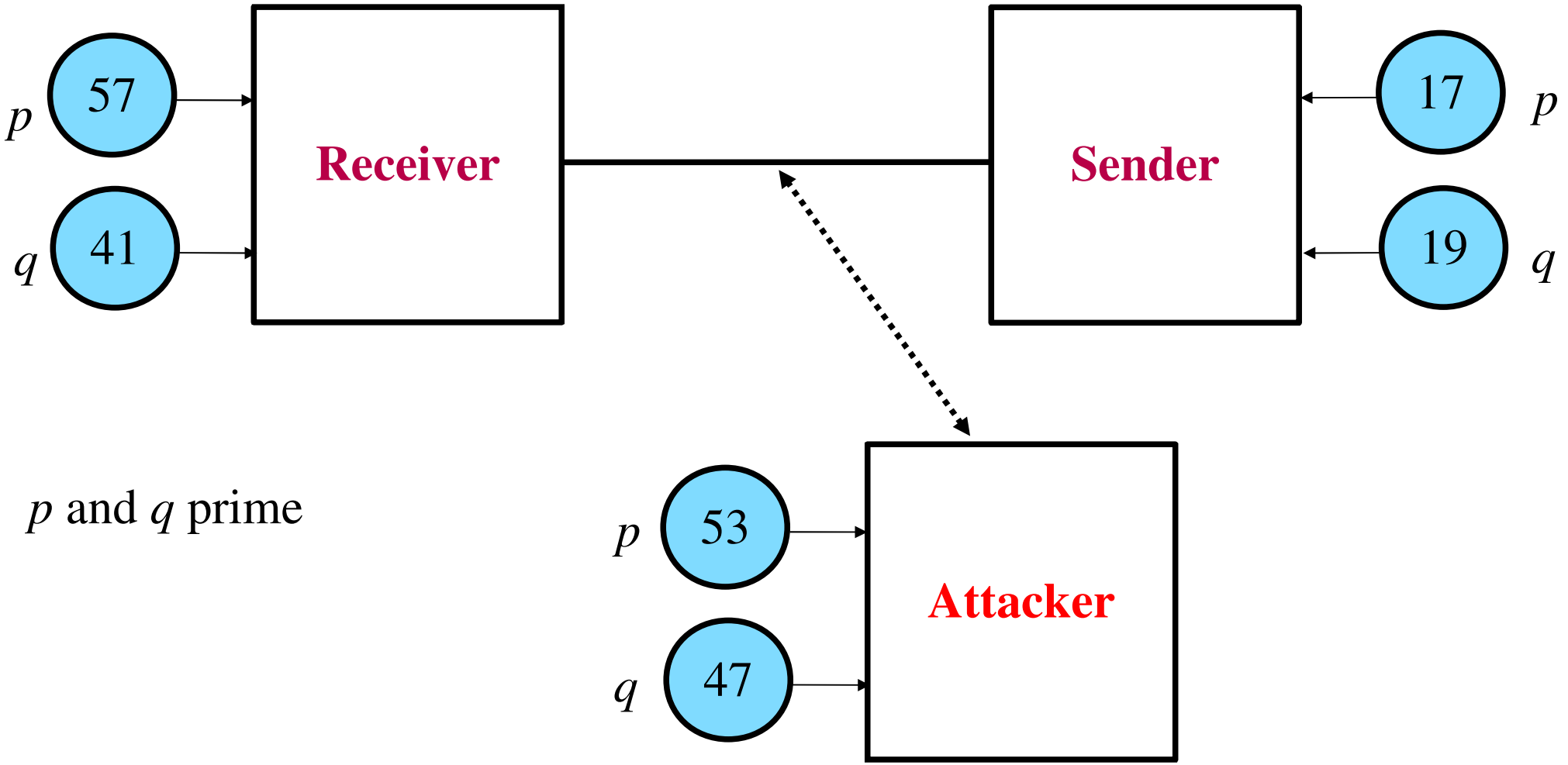
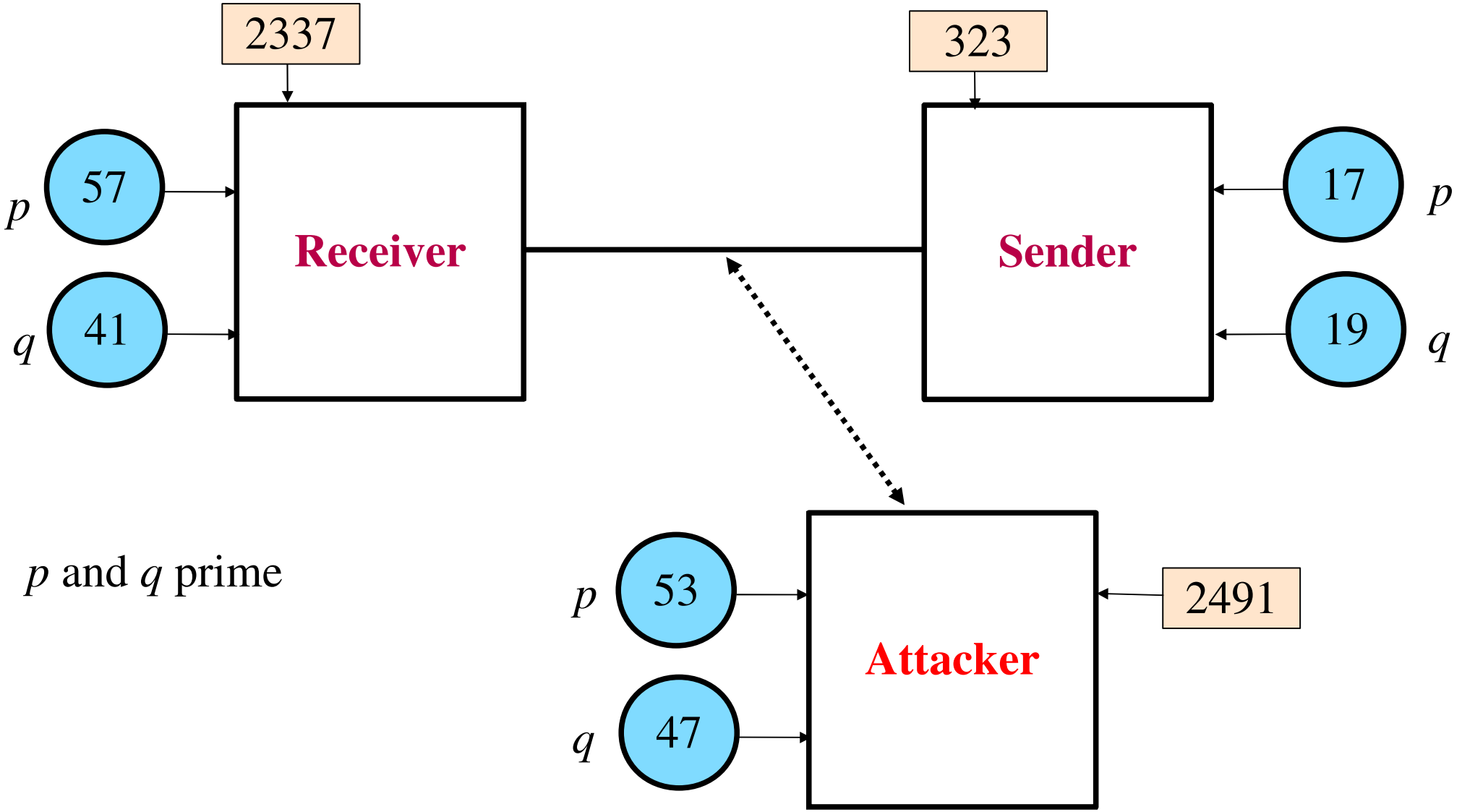
Public Key Cryptosystems - RSA



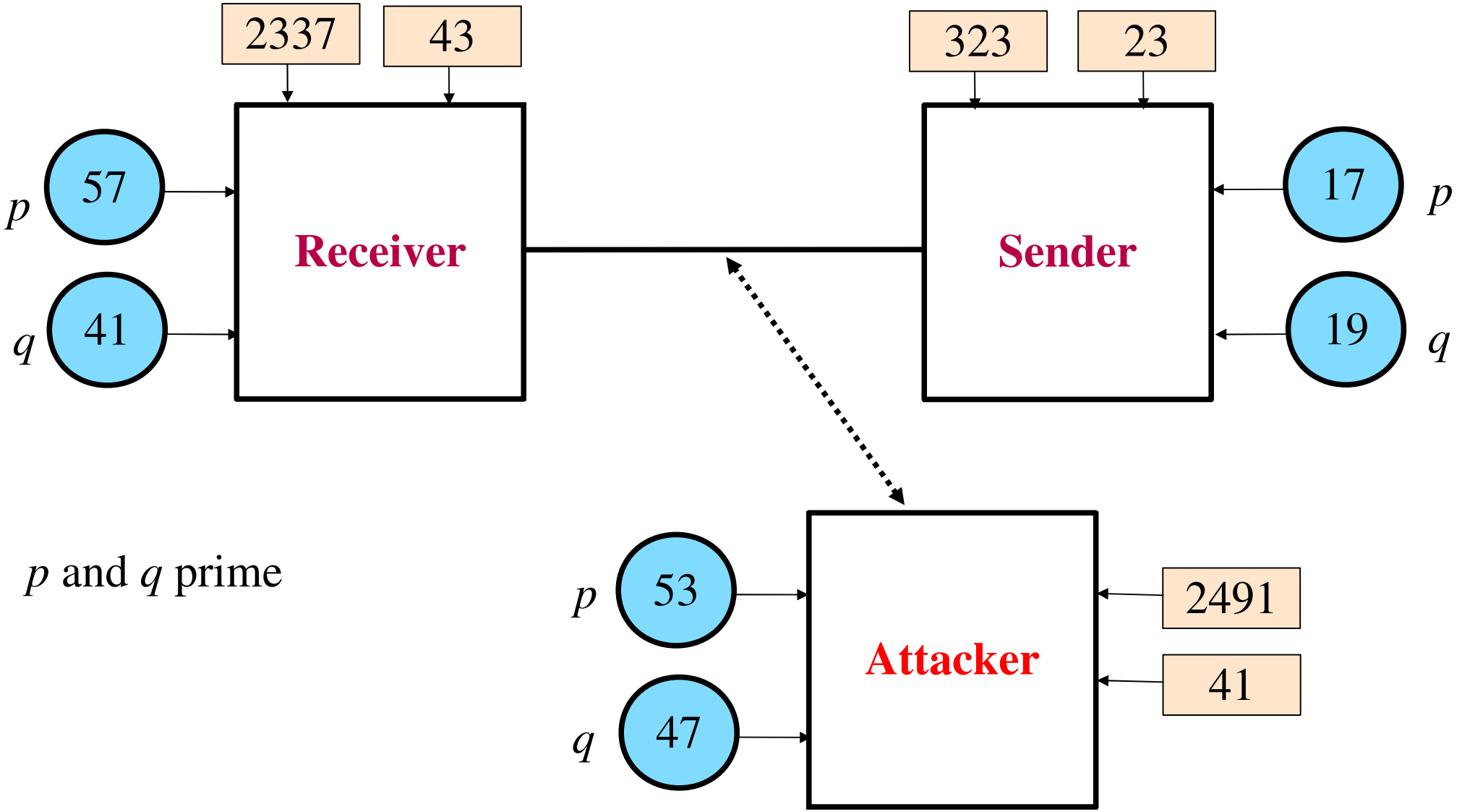
Public Key Cryptosystems - RSA

Compute numbers $n = p*q$



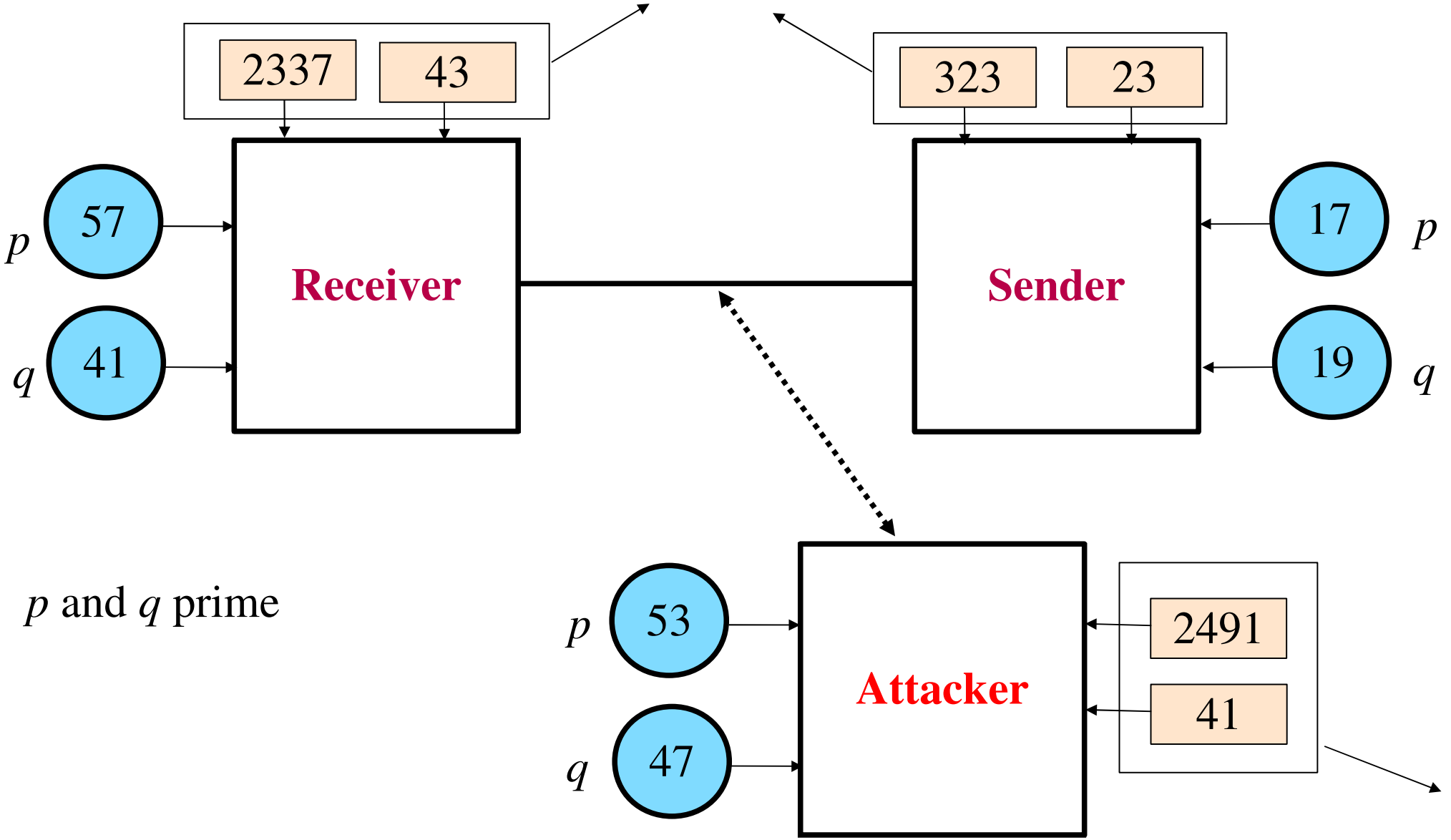
Public Key Cryptosystems - RSA

Choose e prime relative to $(p\text{-}1)(q\text{-}1)$



Public Key Cryptosystems - RSA

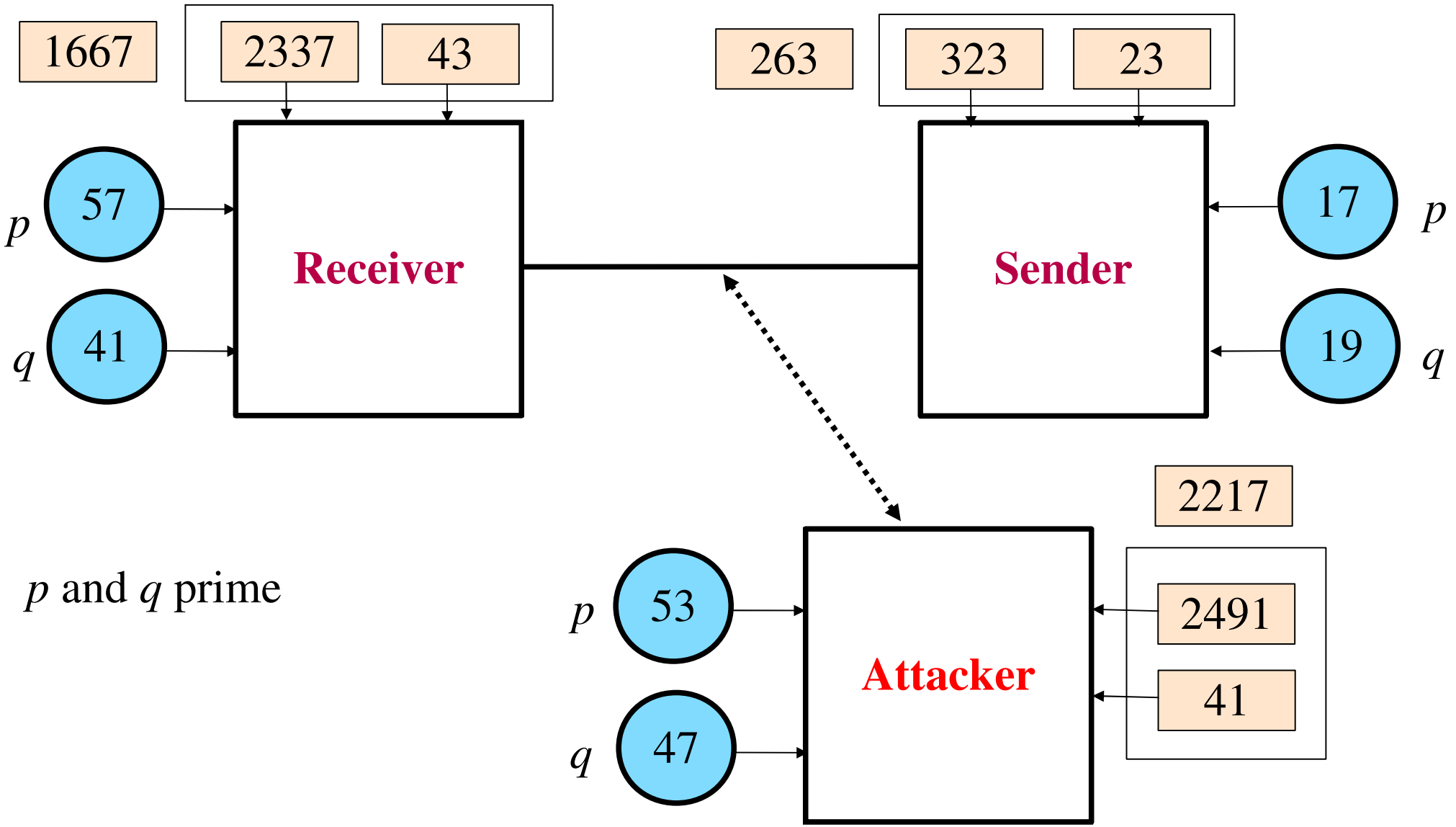
Publish <*n*,*e*> pair as the public key



p and *q* prime

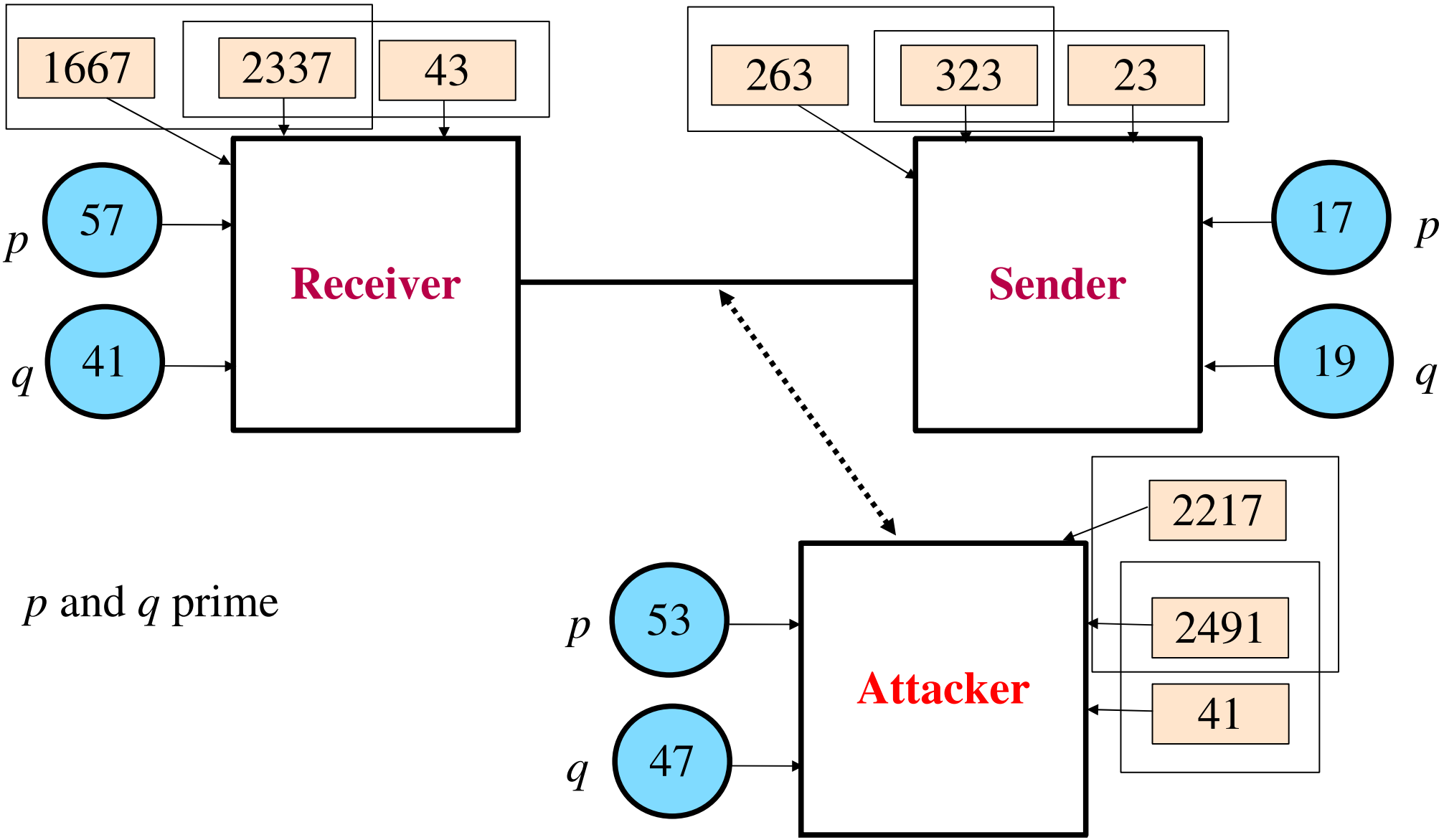
Public Key Cryptosystems - RSA

Find d such that (e*d -1) is divisible by (p-1)(q-1)



Public Key Cryptosystems - RSA

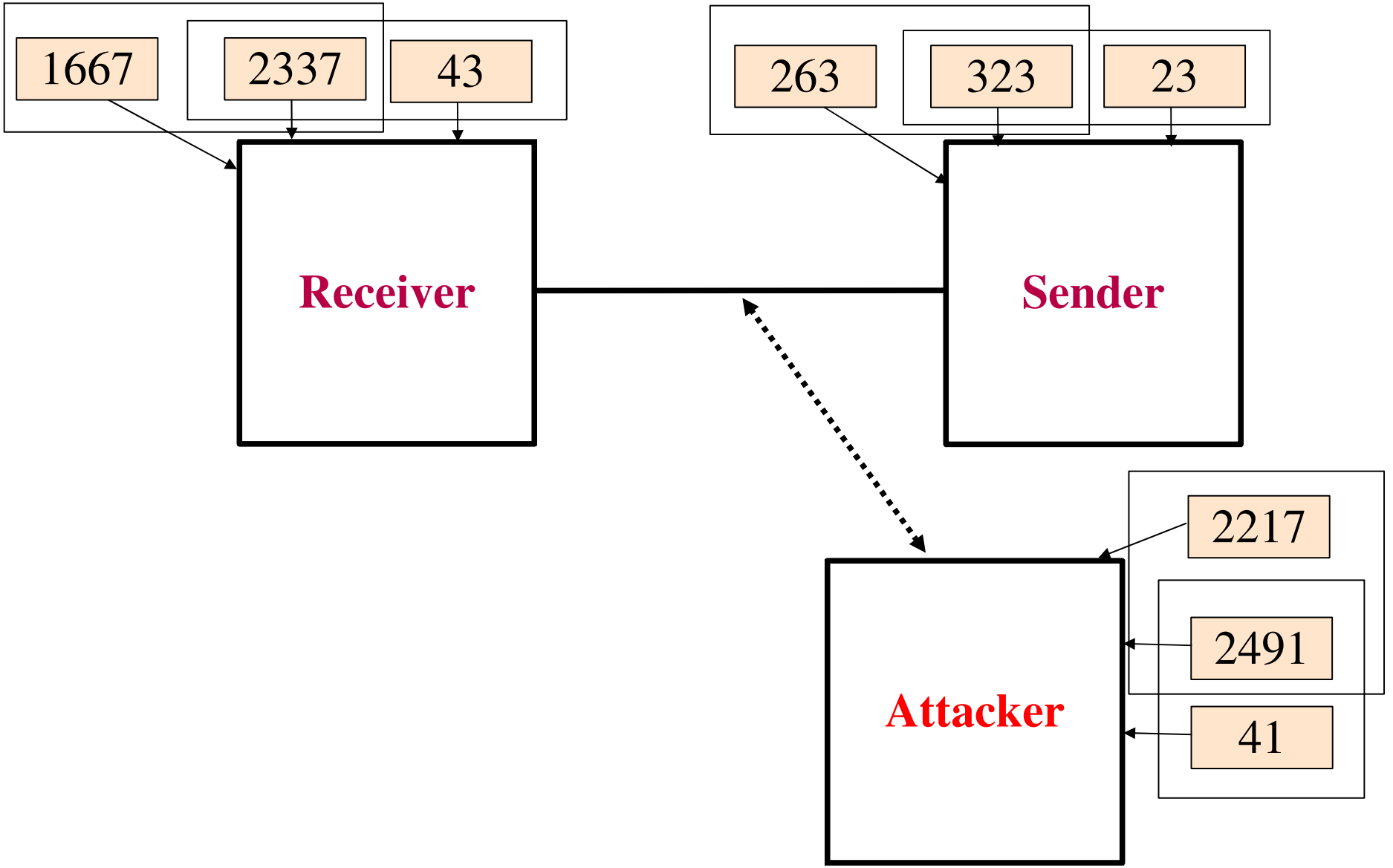
Keep <*d*,*n*> as the private key



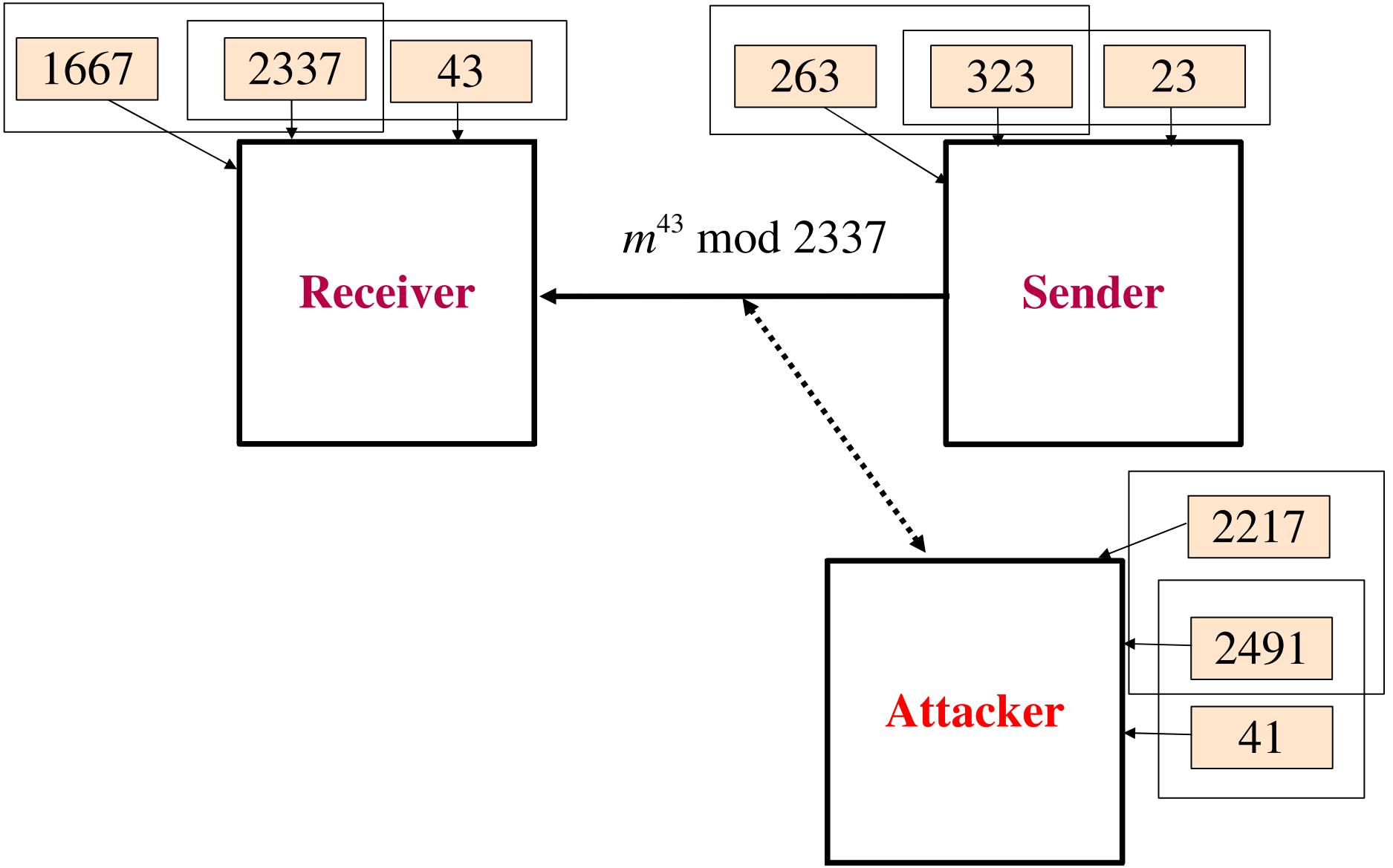
p and *q* prime

Public Key Cryptosystems - RSA

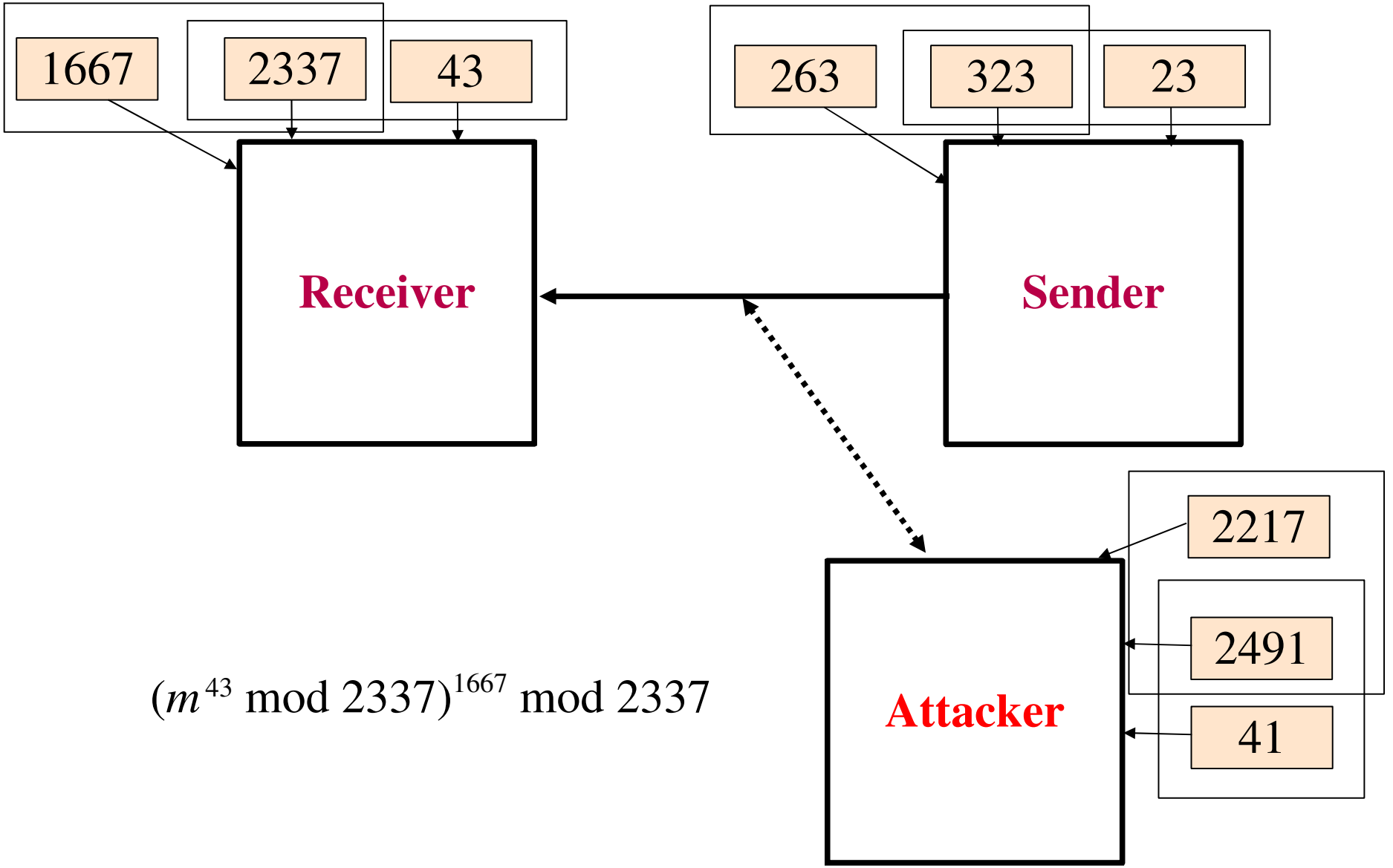
Toss p and q



Public Key Cryptosystems - RSA

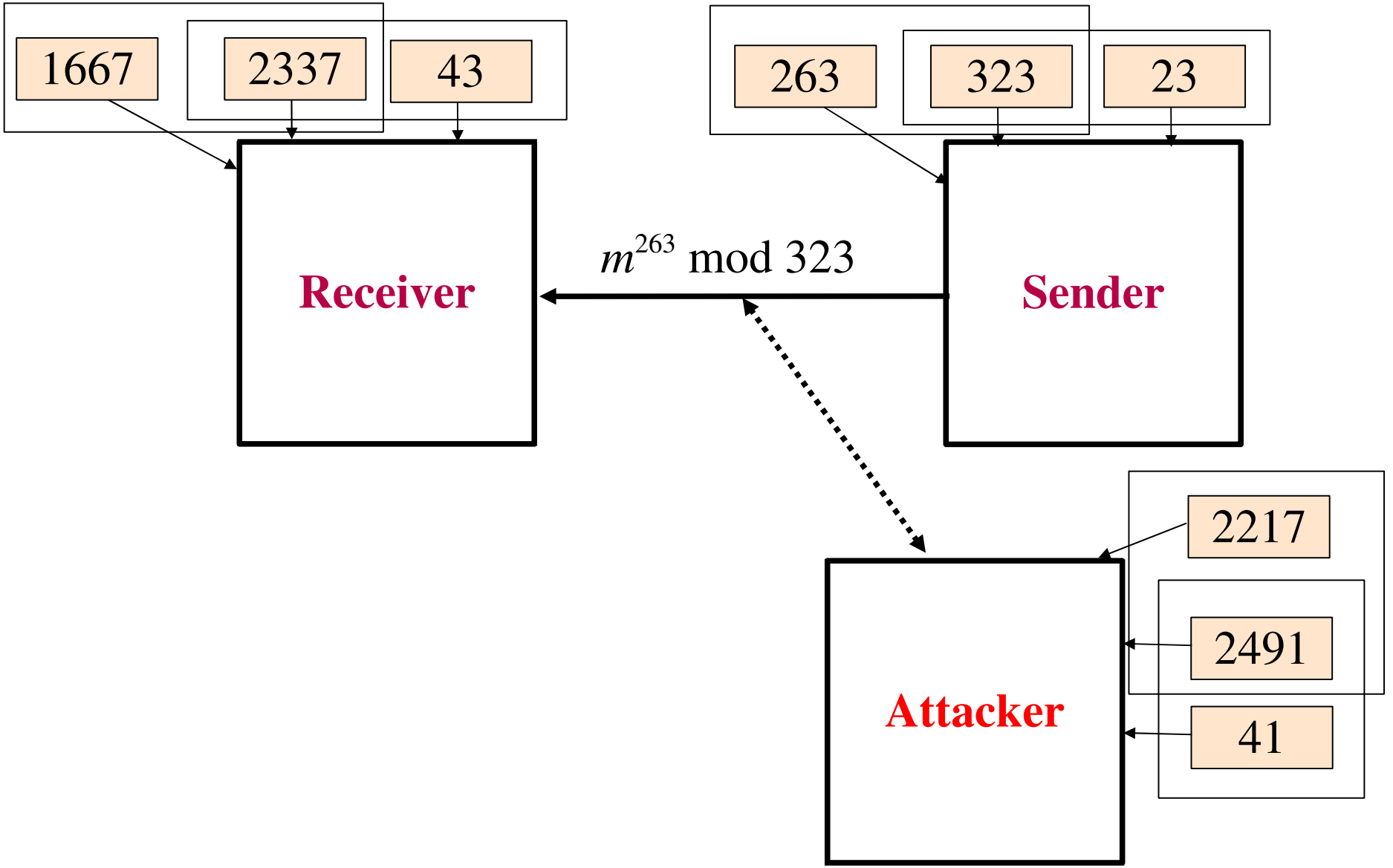


Public Key Cryptosystems - RSA

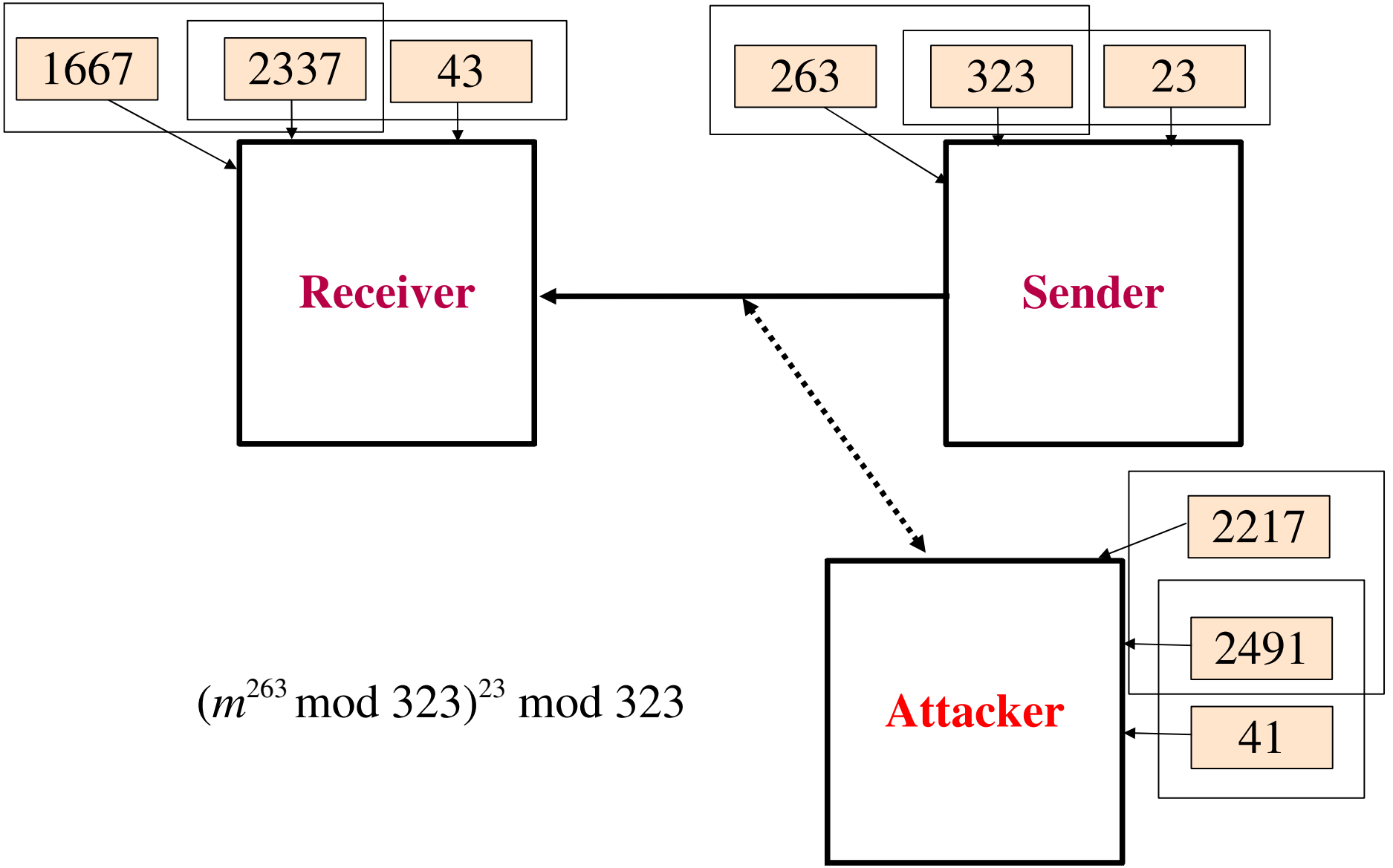


$(m^{43} \bmod 2337)^{1667} \bmod 2337$

Public Key Cryptosystems - RSA, signing



Public Key Cryptosystems - RSA, signing



$(m^{263} \bmod 323)^{23} \bmod 323$

Public Key Cryptosystems - RSA, exponentiating

$$25663^{55637} \mod 78837$$

Public Key Cryptosystems - RSA, exponentiating

$$25663^{55637} \mod 78837$$

Yikes!

Public Key Cryptosystems - RSA, exponentiating

$$25663^{55637} \bmod 78837$$ Yikes!

Rescued by:

$$a^x * b^y \bmod p = (a^x \bmod p) * (b^y \bmod p) \bmod p$$

Public Key Cryptosystems - RSA, exponentiating

$$25663^{55637} \mod 78837$$

$$25663^{2} \mod 78837$$

...

That is, do the modular reduction after each multiplication.

Public Key Cryptosystems - RSA, find primes

Probability that a random number n is prime: $1/\ln(n)$

For 100 digit number this is 1/230.

Public Key Cryptosystems - RSA, find primes

Probability that a random number n is prime: $1/\ln(n)$

For 100 digit number this is 1/230.

But how to test for being prime?

Public Key Cryptosystems - RSA, find primes

Probability that a random number n is prime: $1/\ln(n)$

For 100 digit number this is 1/230.

But how to test for being prime?

If p is prime and $0 < a < p$, then $a^{p-1} = 1 \bmod p$

Ex: $3^{(5-1)} = 81 = 1 \bmod 5$

$36^{(29-1)} = 377111712813960320133663211989001573037506 56$

$= 1 \bmod 29$

Public Key Cryptosystems - RSA, find primes

Probability that a random number n is prime: $1/\ln(n)$

For 100 digit number this is 1/230.

But how to test for being prime?

If p is prime and $0 < a < p$, then $a^{p-1} = 1 \bmod p$

Ex: $3^{(5-1)} = 81 = 1 \bmod 5$

$36^{(29-1)} = 377111712813960320133663211989001573037506 56$

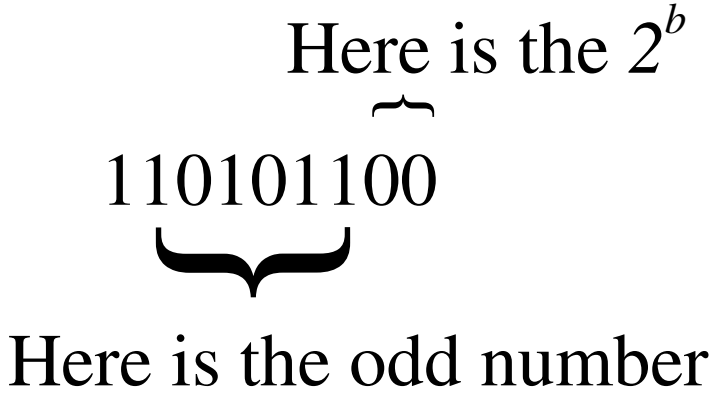
$= 1 \bmod 29$

Pr(p isn't prime but $a^{p-1} = 1 \bmod p$) = 1/1000000000000000

Public Key Cryptosystems - RSA, find primes

Can always express a number n-1 as $2^b c$ for some odd number c.

ex: $48 = 2^4 3$



Public Key Cryptosystems - RSA, find primes

Can always express a number n-1 as $2^b c$ for some odd number c.

ex: $48 = 2^4 3$

Then can compute a^{n-1} mod n by computing a^c mod n and squaring the result b times. If the result is not 1 then n is not prime.

Public Key Cryptosystems - RSA, find primes

Trivial square roots of 1 mod p : 1 mod p and -1 mod p

If p is prime, there are no nontrivial square roots of 1 mod p

Let x be a square root of 1 mod p. Then $x^2 = 1 \bmod p$.

Or, $(x\text{-}1)(x+1) = 0 \bmod p$.

But x-1 and x+1 are divisible by prime p. Hence, the product cannot be divisible by p. Therefore x does not exist.

Public Key Cryptosystems - RSA, find primes

Consider $n-1 = 2^b c$ again. If p is prime then $a^c = 1 \bmod p$ or for some r, $a^{2^r c} = -1 \bmod p$.

Public Key Cryptosystems - RSA, find primes

Choose a random odd integer p to test.
Calculate b = # times 2 divides p-1.

Calculate m such that $p = 1 + 2^b m$.

Choose a random integer a such that $0 < a < p$.

If $a^m \equiv 1 \bmod p$ || $a^{2^j m} \equiv -1 \bmod p$, for some $0 \leq j \leq b\text{-}1$, then p passes the test. A prime will pass the test for all a.

Public Key Cryptosystems - RSA, find primes

Choose a random odd integer p to test.
Calculate b = # times 2 divides p-1.

Calculate m such that $p = 1 + 2^{b} m$.

Choose a random integer a such that $0 < a < p$.

If $a^{m} \equiv 1 \bmod p$ || $a^{2^{j}m} \equiv -1 \bmod p$, for some $0 \leq j \leq b$-1,
then p passes the test. A prime will pass the test for all a.

A non prime number passes the test for at most 1/4 of all possible a.
So, repeat N times and probability of error is $(1/4)^{N}$.

Public Key Cryptosystems - RSA, picking *d* and *e*

Choose *e* first, then find *p* and *q* so (*p*-1) and (*q*-1) are relatively prime to *e*

RSA is no less secure if e is always the same and small

Popular values for *e* are 3 and 65537

For $e = 3$, though, must pad message or else ciphertext = plaintext

Choose $p \equiv 2 \bmod 3$ so $p-1 = 1 \bmod 3$

So, choose random odd number, multiply by 3 and add 2,
then test for primality