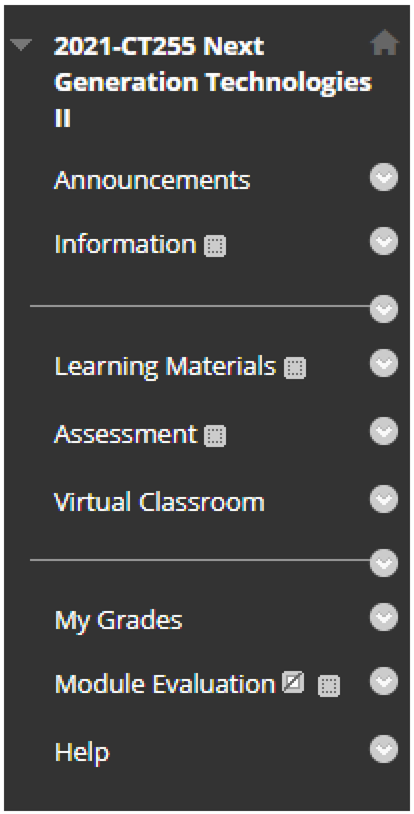
- Background:
  - M.Sc. Computer Science
  - PhD (Computer Science)
  - Since 2002: Senior Lecturer in the School of Computer Science
  - 1999 - 2002: Senior Embedded Systems Design Engineer (Ireland)
  - 1997 - 1999: Embedded Systems Design Engineer (Germany)
  - 1994 - 1997: Junior Lecturer and Researcher (Germany)

- Research Interests:
  - Many, including cybersecurity
- Contact:
  - michael.schukat@nuigalway.ie
  - Office IT402

Module Overview

- CT255 consists of 2 parts, each worth 50%
  - Semester 1: Cybersecurity (me)
  - Semester 2: Game Programming (Dr. Sam Redfern)
- 2 hours exam paper next April / May
  - Previous exam papers available from NUIG library database
    - **Cybersecurity is a new S1 topic since 2020/21**

Blackboard Learning Materials



- Lecture code **2223-CT255**
  - If you encounter any problems, contact me via email
- Announcements
- Information
- Learning Materials
- Assessment
- Virtual Classroom
- Discussion Forum

NUI Galway
OÉ Gaillimh

Lecture Organisation

- Where possible we'll apply the concept of **flipped learning**:
  - You'll study a set of material prior to the weekly lectures, circulated via Blackboard
  - If you have specific questions about content, please let me know by the **Friday before the lecture**, so that I can incorporate them into my lecture slot the following day (Monday)
  - On occasions you'll complete a marked Blackboard quiz

NUI Galway
OÉ Gaillimh

Weekly Classroom Activities

- **Please have a charged mobile device with you to access Blackboard or other learning tools, e.g.**
  - Qwickly to record attendance
- Lectures will incorporate further examples, case studies and other activities
  - Small group activities
  - Interactive discussions (using Menti)
  - Post-lecture reflective journal submissions

Labs

- Labs starting in week 3 (Wed 14:00-16:00), **to be confirmed**
  - F2F labs in IT101
  - 2 groups, therefore 1 hour of lab time per student and week
  - Lab attendance is not compulsory, but recommended

Breakdown of Marks

- See Blackboard *Information* section
  - Breakdown of CA may change slightly

NUI Galway
OÉ Gaillimh

Part 2:

MENTI

What is Cybersecurity?

- Cybersecurity is the practice of protecting systems, networks, and programs from digital attacks. These cyberattacks are usually aimed at accessing, changing, or destroying sensitive information; extorting money from users; or interrupting normal business processes
  Source: Cisco

Some interesting Videos

Hacker Shows Off Lethal Attack By Controlling Wireless Medical Device

BY JORDAN ROBERTSON | FEB. 29, 2012 10:00 AM EDT | POSTED IN HACKERS, MEDICAL PRIVACY, POSTS, SECURITY, VIDEO | 15 COMMENTS

Recommend 366 Tweet 250 Share 89 +1 74 Email Print

Photographer: David Paul Morris/Bloomberg

Barnaby Jack uses a mannequin equipped with an insulin pump to show the vulnerabilities of wireless medical devices.

- https://www.youtube.com/watch?v=D2mxKEa2xmA
- https://www.youtube.com/watch?v=THpcAd2nWJ8
- https://www.youtube.com/watch?v=YJ8PZeRwweA

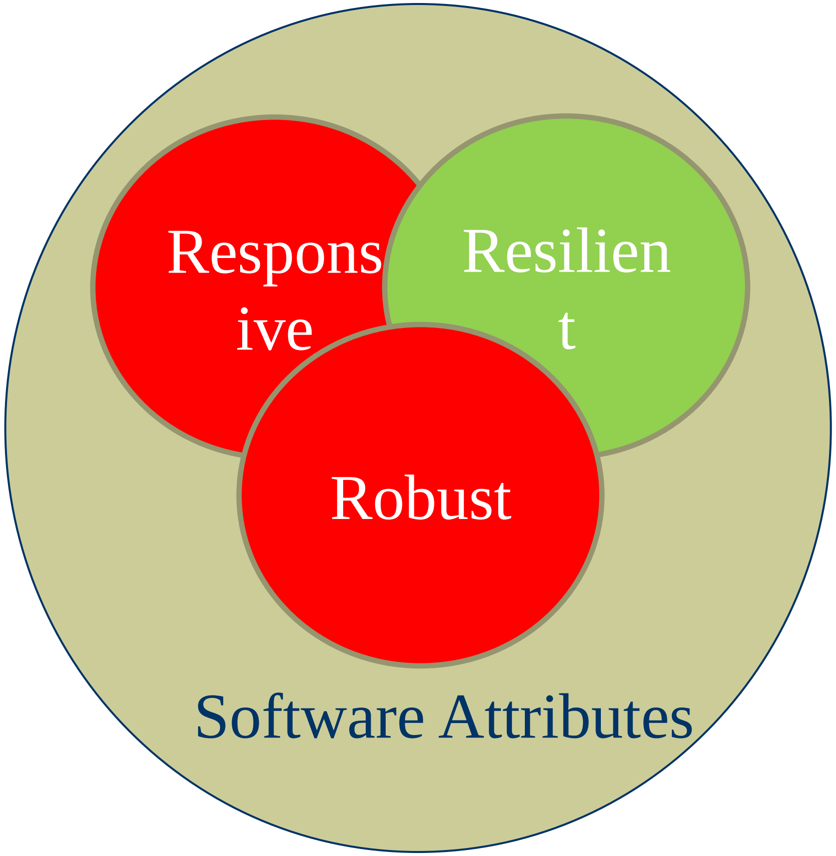
S1 Main Learning Outcomes

- To provide you with a solid understanding and the practical application of:
  - GDPR
  - Social Engineering techniques
  - Cybersecurity principals and concepts, including
    - Private and public key encryption
    - Data authentication
    - Passwords and password cracking

NUI Galway
OÉ Gaillimh

Cybersecurity Roadmap

- CT255 – Semester 1
  - Introduction to Cybersecurity
- CT417 – Year 4
  - Secure and resilient software
- CT420 – Year 4
  - Real-time systems, mission critical and robust software
- CT437 – Year 4
  - Advanced Cybersecurity



NUI Galway
OÉ Gaillimh

The next Steps

Week	Task 1	Task 2
Week 1	Introduction (now)	Study week 2 material (GDPR)
Week 2	GDPR case study and small groups activities	Study week 3 material

Please check my Blackboard posts!

NUI Galway
OÉ Gaillimh

CT255
Introduction to Cybersecurity

Lecture 1
GDPR

Dr. Michael Schukat, 2019-22

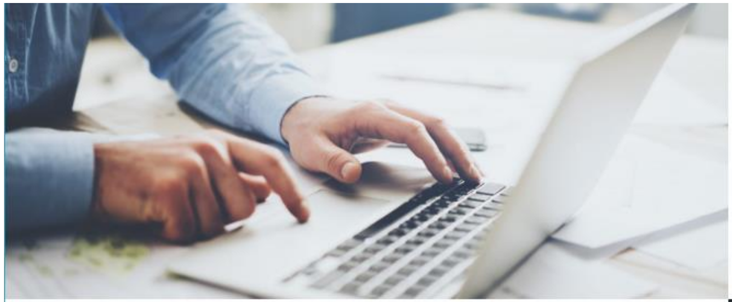
Motivation

- Cyberattacks are aimed at **accessing, changing, or destroying sensitive information**, extorting money, or interrupting normal business processes
- So managing sensitive data may reduce the attack probability or at least its impact
- GDPR provides such a regulatory framework

General Data Protection Regulation

- GDPR is a binding regulation in EU law on data protection in the EU and the European Economic Area (EEA), that became enforceable on 25 May 2018
- It also addresses the transfer of personal data outside the EU and EEA areas
- The GDPR's primary aim is to **enhance individuals' control and rights over their personal data and to simplify the regulatory environment for international business**
- The regulation **contains provisions and requirements related to the processing of personal data of individuals** who are located in the EEA, and applies to any enterprise—**regardless of its location and the data subjects' citizenship or residence**—that is processing the personal information of individuals inside the EEA

GDPR Summary: https://www.gdpreu.org/



After four years of preparation and debate the GDPR was finally approved by the EU Parliament on 14 April 2016. It was enforced on 25 May 2018 – and organisations that are not compliant could now face heavy fines.

This website is a resource to educate organisations about the main elements of the General Data Protection Regulation (GDPR) and help them become GDPR compliant. The guidance offered across this website will ensure that companies have effective data rights management strategies enforced.

The EU General Data Protection Regulation (GDPR) replaces the Data Protection Directive 95/46/EC and is designed to:

- Harmonize data privacy laws across Europe,
- Protect and empower all EU citizens data privacy
- Reshape the way organizations across the region approach data privacy.

GDPR reshapes the way in which sectors manage data, as well as redefines the roles for key leaders in businesses, from CIOs to CMOs. CIOs must ensure that they have watertight consent management processes in place, whilst CMOs require effective data rights management systems to ensure they don't lose their most valuable asset – data.

The key articles of the GDPR, as well as information on its business impact, can be found throughout this site.

NUI Galway
OÉ Gaillimh

What is Data Protection?

- Data protection is about an **individual's fundamental right for privacy**
- When an individual gives their personal data to any organisation, the recipient has the duty to keep the data safe and private
- Data protection legislation
  - governs the way we deal with personal data / information
  - provides a mechanism for safeguarding privacy rights of individuals in relation to the processing of their data
  - upholds rights and enforces obligations

Mx6

Personal Data

- Any information relating to an identified or identifiable natural person ('data subject’)
  - an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the **physical, physiological, genetic, mental, economic, cultural or social identity** of that natural person"
- Applies to printed and electronic data

Sensitive Personal Data

- Racial origin
- Political opinions
- Religious or philosophical beliefs
- Trade Union membership
- Genetic Data (e.g. biological samples)
- Biometric Data (e.g. fingerprints)
- Data concerning health
- Data concerning a person's sex life or sexual orientation
- **Explicit consent is required to process special categories of personal data**

HTTP Cookies

- An (HTTP) cookie is a small piece of data stored on the user's computer by the web browser while browsing a website
- Cookies were designed to be a reliable mechanism for websites to remember stateful information (such as items added in the shopping cart in an online store) or to record the user's browsing activity
- They can also be used to remember pieces of information that the user previously entered into form fields
- Authentication cookies are the most common method used by web servers to know whether the user is logged in or not, and which account they are logged in with

Cookie Implementation

- Cookies are arbitrary pieces of data (i.e. large random strings), usually chosen and first sent by the web server, and stored on the client computer by the web browser
- The browser then sends them back to the server with every request
- Browsers are required to:
  - support cookies as large as 4,096 bytes in size
  - support at least 50 cookies per domain (i.e. per website)
  - support at least 3,000 cookies in total

NUI Galway
OÉ Gaillimh

Setting a Cookie - Example

- A browser sends its first request for the homepage of www.example.org, resulting in the GET request

```
GET /index.html HTTP/1.1
Host: www.example.org
...
```

- The server responds with

```
HTTP/1.0 200 OK
Content-type: text/html
Set-Cookie: theme=light
Set-Cookie: sessionToken=abc123; Expires=Wed, 09 Jun 2021 10:18:14 GMT
...
```

- Later client requests to this server will contain these cookies:

```
GET /spec.html HTTP/1.1
Host: www.example.org
Cookie: theme=light; sessionToken=abc123
…
```

Cookie Structure

- A cookie consists of the following components:
  - Name
  - Value
  - Zero or more attributes (name/value pairs) Attributes store information such as the cookie's expiration, domain, and flags (such as *Secure* and *HttpOnly*)

```
HTTP/1.0 200 OK
Content-type: text/html
Set-Cookie: theme=light
Set-Cookie: sessionToken=abc123; Expires=Wed, 09 Jun 2021 10:18:14 GMT
...
```

Session Cookies

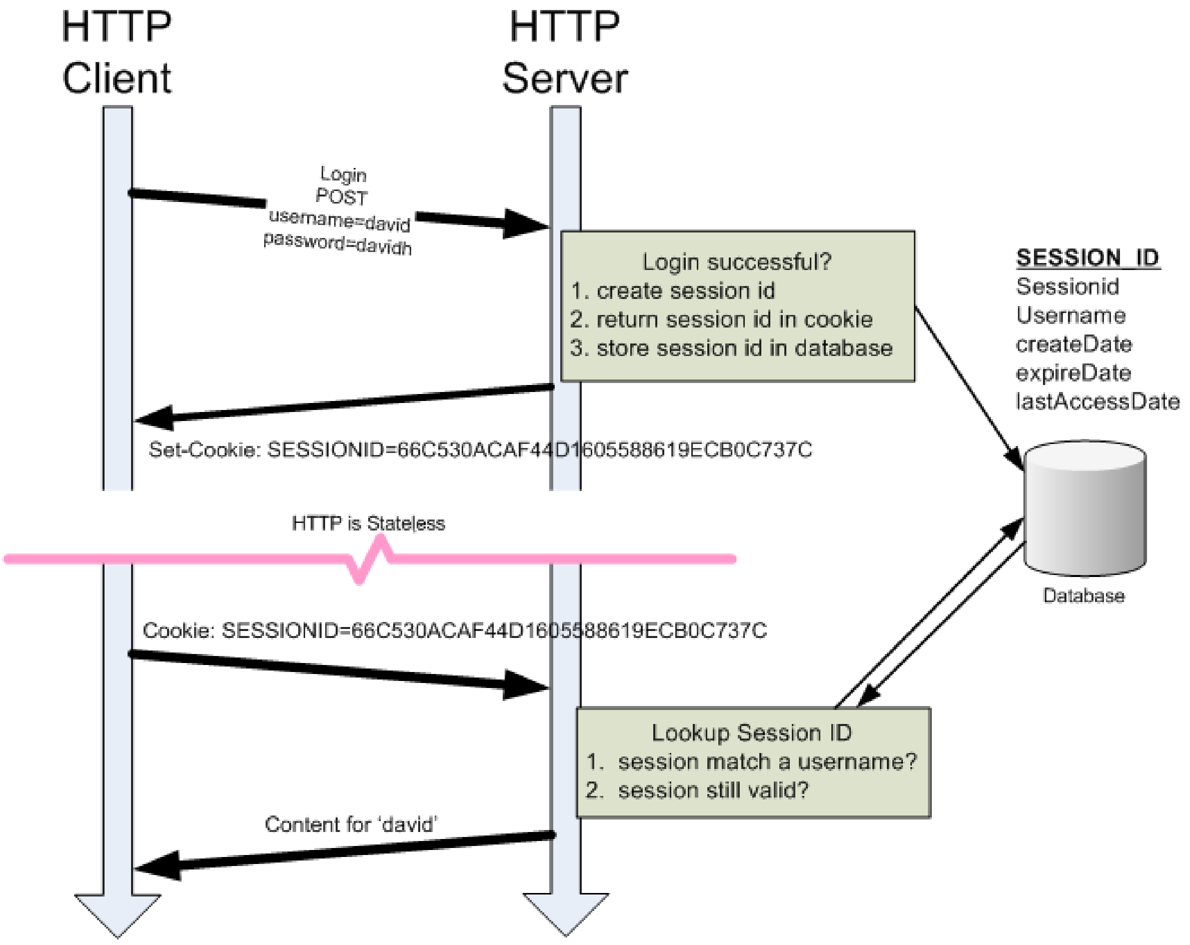
- A session cookie (aka in-memory cookie, transient cookie or non-persistent cookie) exists only in temporary memory while the user navigates its website
- Web browsers normally delete session cookies when the user closes the browser
- Session cookies do not have an expiration date assigned to them, which is how the browser knows to treat them as session cookies
- Example: “theme” cookie on previous slide

Persistent Cookie

- A persistent cookie expires at a specific date or after a specific length of time
- For the persistent cookie's lifespan set by its creator, its information will be transmitted to the server every time the user visits the website that it belongs to
- … or every time the user views a resource belonging to that website from another website (such as an advertisement).
  For this reason, persistent cookies are sometimes referred to as tracking cookies because they can be used by advertisers to record information about a user's web browsing habits
- However, they are mainly used for legitimate reasons, such as keeping users logged into their accounts on websites, to avoid re-entering login credentials at every visit
- Example: “sessionToken” cookie in the previous example

NUI Galway
OÉ Gaillimh

Session Management via Persistent Cookies



NUI Galway
OÉ Gaillimh

Cookie Attributes

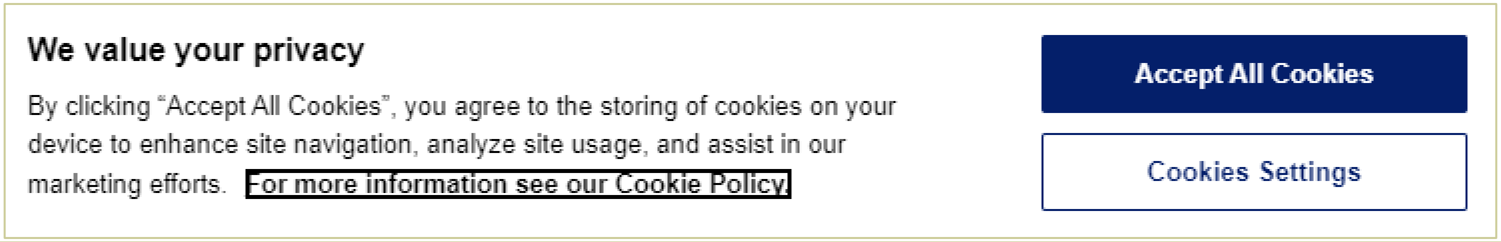
- Consider the following response header sent by a webserver that contains 3 persistent cookies:

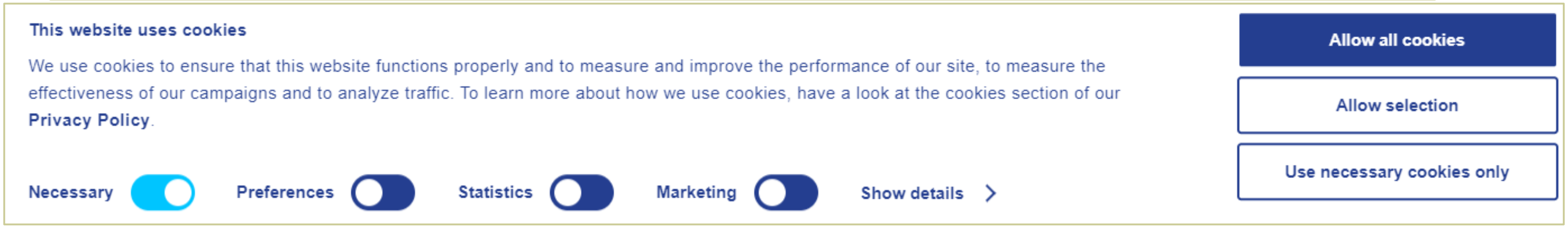
```
HTTP/1.0 200 OK
Set-Cookie: LSID=DQAAAK…Eaem_vYg; Path=/accounts; Expires=Wed, 13 Jan 2021 22:23:01 GMT; Secure; HttpOnly
Set-Cookie: HSID=AYQEVn…DKrdst; Domain=.foo.com; Path=/; Expires=Wed, 13 Jan 2021 22:23:01 GMT; HttpOnly
Set-Cookie: SSID=Ap4P…GTEq; Domain=foo.com; Path=/; Expires=Wed, 13 Jan 2021 22:23:01 GMT; Secure; HttpOnly
…
```

- The *Domain* and *Path* attributes define the cookie's scope
- The *Secure* attribute makes sure that the cookie can only be transmitted over an encrypted connection (i.e. HTTPS → later), making it a **secure cookie**
- The *HttpOnly* attribute directs browsers not to expose cookies through channels other than HTTP / HTTPS requests
  This means that this **HttpOnly cookie** cannot be accessed via client-side scripting languages (notably JavaScript)

GDPR and Cookies

- Generally, a user's consent must be sought before a cookie is installed in a web browser





- There are two exemptions:
  - The communications exemption
  - The strictly necessary exemption

Mx7

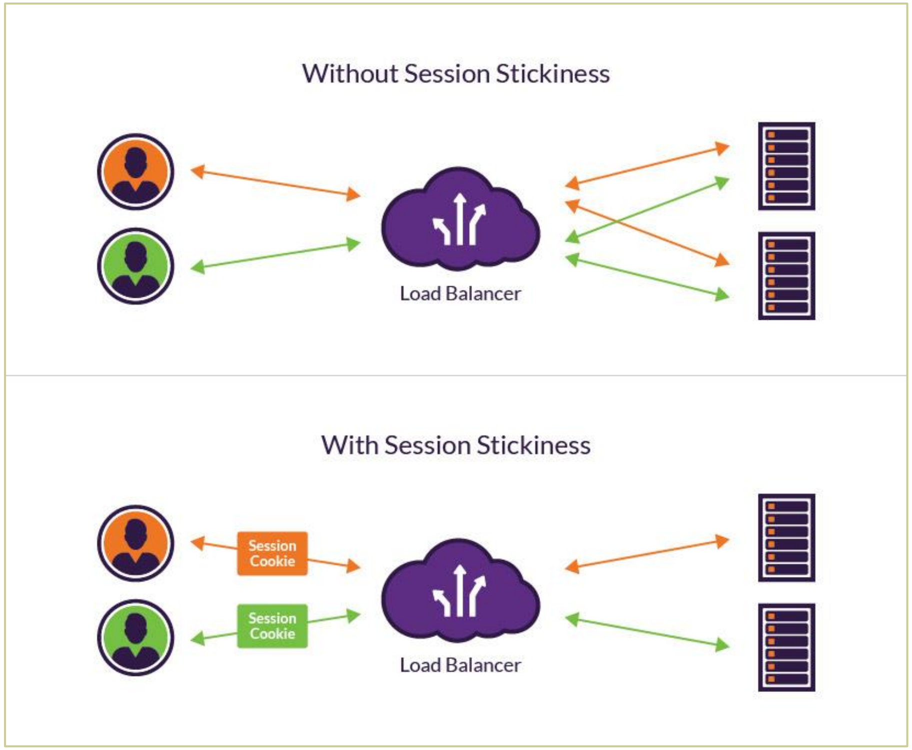
NUI Galway
OÉ Gaillimh

The *Communications Exemption*

- This applies to cookies whose sole purpose is for carrying out the transmission of a communication over a network, for example to identify the communication endpoints
- Example: load-balancing cookies that distribute network traffic across different backend servers, aka **session stickiness**
  - Here a load balancer **creates an affinity** between a client and a specific network server for the duration of a session using a cookie with a random and unique tracking id
  - Subsequently, for the duration of the session, the load balancer routes all of the requests of this client to a specific backend server using the tracking id

Session Stickiness

- Top image:
  - No load balancing at all
- Bottom image:
  - The LB generates and returns a tracking cookie back to a client when its session is initiated
  - This cookie is tagged to every subsequent client request and allows the LB to forward the request to always the same server (therefore the stickiness)



The *strictly necessary Exemption*

- Must be linked to a service delivered over the internet, i.e. a website or an app
- This service must have been explicitly requested by the user (i.e. typing in the URL) and the use of the cookie must be restricted to what is strictly necessary to provide that service
- Note that cookies related to advertising are not strictly necessary and must be consented to

Example for the *strictly necessary Exemption*

- A website uses session cookies to keep track of items a user places in an online shopping basket
  - Assuming this cookie will be deleted once the session is over
- Cookies that record a user's language or country preference

NUI Galway
OÉ Gaillimh

Data Processing

- Performing any operation on personal data, manually or by automate means, including:
  - Obtaining
  - Storing
  - Transmitting
  - Recording
  - Organising
  - Altering
  - Disclosing
  - Erasing

Entities in GDPR

- GDPR distinguishes between:
  - The Data Subject
  - The Data Protection Officer (DPO)
  - The Data Controller
  - The Data Processor

Mx5

NUI Galway
OÉ Gaillimh

The Data Subject

- This is the person to whom the data relates
  - GDPR only applies to living individuals
- However, any duty of confidence in place prior to the death extends beyond that point

NUI Galway
OÉ Gaillimh

The Data Protection Officer (DPO)

- The primary role of the DPO is to ensure that her organisation processes the personal data of its staff, customers, and other data subjects in compliance with the applicable data protection rules
- It is a mandatory role within three different scenarios:
  - When the processing is undertaken by a public authority or body
  - When an organisation's main activities require the frequent and large-scale monitoring of individual people
  - Where large scale processing of special categories of data or data relating to criminal records forms the core activities
- The Data Protection Officer is required to be an expert within this field, along with the requirement for them to report to the highest management level.
  - With this being a challenging aspect of GDPR compliance for smaller organisations, there is the option to make an external appointment of a third-party

The Data Controller

- The Data Controller is the company or an individual who has overall control over the processing of personal data
- The Data Controller takes on the responsibility for GDPR compliance
  - A Data Controller needs to have had sufficient training and be able to competently ensure the security and protection of data held within the organisation

M

NUI Galway
OÉ Gaillimh

The Data Processor

- The Data Processor is the person who is responsible for the processing of personal information
- Generally, this role is undertaken under the instruction of the data controller
  - This might mean obtaining or recording the data, it's adaption and use. It may also include the disclosure of the data or making it available for others
- Generally, the Data Processor is involved in the more technical elements of the operation, while the interpretation and main decision making is the role of the Data Controllers

Cloud Services and GDPR

- A Cloud Service Provider will be considered a **Data Processor** under GDPR if it provides data processing services (e.g. storage) on behalf of the Data Controller even without determining the purposes and means of processing
- A Cloud Service Provider that offers personal data processing services directly to Data Subjects will be **Data Controller**

NUI Galway
OÉ Gaillimh

Some Key Benefits for Data Subjects

- More information must be given to data subjects (e.g. how long data will be kept, right to lodge a complaint)
- Must explain and document legal basis for processing personal data
- Tightens the rules on how consent is obtained (must be distinguishable from other matters and in clear plain language)
- Must be as easy to withdraw consent as it is to give it
- Mandatory notification of security breaches without undue delay
  - To data protection commissioner within 72 hours

NUI Galway
OÉ Gaillimh

Personal Data Security Breaches

- Disclosure of confidential data to unauthorised individuals
- Loss or theft of data or equipment on which data is stored
- Hacking, viruses or other security attacks on IT equipment/ systems / networks
- Inappropriate access controls allowing unauthorised use of information
- Emails containing personal data sent in error to wrong recipient
- Applies to paper and electronic records

Some Key Benefits for Data Subjects

- Right of Access (copy to be provided within one month)
- Right to erasure (i.e. right to be forgotten)
- Right to restriction of processing
- Right to object to processing
- Right not to be subject to a decision based solely on automated processing

GDPR Overview

- The GDPR sets out several key principles:
  1. Lawfulness
  2. Fairness and transparency
  3. Purpose limitation
  4. Data minimisation
  5. Accuracy
  6. Storage limitation
  7. Integrity and confidentiality (security)
  8. Accountability

NUI Galway
OÉ Gaillimh

GDPR: Lawfulness

- You must **identify valid grounds** under the GDPR (known as a 'lawful basis') for collecting and using personal data
- Processing shall be lawful only if and to the extent that at least one of the following applies:
  - Consent
  - Necessary for the performance of a contract
  - Necessary for compliance with a legal obligation
  - Necessary to protect the vital interests of the data subject or another person
  - Necessary for the performance of a task carried out in the public interest
  - Necessary for the purpose of the legitimate interests

GDPR: Fairness and Transparency

- You must **use personal data in a way that is fair**; this means you must not process the data in a way that is unduly detrimental, unexpected or misleading to the individuals concerned
- You must be **clear, open and honest with people** from the start about how you will use their personal data
- At the time personal data is being collected from data subjects, they must be informed via a "Data Protection Notice"

Data Protection Notice

- A data protection notice entails the following:
  - The identity and contact details of the data controller
  - The contact details of the data protection officer
  - The purpose of the processing and the legal basis for the processing
  - The recipients or categories of recipients of the data
  - Details of any transfers out of the EEA, safeguards in place and the means by which to obtain a copy of them
  - The data retention period used or criteria to determine same
  - The individual's rights (access, rectification and erasure, restriction, complaint)

GDPR: Purpose Limitation

- You must be **clear about what your purposes** for processing are from the start
- You need to **record your purposes** as part of your documentation obligations and specify them in your privacy information for individuals
- You **can only use the personal data for a new purpose** if either this is compatible with your original purpose, you get consent, or you have a clear basis in law

X

M

NUI Galway
OÉ Gaillimh

GDPR: Data Minimisation

- You must ensure the personal data you are processing is:
  - **adequate** – sufficient to properly fulfil your stated purpose
  - **relevant** – has a rational link to that purpose
  - **limited** to what is necessary – you do not hold more than you need for that purpose

X

M

NUI Galway
OÉ Gaillimh

GDPR: Accuracy

- You should take all reasonable steps to ensure the personal data you hold **is not incorrect or misleading** as to any matter of fact
- You may need to **keep the personal data updated**, although this will depend on what you are using it for
- If you **discover that personal data is incorrect or misleading**, you must take reasonable steps to correct or erase it as soon as possible
- You must **carefully consider any challenges to the accuracy** of personal data

(M)

GDPR: Storage Limitation

- You must not keep personal data **for longer than you need it**
- You need to think about – and be able to justify – **how long you keep personal data**; this will depend on your purposes for holding the data
- You need a policy **setting standard retention periods** wherever possible, to comply with documentation requirements
- You should also **periodically review the data you hold**, and erase or anonymise it when you no longer need it
- You must **carefully consider any challenges to your retention of data**; individuals have a right to erasure if you no longer need the data
- You can **keep personal data for longer if you are only** keeping it for public interest archiving, scientific or historical research, or statistical purposes

NUI Galway
OÉ Gaillimh

GDPR: Accountability and Governance

- Accountability is one of the data protection principles - it makes you responsible for complying with the GDPR and says that **you must be able to demonstrate your compliance**
- You need to put in place appropriate technical and organisational measures to meet the requirements of accountability

NUI Galway
OÉ Gaillimh

GDPR: Accountability and Governance

- Accountability requires controllers to maintain records of processing activities in order to demonstrate how they comply with the data protection principles, i.e.
  - Inventory of personal data
  - Providing assurance about compliance
  - Need to document
    - Why it is held
    - How it is collected
    - When it will be deleted
    - Who may gain access to it

NUI Galway
OÉ Gaillimh

GDPR: Integrity and Confidentiality

- A key principle of the GDPR is that you process personal data securely by means of ‘appropriate technical and organisational measures’ – this is the ‘security principle’
- Doing this requires you to consider things like risk analysis, organisational policies, and physical and technical measures
- Where appropriate, you should look to use measures such as **pseudonymisation and encryption**
- Your measures must ensure the ‘**confidentiality, integrity and availability**’ of your systems and services and the personal data you process within them
- The measures must also enable you to **restore access and availability** to personal data in a timely manner in the event of a physical or technical incident

CT255
INTRODUCTION TO CYBERSECURITY

INTRODUCTION CRYPTOGRAPHY

Dr. Michael Schukat

OÉ Gaillimh
NUI Galway

Lecture Overview

- In this slide deck we are looking into some classical cryptographic concepts / algorithms, thereby identifying their weaknesses
- This levels the ground for our next topic, i.e. modern cryptography

Recap: What is Cybersecurity?

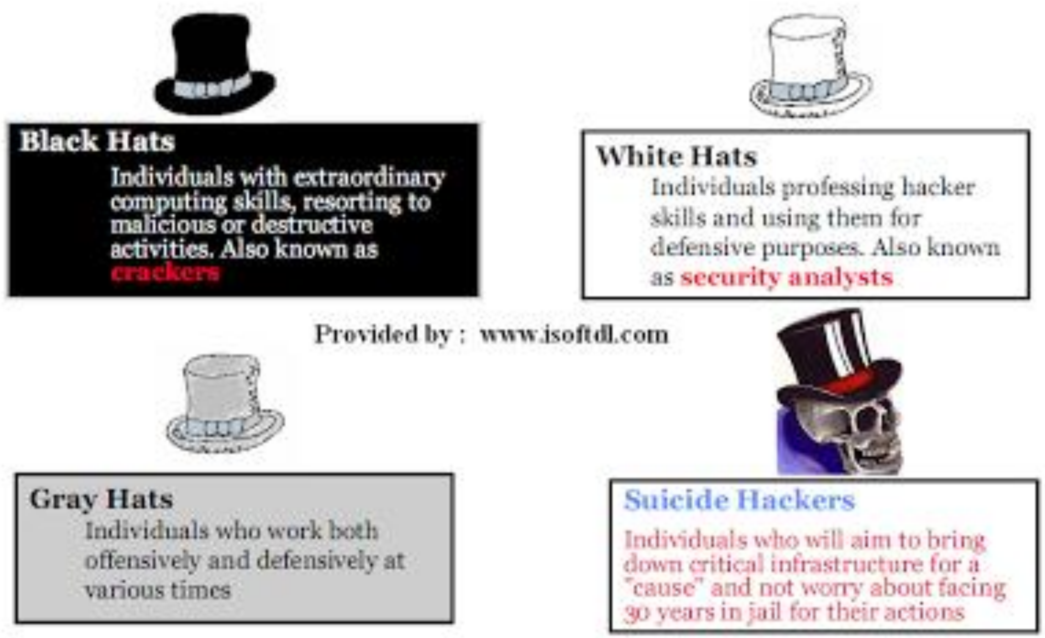
- Cybersecurity is the practice of protecting systems, networks, and programs from digital attacks. These cyberattacks are usually aimed at accessing, changing, or destroying sensitive information; extorting money from users; or interrupting normal business processes
  Source: Cisco

If this was Hogwarts…

- … the equivalent of this subject would have been taught by:

- What subject are we talking about?

Our Witches and Wizards



You find them Everywhere…

CONNACHT TRIBUNE

NEWS SPORT LIFE ENTERTAINMENT BUSINESS PROPERTY CARS

ELECTIONS

By **Bernie Ni Fhlatharta** - May 21, 2013

A Claregalway man is facing the prospect of up to 20 years in a US prison after he was named this week by the FBI as a founder member of an international internet hacking group.

[redacted] from Cloonbiggeen, Claregalway, is charged with two counts of computer hacking conspiracy – each conspiracy count carries a maximum sentence of ten years in

[redacted] is alleged by the FBI to be a member of 'LulzSec', a group of internet hackers that is a spin-off of the Anonymous hacking group. Both groups have launched numerous cyber attacks on high profile websites around the world.

[redacted] a biopharmaceutical chemistry student at NUI Galway and a past pupil of Calasanctius College, Oranmore, is listed in the FBI's court papers as being 25, however, it is understood he is only 19 or 20.

Example SQL Injections

- SQL injection is a code injection technique, used to attack data-driven applications, in which malicious SQL statements are inserted for execution
- A way of exploiting user input and SQL Statements to compromise the database and/or retrieve sensitive data

Case Study

- Consider a SQL injection attack on an Irish online retailer revealed the following database table called "CustomerAccounts":

CustomerId	EncryptedIBAN
23	XPF7F3FD78FS8HGF9S5SL6
367	XPHDSYUEGSD68G4AS8AG56
66	XPEFGS567DS09123SD342G

- In a plaintext IBAN, The first two letters denote the country code (e.g., IE for Ireland), then two check digits, and finally a country-specific Basic Bank Account Number (BBAN), which includes the domestic bank account number, branch identifier, and potential routing information

In-Class Activity

- What are your observations / ideas regarding the entries in "EncodedIBAN“, e.g.:
  - How does the transformation work?
  - Any patterns you can see?

Some basic Terminology

- Cryptography
  - The art of encompassing the principles and methods of transforming an intelligible message into one that is unintelligible, and then retransforming that message back to its original form.
    - Intelligible means “able to be understood” or comprehensible

Some basic Terminology

- Plaintext
  - The original intelligible message, e.g. “IE64IRCE92050112345678”
- Ciphertext
  - The transformed message, e.g. “XPHDSYUEGSD68G4AS8AG56”
- Cipher
  - An algorithm for transforming an intelligible message into one that is unintelligible
- Key
  - Some critical information used by the cipher, known only to the sender & receiver; selected from a **keyspace** K (i.e. a set of all possible keys)

Some basic Terminology

- Encipher (encode)
  - The process of converting plaintext to ciphertext using a cipher and a key
- Encryption
  - The mathematical function $E_K()$ mapping plaintext P to ciphertext using the specified key K:

    $$C = E_K(P)$$

Some basic Terminology

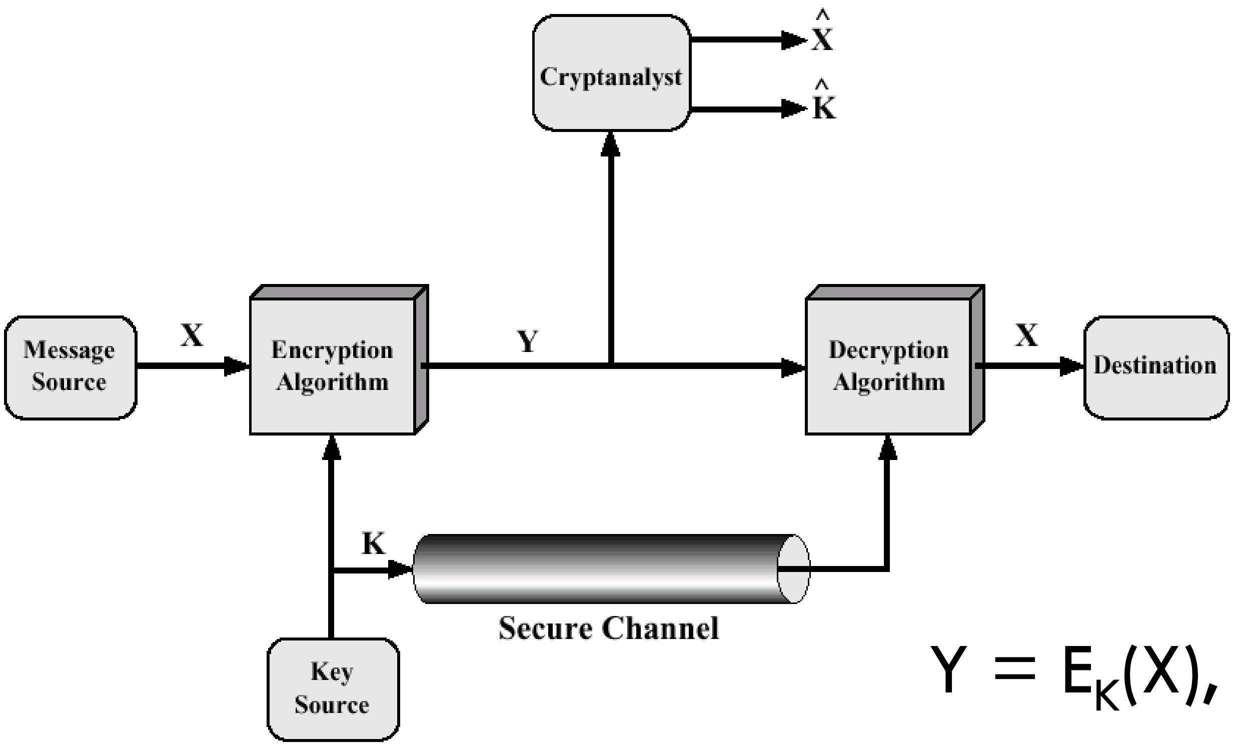
- Decipher (decode)
  - The process of converting ciphertext back into plaintext using a cipher and a key
- Decryption:
  - The mathematical function $E_K^{-1}()$ mapping ciphertext C to plaintext P using the specified key K:

    $$P = E_K^{-1}(C)$$

Basic Terminology

- Cryptanalysis
  - The study of principles and methods of transforming an unintelligible message back into an intelligible message without knowledge of the key
- Cryptology
  - The field encompassing both cryptography and cryptanalysis

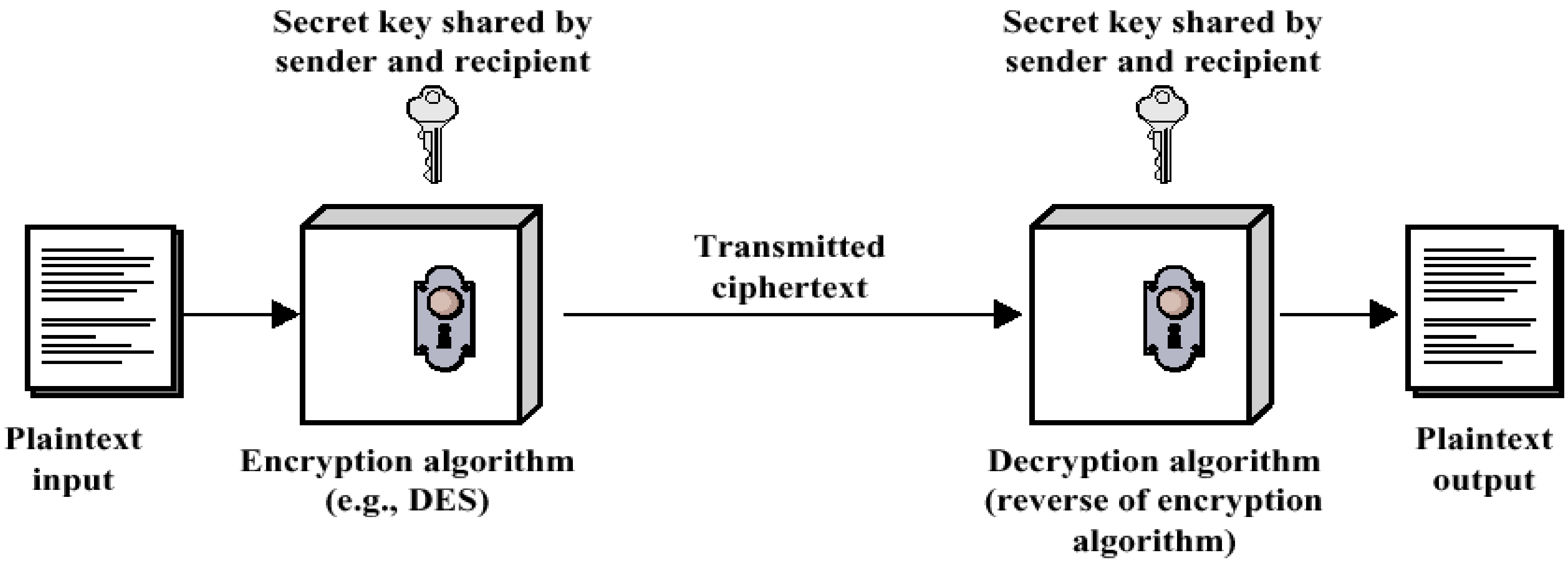
Model of Conventional Cryptosystem



$$Y = E_K(X),\ X = E_K^{-1}(Y)$$

Classical Cryptography

- Ancient ciphers have been in use for over 5,000 years
- Already used by ancient Egyptians, Hebrews and Greeks
- Normally they would follow the following scheme:



Caesar Cipher

- 2000 years ago Julius Caesar used a simple substitution cipher, now known as the Caesar cipher
- First attested use in military affairs (Gallic Wars)
- Replace each letter by 3rd letter on, e.g.

  L FDPH L VDZ L FRQTXHUHG ->
  I CAME I SAW I CONQUERED

- We can describe this mapping (or translation alphabet) as:

  Plain: ABCDEFGHIJKLMNOPQRSTUVWXYZ
  Cipher: DEFGHIJKLMNOPQRSTUVWXYZABC

Generalised Caesar Cipher

- More generally can use any shift from 1 to 25, i.e. replace each letter of message by a letter a fixed distance away
- Specify key letter as the letter a plaintext A maps to,
  - e.g. a key letter of F means
    A maps to F, B to G, ... Y to D, Z to E
    e.g. shift letters by 5 places
- Hence have 26 (25 useful) ciphers

- Try all 25 possibilities until you recover some meaningful text

KEY	PHHW PH DIWHU WKH WRJD SDUWB
1	oggv og chvgt vjg vqic rctva
2	nffu nf bgufs uif uphb qbsuz
3	meet me after the toga party
4	ldds ld zesdq sgd snfz ozqsx
5	kccr kc ydrcp rfc rmey nyprw
6	jbbq jb xcqbo qeb qldx mxoqv
7	iaap ia wbpan pda pkcw lwnpu
8	hzzo hz vaozm ocz ojbv kvmot
9	gyyn gy uznyl nby niau julns
10	fxxm fx tymxk max mhzt itkmr
11	ewwl ew sxlwj lzw lgys hsjlq
12	dvvk dv rwkvi kyv kfxr grikp
13	cuuj cu qvjuh jxu jewq fqhjo
14	btti bt puitg iwt idvp epgin
15	assh as othsf hvs hcuo dofhm
16	zrrg zr nsgre gur gbtn cnegl
17	yqqf yq mrfqd ftq fasm bmdfk
18	xppe xp lqepc esp ezrl alcej
19	wood wo kpdob dro dyqk zkbdi
20	vnnc vn jocna cqn cxpj yjach
21	ummb um inbmz bpm bwoi xizbg
22	tlla tl hmaly aol avnh whyaf
23	skkz sk glzkx znk zumg vgxze
24	rjjy rj fkyjw ymj ytlf ufwyd
25	qiix qi ejxiv xli xske tevxc

In-Class Activity

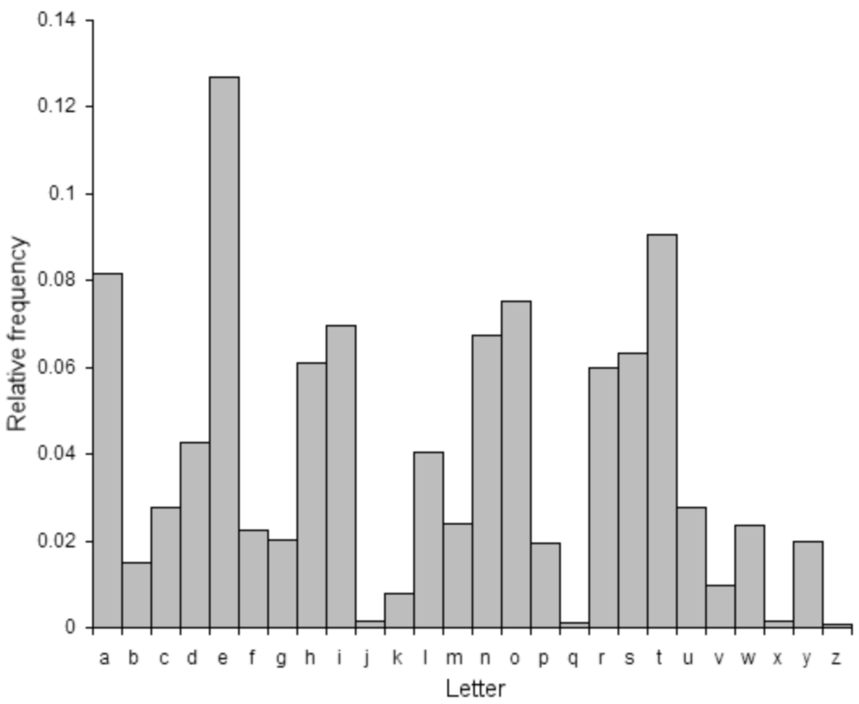
- Encode the plaintext "**KENSENTME**" using the Caesar cipher

Simple Substitution Cipher

- Cipher: Replace each plaintext letter with the corresponding ciphertext alphabet letter (only one letter at a time, therefore "simple")
- Plaintext alphabet: ABCDEFGHIJKLMNOPQRSTUVWXYZ
- Ciphertext alphabet (i.e. the key): ZEBRASCDFGHIJKLMNOPQTUVWXY
- Plaintext message: FLEEATONCEWEAREDISCOVERED
- Ciphertext message: SIAAZQLKBAVAZOARFPBLUAOAR
- **26! (= 4.0329146 * 10^{26}) possible key combinations … unbreakable?**

Cryptanalysis via Letter Frequency Distribution in English Language

- Human languages are redundant
- Letters are not equally commonly used
- In the English language,
  - E is by far the most common letter followed by T,R,N,I,O,A,S
  - other letters like Z,J,K,Q,X are fairly rare
  - certain letter combinations, e.g. TH, are quite common
- There are tables of single, double & triple letter frequencies for various languages
- See the example code on the next slide



C-Program for Frequency Analysis of single Characters

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int main(int argc, char *argv[])
{
    FILE *fp;
    int data[26];
    char c;
    int i;

    memset(data, 0, sizeof(data));

    if (argc != 2)
      return(-1);

    if ((fp = fopen(argv[1], "r")) == NULL)
            return(-2);

    while (!feof(fp))
    {
            c = toupper(fgetc(fp));

            if ((c >= 'A') && (c <= 'Z'))
                    data[c - 65]++;
    }

    for (i = 0; i < 26; i++)
            printf("%c: %i\n", i + 65, data[i]);

    fclose(fp);
    return(1);
}
```

Example Cryptanalysis of Simple Substitution Cipher

- Given ciphertext: UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZVUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSXEPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
- Count number of occurrences of each letter in text
- Guess ciphertext letters P & Z are plaintext letters e and t (we use small letters to distinguish between both): UtQSOVUOHXMOeVGeOteEVSGtWStOeFeESXUDBMETSXAItVUEeHtHMDtSHtOWSFeAeeDTSVeQUZWYMXUtUHSXEeYEeOeDtStUFeOMBtWeFUetHMDJUDTMOHMQ

Example Cryptanalysis

- Guess (!) Z?P means *the*:
  UtQSOVUOHXMOeVGeOteEVSGtWStOeFeESXUDBMET
  SXAItVUEeHtHMDtSHtOWSFeAeeDTSVeQUZWYMXUtUH
  SXEeYEeOeDtStUFeOMBt***W***eFUetHMDJUDTMOHMQ
- Assume W is *h*:
  UtQSOVUOHXMOeVGeOteEVSGt**h**StOeFeESXUDBMETS
  XAItVUEeHtHMDtSHtO**h**SFeAeeDTSVeQUZWYMXUtUHSX
  EeYEeOeDtStUFeOMBt**h**eFUetHMDJUDTMOHMQ

Example Cryptanalysis

- Guess word *that*, translating S into a:
  UtQSOVUOHXMOeVGeOteEVSG***thSt***OeFeESXUDBMET
  SXAItVUEeHtHMDtSHtOhSFeAeeDTSVeQUZWYMXUtUH
  SXEeYEeOeDtStUFeOMBtheFUetHMDJUDTMOHMQ
- Ciphertext becomes:
  UtQ**a**OVUOHXMOeVGeOteEV**a**G***that***OeFeE**a**XUDBMET
  aXAItVUEeHtHMDt**a**HtOhsFeAeeDT**a**VeQUZWYMXUtUH
  aXEeYEeOeDt**a**tUFeOMBtheFUetHMDJUDTMOHMQ

Example Cryptanalysis

- Guess that AeeD means *been*:
  UtQaOVUOHXMOeVGeOteEVaGthatOeFeEaXUDBM ETaXAItVUEeHtHMDtaHtOhsFe**AeeD**TaVeQUZWYMXU tUHaXEeYEeOeDtatUFeOMBtheFUetHMDJUDTMOHM Q
- Resulting in (with A→b and D→n):
  UtQaOVUOHXMOeVGeOteEVaGthatOeFeEaXUnBM ETaX**b**ItVUEeHtHM**n**taHtOhsFe**been**TaVeQUZWYMXUt UHaXEeYEeOe**n**tatUFeOMBtheFUetHM**n**JU**n**TMOHMQ

Example Cryptanalysis

- Is HMntaHt meaning *contact*?
  UtQaOVUOHXMOeVGeOteEVaGthatOeFeEaXUnBMET
  aXbItVUEeHt**HMntaHt**OhsFebeenTaVeQUZWYMXUtUH
  aXEeYEeOentatUFeOMBtheFUetHMnJUnTMOHMQ
- Therefore (with H→ c and M→ o):
  UtQaOVUO**c**X**o**OeVGeOteEVaGthatOeFeEaXUnBoETa
  XbItVUEe**ct****contact**OhaFebeenTaVeQUZWY**o**XUtU**c**aXEe
  YEeOentatUFeO**o**BtheFUet**co**nJUnT**o**O**co**Q

Example Cryptanalysis

- Does VUEect mean *direct*?
  UtQaOVUOcXoOeVGeOteEVaGthatOeFeEaXUnBoETaX
  bIt**VUEect**contactOhaFebeenTaVeQUZWYoXUtUcaXEeY
  EeOentatUFeOoBtheFUetconJUnToOcoQ
- Therefore (with V→ d, U → i and E→ r):
  itQaO**di**OcXoOe**d**GeOte**r**daGthatOeFe**r**aX**i**nBorTaXbIt
  directcontactOhaFebeenTadeQ**i**ZWYoX**i**t**i**caX**r**eY**r**eOent
  at**i**FeOoBtheF**i**etconJ**i**nToOcoQ

Example Cryptanalysis

- Does GeOterdaG mean yesterday? itQaOdiOcXoOed**<u>GeOterdaG</u>**thatOeFeraXinBorTaXbIt directcontactOhaFebeenTadeQiZWYoXiticaXreYreOent atiFeOoBtheFietconJinToOcoQ
- Therefore (with G→ y and O → s): itQa**<u>s</u>**di**<u>s</u>**cXo**<u>s</u>**ed**<u>yesterday</u>**that**<u>s</u>**eFeraXinBorTaXbItdirect contactshaFebeenTadeQiZWYoXiticaXreYre**<u>s</u>**entatiFeso BtheFietconJinToscoQ

Example Cryptanalysis

- Moscow calling?
  itQasdiscXosedyesterdaythatseFeraXinBorTaXbltdirectcontactshaFebeenTadeQiZWYoXiticaXreYresentatiFesoBtheFietconJin**ToscoQ**
- Therefore (with T → m and Q → w):
  it**w**asdiscXosedyesterdaythatseFeraXinBor**m**aXbltdirectcontactshaFebeen**m**ade**w**iZWYoXiticaXreYresentatiFesoBtheFietconJin**moscow**

Example Cryptanalysis

- X means *l*, F means *v*, B means *f*?
  itwas**discXosed**yesterdaythat**seFeraXinBormaX**bltdir ectcontactshaFebeenmadewiZWYoXiticaXreYresentati FesoBtheFietconJinmoscow
- Therefore:
  itwas**disclosed**yesterdaythat**severalinformal**bltdirectc ontactshavebeenmadewiZWYoliticalreYresentativesoft hevietconJinmoscow

Example Cryptanalysis

- I means *u*, Z means *t*, W means *h*, Y means *p*?
  itwasdisclosedyesterdaythatseveralinformal**blt**directcontactshavebeenmade**wiZW**YoliticalreYresentativesofthevietconJinmoscow
- Therefore:
  itwasdisclosedyesterdaythatseveralinformal**but**directcontactshavebeenmade**with**politicalrepresentativesofthevietconJinmoscow

Example Cryptanalysis

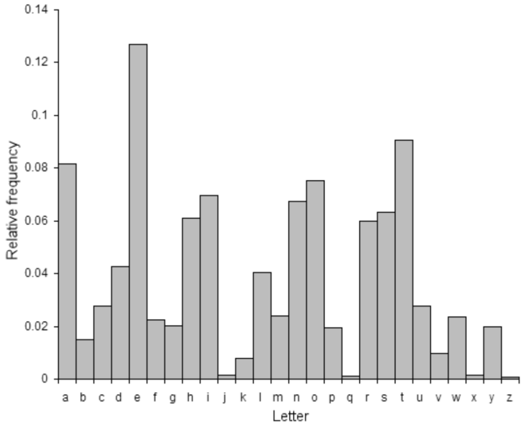
- Finally: J means *g*:
  itwasdisclosedyesterdaythatseveralinformalbutdirectc
  ontactshavebeenmadewithpoliticalrepresentativesofth
  e**vietconJ**inmoscow
- Therefore (with spaces added):
  it was disclosed yesterday that several informal but
  direct contacts have been made with political
  representatives of the vietcong in moscow

Known Plaintext Attacks (KPA)

- The known-plaintext attack (KPA) is an attack model for cryptanalysis where the attacker has access to both
  - (some of the) the plaintext (called a crib),
  - and its encrypted version
- Recall the IBAN example

In-Class Activity

- You are presented with the following ciphertext which is based on a simple substitution cipher:
  JEPOUMJWFIFSFCVUNZIPNFJTNZDBTUMFGVMMTUPQ
- You know the original plaintext message consists of capital letters only (no spaces) and contains the following plaintext crib:
  MYHOMEISMYCASTLE
- How could you tackle this?



Playfair Cipher

- Not even the large number of keys in a monoalphabetic cipher provides security!
  - A monoalphabetic cipher is any cipher in which the letters of the plain text are mapped to cipher text letters based on a single alphabetic key
- One approach to improving security was to encrypt multiple letters
- The **Playfair Cipher** is an example for such an approach
- Algorithm was invented by Charles Wheatstone in 1854, but named after his friend Baron Playfair

Playfair Cipher

I/J	R	E	L	A
N	D	B	C	F
G	H	K	M	O
P	Q	S	T	U
V	W	X	Y	Z

- How it works:
  - Create a 5x5 grid of letters; insert the keyword as shown, with each letter only considered once; fill the grid with the remaining letters in alphabetic order
  - Letters are encrypted in pairs
  - Repeats have an X inserted: BALLOON -> BA LX LO ON
  - Letters that fall in the same row are each replaced with the letter on the right (OK becomes GM)
  - Letters in the same column are replaced with the letter below (FO becomes OU)
  - Otherwise each letter gets replaced by the letter in its row but in the other letters column (QM becomes TH)
- But again … Playfair can be cracked through frequency analysis of letter pairs

Security of Playfair Cipher

- Security much improved over simple monoalphabetic cipher, since we have 26 x 26 = 676 combinations
- This requires a 676 entry frequency table to analyse (verses 26 for a monoalphabetic) and correspondingly more ciphertext
- It was widely used for many years, e.g. by US & British military in WW1
- But it **can** be broken via frequency analysis of pairs of letters, given a few hundred letters

M

In-Class Activity

- Consider the Playfair Cipher and the key "PRUNEJUICE"
- Encipher the following plaintext: "KENSENTMEX"
- What is the resulting ciphertext?

Vigenère Cipher

- Blaise de Vigenère is generally credited as the inventor of the "polyalphabetic substitution cipher“
  - A monoalphabetic cipher is any cipher in which the letters of the plain text are mapped to cipher text letters based on a single alphabetic key
  - A polyalphabetic substitution ciphers uses multiple substitution alphabets
- To improve security use many monoalphabetic substitution alphabets
- Hence each letter can be replaced by many others
- Use a key to select which alphabet is used for each letter of the message
- i^{th} letter of key specifies i^{th} alphabet to use
- Use each alphabet in turn
- Repeat from start after end of key is reached

Vigenère Example

- Write the plaintext out and under it write the keyword repeated
- Then using each key letter in turn as a Caesar cipher key
- Encrypt the corresponding plaintext letter. Example:

Plaintext THISPROCESSCANALSOBEEXPRESSED
Keyword CIPHERCIPHERCIPHERCIPHERCIPHE
Ciphertext VPXZTIQKTZWTCVPSWFDMTETIGAHLH
In this example have the keyword "CIPHER". Hence have the following translation alphabets:
C -> CDEFGHIJKLMNOPQRSTUVWXYZAB
I -> IJKLMNOPQRSTUVWXYZABCDEFGH

... ...
ABCDEFGHIJKLMNOPQRSTUVWXYZ

to map the above plaintext letters

In-Class Activity (Menti)

- Encode the plaintext "**KENSENTME**" using the Vigenère cipher and the keyword "BABA"

How to crack the Vigenère Cipher

- Search the ciphertext for repeated strings of letters; the longer strings you find the better
- For each occurrence of a repeated string, count how many letters are between the first letters in the string and add one
- Factor the number you got in the above computation (e.g. 2, 5 and 10 itself are factors of 10)
- Repeat this process with each repeated string you find and make a table of common factors. The most common factor is probably the length of the keyword that was used to encipher the ciphertext. Call this number 'n'
- Do a frequency count on the ciphertext, on every nth letter. You should end up with n different frequency counts
- Compare these counts to standard frequency tables to figure out how much each letter was shifted by
- Undo the shifts and read off the message!

Example

Key: ABCDAB CD ABCDA BCD ABCDABCDABCD

Plaintext: **CRYPTO** IS SHORT FOR **CRYPTO**GRAPHY

Ciphertext: **CSASTP** KV SIQUT GQU **CSASTP**IUAQJB

Distance is 16, therefore the key length is either 2, 4, 8 or 16 characters

In-Class Activity

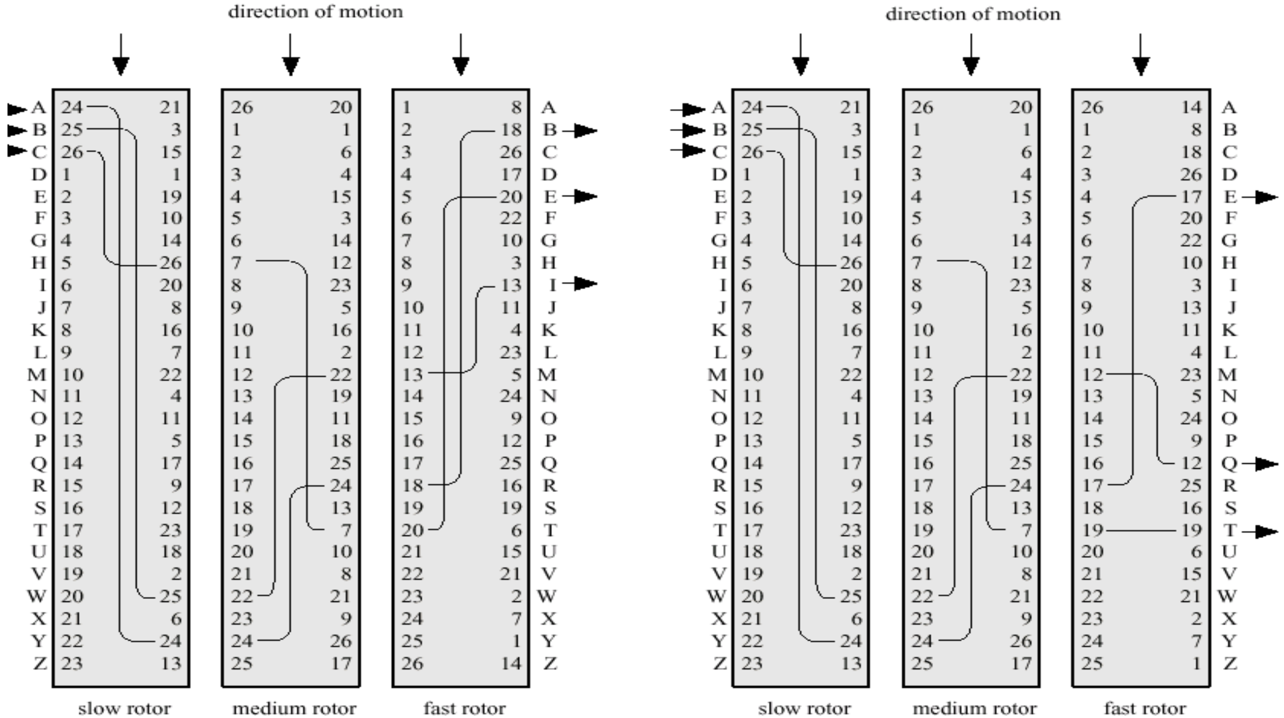
- Consider the following ciphertext that has been encoded using a Vigenère Cipher:

DYDUXRMHTVDVNQDQNWDYDUXRMHARTJGWNQD

- Q1: Which repeating strings can you identify?
- Q2: What is the distance of their appearances?
- Q3: Subsequently, what is the probable key length?

Rotor Ciphers

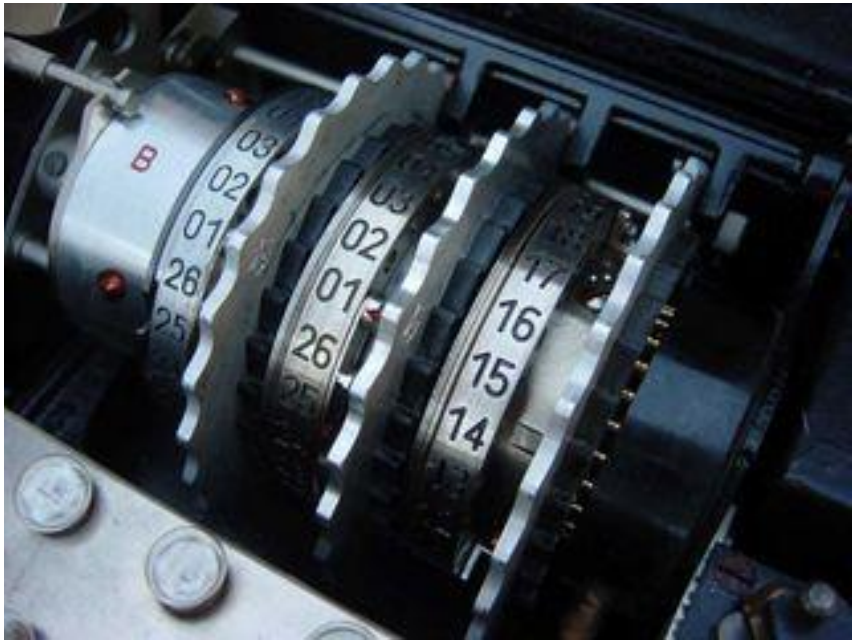
- The mechanisation / automation of encryption
- A N-stage polyalphabetic substitution algorithm modulo 26.
- 26^N steps before a repetition (N = 5 cylinders == 11881376 steps)



(a) Initial setting

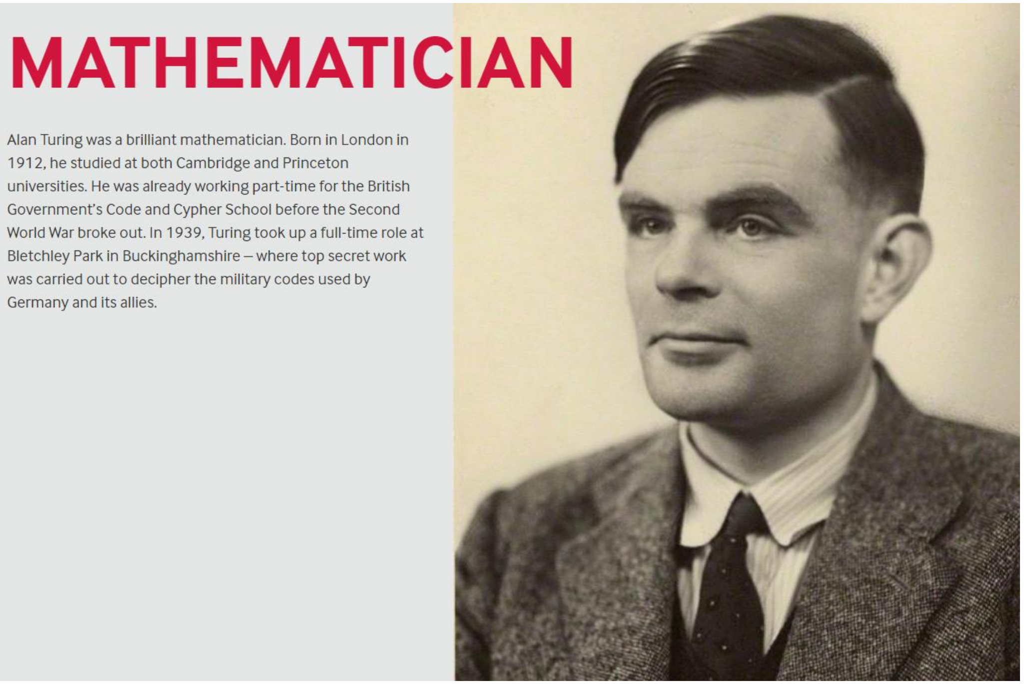
(b) Setting after one keystroke

The Enigma Machine



How Alan Turing broke the Enigma Code

- https://www.iwm.org.uk/history/how-alan-turing-cracked-the-enigma-code
- The Imitation Game (Film, 2014)
- https://www.youtube.com/watch?v=-mdSvGUd0_c



Breaking Enigma using Cribs

- The starting point for breaking Enigma were based on the following:
  - Plaintext messages were likely to contain certain phrases, e.g.
    - Weather reports contained the term ”WETTER VORHERSAGE”
    - Military units often sent messages containing “KEINE BESONDEREN EREIGNISSE”, i.e. “nothing to report”
  - A plaintext letter was never mapped onto the same ciphertext letter

Breaking Enigma using Cribs (Wikipedia)

- While the cryptanalysts in Bleachy Park did not know where exactly these cribs were placed in an intercepted message, they could exclude certain positions (i.e. Position 1 and 3):

Ciphertext	O	H	J	Y	P	D	O	M	Q	N	J	C	O	S	G	A	W	H	L	E	I	H	Y	S	O	P	J	S	M	N	U
Position 1			K	E	I	N	E	B	E	S	O	N	D	E	R	E	N	E	R	E	I	G	N	I	S	S	E				
Position 2				K	E	I	N	E	B	E	S	O	N	D	E	R	E	N	E	R	E	I	G	N	I	S	S	E			
Position 3					K	E	I	N	E	B	E	S	O	N	D	E	R	E	N	E	R	E	I	G	N	I	S	S	E		
	Positions 1 and 3 for the possible plaintext are impossible because of matching letters. The red cells represent these *crashes*. Position 2 is a possibility.																														

- From here on, possible rotor start positions and rotor wiring would be systematically examined using a “the bombe”, an electromechanical device designed by Alan Turing

Transposition Ciphers

- Now consider classical **transposition** or **permutation** ciphers
- These hide the message by rearranging the letter order without altering the actual letters used
- This can be recognised since ciphertext has the same frequency distribution as the original text

Rail Fence Cipher

- Write message letters out diagonally over a number of rows, then read off cipher row by row.
- Example: write message out as:

```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

- Resulting ciphertext:

```
MEMATRHTGPRYETEFETEOAAT
```

In-Class Activity (Menti)

- The following ciphertext was encoded using the rail fence cipher over X rows: LEOREEOFEATUHPSMTELE
- Please decode

Row Transposition Ciphers

- This is a more complex transposition.
- Write letters of message out in rows over a specified number of columns.
- Then reorder the columns according to some key before reading off the columns.
- Example:

```
Key:          4 3 1 2 5 6 7
Plaintext:    A T T A C K P
              O S T P O N E
              D U N T I L T
              W O A M X Y Z

Ciphertext: TTNA APTM TSUO AODW COIX KNLY PETZ (spaces are inserted to
                                                 improve readability)
```

Product Ciphers

- Ciphers using substitutions or transpositions are not secure because of language characteristics
- Hence consider using several ciphers in succession to make harder:
  - two substitutions make a more complex substitution
  - two transpositions make more complex transposition
  - but a substitution followed by a transposition makes a new much harder cipher
- This is bridge from classical to modern ciphers

Steganography

Steganography

- An alternative to encryption
- Hides existence of message:
  - Using only a subset of letters/words in a longer message marked in some way
  - Using invisible ink
  - Hiding in LSB in graphic image or sound file
- Drawback:
  - Not very economical in terms of overheads to hide a message (see also assignment)

(Silly) Steganography Example

Shopping List:

- LEEKS
- EGGS
- TOMATOS
- MARGERINE
- EDAMER CHEESE
- GRAPES
- ONIONS

(Silly) Steganography Example

Shopping List:

- LEEKS
- EGGS
- TOMATOS
- MARGERINE
- EDAMER CHEESE
- GRAPES
- ONIONS

Example for Steganography

- Assume an x-by-y pixels image is stored in RGB format.
- For each pixel each colour component (R, G and B) intensity is represented by a byte
- So the image can be stored in a byte array of size [x][y][3]
- For each entry we change the LSB to hide bitwise a message, e.g.
-

R	G	B	becomes	R	G	B
01010110	11100101	10110000		01010111	11100100	10110000
11111111	10101001	00101010		11111111	10101000	00101011
11001101	10011001	11001010		11001100	10011001	11001010
…				…		

- This transformation allows the storage of the bit pattern 100101010, while preserving the main image characteristics.
- Since only the LSB of the colour information changes, the image is only very slightly distorted.
- However, image compression (e.g. JPEG) will interfere with steganographic content!

CT255
Introduction to Cybersecurity

Lecture 3

Human Security - Passwords

Dr. Michael Schukat, 2021-22

Background and Lecture Overview

- Security is only as good as its weakest link, and in many organisations this link is the human factor
- In today's lecture we'll study different authentication methodologies, including passwords, and their inherent weaknesses

Learning Outcomes

- You'll be able to:
  - Distinguish between different authentication methods, their strengths and weaknesses
  - Explore strategies to predict user passwords

What is a Password?

- A memorized secret used to confirm the identity of a user
  - Typically an arbitrary string of characters including letters, digits, or other symbols
  - A purely numeric secret is called a personal identification number (PIN)
- The secret is memorized by a party called the **claimant** while the party verifying the identity of the claimant is called the **verifier**
- Claimant and verifier communicate via an **authentication protocol**

Some Password Alternatives

- One-time password (OTP)
  - Transaction authentication number (TAN) list used for online banking – they can only be used once
- Time-synchronized one-time passwords

- Biometric methods
  - fingerprints, irises, voice, face
- Cognitive passwords
  - Use question and answer cue/response pairs to verify identity

Examples for TAN Lists

TAN-Liste für StudIS erstellt am 20.11.2017

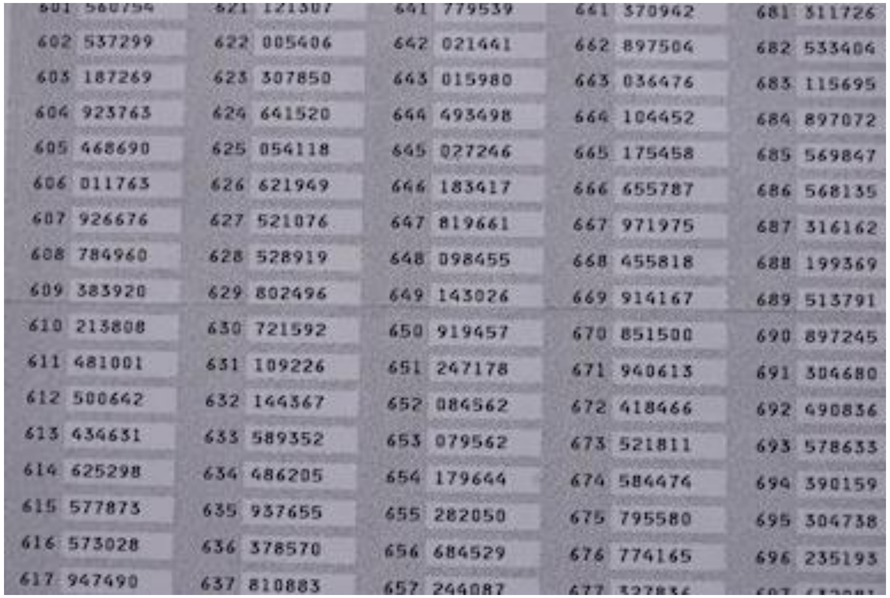
Diese TAN-Liste muss unmittelbar nach der Erzeugung mit der ersten TAN freigeschaltet werden.

This TAN-list has to be activated immediately with the first tan of this list.

TAN	Bemerkungen	TAN	Bemerkungen
443396	Freischalten dieser TAN-Liste Activate this TAN-list	254345	
564055		107066	
284347		461397	
387404		477615	
534976		497612	
187902		937527	
204473		357818	
687655		738565	
293700		491702	
984747		897643	
716142		259718	
324188		976025	
858152		862605	
185830		536734	
728760		132932	
850885		457904	
848746		858799	
537188		129830	
275827		513355	
783379		708786	
934024		715014	
953396		940817	
266699		647592	
168040		776139	Erstellen einer weiteren TAN-Liste Create a further TAN-list
607441		315877	Freischalten der weiteren TAN-Liste Activate a further TAN-list

Weitere Möglichkeiten, an eine new TAN-Liste zu kommen, finden Sie hier http://cms.uni-konstanz.de/studis/tan

Further possibilities to get a new TAN-list are described here http://cms.uni-konstanz.de/studis/tan



601	560754	621	121307	641	779539	661	370942	681	311726
602	537299	622	005406	642	021441	662	897504	682	533406
603	187269	623	307850	643	015980	663	036476	683	115695
604	923763	624	641520	644	493498	664	104452	684	897072
605	468690	625	054118	645	027246	665	175458	685	569847
606	011763	626	621969	646	183417	666	655787	686	568135
607	926676	627	521076	647	819661	667	971975	687	316162
608	784960	628	528919	648	098455	668	455818	688	199369
609	383920	629	802496	649	143026	669	914167	689	513791
610	213808	630	721592	650	919457	670	851500	690	897245
611	481001	631	109226	651	247178	671	940613	691	304680
612	500642	632	144367	652	084562	672	418466	692	490836
613	434631	633	589352	653	079562	673	521811	693	578633
614	625298	634	486205	654	179644	674	584476	694	390159
615	577873	635	937655	655	282050	675	795580	695	304738
616	573028	636	378570	656	684529	676	774165	696	235193
617	947490	637	810883	657	244087	677	[illegible]	[illegible]	[illegible]

NUI Galway
OÉ Gaillimh

Algorithmic Generation of OTP

- Paper-based TANs are hard to manage
- On the other hand both claimant and verifier need to have a copy of every OTP (possibly hundreds of them)
- Idea: Each new OTP may be created from the past OTPs used
- An example of this type of algorithm, credited to Leslie Lamport, uses a one-way function (hash function)

One-Way Functions

- A one-way function H produces a fixed-size output h based on a variable size input s
  - H(s) = h
  - H is also called a hash function, h is called a hash (value)
  - Example:
    H("KenSentMe!") = "7b24afc8bc80e548d66c4e7ff72171c5"
- Important: **One way property**:
  For a given hash code h it is infeasible to find s that H(s) = h

NUI Galway
OÉ Gaillimh

Leslie Lamport's Algorithm

- For every claimant a random seed (starting value) s is chosen
- A hash function H(s) is applied repeatedly (for example, 1000 times) to the seed, giving a value of: $H(H(H(\dots H(s) \dots)))$
- This value, also called $H^{1000}(s)$, is stored by the verifier
- The claimant keeps the seed s

Leslie Lamport's Algorithm

- The user's first login uses an OTP p derived by applying H 999 times to the seed, i.e. $H^{999}(s)$)
- The verifier can authenticate that this is the correct OTP, because $H(p) = H^{1000}(s)$, the value stored
- The value stored is then replaced by p and the user is allowed to log in

NUI Galway
OÉ Gaillimh

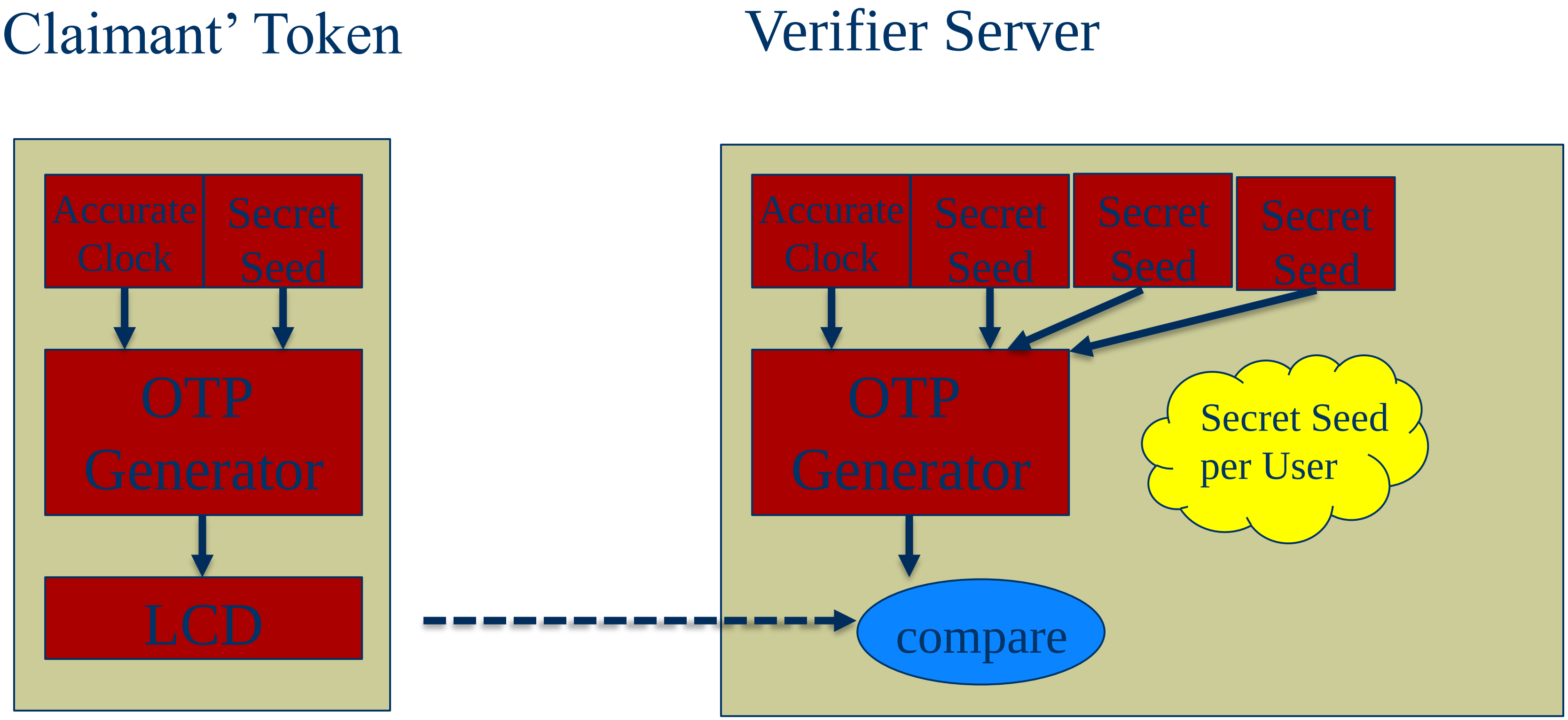
Leslie Lamport's Algorithm

- The next login must be accompanied by $H^{998}(s)$
- Again, this can be validated because hashing gives $H^{999}(s)$ which is p, the value stored after the previous login
- The new value replaces p and the user is authenticated
- This process can be repeated another 997 times, each time the password will be H applied one fewer times

Time-synchronised OTP

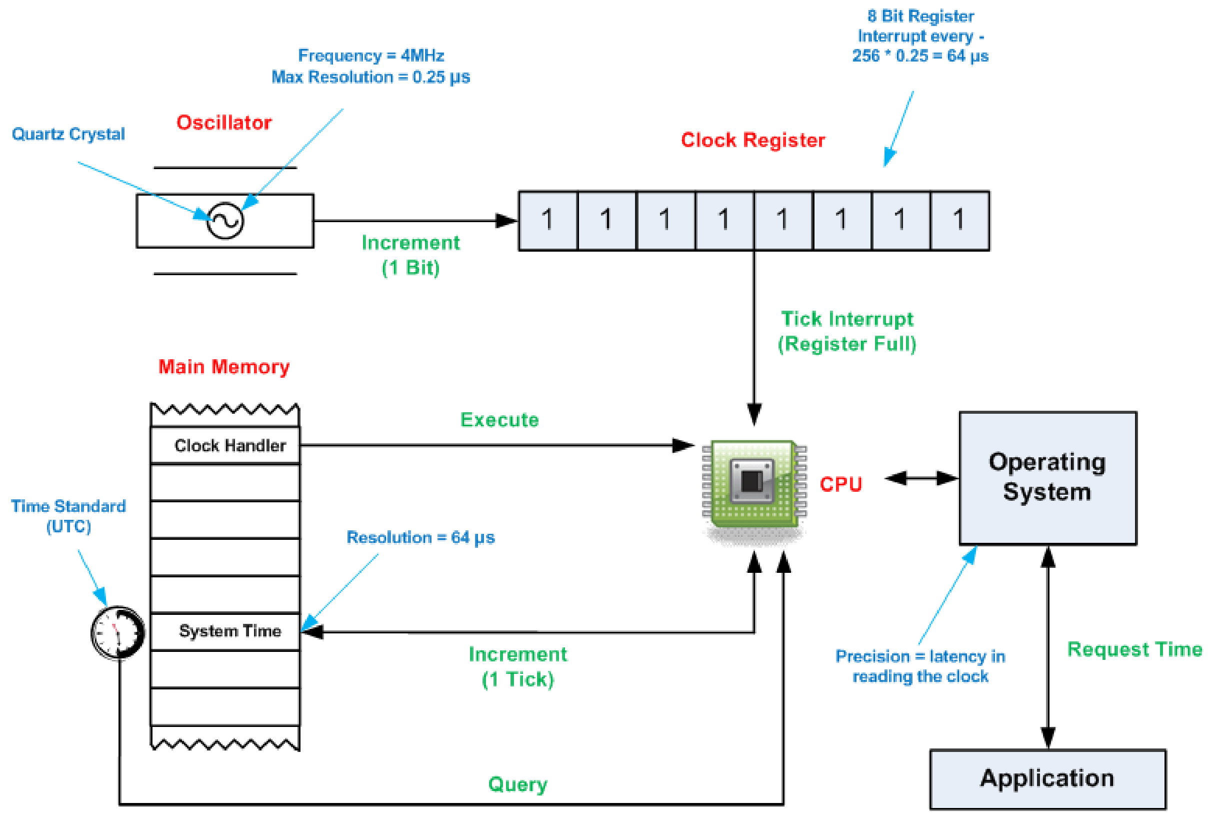
- Each user has a unique piece of hardware called a security token that generates an OTP (e.g. mobile phone or gadget with LCD)
- Inside the token is an accurate clock that has been synchronized with the clock of the verifier
- Both claimant token and verifier server calculate identical OPTs that are based on time

Time-synchronised OTP



NUI Galway
OÉ Gaillimh

Problem here: An accurate Token Clock



NUI Galway
OÉ Gaillimh

Some new Biometric Methods

- Hand geometry
  Measurement and comparison of the (unique) different physical characteristics of the hand
- Palm vein authentication
  Uses an infrared beam to penetrate the users hand as it is waved over the system; the veins within the palm of the user are returned as black lines
- Retina scan
  Provides an analysis of the capillary blood vessels located in the back of the eye
- Iris scan
  Provides an analysis of the rings, furrows and freckles in the colored ring that surrounds the pupil of the eye
- Face recognition, signature and voice analysis

NYT Article (18/01/20) about Start-Up Company Clearview AI

The New York Times

The Secretive Company That Might End Privacy as We Know It

A little-known start-up helps law enforcement match photos of unknown people to their online images — and "might lead to a dystopian future or something," a backer says.

NUI Galway
OÉ Gaillimh

Reclaim your Face

- https://reclaimyourface.eu/
- https://reclaimyourface.eu/how-to-reclaim-your-face-from-clearview-ai/

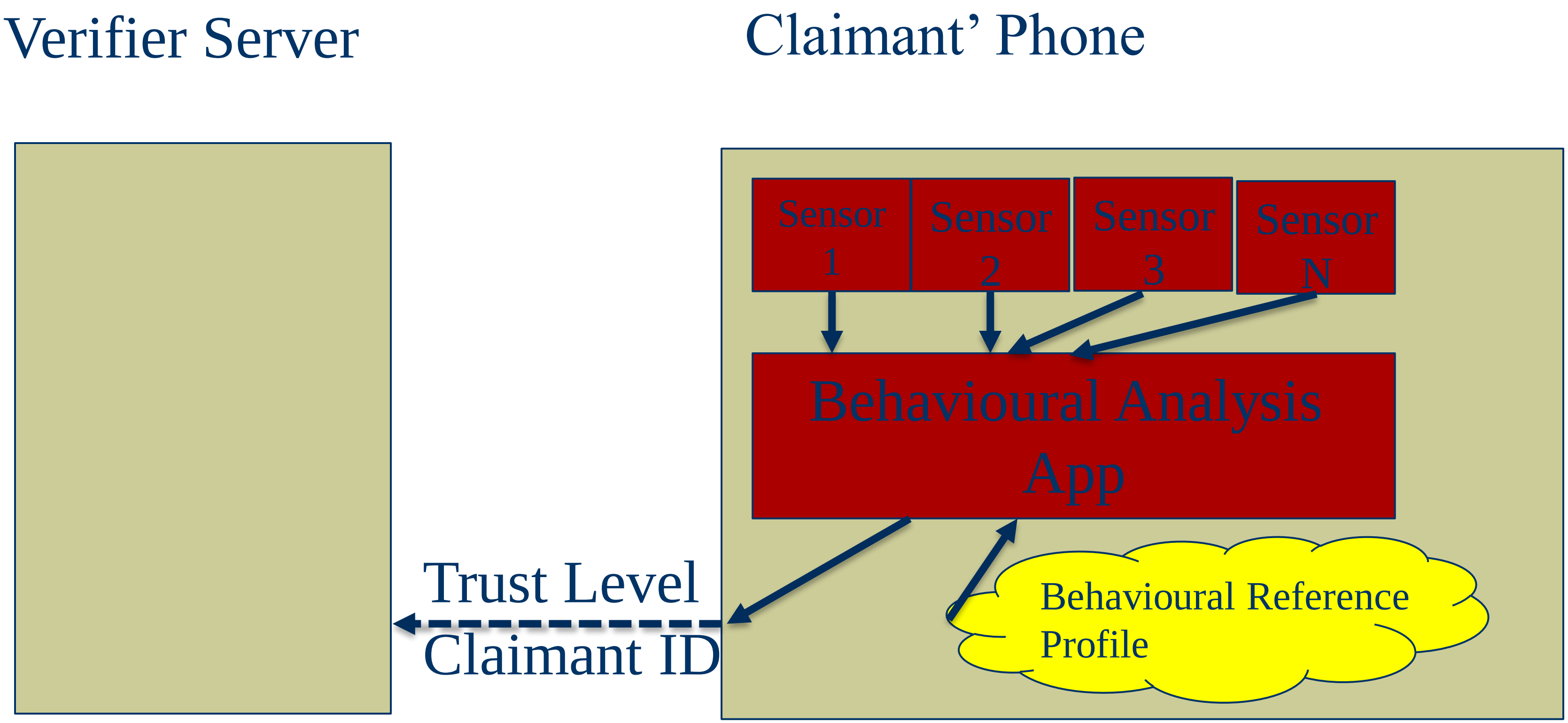
NUI Galway
OÉ Gaillimh

The Pitfalls of Biometrics

- https://www.youtube.com/watch?v=ZPG3XQhZVII
- Please watch!

NUI Galway
OÉ Gaillimh

Behavioural Biometrics



NUI Galway
OÉ Gaillimh

Multi-Factor Authentication

- This may include a combination of the following:
  - Some physical object in the possession of the user, e.g. a USB stick with a secret token, a bank card, a key, etc.
  - Some secret known to the user, such as a password, PIN, TAN, etc.
  - Some physical characteristic of the user (biometrics), such as a fingerprint, eye iris, voice, typing speed, pattern in key press intervals, etc.
  - Somewhere you are, such as connection to a specific computing network or utilizing a GPS signal to identify the location

Most common passwords according to Internet Security Company SplashData

Rank	2011[4]	2012[5]	2013[6]	2014[7]	2015[8]	2016[3]	2017[9]	2018[10]
1	password	password	123456	123456	123456	123456	123456	123456
2	123456	123456	password	password	password	password	password	password
3	12345678	12345678	12345678	12345	12345678	12345	12345678	123456789
4	qwerty	abc123	qwerty	12345678	qwerty	12345678	qwerty	12345678
5	abc123	qwerty	abc123	qwerty	12345	football	12345	12345
6	monkey	monkey	123456789	123456789	123456789	qwerty	123456789	111111
7	1234567	letmein	111111	1234	football	1234567890	letmein	1234567
8	letmein	dragon	1234567	baseball	1234	1234567	1234567	sunshine
9	trustno1	111111	iloveyou	dragon	1234567	princess	football	qwerty
10	dragon	baseball	adobe123[a]	football	baseball	1234	iloveyou	iloveyou
11	baseball	iloveyou	123123	1234567	welcome	login	admin	princess
12	111111	trustno1	admin	monkey	1234567890	welcome	welcome	admin
13	iloveyou	1234567	1234567890	letmein	abc123	solo	monkey	welcome
14	master	sunshine	letmein	abc123	111111	abc123	login	666666
15	sunshine	master	photoshop[a]	111111	1qaz2wsx	admin	abc123	abc123
16	ashley	123123	1234	mustang	dragon	121212	starwars	football
17	bailey	welcome	monkey	access	master	flower	123123	123123
18	passw0rd	shadow	shadow	shadow	monkey	passw0rd	dragon	monkey
19	shadow	ashley	sunshine	master	letmein	dragon	passw0rd	654321
20	123123	football	12345	michael	login	sunshine	master	!@#$%^&*
21	654321	jesus	password1	superman	princess	master	hello	charlie
22	superman	michael	princess	696969	qwertyuiop	hottie	freedom	aa123456
23	qazwsx	ninja	azerty	123123	solo	loveme	whatever	donald
24	michael	mustang	trustno1	batman	passw0rd	zaq1zaq1	qazwsx	password1
25	Football	password1	000000	trustno1	starwars	password1	trustno1	qwerty123

Source: Wikipedia

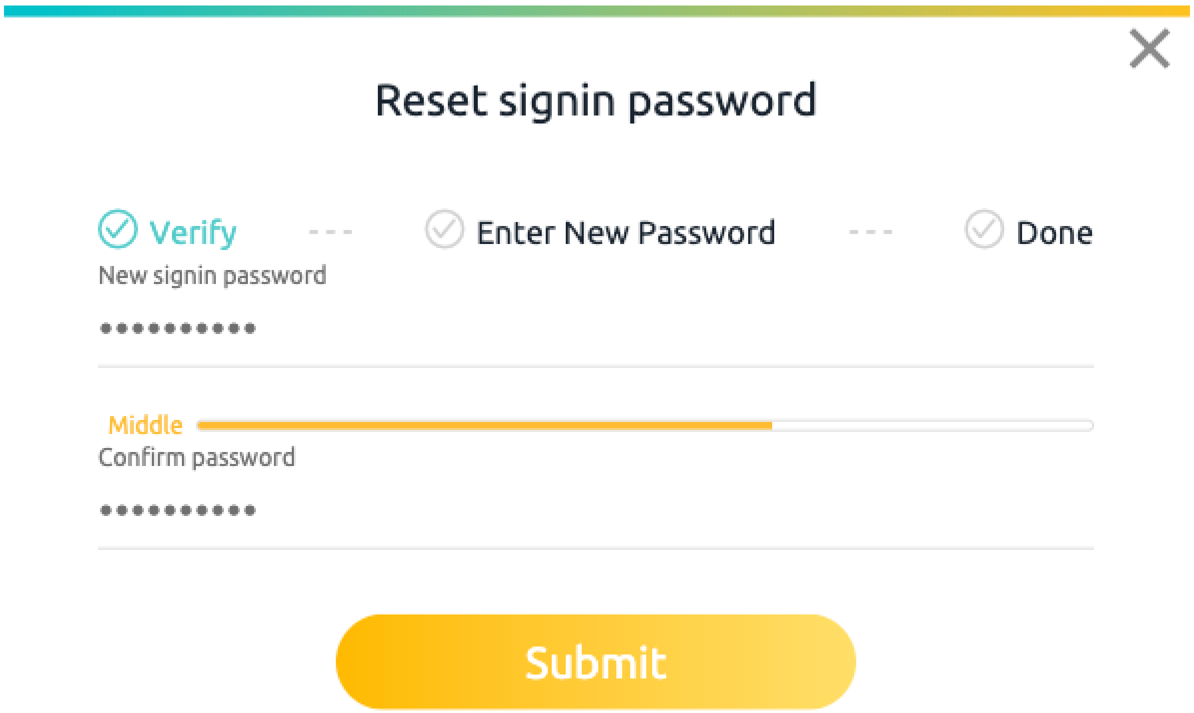
ty

NUI Galway
OÉ Gaillimh

How to enforce strong Passwords?

- Minimum length (>8 characters)
- Capital and small letters mixed
- Letters, digits, and other symbols mixed
- Don’t reuse old passwords
- **Is all the above sufficient to create strong passwords?**

Example for new Password Validation



The Guardian Headline

Trump's Twitter hacked after Dutch researcher claims he guessed password – report

Victor Gevers claimed he had access to president's account, De Volkskrant reported, but Twitter said 'we've seen no evidence'

Donald Trump holds a campaign rally in Gastonia, North Carolina, on 21 October. Photograph: Tom Brenner/Reuters

Donald Trump's Twitter account was allegedly hacked last week, after a Dutch researcher correctly guessed the president's password: "maga2020!", Dutch media reported.

NUI Galway
OÉ Gaillimh

maga2020! Who would use this Password?

- While this story is disputed by the US government, it shows the pitfalls of using readily available information for personal passwords
- BTW after the news broke, the apparent victim switched to two-factor authentication to access their Twitter account ;-)
  - Of course only until the person got banned from using Twitter :-)
- https://www.theguardian.com/us-news/2020/oct/22/trump-twitter-hacked-dutch-researcher-password

NUI Galway
OÉ Gaillimh

The Human Factor

- In 2013 a Google research project concluded that
  - most people of use “readily available” information to generate passwords
  - subsequently some educated guesses often allow to reveal them
- So what is readily available information?

Readily available Information

1. Pet names
2. A notable date, such as a wedding anniversary
3. A family member's birthday
4. Your child's name
5. Another family member's name
6. Your birthplace
7. A favourite holiday
8. Something related to your favourite sports team
9. The name of a significant other

NUI Galway
OÉ Gaillimh

Public Sources to retrieve such Information

In-Class Activity: Your Personal Password Score

- Consider:
  - all **unique** passwords you currently use
  - your personal social media footprint; analyse your own posts for any "readily available" information that you incorporated into one of your current passwords
- Consider
  - direct and indirect information
  - password fragments

In-Class Activity: Your Personal Password Score

- Direct information
  - E.g. your dog's name, e.g. password "Carly"
- Indirect information
  - E.g. a member of your favourite soccer team, for example password "Klopp" if you are a Liverpool FC fan
  - In your social media posts consider both text and images
- Password fragments
  - E.g. "!**Klopp**4ever" would qualify

NUI Galway
OÉ Gaillimh

In-Class Activity: Your Personal Password Score

1. Estimate the total number of your passwords or password fragments that can be recovered via
   - direct information
   - indirect information

   retrieved from your social media footprint

   Note that each password should only count once, i.e. it can be either recovered or not
2. Divide both numbers by the total number of unique passwords that you use at the moment, and multiply the values with 100 (to get a percentage)

Example

- Scanning my social media posts revealed that:
  - 2 password can be (fully or partially) revealed via direct information, as they contain the names of my pet rabbits mentioned in some of my posts: Leo and Enda
  - 4 password can be (fully or partially) revealed via indirect information (see Facebook post), i.e. they contain (former) LFC players Alisson, van Dijk, Gomez and Firmino
- I use a total of 10 different passwords at the moment, therefore
  - (2/10) * 100 = 20%
  - (4/10) * 100 = 40%

- In summary
  - 20% of my passwords are linked to direct information
  - 40% of my passwords are linked to indirect information
  - **Therefore, my personal password score is 60%, i.e. More than half my passwords are linked to publically available information**

In-Class Activity: Your Personal Password Score

- Please calculate / estimate your **personal password score** (0% - 100%)

Scary Statistics about the Password Reuse Problem*

- A Google survey found that at least 65% of people reuse passwords across multiple sites
- Another recent survey found that 91% of respondents claim to understand the risks of reusing passwords across multiple accounts, but 59% admitted to doing it anyway
- The average person reuses each password as many as 14 times
- 72% of individuals reuse passwords in their personal life

*Source: https://securityboulevard.com/2020/04/8-scary-statistics-about-the-password-reuse-problem/

NUI Galway OÉ Gaillimh

CT255
Introduction to Cybersecurity

Lecture 5

Human Security

- Social Engineering -

Dr. Michael Schukat, 2019-2022

Social Engineering

- The use of deception to manipulate individuals into divulging confidential or personal information that may be used for fraudulent purposes; this includes
  - Credit card details
  - PPS number
  - Bank account details
  - Login IDs and passwords

Phishing

- Attackers use emails, social media, instant messaging and SMS to trick victims into providing sensitive information or visiting malicious URL in the attempt to compromise their systems
- Study the email on the next slide. Why is it a phishing email?

Notice We have update on our Policy Update

service team <support@paypal.service.support.com>

If there are problems with how this message is displayed, click here to view it in a web browser.

Sent: Tue 10/09/2019 19:27

To: Schukat, Michael

PayPal

We need your help!

We recently update our online service for security reasons, and we need your help to give more security for your PayPal account.

What i have to do?

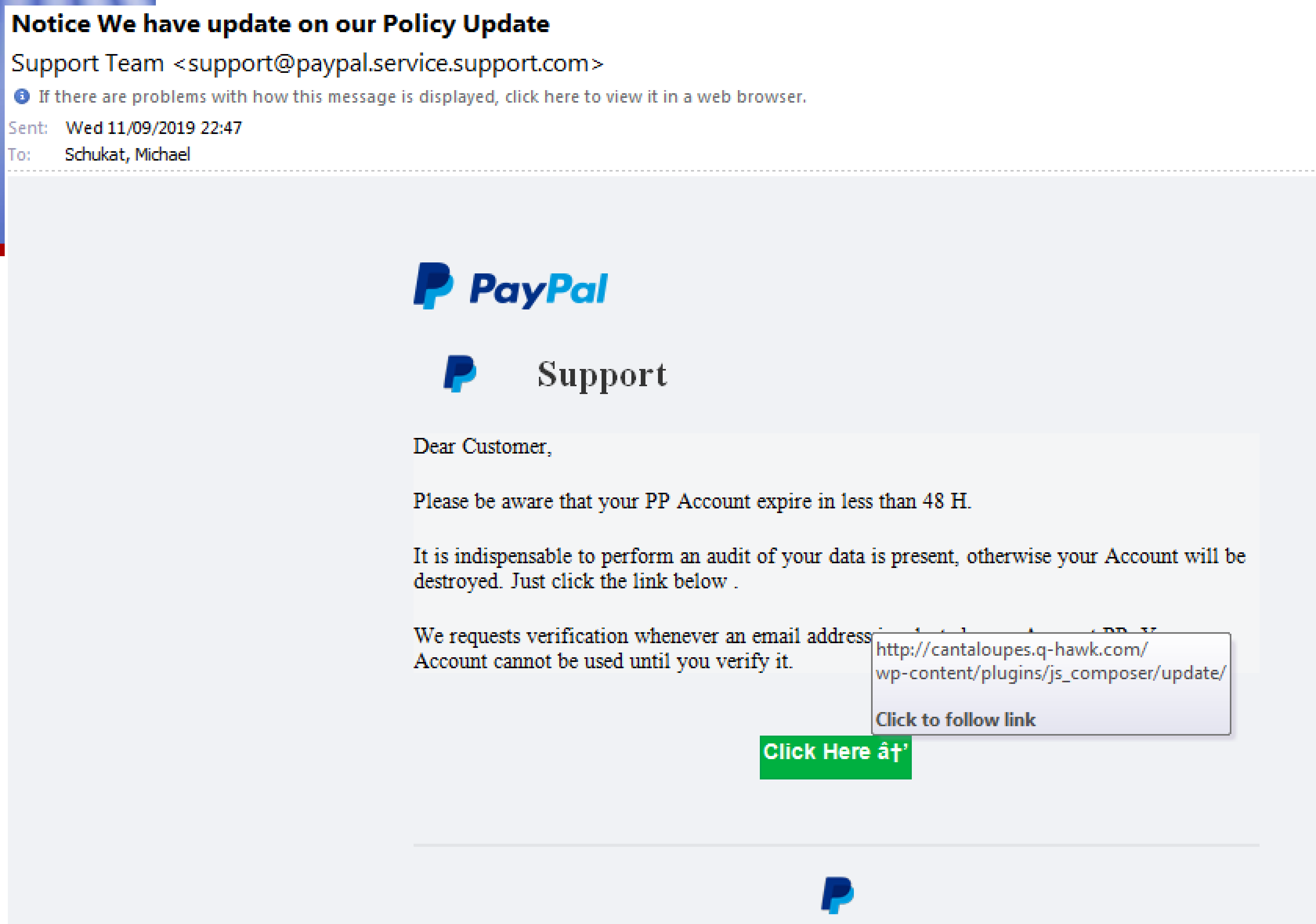
We need to reconfirm all your account information by clicking on the link bellow and follow some easy steps to confirm and secure your PayPal account.

Log in

Thanks,

Review Departmnet

PayPal Inc 2019



Notice We have update on our Policy Update

Support Team <support@paypal.service.support.com>

If there are problems with how this message is displayed, click here to view it in a web browser.

Sent: Wed 11/09/2019 22:47

To: Schukat, Michael

PayPal

Support

Dear Customer,

Please be aware that your PP Account expire in less than 48 H.

It is indispensable to perform an audit of your data is present, otherwise your Account will be destroyed. Just click the link below .

We requests verification whenever an email address
Account cannot be used until you verify it.

http://cantaloupes.q-hawk.com/wp-content/plugins/js_composer/update/

Click to follow link

Click Here â†'

Spear Phishing vs. Phishing vs. Whaling Attacks

- **Phishing** involves sending malicious emails from supposed trusted sources to as many people as possible, assuming a low response rate
- In **spear phishing** the perpetrator is disguised as a trusted individual (boss, friend, spouse)
- **Whaling** uses deceptive email messages targeting high-level decision makers within an organization, such as CEOs and other executives.
  - Such individuals have access to highly valuable information, including trade secrets and passwords to administrative company accounts

Example for Spear Phishing Email

Re: Afternoon

Professor Ciarán Ó hÓgartaigh <vice.chancell@virginmedia.com>
To

This message was sent with High importance.

Good Afternoon, Please let me know if you are unoccupied to run an errand for me? Let me know if you can.

Thank you
Professor Ciarán Ó hÓgartaigh
President of NUI Galway
National University of Ireland
Galway,
University Road,
Galway, Ireland

Sent from my iPad

Example for Spear Phishing Email Trail #1

From: Michael Madden [mailto:michaelmadden0901@gmail.com]
Sent: 01 November 2019 12:51
To: Schukat, Michael
Subject: Are you at work today

Available at the moment ?

Professor Michael Madden,

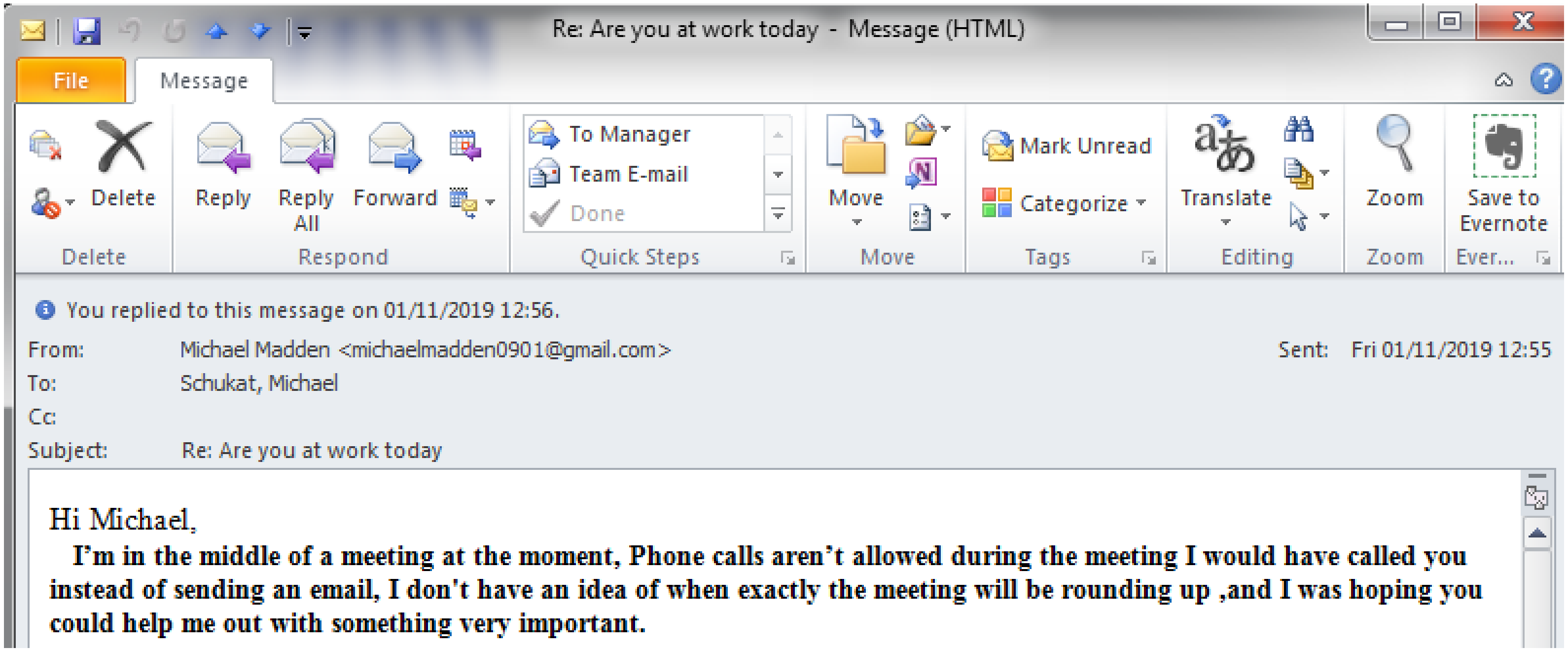
Head of school.

Example for Spear Phishing Email Trail #2

On Fri, Nov 1, 2019 at 1:53 PM Schukat, Michael <michael.schukat@nuigalway.ie> wrote:

> I am working from home, but can give you a call now.
>
> Regards,
>
> Michael

Example for Spear Phishing Email Trail #3



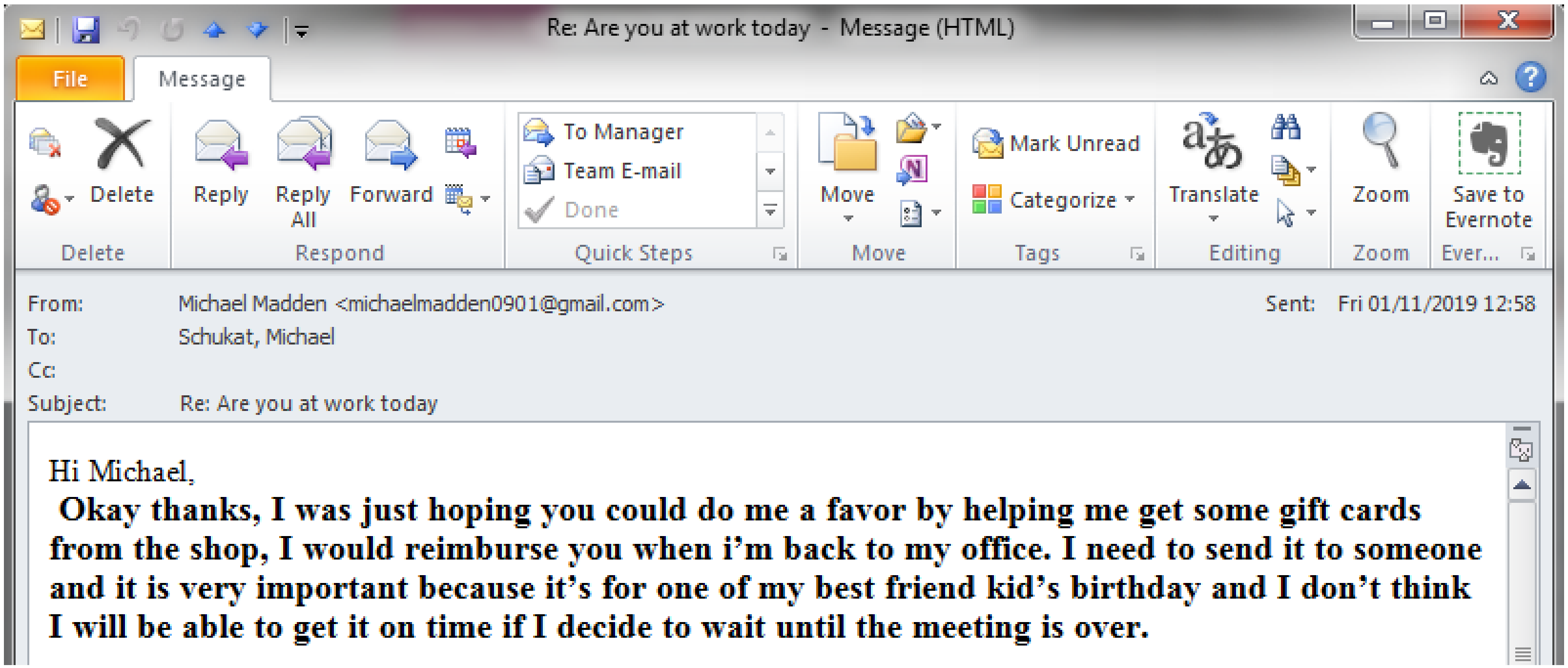
Example for Spear Phishing Email Trail #4

On Fri, Nov 1, 2019 at 1:56 PM Schukat, Michael <michael.schukat@nuigalway.ie> wrote:

> Ok, what I can I do?
>
> Regards,
>
> Michael

Ollscoil na Gaillimhe
University of Galway

Example for Spear Phishing Email Trail #5



Re: Are you at work today - Message (HTML)

From: Michael Madden <michaelmadden0901@gmail.com> Sent: Fri 01/11/2019 12:58
To: Schukat, Michael
Cc:
Subject: Re: Are you at work today

Hi Michael,

Okay thanks, I was just hoping you could do me a favor by helping me get some gift cards from the shop, I would reimburse you when i'm back to my office. I need to send it to someone and it is very important because it's for one of my best friend kid's birthday and I don't think I will be able to get it on time if I decide to wait until the meeting is over.

Example for Spear Phishing Email Trail

- Guess what happens next…

Smishing

- Smishing is short for SMS phishing and it works much the same as phishing
- Users are tricked into downloading a Trojan horse or virus onto their phones from an SMS text as opposed from an email onto their phone

Vishing

- Also called VoIP phishing
- It is the voice counterpart to phishing, e.g.
  - An email message asks the user to make a telephone call
  - Victims receive an unsolicited call
- Many different variations, see for example
  - https://www.youtube.com/watch?v=BEHl2lAuWCk
  - https://www.youtube.com/watch?v=PWVN3Rq4gzw

Alethe Denis, Winner of the Social-Engineering Competition @Defcon 2019

https://www.alethedenis.com/

FYI: Defcon

- DEF CON (also written as DEFCON, Defcon or DC) is one of the world's largest and most notable hacker conventions, held since 1993 annually in Las Vegas, Nevada

Pretexting

- Pretexting is defined as the practice of presenting oneself as someone else in order to obtain private information
- It is more than just creating a lie, in some cases it can be creating a whole new identity and then using that identity to manipulate the receipt of information
- Pretexting goes hand-in-hand with vishing

Quid Pro Quo

- Goes hand-in-hand with vishing
- Such an attack promises a service or a benefit based on the execution of a specific action
- Example:
  - A hacker attempts to contact via phone the employees of the target organisation then offers them some kind of upgrade or software installation
    They might request victims to facilitate the operation by disabling the AV software temporarily to install a malicious application

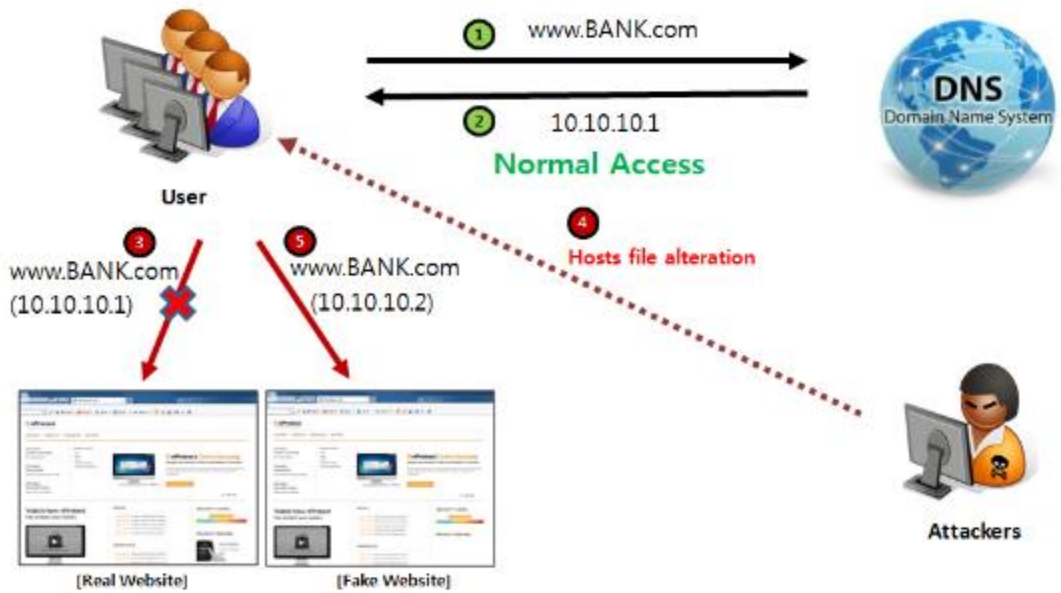
Watering Hole

- A watering hole attack consists of injecting malicious code into public Web pages of a site that the target uses to visit
  - https://www.youtube.com/watch?v=20jp-teI5no
- The attackers typically compromise websites within a specific sector that are typically visited by specific individuals of interest for the attacks
- Example: Blackboard ← → students

Pharming

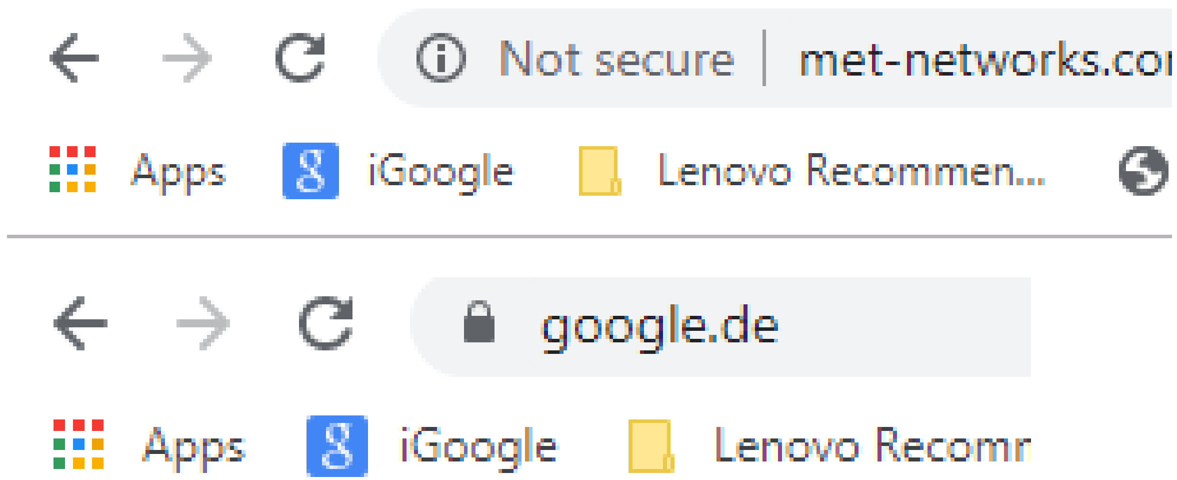
- Pharming scams redirect users to copies of popular websites where personal data like user names, passwords and financial information can be ‘farmed’ and collected for fraudulent use

Pharming via DNS Poisoning / DNS Spoofing



Domain Spoofing Pharming and how to detect it

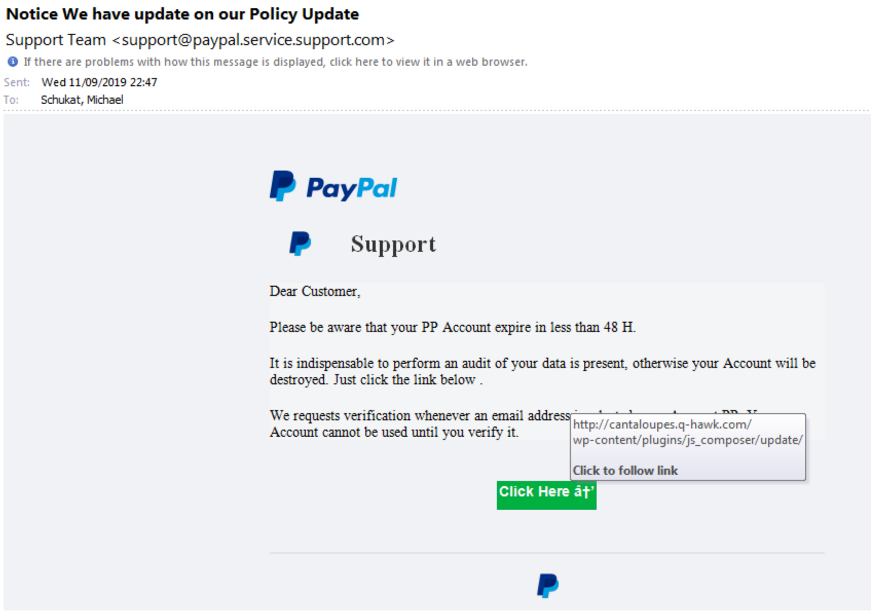
- Used domain spoofing (in which the domain appears authentic)



bad!

good!

Simple Pharming and how to detect it



OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

Baiting

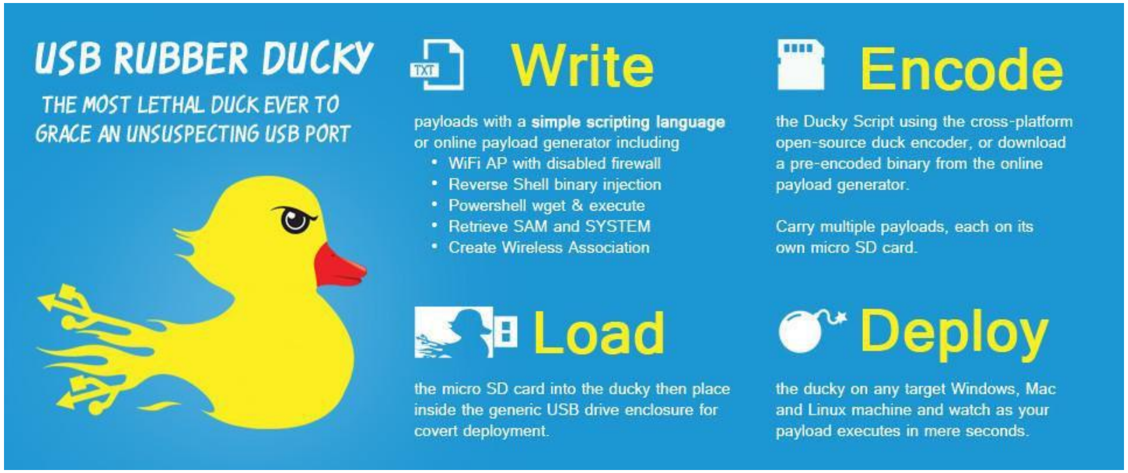
- Baiting that exploits the human's curiosity
- Example USB drop attacks
  - Leave infected USBs tokens in the parking lot of a target organization and wait for internal personnel insert them in the corporate PC
  - See https://www.redteamsecure.com/usb-drop-attacks-the-danger-of-lost-and-found-thumb-drives/
- Funny: https://www.youtube.com/watch?v=GQMsOH-yDBU

USB Baiting

- USB baiting exploits the human's curiosity
  - You find a memory stick and want to know what's stored in it
- Example (USB drop attack): Leave infected USBs tokens in the parking lot of a target organization and wait for personnel inserting them in a corporate PC; three things may happen:
    - The user clicks on one of the files on the drive, which unleashes a malicious code that automatically activates upon viewing and can download further malware from the Internet
    - Alternatively the user is directed to a phishing website
    - HID (Human Interface Device) spoofing – see next slide

USB Baiting and HID spoofing

- The USB stick will trick the computer into thinking a keyboard is attached. When plugged into a computer, it injects keystrokes to command the computer to give a hacker remote access to the victim's computer
- USB Rubber Ducky – the most lethal duck ever!
- https://www.youtube.com/watch?v=sbKN8FhGnqg



OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

Tailgating aka Piggybacking

- Attacker seeking physical entry to a restricted area which lacks the proper authentication
- Example:
  - An attacker can walk in behind a person who is authorised to access the area
  - In a typical attack scenario, a person impersonates a delivery driver or a caretaker who is packed with parcels and waits when an employee opens their door

CT255
Introduction to Cybersecurity

Lecture 5

Human Security

- Social Engineering -

Dr. Michael Schukat, 2019-2022

Social Engineering

- The use of deception to manipulate individuals into divulging confidential or personal information that may be used for fraudulent purposes; this includes
  - Credit card details
  - PPS number
  - Bank account details
  - Login IDs and passwords

Phishing

- Attackers use emails, social media, instant messaging and SMS to trick victims into providing sensitive information or visiting malicious URL in the attempt to compromise their systems
- Study the email on the next slide. Why is it a phishing email?

Notice We have update on our Policy Update

service team <support@paypal.service.support.com>

If there are problems with how this message is displayed, click here to view it in a web browser.

Sent: Tue 10/09/2019 19:27

To: Schukat, Michael

PayPal

We need your help!

We recently update our online service for security reasons, and we need your help to give more security for your PayPal account.

What i have to do?

We need to reconfirm all your account information by clicking on the link bellow and follow some easy steps to confirm and secure your PayPal account.

Log in

Thanks,

Review Departmnet

PayPal Inc 2019

Notice We have update on our Policy Update

Support Team <support@paypal.service.support.com>

If there are problems with how this message is displayed, click here to view it in a web browser.

Sent: Wed 11/09/2019 22:47

To: Schukat, Michael

PayPal

Support

Dear Customer,

Please be aware that your PP Account expire in less than 48 H.

It is indispensable to perform an audit of your data is present, otherwise your Account will be destroyed. Just click the link below .

We requests verification whenever an email address
Account cannot be used until you verify it.

http://cantaloupes.q-hawk.com/wp-content/plugins/js_composer/update/

Click to follow link

Click Here â†'

Spear Phishing vs. Phishing vs. Whaling Attacks

- **Phishing** involves sending malicious emails from supposed trusted sources to as many people as possible, assuming a low response rate
- In **spear phishing** the perpetrator is disguised as a trusted individual (boss, friend, spouse)
- **Whaling** uses deceptive email messages targeting high-level decision makers within an organization, such as CEOs and other executives.
  - Such individuals have access to highly valuable information, including trade secrets and passwords to administrative company accounts

Example for Spear Phishing Email

Re: Afternoon

PC

Professor Ciarán Ó hÓgartaigh <vice.chancell@virginmedia.com>

To

ⓘ This message was sent with High importance.

Good Afternoon, Please let me know if you are unoccupied to run an errand for me? Let me know if you can.

Thank you

Professor Ciarán Ó hÓgartaigh
President of NUI Galway
National University of Ireland
Galway,
University Road,
Galway, Ireland

Sent from my iPad

Example for Spear Phishing Email Trail #1

From: Michael Madden [mailto:michaelmadden0901@gmail.com]
Sent: 01 November 2019 12:51
To: Schukat, Michael
Subject: Are you at work today

Available at the moment ?

Professor Michael Madden,

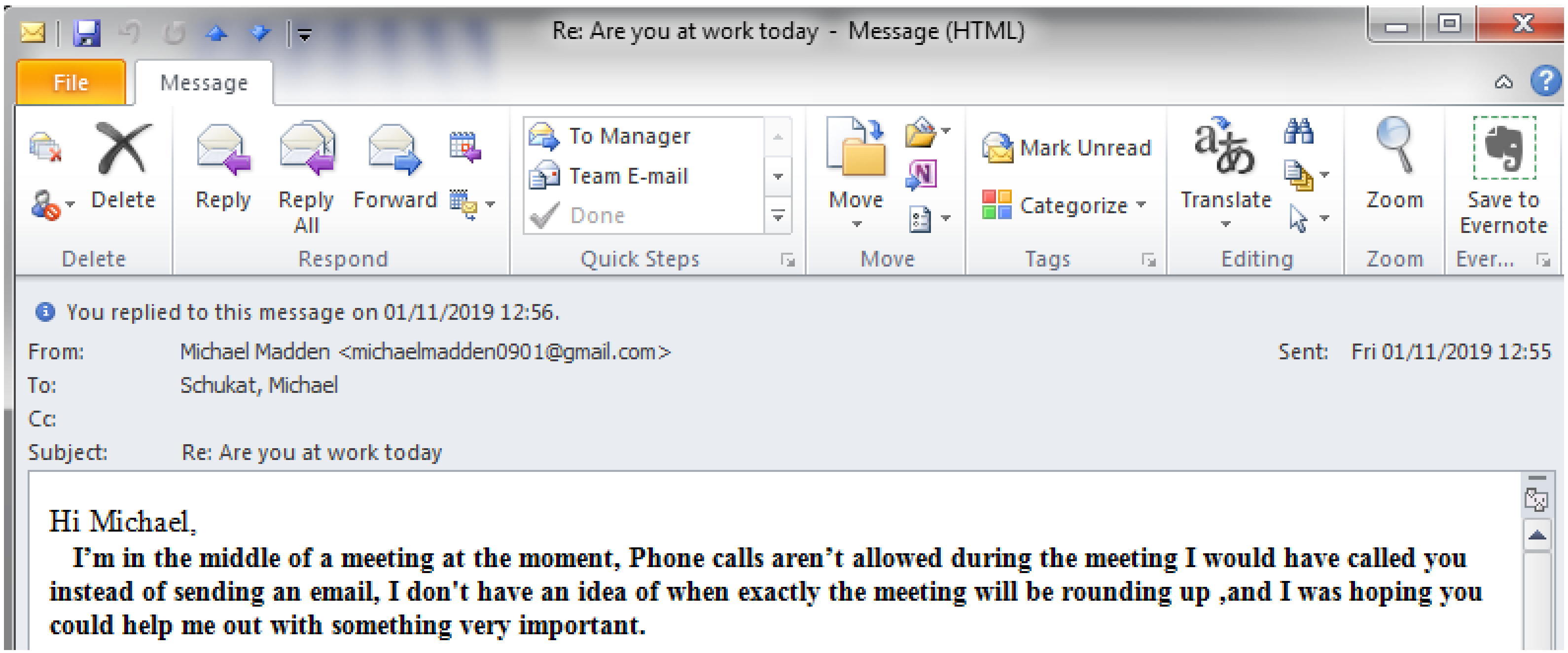
Head of school.

Example for Spear Phishing Email Trail #2

On Fri, Nov 1, 2019 at 1:53 PM Schukat, Michael <michael.schukat@nuigalway.ie> wrote:

> I am working from home, but can give you a call now.
>
> Regards,
>
> Michael

Example for Spear Phishing Email Trail #3



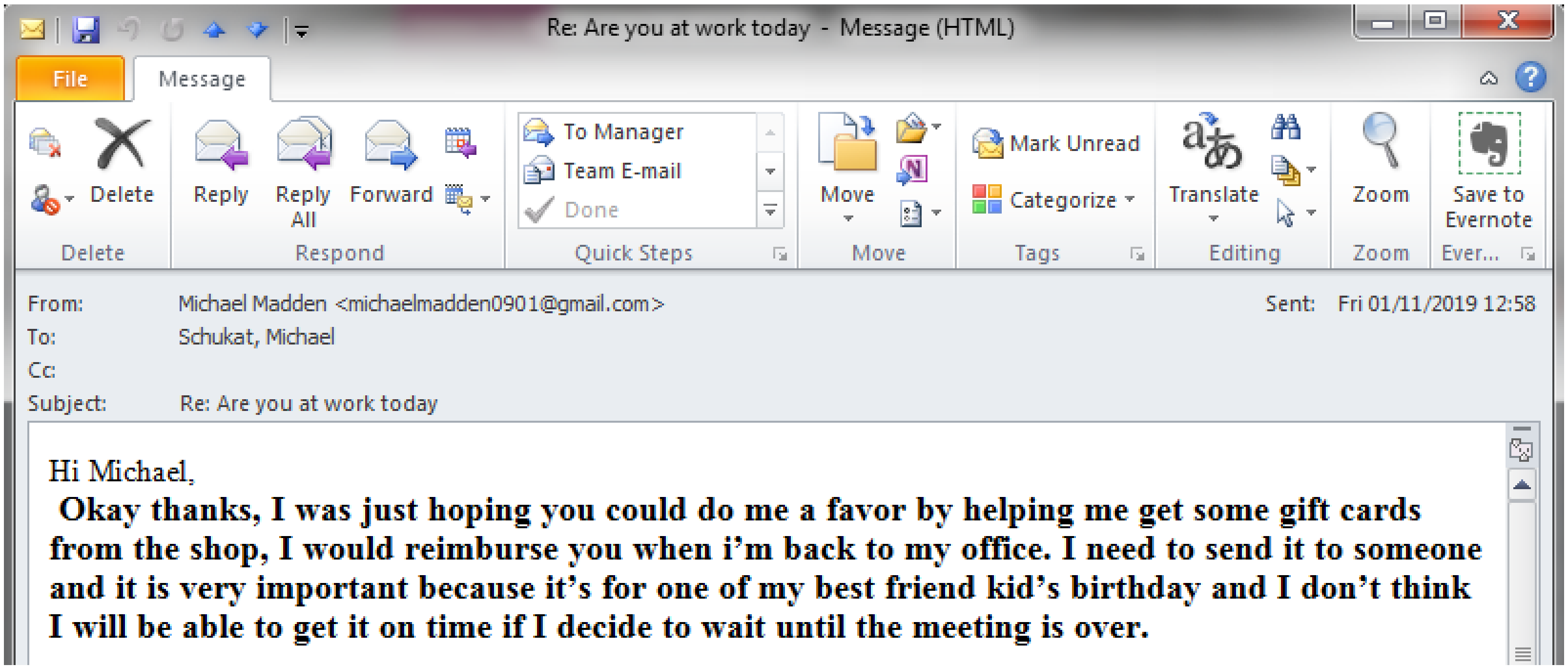
OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

Example for Spear Phishing Email Trail #4

On Fri, Nov 1, 2019 at 1:56 PM Schukat, Michael <michael.schukat@nuigalway.ie> wrote:

> Ok, what I can I do?
>
> Regards,
>
> Michael

Example for Spear Phishing Email Trail #5



Example for Spear Phishing Email Trail

- Guess what happens next…

Smishing

- Smishing is short for SMS phishing and it works much the same as phishing
- Users are tricked into downloading a Trojan horse or virus onto their phones from an SMS text as opposed from an email onto their phone

Vishing

- Also called VoIP phishing
- It is the voice counterpart to phishing, e.g.
  - An email message asks the user to make a telephone call
  - Victims receive an unsolicited call
- Many different variations, see for example
  - https://www.youtube.com/watch?v=BEHl2lAuWCk
  - https://www.youtube.com/watch?v=PWVN3Rq4gzw

Alethe Denis, Winner of the Social-Engineering Competition @Defcon 2019

https://www.alethedenis.com/

FYI: Defcon

- DEF CON (also written as DEFCON, Defcon or DC) is one of the world's largest and most notable hacker conventions, held since 1993 annually in Las Vegas, Nevada

Pretexting

- Pretexting is defined as the practice of presenting oneself as someone else in order to obtain private information
- It is more than just creating a lie, in some cases it can be creating a whole new identity and then using that identity to manipulate the receipt of information
- Pretexting goes hand-in-hand with vishing

Quid Pro Quo

- Goes hand-in-hand with vishing
- Such an attack promises a service or a benefit based on the execution of a specific action
- Example:
  - A hacker attempts to contact via phone the employees of the target organisation then offers them some kind of upgrade or software installation
    They might request victims to facilitate the operation by disabling the AV software temporarily to install a malicious application

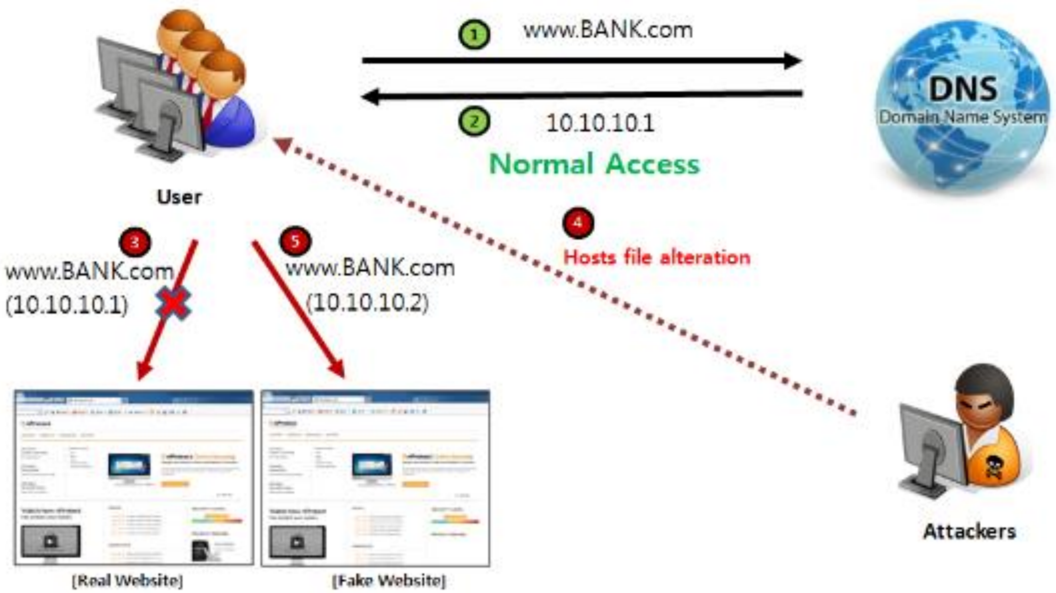
Watering Hole

- A watering hole attack consists of injecting malicious code into public Web pages of a site that the target uses to visit
  - https://www.youtube.com/watch?v=20jp-teI5no
- The attackers typically compromise websites within a specific sector that are typically visited by specific individuals of interest for the attacks
- Example: Blackboard ← → students

Pharming

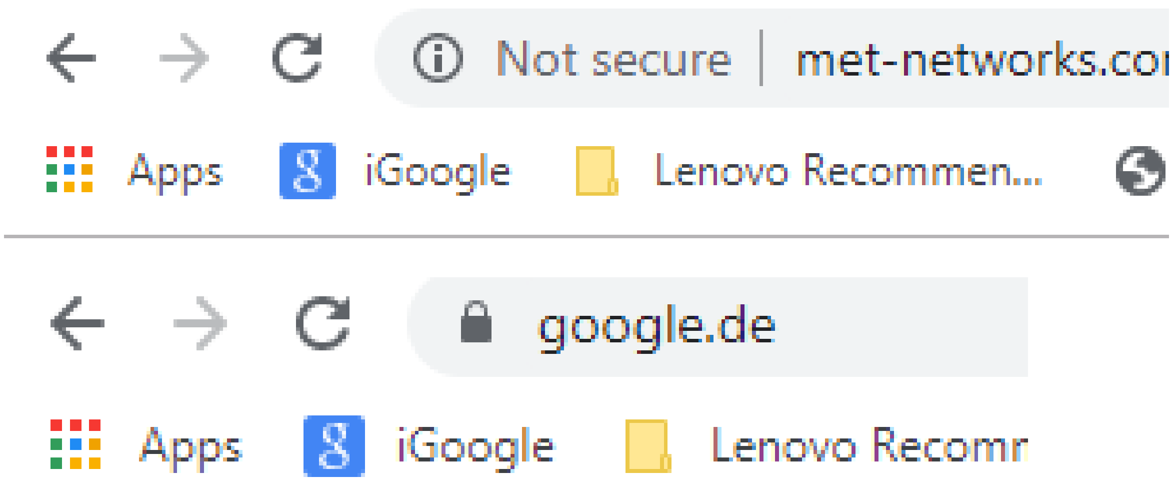
- Pharming scams redirect users to copies of popular websites where personal data like user names, passwords and financial information can be ‘farmed’ and collected for fraudulent use

Pharming via DNS Poisoning / DNS Spoofing



Domain Spoofing Pharming and how to detect it

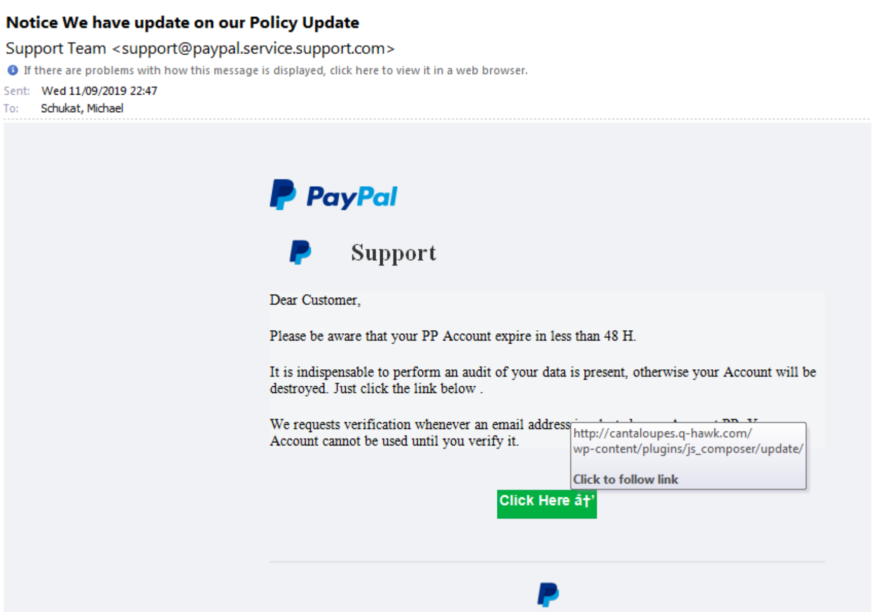
- Used domain spoofing (in which the domain appears authentic)



bad!

good!

Simple Pharming and how to detect it



Ollscoil na Gaillimhe
University of Galway

Baiting

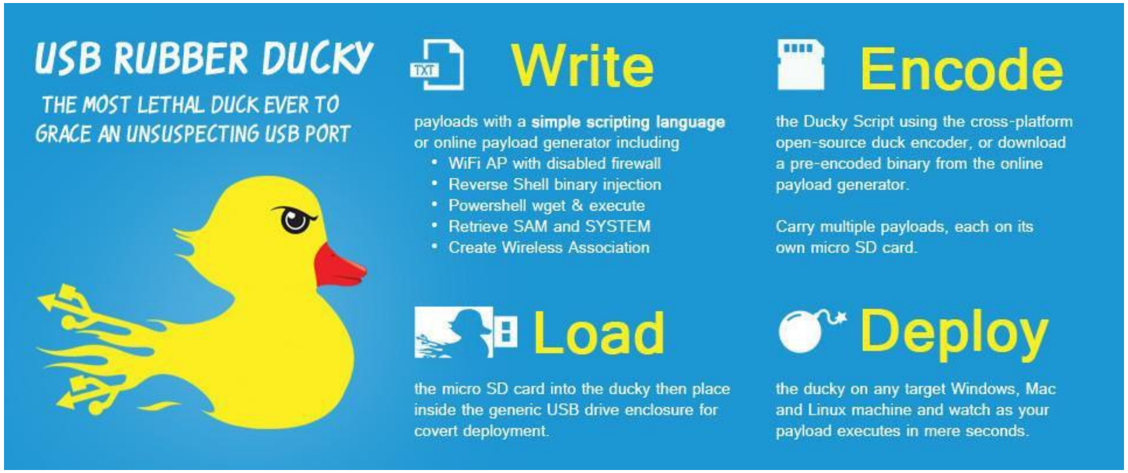
- Baiting that exploits the human's curiosity
- Example USB drop attacks
  - Leave infected USBs tokens in the parking lot of a target organization and wait for internal personnel insert them in the corporate PC
  - See https://www.redteamsecure.com/usb-drop-attacks-the-danger-of-lost-and-found-thumb-drives/
- Funny: https://www.youtube.com/watch?v=GQMsOH-yDBU

USB Baiting

- USB baiting exploits the human's curiosity
  - You find a memory stick and want to know what's stored in it
- Example (USB drop attack): Leave infected USBs tokens in the parking lot of a target organization and wait for personnel inserting them in a corporate PC; three things may happen:
  - The user clicks on one of the files on the drive, which unleashes a malicious code that automatically activates upon viewing and can download further malware from the Internet
  - Alternatively the user is directed to a phishing website
  - HID (Human Interface Device) spoofing – see next slide

USB Baiting and HID spoofing

- The USB stick will trick the computer into thinking a keyboard is attached. When plugged into a computer, it injects keystrokes to command the computer to give a hacker remote access to the victim's computer
- USB Rubber Ducky – the most lethal duck ever!
- https://www.youtube.com/watch?v=sbKN8FhGnqg



Tailgating aka Piggybacking

- Attacker seeking physical entry to a restricted area which lacks the proper authentication
- Example:
  - An attacker can walk in behind a person who is authorised to access the area
  - In a typical attack scenario, a person impersonates a delivery driver or a caretaker who is packed with parcels and waits when an employee opens their door

CT255
INTRODUCTION TO CYBERSECURITY
DIFFIE-HELLMAN KEY EXCHANGE

Dr. Michael Schukat

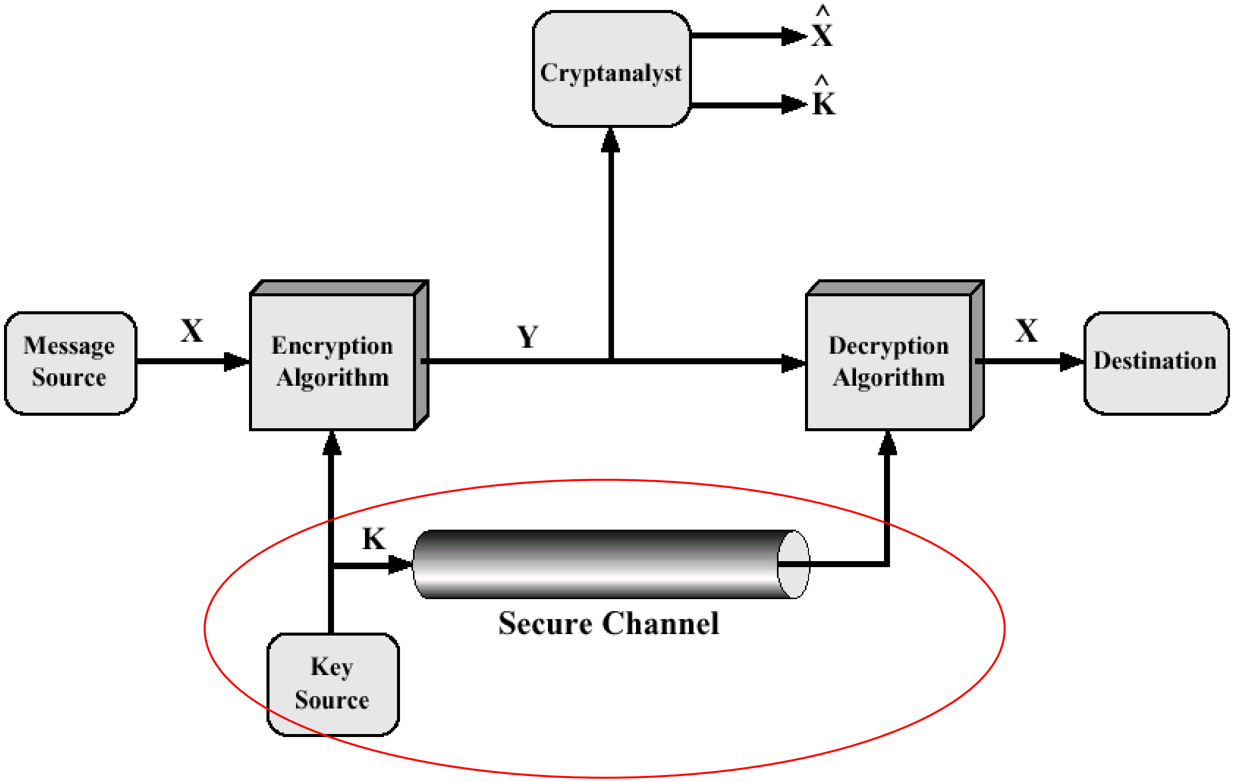
OÉ Gaillimh
NUI Galway

Lecture Content

- Diffie-Hellman Key exchange
- Man-in-the-Middle (MitM) attacks
- Optimisation techniques for public key encryption

Model of Conventional Cryptosystem

Problem: How to securely circulate a secret key?

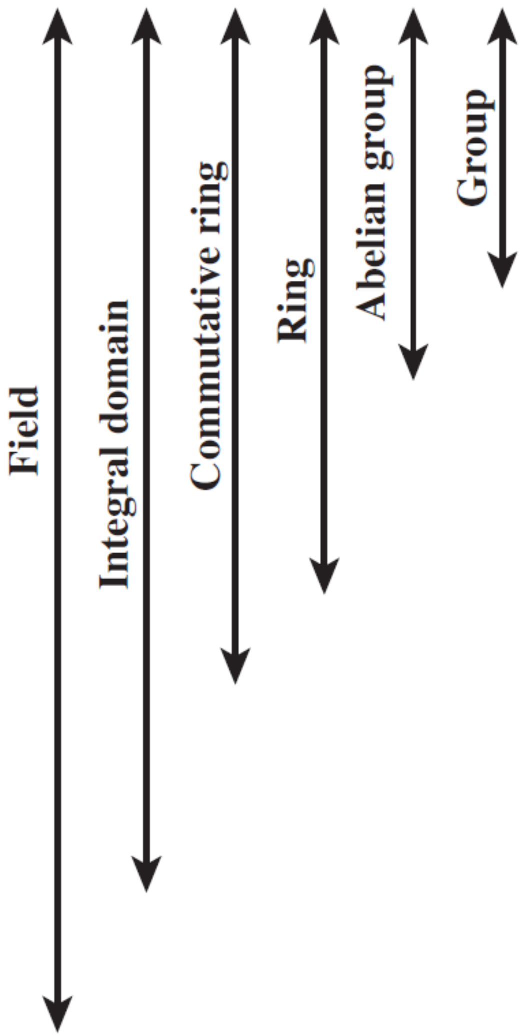


$$Y = E_K(X),\ X = E_K^{-1}(Y)$$

Groups, Rings and Fields (Wikipedia)

- In mathematics,
  - a **group** is a set equipped with a binary operation that is associative, has an identity element, and is such that every element has an inverse, e.g. (Z, +)
  - a **ring** is a set equipped with two binary operations satisfying properties analogous to those of addition and multiplication of integers, e.g. (Z, +, *)
  - a **field** is a set on which addition, subtraction, multiplication, and division are defined and behave as the corresponding operations on rational and real numbers do

Properties of Groups, Rings and Fields (Stallings)



(A1) Closure under addition:	If a and b belong to S, then $a + b$ is also in S
(A2) Associativity of addition:	$a + (b + c) = (a + b) + c$ for all a, b, c in S
(A3) Additive identity:	There is an element 0 in R such that $a + 0 = 0 + a = a$ for all a in S
(A4) Additive inverse:	For each a in S there is an element $-a$ in S such that $a + (-a) = (-a) + a = 0$
(A5) Commutativity of addition:	$a + b = b + a$ for all a, b in S
(M1) Closure under multiplication:	If a and b belong to S, then ab is also in S
(M2) Associativity of multiplication:	$a(bc) = (ab)c$ for all a, b, c in S
(M3) Distributive laws:	$a(b + c) = ab + ac$ for all a, b, c in S $(a + b)c = ac + bc$ for all a, b, c in S
(M4) Commutativity of multiplication:	$ab = ba$ for all a, b in S
(M5) Multiplicative identity:	There is an element 1 in S such that $a1 = 1a = a$ for all a in S
(M6) No zero divisors:	If a, b in S and $ab = 0$, then either $a = 0$ or $b = 0$
(M7) Multiplicative inverse:	If a belongs to S and $a \neq 0$, there is an element a^{-1} in S such that $aa^{-1} = a^{-1}a = 1$

Modular Arithmetic

- In mathematics, modular arithmetic is a system of arithmetic for integers, where numbers wrap around when reaching a certain value n, called the modulus
  - Recall modulus operator "%" in C and other languages, i.e. "division with rest" with rest being the modulus
  - Example: 75 / 6 = 12 remainder 3 ➔ 75 % 6 = 3
- The ring of integers modulo n, denoted Z/nZ or Z/n
- Z/nZ is defined for n > 0 as: $\mathbb{Z}/n\mathbb{Z} = \{\overline{a}_n \mid a \in \mathbb{Z}\} = \left\{\overline{0}_n, \overline{1}_n, \overline{2}_n, \ldots, \overline{n-1}_n\right\}$
- With:
  - $\overline{a}_n + \overline{b}_n = \overline{(a+b)}_n$
  - $\overline{a}_n - \overline{b}_n = \overline{(a-b)}_n$
  - $\overline{a}_n \overline{b}_n = \overline{(ab)}_n.$

Example: Normal Multiplication

*	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	2	4	6	8	10	12	14	16
3	0	3	6	9	12	15	18	21	24
4	0	4	8	12	16	20	24	28	32
5	0	5	10	15	20	25	30	35	40
6	0	6	12	18	24	30	36	42	48
7	0	7	14	21	28	35	42	49	56
8	0	8	16	24	32	40	48	56	64

Example: Multiplication Z/9Z

Mx3

*	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	2	4	6	8	1	3	5	7
3	0	3	6	0	3	6	0	3	6
4	0	4	8	3	7	2	6	1	5
5	0	5	1	6	2	7	3	8	4
6	0	6	3	0	6	3	0	6	3
7	0	7	5	3	1	8	6	4	2
8	0	8	7	6	5	4	3	2	1

Diffie-Hellman Key Exchange

- **Diffie-Hellman provides secure key exchange between two partners**
  - The negotiated key is subsequently used for private key encryption / authentication
- It uses the **multiplicative group of integers modulo n $(Z/nZ)^x$**
- It is based on the difficulty of computing discrete logarithms over such groups, e.g.

$$6^3 \bmod 17 = 216 \bmod 17 = 12 \qquad \text{(easy)}$$

$$12 = 6^y \bmod 17? \qquad \text{(difficult)}$$

- It uses modulo n ("division with rest") operation.
- The core equation for the key exchange is

$$K = (A)^B \bmod q$$

Diffie-Hellman: Global Public Elements

- Select prime number q and positive integer a, whereby $a < q$ and a is a **primitive root** of q.

- Definition: a is a primitive root of q, if numbers
  $a \bmod q,\ a^2 \bmod q,\ \cdots\ a^{(q-1)} \bmod q$
  are distinct integer values between 1 and $(q - 1)$ in some permutation, i.e. elements of $(Z/qZ)^x$

M

- Example: $a = 3$ is a primitive root of $(Z/5Z)^x$, $a = 4$ is not:

$3^1 = 3 = 0 * 5 + \mathbf{3}$	$4^1 = 4 = 0 * 5 + \mathbf{4}$
$3^2 = 9 = 1 * 5 + \mathbf{4}$	$4^2 = 16 = 3 * 5 + \mathbf{1}$
$3^3 = 27 = 5 * 5 + \mathbf{2}$	$4^3 = 64 = 12 * 5 + \mathbf{4}$
$3^4 = 81 = 16 * 5 + \mathbf{1}$	$4^4 = 256 = 51 * 5 + \mathbf{1}$

Generation of Secret-Key: Part 1

- Both users share a (public) prime number q and primitive root a
- User A:
  - Select secret number XA with XA < q
  - Calculate public value $YA = a^{XA} \bmod q$ (← difficult to reverse)
  - YA is send to user B
- User B:
  - Select secret number XB with XB < q
  - Calculate public value $YB = a^{XB} \bmod q$ (← difficult to reverse)
  - YB is send to user A

Generation of Secret-Key: Part 2

- User A:
  - User A owns X_A and receives Y_B
  - Generate secret key: $K = (Y_B)^{X_A} \bmod q$
- User B:
  - User B owns X_B and receives Y_A
  - Generate secret key: $K = (Y_A)^{X_B} \bmod q$
- **Both keys are identical!**

Generation of Secret-Key: Part 2

$$K = (YB)^{XA} \bmod q$$

$$= (a^{XB} \bmod q)^{XA} \bmod q$$

$$= (a^{XB})^{XA} \bmod q$$

$$= a^{XB\ XA} \bmod q = a^{XA\ XB} \bmod q$$

$$= (a^{XA})^{XB} \bmod q$$

$$= (a^{XA} \bmod q)^{XB} \bmod q$$

$$= (YA)^{XB} \bmod q$$

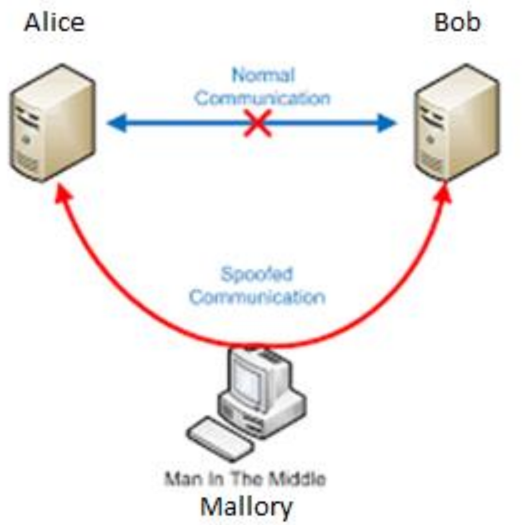
Example for Diffie-Hellman

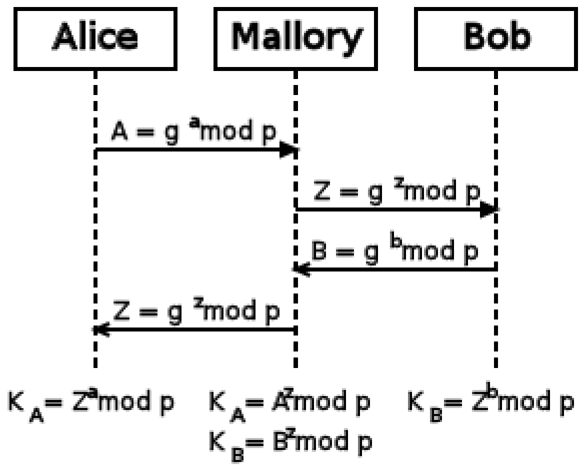
- Let q = 5 and a = 3;
- XA = 2, therefore YA = $a^{XA} \bmod 5 = 4$
- XB = 3, therefore YB = $a^{XB} \bmod 5 = 2$
- User A: $K = (YB)^{XA} \bmod q = 2^2 \bmod 5 = 4$
- User B: $K = (YA)^{XB} \bmod q = 4^3 \bmod 5 = 4$

Diffie-Hellman in Practice

- The algorithm is used in tandem with a variety of secure network protocols
  - Provision of secure end-to-end connection
  - No endpoint authentication though!
    - You can't validate who you are talking to
  - Modulus p typically has a minimum length of 1024 bits

DH and Man-in-the-Middle (MitM) Attacks





- Mallory is a MitM attacker and performs message interception and message fabrication
- Mallory establishes two individual (secure) connections with Alice and Bob
- Both Alice and Bob are unaware of Mallory's existence (as there is no authentication)

In-Class Activity: Diffie-Hellman MitM Attack

- Let q = 5 and a = 3;
- $X_{Alice} = 2$, therefore $Y_{Alice} = a^{XAlice} \bmod 5 = 4$
- $X_{Bob} = 3$, therefore $Y_{Bob} = a^{XBob} \bmod 5 = 2$
- $X_{Malory} = 1$, therefore $Y_{Malory} = a^{XMalory} \bmod 5 = 3$
- What session keys between
  - Alice and Malory
  - Malory and Bob

  are generated?
- Note: User A's key $K = (YB)^{XA} \bmod q$
- Note: User B's key $K = (YA)^{XB} \bmod q$

Mx2

Solution

- Alice sends “4” to Bob, but this message is intercepted by Malory
- Bob sends “2” to Alice, but this message is intercepted by Malory
- Malory sends “3” to both parties, claiming to be either Bob or Alice
- Alice receives “3” and calculates K as follow: $K = 3^2 \bmod 5 = 4$
  - Malory calculates $4^1 \bmod 5 = 4$
- Bob receives “3” and calculates K as follow: $K = 3^3 \bmod 5 = 2$
  - Malory calculates $2^1 \bmod 5 = 2$
- Alice and Bob think they just mutually agreed on a shared secret key
- They have no idea that Malory is a MitM and can read, manipulate and fabricate messages between both sides

Computational Aspects of Diffie-Hellman

- Assume you have to evaluate the expression $C = 503^{23} \bmod 899$ as part of the DH algorithm
- $503^{23} = 1.36792931379540842325043971010 6 \times 10^{62}$ cannot be properly represented using an ordinary integer or floating point variable!
- In order to solve this problem the exponentiation must be broken down into smaller steps, e.g.
  - $503^{23} \bmod 899 = ((503^{6} \bmod 899) \times (503^{6} \bmod 899) \times (503^{6} \bmod 899) \times (503^{5} \bmod 899)) \bmod 899$
  - $503^{6} \bmod 899 = ((503^{3} \bmod 899) \times (503^{3} \bmod 899)) \bmod 899$
  - $503^{5} \bmod 899 = ((503^{3} \bmod 899) \times (503^{2} \bmod 899)) \bmod 899$
  - $503^{3} \bmod 899 = ((503^{2} \bmod 899) \times 503) \bmod 899$

Computational Aspects of Diffie-Hellman

- or even iteratively:
  $503^{23} \bmod 899 =$
  $(((((((503^2 \bmod 899) \times 503) \bmod 899) \times 503) \bmod 899) \times \cdots \times 503) \bmod 899$

- This expression consists of 22 nested multiplications and 22 nested modulus operations and can be easily calculated by using a loop

CT255
Introduction to Cyber-Security

Lecture 8

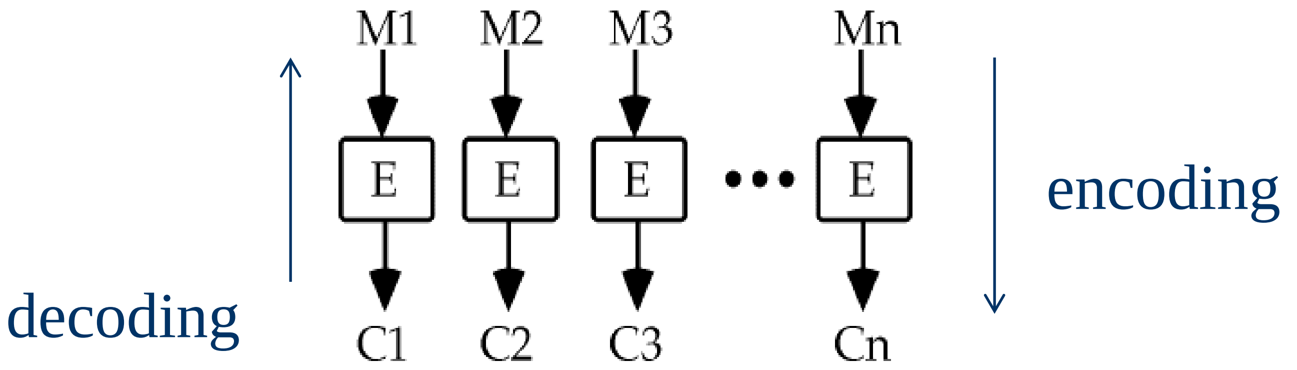
Block Ciphers and Stream Ciphers

Dr. Michael Schukat, 2019-2022

BLOCK CIPHERS

Encryption Algorithms based on Block Ciphers

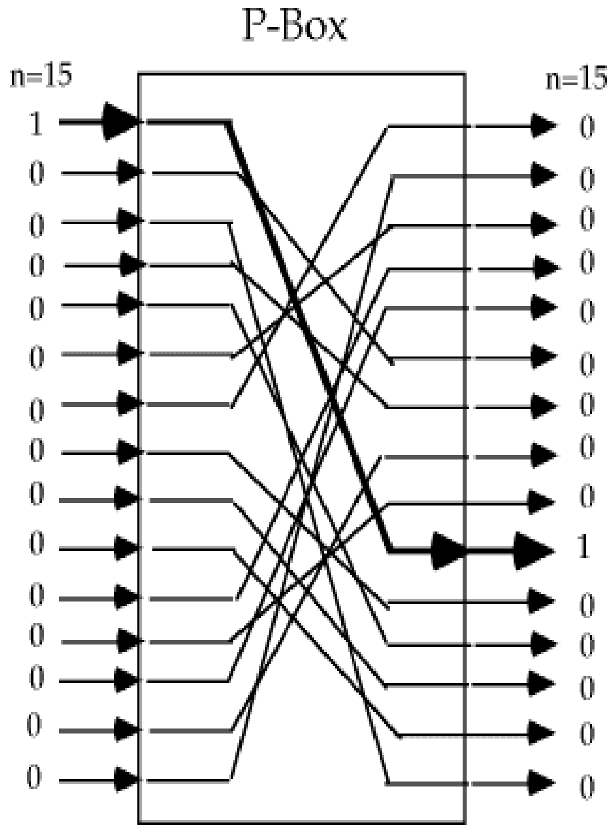
- In a block cipher the message is broken into blocks M1, M2, etc. of K bits length, each of which is then encrypted



  - Most ciphers we saw before process blocks of just one character
- Claude Shannon suggested to use the two primitive cryptographic operations as building blocks for such ciphers:
  - substitution
  - permutation

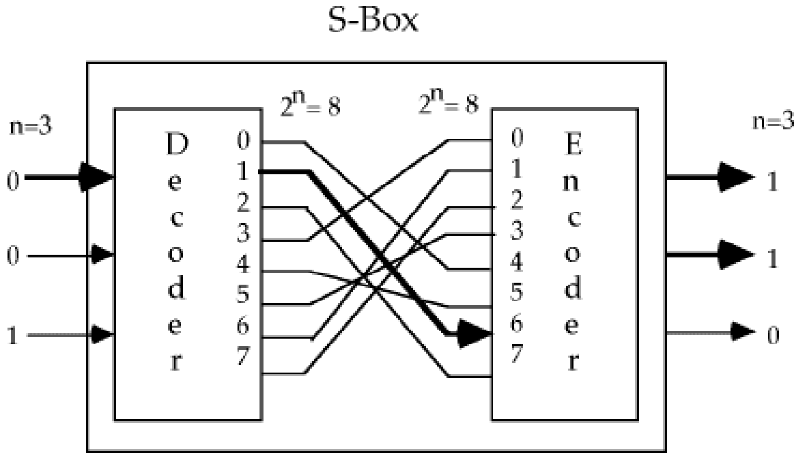
The Permutation Operation

- A binary word (i.e. block) has its bits reordered (permuted)
- The re-ordering forms the key
- Operation represented by a **P-box**
- The example allows for 15! = 1,307,674,368,000 combinations
- The key describes the combination used



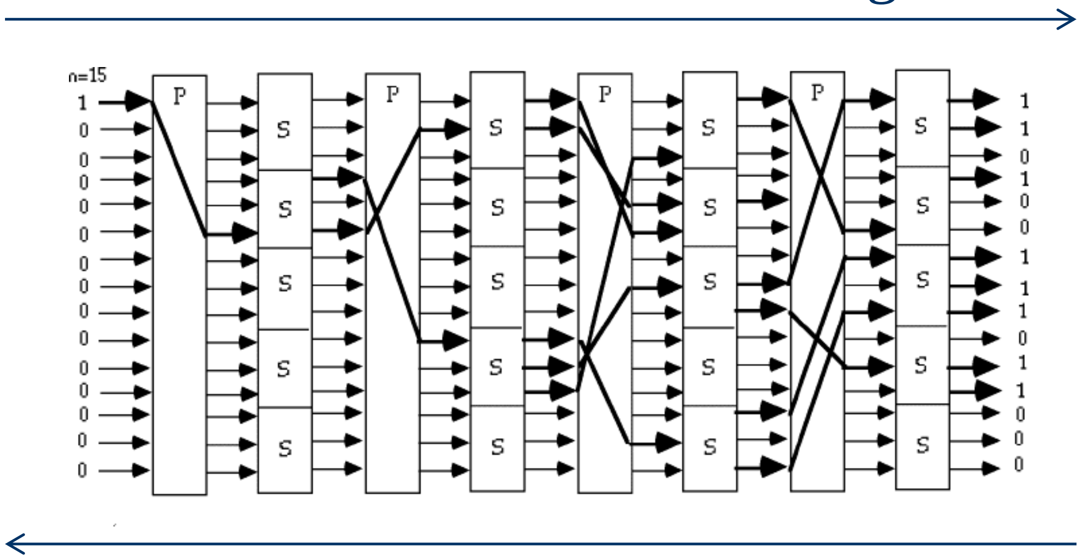
The Substitution Operation

- A binary word is replaced by some other binary word
- The whole substitution function forms the key
- Operation represented by an **S-box**
- The box below allows for 8! = 40320 combinations
- The key describes the combination used



Substitution-Permutation Network

encoding



decoding

- The key describes the internal wiring of all S-boxes and P-boxes
- The same key can be used for encoding and decoding, hence it is a **private key encryption algorithm**
- The direction of the process determines encoding / decoding

Ollscoil na Gaillimhe
University of Galway

Confusion and Diffusion

- A cipher needs for obvious reasons to completely obscure statistical properties of original message
- Shannon introduced two terms to describe this:
  - **Diffusion** seeks to make the statistical relationship between the plaintext and ciphertext as complex as possible
  - **Confusion** seeks to make the relationship between the statistics of the ciphertext and the value of the encryption key as complex as possible
- Both thwart attempts to deduce the key used via a cryptanalysis (as seen before)

Confusion and Diffusion in Practice

- Example DES (→later): A swap of a single bit either in the key or in the plaintext result in a significant change in the ciphertext
- Note that DES encrypts a message over 16 iterations (rounds)

(a) Change in Plaintext		(b) Change in Key	
Round	Number of bits that differ	Round	Number of bits that differ
0	1	0	0
1	6	1	2
2	21	2	14
3	35	3	28
4	39	4	32
5	34	5	30
6	32	6	32
7	31	7	35
8	29	8	34
9	42	9	40
10	44	10	38
11	32	11	31
12	30	12	33
13	30	13	28
14	26	14	26
15	29	15	34
16	34	16	35

Important Block Cipher Principle: Reversible Transformation

- Transformations must be reversible or non-singular, e.g.

Plaintext		Ciphertext
00	↔	11
01	↔	10
10	↔	00
11	↔	01

vs.

Plaintext		Ciphertext
00	↔	11
01	↔	10
10	↔	**00**
11	↔	**00**
	?	

- There must be a 1:1 association between a n-bit plaintext and an-bit ciphertext, otherwise mapping (encryption) is irreversible

OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

Features of Private-Key Cryptography / Ciphers

- Traditional private/secret/single key cryptography uses one key, shared by only sender and receiver
- The algorithm / cipher itself is public, i.e. not a secret
- If the key is disclosed, communications are compromised
- The key is also **symmetric**, parties are equal
- Hence methods does not protect sender from receiver forging a message & claiming is sent by sender
- Examples include DES (Data Encryption Standard) and AES (Advanced Encryption Standard)

Examples AES

- Advanced Encryption Standard, successor of DES
- Modern block cipher with 128 bits block length
- Uses 128, 192 or 256 bit long keys
- The de-facto standard for secure encryption
- Widely used for
  - File / data encryption
  - Secure network (e.g. Internet) Communication

Why does Block and Key Length matter?

- Cryptographic algorithms with short block length can be tackled as seen with substitution cipher
- Large keys and long blocks prevent **brute-force attacks / searches**
  - Take the ciphertext and try all possible key combinations (or block permutations), until the decoded text makes sense

Brute Force Search / Attacks

- A 56-bit key has a key space that contains 2^{56} keys
  - A prominent early day symmetric cipher called DES (Data Encryption Standard) used 56 bit keys… it is deemed unsafe since the 1990s
- A 128-bit key has 3.4E38 possible combinations
  - Generally accepted minimum key length today

Brute Force Search

- Always possible to simply try every key
- Most basic attack, effort proportional to key size
- Assume that you either know or recognise plaintext

Key Size (bits)	Number of Alternative Keys	Time required at 1 decryption/µs	Time required at 10^6 decryptions/µs
32	$2^{32} = 4.3 \times 10^9$	2^{31} µs = 35.8 minutes	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	2^{55} µs = 1142 years	10.01 hours
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} µs = 5.4×10^{24} years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} µs = 5.9×10^{36} years	5.9×10^{30} years
26 characters (permutation)	$26! = 4 \times 10^{26}$	2×10^{26} µs = 6.4×10^{12} years	6.4×10^6 years

The Feistel Cipher

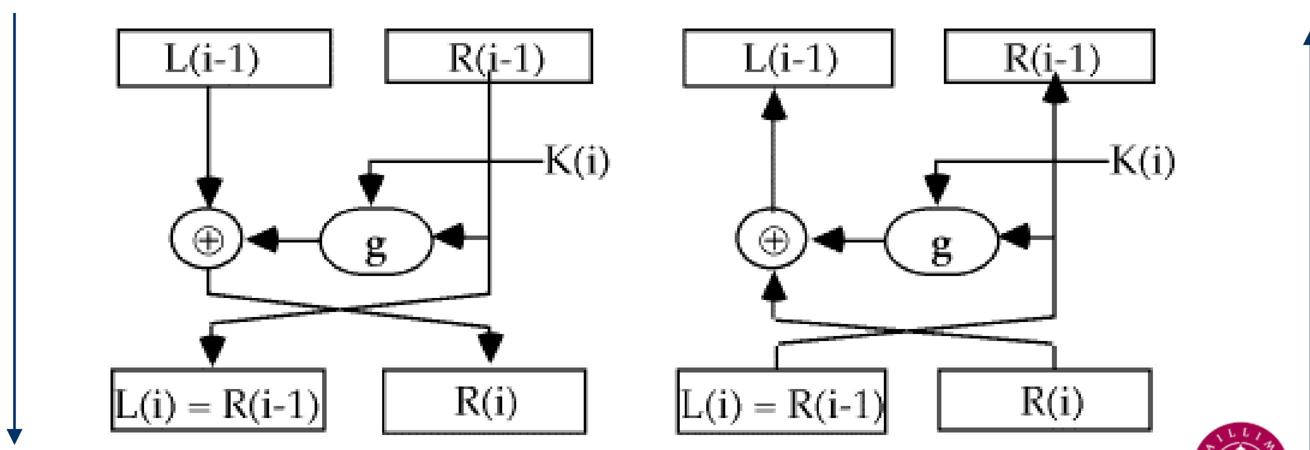
- In practice we need to be able to decrypt messages, as well as to encrypt them, hence either:
  - have to define inverses for each of the S & P-boxes, but this doubles the code/hardware needed, or
  - define a structure that is easy to reverse, so can use basically the same code or hardware for both encryption and decryption
- A **Feistel cipher** is such a structure
  - It is based on concept of the **invertible product cipher**
  - Most symmetric block ciphers are based on a Feistel Cipher structure

The Feistel Cipher

- Horst Feistel, working at IBM Thomas J Watson Research Labs, devised a suitable invertible cipher structure in early 70's
- One of Feistel's main contributions was the invention of a suitable structure which adapted Shannon's S-P network in an easily invertible structure
- Essentially the same hardware or software is used for both encryption and decryption, with just a slight change in how the keys are used

The Feistel Cipher – A Single Round

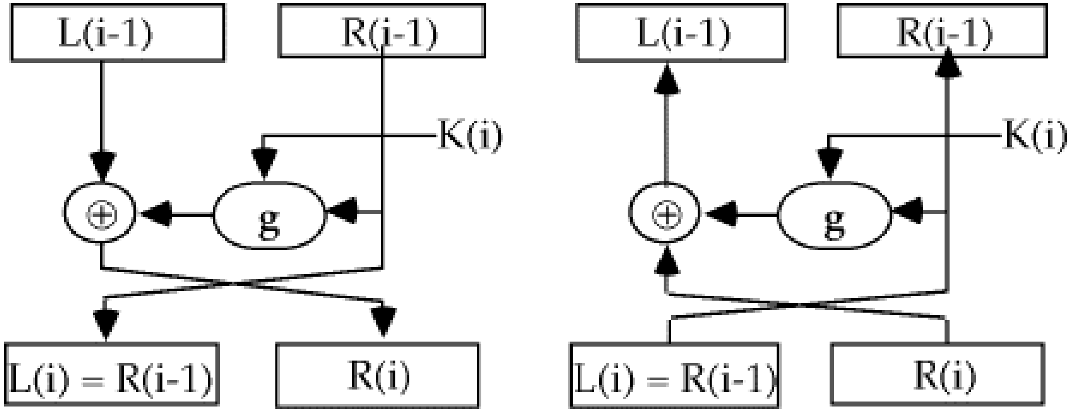
- The idea is to partition the input block into two halves, L(i−1) and R(i−1), and use only R(i−1) in the ith round (part) of the cipher
- The function g incorporates one stage of the S-P network, controlled by part of the key K(i) known as the ith subkey



OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

The Feistel Cipher – A single Round

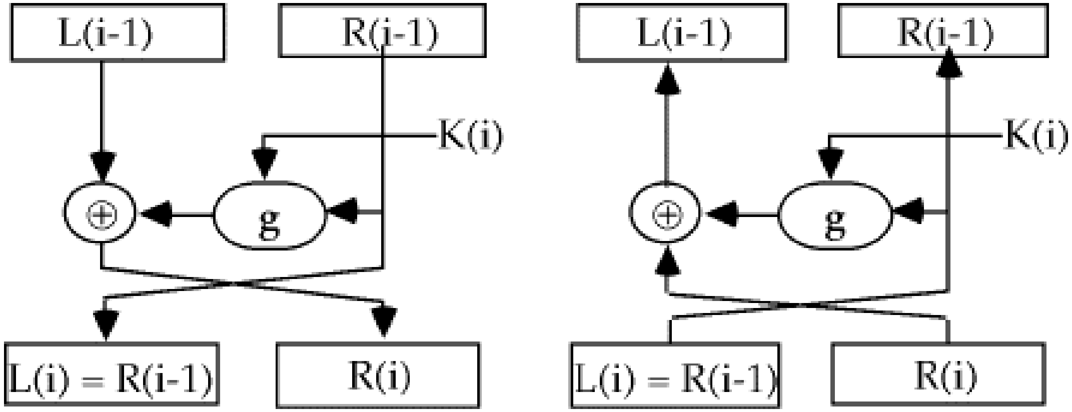
- A round of a Feistel cipher can be described functionally as:
  - `L(i) = R(i-1)`
  - `R(i) = L(i-1) EXOR g(K(i), R(i-1))`



Symmetry of Bitwise EXOR

- A EXOR B = C
  A EXOR C = B
  C EXOR B = A

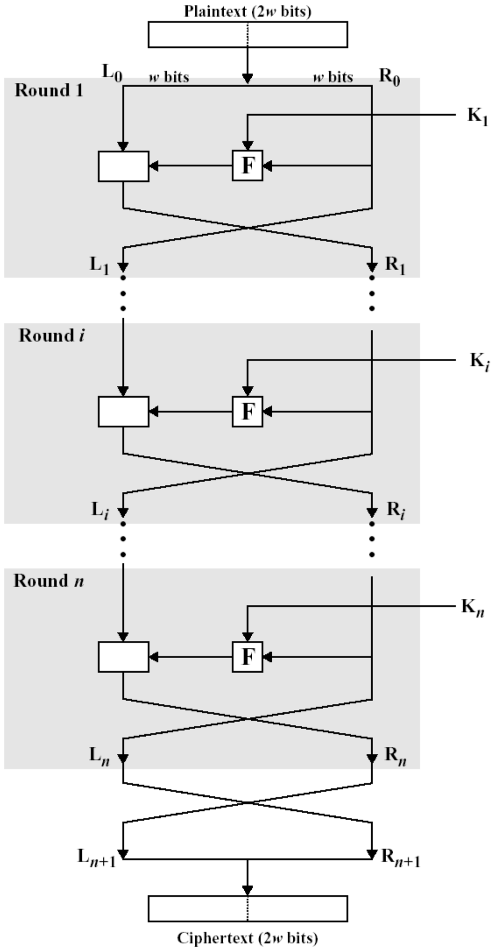
	0	1
0	0	1
1	1	0



Example

- Encoding of 01011110:
  - L(i - 1) = 0101 R(i - 1) = 1110
  - g(K(i), R(i-1)) = 1001 L(i) = 1110
  - R(i) = 0101 XOR 1001 = 1100
  - Therefore 01011110 becomes 11101100
- Decoding of 11101100:
  - L(i) = 1110 R(i) = 1100
  - g(K(i), R(i-1)) = 1001 R(i - 1) = 1110
  - L(i - 1) = 1100 XOR 1001 = 0101
  - Therefore 1110 1100 becomes 01011110

A Feistel Network



- Perform multiple transformations (single rounds) sequentially, whereby output of i^{th} round becomes the input of the $(i+1)^{th}$ round
- Every round gets is own subkey, which is derived from master key
- Decryption process goes from bottom to top

OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

Feistel Cipher Design Elements

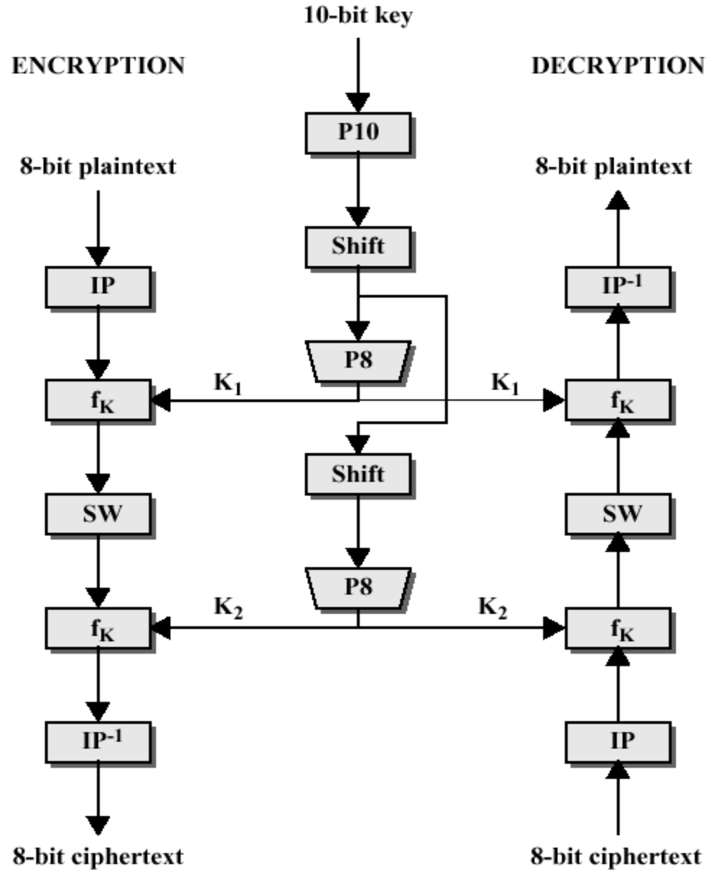
- Block size
- Key size
- Number of rounds
- Subkey generation algorithm
- Round function
- Fast software encryption/decryption

Simple Methods for Subkey Generation

- Multiple subkeys are based on a bigger master key
- Method 1:
  - MK: 010100010100011110101001
  - SKs:010100010100011110101001
- Method 2:
  - MK: 010100010100111
  - SKs:010100010100111

Example for private Key Block Cipher: Simple DES

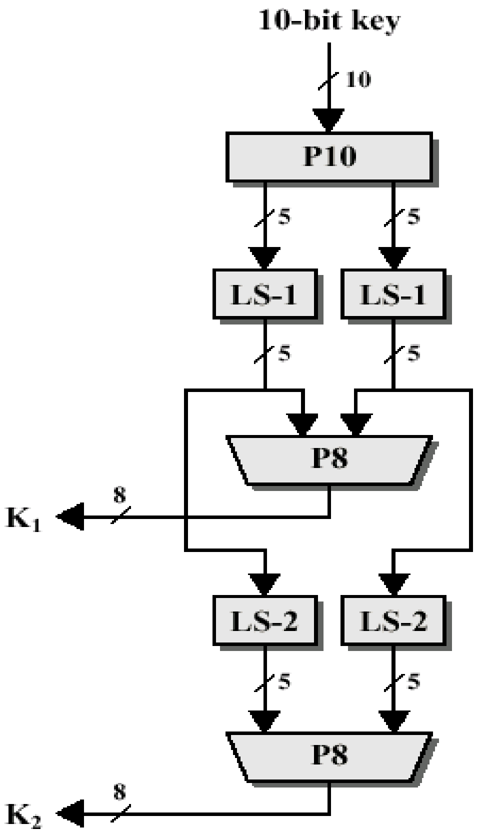
- An educational version of DES (Data Encryption Standard), the first widely used private key encryption algorithm:
  - 8 bit blocks and 10 bit keys
  - IP, IP^{-1} = (initial) permutation
  - P10 = 10 bit permutation
  - P8 = 8 bit permutation and selection.
  - SW = swap 2 halves



OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

FYI: Simple DES – Key Generation

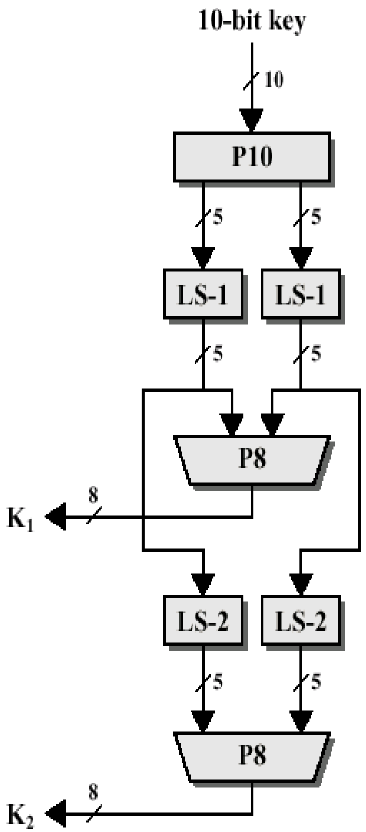
- P10: Permutation
  3 5 2 7 4 10 1 9 8 6
- LS-1: Left-shift 1
  Circular shift by 1 bit.
- P8: Permutation
  6 3 7 4 8 5 10 9
- LS-2: Left Shift 2
  Circular shift by 1 bit.
- P8: Permutation
  6 3 7 4 8 5 10 9



OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

FYI: Example for Sub-Key Generation

- 10-bit key: 0110010110
- P10 permutation: 3 | 5 | 2 | 7 | 4 | 10 | 1 | 9 | 8 | 6
  10100 00111
- Circular left shift: 01001 01110
- P8 Permutation: 6 | 3 | 7 | 4 | 8 | 5 | 10 | 9
  K1: 00101101
- Circular left shift: 10010 11100
- P8 Permutation: 6 | 3 | 7 | 4 | 8 | 5 | 10 | 9
  K2: 10111000



FYI: Structure of f_K

- E/P expansion permutation
  4 1 2 3 2 3 4 1
- 2 S-boxes S0 and S1

S0:

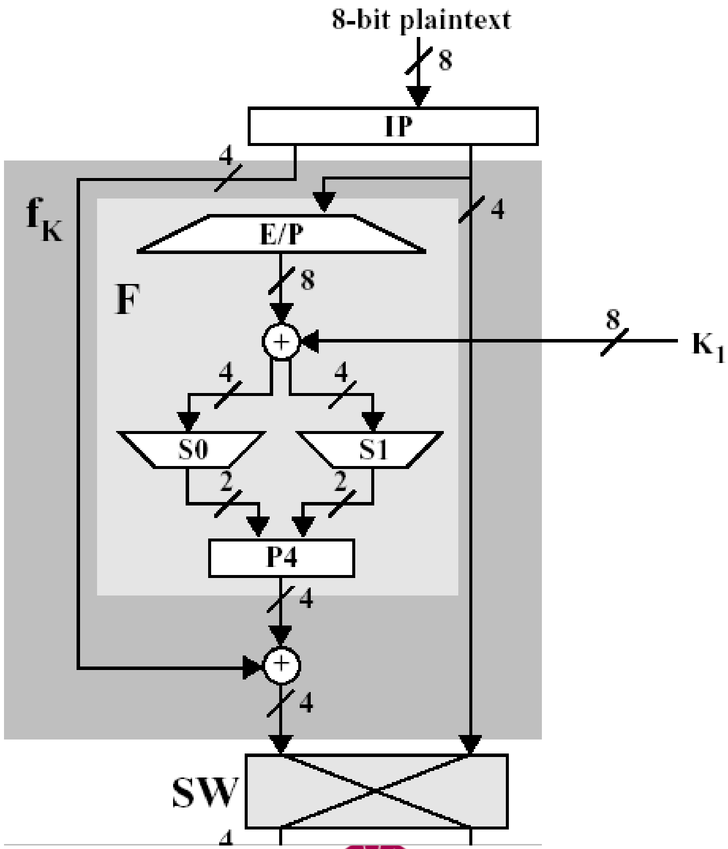
	0	**1**	**2**	**3**
0	1	0	3	2
1	3	2	1	0
2	0	2	1	3
3	3	1	3	2

S1:

	0	**1**	**2**	**3**
0	0	1	2	3
1	2	0	1	3
2	3	0	1	2
3	2	1	0	3

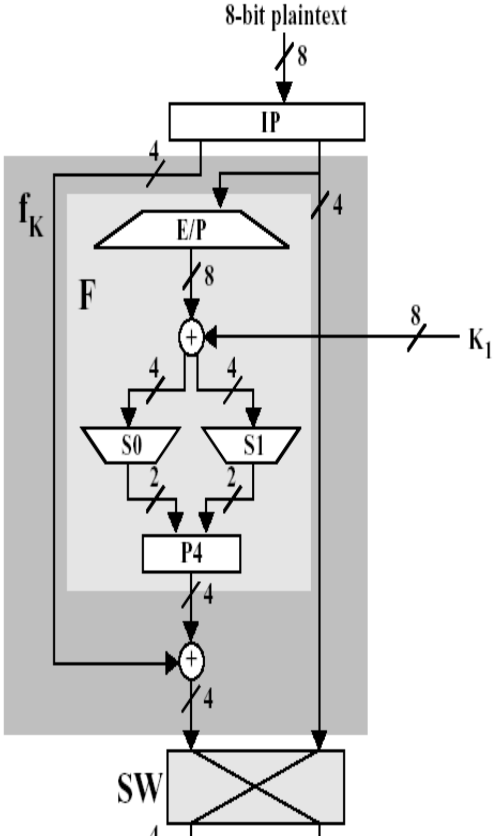
The 1st and 4th input bits specify a row, the 2nd and 3rd input bits represent a column. The corresponding entry in a table represents the output

- P4 permutation 2 4 3 1



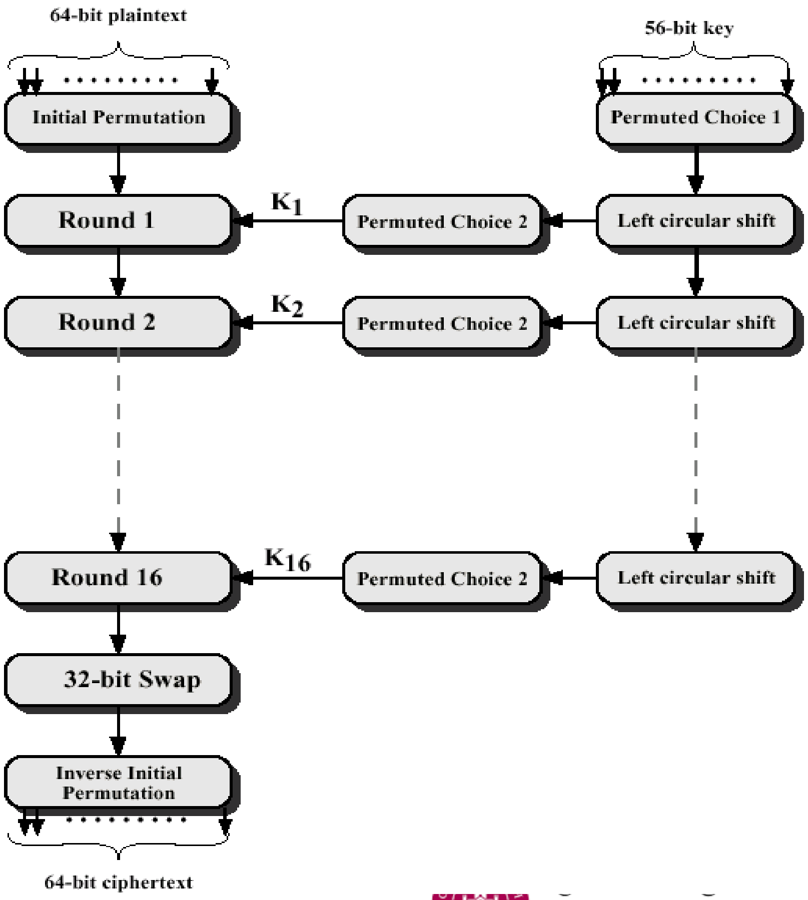
FYI: Example for f_K

- Input after IP: 01001101
- Left part: 0100
- E/P: 4|1|2|3|2|3|4|1
  11101011
- EX-OR K1: 00101101
  11000110
- S0 and S1: See previous page
  1011
- P4 permutation: 2|4|3|1
  0111
- EX-OR left part: 0100
  0011
- Concatenate right block: 1101
  00111101
- Swap: 11010011



DES

- 64 bit plain text
- 56 bit key and 48 bit sub-keys
- 16 rounds

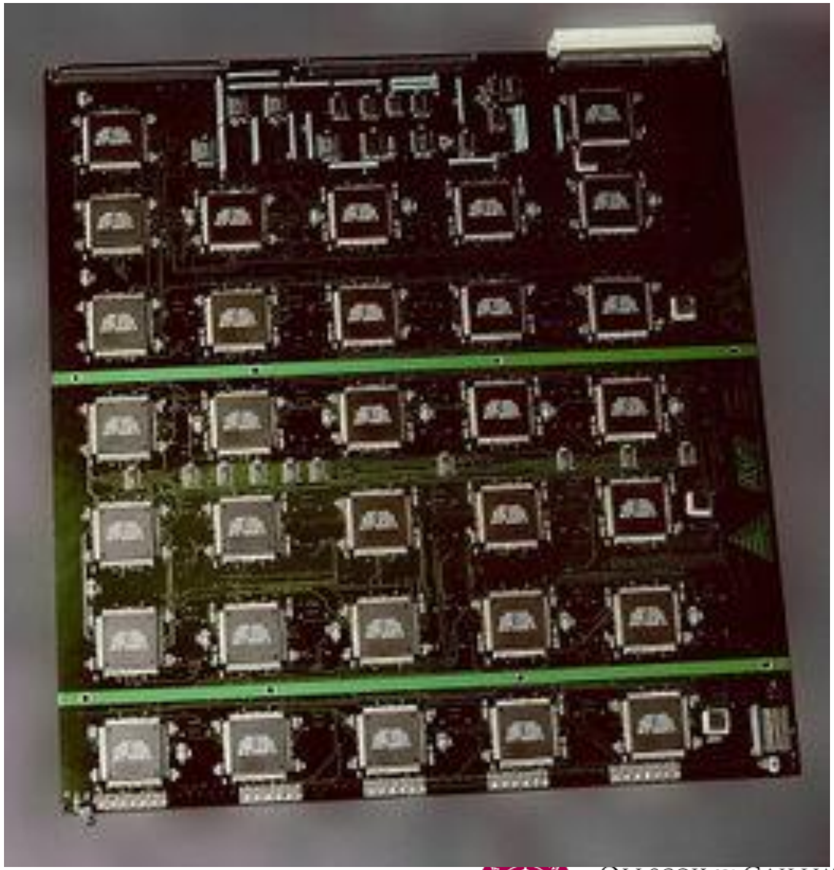


Strength of DES – Key Length?

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ possible values
- Brute force search looks hard …
- But advances in 1990s have shown that it is possible:
  - In 1997 on Internet in a few months (using a PC cluster)
  - In 1998 on dedicated hardware in a few days
  - In 1999 above combined in 22 hrs!
- As a result, alternatives to DES had to be considered

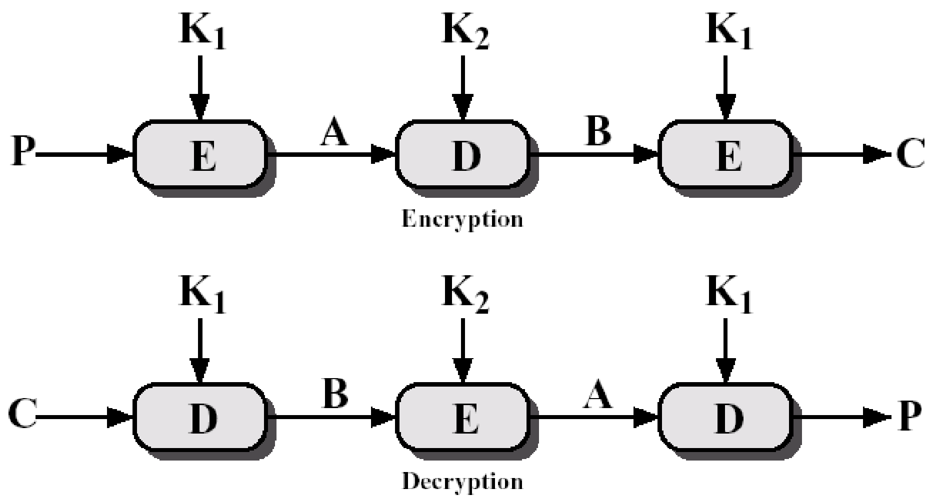
The DES Cracking Machine

- Developed by Electronic Frontier Foundation (EFF)
- Image shows a single circuit board.
- The entire machine consisted of 1,536 custom chips

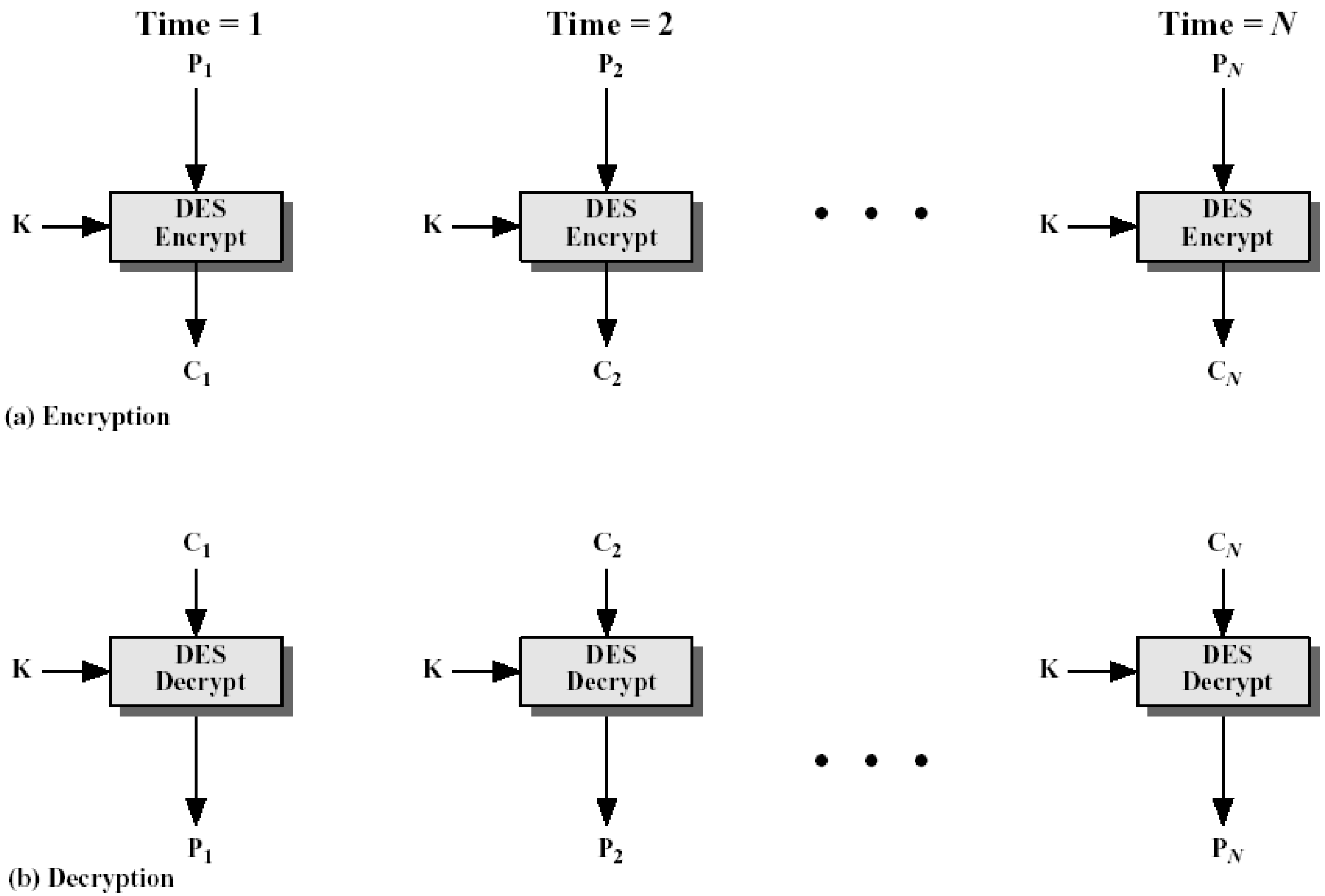


Triple DES

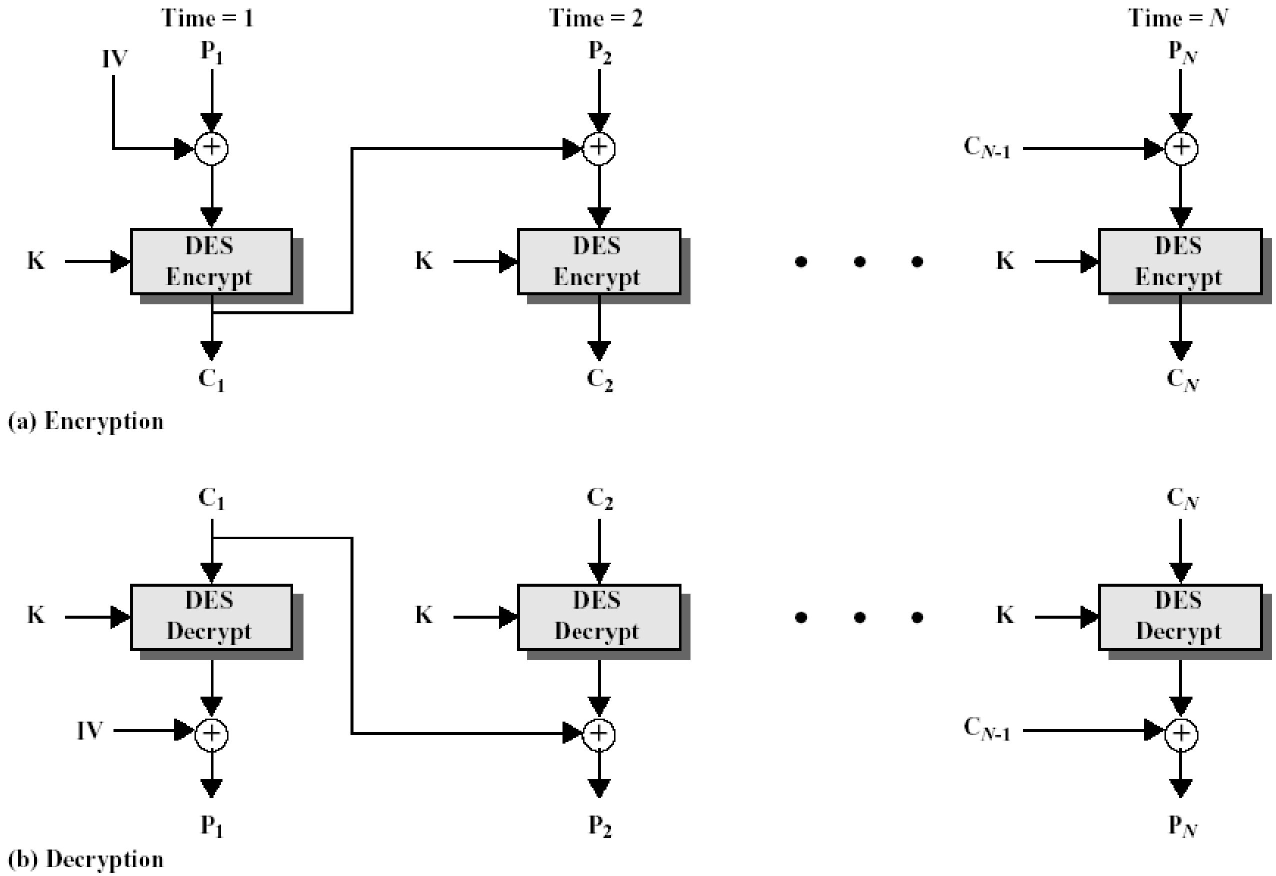
- Based on 2 (56-bit each) keys and three stages
- Symmetry preserved, therefore same concatenation is used for encoding and decoding



Modes of Operation: Electronic Codebook (EBC) Mode



Modes of Operation: Cipher Block Chaining (CBC) Mode



STREAM CIPHERS

Stream Ciphers

- So far we have examined block ciphers that process n-bytes at a time
- Stream ciphers in contrast process the message bit by bit (as a stream)
- They require a stream key K that is a pseudo-random sequence of 0s and 1s
- This bit-stream K is combined (EXORed) with the plaintext M bit by bit to generate the cipher text C:

$$C_i = M_i \text{ EXOR } K_i$$

- The randomness of the stream key completely destroys any statistically properties in the message
- The receiver generates the identical bit stream K and decodes the message C:

$$M_i = C_i \text{ EXOR } K_i$$

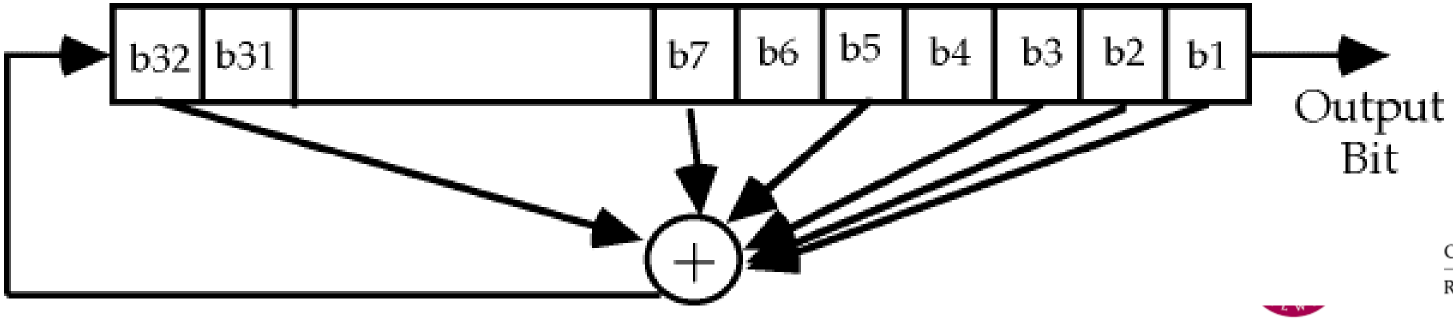
- Vernam Cipher or One-Time Pad is a famous stream cipher

Vernam Cipher

- Vernam cipher requires as many (random) key bits as message is long
  - Every message requires a new key, as reusing a stream key may allow an attacker to recover it!
- Such keys must be distributed securely between end-points
  - Very complicated, tedious and uneconomic, as a single stream key may consist of millions of bits
- For practical reasons stream ciphers based on pseudo-random generators (PRG) are used
  - PRGs are often based on Linear Feedback Shift Registers (LFSRs)
  - Only a seed value to initialise the PRG must be shared

Linear Feedback Shift Registers (LFSR)

- Consist a binary shift register of some length along with a linear feedback function that operates on some of those bits
- Each time a bit is needed, all bits are shifted right by one position
- The bit bumped out is the bit used as (pseudo-random) output from the LFSR
- A new bit is formed from the linear feedback function of some bits
- Correctly designed LFSRs generate a very long pseudo-random sequence before repeating
- LFSRs require an initialisation vector (i.e., seed) for their shift register

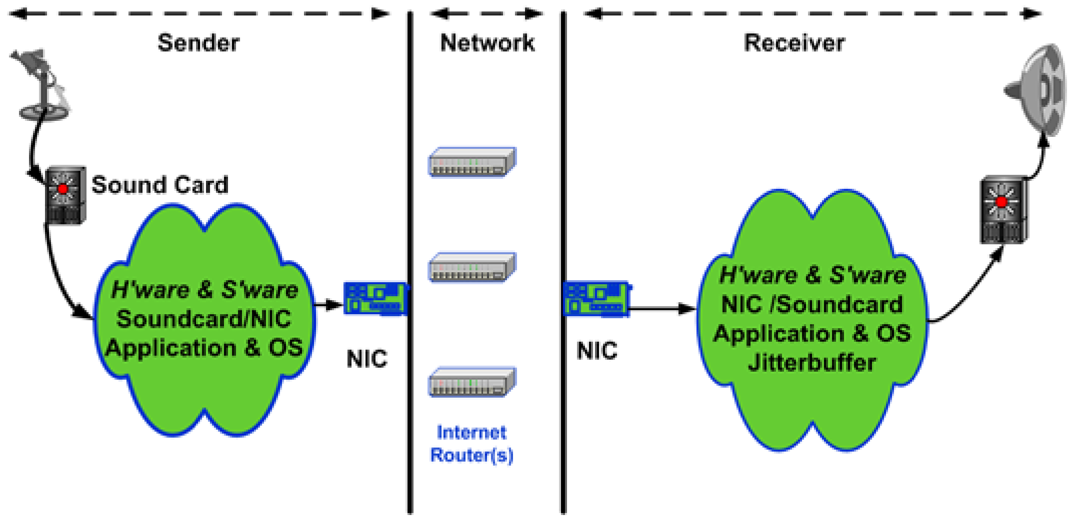


Example for an 8-Bit LFSR

- Initialisation vector: 10100110 (B_7 ··· B_0)
- Feedback Function: B_7 EXOR B_4 EXOR B_1
- Right shift after each cycle (B_0 shifted out)
- Iteration 0: 10100110
- Iteration 1: 01010011 >> 0
- Iteration 2: 00101001 >> 1
- Iteration 3: 00010100 >> 1
- Iteration 4: 10001010 >> 0
- … …

The feedback function returns a "1", if an odd number of inputs is set to "1"

Example VoIP (Voice over the Internet Protocol)



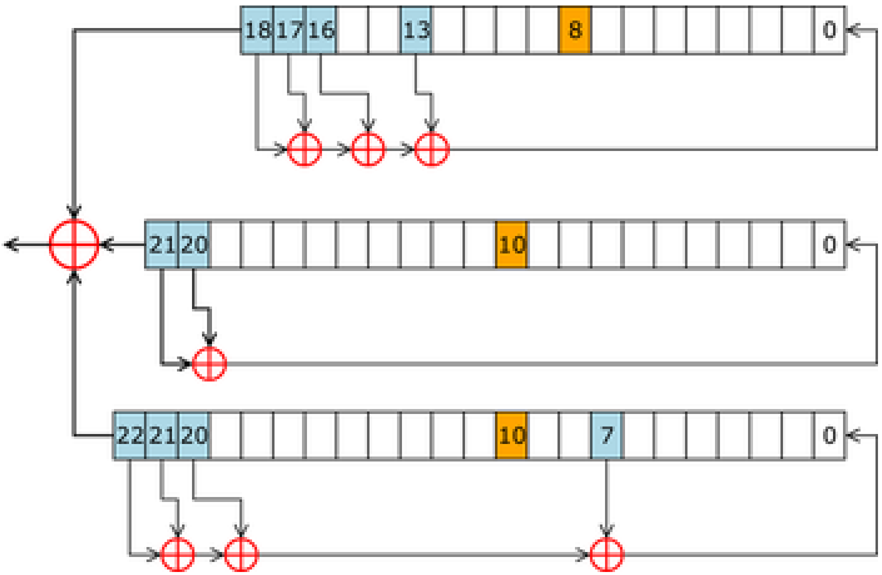
- The sender's voice is digitised and the resulting bit stream is encrypted using a stream cipher before being sent to the receiver over a network link
- Sender and receiver share the same seed value for their PRG

Stream Ciphers in Mobile Communication (early 2000s)

- Mobile phone conversations are sent as sequences of frames between both end points
  - Voice samples are collected and digitised by the mobile phone
- Every 4.6 milliseconds a 228-bits long frame consisting of digitised voice is processed and send out
- A5/1 is an LFSR-based algorithm that was used to produce 228 bits of key stream which is EXORed with the frame
- A5/1 is initialised using a 64-bit key

A5/1

- 3 independent LFSRs:
  - 19 bits
  - 22 bits
  - 23 bits
- The **majority bit** is the XORed output of all 3 LFSRs
- Each register is only shifted to the left, if their clocking bits (B8, B10, and B10 respectively) match the majority bit



A5/1

- A5/1 was originally introduced in 1987
- It was protected as a "trade secret“, but has subsequently been reverse engineered during the 90s
- As a result A5/2 was introduced, which has been broken as well
- A5/3 (KASUMI) was released in late 2002
  - Block-cipher based on Feistel network

RC4

- RC4 is a PRG designed by Ron Rivest of RSA Security in 1987
- RC4 was initially a trade secret, but 1994 a description of it was anonymously posted in the Internet
- It consists of a
  - key-scheduling algorithm (KSA) and a
  - pseudo-random generation algorithm (PRGA)

RC4: The Key-Scheduling Algorithm (KSA)

- Requires a keyword (stored in `key[]`) with a specific `keylength`
- An 256 byte long permutation vector `S[]` is generated:

```
for i from 0 to 255
    S[i] := i;
j := 0;
for i from 0 to 255
    j := (j + S[i] + key[i mod keylength])
            mod 256;
    swap(S[i],S[j]);
```

Ollscoil na Gaillimhe
University of Galway

RC4: The Pseudo-Random Generation Algorithm (PRGA)

- PRGA returns one byte at a time:

```
i := 0;
j := 0;
while GeneratingOutput:
    i := (i + 1) mod 256;
    j := (j + S[i]) mod 256;
    swap(S[i],S[j]);
    output S[(S[i] + S[j]) mod 256];
```

RC4

- Not an LFSR-based design, but rather a more general pseudo-random number generator design
- Can be efficiently implemented in software
- Broken and not used any more!

CT255
Introduction to Cyber-Security

Lecture 9

Message Authentication

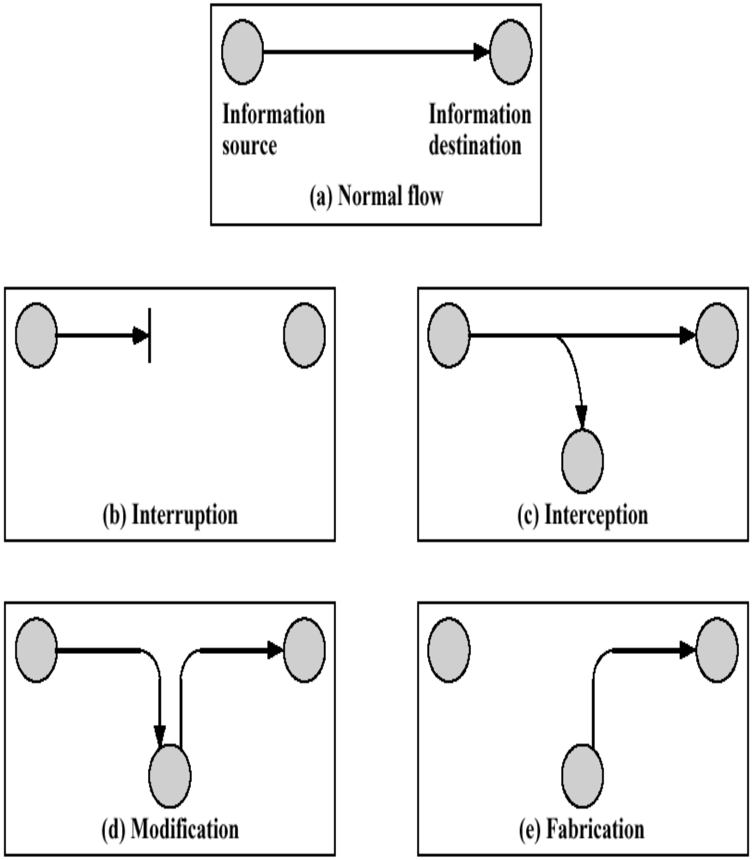
Dr. Michael Schukat, 2019-2022

Outline

- Types of security attacks
- Message Authentication
- Hash functions revisited

Types of Security Attacks

- Interception - of info-traffic flow, attacks confidentiality
- Interruption - of service, attacks availability
- Modification - of info, attacks integrity
- Fabrication - of info, attacks authentication



Passive Attacks

- Are in the nature of eavesdropping or monitoring of transmissions:
  - Release of message content
  - Traffic analysis
    - Analyse pattern of messages (sender, receiver, timing) rather than content
  - Tools like Wireshark allow eavesdropping on network traffic

Active Attacks

- Involved modification or creation of data stream:
  - Masquerade
    - Pretend to be a different entity
  - Replay
    - Retransmission of captured data
  - Modification of message
  - Denial of service (DoS)
    - Inhibits the normal use of communication services

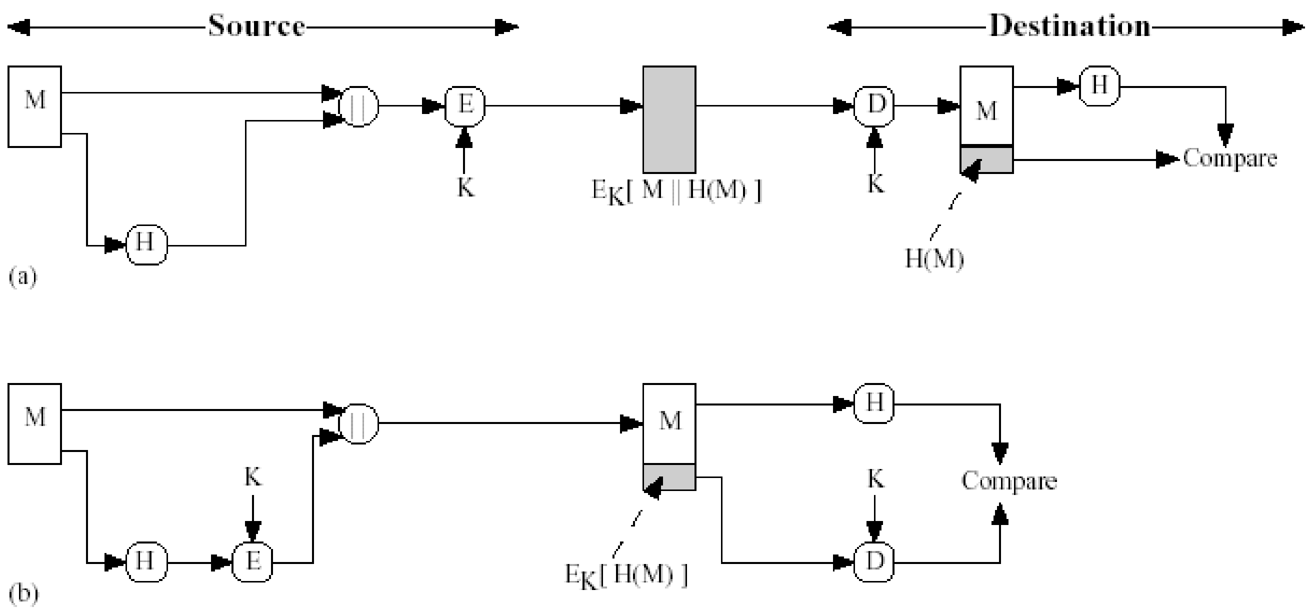
Message Authentication

- There are four types of attacks in the context of communication across a network, which are addressed by message authentication:
  - **Masquerad**e: insertion of messages into the network from a fraudulent source
  - **Content modification**
  - **Sequence modification**
  - **Timing modification**: delete or repeat messages
- Message authentication is concerned with:
  - Protecting the integrity of a message
  - Validating identity of originator
  - Validating sequencing and timeliness
  - Non-repudiation of origin (dispute resolution)

Hash Functions

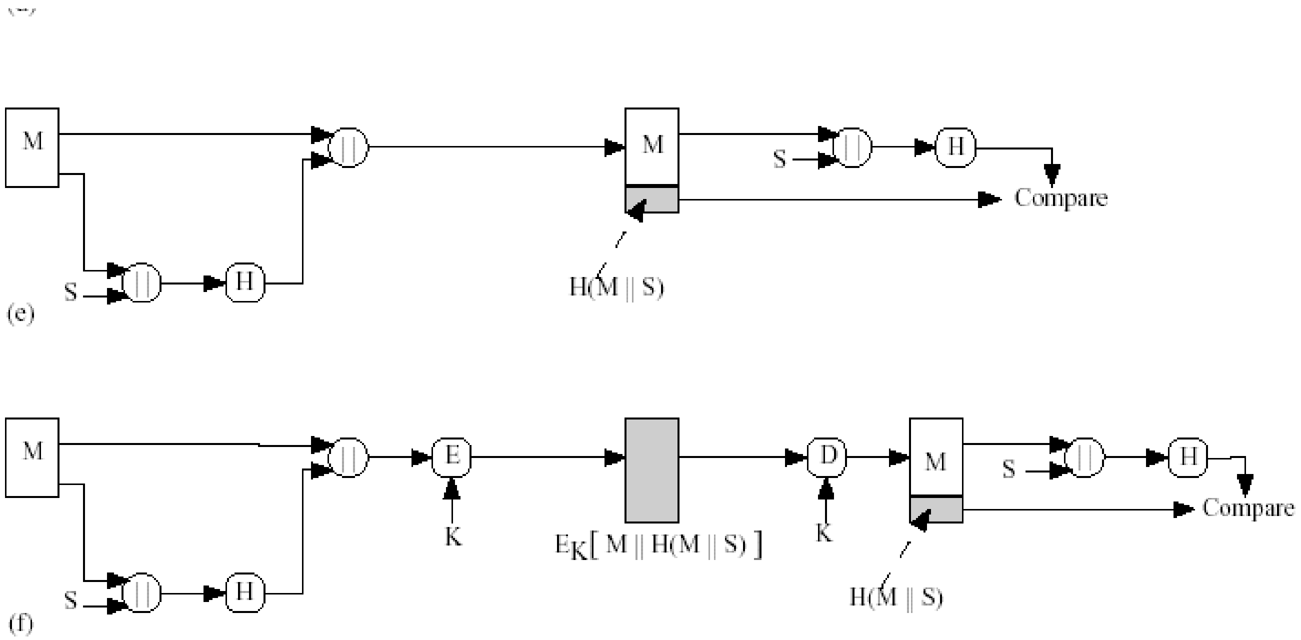
- A hash function is a variation of a MAC, which produces a fixed size hash code ("**fingerprint**") based on a variable size input message
- A hash function is public and is not keyed, therefore the hash value must be encrypted
- Traditional CRCs are to weak and cannot be used (see requirements for hash functions)
- 128-512 bits hash values are regarded as suitable

Basic Uses of Hash Functions



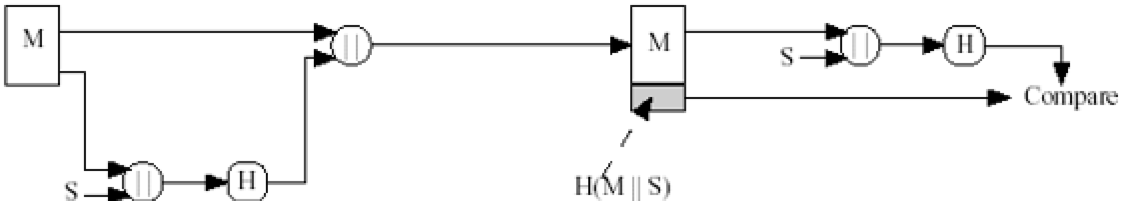
LSCOIL NA GAILLIMHE
IVERSITY OF GALWAY

Basic Uses of Hash Functions



Recall: Requirements for Hash Functions H(x)

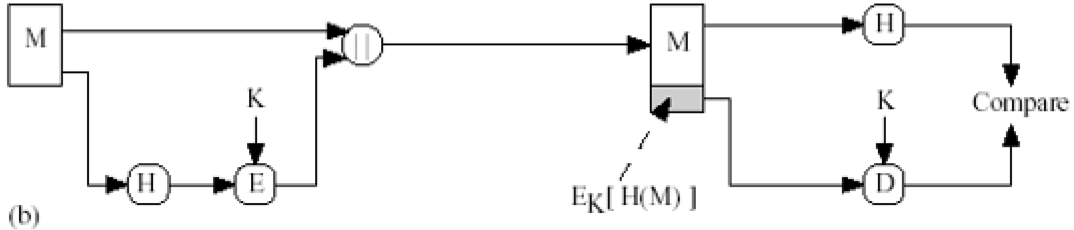
- **One way property**:
  For a given hash code h it is infeasible to find x that H(x) = h
- Reason:
  See Figure (e): An opponent could reveal secret key s otherwise



(e)

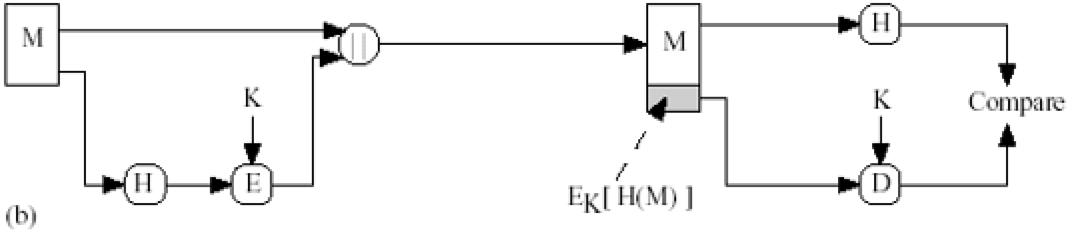
Recall: Requirements for Hash Functions H(x)

- **Weak collision resistance**:
  For a given block (or text) x it it is infeasible to find another block (or text) y with
  y != x with H(x) = H(y)
- Reason:
  See Figure (b): An opponent can calculate the hash code for M, find an alternate message with the same hash code, and send it together with the encrypted (original) hash code to the receiver



Recall: Requirements for Hash Functions H(x)

- **Strong collision resistance**:
  It is computational infeasible to find a pair of blocks (or texts) (x, y) with H(x) = H(y)
- Reason:
  See Figure (b), where the message is not encoded and no additional secret key for the hash function is used.
  Attack is based on (counterintuitive) **Birthday Paradox**



(b)

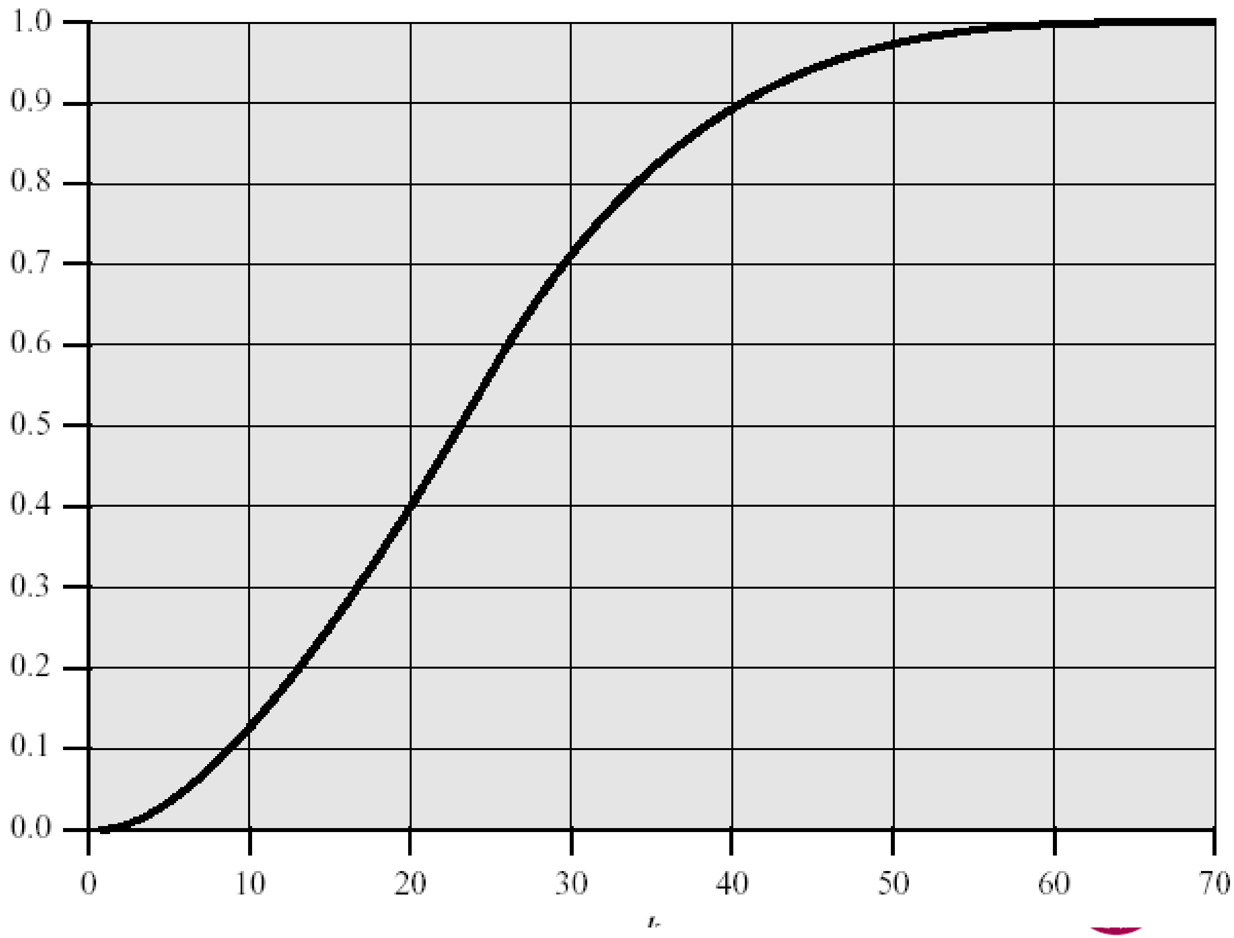
Recall: Birthday Paradox

- What is the minimum value k such that the probability is greater than 50% that at least 2 people in a group of k people have the same birthday, assuming that a year has 365 days?

- Intuitively someone would assume that $k = 365 / 2 =$ **183**

- Probability theory shows, that $k =$ **23** is sufficient!

Birthday Paradox



CT255 (S1) Summary

- We covered:
  - GDPR
  - Basic Cryptographic concepts including
    - Classic cryptography
    - Block ciphers, stream ciphers
    - Hash functions and rainbow tables
  - User passwords social engineering

Week 12 MCQ

- Open book, worth 5% (out of 50%)
- 20 random questions covering all CT255 topics
- 20 minutes time to complete
- One question at a time is shown
- Backtracking is not allowed
- Monday 21/11, 13:30 – 13:50 sharp
  - i.e. quiz has to be submitted by 13:50