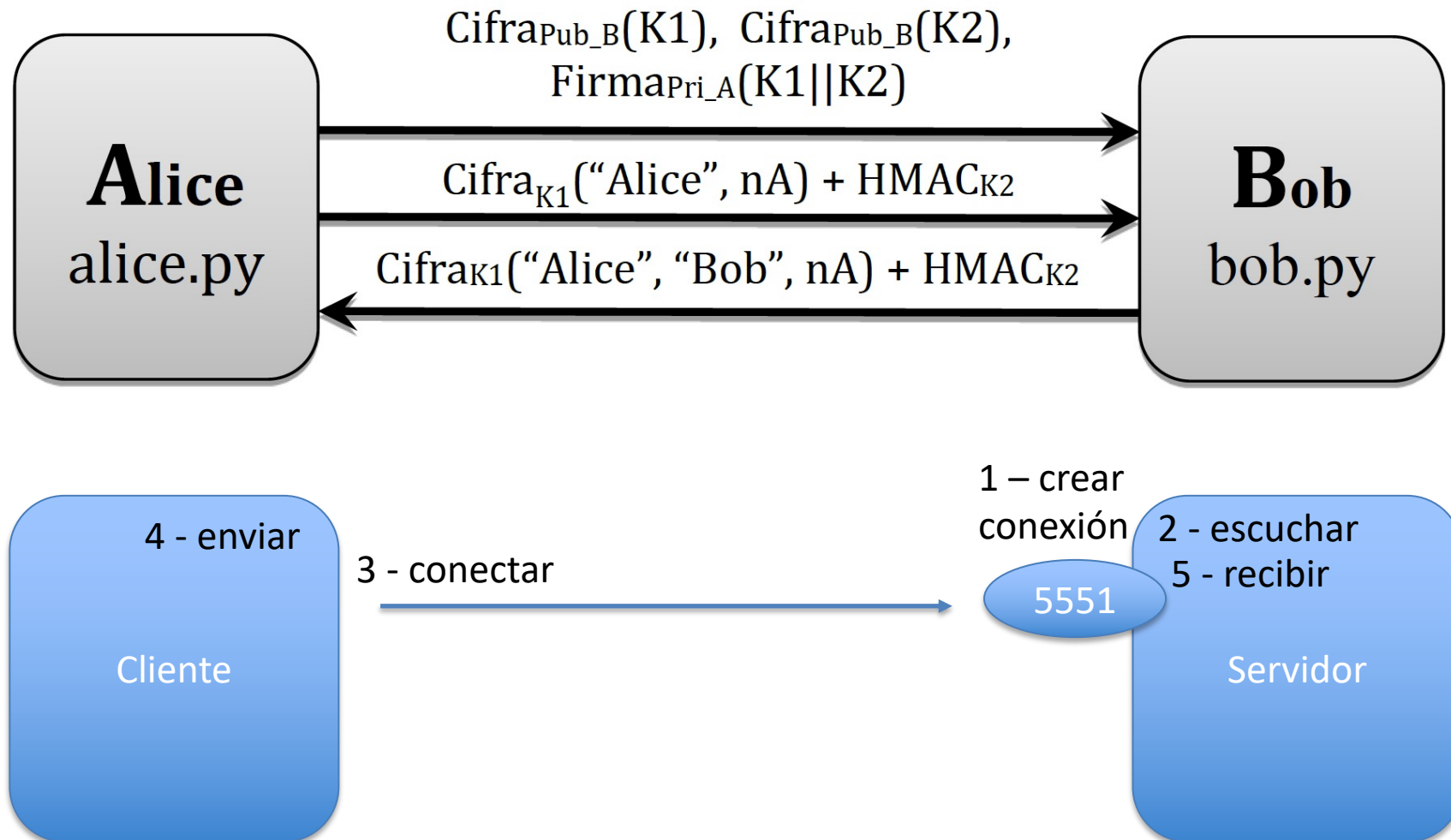


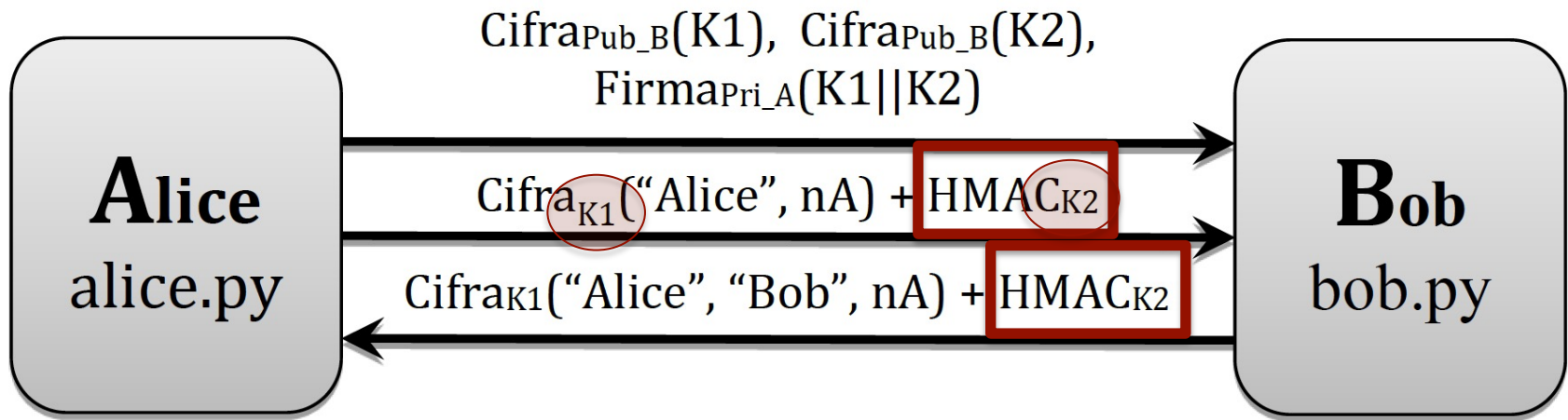
SEGURIDAD DE LA INFORMACIÓN

PROTOCOLOS (SOCKETs Y JSON)

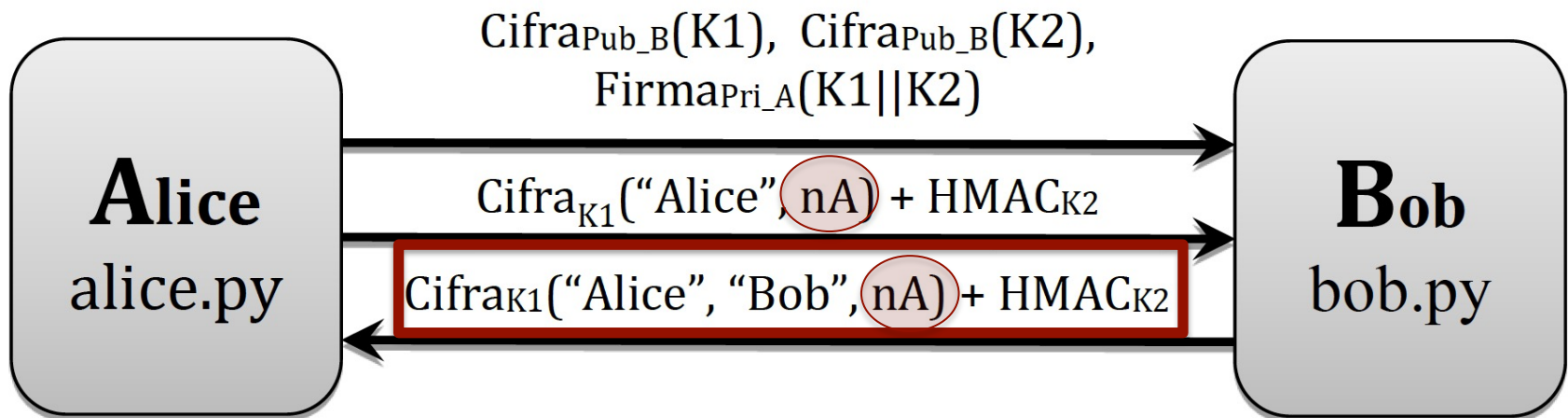
SOCKETS



MODO DE INTEGRIDAD Y AUTENTICACIÓN

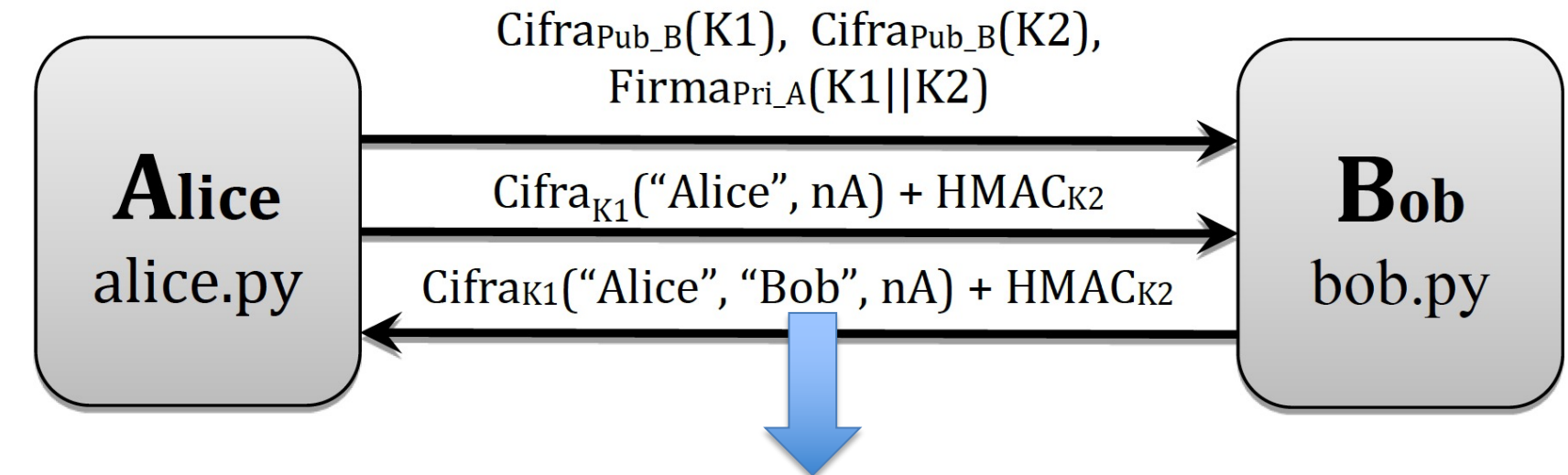


CONTRA ATAQUES DE REPLAY



CONCATENACIÓN –

¿ Cómo gestionar las concatenaciones ?



K1 = 0023232....1211

K2 = 1123 1233

Alice + Bob + nA

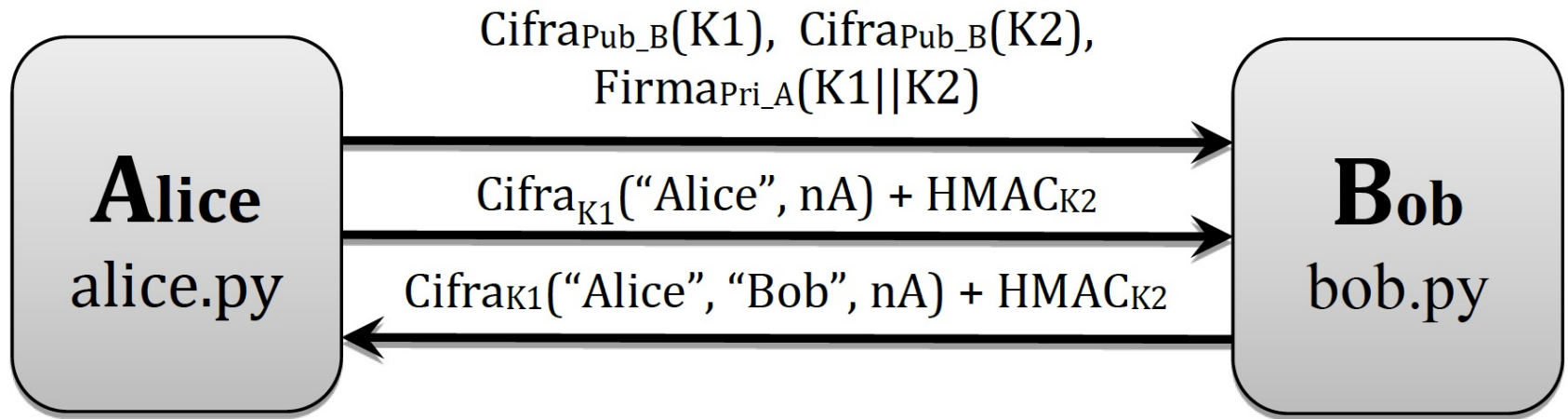
Concatenar es fácil, pero
¿¿qué pasa con la desconcatenación ??

CONCATENACIÓN – SOLUCIONES

Opción 1: JSON

Opción 2: Sin JSON y enviar cada información en mensajes distintos

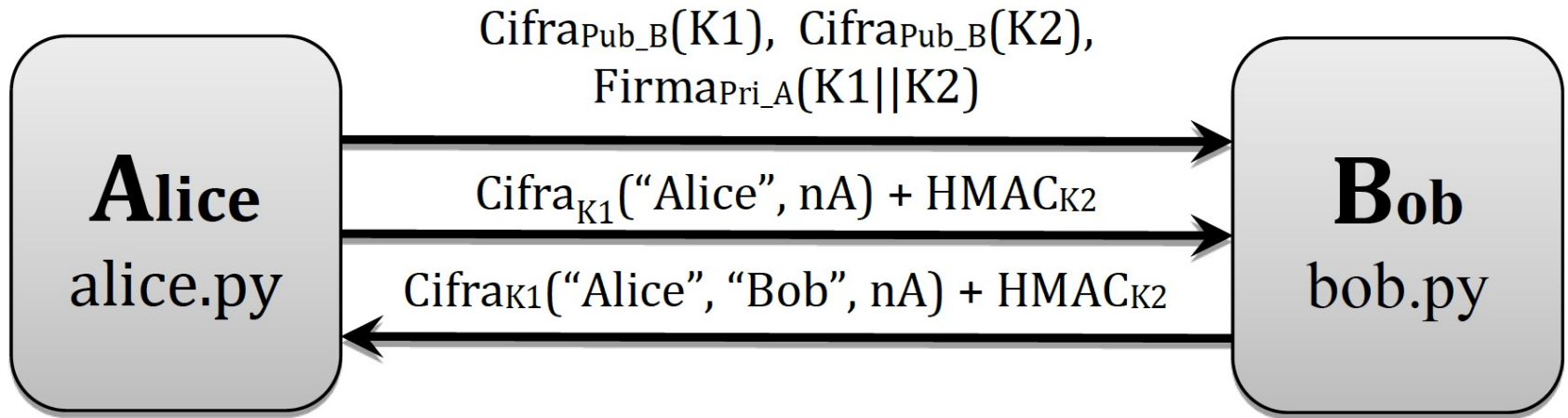
CONCATENACIÓN – OPCIÓN 1



JSON

JSON (JavaScript Object Notation) –

permite crear cadena de caracteres sujetas a un estándar en concreto, como una alternativa al formato XML =>> ["antonio","menganito","fulanito", "13"]

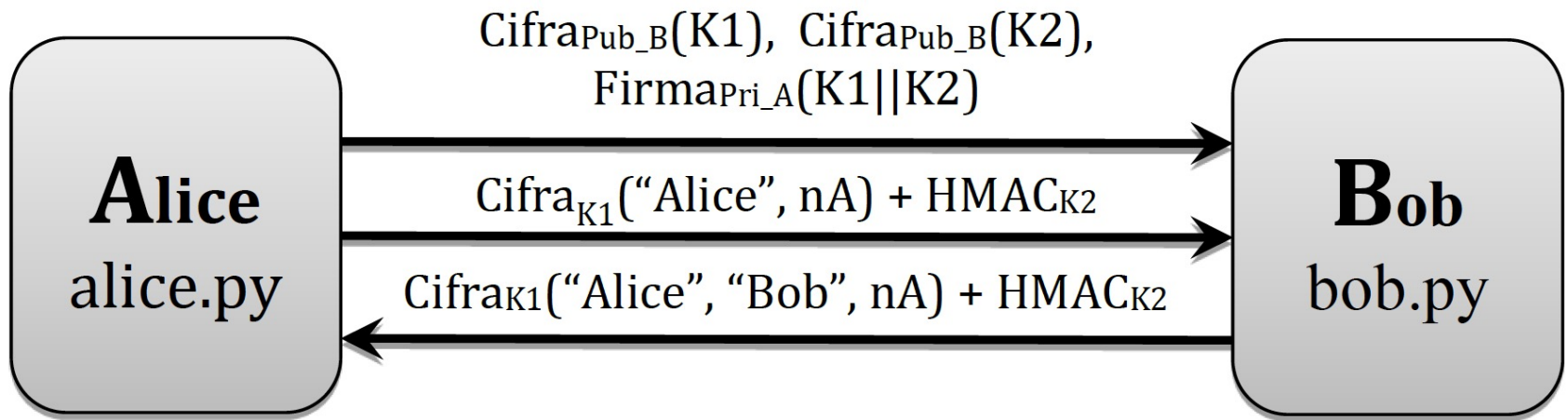


JSON se aplica para:

- poder enviar mensajes concatenados: A + B y desconcatenar de forma sencilla y rápida
 - =>> ["antonio","menganito","fulanito", "13"]
- Además, si no se aplica JSON entonces es fácil concatenar, pero no es nada fácil desconcatenar
 - antoniomenganitofulanito13 → ¿ qué separo ?

JSON (JavaScript Object Notation) –

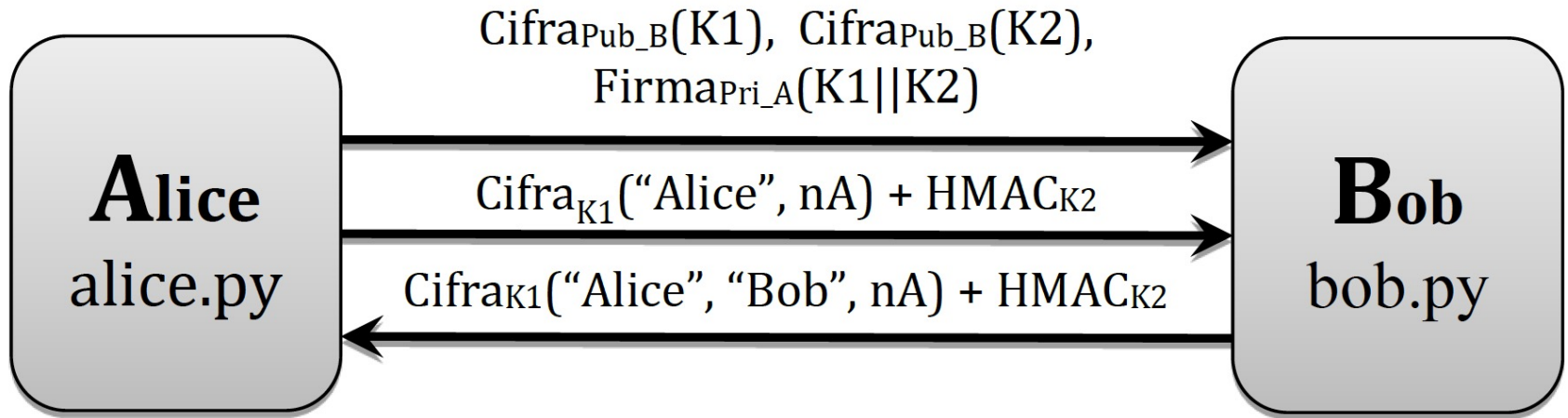
permite crear [cadenas de caracteres](#) sujetas a un estándar en concreto, como una alternativa al formato XML =>> ["antonio","menganito","fulanito", "13"]



```
mensaje = [] # Array vacio
mensaje.append(alice) # Donde alice == "Alice"
mensaje.append(nonce.hex()) # Conversion de Bytes a Hexadecimal
jStr = json.dumps(mensaje) # Convertimos un Array Python a string
```


JSON (JavaScript Object Notation) –

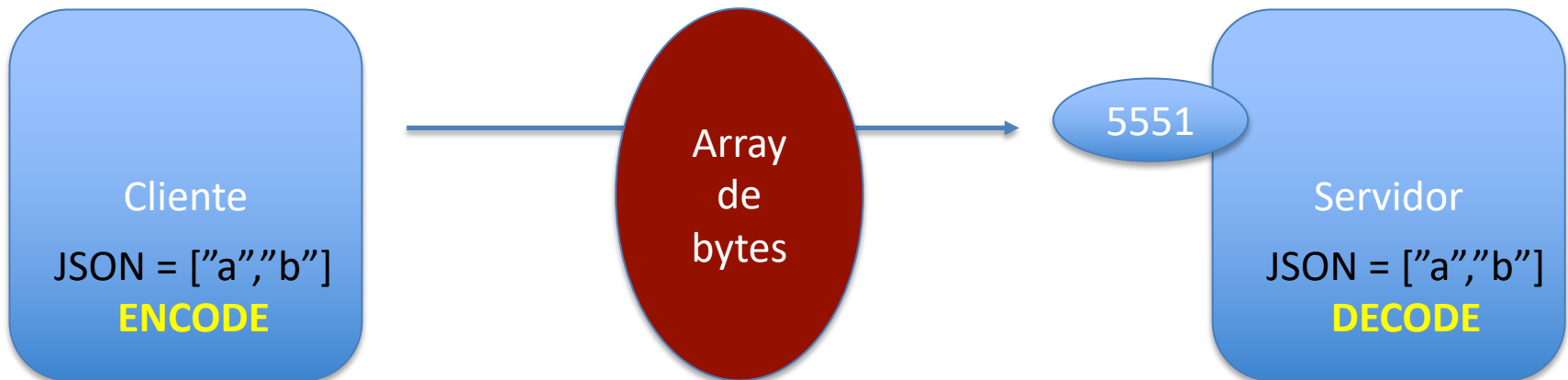
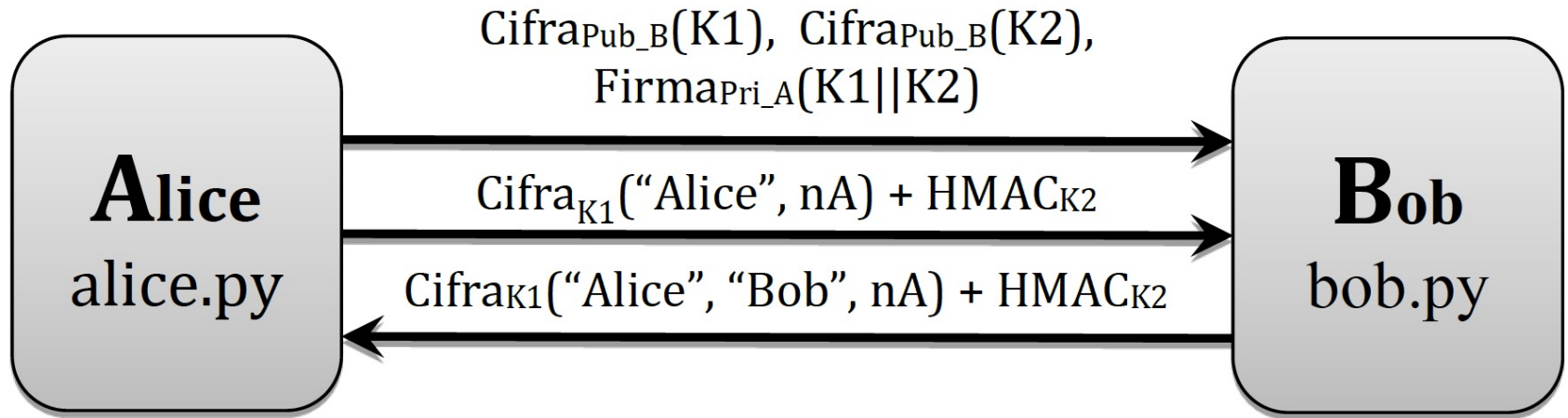
permite crear cadena de caracteres sujetas a un estándar en concreto, como una alternativa al formato XML =>> ["antonio","menganito","fulanito", "13"]



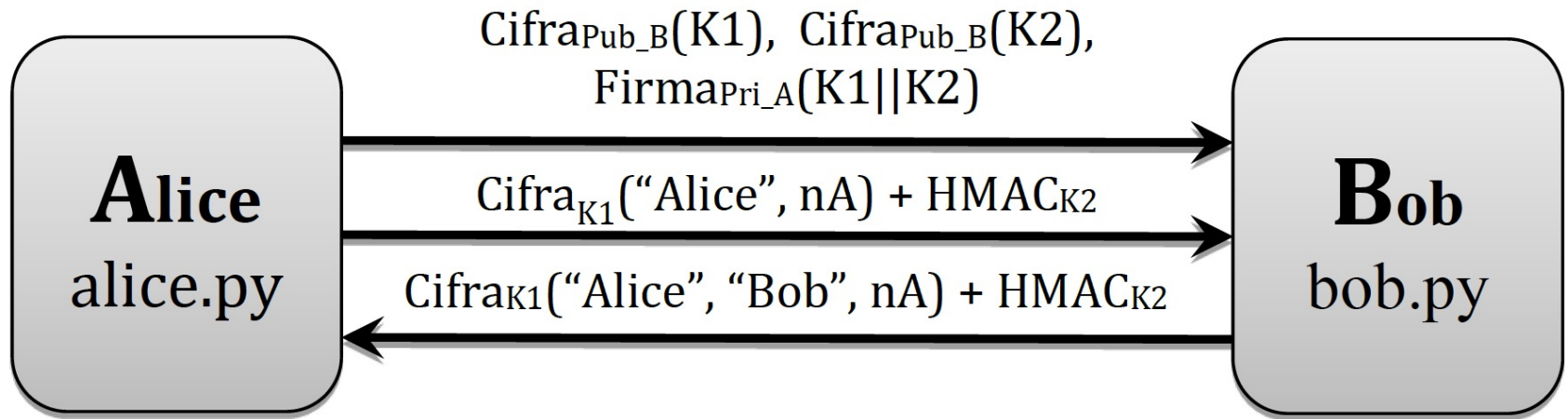
```
mensaje = json.loads(jStr) # Recuperamos un Array Python de un string
alice, nonce_cadenaHEX = mensaje
nonce = bytearray.fromhex(nonce_cadenaHEX) # De Hexadecimal a Bytes
```

JSON (JavaScript Object Notation) –

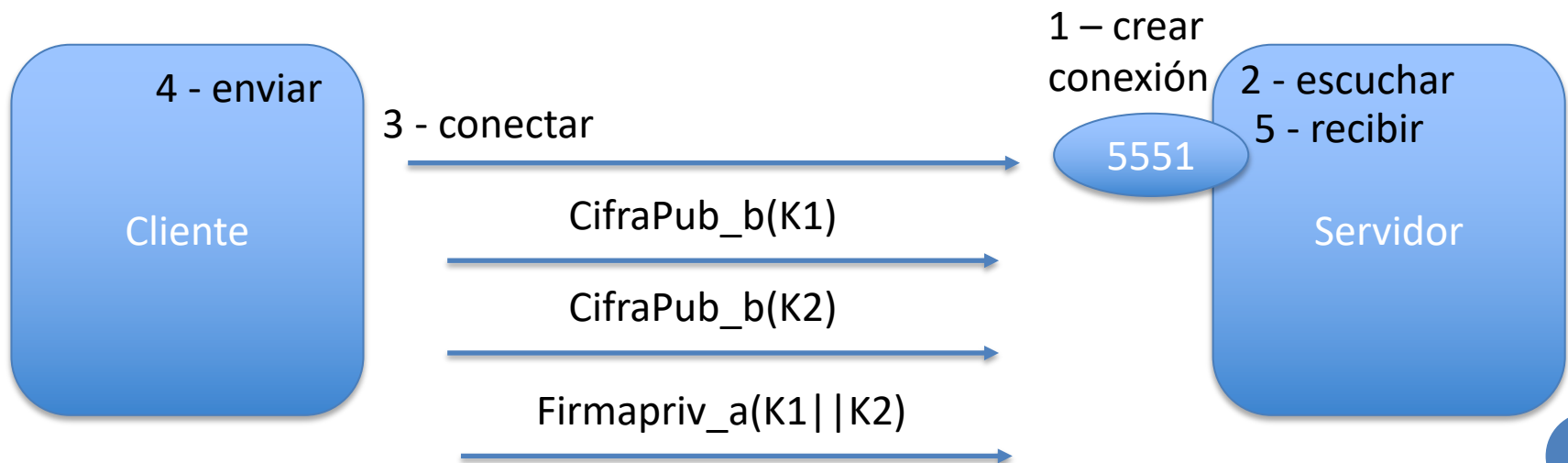
permite crear cadenas de caracteres sujetas a un estándar en concreto, como una alternativa al formato XML =>> ["antonio","menganito","fulanito", "13"]



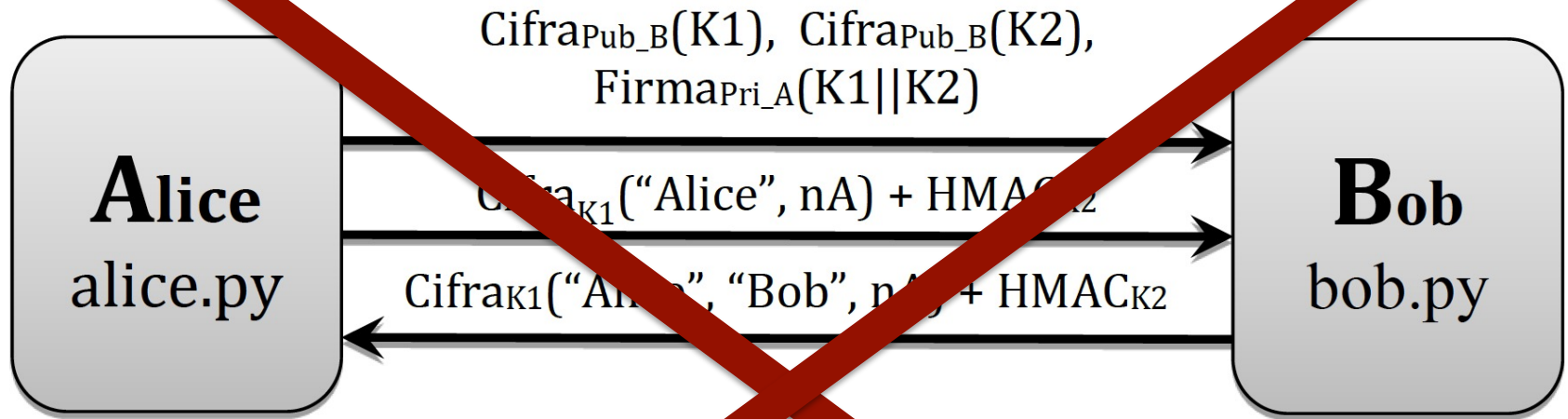
CONCATENACIÓN – OPCIÓN 2



Enviar de forma bruta secuencias de información



CONCATENACIÓN – OPCIÓN 2



Enviar de forma bruta secuencial de información

