Smartphone Cryptography

A comparison of techniques for encrypted data communication

By Tom Nicholls

Overview

- Presentation ~15min

  - Introduction
  - Next steps & research
  - Design & Implementation
  - Analysis
  - Results conclusion
  - Further work & improvements
  - Conclusion of the project

Overview

- Demonstration ~5min
    - Key Exchange Protocol
    - Smartphone test results

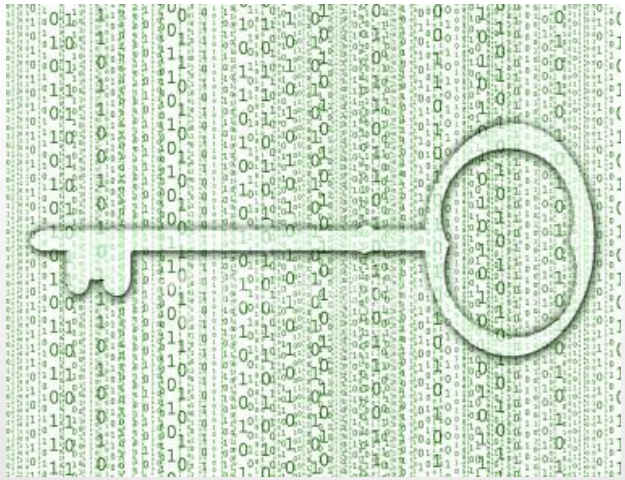
Overview

- Questions ~5min

Introduction

Problem -

- Increased popularity of smartphones
- Importance of keeping personal data safe
- SMS style of encrypted data or message exchange



Introduction

Project Basics -

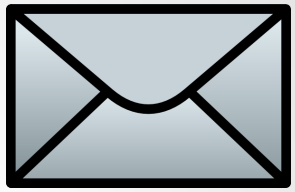
- Do applications that perform this function exist?
- How do they work?
- What is the performance of these style of applications?
- How could they be improved upon?

Research & Next Steps

- Currently available applications
- Cryptographic techniques
- Relevant Factors to compare and analyse
- System design

Currently available applications

- Standard SMS messaging
  - Not completely secure



- Cloak SMS Free
  - Developed by Hamish Medlin
  - AES symmetric key encryption

- RSA Cipher Cat
  - Developed by Miasoft
  - RSA symmetric key encryption

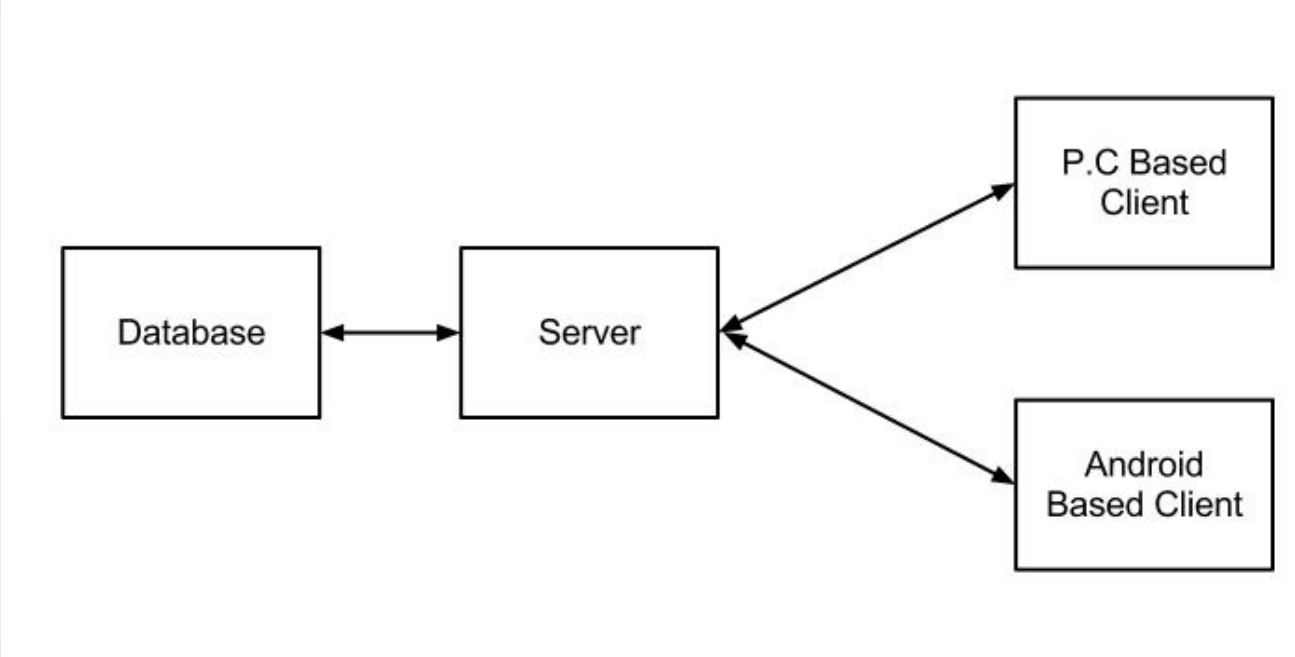
Cryptographic techniques

- Symmetric Key
  - AES
- Asymmetric Key
  - RSA
  - ECC

Comparison Factors

- Difficulty of techniques required to break encryption
- Key size
- Data usage
- Time required for key generation
- Time required for Encryption and Decryption

Basic System design



Design & Implementation

- Main System
- Cryptographic Schemes
- Data Communication

Main System

- Server
  - Allows connection from P.C and Android based clients via sockets
  - Accesses the database
  - Acts as a trusted third party with its own public/private keys
  - Facilitates the sending and receiving of messages between registered clients

Main System

- Database
  - My SQL database management system
  - Stores the users:
    - Unique ID (primary key)
    - I.P address
    - Public Key Location
    - Message locations
    - File locations for key exchange protocol

Main System

- Clients
  - Connects to the server via sockets
  - Generate and store its own keys
  - Specify which other client the user wishes to send data to
  - Complete the key exchange protocol required
  - Send user inputted data messages to the server

Cryptographic Techniques

- AES - Advanced Encryption Scheme
  - Symmetric- key algorithm
  - Input: Block of plaintext and a key
  - Applies several rounds of transformations
  - Output: Ciphertext block
  - Decryption is done by reversing the process

AES

Plaintext block of size 128 bits (16 bytes) in a 4x4 byte array

Key of size 128, 192 or 256 bits (16, 24 or 32 bytes)

Key size determines the number of transformation rounds

16 byte key - 10 rounds
24 byte key - 12 rounds
32 byte key - 14 rounds

AES

1. Key Expansion
   Expands the key to provide a 4x4 byte Round Key for each round

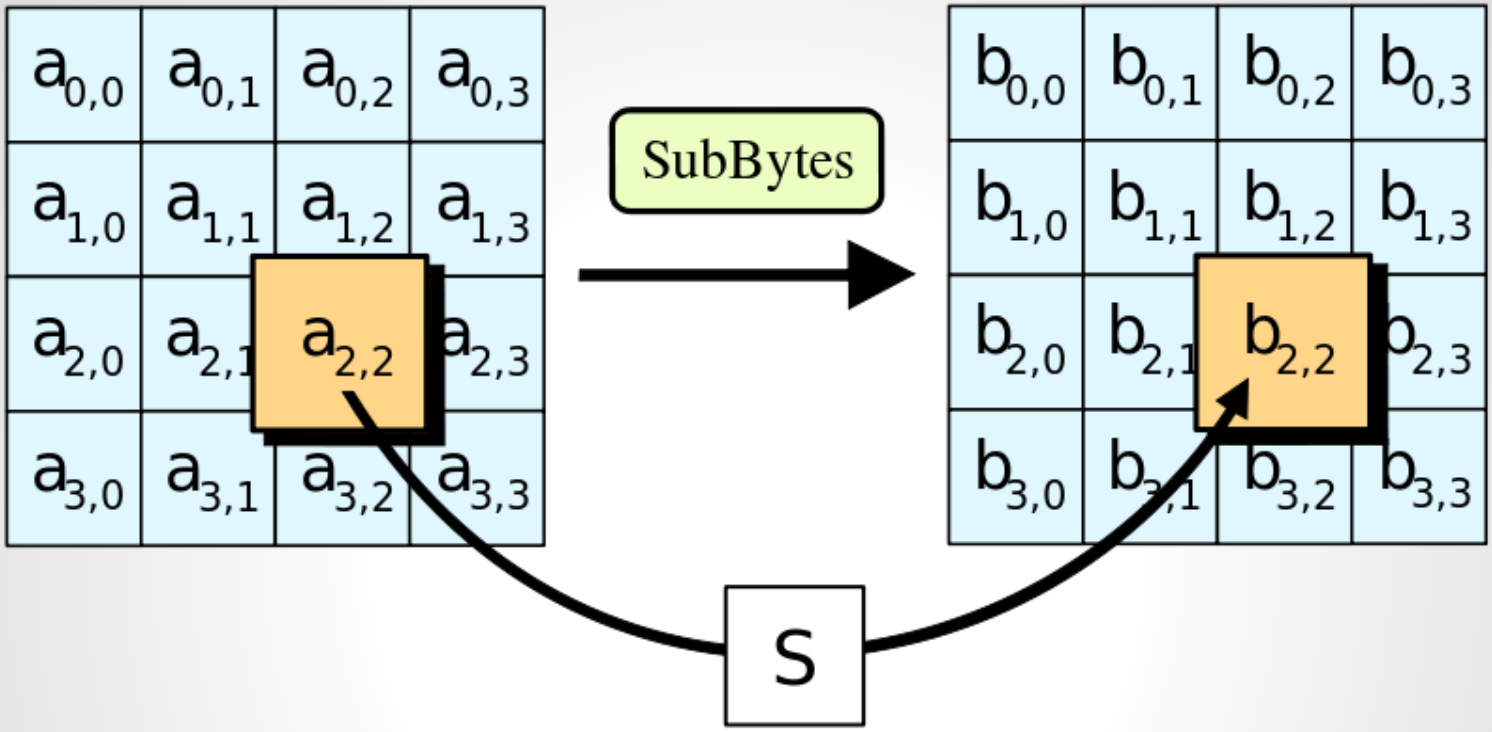
   16 byte key - 4 words
   Expanded to 44 words (176)
   Provides 11, 4 word round keys

2. Initial Round
   AddRoundKey

3. Round 1 to N-1
   SubBytes
   ShiftRows
   MixColumns
   AddRoundKey

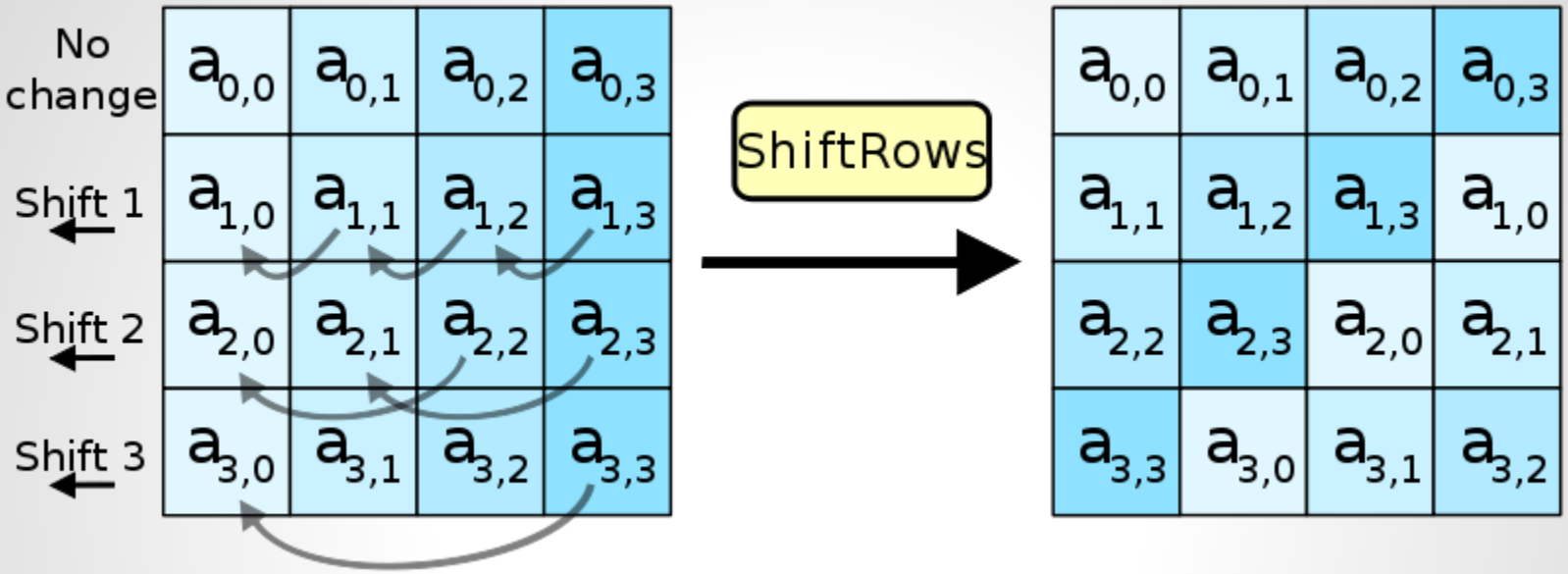
4. Final Round
   SubBytes
   ShiftRows
   AddRoundKey

AES - SubBytes

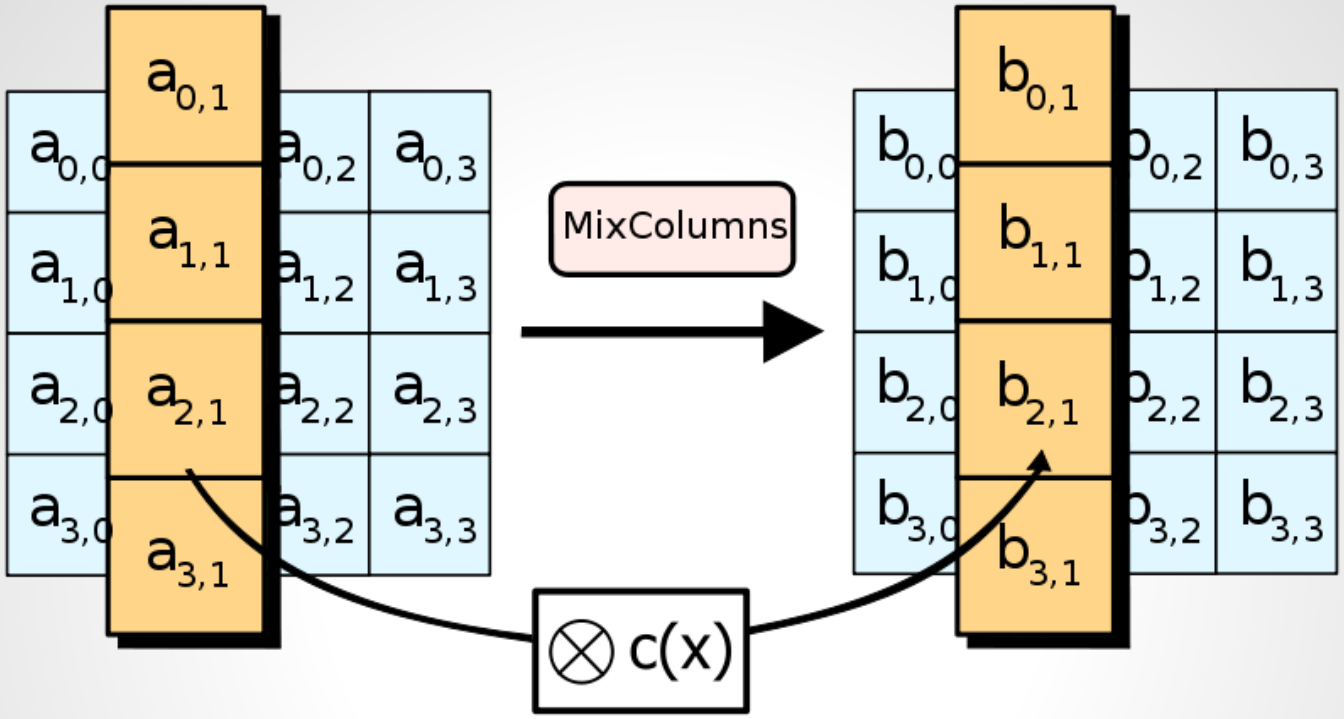


63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

AES - ShiftRows

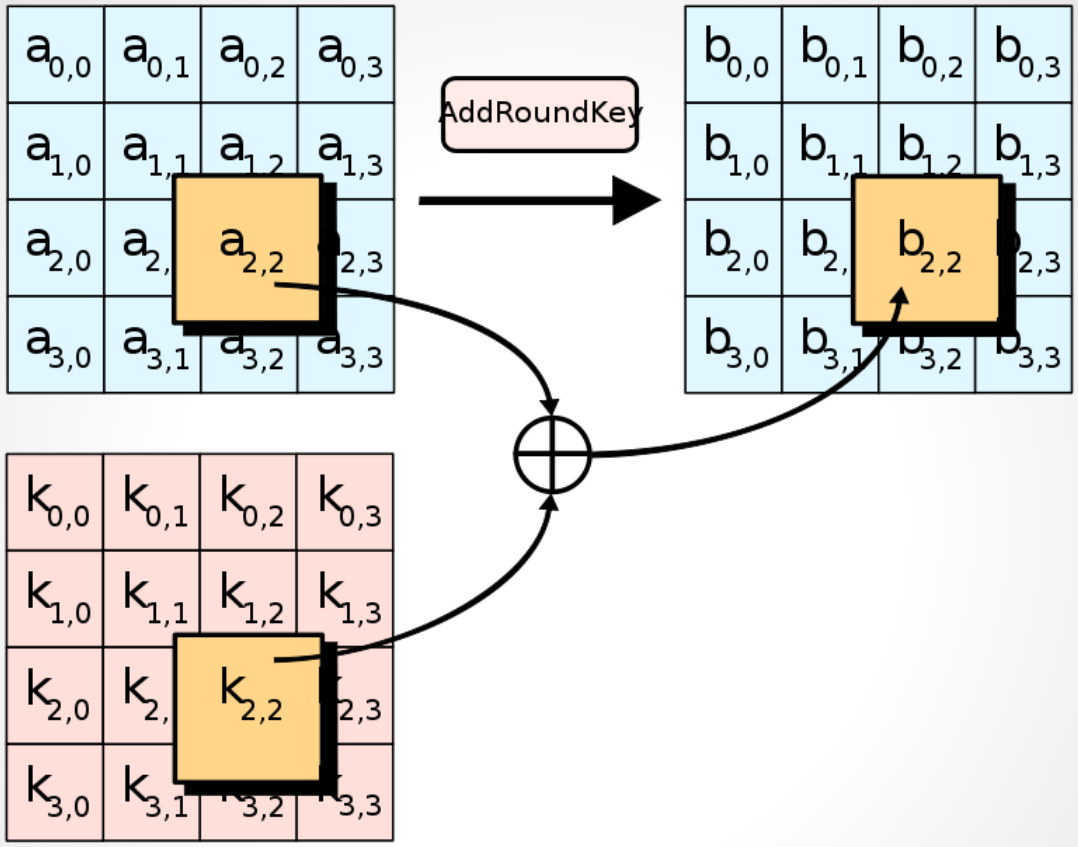


AES - MixColumns



$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}.$$

AES - AddRoundKey



Only stage which makes use of the key
Other stages provide confusion, diffusion and nonlinearity

Cryptographic Techniques

- RSA - Rivest-Shamir-Aldeman

  Based upon the difficulty of factoring large integers

RSA - Key Generation

1. Choose two distinct prime numbers p and q

2. Calculate $n = pq$

3. Calculate $\Phi(n) = (p - 1)(q - 1)$, where $\Phi(n)$ is the number of coprime integers k such that $1 \leq k \leq n$.

4. Choose e such that $1 \leq k \leq n$ and $\Phi(n)$ are coprime

5. Determine $d = e^{-1} \bmod \Phi(n)$, where d is the multiplicative inverse of e mod $\Phi(n)$

Private Key - (d,n)

Public Key - (e,n)

RSA - Encryption/Decryption

Encryption: $C = M^e \bmod (n)$

Decryption: $M = C^d \bmod (n)$

$M = C^d \bmod (n) = (M^e)^d \bmod (n) = M^{ed} \bmod (n) = M$

It is infeasible to determine d given e and n

Cryptographic Techniques

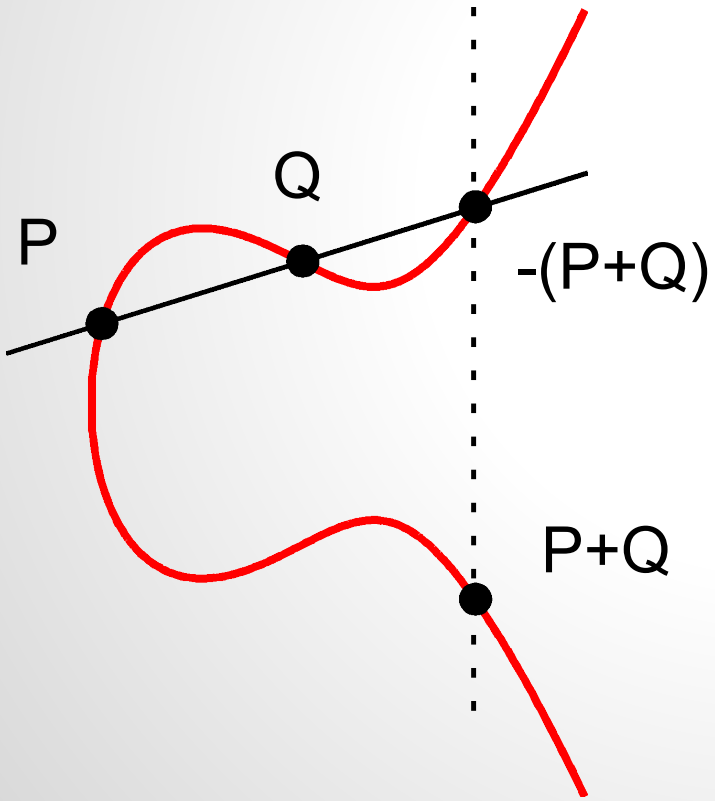
- ECC - Elliptic Curve Cryptography

  Based upon the difficulty of finding the discrete logarithm of a random elliptic curve element, with respect to a publicly known base point

ECC - Elliptic Curves over Real Numbers

Plane curve with points satisfying

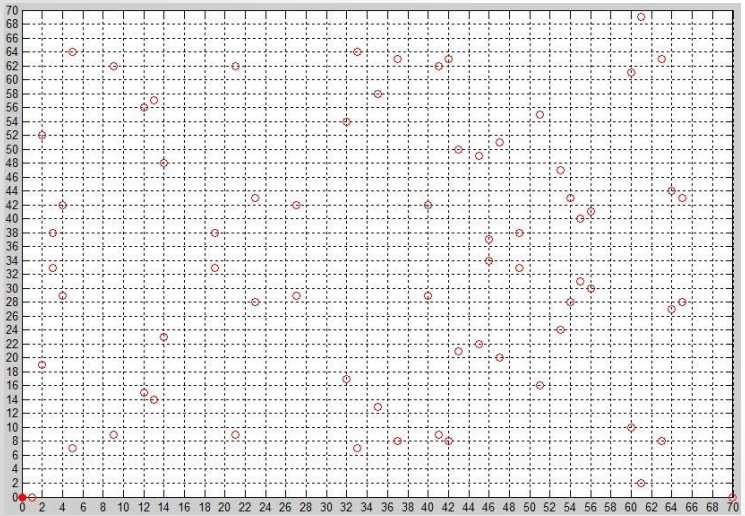
$$y^2 = x^3 + ax + b \qquad E(a,b)$$



ECC - Elliptic Curves over Z_p

Elliptic curves in which the variables and coefficients are restricted to elements of Z_p

$$y^2 \bmod p = (x^3 + ax + b) \bmod p \qquad E_p(a,b)$$



ECC - Cryptography

$Q = kP$

$E_p(a,b)$

$k < p$

$k = ?$

ECC - Cryptography

1. Global Public Elements -
   - $E_q(a,b)$, q prime
   - G , point with order n (large)

2. User A Key Generation -
   - Private: $n_A < n$
   - Public: $P_A = n_A \times G$

3. User B Key Generation -
   - Private: $n_B < n$
   - Public: $P_B = n_B \times G$

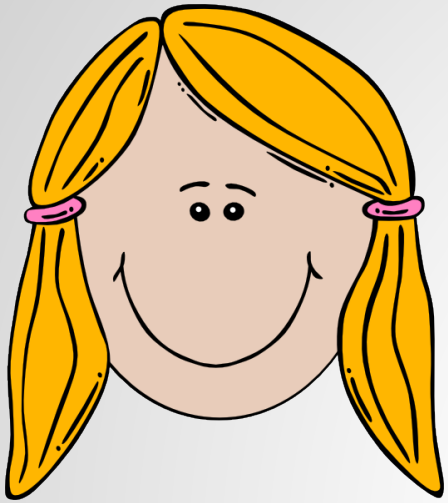
ECC - Cryptography

4. Encryption -
   - Message m encoded as a point P_m
   - A chooses a random positive k
   - $C_m = \{ kG , P_m + kP_B \}$

5. Decryption -
   - $P_m + kP_B - n_B(kG) = P_m + k(n_BG) - n_B(kG) = P_m$

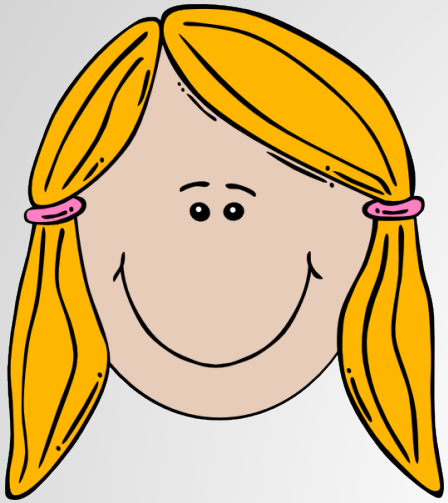
Data Communication

- Exchanging keys between client users
- Sending and receiving encrypted messages

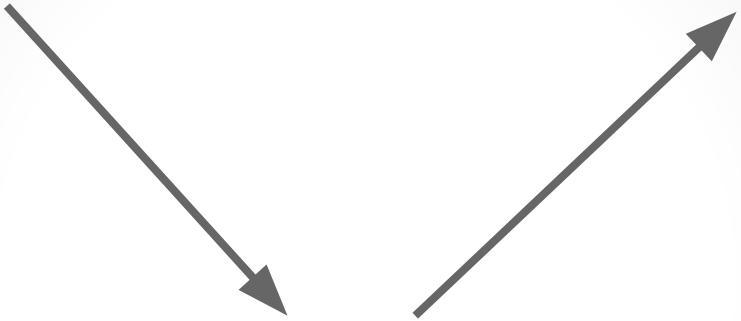


Alice

Bob

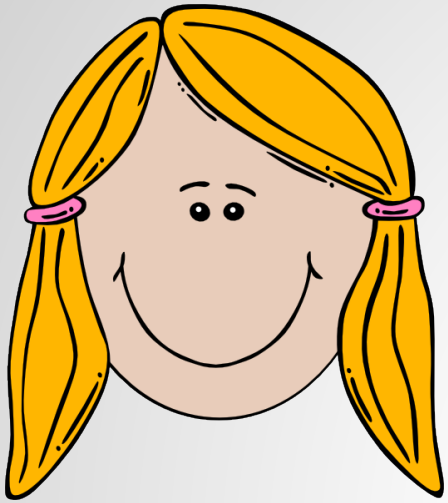


Alice

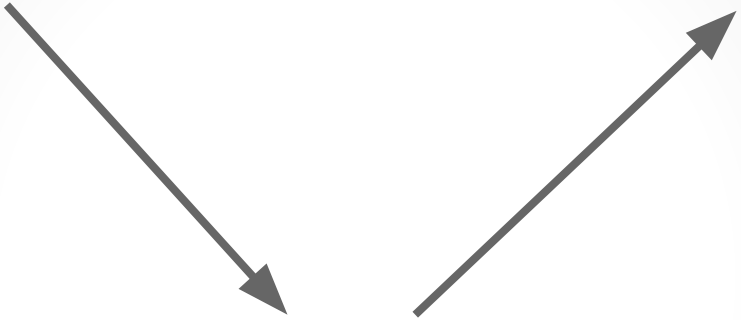


Trent

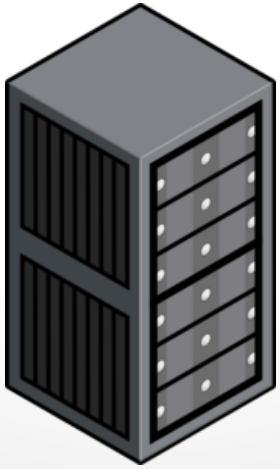
Bob



Client 1



Client 2



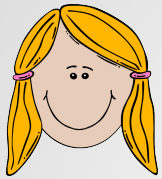
Trusted Third party server

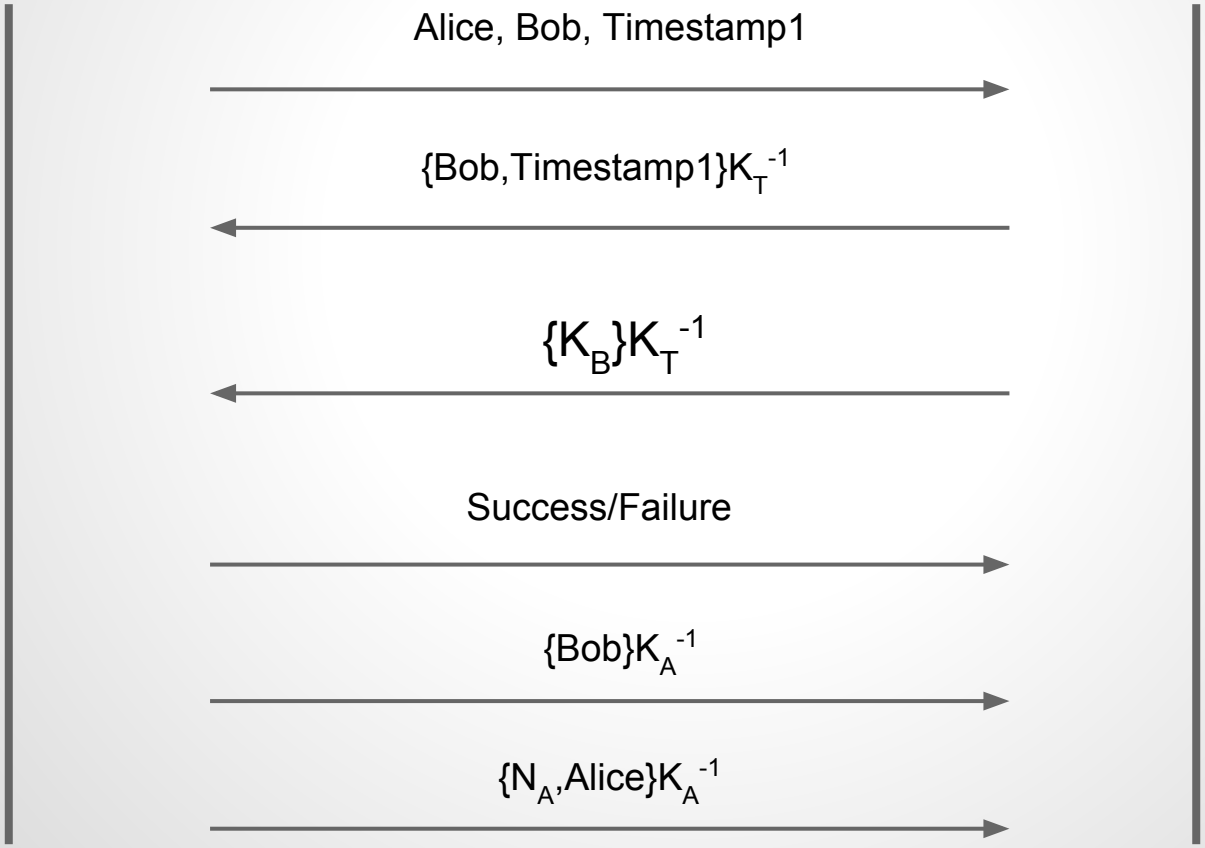
Database

New User

- New user to the system
- Generates their own public and private keys
- Registers their public key with Trent
- Receives and stores Trents own public key
- Allows encrypted communication

Key Exchange





Key Exchange

$\{N_A, Alice\}K_A^{-1}$

Agree to register

Bob, Alice, Timestamp2

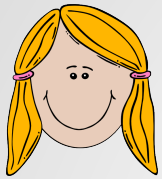
$\{Alice, Timestamp2\}K_T^{-1}$

$\{K_A\}K_T^{-1}$

$\{Alice\}K_B^{-1}$

$\{N_A, N_B, Bob\}K_B^{-1}$

Key Exchange

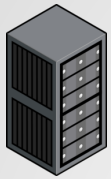


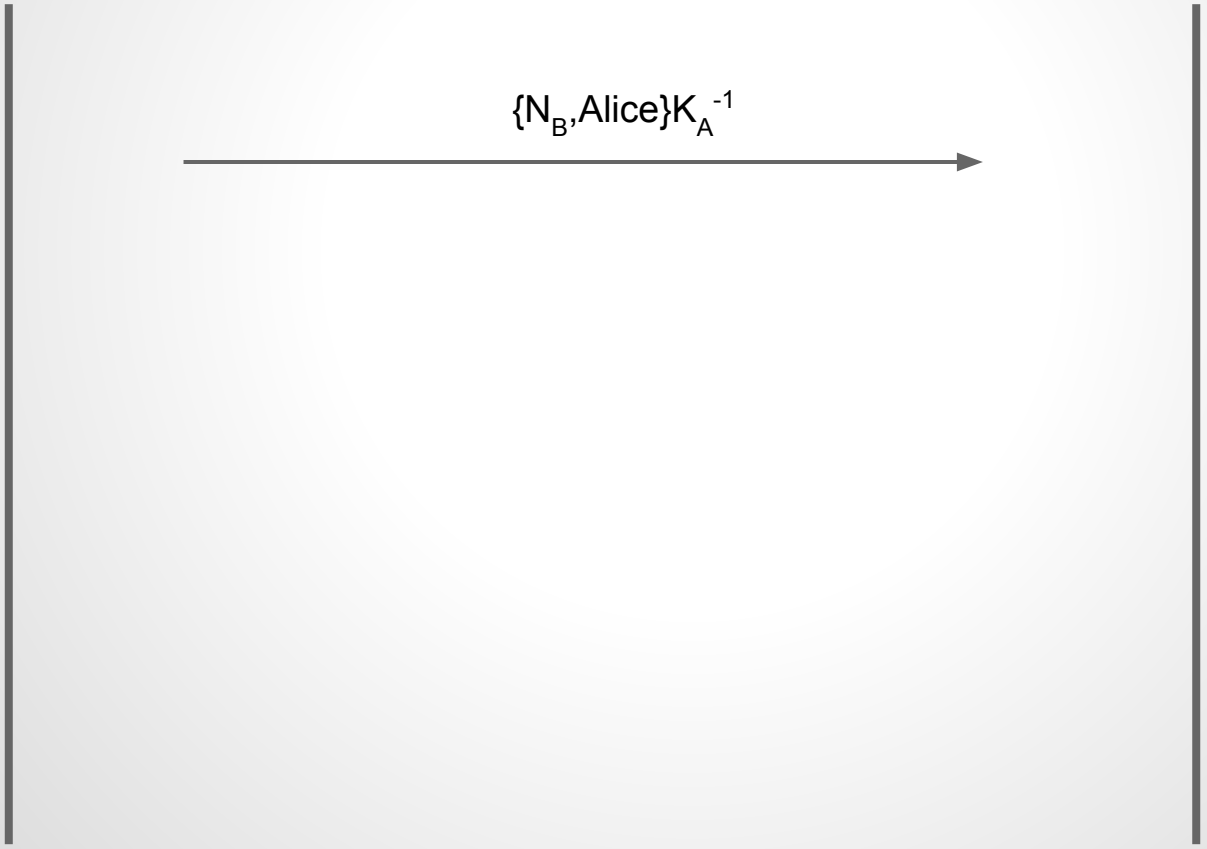
$\{N_A, N_B, Bob\}K_B^{-1}$

Bob

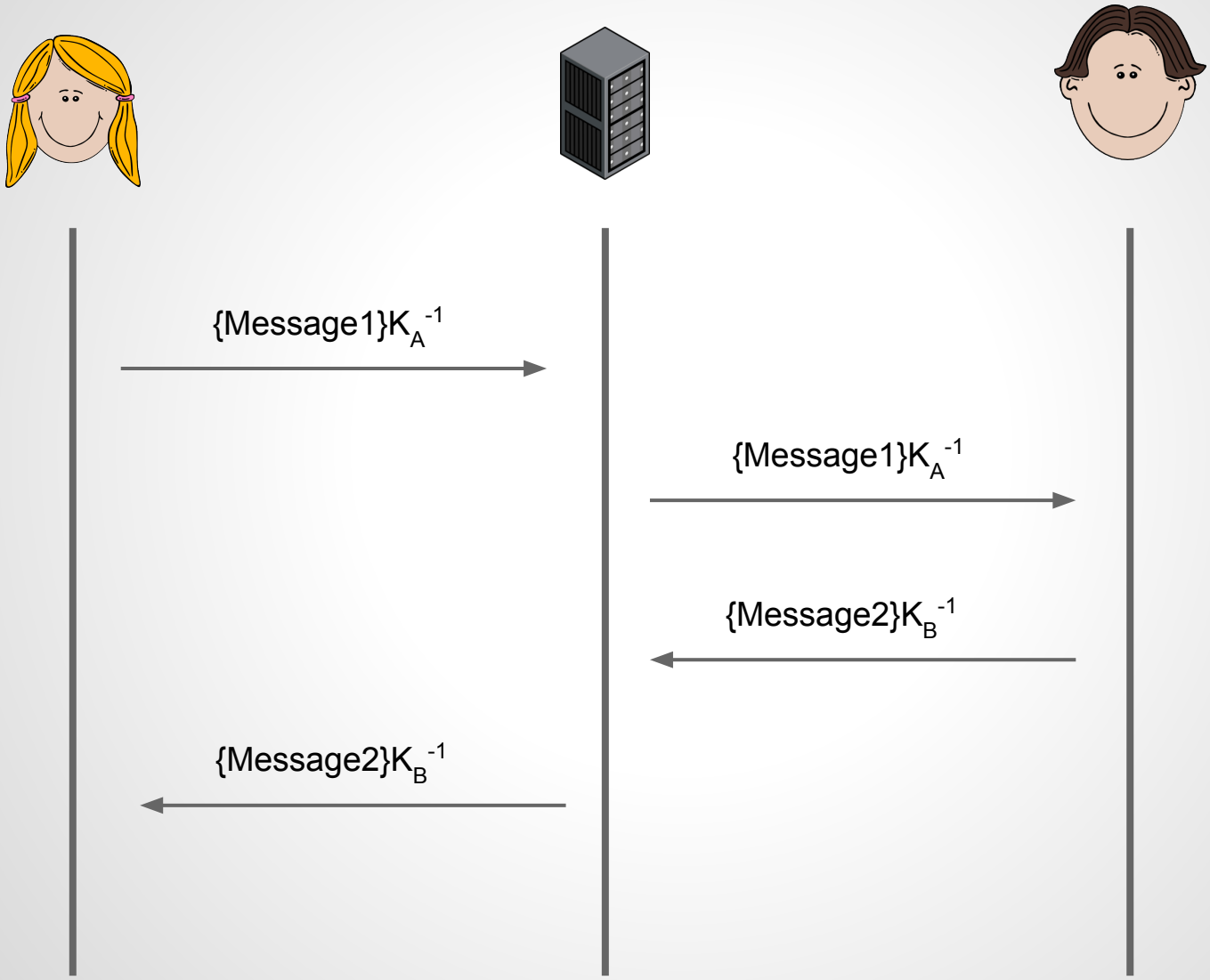
$\{N_B, Alice\}K_A^{-1}$

Key Exchange





Encrypted message sending



Analysis

- Test Results

- Comparison of cryptographic schemes

Test Results

Key Exchange Protocol (using RSA)

Section	Cost (bytes)
Part 1	1146
Part 2	1413
Part 3	563
Part 4	291
Total (Protocol)	**3413**
Sending a message	289
Receiving a message	293
Total (Send/Receive)	**582**

Test Results

Smartphone App - Comparison

	AES	RSA
Generate Keys (milliseconds)	2	781
Encryption (milliseconds)	4	5
Decryption (milliseconds)	9	45
Pre Encryption file size (bytes)	182	182
Post Encryption file size (bytes)	192	256

Comparison of cryptographic schemes

Key Size (bits):

AES	RSA	ECC
56	512	112
128	3072	256
192	7680	384
256	15360	512

512 bit ECC key provides the same security as a 15360 bit RSA key

For equal key lengths, computational effort is similar

Results conclusion

Average computational power and data usage allowance per client

Combination of AES and ECC would provide the best form of security

RSA is more widely used in place of ECC

RSA is an established scheme, but the required key sizes are becoming bigger

ECC provides smaller key sizes and lower energy usage

Further work & Improvements

User testing

Other uses for this system

Conclusion of the project

- Positives
- Negatives
- What I have learnt

Positives

Achievements within the project

Project outcome

Skills and knowledge learnt

Negatives

Astounding size of the cryptographic field

Learning of new software languages and mathematical material

Personal time management

What I have learnt

Personal time management

Better coding and software development practices

Increased knowledge of mathematics and computer science

Demonstration

- Message communication including key exchange protocol

- Android app for analysis

Questions?