In [9]:
```
import random
import hashlib

g=9
p=1001

a=random.randint(5, 10)

b=random.randint(10,20)

A = (g**a) % p
B = (g**b) % p

print('g: ',g,' (a shared value), n: ',p, ' (a prime number)')

print('\nAlice calculates:')
print('a (Alice random): ',a)
print('Alice value (A): ',A,' (g^a) mod p')

print('\nBob calculates:')
print('b (Bob random): ',b)
print('Bob value (B): ',B,' (g^b) mod p')

print('\nAlice calculates:')
keyA=(B**a) % p
print('Key: ',keyA,' (B^a) mod p')
print('Key: ',hashlib.sha256(str(keyA).encode()).hexdigest())

print('\nBob calculates:')
keyB=(A**b) % p
print('Key: ',keyB,' (A^b) mod p')
print('Key: ',hashlib.sha256(str(keyB).encode()).hexdigest())
```

```
g:  9  (a shared value), n:  1001  (a prime number)

Alice calculates:
a (Alice random):  5
Alice value (A):  991  (g^a) mod p

Bob calculates:
b (Bob random):  11
Bob value (B):  900  (g^b) mod p

Alice calculates:
Key:  100  (B^a) mod p
Key:  ad57366865126e55649ecb23ae1d48887544976efea46a48eb5d85a6eeb4d306

Bob calculates:
Key:  100  (A^b) mod p
Key:  ad57366865126e55649ecb23ae1d48887544976efea46a48eb5d85a6eeb4d306
```

In [ ]: