# SecureEmailConnect

By: Nate Loria, Ryan Hilty, Adam Cook, and Cory Wiard

# Introduction

Original Idea:

Create an algorithm that encrypt/decrypt messages

Use efficiency to compare our algorithm to other algorithms

What Changed?

Prof. Mohan liked our idea, but challenged us to create something larger

Encrypt emails to secure confidential information that could be passed through email

# Email Tester

How does it work?

- Gmail API
- 3 different algorithms
- External Jar files with methods that aid in connecting and sending EMails

# Methods

Important Methods:

- setMessageContent();
- setMessageDate();
- setMessageSender();
- setMessageSubject();

# Methods Cont.

Connection Methods:

- createSession();
- imapsConnect();
- createFolder();
- setHost("imap.gmail.com");

# Methods Cont.

## Encryption Methods:

Algorithm One:

- encript(String);
- unencript(String);

Algorithm Two:

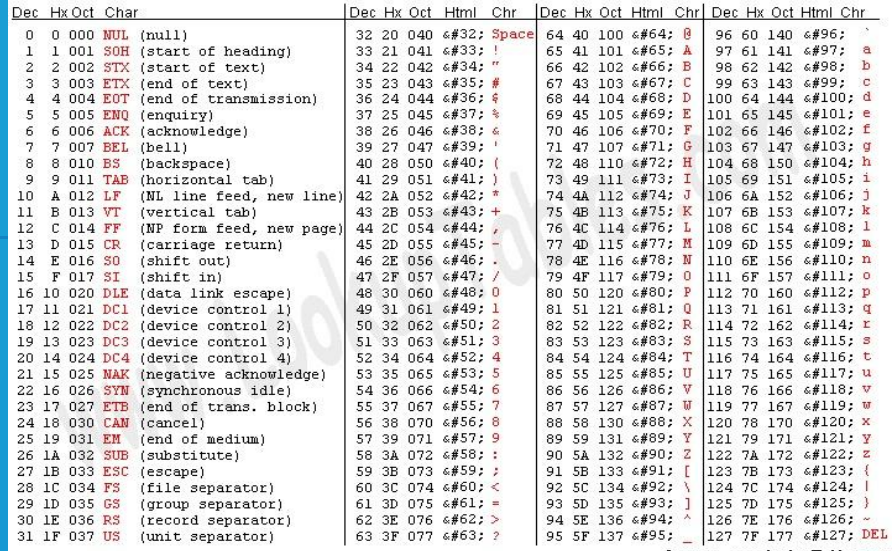
- encypt(String, int);
- decrypt(String, int);

Algorithm Three:

- encrypt(String, String);
- decrypt(String, String);
- setKey();
  - Returns String (key)

# Algorithm 1 (ASCII Non Character Encryption)

- Modified Caesar Cipher
- Usage of ACII
- Generation of special characters
- Special shift



| Dec | Hx | Oct | Char | Dec | Hx | Oct | Html | Chr | Dec | Hx | Oct | Html | Chr | Dec | Hx | Oct | Html | Chr |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 000 | NUL (null) | 32 | 20 | 040 |   | Space | 64 | 40 | 100 | @ | @ | 96 | 60 | 140 | ` | ` |
| 1 | 1 | 001 | SOH (start of heading) | 33 | 21 | 041 | ! | ! | 65 | 41 | 101 | A | A | 97 | 61 | 141 | a | a |
| 2 | 2 | 002 | STX (start of text) | 34 | 22 | 042 | " | " | 66 | 42 | 102 | B | B | 98 | 62 | 142 | b | b |
| 3 | 3 | 003 | ETX (end of text) | 35 | 23 | 043 | # | # | 67 | 43 | 103 | C | C | 99 | 63 | 143 | c | c |
| 4 | 4 | 004 | EOT (end of transmission) | 36 | 24 | 044 | $ | $ | 68 | 44 | 104 | D | D | 100 | 64 | 144 | d | d |
| 5 | 5 | 005 | ENQ (enquiry) | 37 | 25 | 045 | % | % | 69 | 45 | 105 | E | E | 101 | 65 | 145 | e | e |
| 6 | 6 | 006 | ACK (acknowledge) | 38 | 26 | 046 | & | & | 70 | 46 | 106 | F | F | 102 | 66 | 146 | f | f |
| 7 | 7 | 007 | BEL (bell) | 39 | 27 | 047 | ' | ' | 71 | 47 | 107 | G | G | 103 | 67 | 147 | g | g |
| 8 | 8 | 010 | BS (backspace) | 40 | 28 | 050 | ( | ( | 72 | 48 | 110 | H | H | 104 | 68 | 150 | h | h |
| 9 | 9 | 011 | TAB (horizontal tab) | 41 | 29 | 051 | ) | ) | 73 | 49 | 111 | I | I | 105 | 69 | 151 | i | i |
| 10 | A | 012 | LF (NL line feed, new line) | 42 | 2A | 052 | * | * | 74 | 4A | 112 | J | J | 106 | 6A | 152 | j | j |
| 11 | B | 013 | VT (vertical tab) | 43 | 2B | 053 | + | + | 75 | 4B | 113 | K | K | 107 | 6B | 153 | k | k |
| 12 | C | 014 | FF (NP form feed, new page) | 44 | 2C | 054 | , | , | 76 | 4C | 114 | L | L | 108 | 6C | 154 | l | l |
| 13 | D | 015 | CR (carriage return) | 45 | 2D | 055 | - | - | 77 | 4D | 115 | M | M | 109 | 6D | 155 | m | m |
| 14 | E | 016 | SO (shift out) | 46 | 2E | 056 | . | . | 78 | 4E | 116 | N | N | 110 | 6E | 156 | n | n |
| 15 | F | 017 | SI (shift in) | 47 | 2F | 057 | / | / | 79 | 4F | 117 | O | O | 111 | 6F | 157 | o | o |
| 16 | 10 | 020 | DLE (data link escape) | 48 | 30 | 060 | 0 | 0 | 80 | 50 | 120 | P | P | 112 | 70 | 160 | p | p |
| 17 | 11 | 021 | DC1 (device control 1) | 49 | 31 | 061 | 1 | 1 | 81 | 51 | 121 | Q | Q | 113 | 71 | 161 | q | q |
| 18 | 12 | 022 | DC2 (device control 2) | 50 | 32 | 062 | 2 | 2 | 82 | 52 | 122 | R | R | 114 | 72 | 162 | r | r |
| 19 | 13 | 023 | DC3 (device control 3) | 51 | 33 | 063 | 3 | 3 | 83 | 53 | 123 | S | S | 115 | 73 | 163 | s | s |
| 20 | 14 | 024 | DC4 (device control 4) | 52 | 34 | 064 | 4 | 4 | 84 | 54 | 124 | T | T | 116 | 74 | 164 | t | t |
| 21 | 15 | 025 | NAK (negative acknowledge) | 53 | 35 | 065 | 5 | 5 | 85 | 55 | 125 | U | U | 117 | 75 | 165 | u | u |
| 22 | 16 | 026 | SYN (synchronous idle) | 54 | 36 | 066 | 6 | 6 | 86 | 56 | 126 | V | V | 118 | 76 | 166 | v | v |
| 23 | 17 | 027 | ETB (end of trans. block) | 55 | 37 | 067 | 7 | 7 | 87 | 57 | 127 | W | W | 119 | 77 | 167 | w | w |
| 24 | 18 | 030 | CAN (cancel) | 56 | 38 | 070 | 8 | 8 | 88 | 58 | 130 | X | X | 120 | 78 | 170 | x | x |
| 25 | 19 | 031 | EM (end of medium) | 57 | 39 | 071 | 9 | 9 | 89 | 59 | 131 | Y | Y | 121 | 79 | 171 | y | y |
| 26 | 1A | 032 | SUB (substitute) | 58 | 3A | 072 | : | : | 90 | 5A | 132 | Z | Z | 122 | 7A | 172 | z | z |
| 27 | 1B | 033 | ESC (escape) | 59 | 3B | 073 | ; | ; | 91 | 5B | 133 | [ | [ | 123 | 7B | 173 | { | { |
| 28 | 1C | 034 | FS (file separator) | 60 | 3C | 074 | < | < | 92 | 5C | 134 | \ | \ | 124 | 7C | 174 | | | \| |
| 29 | 1D | 035 | GS (group separator) | 61 | 3D | 075 | = | = | 93 | 5D | 135 | ] | ] | 125 | 7D | 175 | } | } |
| 30 | 1E | 036 | RS (record separator) | 62 | 3E | 076 | > | > | 94 | 5E | 136 | ^ | ^ | 126 | 7E | 176 | ~ | ~ |
| 31 | 1F | 037 | US (unit separator) | 63 | 3F | 077 | ? | ? | 95 | 5F | 137 | _ | _ | 127 | 7F | 177 | &#127; | DEL |

Source: www.LookupTables.com

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 128 | Ç | 144 | É | 161 | í | 177 | ▒ | 193 | ┴ | 209 | ╤ | 225 | ß | 241 | ± |
| 129 | ü | 145 | æ | 162 | ó | 178 | ▓ | 194 | ┬ | 210 | ╥ | 226 | Γ | 242 | ≥ |
| 130 | é | 146 | Æ | 163 | ú | 179 | │ | 195 | ├ | 211 | ╙ | 227 | π | 243 | ≤ |
| 131 | â | 147 | ô | 164 | ñ | 180 | ┤ | 196 | ─ | 212 | ╘ | 228 | Σ | 244 | ⌠ |
| 132 | ä | 148 | ö | 165 | Ñ | 181 | ╡ | 197 | ┼ | 213 | ╒ | 229 | σ | 245 | ⌡ |
| 133 | à | 149 | ò | 166 | ª | 182 | ╢ | 198 | ╞ | 214 | ╓ | 230 | µ | 246 | ÷ |
| 134 | å | 150 | û | 167 | º | 183 | ╖ | 199 | ╟ | 215 | ╫ | 231 | τ | 247 | ≈ |
| 135 | ç | 151 | ù | 168 | ¿ | 184 | ╕ | 200 | ╚ | 216 | ╪ | 232 | Φ | 248 | ° |
| 136 | ê | 152 | _ | 169 | _ | 185 | ╣ | 201 | ╔ | 217 | ┘ | 233 | Θ | 249 | · |
| 137 | ë | 153 | Ö | 170 | ¬ | 186 | ║ | 202 | ╩ | 218 | ┌ | 234 | Ω | 250 | · |
| 138 | è | 154 | Ü | 171 | ½ | 187 | ╗ | 203 | ╦ | 219 | ▮ | 235 | δ | 251 | √ |
| 139 | ï | 156 | £ | 172 | ¼ | 188 | ╝ | 204 | ╠ | 220 | ▄ | 236 | ∞ | 252 | _ |
| 140 | î | 157 | ¥ | 173 | ¡ | 189 | ╜ | 205 | ═ | 221 | ▌ | 237 | ϕ | 253 | ² |
| 141 | ì | 158 | _ | 174 | « | 190 | ╛ | 206 | ╬ | 222 | ▐ | 238 | ε | 254 | ■ |
| 142 | Ä | 159 | ƒ | 175 | » | 191 | ┐ | 207 | ╧ | 223 | ▀ | 239 | ∩ | 255 | |
| 143 | Å | 160 | á | 176 | ░ | 192 | └ | 208 | ╨ | 224 | α | 240 | ≡ | | |

Source: www.LookupTables.com

# Algorithm 2 (CaesarCipher)

- One of earliest known cryptography

- Type of substitution of cipher
  - Letters in plain text are “shifted” down by letters in the alphabet
- Ex.
  - Shift of A by 1 = B

# Algorithm 3 (AES)

- Advanced Encryption Standard
  - Uses Java Cryptography Extension
- Provides framework and implementation for encryption and key generation
- Creates 3 extensions imported

1. Key Generator
   a. generates secret keys in program
2. Secret Key can be a number, word, or string of rand. letters
3. Cipher- cryptographic algorithm used to encrypt and decrypt data

# AES Cont.

- Takes user message (string) encodes it into a sequence of bytes and stores it into an array
  - byte[] plainTextByte = plainText.getBytes();
  - byte[] encrypt = cipher.dofinal(plainTextByte)
- AES uses the package Base64 to implement an encoder and decoder for byte data
- Base64.Encoder encoder = Base64.getEncoder();
- String encryptedText = encoder.encodeToString(encryptedByte);

# Challenges

- Exceptions
- Using the correct server to connect to  to send emails
- Intaking a string without any extra character
- Transporting key data with messages
- Ensuring the key for AES is encoded correctly

# Memory Testing

### Initial Memory Usage

The initial memory allocated to the encryption is taken

### Execution of Encryption

Execution of encryption method, usage of memory collected

### Total Memory Usage

Total memory usage is calculated from the difference of initial and available through execution

### Methods

```
Runtime runtime = Runtime.getRuntime();
runtime.gc();
long memory = runtime.totalMemory() - runtime.freeMemory();
```

# Memory Usage

```
**********  Total Memory Usage of AlgorithmOne  **********

Total Memory Allocated: 8388608

Total Memory used in bytes: 3138024
Total Memory used in kilobytes: 3064
Total Memory used in megabytes: 2


Message sent successfully!

**********  Total Memory Usage of SecureEmailConnect  **********

Total Memory Allocated: 8388608

Total Memory used in bytes: 4789256
Total Memory used in kilobytes: 4677
Total Memory used in megabytes: 4

ewire23-14:CS110-Final-Project adam$
```

# Memory Usage

```
**********  Total Memory Usage of AlgorithmAES  **********

Total Memory Allocated: 8388608

Total Memory used in bytes: 3154008
Total Memory used in kilobytes: 3080
Total Memory used in megabytes: 3


Message sent successfully!

**********  Total Memory Usage of SecureEmailConnect  **********

Total Memory Allocated: 8388608

Total Memory used in bytes: 4781832
Total Memory used in kilobytes: 4669
Total Memory used in megabytes: 4
```

```
Please choose which algorithm to encrypt with (1, 2 or 3):
2

Please enter the amount to encrypt by (1-26):
20

**********  Total Memory Usage of AlgorithmOne **********

Total Memory Allocated: 8388608

Total Memory used in bytes: 3133512
Total Memory used in kilobytes: 3060
Total Memory used in megabytes: 2

**********  Total Memory Usage of SecureEmailConnect  **********

Total Memory Allocated: 8388608

Total Memory used in bytes: 3157880
Total Memory used in kilobytes: 3083
Total Memory used in megabytes: 3


Message sent successfully!
```

# Future Additions

- Creating a full website
- Additional Encryption methods
- Implementation in Allegheny
- Encrypting Different file types