

CIS 3362 9/20/23

Transposition

1. Permutation
2. Column Transposition
3. Double Transposition

Permutation

Key

2	5	1	3	4
---	---	---	---	---

2 3 5 4 1
H O U S E

v v

LETUS	GO TO	THE STORE	EXXX
-------	-------	-----------	------

ESLTUOTGT OEOHSTEXRXX

	1	2	3	4	5
E	S	L	T	U	
O	T	G	T	O	
E	O	H	S	T	
E	X	R	X	X	

Swap columns
reorder columns
so the top makes
words

Column Transposition

Keyword: C O M P U T E R

(could be perm)

* * * * *
1 4 3 5 8 7 2 6

T H I S M Y T E
S T F O R T R A
N S P O S I T I
O N L E T S S E
E H O W G A R B
L E D T H E O U
T P U L L O O K

57
letters

$57 \div 8 = 1$
small = $57/8 = 7$
large = $7+1 = 8$

Cipher

S

1 2 3 4 5 6 7 8

T S N O E L T S T R T S R O O K F P L O D U

H T S N H E A S O O E W T L E A I E B U K Y T I S A E O

M R S T G H L

loop both ways each group

⇒ 1 4 3 5 8 5 7 2 6
7 2 6

T H I S M Y T E
S T F O R T R A
N S P O S I T I
O N L E T S S E
E H O W G A R B
L E D T H E O U
T P U L L O O K
S

We can
read it!

1 6 10 2 8 3 9 4 7 11 5
A B R A C A D A B R A

Double Transposition

2 diff keywords

```
char* doublet(char* plain, char* key1, char* key2) {
```

```
    char* mid = trans(plain, key1);
```

```
    return trans(mid, key2);
```

```
}
```

```
char* invdoublet(char* cipher, char* key1, char* key2) {
```

```
    char* mid = invtrans(cipher, key2);
```

```
    return invtrans(mid, key1);
```

```
}
```

We can get an equivalent single trans
w/a keyword of length $\text{len}(\text{key1}) + \text{len}(\text{key2})$

Can get more obfuscation from combining
multiple simple operations + this is easier than
trying to come up with a single more
complicated equivalent operation.