Salt

1. Create a public key (e, n)

*n = p*q*, where *p* and *q* are two prime numbers
e must be relatively prime to *(p-1)*(q-1); gcd(e, (p-1)*(q-1)) = 1*
~just pick an e that works

p = 43, q = 59
n = p*q = 2537
e = gcd(e, (p -1)*(q -1)) = gcd(e, 42*58) = 1
e = 13

private key (e, n) = 13, 2537

2. Create a private key (d, n)

d is the inverse of *e mod (p-1)*(q-1)*
~use the extended Euclidean algorithm to solve for d

What is the inverse of 13 mod 42*58?
42*58 = 2436
Using the extended Euclidean algorithm we find that d = 937.

private key (937, 2537)

3. Receives the message and decodes

$M = C^d \bmod n$, where *C* is the encoded message and *d* and *n* are from the private key
~use fast modular exponentiation to solve for M

$2081^{937} \bmod 2537 = 1819$
$2182^{937} \bmod 2537 = 1415$

M = 18, 19, 14, 15 = STOP

Pepper

1. Receives a public key (e, n)

private key (e, n) = 13, 2537

2. Encodes a message

$C = M^e \bmod n$, where *M* is the message, and *e* and *n* are from the public key
~ use fast modular exponentiation (FME) to solve for C

s = 18, t = 19, o = 14, p = 15
Because 2525 < 2537 < 252525 we must group the these numbers into blocks of four digits, leaving us with 1819 and 1415
$1819^{13} \bmod 2537 = 2081$
$1415^{13} \bmod 2537 = 2182$
C = [2081, 2182]

Alphabet conversion table

A	=	1	K	=	11	U	=	21
B	=	2	L	=	12	V	=	22
C	=	3	M	=	13	W	=	23
D	=	4	N	=	14	X	=	24
E	=	5	O	=	15	Y	=	25
F	=	6	P	=	16	Z	=	26
G	=	7	Q	=	17			
H	=	8	R	=	18			
I	=	9	S	=	19			
J	=	10	T	=	20			