

K.E.G. Card Cipher

(Kolor Encryption Gate)

by Karl Zander

karl.c.zander@gmail.com

Abstract. KEG is a playing card cipher designed for use with ordinary playing cards. It's motions resemble a rotor machine but play resembles a card game. It was designed to be resistant to modern attacks both by human and machine. It's key length and strength is 52 factorial (52!). KEG's key stream is designed to be non-periodic.

Design:

KEG is a non-linear stream cipher that fits in the palm of your hand. The discard of cards and pick up upon encountering the gate color provides non-linearity to what would seem like an ordinary rotor machine construction. KEG was designed to be easy to use and easy to remember for users who want strong encryption for A-Z messages.

KEG's output resembles an ideal cipher and in testing is not distinguishable from a truly random A-Z sequence.

Specification:

State = 52 cards

Key Length = 52 cards

Gate Color = color of the first card in the keyed deck

KEG is a hand held rotor machine that feels like a card game. It consists of an encryption deck pile and discard pile.

Operational setup:

Start by creating a lookup dictionary that converts a card into it's assigned number. Such a lookup dictionary should look like this:

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|----|----|----|----|----|-----|----|----|----|
| AC | 2C | 3C | 4C | 5C | 6C | 7C | 8C | 9C | 10C | JC | QC | KC | AS | 2S | 3S | 4S | 5S | 6S | 7S | 8S | 9S | 10S | JS | QS | KS |
| 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 |
| 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 |

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|----|----|----|----|----|-----|----|----|----|
| AH | 2H | 3H | 4H | 5H | 6H | 7H | 8H | 9H | 10H | JH | QH | KH | AD | 2D | 3D | 4D | 5D | 6D | 7D | 8D | 9D | 10D | JD | QD | KD |
| 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 |

Card values are used for modular addition of plaintext letters.

Stepping values

AC 2C 3C 4C 5C 6C 7C 8C 9C 10C JC QC KC AS 2S 3S 4S 5S 6S 7S 8S 9S 10S JS QS KS
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

AH 2H 3H 4H 5H 6H 7H 8H 9H 10H JH QH KH AD 2D 3D 4D 5D 6D 7D 8D 9D 10D JD QD KD
26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51

Encryption Pile Stepping:

Before any encryption of plain text is done, the encryption pile is stepped.

The stepping card is always the second card in the encryption pile.

1. The stepping card's value is looked up in the key and the color is noted.
2. If the color of the card matches the gate color then the discard pile is picked up and placed at the rear of the encryption pile.
3. Discard the stepping card.
4. Step the encryption pile by removing the first card and placing it at the rear.

Finally, step the encryption pile by the noted stepping card's value (0-51). This means repeating step #4 by the value of the stepping card.

Encryption:

Step the encryption deck by using the Encryption Pile Stepping instructions.

Encryption of the first letter in the plaintext is performed by modular addition of the letter (0-25) with the card value of the first card in the encryption pile.

Decryption:

Step the encryption deck by using the Encryption Pile Stepping instructions.

Decryption of the first letter in the ciphertext is performed by modular subtraction of the letter (0-25) with the card value of the first card in the encryption pile.

Test Vector:

Key: 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, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51

Example message: LETUSPLAYKEGTOGETHER
Example cipher text: ONONIANLXQHEYCNUGIAA