CS558 Network Security

Lecture 11: SSL and TLS

hello

Transport layer Security

BOSTON UNIVERSITY

CS558 Network Security

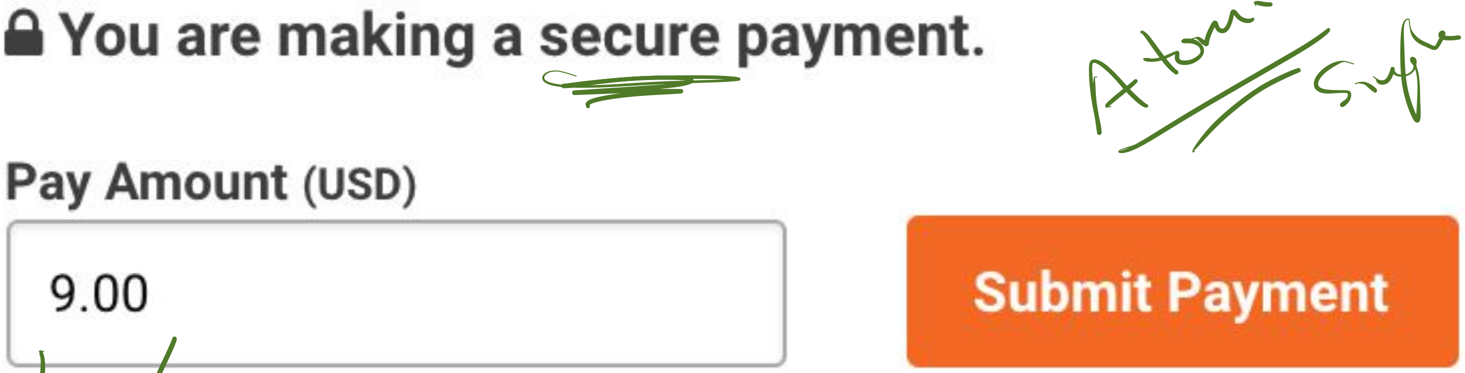
Lecture 11: ~~SSL~~ and TLS*

* Version (1.2)

Where we have been so far

Built the backbone of secure internet

TLS / SSL reduces all the attacks to DOS



🔒 You are making a secure payment.

Pay Amount (USD)

9.00

Atomic

Single use

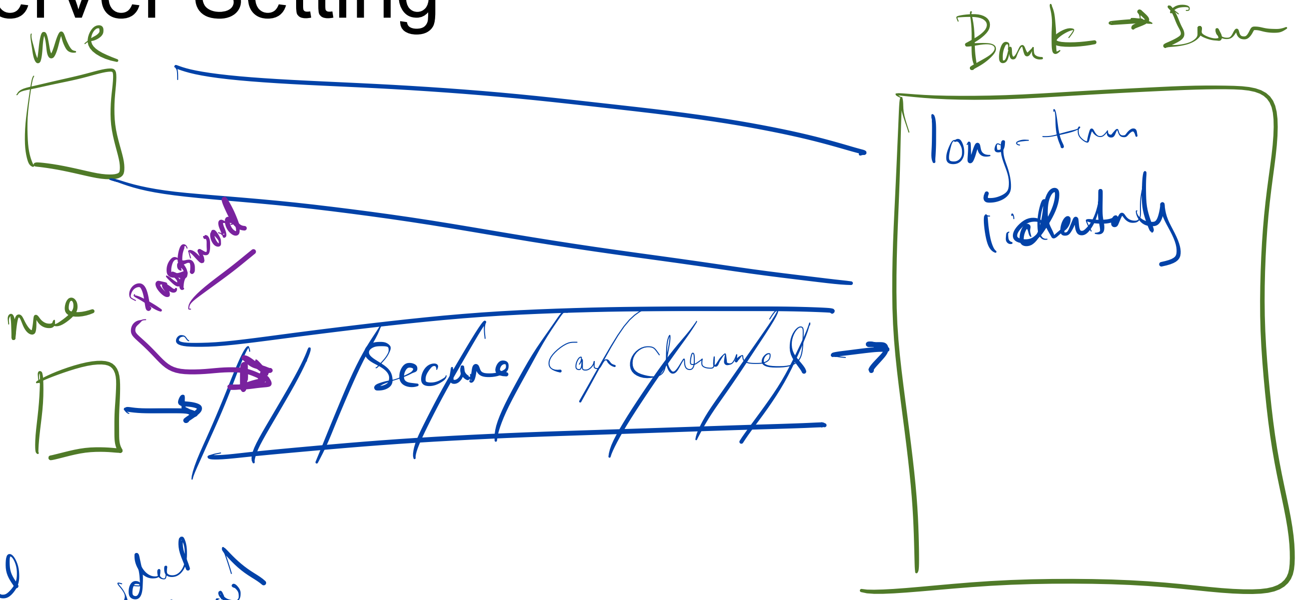
actually only $9

no up and down

Private

from whom & to whom

Client Server Setting

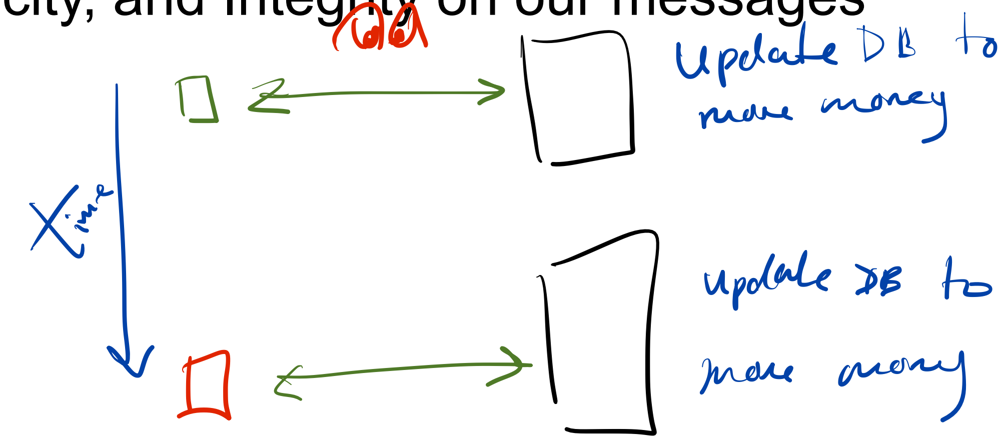


no real id provided by the protocol

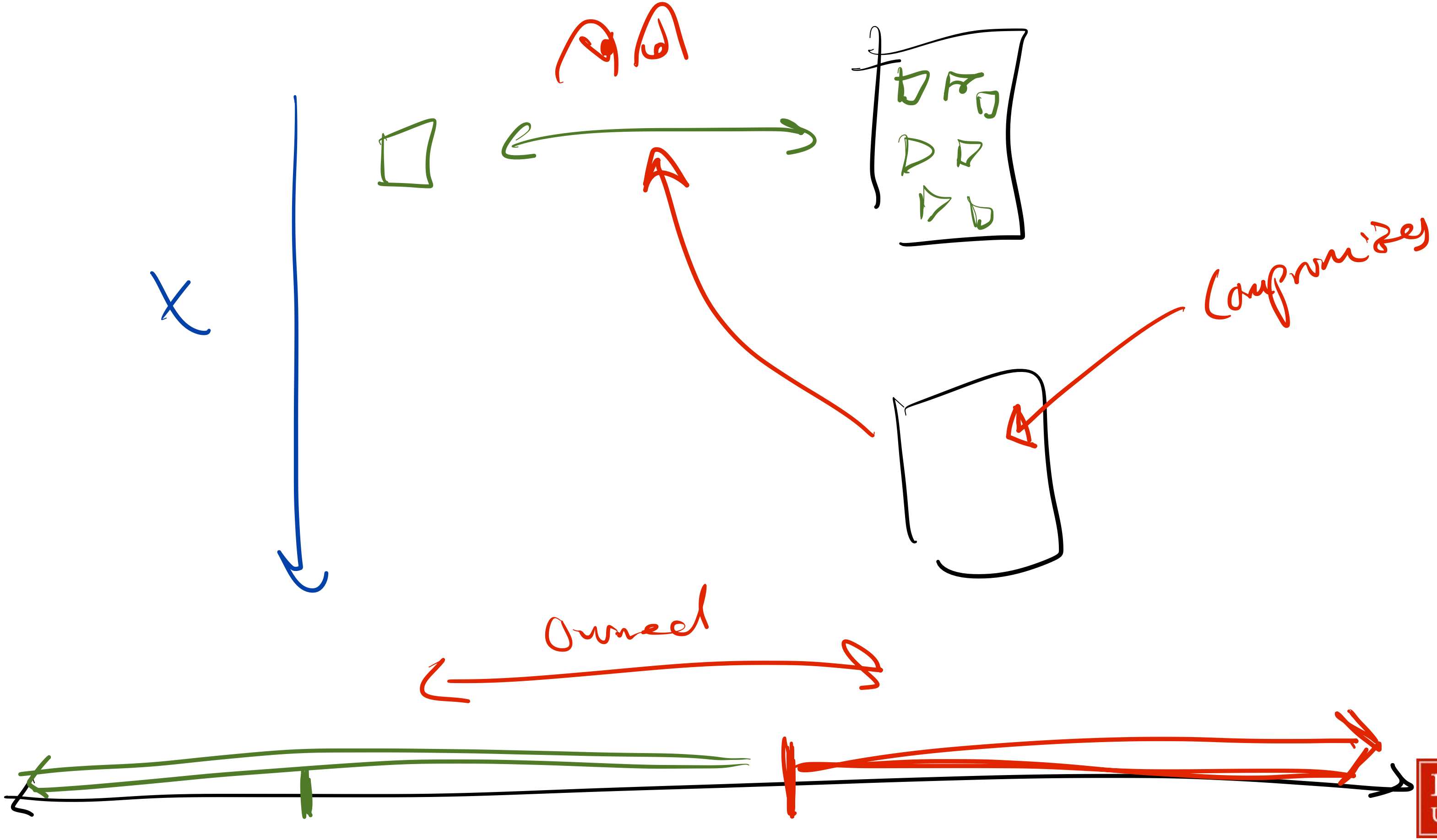
auth for client:

Secure Communication Goals

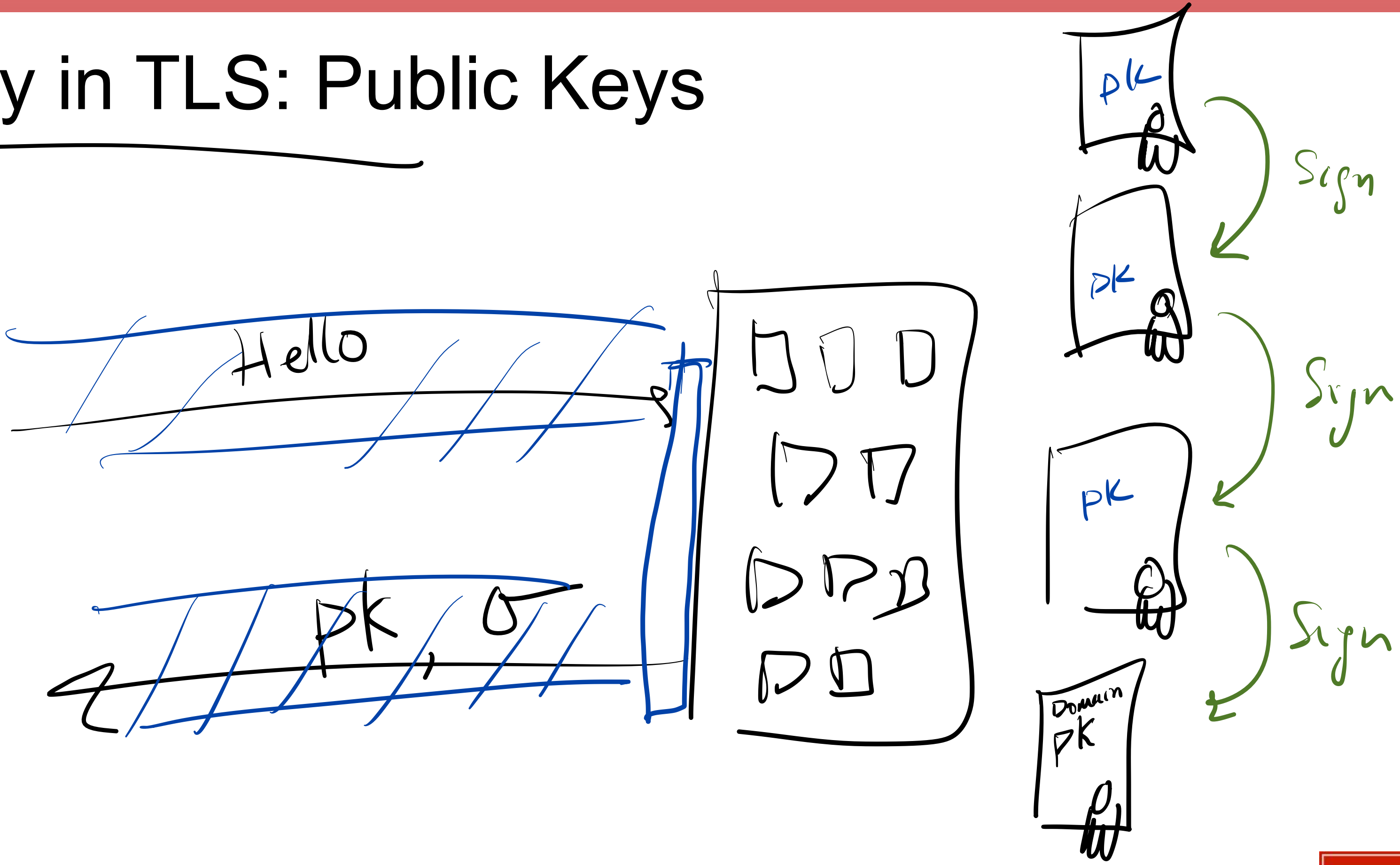
- Confidentiality, Authenticity, and Integrity on our messages



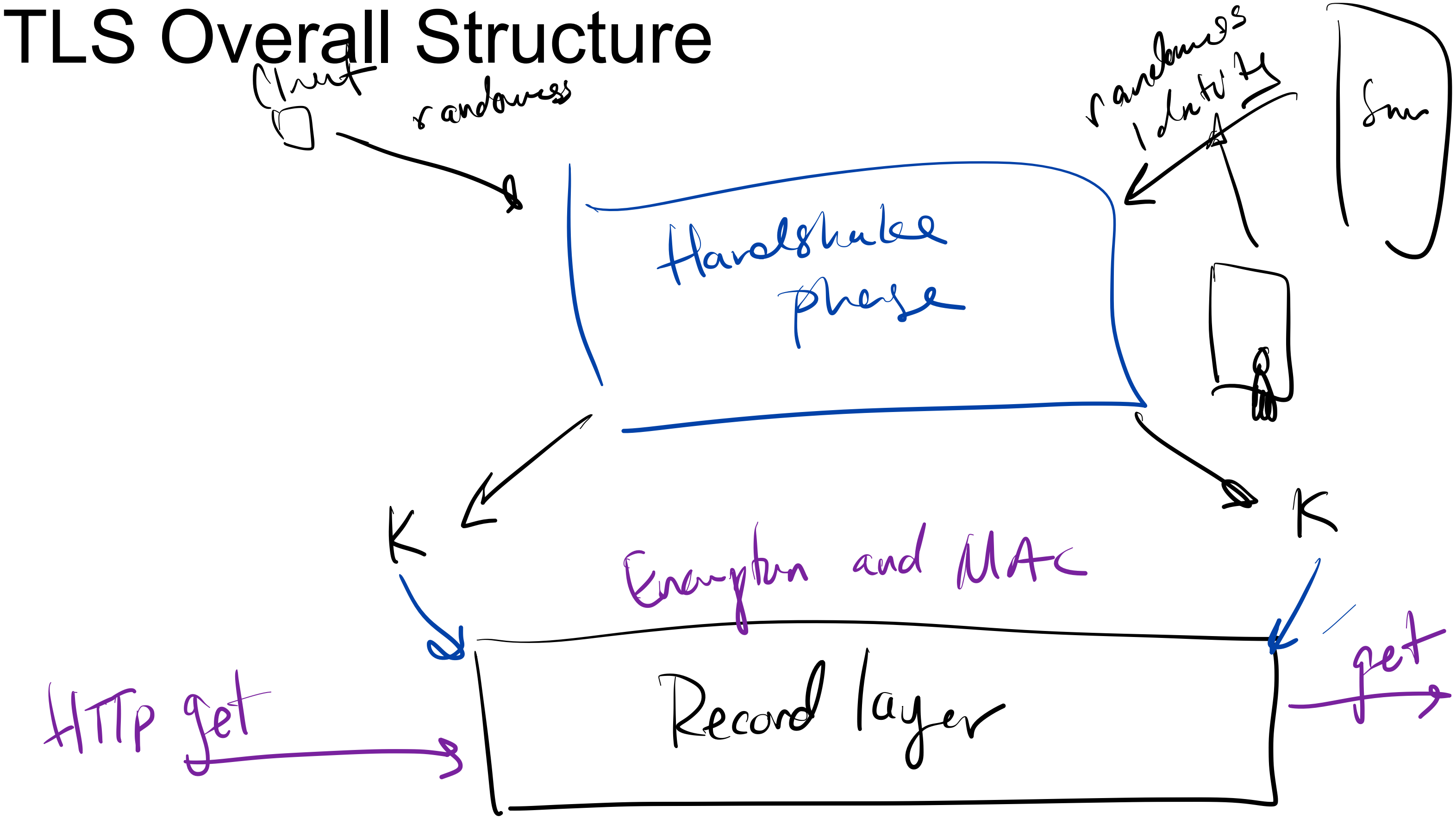
1. Replay Attacks
2. Perfect Forward Secrecy

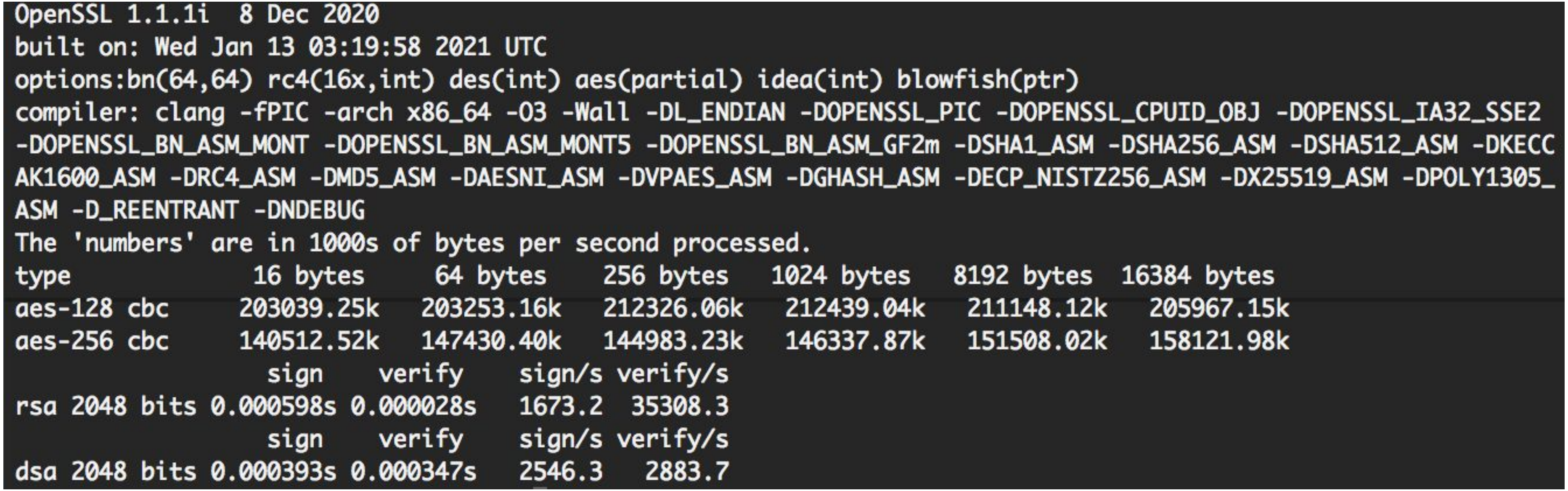

X
Compromizes
Owned

Identity in TLS: Public Keys



TLS Overall Structure





```
OpenSSL 1.1.1i  8 Dec 2020
built on: Wed Jan 13 03:19:58 2021 UTC
options:bn(64,64) rc4(16x,int) des(int) aes(partial) idea(int) blowfish(ptr)
compiler: clang -fPIC -arch x86_64 -O3 -Wall -DL_ENDIAN -DOPENSSL_PIC -DOPENSSL_CPUID_OBJ -DOPENSSL_IA32_SSE2
-DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DSHA512_ASM -DKECC
AK1600_ASM -DRC4_ASM -DMD5_ASM -DAESNI_ASM -DVPAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DX25519_ASM -DPOLY1305_
ASM -D_REENTRANT -DNDEBUG
The 'numbers' are in 1000s of bytes per second processed.
type             16 bytes     64 bytes    256 bytes   1024 bytes   8192 bytes  16384 bytes
aes-128 cbc     203039.25k   203253.16k   212326.06k   212439.04k   211148.12k   205967.15k
aes-256 cbc     140512.52k   147430.40k   144983.23k   146337.87k   151508.02k   158121.98k
                  sign    verify    sign/s verify/s
rsa 2048 bits 0.000598s 0.000028s   1673.2  35308.3
                  sign    verify    sign/s verify/s
dsa 2048 bits 0.000393s 0.000347s   2546.3   2883.7
```

~200x Faster

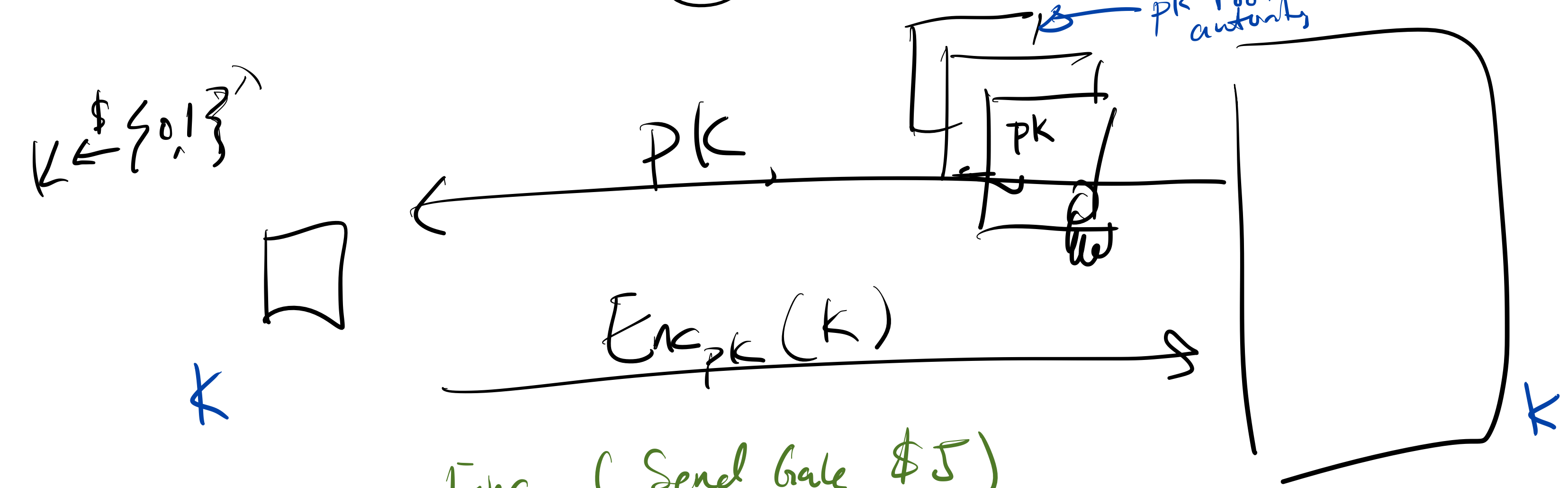
Public Key to Symmetric Key: Two Options*

1. Key Encapsulation

2. Key Exchange

Key Encapsulation

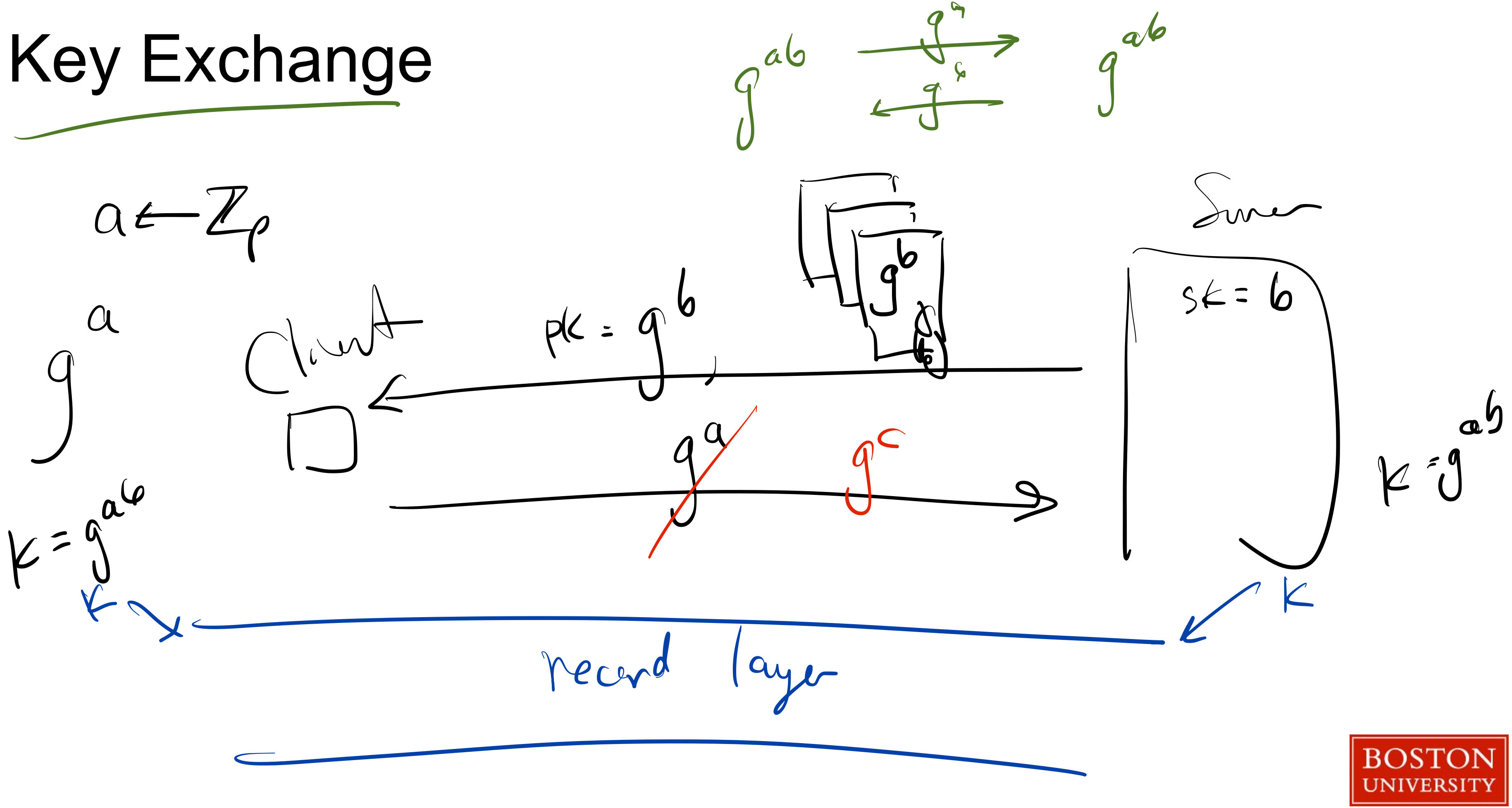
1. they should agree on K
2. no one else should learn K



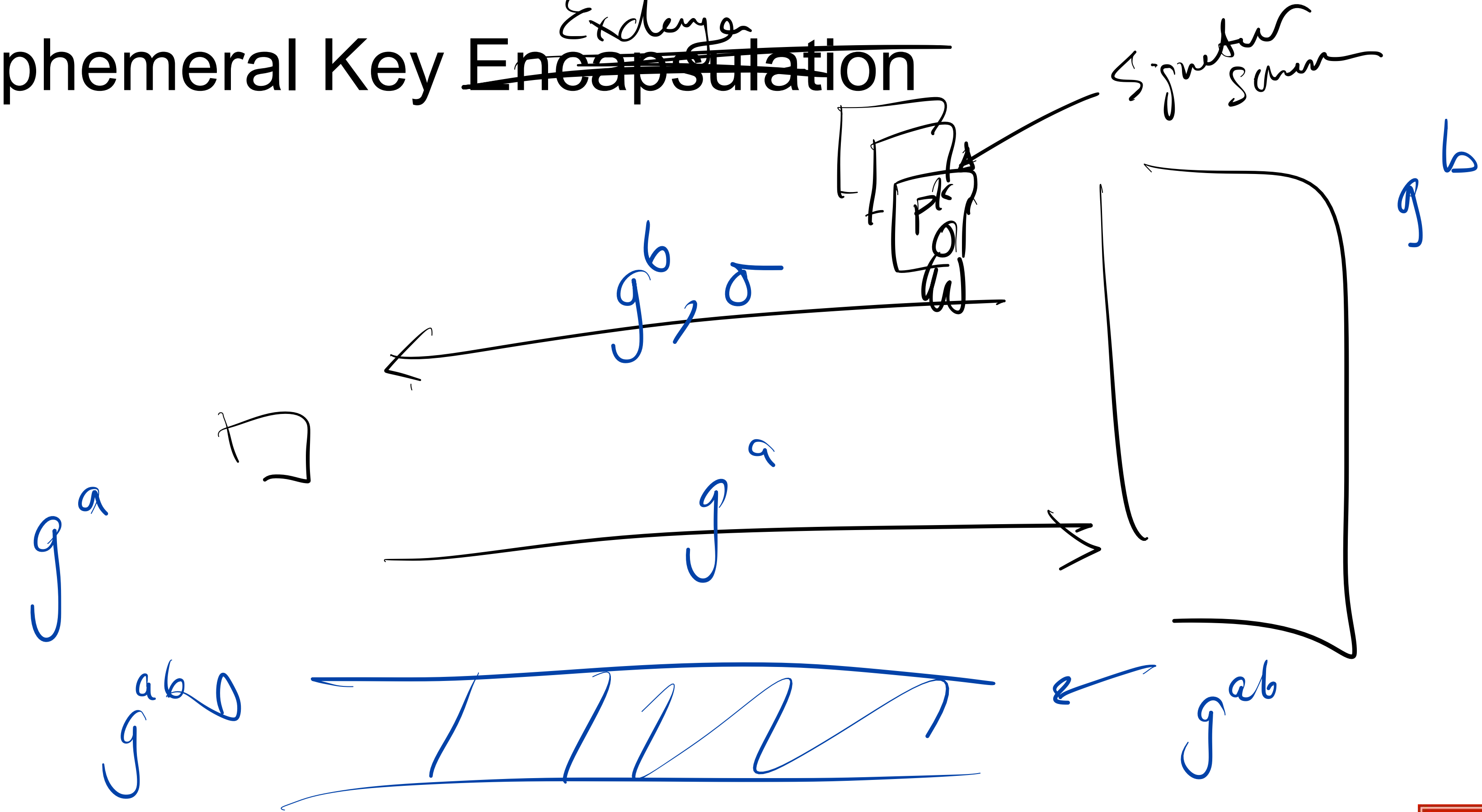
Enc_K(Send Gale $5)

Enc + MAC
under K

Key Exchange



Ephemeral Key ~~Encapsulation~~ Exchange


Signature Scheme
pk
g^b, σ
g^b
g^a
g^a
g^{ab}
g^{ab}