MATH3092: Introduction to Cryptology and The AES Method

The mathematics behind cryptography and cryptoanalysis and its applications.

Delice Mambi-Lambu

School Of Mathematical Sciences
Faculty Of Social Sciences
University Of Southampton
Supervisor : Dr. Jan Spakula
Student I.D: 30077966

Academic Year 2020/21

1 Introduction

This project and report are about understanding how basic principles in mathematics can be used and applied in the modern-day. In this case, we will be focusing on the subject area of network security. The fundamental notions and principles used in network security today were built upon the foundations set upon by cryptology. So, in this report, we will seek to learn about cryptology and how it branches off to other areas such as cryptoanalysis and cryptography. Furthermore, we will briefly look into the definitions and, generally, how cryptography works. Briefly, looking at an application of cryptology (in the historical sense). After that, we hope to further plunge into the world of cryptography – as we will argue the reasons for and against it. This will then discuss mathematical concepts used in cryptology and what is used for modern-day cryptography. Here, we will highlight the different types of methods and algorithms used today and how they relate to cryptography's critical principles. Lastly, we will then discuss a modern-day application of cryptography and how the basic principles can be used and applied to the algorithm to ensure that network security is upheld.

1.1 Further Note: Being an SLpD Student

I am an SLpD student with dyslexia. I am unaware of how that would affect the scores and/or outcome for the report. However, I would like to put this in as a caveat for how the report will go. I wrote this to the best of my ability and with help of writing computer applications from enabling services to ensure that this report for this project will be grammatically correct and fluid enough for you to read. If it suffices that there are areas where the paragraph does not make sense or the clarity of the sentence is lacking (in the grammatical sense, not mathematical or content-wise). Please understand that this was not done out of ill-will.

2 Abstract

This project and report were done through research and conversion of research of summarizing or generalizing the key points, fundamental ideas, and concepts to be a part of this report.

In the first semester of the academic year, the focus was on the critical concepts of cryptology; hence, we mainly focused on researching information rather than mathematical concepts used or implicated in cryptology. We mainly watched lectures based on a series by Chritopof Parr (of the Max Planck Institute for Security and Privacy in Bochum), but I mainly focused on the key concepts rather than researching mathematics. This led me to have a Semester 1 project that was weak and lacking weight mathematically.

Therefore, in the second semester of the academic year, I focused on the mathematics of the project. I wanted to keep my work as precise as possible, so I researched the mathematical concepts used in cryptology and the AES algorithm. Luckily, I came across two key concepts – one being modular arithmetic and the other being fields (which we will explore in the report). I have previously studied these concepts earlier in modules (i.e. Linear Algebra II, Number Theory, Group Theory). This led me to review lecture material from those years studied produced by Jan Spakula, James Renshaw, Stephen Theriault and Graham Niblo. Now, we have the mathematical material to write. The rest of the report comprised of information and further mathematical information from books, web pages, and videos (Mainly from Christof Paar and Chirag Bhalodia).

The Result – I understood how key concepts and fundamental principles from historical cryptology have evolved and how they can be applied to modern-day network security. I also learned about how mathematical concepts such as Fields and Modular Athematic can be helpful in cryptography. Also, I get to learn the advantages and disadvantages of cryptology and the effectiveness of it.

2.1 Further Note: Referencing

In this section, I will describe how the referencing will work for this report. In every section, a subscript numbering system will be used to denote that a piece of information came from a referenced part of material. At the bottom of the page as a part of the footnote of the page, there will be a list of numbers referring to the material in the section. This will also be referenced to the material that will be seen in the bibliography. A detailed summary of all sources used to write the section will be detailed in the section where no subscripts are seen.

For example, for a sentence: The type of processes is the one that uses a single piece of information for inscription and decoding.[1]

[1] This Section is from Reference [0] chapter 2

3 Introduction to Cryptology [2]

3.1 Basis Concepts

We begin to introduce the concept of **Cryptology**, cryptography and cryptoanalysis. Essentially, **Cryptology**is the study of the methods used to encrypt information from sources to targets and measuring the effectiveness of those methods. Via measuring how easy it is to intercept those messages. This is a measure of how secure a cryptosystem is.

Cryptoanalysis - First, cryptoanalysis is the science of reading secrets and breaking them. In the cryptoanalysis section we will briefly look at the historical aspect of the algorithms and the advantages and disadvantages of those algorithms. We will then explore the reasons why innovations had to be made to ensure that the algorithm becomes more efficient.

Cryptography - In a nutshell, cryptography is the art and science of concealing pieces of information to insure secrecy in information security. For this project, we will mainly be focusing on cryptography as we will be looking at the methods and how the computations of those methods are carried out.

3.1.1 Definitions

Plain-text - x - is the unencrypted data or initial values for a given data. this will later go through the encryption function e to become the ciphertext y

Cipher-text - y - This is the encrypted data that will then undergo our decryption function d to become plain-text x.

The Encryption Function - The encryption function that encrypt the plain-text x to cipher-text y.

The Decryption Algorithm - The decryption function that decrypts the cipher-text y to plain-text x.

Cipher - is an algorithm for performing encryption or decryption.

Cryptosystem - is the implementation of cryptographic techniques and their accompanying structure to ensure information security. A cryptosystem is also known as a cipher system. The structure of a basic cryptosystem are as follows:

- Plain-text
- Encryption Algorithm
- Cipher-Text
- Decryption Algorithm
- Encryption Key
- Decryption Key

3.2 Historical Cipher: Caesar Shift

Let's look at a basic example of an historical cipher and apply what we have learned in cryptology so far to analyse. The historical cipher we'll be looking at is the Caesar Shift Cipher. This cipher relies on shifting the letters by a constant number (normally three). Then, the person who would receive the scrambled piece of information would decrypt it by shifting the letters back by the same number to obtain the original piece of information. The Caesar cipher was named after Julius Caesar, who used it with a three-shift to ensure that his commands were protected when in the military. However, it is unknown if he had come up with the cipher or not.

[2] This section was comprised via the use of sources [2], [3], [5], [7] , [9] and [16]. Most notably the example given by section 3.2 is from the video lecture 1 from source [16]. The rest of a sources are used to summarize the key points for this section.

Method
$$\begin{bmatrix} Plaintext & A & B & C & D & E & F & G & H & I \\ J & K & L & M & N & O & P & Q & R & S \\ T & U & V & W & X & Y & Z & & & \\ Ciphertext & X & Y & Z & A & B & C & D & E & F \\ G & H & I & J & K & L & M & N & O & P \\ Q & R & S & T & U & V & W & & & \end{bmatrix}$$

This is a Cesar shift of 3 - where the encryption function and the negation of the encryption function is given by the equations (for any given $n \in \mathbb{Z}$).

Where the encryption function is:

$$E_n \equiv (x + n) \quad mod \quad 26$$

and the decryption function is :

$$D_n \equiv (x - n) \quad mod \quad 26$$

Example for $n = 3$

plain-text x - we attack London at twenty-three hundred hours.

cipher-text y - dwwdfn orqgrq dw wzhqwb-wkuhh kxqguhg krxuv.

Analysis The problem with this cipher is that it is very easy to break up the code and intercept the message in the shift. This is a problem many people faced during history including the Germans in WWII - where they had used a variation of the Cesar shift to encrypt their instructions to send from home base in Berlin to their war generals around the continent. Thanks to Turning and co and the invention of the enigma machine. By the method of **frequency analysis** and **brute force methods**, they were able to intercept German forces hence aiding the war effort to an eventual win. From this, we also get that out of a multitude of ways to break up code. The main two methods of doing so are **frequency analysis** and **brute force attack**.

3.3 The Debate For And Against Cryptography

3.3.1 Arguments For Cryptography

- For Privacy: This is to ensure that no person can read the messages except the receiver of said message
- For Authenticity: For a person to prove their own identity - an example of this is when making online payments through Mastercard or Visa
- For Integrity - To ensure that the receiver gets the original message that has not been altered it.
- For Non-Repudation: This shows that a sender sent a message and it clears up the false claims something was sent, when it wasn't.
- ' For Key-Exchange - where crypto-keys are shared between sender and receiver.

3.3.2 Arguments Against Cryptography

- A strongly encrypted authentic piece of information will make it very difficult to access even for a person who understands the code and how to manipulate in the unsecured channel. However, there are areas where these systems in a network are vulnerable. Generally, at the decision-making phase. The network can then be attacked by a intruder.
- If the algorithm is poorly designed and has poor protocol, there is no insurance or protections against the systems. Making them vulnerable for attacks - It goes without saying **NEVER** use a untested cryptoalgorithm.
- The study of cryptography is not cost effective. Hence, there will be times that if there is no budget, most cost friendly system of network protection might be used. Chances are the lower cost the method of cryptography is used, the lesser the quality hence making it easier for attackers to penetrate via use of brute force
- Further methods are required to guard information. As high availability is one of the fundamental concepts of information security. This cannot be guaranteed through cryptography. Hence, more procedures must be undertaken to ensure security.

3.4 Types Of Algorithms In Cryptography

Here, we will explore the types of algorithms involved in the study of cryptography. In cryptography, there are two types of algorithms. That being symmetric algorithms and asymmetric algorithms. We will mostly focus on symmetric algorithms but for the sake of discourse - lets define them both:

3.4.1 Asymmetric Algorithms

These are the algorithms in cryptography that uses multiple keys for both encryption and decryption - These algorithms are usually used to ensure that the purpose of authenticity, non-repudiation and key exchange are fulfilled. This is a more newer field in terms of development in the last 300-400 years. This is also known as 'Public Key Cryptography'. Asymmetric Algorithms are dependent on mathematical functions - which are straight forward to formulate. However, that means finding the inverse function or the decryption function more difficult to find. Examples of this can be when you must multiply values let's say $3^2 \cdot 5^1 = 45$ that's straight forward. But, expressing 45 as a product of it primes takes a longer time as we must use the algorithm of prime factor decomposition to get the intended result. This makes Asymmetric algorithms, longer and harder to compute.

Example's: Diffie-Hellman, DSS, ELGamal

3.4.2 Symmetric Algorithms

These are algorithms in cryptography that uses a single key for both encryption and decryption - These algorithms are usually used to ensure that the purpose of privacy and integrity are fulfilled. We have that it creates a fixed length of bits known as a block cipher with a secret key that the sender uses to encrypt the data, then traveling through a secure channel. The receiver then uses the *cipher-text y* received to decipher it. The algorithms that use a block cipher are referred to as 'Block Algorithms'. There is another set of algorithms that fall in the symmetric umbrella - those are called 'Stream algorithms'. These algorithms use stream ciphers and these are defined as algorithms such that they operate on just a single bit at a time and the key is constantly changing whilst the algorithm is being carried out (clearly this is more time consuming and less efficient). Hence, this results in a few problems, one being error propagation - where this would lead to the transmitted message becoming 'not clear-enough' or garbled. Given that, Block Algorithms are more efficient and effective we will be investigating these ones more. The problem with Symmetric Algorithms is that key management was difficult to secure in a safe manor as the keys travelled along a public channel. Therefore, asymmetric algorithms were created. However Symmetric algorithms are more secure and faster.

Example's: DES, AES, IDEA, RC4 (Stream), RC5 and RC6

4 Mathematical Concepts

4.1 Motivation

In this section, we will discuss the core mathematical concepts in modern-day and historical cryptography. First, we will discuss Modular Arithmetic as this is the primary key player in cryptography. The art of clock mathematics is essential to know as usually in cryptography. Many pieces of information, numbers or bits can be reorganised into a set of reoccurrences that, once defined, can be manipulated to fulfil a function – That being, the encryption and decryption of a bit. Hence, we will explore what modular arithmetic work leads to introducing more complicated concepts such as Rings and Groups. After that, we will discuss the notion of Fields in-depth as we will see that rings and groups have limitations when it comes to the computation of elements in cryptography. As Fields are one of the important mathematical concepts needed in cryptography.

4.2 Modular Arithmetic [3]

4.2.1 Finite Sets

We will now introduce the notion of a Finite Set.

Definition Let have a set of elements. That set is said to be a Finite Set if the set of elements contains the void set or we are able to use the process of counting elements to seek a limit, hence defining its finality. A finite set can be expressed in multiple ways. The number of distinct elements counted in a finite set S is denoted by $n(S)$.

For Example:

$$S = \{1, ..., n\}$$

or even

$$x_1, x_2, ..., x_n \quad and \quad (x_i \in S, 1 \leq i \leq n)$$

Examples of Finite Set

- The Alphabets A = {A, B, C,..., Z}. These contain 26 elements hence the $n(A) = 26$
- B = {1, 2, 3,...,10}. These contain 10 elements hence the $n(B) = 10$
- However, $D = \mathbb{Z}$ wouldn't be considered a finite set. This is because there is no countable value of n that is a natural number that we can say there is a limit to stop the process of counting. Hence, we say that this set is an infinite set.

We will be working with a finite number of elements and it be defined as being a part of sets. When it comes to discussing the Modular arithmetic, Groups, Rings and Field elements of cryptography.

4.2.2 Modulo Operator

Let's, now define the key concept of modular arithmetic, the *modulo operator*.

Definition Let a, r, m ∈Z and m> 0. Then it suffices that :

$$a \equiv r \mod m \tag{1}$$

Where m represents the modulo , a represents any value on the $\mathbb{Z}$ plane. and r represents the remainder that this calculated from applying the modulo operator. It also suffices that m must divide $(a - r)$ for the modulo operator to work. i.e :

$$m|(a - r) \tag{2}$$

Whereas if m does not divide $(a - r)$ the modulo operator breaks down hence we get that we end up with a value that is not in $\mathbb{Z}$ and we end up with quotient values.

[3] This section is comprised via the use of sources [5], [13], and examples are a summary from source [16] most notably lecture 2

4.2.3 Computation Of The Remainder

The remainder can be computed by following the algorithm: Given a , m , q ∈Z

$$a = qm + r \tag{3}$$

It must be noted that the remainder **is not unique** This is because there several values of q that can be played with to give a different value of the remainder r.

Example Let $a = 29$ and $m = 5$

$$29 = 5 \cdot 5 + 4, \quad r = 4, \quad (29 - 4) = 25 \quad \text{and} \quad 5|25 \tag{4}$$

Is Divisible. However, we can also have

$$29 = 4 \cdot 5 + 9, \quad r = 9, \quad (29 - 9) = 20 \quad \text{and} \quad 4|20 \tag{5}$$

Is Also Divisible.

This can occur for infinite values of q which lie in the $\mathbb{Z}$ Plane. Hence, this shows that the remainder is not unique. This is important as it helps us in introducing the next notion in modular arithmetic that being Equivalence Classes.

4.2.4 Equivalence Classes

Definition We will define the equivalence class for any modular operator: Given values of a, $m \in \mathbb{Z}$ then we define the equivalence class of $a \quad (mod \quad m)$ to be $[a]_m$. This represents the set of all integers that are congruent to $a_i \quad (mod \quad m)$.

Properties Of Equivalence Classes Let A be a non-empty set and let $\sim$ be a equivalence relation. Then,

- For a given $a \in \mathbb{Z}$, a $\in [a]$.
- For a given $a, b \in \mathbb{Z}$, $a \equiv b$ (mod n) ,if and only if $[a] = [b]$.
- For every given a, b ∈Z, [a] = [b] or [a] ∩$[b] = \emptyset$.

So, why is this important? If we can to take our large pieces of data / bits we can break them up into classes. This makes things simpler as we will be able to calculate things more efficiently. Hence, enabling to encode and decode efficiently and effectively.

Example Lets Review the Modulo Operator from the previous section again.

Let $a = 14$ and $m = 5$

$$14 \equiv 4 \quad \text{mod} \quad 5 \quad \text{and} \quad 14 \equiv 9 \quad \text{mod} \quad 5 \tag{6}$$

Which relates to:

$$14 = 2 \cdot 5 + 4 \quad \text{and} \quad 14 = 1 \cdot 5 + 9 \tag{7}$$

We see from the previous section, that for the modulus of 5 the value of 14 is relative to the remainder values of 4 and 9. This can continue for an infinite sequence of values that are relative to 14 or its base remainder of 4 so it suffices that its equivalence class turns out to be.

$$\{..., -1, 4, 9, 14, 19, 24, 29, ...\} \tag{8}$$

Thus the segment of the equivalence classes for the class [5] turn out to be:

Class	Equivalence Class	Elements
A	[0]	{..., -10, -5, 0, 5, 10, ... }
B	[1]	{..., -9, -4, 1, 6, 11 ... }
C	[2]	{..., -8, -3, 2, 7, 12 ... }
D	[3]	{..., -7, -2, 3, 8, 13, ... }
E	[4]	{..., -6, -1, 4, 9, 14, ... }

Example As you've seen previously. We have classified our equivalence classes for $(mod \quad 5)$ hence you can see that there are 5 classes for this case - I've labeled this A to E. From this we can carry out computations or operations by simplifying the number and computing it.

Hence, For $A = 20$ and $B = 24$:

$$A \cdot B = \alpha \Rightarrow \quad 20 \cdot 24 = -9 \mod 5 \equiv 1 \mod 5 \tag{9}$$

Or even - we could have a class represent different numbers. As long as the numbers belong to the same class the calculation does compute - This example has $D = 8$, 13 and $B = 16$:

$$D \cdot B - D = \beta \Rightarrow 13 \cdot 16 - 8 = 208 - 8 = 200 \equiv 0 \mod 5 \tag{10}$$

Important Application: We can get even more complicated when it comes to calculating modular equivalents for Large numbers (i.e 3^9 or 11^5).

Example lets have:

$$3^8 \quad mod \quad 7 \tag{11}$$

There are multiple ways, we can calculate this by relations to equivalence classes.

Way One: We can calculate directly via the use of the remainder theorem to it base value or remainder. This is more energy draining but nonetheless.

$$3^8 = 6561 \equiv 2 \quad mod \quad 7 \tag{12}$$

Calculated via

$$6561 \equiv q \cdot n + r \Rightarrow 937 \cdot 7 + 2 \tag{13}$$

Way Two Via the use of prime factor decomposition - this method is more energy efficient:

$$3^8 = 3^4 \cdot 3^4 = 81 \cdot 81 \quad mod \quad 7 \tag{14}$$

Applying the use of equivalence classes for $(mod \quad 7)$

$$81 \cdot 81 \quad mod \quad 7 \equiv -4 \cdot -4 \quad mod \quad 7 \tag{15}$$

So far, we have investigated mostly the operations and computations carried out in modular arithmetic and how it applies to modern day cryptography and its importance. Now, we would go on to look further into how these operations in modular arithmetic corresponds to its structure and why the structure of the set of elements used in modular arithmetic and cryptography is important in carrying out operations.

4.3 Abstract Structures

In the last section, we discussed how operations and computations can be carried out over a set of elements and how its relatable to the concept of modular arithmetic. As we've seen we seen how a closed set can have operations carried out upon it. Here we will discuss the abstract structures of the set and how it relates to the elements used in modern day cryptography.

4.3.1 Groups [4]

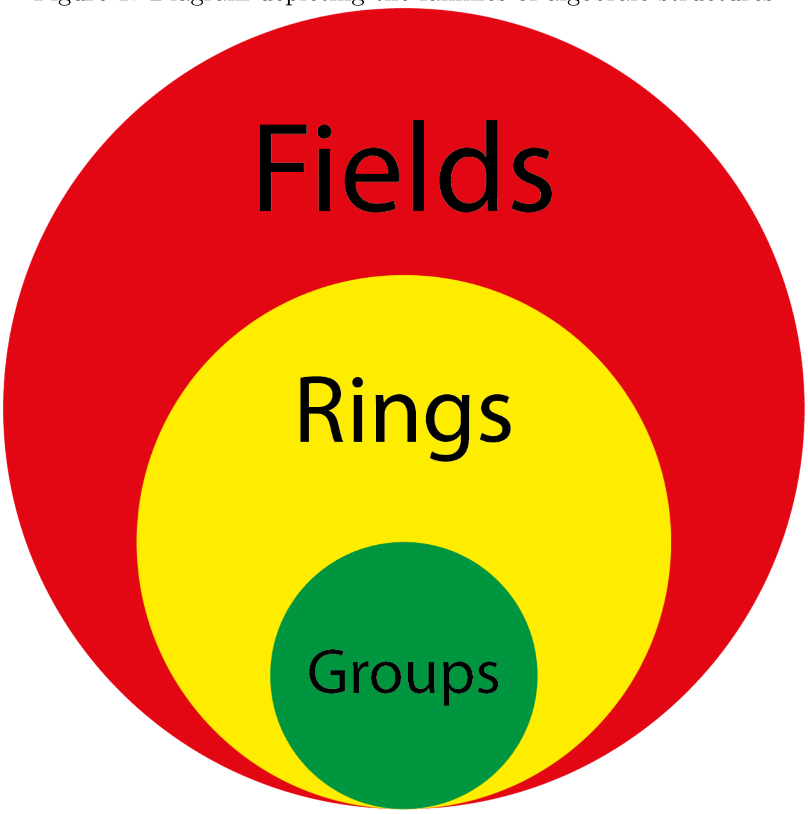
Let's introduce the base notion involved in describing structures in abstract algebra - we begin with the fundamental definition of a group:

Definition Let G be a non-empty set with the operator "*" . Therefore, it must suffice that G is a group if it suffices that if it is closed and it follows the following axioms:

1. The Group must be associative - $a * (b * c) = (a * b) * c \quad \forall a, b, c \quad \in G.$
2. The Group must have a identity element - $\exists e \in G$ such that $a * e = a = e * a \; \forall$ a∈G.
3. For every given element $a \in G$. There must exist an inverse element $a^{-1} \in G$ for every member of the group. Such that $\forall a \in G$ the operations $a * a^{-1} = e = a^{-1} * a$.

[4] This section comes from summarizing sources [5] , [13] and [15] notes. Such that the definitions, proofs and the relevant theory come from these sources

Figure 1: Diagram depicting the families of algebraic structures



Examples Of Groups

1. The set $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{C}$. Are a group under the binary operation "$+$".
2. $[a]_n$ is a group under the binary operation $+_n$. (we will touch on this after on)
3. **HOWEVER,** $\mathbb{N}$ is not a group under the binary operation $+$ as no inverse exists for such.

Basic properties of groups Here we find that the following properties hold for all groups:

- The Identity Element Is Unique
  - **Proof**: Suppose that e and f are both identities for a binary operation on a set G.Then $e = e{*}f = f$. The first equality holds because f is an identity. The second holds because e is an identity.
- The Inverse Element is unique.
  - **Proof**: Suppose that b and c are both inverses for a. Then $b = b * (a * c) = (b * a) * c = c$.The first equality holds because $a * c$ is an identity, the second because * is associative, and the third because $b * a$ is an identity.

How does the Modulo Operator Relate to Groups?

Here we will show how the modulo operator can be defined as a group up to a certain constraint we will also go on to show why we can just use groups as a basis for the structure of the finite set of elements due to its constraints. Hence, we will prove for the group $[a]_n$ and also explain it cannot work for multiplication.

Proof That the set $[a]_n$ with the binary operation '+' is a group.

- Is it closed under its binary operation ? : We know that from $\mathbb{Z}$ the set is closed under addition. Hence, given that $[a]_n$ contains elements from $\mathbb{Z}$ and also a subset of it. We can take that $[a]_n$ is also closed under addition.

- Is it associative? : We have that,

$$([a]+[b])+[c] = [a+b]+[c] = [(a+b)+c] = [a+(b+c)] = [a]+[b+c] = [a]+([a+b])$$

Here, we see the from the definition of addition in $[a]_n$ we can combine the elements and use the concept of associativity in $\mathbb{Z}$ to redistribute the terms to get the desired result.

- Does it contains an identity element? : We have that,

$$[a]+[0] = [a+0] = [a]$$

and

$$[0]+[a] = [0+a] = [a]$$

. This follows from the definition of $[a]_n$ and via combining the equivalences classes we are able to show that only one unique value can be taken as the identity that being [0].

- Does there exist an inverse? : We have that,

$$[a]+[-a] = [a+(-a)] = [0]$$

and

$$[-a]+[a] = [(-a)+a] = [0]$$

Here, we see that from the definition of $[a]_n$ that we are able to combine and redistribute the equivalence classes to get the identity of [0] hence it suffices that the inverse element for the group $[a]_n$ with the binary operation "+" is [-a].

Hence, we have that we have proven all the axioms of a group and the fact that the set is closed. Hence we can say that the set $[a]_n$ with the binary operation "+" is a group.

So we have that $([a]_n, +)$ is a group but what about $([a]_n, \cdot)$ - here we find that as this set can get a check mark for being associative , being closed and also having a identity element. However, given that [0] is in the set $[a]_n$ a few problems arise: one being that that there does not exist a multiplicative inverse (that is an integer) for [0] such that.

$$[k]*[0] = [1]$$

This is a big problem because as we are not able to form a group under multiplication means we cannot base the structure of the sets used in modern day cryptography as a group as it would not be able to carry out all operations in crypto to ensure the encryption function and decryption function are fulfilled. Hence, we find that we need to base the structure of our set on another concept. However, there is a caveat as it is possible to restrict the set in a way to make sure a group structure can be achieved. That being forming a group of units.

4.3.2 Rings [5]

Climbing up the abstract ladder we find the next family of structures. That being a ring - Hence, lets introduce the notion of this structure.

Definition Let R be a non-empty set with the operators + and $\cdot$ being binary operators. Hence, we get that for :

$$+ : R \times R \Rightarrow R, (a,b) \longmapsto a+b$$
$$\cdot : R \times R \Rightarrow R, (a,b) \longmapsto a \cdot b$$

Such that, R must also satisfy the following properties:

1. R is an abelian group under +.
2. It must be associative for the operation $\cdot$ - As in, $\forall \quad a,b,c \in R$

$$a\cdot(b\cdot c) = (a\cdot b)\cdot c$$

3. It must have these distributive properties - $\forall a,b,c \in R$

$$a\cdot(b+c) = a\cdot b + a\cdot c$$

and

$$(b+c)\cdot a = b\cdot a + c\cdot a$$

[5] This section comes from summarizing sources [5] , [14], and [15] notes. Such that the definitions and proofs come from this source Examples are given by the video lectures provided by source [16]

Remarks

- The additive identity is called the zero of the ring - **0**
- There may be times where the ring has an multiplicative identity (although not necessary). Such rings are said to have a multiplicative identity of **1** and these are called **Rings with identity**
- If a ring has a identity it may suffice that there does not exists multiplicative inverses for certain elements. For example, the element **0** will never have a multiplicative inverse.
- The operations carried out over a ring do not have to be commutative - Rings that are commutative are called **Commutative Rings**.

Examples Of Rings

1. The set $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{C}$. Are a commutative rings with identity. .
2. The set of all 2×2 Real matrices form a ring under matrix addition and multiplication. It is a non-commutative ring with identity **I**.
3. $[a]_n$ forms a commutative ring with identity under addition and multiplication $(mod \quad n)$ This is also a finite ring [given its a finite set] . (we will touch on this after on)
4. **HOWEVER,** $(\mathbb{R}^3, + \,, \cdot)$ is not a RING, given it has no identity element for multiplication and fails the criteria of being associative.

Basic properties of rings Here we find that the following properties hold for all rings:

- $$x0 = 0x = 0 \quad \forall x \in R$$
  - **Proof**:
    $$x0 = x(0+0) = x0 + x0 \Rightarrow x0 = 0$$
- $$(-x)y = x(-y) = -(xy)$$
  and
  $$(-x)(-y) = xy \quad \forall x, y \in R$$
  - **Proof**:
    $$xy + (-x)y = (x + (-x))y = 0y \Rightarrow (-x)y = -(xy)$$
    and
    $$(-x)(-y) = -x(-y) = -(-(xy)) = xy$$

How does the Modulo Operator Relate to Rings?

Here we discuss the concept of the integer ring $[a]_n$, which consists of the set:

$$[a]_n = \{[0], ..., [n-1]\}$$

Here we can apply the definition of the operators of a ring to fit the context of using an integer ring. Hence, using the two operators $+$ and $\cdot$ we get that $\forall \quad a, b, c \in \mathbb{Z}_n$ we get:

$$a + b \equiv c \quad (mod \quad n) \tag{16}$$

$$a \cdot b \equiv d \quad (mod \quad n) \tag{17}$$

These represent the calculation/ operation rules for a integer ring $[a]_n$ We will now try to manipulate this to get what we want which is being able to find a multiplicative inverse. This is important in cryptography as we need to be able to apply these operations to our decryption to get the piece of information, we tried to keep secure into its original form.

Example So, let's return to our modulo operator and pick a value m and a. Let $a = 2$ and m 9 Hence we have :

$$a = 2 \quad (mod \quad 9)$$

$$a^{-1} = ? \quad (mod \quad 9)$$

What inverse element can we multiply to a to get the identity. let $a^{-1} = 2^{-1}$. Hence,

$$2 \cdot 2^{-1} \equiv 1 \quad (mod \quad 9)$$

But this is wrong - given that $2^{-1} = 0.5$ and $0.5 \notin \mathbb{Z}$. So, what do we do here. We must look to choose another value of the multiplicative inverse that lies in the set $[a]_n = \{[0], [1], [2], [3], [4], [5], [6], [7], [8]\}$.

So let's now choose 5. From this we get

$$a \cdot a^{-1} = 2 \cdot 5 = 10 \equiv 1 \quad (mod \quad n)$$

As required, this is given as $gcd(2, 9) = 1$ so we are guaranteed to have an inverse as it does exist. Where we see a case were $gcd(a, 9) \neq 1$ it suffices that the inverse does not exist. From this we learn that our two values must be co-prime for an inverse to exist.

For some elementary ciphers we can use the structure of rings to cover the operations that occur as a multiplicative inverse is not always necessary so if you use a simple cipher - i.e Cesar shift. You can base all elements in the set as a part of a ring. The problem with rings is when we deal with more complex ciphers where a multiplicative inverse is necessary for all elements in the set. To be able to compute the elements. For modern day ciphers we need to use a structure that would be able to account for multiplicative inverses and all operations. Hence, we will now go on to discuss fields in the next section.

4.3.3 Fields[6]

We will now start discussing the notion of fields. Understanding fields is important because their advantages of basing operations in algorithms over fields in comparison to basing them over group or simplistic rings. Here, we get more complex as we start to see how defining a structure an algorithm as a field would allow us to be able to compute more complex calculations and allows us to be able to have multiplicative inverses – something that wasn't possible with rings due to the fact that the structure of a ring did not allow that in the most critical cases (due to the impossibility of finding a inverse).

Let's define what a field is,

Definition A field F is a commutative ring with identity, $(R, +, \cdot)$ with the following properties:

1. All elements of F form an additive group with the group operation "+" and the neutral element **0**.
2. All elements of F except **0** form a multiplicative group with the group operation "×" and the neutral element I.
3. When the two group operations are mixed, the distributivity law follows that

$$\forall a, b, c \in F \quad a \cdot (b + c) = a \cdot b + a \cdot c$$

Example Of Fields

1. The set, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{C}$. All form a field.

Remarks

- If F and E are fields with $F \subset E$. Then we can say that E is an extension of f.

Finite Fields As mention previously, in modern day cryptography, most algorithms work with finite sets as its more easier to manage and manipulate. Another reason, is that we can also work in finite fields - we will look more into it in this section:

Definition A *Finite Field* is a field with a finite number of elements. The number of elements contained in the set is called the *Order* of the field.

[6] This section comes from a summarising of sources [5], [6], [14], [15], [17] and [18] notes. Such that the definitions and proofs come from this source. Examples are given by the video lectures provided by source [16]

As mentioned previously we can only have a field with a certain number of elements. That being a prime number or a power of a prime number i.e if we had a field of order m. It must suffice that $m = p^n$ (where $n > 0$). p refers to the *characteristic* of the finite field. Because, for the operations of addition and multiplication to be carried out over integers modulo we must have m in its base form - in this case prime numbers. Finite fields can be represented in multiple ways. The way we denote a finite field is by writing $GF(p^n)$.

Examples of Finite Fields

- $GF(11)$ exists as a finite field with 11 elements.
- $GF(81) = GF(3^4)$ exists as a finite field with 81 elements.
- $GF(256) = GF(2^8)$ exists as a finite field with 256 elements. This is a notable field as most cryptographic algorithms use this field a basis for all operations in the cipher to occur.
- Not $GF(12)$ - This cannot be defined as a finite field since $12 = 2^3 \cdot 3 \neq p^m$.

Now we need to further define the different types of finite fields. This is important as we will continue to discuss the the operations that happen for each finite field. There are two main finite fields:

1. Prime Fields - where the finite fields have $m = 1$, Such that $GF(p)$
2. Extension Fields - Where the finite fields have $m > 1$. For $GF(p^n)$

We are particularly interested in the Extension Fields of $GF(2^m)$ - mainly, because this is the easiest to manipulate and that most modern day cryptographers structure their algorithms on the $GF(2^m)$ field.

4.3.4 Prime Field Arithmetic

Here, we will explore the arithmetic and operations that occur over a prime field $GF(p)$. Where $GF(p)$ is defined by the set of integers $\{0, 1, .., p-1\}$.

Addition, Subtraction and Multiplication Let $a, b, c, d, e \in GF(p)$ where $GF(p) = \{0, 1, ..., p-1\}$. Then we have :

$$a + b \equiv c \quad (mod \quad p)$$
$$a - b \equiv d \quad (mod \quad p)$$
$$ab \equiv e \quad (mod \quad p)$$

Not much changes from what we have done previously. However, we do tend to find that not conditions of a field are satisfied with these conditions. Hence, there may be times where we may need to redefine the field to get the required result.

Inversion Let $a \in GF(p)$. Then, the inverse a^{-1} must satisfy the equation:

$$a \cdot a^{-1} \equiv 1 \quad (mod \quad p) \tag{18}$$

The inverse can be calculated via the use of the extended Euclidean algorithm.

4.3.5 Extension Field Arithmetic

Now, on to the juicer side of Finite Fields, the arithmetic of extension fields. Most notably extension fields of $GF(2^m)$ for $m > 1$, The difference between extension fields of $GF(2^m)$ and prime fields, is that the difference in element representation. Where in prime fields elements can be represented as numbers. However, with extension fields of $GF(2^m)$ - the elements are represented as polynomials as the combined polynomials are represented as a part of a function composing of polynomials denoted as $A(x)$.The coefficients of the polynomial are form the basis for the function.

For any given value of m we get the function $A(x)$ for $GF(2^m)$ is:

$$a_{m-1}x^{m-1} + ... + a_1x + a_0 = A(x) \in GF(2^m)$$

Example We can look at an example of this - Let have $m = 3$. Hence $GF(2^3) = GF(8)$. Then we have,

$$a_2x^2a_1x + a_0 = A(x)) \in GF(8)$$

Our basis for this polynomial are a_2, a_1 and a_0.

However, there are 8 elements in this extended field. Hence, we must find all the possible combinations that take use of the basis to compute all elements. Through calculation we find that the elements of $GF(8)$:

$$GF(2^3) = \{0, 1, x, x + 1, z^2, x^2 + 1, x^2 + x, x^2 + x + 1\}$$

A matrix comprising of all combinations can be used to calculate the elements in the field.

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

Definition for Addition and Subtraction Let $A(x)$, $B(x) \in GF(2^m)$. The sum of the two elements are computed by applying the formula [7]:

$$C(x) = A(x) + B(x) = \sum_{n=0}^{m-1} c_i x^i . c_i \equiv a_i + b_i \quad mod \quad 2$$

and the difference can be calculated by applying the formula:

$$C(x) = A(x) - B(x) = \sum_{n=0}^{m-1} c_i x^i . c_i \equiv a_i - b_i \equiv a_i + b_i \quad mod \quad 2$$

Example for addition Let have a extended field of $GF(2^3)$. Then :

$$A(x) = x^2 + x + 1$$

and

$$B(x) = x^2 + 0 + 1$$

Then, for summation we have:

$$= (1 + 1)x^2 + x + (1{+}!) \quad mod \quad 2$$

$$= x \quad mod \quad 2$$

Definition for Multiplication Let $A(x)$, $B(x) \in GF(2^m)$ and let[8]:

$$P(x) = \sum_{n=0}^{m-1} p_i x^i . p_i \in GF(2)$$

be an irreducible polynomial. Then the multiplication of the two elements A(x), B(x) is computed as:

$$C(x) \equiv A(x) \cdot B(x) \quad mod \quad P(x)$$

4 Hence from this, we get that every field requires a irreducible polynomial $P(x)$ of degree m. Although not polynomials are irreducible hence there will be times where multiplication cannot be carried out.

[7] quotation from source [5] and [15]

[8] quotation from source [5] and [15]

Example Again, Let have a extended field of $GF(2^3)$. Then :

$$A(x) = x^2 + x + 1$$

and

$$B(x) = x^2 + 1$$

Then,

$$A \cdot B = (x^2 + x + 1) \cdot (x^2 + 1)$$

and the calculation results in,

$$= x^4 + x^3 + x + 1$$

The problem with this result is that it is not an element in the field of $GF(2^3)$. Hence, we must reduce the polynomial by an irreducible polynomial to achieve the goal of finding an element in the field. So, let:

$$C'(x) = x^4 + x^3 + x + 1$$

we chose our irreducible polynomial $P(x)$ to be:

$$x^3 + x + 1$$

and we perform algebraic division to achieve the end result $C(x)$:

$$x^2 + x$$

A element that belongs to the field. So,

$$x^2 + x \equiv A \cdot B \quad mod \quad P(x)$$

Inverse Again, the inverse must satisfy :

$$A(x) \cdot A^{-1}(x) \equiv 1 \quad mod \quad P(x)$$

The element $A^{-1}(x)$ can be calculated via the extended Euclidean algorithm.

We now covered all the mathematical concepts required for modern day cryptography. We should now go on to on to an application of modern-day cryptography. The AES Method.

5 The AES Method [9]

We will now look to briefly discuss and application of what we've learned so far. Before, we looked at the introductory components in cryptology and the mathematics behind modern-day cryptoanalysis. As we defined the structure of types of ciphers in a cryptosystem and how finite sets and modulo operators can be applied to ciphers. We now look to discuss the AES algorithm (Also known as Rijndael) We will discuss the historical aspects of how the algorithm came to be the most important cryptographic algorithm on Earth. Then, we will discuss how the encryption and decryption function works. Finally, analysing the disadvantages and advantages of the algorithm.

5.1 History of Rijndael [10]

- In the late 90s, there was a growing need for a new cipher to succeed the DES. This is because it was believed that the DES has been inadequate in fulfilling its main objectives. As, its vulnerabilities were becoming well to known to predators. The size of the bit key was too small in comparison to what was needed to be carried at that time (56 Bit-Key) hence this led to computations in the algorithm taking a long time to complete. Thereby, making the algorithm too slow and inefficient.
- Hence, on January 2 1997 - NIST decides to announce the need for a successor for the DES - In hope that they could get the international crypto community to invest their input into form a new cipher which would become the new cipher. They state that the new cipher would need to satisfy the requirements of:
  1. Must be publicly defined and publicly available.
  2. Must have a variable length key most preferably in the region of 128, 192 and 256 length bit keys.
  3. Can be implemented in both hardware and software.
- The Judgement criteria would go as follows:
  1. Level of security
  2. The computational efficiency
  3. Memory requirements
  4. Hardware and Software suitability
  5. The level of simplicity and flexibility
  6. The licensing Requirements
- Surprisingly, only 15 submitted when it was expected that at least 50 would be submitted. The Judge panel will narrow this number down to 5 finalists. Those were:
  - MARS - IBM
  - RC6 - RSA Laboritories
  - Rijndael - by Daemon and Rijment
  - Serpent -Anderson, Bham, Knudsen
  - Twofish - Schneier, Kelsey, Whitting and others
- The judge panel would go on to decide that the Rijndael cipher would become the Advance Encryption Standard (AES) in 2001. As it had met all security, cost and implementation criteria

5.2 Features Of The AES [11]

- AES is a block cipher with a block length of 128 bits
- AES allows for three different key lengths: 128, 192, or 256 bits
- The Encryption function consists of 10 rounds of processing for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys.
- All rounds except the last are identical.

[9] This section was comprised and summarized from a selection of sources - [1], [3], [4] , [5] , [6] , [7], [8], [9], [10], [11], [12], [15], [16] and [20] . The main method is provided as as summary of the lecture 7 and 8 from source [16]

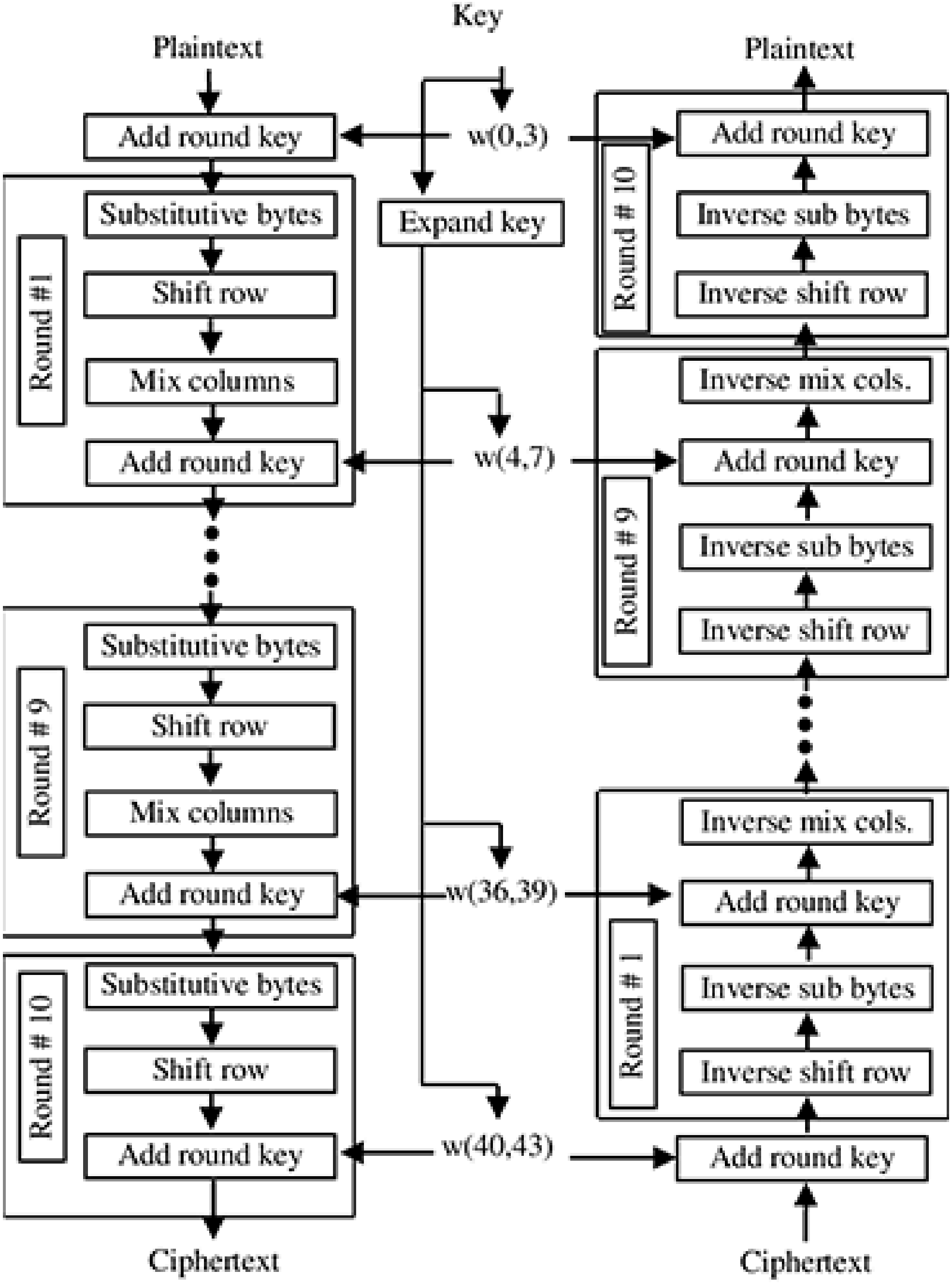
[10] This section is a summary of [6] lecture notes from the powerpoint

[11] This is quoted from source [10] section 8.1

- The encryption function substantially different in comparison to the decryption function. Whereas the DES both encryption and decryption functions are similar.
- Unlike DES, AES is an example of key-alternating block ciphers.
- Like DES, AES is an iterated block cipher in which plaintext is subject to multiple rounds of processing, with each round applying the same overall transformation function to the incoming block.
- Lastly, to compare the DES and AES -DES is a bit-oriented cipher, AES is a byte-oriented cipher.
- All multiplication that occur in the AES algorithm happens over a Extension field of $GF(2^8)$. Whereas the addition is done via the use of a XOR operator.

5.3 Outline Of The Algorithm [12]

Figure 2: Diagram of How the AES Algorithm works- From Source [21]



[12] Mainly Summarised from sources [5], [10], and [15]

In a nutshell, the AES repeats 4 major steps to encrypt data. It takes a 128 but block of data and a key to produce a *cipher-text* as its output. In order, the four main steps are :

1. Sub Bytes
2. Shift Rows
3. Mix Columns
4. Add Key

We must note that the number of rounds performed by the algorithm depends on the size of the key. Hence, we have that:

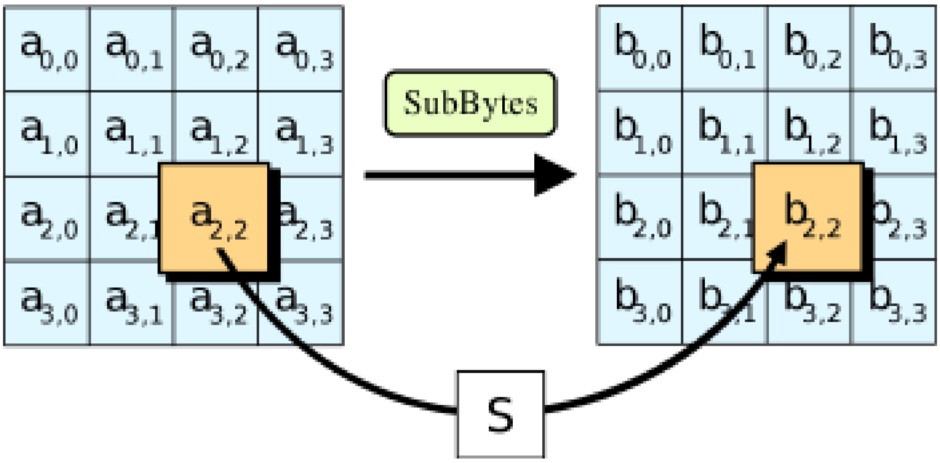
Key Size	Rounds
128	10
192	12
256	14

The larger the number of keys we have the more secure the data would be. However, this is more expensive and time consuming as the amount of time to encrypt increases.

5.3.1 Sub Bytes

In this step we see that every element of the matrix is replaced by an element in the s-box matrix. This is a non-linear substitution which know is invertible as from the previous section that our extension field $GF(2^8)$ is used as a generating function for the algorithm and the extension fields are used to help construct the s-box matrix. For each input byte of, {AB}:

Figure 3: SubByte step being carried out (Source [19])


$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
SubBytes
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
S

1. Let {AB} = the multiplicative inverse of {XY} in $GF(2^8)$
2. Let {XY}' = An affine transform of {AB}.

An example of a transformation is:

$$\{A8\} \Rightarrow \{C2\}, \{21\} \Rightarrow \{FD\}, \{27\} \Rightarrow \{CC\}, \{CF\} \Rightarrow \{8A\}$$

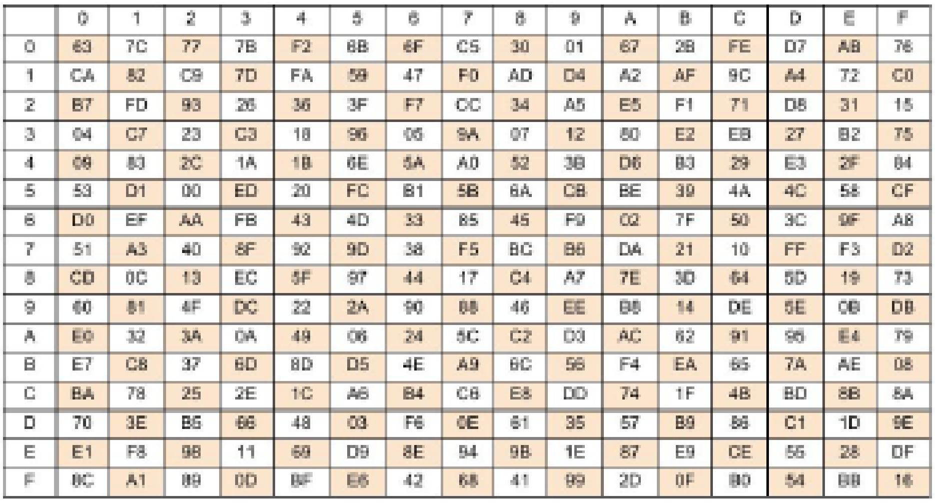
5.3.2 Shift Rows

In this step, rows the block are cylindrically shifted in the left direction. We have that the first row does not experience any changes. However, the second one is shifted by one, the third by 2 and the fourth by 3.

Figure 4: SubByte step : Affine Transform being carried out - From Source [6]

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Figure 5: SubByte step : S-Box The substitution values for the byte XY - In Hexadecimal Format -From Source [22]



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

5.3.3 Mix Columns

This is the most important step of the lot. Every column gets treated as a four-term polynomial. Then a polynomial is applied to each column, in hopes to get a new polynomial. This is important because it causes the flip of bits to spread across all over the block and then the block would be multiplied with a fixed matrix. This again is important as the multiplication occurs over an extension field.

The polynomial returned $C(x) =$

$$\{03\}x^3 + \{01\}x^2 + \{031\}x + \{02\}$$

Figure 6: Shift Rows step : Cyclical Shift of the rows of the state (Source [19])

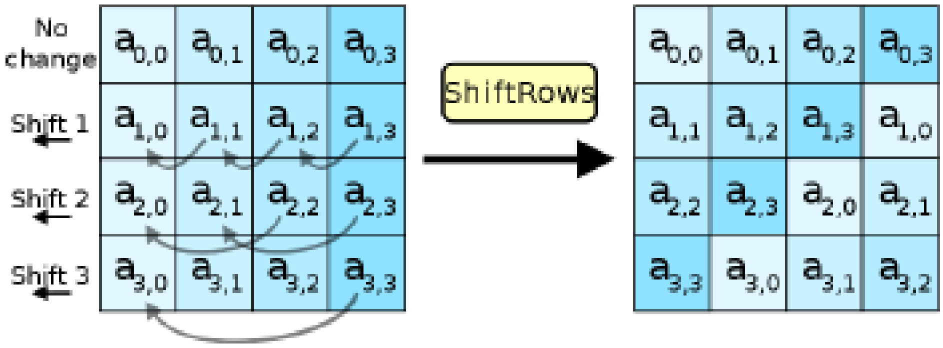
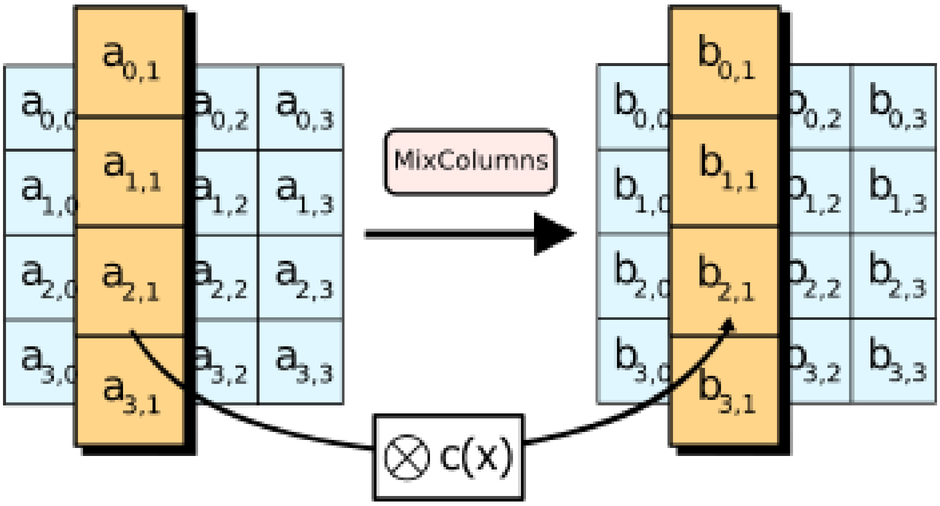

No change
Shift 1
Shift 2
Shift 3
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
ShiftRows
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,1}$ $a_{1,2}$ $a_{1,3}$ $a_{1,0}$
$a_{2,2}$ $a_{2,3}$ $a_{2,0}$ $a_{2,1}$
$a_{3,3}$ $a_{3,0}$ $a_{3,1}$ $a_{3,2}$

Figure 7: Mix Columns Step: How the new polynomial is calculated - From Source [6]

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 02\ 03\ 01\ 01 \\ 01\ 02\ 03\ 01 \\ 01\ 01\ 02\ 03 \\ 03\ 01\ 01\ 02 \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

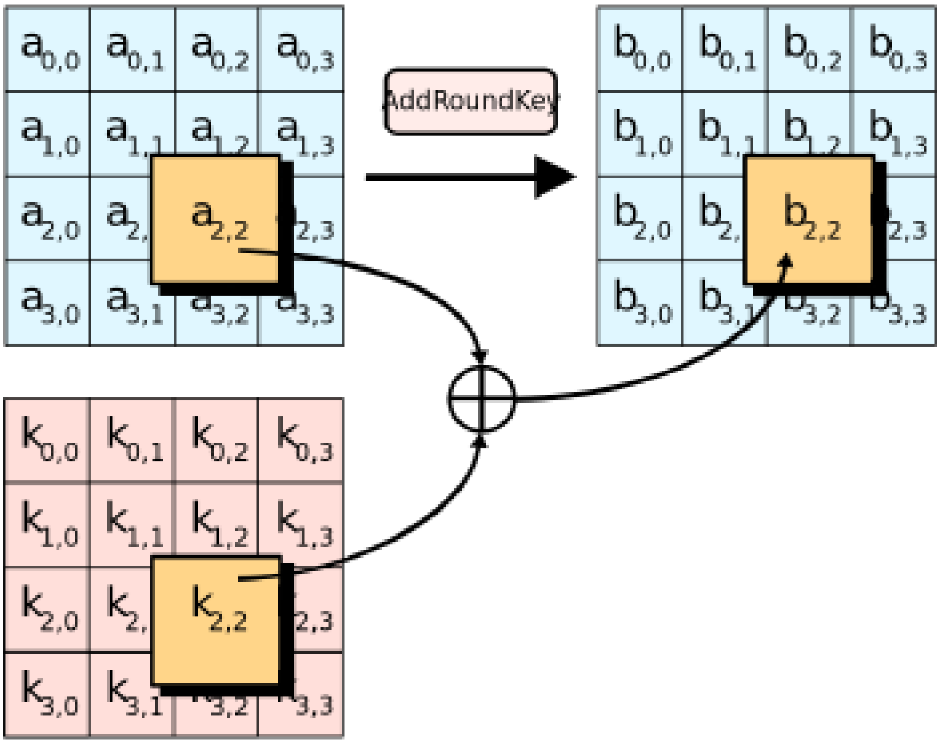
Figure 8: Mix Columns Step: The Flipping of Bits (Source [19])


$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
MixColumns
$\otimes c(x)$
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$

5.3.4 Adding A Round Key

In this step, A transformation takes place where the state is modified by combining it with a round key with the bit wise XOR operation. Once this step is completed the keys are no longer available for this step and get disregarded as using the same key would weaken the system. Which would cause trouble. Hence, to overcome this issue problem keys are expanded.

Figure 9: Adding A Round Key (Source [19])


$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
AddRoundKey
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
$k_{0,0}$ $k_{0,1}$ $k_{0,2}$ $k_{0,3}$
$k_{1,0}$ $k_{1,1}$ $k_{1,2}$ $k_{1,3}$
$k_{2,0}$ $k_{2,2}$ $k_{2,3}$
$k_{3,0}$ $k_{3,1}$ $k_{3,2}$ $k_{3,3}$

5.3.5 Inverse

For us to be able to decrypt the function. We will have to invert everything in reverse order. Hence:

1. Add Key
2. inverse Shift Rows
3. inverse Sub Bytes
4. Add Key
5. inverse Mix Columns

Each sub-function of the cipher is invertible. Hence, here's how to compute the inverse:

1. Sub Bytes: The inverse is the reverse of the Affine Cipher, then find the inverse of the result.
2. Shift Rows: Shift the row by reverse amounts
3. Mix Columns: Apply the inverse matrix to each column
4. Add Round Key: Use the add round key again with same key.

There, we have the full AES algorithm.

5.4 The Security Of AES [13]

There have been some key-related attacks on the AES algorithms - mainly on the 192-bit, 256-bit versions. However, these attacks are 'lesser attacks' and these are better than the exhaustive search. As, it would be totally ineffective and infeasible for real world applications. This is mainly due to the fact that there is that the number of possible combinations increases exponentially, as the key size increases as well.

[13] Summarised from sources [15] and [20] - Both tables are quoted from source [20]

Key Size (bytes)	Possible Combinations
1	2
2	4
4	16
8	256
16	65536
32	4.2×10^9
56 (DES)	7.2×10^{16}
64	1.8×10^{19}
128 (AES)	3.4×10^{38}
192 (AES)	6.2×10^{57}
256 (AES)	1.1×10^{77}

So, what does this mean? it means as there more computations to complete in order to complete the task - it suffices that it takes more time to break the algorithm. In fact, a long time to crack- as we can see:

Key Size (bytes)	Time to Crack
56 (DES)	399 seconds
128 (AES)	1.02×10^{18} Years
192 (AES)	1.872×10^{37} Years
256 (AES)	3.31×10^{56} Years

As you can see even if we managed to build a supercomputer that can calculate such combinations. It would take about 1 billion billion years to be able to find out one of the keys of the weakest AES algorithm. Clearly, this is inefficient because by then the universe would be dead and that would make doing this infeasible. So, due to the absolute strength of the AES - we can determine that this is the most secure cipher on Earth. It would have to take a god tier level of mathematics to be able to effectively crack the code to break this cipher down.

6 Reflection and Conclusion

At the start of the project – I stated that I wanted to discuss how the core principles of cryptology can be used and applied to modern day cryptography. By, stating that we defined the basic notions, investigated an example of a historical cipher, discussed the benefits and reasons for and against modern day cryptography. This led us to the main chunk of the report where we discussed the mathematical concepts that are involved in modern day cryptography. Where we talked about Modular arithmetic and Abstract structures and how they relate to each other. Then we looked at an application of the principles and concepts we have established and talked about the AES algorithm. We talked about the history, how the algorithm works and the security of the algorithm – by linking what we learned from the prerequisites to the application.

Doing this project, was difficult due to the uncontrollable circumstances. However, I was able to complete what I wanted to get done for second semester. Although, I had to change my goals to make them more achievable. I regret the fact that I didn't get a chance to talk more about modern-day cryptography and other algorithms and compare them – such as doing the IDEA, DES, and other algorithms. The focus for semester two was to have the mathematical aspect nailed down, as that had marked me down last semester – in which I do believe I have achieved. But I do think this had led me to have less time to focus on the application aspect of things. This may have led the AES section of the report to be weaker than intended. But, coupled with a stronger mathematical section I believe everything should average out at the end. In future, I should be more focused on balancing out the report and my time to ensure that the quality of the report improves.

Nonetheless, I enjoyed completing this project. Completing this project taught me more about the abstract world of mathematics and how people can come together as a part of a community and innovate. The task taught me about how to get through adversity as I managed to find a way to complete the task and get through tough times.

Finally, I thank you for the opportunity to be your tutee and taking care of me this year.

Thank you, God Bless and Good Luck for the future. ;)

Delice

7 Bibliography

1 Educative: Interactive Courses for Software Developers. (n.d.). What is the AES algorithm? [online] Available at: https://www.educative.io/edpresso/what-is-the-aes-algorithm. [Accessed 4 Mar. 2021].

2 apprize.best. (n.d.). Substitution-Permutation Networks - Symmetric Ciphers and Hashes - Modern Cryptography: Applied Mathematics for Encryption and Informanion Security (2016). [online] Available at: https://apprize.best/security/cryptography/8.html [Accessed 4 Mar. 2021].

3 McCallion, J. (2013). What is AES encryption? [online] IT PRO. Available at: https://www.itpro.co.uk/security/29671 is-aes-encryption. [Accessed 15 Mar. 2021].

4 Paquet, C. (2013). Implementing Cisco IOS network security : (IINS 640-554) foundation learning guide. Indianapolis, Indiana, Usa: Cisco Press. [Accessed 19 Mar. 2021].

5 Preneel, B. (2014). Understanding cryptography. Springer. [Accessed 19 Mar. 2021].

6 Heron, S. (2009). Advanced Encryption Standard (AES). Network Security, 2009(12), pp.8–12. [Accessed 19 Mar. 2021].

7 Kessler, G., 2021. An Overview of Cryptography. [online] Garykessler.net. Available at: ¡https://www.garykessler.net/l [Accessed 25 Mar. 2021].

8 Lake, J. (2018). What is AES encryption (with examples) and how does it work? [online] Comparitech. Available at: https://www.comparitech.com/blog/information-security/what-is-aes-encryption/.[Accessed 25 Mar. 2021].

9 Chirag Bhalodia. (n.d.). AES Example — AES Encryption Example. [online] Available at: https://www.youtube.com/ [Accessed 5 Apr. 2021].

10 Kak, A. (2016). Lecture 8: AES: The Advanced Encryption Standard Lecture Notes on ” Computer and Network Security ”. [online] Available at: https://engineering.purdue.edu/kak/compsec/NewLectures/Lecture8.pdf. [Accessed 8 Apr. 2021].

11 Federal Information Processing Standards Publication 197 Announcing the ADVANCED ENCRYPTION STANDARD (AES). (2001). [online] . Available at: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. [Accessed 15 Apr. 2021].

12 Sciencedirect.com. (2014). advanced encryption standard - an overview — ScienceDirect Topics. [online] Available at: https://www.sciencedirect.com/topics/computer-science/advanced-encryption-standard. [Accessed 19 Apr. 2021].

13 Stephen Theriault, Graham Niblo and Sejong Park (2019) MATH2003: Group Theory v3 [Accessed 1 May.2021]

14 www-groups.mcs.st-andrews.ac.uk. (n.d.). Definition and examples. [online] Available at: http://www-groups.mcs.st-andrews.ac.uk/ john/MT4517/Lectures/L3.html [Accessed 18 May 2021]

15 Daemen, J., Rijmen, V. and Springer-Verlag Gmbh (2020). The Design of Rijndael AES - The Advanced Encryption Standard. Berlin [Heidelberg] Springer [Accessed 7 May 2021]

16 www.youtube.com. (n.d.). Introduction to Cryptography by Christof Paar - YouTube. [online] Available at: https://www.youtube.com/channel/UC1usFRN4LCMcfIV7UjHNuQg [Accessed 19 May 2021]

17 www.cs.cornell.edu. (n.d.). Modular arithmetic (CS 2800, Spring 2016). [online] Available at: https://www.cs.cornell.ed modular.html [Accessed 19 May 2021]

18 $www.youtube.com.(n.d.).-YouTube.[online]Availableat:https://www.youtube.com/watch?v=Buv4Y74_z7It=629s[Accessed19May2021]$

19 $[PicturesandDIagrams]WikipediaContributors(2019).AdvancedEncryptionStandard.[online]Wikipedia.Availablea$ $https://en.wikipedia.org/wiki/Advanced_Encryption_Standard.[acessed20May2021]$

20 User, S. (n.d.). How Safe is AES Encryption? [online] www.kryptall.com. Available at: https://www.kryptall.com/index 09-24-06-28-54/how-safe-is-safe-is-aes-encryption-safe. [Accessed 20 May 2021]

21 Mathur, N. and Bansode, R. (2016). AES Based Text Encryption Using 12 Rounds with Dynamic Key Selection. Procedia Computer Science, 79, pp.1036–1043 [Accessed 20 May 2021]

22 Anu Vazhayil. (2015). AES – Advanced Encryption Standard. [online] Available at: https://captanu.wordpress.com/20
[Accessed 20 May 2021]