COMP2300/COMP6300 - Applied Cryptography

Introduction

Les Bell, les.bell@mq.edu.au

Department of Computing

MACQUARIE
University

MACQUARIE
University

Table of Contents

MACQUARIE University

Welcome to Country

We would like to acknowledge the traditional custodians of this land, the Wattamatagal Clan of the Dharug Nation, and pay our respects to Elders past, present and future.

Welcome to Country

Welcome

- Staff

Convenor	Dr. Hassan Asghar, hassan.asghar@mq.edu.au
Lecturer	Les Bell, les.bell@mq.edu.au
Workshops	Mohamadali Mehrabi, moahamadali.mehrabi@hdr.mq.edu.au and Nazim Sheikh, nazim-uddin.sheikh@mq.edu.au

Unit Outline

COMP2301 - Symmetric Cryptography

- Overview and Introduction
  - Unit Overview
  - Security Properties of Information and Systems
  - Categorization of Cryptoprimitives
  - Using Cryptographic Libraries
- Symmetric Cryptoprimitives
- Hashing and Message Authentication Codes
- Encrypting Data at Rest (Disk and File Encryption)
- Entity authentication and authenticity of origin

COMP2302 - Public Key Cryptography

- Mathematical Essentials
- Public Key Cryptography and Solving the Key Exchange Problem
- Digital Signatures and Non-repudiation
- Encrypting Data in Motion (Email, Network Protocols)

COMP2303 - Advanced Techniques and Cryptanalysis

- Anonymity, Zero Knowledge Proofs, Voting Protocols
- Blockchain and Cryptocurrencies
- Quantum Cryptography and Post-Quantum Cryptography
- Cryptanalytic Attacks

MACQUARIE University

Unit Guide - Lectures and Readings

Lectures

Recordings of lectures (Echo360) will be provided via iLearn

Textbooks/Readings

- Smart, Nigel P., Cryptography Made Simple, Springer International, Switzerland, 2016 [Sma16].
- A. J. Menezes, P. C. van Oorrschot and S. A. Vanstone, Handbook of applied cryptography (HAC), CRC Press, Boca Raton, FL, 1996 [MVOV97]. Download from

  `http://cacr.uwaterloo.ca/hac/`
- R. Anderson, Security Engineering, 2nd ed. (SE) Wiley Publishing, Inc. 2008 [And10]. Download from

  `http://www.cl.cam.ac.uk/~rja14/book.html`

MACQUARIE University

Unit Guide - Bring Your Own Device

- COMP2300 is a *Bring Your Own Device (BYOD)* unit.
  - In olden days we *recommended* students use their own computers
  - For the last four years, we've *required* it
- You should bring your own laptop to the workshops.
- This will make it easier to secure valuable personal cryptographic data and to incorporate good security practices and tools into your daily work.

MACQUARIE University

Unit Guide - Assessment Tasks

- Tutorial Submissions and Quizzes
  - Weekly tasks - best 8 marks used to calculate 10% of unit total marks
- Assignments
  - Assignment 1 - Password-based File Encryption and Decryption - 15% of total marks
  - Assignment 2 - Public-Key Signing and Encryption - 15% of total marks
- Module Examinations
  - Module 1 - Symmetric Cryptography - 10% of marks
  - Module 2 - Public-Key Cryptography - 10% of marks
  - Module 3 - Advanced Applications and Cryptanalysis - 10% of marks
- Hurdles
  - You must attempt at least six (6) of the weekly tutorial tasks
- Late Submission
  - 10% penalty for each day or part day that a submission is late
  - No extensions without an approved application for Special

MACQUARIE University

Module Examinations

Module Examinations will be held in the tutorial workshop lab.

You *must* attend your assigned tutorial/workshop - if you attend the wrong one, you will not be able to log in to the examination.

Changes of workshop *must* be performed through eStudent. I suggest that you use the iLearn discussion forum to request a swap with another student, and you then sit down together and simultaneously withdraw from one class and then enrol in the other.

The unit teaching staff will not be responsible for inability to attempt examinations if you turn up for the wrong tutorial/workshop.

MACQUARIE University

Unit Guide - Disruption to Studies

Special Consideration

- In case of circumstances that have adversely affected your performance (especially in one of the Module Examinations):
- Make sure you understand the policy of the Department. See
  `https://students.mq.edu.au/study/my-study-program/special-consideration`
- In particular, if you are granted special consideration, you may be required to sit a Supplementary Examination.

  In that case:

  Your performance in the final exam will not be considered.

  Your grade will be based on the Supplementary Examination only.

Withdrawal

If you feel you are really out of your depth, or not coping with the workload:

- Last day to withdraw from the unit without academic or financial penalty: March 19th 202 (census date)
- Last day to withdraw from the unit without academic penalty: April 28th 2020

MACQUARIE University

Unit Guide - Academic Integrity

You must complete the Academic Integrity Module before full unit content will become visible.

Unit Guide - Professional Ethics

Infosec Responsibilities

- Infosec professionals have "unusual" skills and capabilities
- Society (represented by professional bodies) expects us to use these:
  - To protect society, shared infrastructure and public resources
  - To act honestly, honourably, justly, responsibly and legally
  - To protect the interests of clients and principals
  - To advance and protect the profession

Unit Guide - Ethics

Specifically:

- You need to be aware of the University's
  - Acceptable Use of IT Resources Policy

    `https://staff.mq.edu.au/work/strategy-planning-and-governance/univers`
  - Information Security Policy

    `http://www.mq.edu.au/about_us/offices_and_units/information_technolog`
  - Network Policy (currently under review)

    `http://www.mq.edu.au/about_us/offices_and_units/information_technolog`
- Especially, do not:
  - Sniff network traffic
  - Attempt to crack other users' credentials

Acceptable Use Policy

- All Authorised Users will be lawful, efficient, economical and ethical in their use of the Universitys Information Technology Resources. Authorised Users, shall so far as possible:
  - respect the rights of all users;
  - ensure Information Technology Resources and related physical resources are used for purposes authorised by the University;
  - ensure the security and integrity of Information Technology Resources; and
  - ensure Information Technology Resources are used in a way which complies with all relevant laws, subordinate legislation of the University, and contractual obligations governing the use of Information Technology Resources.

Introduction to Information Security

Introduction to Information Security And Cryptology

What is "security"?

What is "information security"?

Discuss. . .

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)
- Integrity

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)
- Integrity
- Authenticity of origin

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)
- Integrity
- Authenticity of origin
- Entity authentication

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)
- Integrity
- Authenticity of origin
- Entity authentication
- Non-repudiation

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)
- Integrity
- Authenticity of origin
- Entity authentication
- Non-repudiation
- Anonymity

MACQUARIE University

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)
- Integrity
- Authenticity of origin
- Entity authentication
- Non-repudiation
- Anonymity
- Fairness

MACQUARIE University

Security Properties

Information may possess — and we wish to preserve — certain *security properties*:

- Confidentiality (Secrecy, Privacy)
- Integrity
- Authenticity of origin
- Entity authentication
- Non-repudiation
- Anonymity
- Fairness
- Availability

Confidentiality

Confidentiality is the protection of information from unauthorized access.

It may be enforced by security mechanisms such as

- *cryptography*, so that if the information can be accessed, it still cannot be understood
- *access controls*, which allow access only by authorized subjects

Secrecy may be further refined into

- Secrecy
- Privacy - or at least, one perspective on privacy

MACQUARIE University

Integrity

Protection of information, systems and processes from intentional or accidental unauthorized changes.
Information can include business information, but also system configuration metadata and logs which provide *accountability*.
Assurance of accuracy and reliability of both information and systems.

Authentication of origin

If Bob receives a message from Alice, he must be confident that Alice

- sent exactly that message
- at some time in the recent past
- and in some cases, the stronger requirement that
  - Alice intended the message for Bob

Entity authentication

Message authentication provides no timeliness guarantees, but entity authentication typically involves real-time verification of a claimed identity while the verifier waits.
This is similar to authentication of origin but with an additional concept of timeliness and session management — e.g. identification of the individual logged in *right now*.
This is also known as *identification* or *identity verification*.

MACQUARIE University

Non-repudiation

The converse of authentication.
Authentication confirms what an entity claims about their identity.
Non-repudiation prevents claims of incorrect or non-identity, i.e. the ability to deny having taken certain actions (e.g. placing an order, authorizing a payment).

Anonymity

Anonymity is important in, for example, democratic voting protocols.
Given a set of events, E, indexing users, the system is anonymous if an observer can tell that an event has occurred, but cannot identify which (e.g. somebody voted for a particular party, but not who).

Fairness

Fairness is important when cryptographically signing agreements.
In the real world, the parties sit at a table, each sign one copy of a contract, exchange contracts and sign the other.
No participant in a contract-signing protocol should be able to gain an advantage over the other by halting the protocol before its proper completion.
We achieve this through reference to a Trusted Third Party, or by using probabilistic techniques.

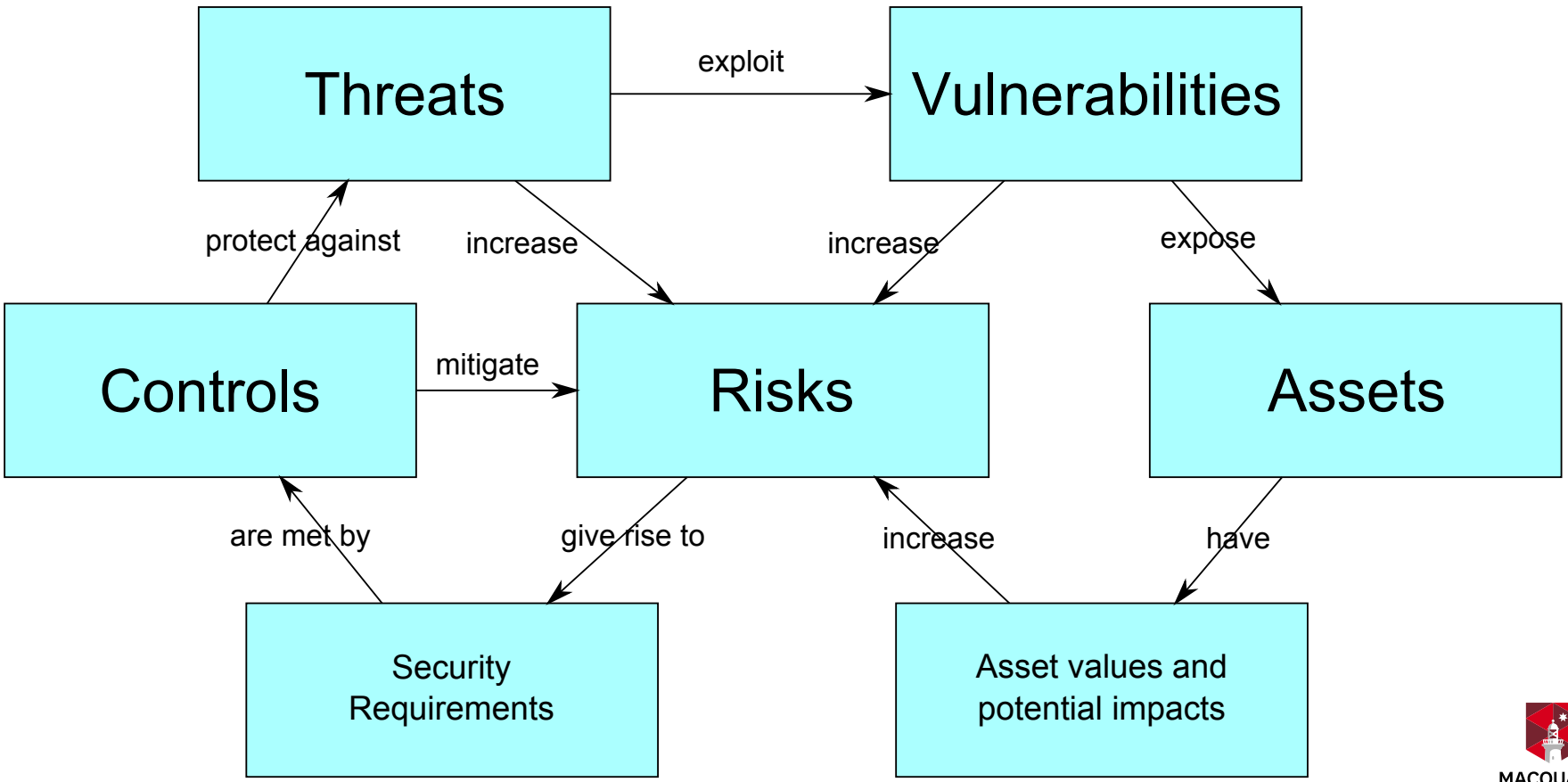
Availability

In the broader field of information security, this means that systems are running, are connected via networks, and can provide the required service in a predictable manner with acceptable performance.
For system and network administrators, this means *uptime*, though they tend to think of this more as stability rather than security.

Development of Algorithms

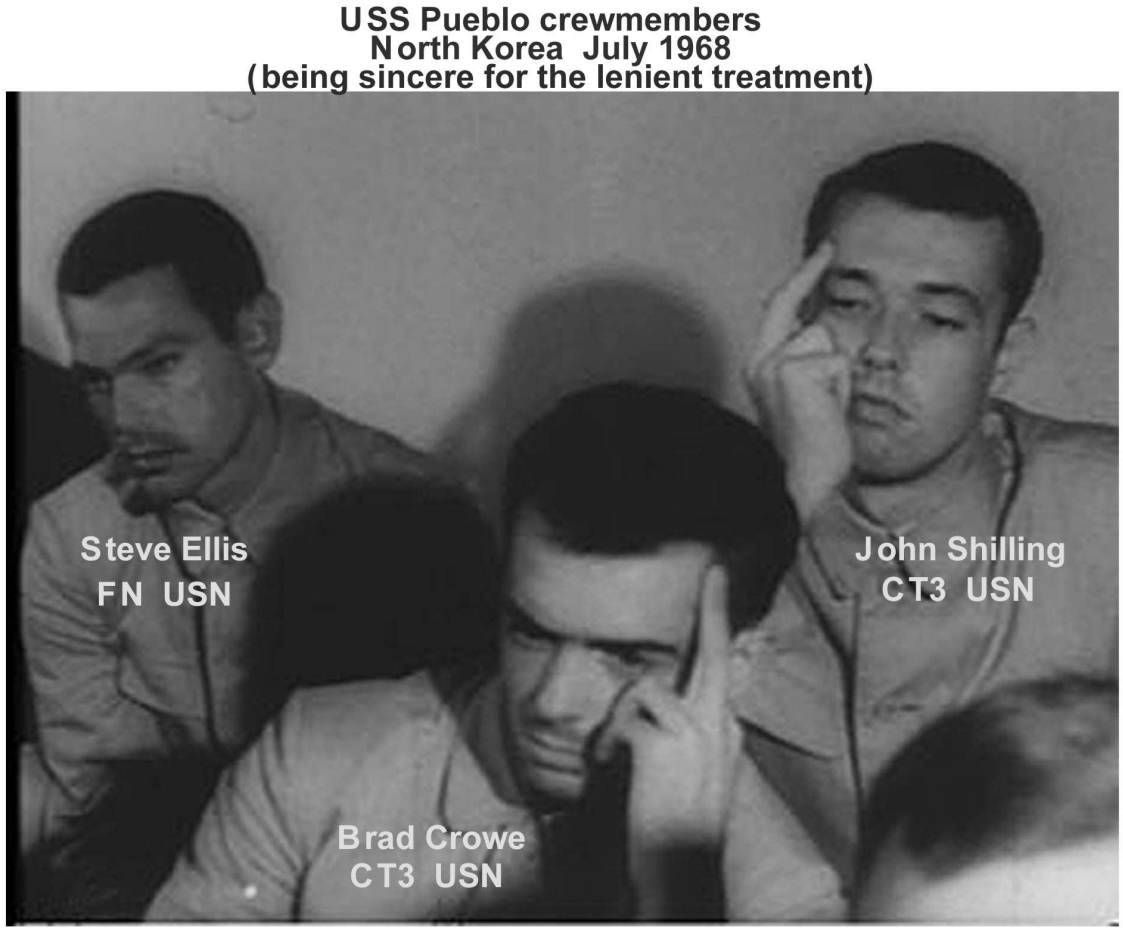
- Black-box security
  - Develop neat algorithms and try to keep them secret
  - Inevitably fails
    - Most algorithms developed this way have weaknesses which will be discovered and exploited
    - Violates Kerckhoff's Second Principle
- Crystal-box security
  - Publish your algorithms and subject to community review
    - Weak algorithms do not survive
  - Implement as open-source software
    - Weaknesses in implementation will be exposed - and fixed

Definitions



MACQUARIE University

Steganography



Introduction to Cryptology

Cryptology really encompasses two fields of research:

- Cryptography - the development of cryptographic techniques
- Cryptanalysis - the breaking of cryptographic techniques.

Cryptography

The name comes from Greek roots meaning "secret writing".
Cryptography itself has two sub-disciplines:

- The development of *ciphers* or *cryptoprimitives*.
- The development of security *protocols* which use the cryptoprimitives to satisfy a security property.

Cryptoprimitives are mostly based upon a range of mathematical problems, generally chosen from the field of discrete mathematics — i.e. performing arithmetic with integers only.
Protocols define who should encrypt what, when and send it where, in order to satisfy some combination of security properties.

Cryptographic Protocols

- Are almost as important as the cryptoprimitives
- Specify what to encrypt, how, when and where in order to satisfy security properties
- Can be quite complex (e.g. electronic voting protocols)
  - Notoriously difficult to design and prove correct
    - c.f. the original Needham-Schroeder Secret-Key protocol
    - Kerberos is an improved derivative of this
  - Advice: leave the development of these to professionals and use the facilities provided by platforms (operating systems, etc.)
  - COMP2300 provides an introduction (only) to protocols

Disciplines Involved in Cryptology

- Probability and statistics
- Algebra
- Number Theory
- Combinatorics
- Signal processing (in steganography, fingerprinting)
- Electronics (in side channel attacks)
- Physics (in quantum computers and crypto)

Disciplines Involved in Information Security

Management (policy, security awareness, security management)

Psychology (social engineering - both positive and negative)

Law investigation, forensics

Risk management

Networking (firewalls, intrusion detection/prevention systems, etc.)

Software development security

Physical security (building construction, CCTV, alarms, etc.)

Disaster recovery planning & business continuity planning

Some Definitions

Plaintext The original format of the message

Ciphertext The encrypted form of the message

Key A parameter, input to the encryption or decryption processes (or both)

Types of Cryptoprimitives

	Block ciphers	Stream ciphers
Symmetric (Secret-key)	DES/DEA IDEA CAST-5 (CAST-128) RC5 & RC-6 Blowfish & Twofish AES (Rijndael)	Caesar cipher One-time pad SEAL, WAKE RC4 A5/1 Trivium
Public-key	Diffie-Hellman key agreement RSA DSA El Gamal Elliptic Curve	
Hashes	MD4, MD5 SHA-1 SHA-2 (256, 384, 512) SHA-3 (Keccak)	

Basic Concepts

- A *symmetric (or secret-key) encryption scheme* can be seen as a triplet:
  - encryption algorithm converting messages into cryptograms
  - decryption algorithm recovering messages form cryptograms
  - key generation algorithm that produces a single cryptographic key and distributes it to the parties (the key for encryption and decryption is the same)
- A *public-key encryption scheme* is similar to private-key one except that
  - the encryption key is public
  - while the decryption key is secret
- A *hash function* is an algorithm that produces a short *digest* (of a fixed length, say 256 bits) such that:
  - it works for any message (very long but also very short)
  - finding another message with the same digest is an intractable problem
- A *signature scheme* is an algorithm that allows a recipient of a signed message to verify that the message was not tampered with and comes from the claimed sender and not another

Comparing Cryptoprimitives

- Applicability
  - Symmetric vs public key
  - Block vs stream
- Strength
  - Dependent upon complexity or sophistication of algorithm, number of rounds applied and possible key lengths
  - Governs key lifetime
    - The more ciphertext we generate, the more likely that an attacker can compromise the key
- Performance
  - Will impact on utility, e.g. slow/complex algorithms may not be usable on mobile or low-powered devices
- Key management

Work Factor

The strength of a cryptoprimitive is expressed in terms of its *work factor*.

This is the minimum amount of work – expressed as operations or clock cycles – required to determine the key value. Given n ciphertexts, we express this as $W_d(n)$.

We cannot find a sufficiently large lower bound for the work factor with public-key cryptosystems, and so will use the *historical work factor*, $\overline{W_d(n)}$ ([MVOV97, S.1.13.4]).

Cryptographic Cast of Characters

Alice	first participant in a protocol
Bob	second participant in a protocol
Carol	participant in three and four-party protocol
Dave	participant in four-party protocols
Eve	eavesdropper
Mallory	a malicious, active attacker
Trent	a trusted third party arbitrator
Peggy	a prover
Victor	a verifier

MACQUARIE University

About Alice

Now most people in Alice's position would give up. Not Alice. She has courage which can only be described as awesome. Against all odds, over a noisy telephone line, tapped by the tax authorities and the secret police, Alice will happily attempt, with someone she doesn't trust, whom she cannot hear clearly, and who is probably someone else, to fiddle her tax returns and to organize a coup d'etat, while at the same time minimizing the cost of the phone call.

Cryptography

> *"A [cryptographer] is someone who doesn't think Alice is crazy."*

John Gordon, "The Alice and Bob After-Dinner Speech" (1984), available online at
`http://downlode.org/Etext/alicebob.html`

Adversary Capabilities

Eve Can eavesdrop, i.e. sniff packets, but cannot modify them

Mallory The 'man in the middle' - can remove, reorder, reinsert and sometimes modify packets

Adversary Intentions

What might the adversary want to do?

- Read an encrypted message
- Obtain a key so she can read multiple past messages
- Obtain a key so she can read future messages
- Change encrypted messages to obtain a benefit
- Pretend to be Alice or Bob
- Find out who Alice and Bob really are
- . . . ?

Classes of Adversary

Script kiddies Not particularly technical, opportunistic, socially motivated

Hackers Technically curious, but not persistent. Might stumble across your systems, break in and play

Crackers Motivated by money. May steal credit card and identity information, trade via carding sites.

Criminal Syndicates Operate botnets which are used for phishing/pharming, spamming or cybercash mining.

Advanced Persistent Threats

The Weakest Link Property

- A security system is only as strong as its weakest link
  - People often miss the weakest link
  - You need to learn to think like an attacker
- Develop a *Security Mindset*
  - Assignment 2 is designed to develop this

Attacks on Cryptoprimitives

Ciphertext-only Attack Attacker has been able to sniff or otherwise obtain ciphertext only

Known-plaintext Attack Attacker has samples of plaintext and corresponding ciphertext

Chosen-plaintext Attack Attacker (often an insider) is able to choose the plaintext of messages and capture the ciphertext

Adaptive Chosen-plaintext Attack A variant of the above, in which the attacker generates new plaintexts based on the captured ciphertext

Chosen-ciphertext Attack An unusual attack, in which the attacker is able to choose ciphertext to be decrypted under his control

Adaptive Chosen-ciphertext Attack Another variant

We will also discuss other specialised attacks, such as the *Birthday Paradox Attack*, *Meet In The Middle Attack*, etc.

Attacks on Protocols

- Replay Attack
- Impersonation or Masquerading
- Dictionary Attack and Forward-Search Attack
- Man-in-the-middle Attack
- Oracle Attack
- Interleaving Attack

The Adversarial Setting

- Attackers are adaptive
  - Unlike "natural" threats
    - Which is why some say qualitative risk analysis is impossible
  - They are also intelligent, clever, malicious and devious
    - And numerous
    - And we may be oblivious, until it is too late
- The asymmetric nature of cybersecurity and cyberwarfare
  - The attacker only has to get lucky once — we have to be lucky *every time*.

MACQUARIE University

Professional Paranoia

- In designing systems, it can be difficult to identify threats
- Assume the adversary is everywhere
- e.g. in an electronic payments system
  - The customer
  - The merchant
  - The customer's bank
  - The merchant's bank
  - The networks between them

The Threat Model

- What are you trying to protect?
- From whom?
- What are the assets and their values?
- What are the threats?
- Often, people miss the obvious
  - E.g. firewalls don't protect against insider attacks

Claude Shannon & Information Theory

How much information does a message contain?
This depends upon the *probability* of that message.
Suppose four weather reports are possible, but the forecast was for sunny weather:

Sunny	$p = 0.70$
Cloudy	$p = 0.15$
Overcast	$p = 0.10$
Rain	$p = 0.05$

A report of "Rain" would be more surprising than a report of "Sunny", and would therefore contain more information.
This was investigated by Claude Shannon at Bell Labs in 1947.

Shannon's Measure of Entropy

Shannon decided that the information content of a message – its *entropy* – should be measured as:

$$H = -\sum_{i=1}^{i} p_i log_2(p_i) \quad (1)$$

So, for the *average* weather report in the previous slide, we have:

$$H = -(0.7log_2(0.7) + 0.15log_2(0.15) + 0.1log_2(0.1) + 0.05log_2(0.05)) \quad (2)$$
$$= 1.319 \quad (3)$$

Rate

The *rate* of a language – the average rate of information transmission in a language – is given by:

$$r = H(M)/N$$

where N is the length of the message, M. For English, this is between 1.0 and 1.5 bits per letter, for large values of N (long messages) — 1.3 is a useful estimate.

The *absolute rate* of a language is the maximum number of bits that can be coded in each character, assuming each character sequence is equally likely. If the language has L characters, this is:

$$R = log_2 L$$

For English, this is therefore $log_2 26$ or 4.7 bits/letter.

Redundancy

The *redundancy* of a language is given by the difference between the absolute rate and the actual rate:

$$D = R - r$$

So, for English, this is 4.7 - 1.3, or 3.4 bits per character.

Claude Shannon estimated that English is 50% redundant, meaning that roughly "half of what we write is determined by the structure of the language and half is determined freely".

Zero redundancy - any 2D array of letters is valid.
High redundancy - crossword puzzles are not possible.
50% redundancy - large crossword puzzles are just possible!
33% redundancy - three-dimensional crossword puzzles should be possible!

Unicity Distance

Shannon defined this as an approximation of the amount of ciphertext such that the sum of the real information (entropy) in the corresponding plaintext plus the entropy of the encryption key equals the number of ciphertext bits used. Any ciphertext longer than this distance is reasonably certain to have only one meaningful decryption.

$$U = log_2(K)/D$$

where K is the number of possible keys and D is the redundancy per letter of the message.

Unicity distance is not a measure of how much ciphertext is required for cryptanalysis, but how much ciphertext is required for there to be only one reasonable solution.

MACQUARIE University

Unicity Distance (cont)

Unicity distance is inversely proportional to redundancy. If redundancy is zero, even a trivial cipher can be unbreakable with a ciphertext-only attack.

Key Length [in bits]	Unicity distance [in characters]
40	5.9
56	8.2
64	9.4
80	11.8
128	18.8
256	37.6

Table 1: Unicity Distances of ASCII Text Encrypted with Algorithms of Varying Key Lengths

The Scytale

The Scytale doesn't change any of the characters in the message - it only changes their order.

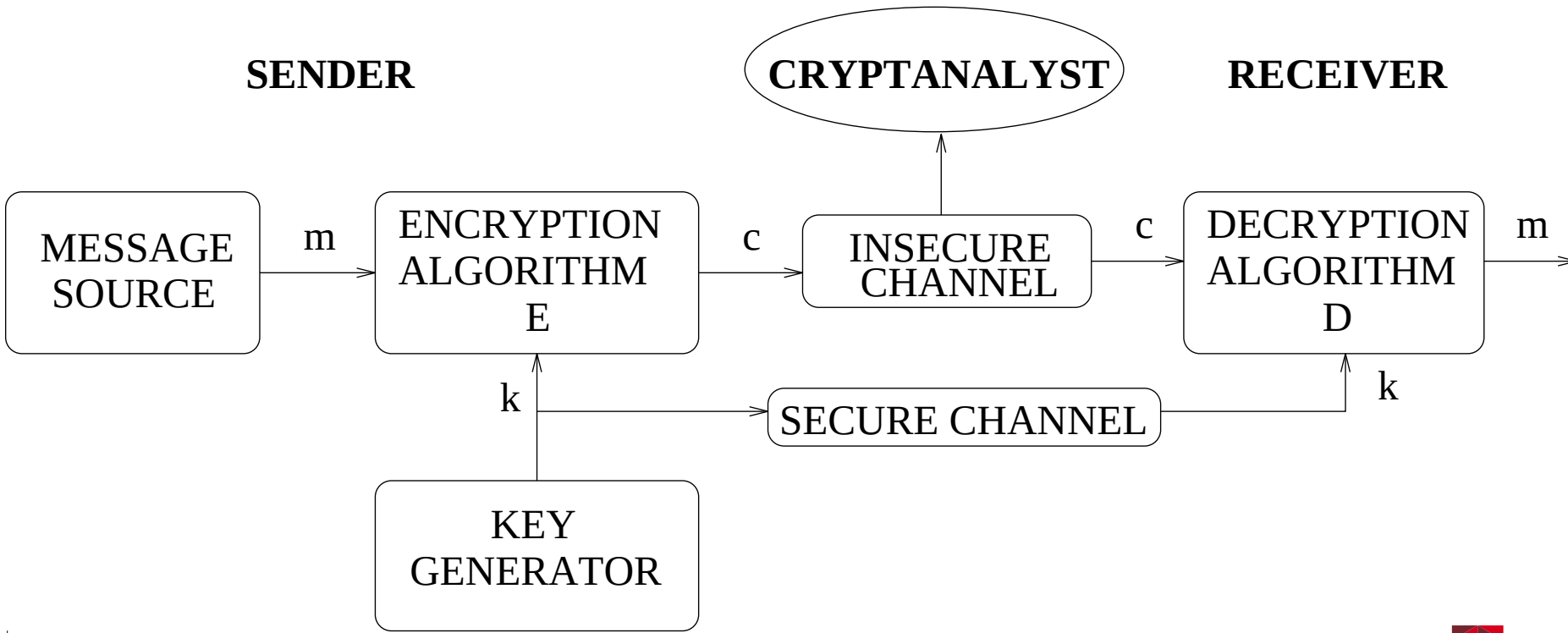
Kerckhoff's Maxims

From "*La Cryptographie Militaire*" (1883):

1. The system should be, if not theoretically unbreakable, unbreakable in practice.
2. Compromise of the system should not inconvenience the correspondents.
3. The method for choosing the particular member (key) of the cryptographic system to be used should be easy to memorize and change.
4. Ciphertext should be transmittable by telegraph.
5. The apparatus should be portable.
6. Use of the system should not require a long list of rules or mental strain.

Note especially 2, also rephrased by Claude Shannon: "The enemy knows the system".

Secret-Key (Symmetric) Cryptography



Caesar Cipher

Message Space: $\mathcal{M} = \{0, 1, \ldots, 25\}$ – letters converted to their positions in the alphabet.

Cryptogram Space: $\mathcal{C} = \{0, 1, \ldots, 25\}$

Key Space: $\mathcal{K} = \{0, 1, \ldots, 25\}$, $\mid \mathcal{K} \mid = 26$

Encryption: $c = E_k(m) = m + k \pmod{26}$.

Decryption: $m = D_k(c) = c - k \pmod{26}$.

Unicity Distance: $N \approx \frac{H(K)}{D} \approx 2$ (actually 1.47) letters ($D = 3.2$).

Cryptanalysis: Uses letter frequency distributions. If encipherment is achieved by a simple letter shift then a frequency count of the letter distributions in the ciphertext will yield the same pattern as the original host language of the plaintext but shifted.

Example - Caesar Cipher

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

Encrypt message= "MACQUARIE" $\rightarrow (12, 0, 2, 16, 20, 0, 17, 8, 4)$
using key= "k" $\rightarrow 10$

Encryption

	M	A	C	Q	U	A	R	I	E
$m \rightarrow$	12	0	2	16	20	0	17	8	4
$c = m + k \mod 26 \rightarrow$	22	10	12	0	4	10	1	18	14
	W	K	M	A	E	K	B	S	O

Decryption

	W	K	M	A	E	K	B	S	O
$c \rightarrow$	22	10	12	0	4	10	1	18	14
$m = c - k \mod 26 \rightarrow$	12	0	2	16	20	0	17	8	4
	M	A	C	Q	U	A	R	I	E

Affine Ciphers

Generalize modular arithmetic with a multiplier

$$C = (aM + b) \mod 26$$

The key is the ordered pair (a,b).

For example K = (1,5) is the Caesar cipher.

For the English alphabet, a and 26 must be coprime (i.e. gcd(a,26) = 1). So all odd values between 1 and 25 will work, except 13.

Affine Cipher

Message Space: $\mathcal{M} = \{0, 1, \ldots, 25\}$ – letters converted to their positions in the alphabet.

Cryptogram Space: $\mathcal{C} = \{0, 1, \ldots, 25\}$.

Key Space: $\mathcal{K} = \{k = (k_1, k_2) \mid k_1, k_2 \in \{0, 1, \ldots, 25\}, gcd(k_1, 26) = 1\}$, $\mid \mathcal{K} \mid = 312$, $H(K) \approx 8.3$.

Encryption: $c = E_k(m) = k_1 m + k_2 \pmod{26}$.

Decryption: $m = D_k(c) = (c - k_2)k_1^{-1} \pmod{26}$

Unicity Distance: $N \approx \frac{H(K)}{D} \approx 2.6$ letters $(D = 3.2)$.

Cryptanalysis: Uses letter frequency distributions. The letter frequencies are still preserved but permuted according to the secret key $k = (k_1, k_2)$.

Affine Cipher – Example

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

Encrypt message="MACQUARIE" $\rightarrow (12, 0, 2, 16, 20, 0, 17, 8, 4)$
using key $k = (k_1, k_2) = (5, 7)$

Encryption

	M	A	C	Q	U	A	R	I	E
$m \rightarrow$	12	0	2	16	20	0	17	8	4
$c = k_1 m + k_2 \mod 26 \rightarrow$	15	7	17	9	3	7	14	21	1
	P	H	R	J	D	H	O	V	B

Decryption

	P	H	R	J	D	H	O	V	B
$c \rightarrow$	15	7	17	9	3	7	14	21	1
$m = (c - k_2)k_1^{-1} \mod 26 \rightarrow$	12	0	2	16	20	0	17	8	4
	M	A	C	Q	U	A	R	I	E

Note that $k_1^{-1} = 21 \mod 26$.

Logic

You *must* be familiar with the basic logic operators, AND (conjunction, $\wedge$), OR (disjunction, $\vee$) and exclusive-OR (exclusive-disjunction, $\oplus$ or $\veebar$):

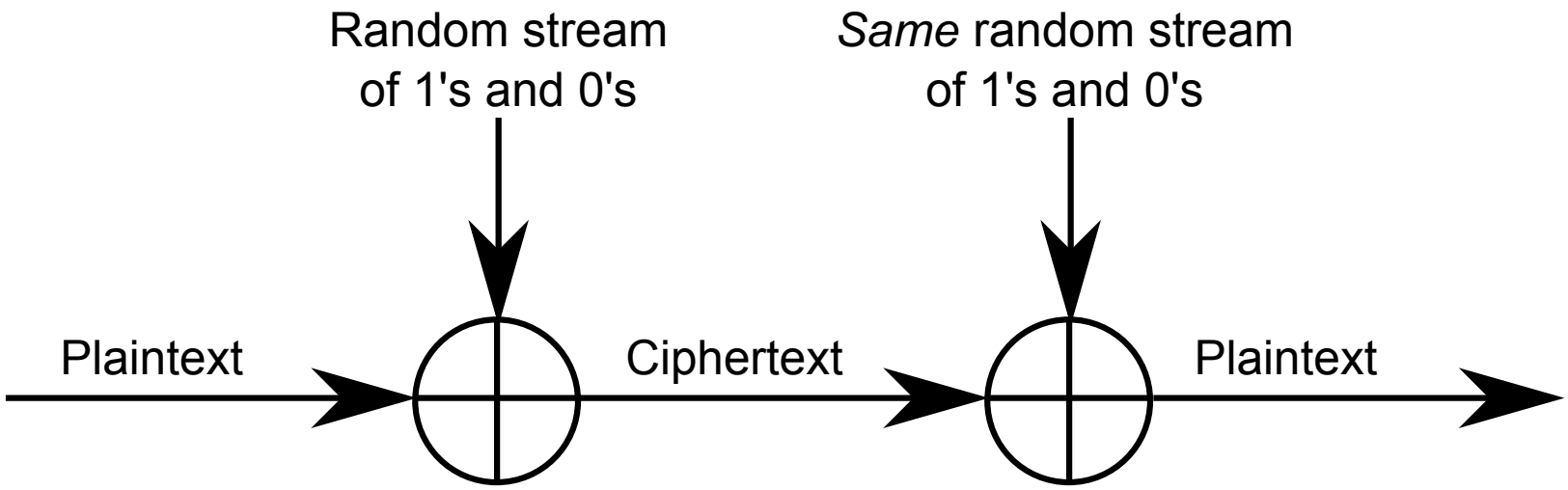
A	B	$A \wedge B$	$A \vee B$	$A \oplus B$
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

You should also know how to perform these operations on both boolean values and bit-strings in your chosen programming language.

What is the relationship between B and $A \oplus B$?

Vernam Encryption

This is simple XOR'ing of the plaintext with the keystream
Is self-inverse



Shannon showed that we can achieve perfect secrecy with this scheme, *if* the keystream is genuinely random - but we need a secure channel to transfer the keystream from sender to receiver, in addition to the ciphertext (but hold that thought!).

MACQUARIE University

Number Theory

For details, see

Smart, *Cryptography Made Simple* [Sma16], Chapter 1 and

Handbook of Applied Cryptography [MVOV97], Section 2.4.

The Integers and Divisibility

$\mathbb{Z}$ is the set of integers, $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$

$a|b$ means a is a divisor of b

For all $a, b, c \in \mathbb{Z}$:

1. $a|a$
2. If $a|b$ and $b|c$ then $a|c$
3. If $a|b$ and $a|c$ then $a|(bx + cy)$ for all $x, y \in \mathbb{Z}$
4. If $a|b$ and $b|a$ then $a = \pm b$

MACQUARIE University

Long divisibility of Integers

For $a, b \in \mathbb{Z}$ and $b \geq 1$:

$$\frac{a}{b}$$

yields

- an integer q - the *quotient*
- an integer r - the *remainder*

such that

$$a = qb + r, 0 \leq r < b$$

MACQUARIE University

div and mod

For $a, b \in \mathbb{Z}$ and $b \neq 0$

$$q = a \text{ div } b = a/b$$

$$r = a \text{ mod } b = a - b(a/b)$$

So, for example:

$73 \text{ div } 17 = 4$

$73 \text{ mod } 17 = 5$

Modular Arithmetic

If a and b are integers, and if n divides $(a - b)$, then a is *congruent* to b, modulo n.

$$a \equiv b \pmod{n}$$

For all $a, a_1, b, b_1, c \in \mathbb{Z}$, the following are true:

$a \equiv b \pmod{n}$ iff a/n and b/n have the same remainder

$a \equiv a \pmod{n}$ – Reflexivity

If $a \equiv b \pmod{n}$ then $b \equiv a \pmod{n}$ – Symmetry

If $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$ then $a \equiv c \pmod{n}$ – Transitivity

If $a \equiv a_1 \pmod{n}$ and $b \equiv b_1 \pmod{n}$ then $a + b \equiv a_1 + b_1 \pmod{n}$ and $ab \equiv a_1 b_1 \pmod{n}$

Equivalence Classes and Residues

The *equivalence class* of an integer a is the set of all integers congruent to a modulo n.

Each integer a is congruent modulo n to a unique integer $0 \leq a \leq n-1$, which is called the *least residue* of a modulo n. If $a = qn + r$, then $a \equiv r \pmod{n}$ and r is the least residue.

$\mathbb{Z}_n$ is the set of (equivalence classes of) the integers $\{0, 1, 2, \ldots, n-1\}$. Addition, subtraction and multiplication in $\mathbb{Z}_n$ are performed modulo n.

Euler's ϕ Function

$\phi(n)$ is the cardinality of the subset of integers, $\{a = 1, 2, .., n-1\}$ which are co-prime to n, i.e. for which $\gcd(a, n) = 1$.

Consider the possible keys (k_1, k_2) for the affine cipher, which requires that $\gcd(k_1, 26) = 1$. All candidate even values of k_1 will not work, and nor will $k_1 = 13$. There are only 12 possible values for k_1, because $\phi(26) = 12$.

For a prime, p, $\phi(p) = p - 1$.

The ϕ function is multiplicative: For a composite, n, which is the product of two primes, p and q,
$\phi(n) = \phi(p)\phi(q) = (p-1) \cdot (q-1)$

E.g. since $26 = 2 \cdot 13$, $\phi(26) = \phi(2)\phi(13) = 1 \cdot 12 = 12$.

See [Sma16, S 1.1.3]

MACQUARIE University

$\mathbb{Z}_n^*$

$\mathbb{Z}_n^*$ is the *multiplicative group* of $\mathbb{Z}_n$, that is

$$\{a \in \mathbb{Z} | \gcd(a, n) = 1\}$$

In particular, if n is a prime, then

$$\mathbb{Z}_n^* = \{a | 1 \leq a \leq n - 1\}$$

The *order* of $\mathbb{Z}_n^*$ is defined to be the number of elements in $\mathbb{Z}_n^*$, i.e. $|\mathbb{Z}_n^*|$. It follows that $|\mathbb{Z}_n^*| = \phi(n)$.

Note also that if $a \in \mathbb{Z}_n^*, b \in \mathbb{Z}_n^*$, then $a \cdot b \in \mathbb{Z}_n^*$, so that $\mathbb{Z}_n^*$ is closed under multiplication.

Multiplicative Inverse and Division

Let $a \in \mathbb{Z}$. The *multiplicative inverse* of a modulo n is an integer $x \in \mathbb{Z}$ such that $ax \equiv 1 \pmod{n}$. *If* such an x exists, then it is unique, and a is said to be *invertible*, or a *unit*. The inverse of a is a^{-1}.

Let $a, b \in \mathbb{Z}_n$. *Division* of a by b modulo n is multiplication of a by b^{-1} modulo n, and is only defined if b is invertible modulo n.

Invertibility

Let $a \in \mathbb{Z}_n$. Then a is invertible if, and only if, $\gcd(a, n) = 1$.

Example: The invertible elements in $\mathbb{Z}_9$ are 1, 2, 4, 5, 7 and 8.

E.g. $4^{-1} \pmod 9 = 7$ because $4 \cdot 7 \equiv 1 \pmod 9$.

Or $7^{-1} \pmod{11} = 8$ because $7 \cdot 8 \equiv 1 \pmod{11}$.

MACQUARIE University

More Generally

More generally:

Let $d = \gcd(a, n)$. The congruence equation

$$ax \equiv b \pmod{n}$$

has a solution x if and only if d divides b, in which case there are exactly d solutions between 0 and $n - 1$; these solutions are all congruent modulo n/d.

MACQUARIE University

Euler's Phi

Fermat's Little Theorem

If $\gcd(a, p) = 1$, then $a^{p-1} \equiv 1 \pmod{p}$.

Also, if $r \equiv s \pmod{(p-1)}$, then $a^r \equiv a^s \pmod{p}$ for all integers a. In other words, when working modulo a prime p, exponents can be reduced modulo $p - 1$. In particular, $a^p \equiv a \pmod{p}$ for all integers a.

Euler's Generalization of Fermat's Little Theorem

If $a \in \mathbb{Z}_n^*$, then $a^{\phi(n)} \equiv 1 \pmod{n}$.

Also, if n is a product of distinct primes, and if $r \equiv s \pmod{\phi(n)}$, then

$$a^r \equiv a^s \pmod{n}$$

for all integers. In other words, when working modulo such an n, exponents can be reduced modulo $\phi(n)$.

Euclid's Algorithm

Let a and b be positive integers such that $a \geq b$. The following algorithm computes $\gcd(a, b)$ in a finite number of steps.

```
1: procedure EUCLID(a, b)                    ▷ The g.c.d. of a and b
2:     r ← a mod b
3:     while r ≠ 0 do                        ▷ We have the answer if r is 0
4:         a ← b
5:         b ← r
6:         r ← a mod b
7:     end while
8:     return b                              ▷ The gcd is b
9: end procedure
```

Euclid's Algorithm in BASIC

And here is Euclid's Algorithm, implemented for the HP-71B pocket computer:

```
10 ! Euclid's Algorithm
20 INTEGER A,B,Q,R
30 INPUT "A? ";A @ INPUT "B? ";B
40 'LOOP': Q=A DIV B @ R=RMD(A,B)
50 DISP A;"=";B;"*";Q;"+";R @ PAUSE
60 A=B @ B=R
70 IF R >= 1 THEN GOTO LOOP
80 DISP "GCD =";A
```

Having to use `GOTO` is painful! And it only handles relatively small integers ...

But it could be worse...

On the HP-16C "Computer Scientist" calculator, in decimal mode:

```
001 g LBL A
002 STO 1
003 Rv
004 STO 0
005 g LBL 0
006 RCL 0
007 g PSE
008 RCL 1
009 g PSE
010 /
011 g PSE
012 RCL 0
013 RCL 1
014 f RMD
015 g PSE
016 1
017 g x>y
018 GTO 1
019 Rv
020 RCL 1
021 STO 0
022 Rv
023 STO 1
024 GTO 0
025 g LBL 1
026 RCL 1
027 g RTN
```

Extended Euclid's Algorithm

1: **procedure** ExtEuclid(a, b) ▷ The g.c.d. of a and b
2: $(c, d) \leftarrow (a, b)$
3: $(u_c, v_c, u_d, v_d) \leftarrow (1, 0, 0, 1)$
4: **while** $c \neq 0$ **do**
5: **Invariant:** $u_c a + v_c b = c \wedge u_d a + v_d b = d$
6: $q \leftarrow \lfloor d/c \rfloor$
7: $(c, d) \leftarrow (d - qc, c)$
8: $(u_c, v_c, u_d, v_d) \leftarrow (u_d - qu_c, v_d - qv_c, u_c, v_c)$
9: **end while**
10: **return** (d, u_d, v_d) ▷ The gcd is d and $\gcd(a, b) = u_d a + v_d b$
11: **end procedure**

Why Is This Important?

Because it's another way to find the multiplicative inverse.

If we want $1/b \pmod{p}$ where $1 \leq b < p$ then we can use the Extended Euclid Algorithm to calculate $\gcd(b, p)$ which must, by definition, be 1.

So we have

$$ub + vp = \gcd(b, p) = 1 \tag{4}$$
$$ub = 1 - vp \tag{5}$$
$$ub = 1 \pmod{p} \tag{6}$$
$$\therefore u = b^{-1} \tag{7}$$

MACQUARIE University

A Final Fact to Remember

If n is prime, we know that $\phi(n) = n - 1$. In addition, if $n = p \cdot q$, where p and q are prime, then $\phi(n) = (p - 1) \cdot (q - 1)$.

You are going to see $(p - 1)(q - 1)$ *a lot* as we discuss public key cryptography, and this is the reason why. The RSA public key cryptosystem is based on the selection of two primes, p and q, which are multiplied to produce the modulus, n.

An RSA key must be invertible modulo $\phi(n)$ - but because $n = pq$, $\phi(n) = (p - 1)(q - 1)$, and when Alice chooses a key, she can test that it is invertible using Euclid's Algorithm and find the inverse using the Extended Algorithm.

Introduction to PARI/GP

PARI/GP is a widely used computer algebra system designed for fast computations in number theory (factorizations, algebraic number theory, elliptic curves . . .), but it also contains a large number of other useful functions to compute with mathematical entities such as matrices, polynomials, power series, algebraic numbers etc., and a lot of transcendental functions.

PARI/GP has an interactive REPL command line which is useful for testing ideas and checking results, but it also has scripting features which can be used for more complex tasks.

Starting PARI/GP

PARI/GP is installed in the COMP3000 lab, so you should find it on the start menu, probably as Pari-2-11-1 or similar. The "gp" program is an interactive shell for PARI.

If you try a calculation, by default the result will give 28 or 38 digits of precision.

You can change this with "\p *n*" where *n* is the desired precision.

Functions

PARI has lots of useful functions for number theory

random() - Generate a random number

print() - Output an argument

frac(x) - Fractional part of x

ceil(x) - Ceiling of x

floor(x) - Floor of x

gcd(x,y) - Greatest common divisor of x and y

lcm(x,y) - Lowest common multiple of x and y

isprime(x) - Is x a prime?

eulerphi(n) - Euler ϕ function

gcdext(x,y) - Returns u, v, d such that $d = \gcd(x, y)$ and $ux + vy = \gcd(x, y)$

Intmods

PARI has an INTMOD type (t_INTMOD) which allows it to perform arithmetic in $\mathbb{Z}$ and $\mathbb{Z}_n$ (explained a few slides back). To create an integer a (mod b), type "Mod(a,b)" – do *not* use $a\%b$ (the usual modulo operator in programming languages). Internally, PARI will perform all arithmetic on integers in the interval $[0, b-1]$.

Programming PARI

PARI has flow control statements which are similar to, but syntactically slightly different from, those in C and Java.

- while(a, expressions) - While a is true, loop
- until(a, expressions) - Until a is true, loop
- forstep(X = a, b, s, expressions) - Loop with X from a to b in increments of s
- forprime(p = a, {b}, expressions) - Loop over all the primes between a and b
- break(n) - Break out of n levels of loops

See Section 3.11 of the "User's Guide to Pari/GP".

Applied Cryptography

- In line with the unit title, we shall be applying the theory of cryptography to real-world problems:
  - Securing your daily work (sensitive files, emails)
  - Developing secure applications software
- For application development, especially, we need a more general and more powerful language than PARI
- We'll use Java - and in particular, the Java Cryptography Architecture:
  - It's comprehensive: full suite of cryptoprimitives
  - Fast enough to do interesting things
  - Standard language at MQ - but similar enough to others (C, C#, Python)
- Next lecture: Number theory and using crypto libraries in Java, and intro to Assignment 1

MACQUARIE University

Bibliography

Ross J. Anderson.
Security Engineering: A Guide to Building Dependable Distributed Systems.
Wiley, 2 edition, November 2010.

A. J Menezes, Paul C Van Oorschot, and Scott A Vanstone.
Handbook of applied cryptography.
CRC Press, Boca Raton, 1997.

Nigel P. Smart.
Cryptography Made Simple.
Information Security and Cryptography. Sprinter International Publishing, Switzerland, 2016.