Badge of Shame

Breaking into Secure Facilities with OSDP

BISHOPFOX

BlackHat USA 2023

TOP SECRET

Secure Facility

THE GOODS

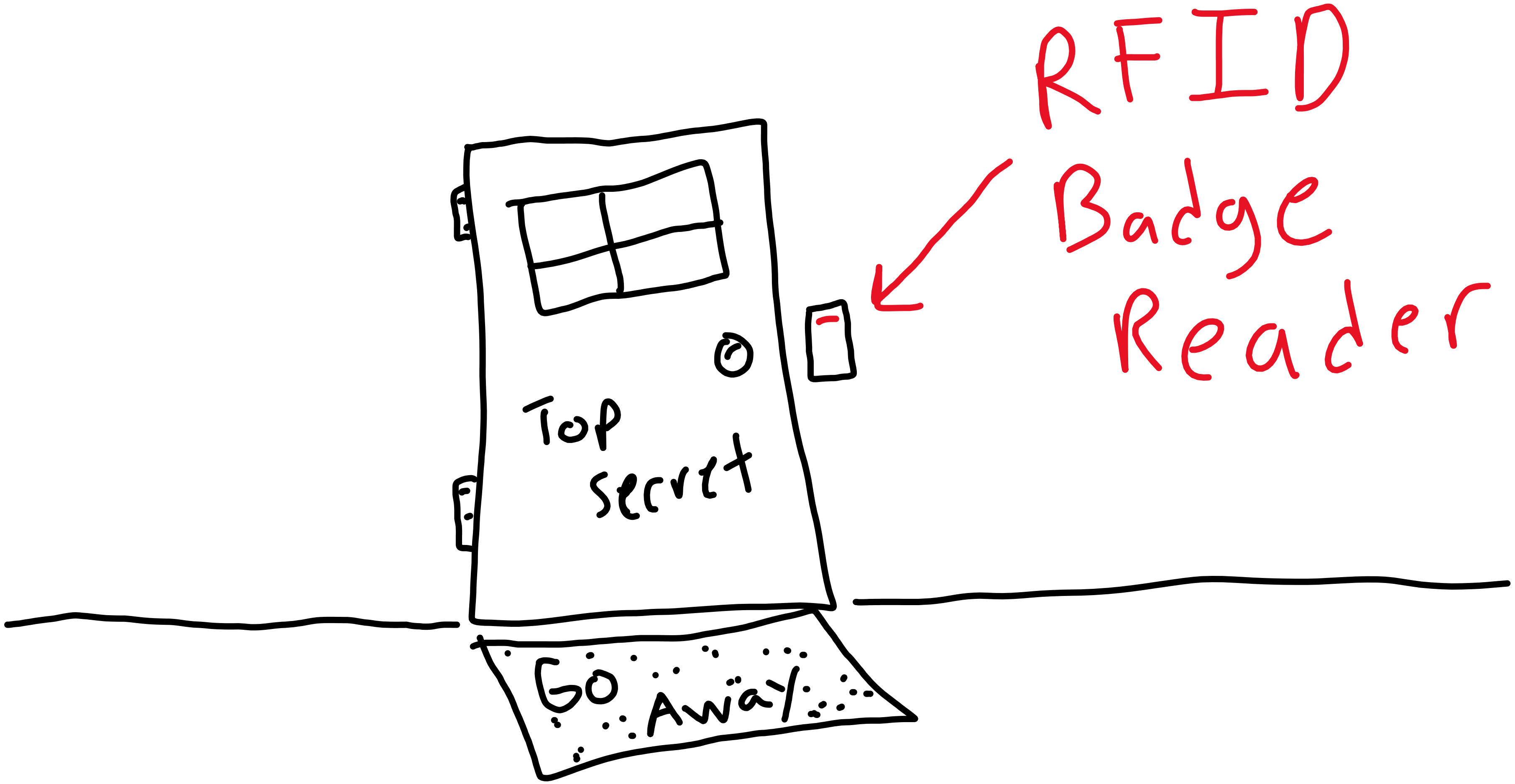
Top Secret Materials

TOP SECRET

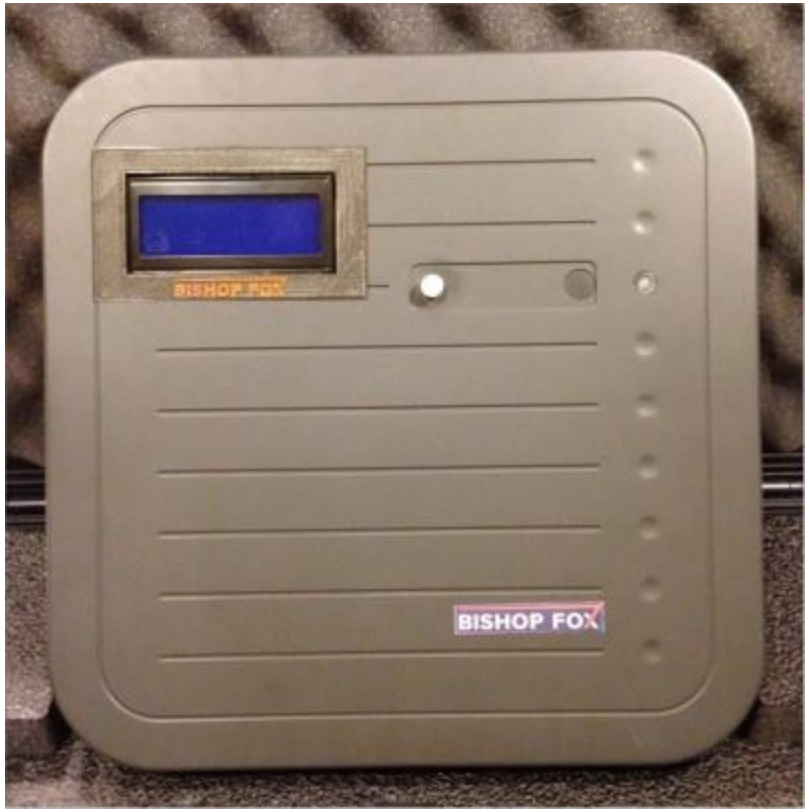
OSDP

TOP SECRET

OSDP



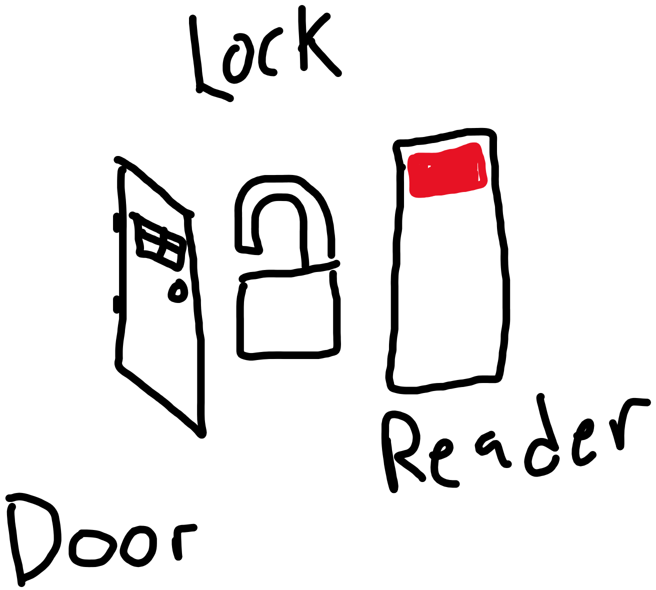
Other Ways to Hack RFID



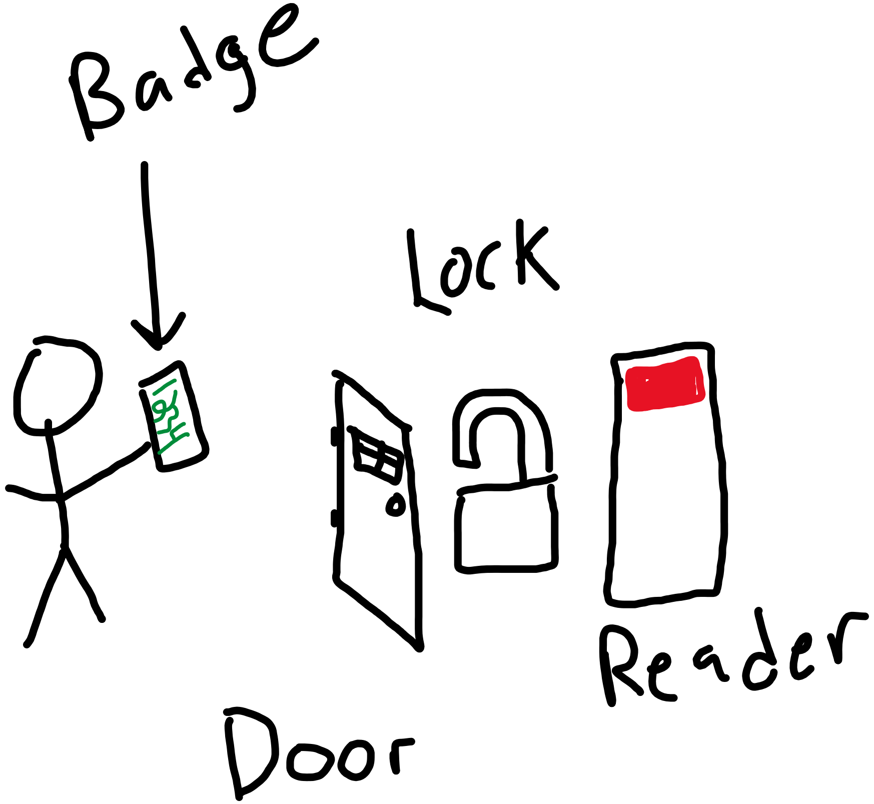
Tastic RFID Thief

I made this!

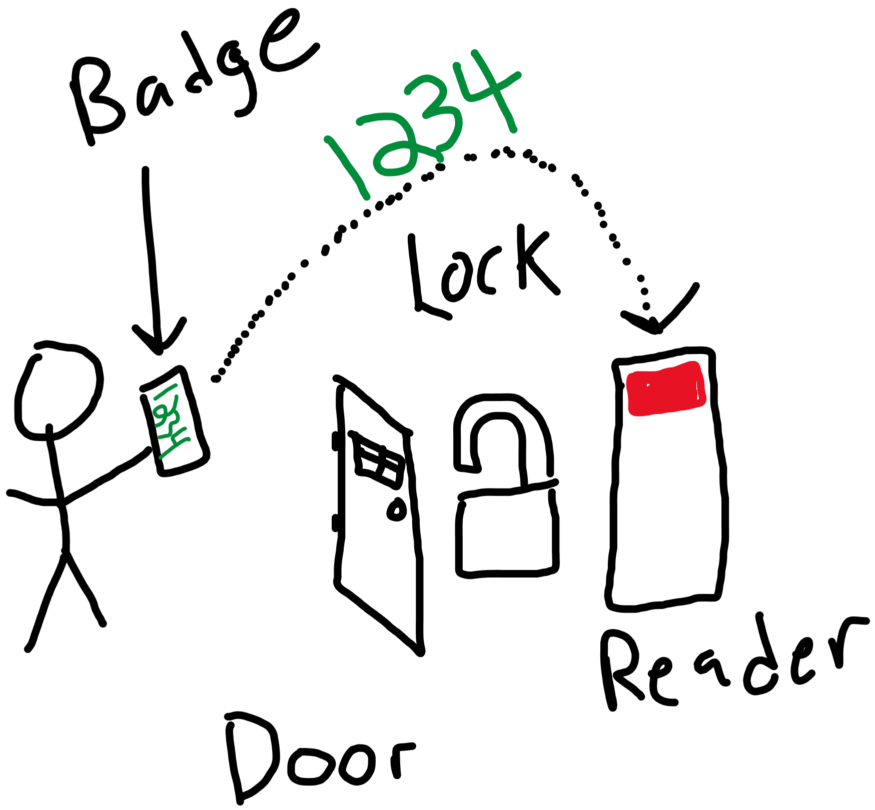
RFID Badge Setup



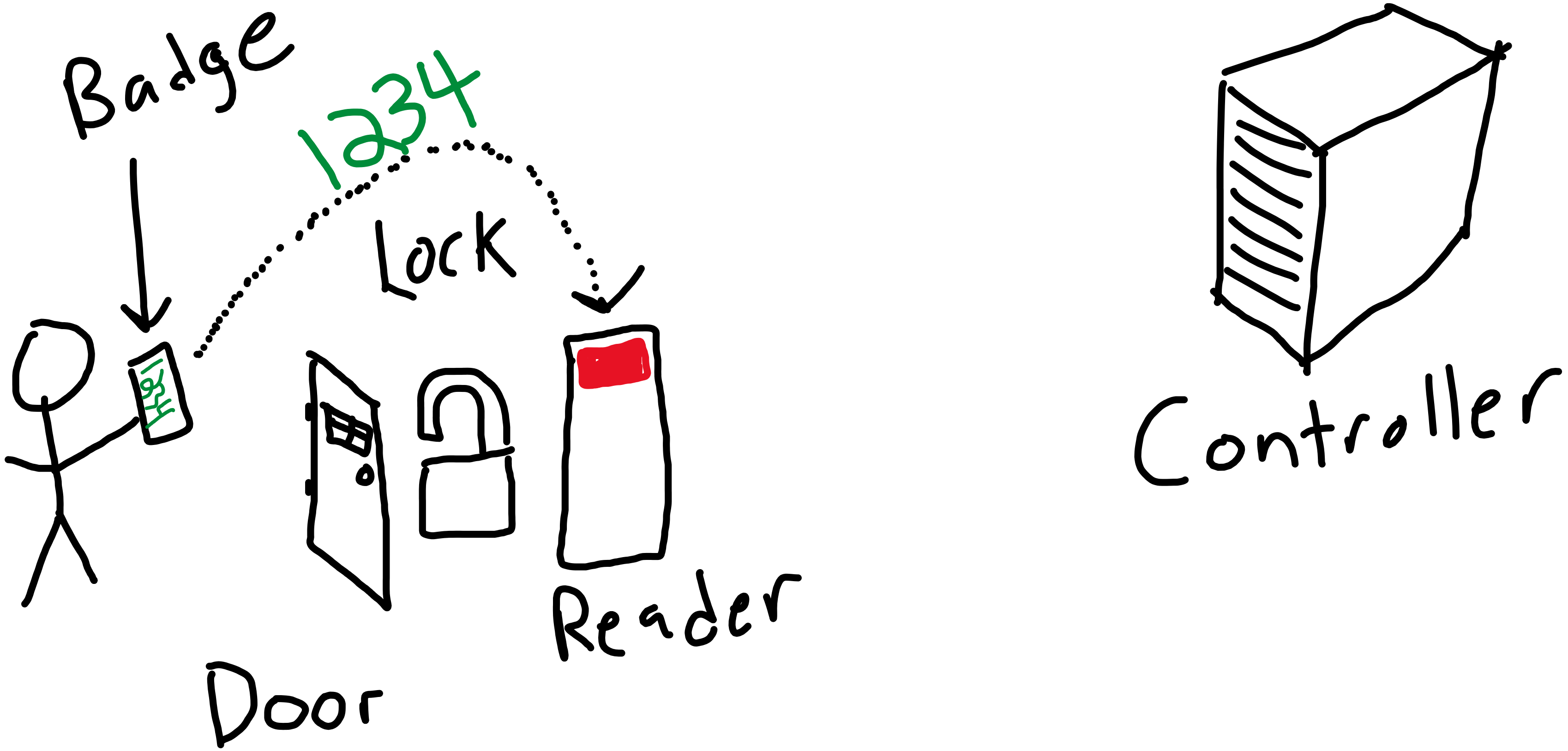
RFID Badge Setup



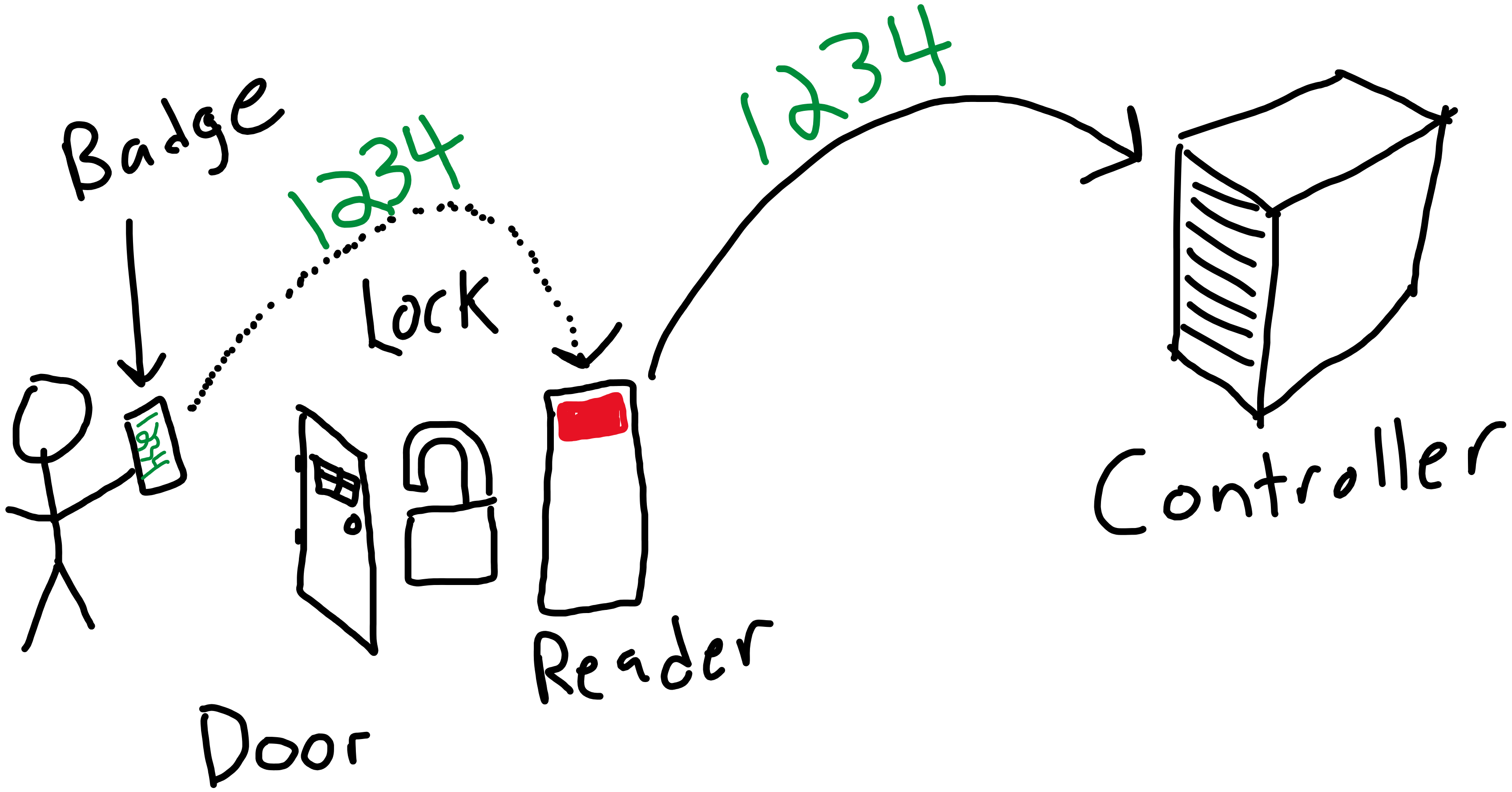
RFID Badge Setup



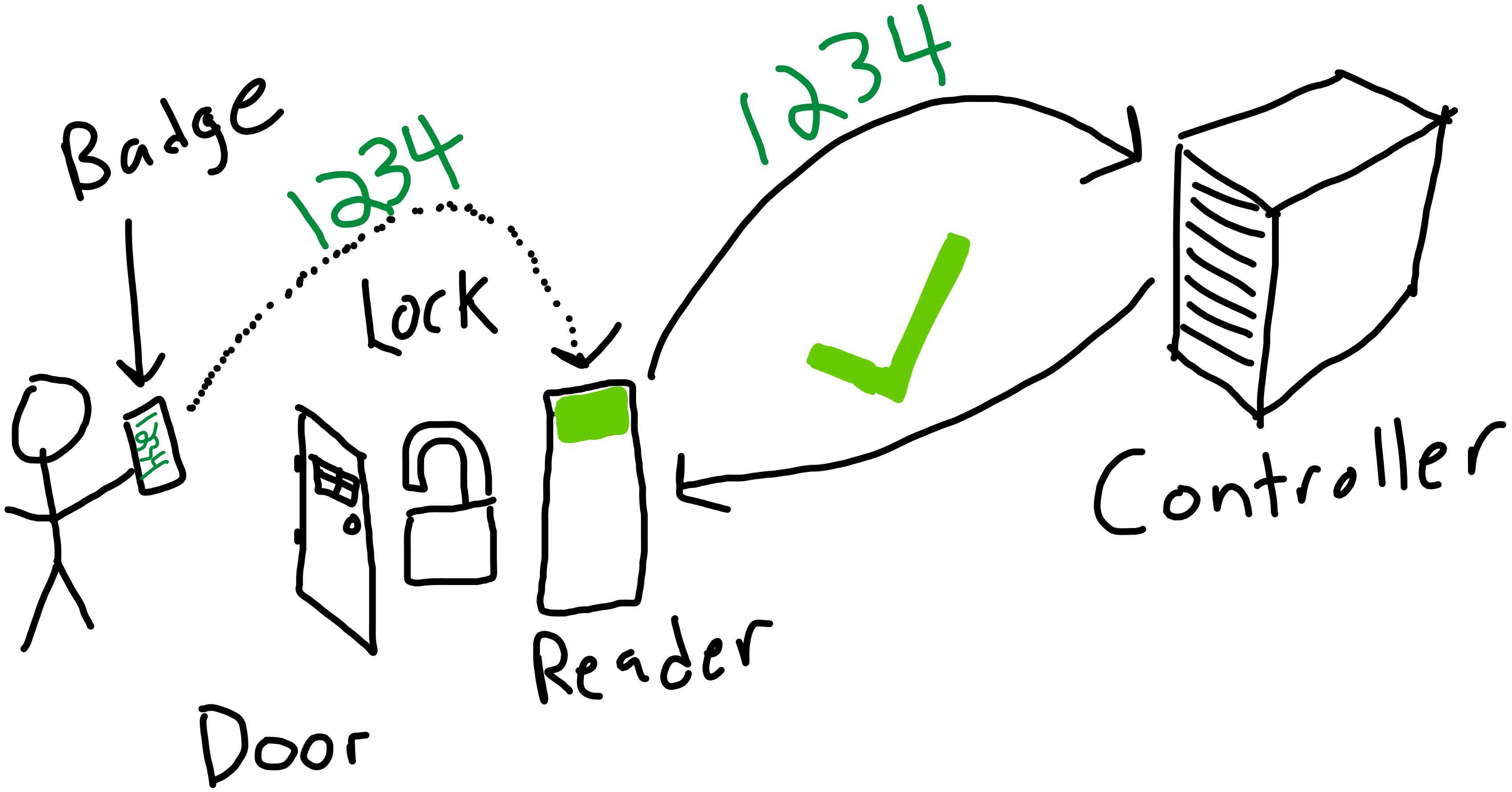
RFID Badge Setup



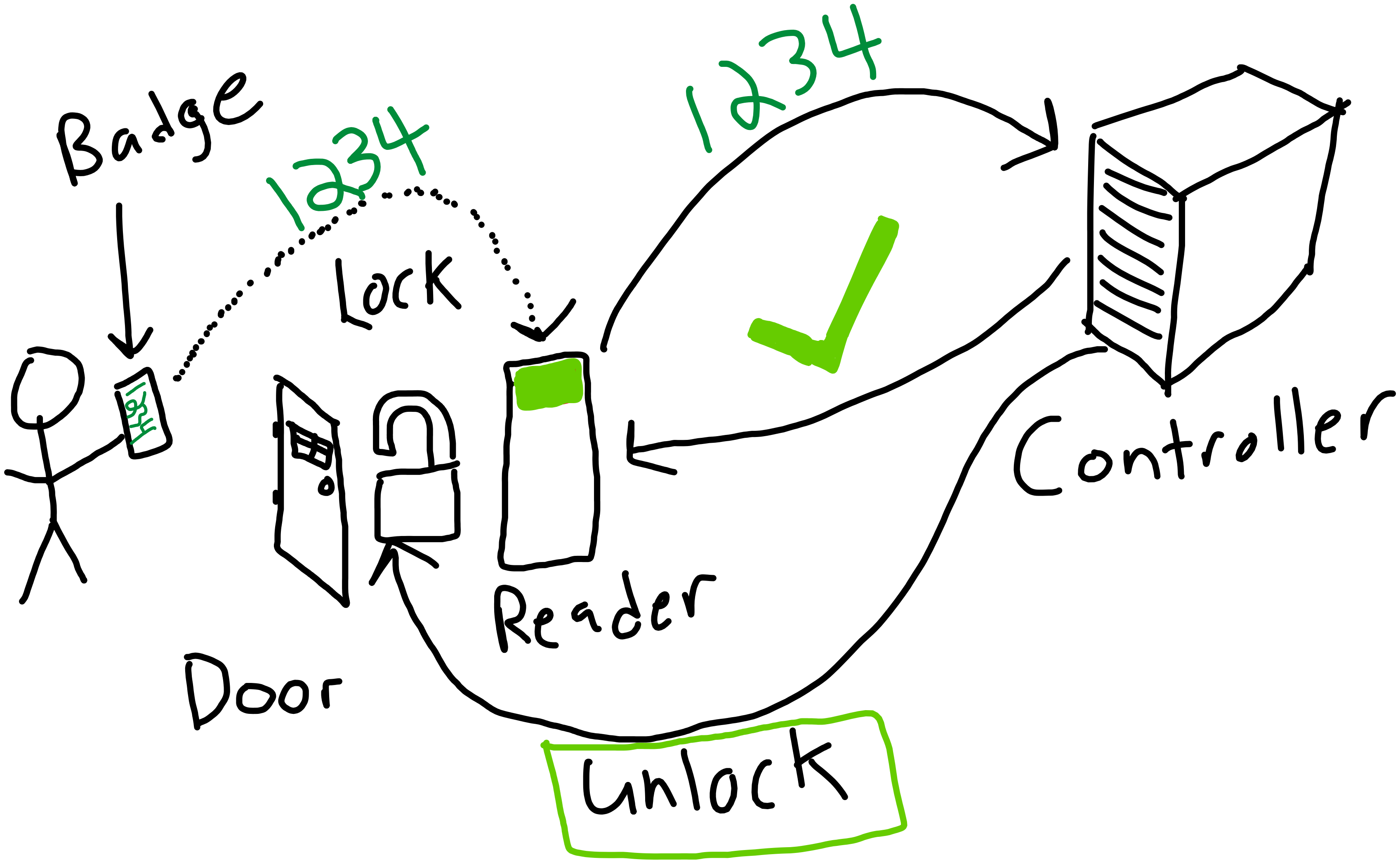
RFID Badge Setup



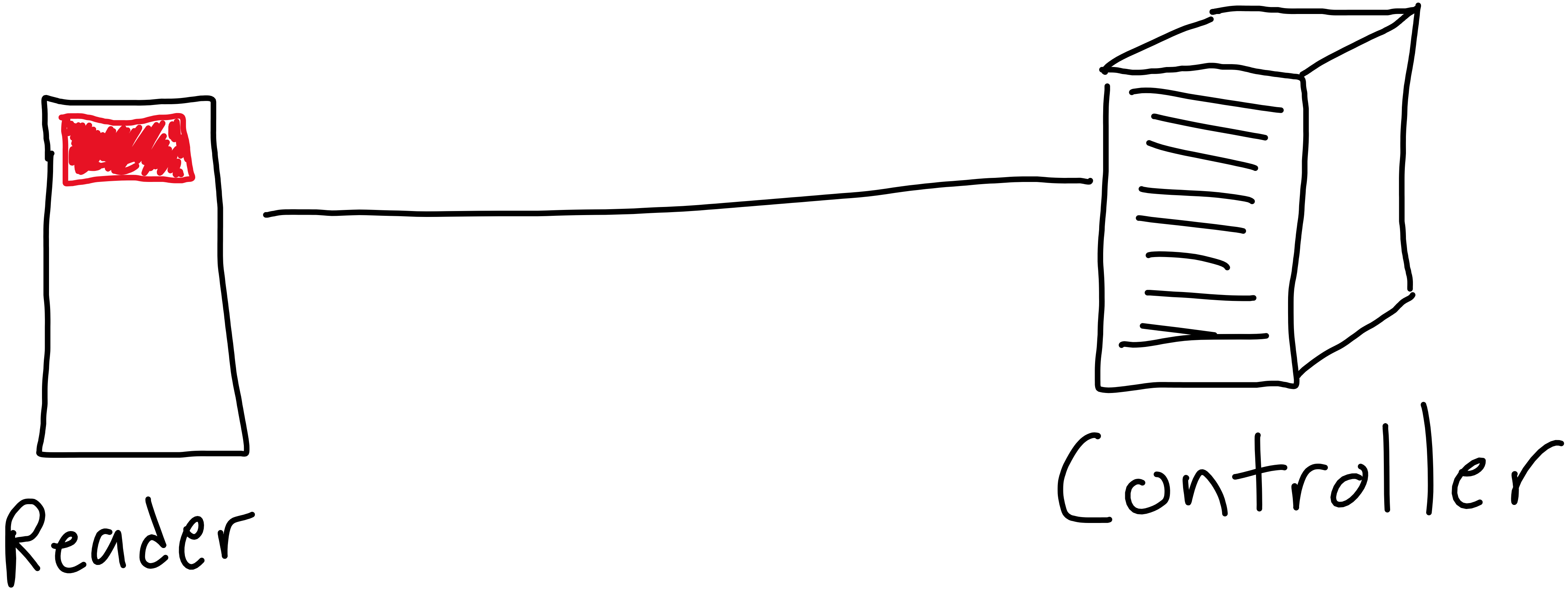
RFID Badge Setup



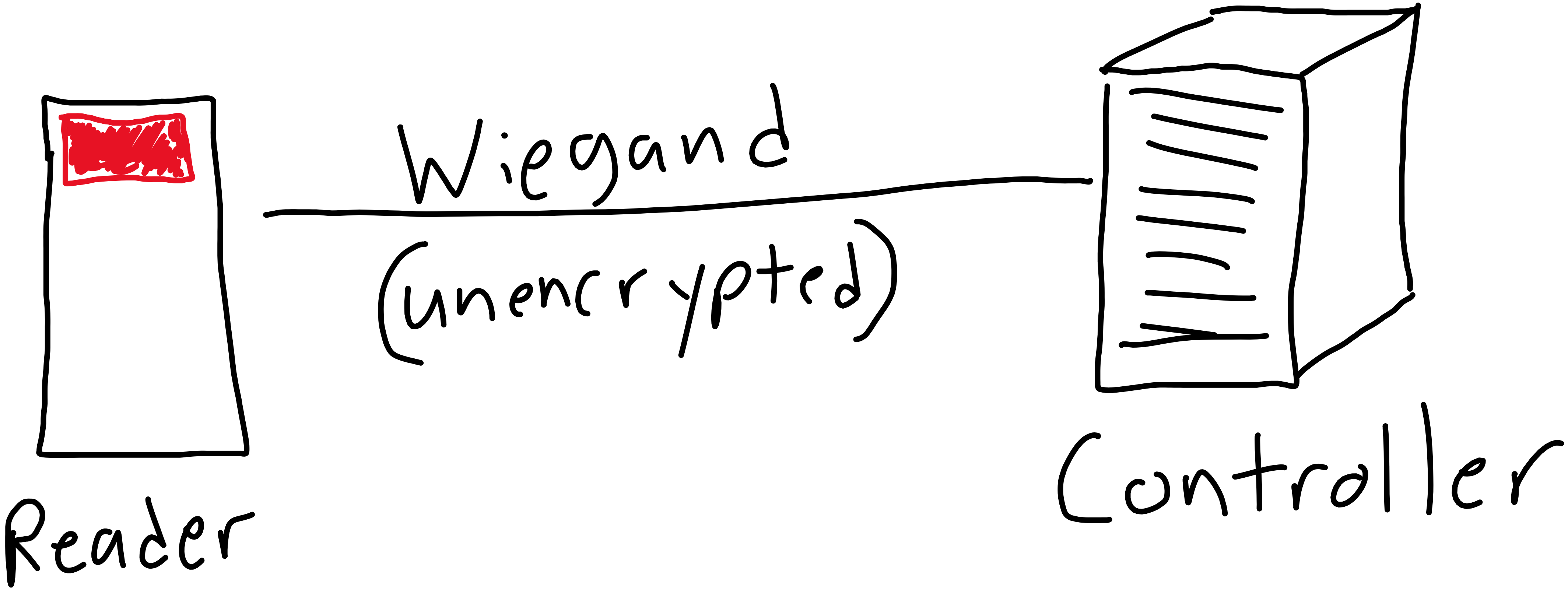
RFID Badge Setup



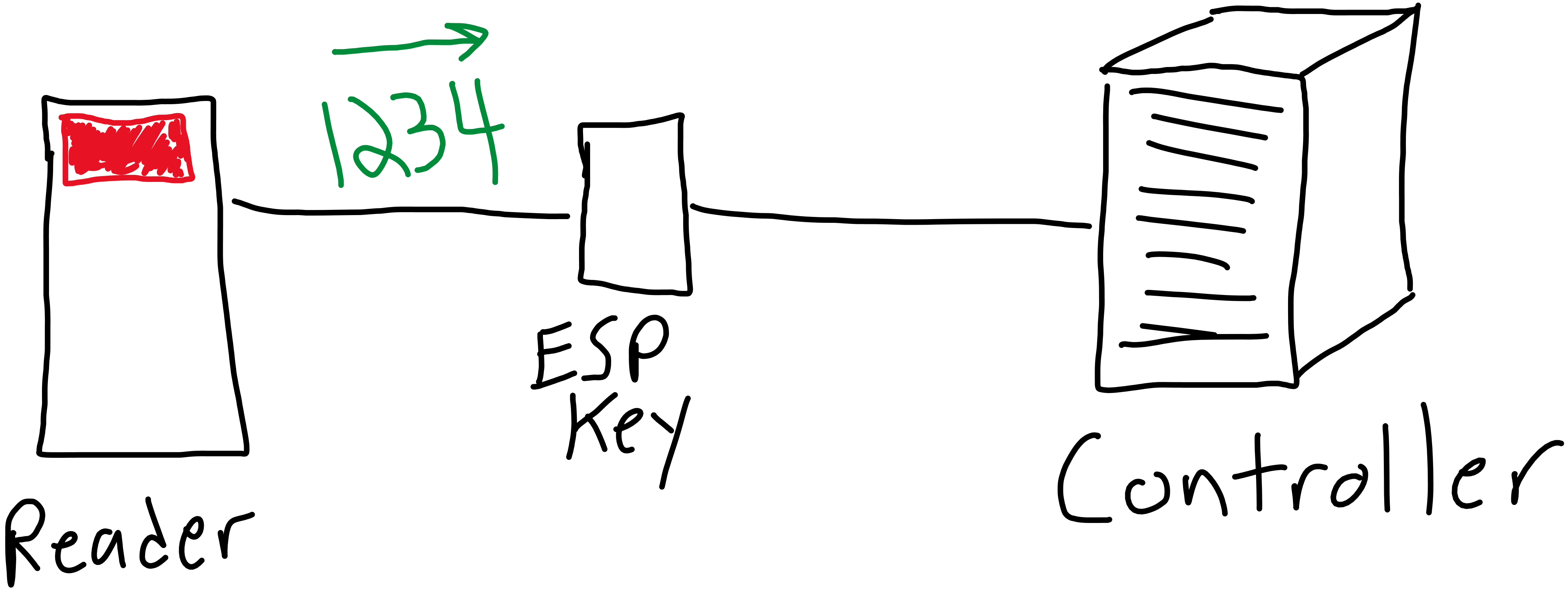
RFID Badge Setup



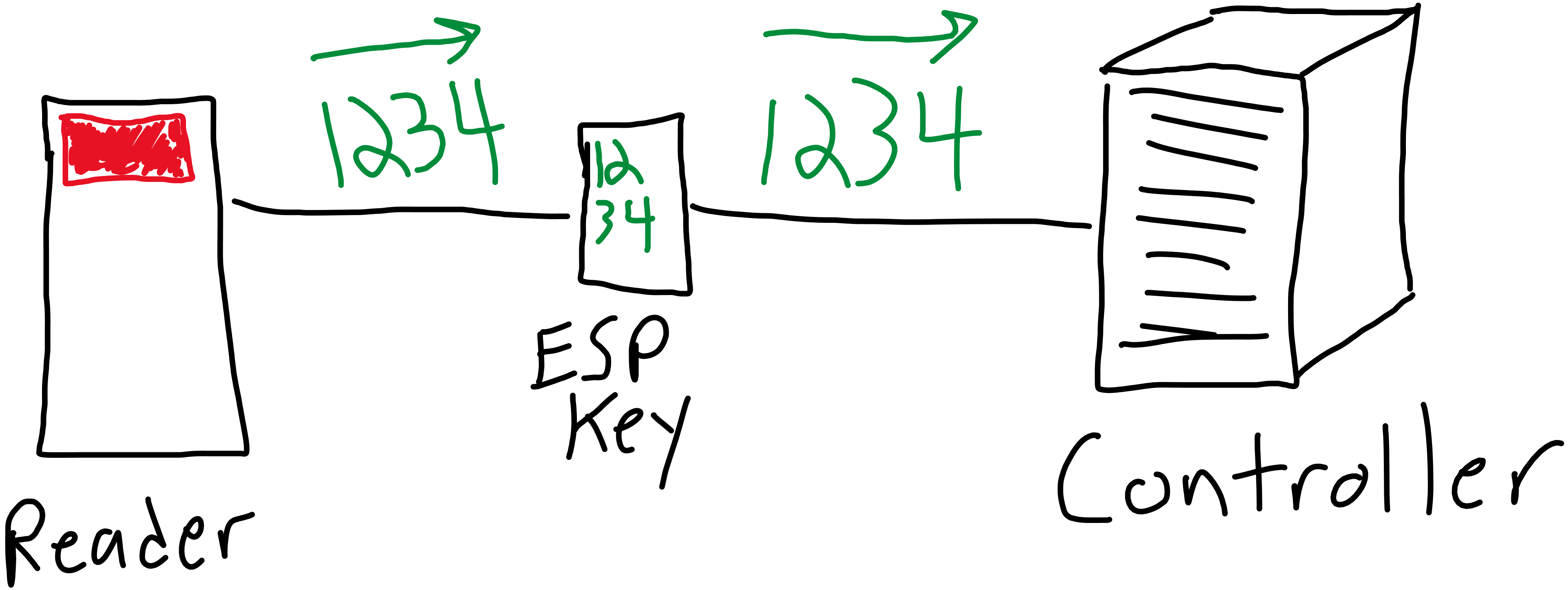
RFID Badge Setup



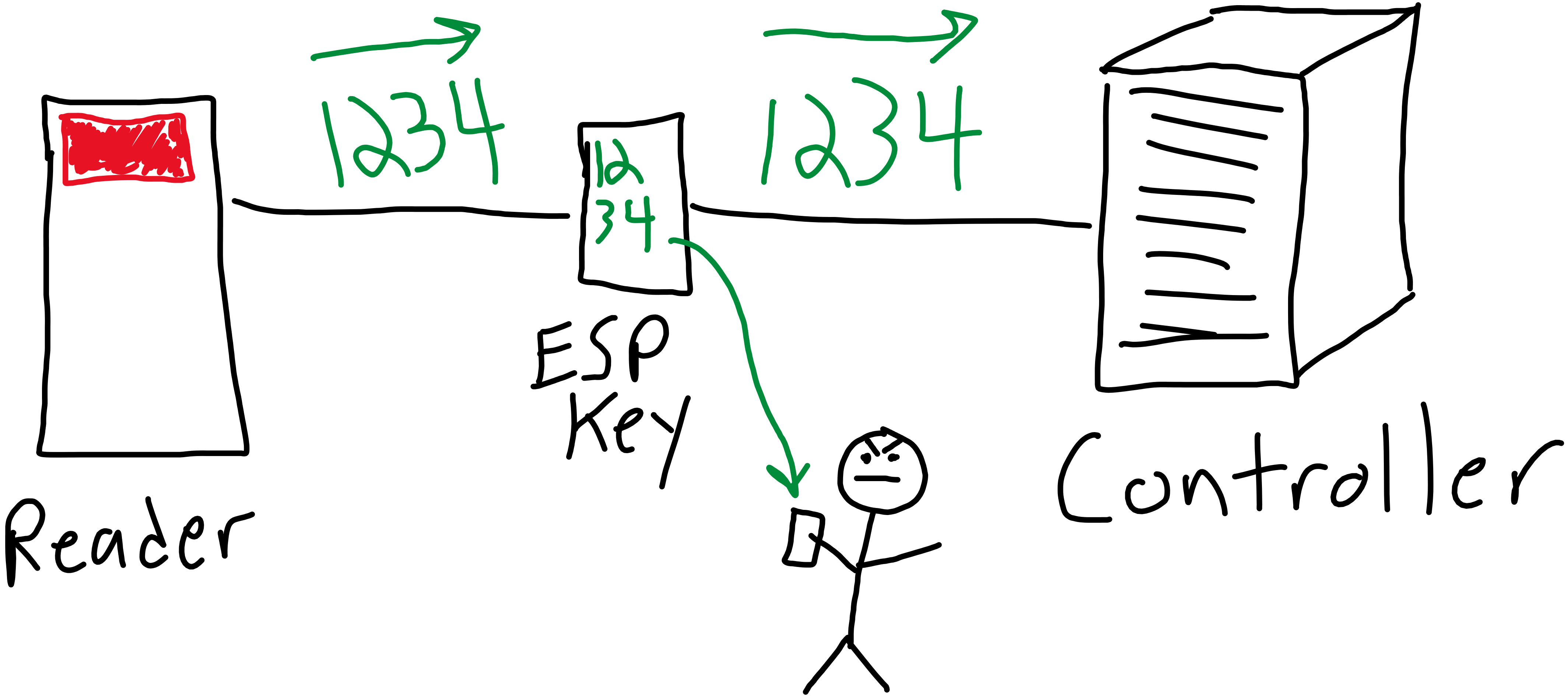
RFID Badge Setup



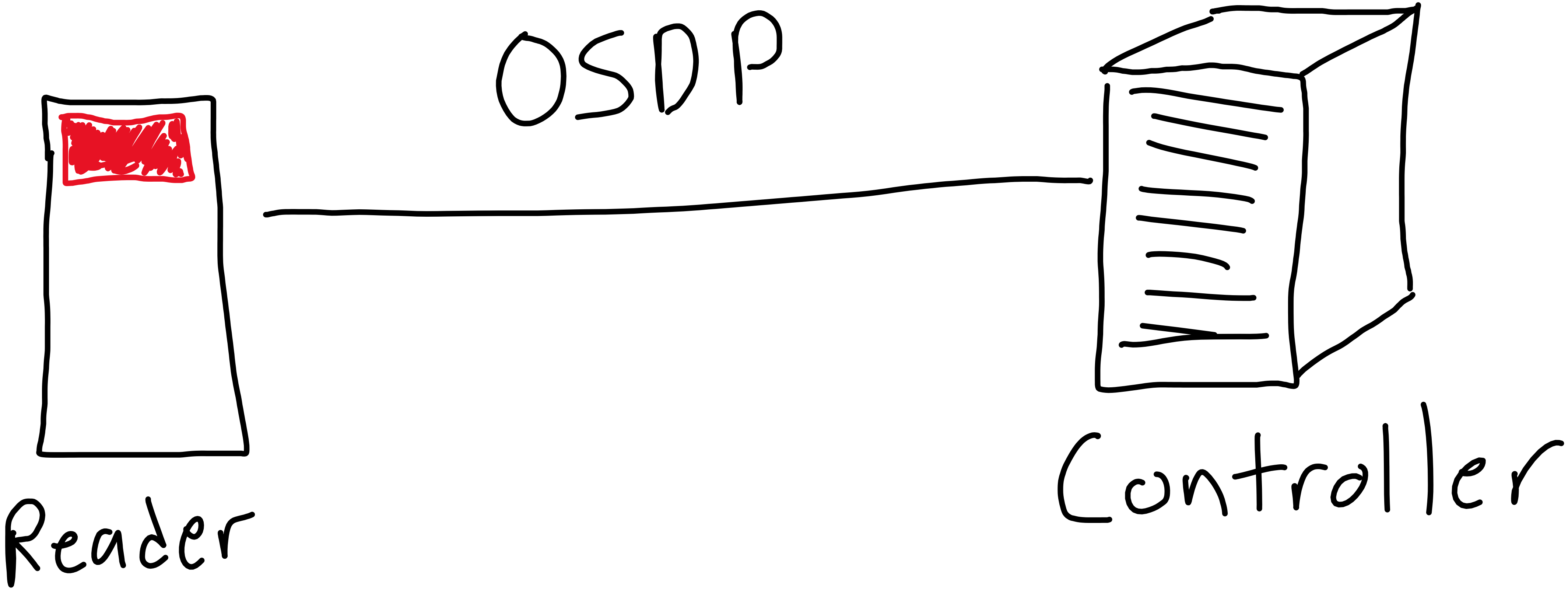
RFID Badge Setup



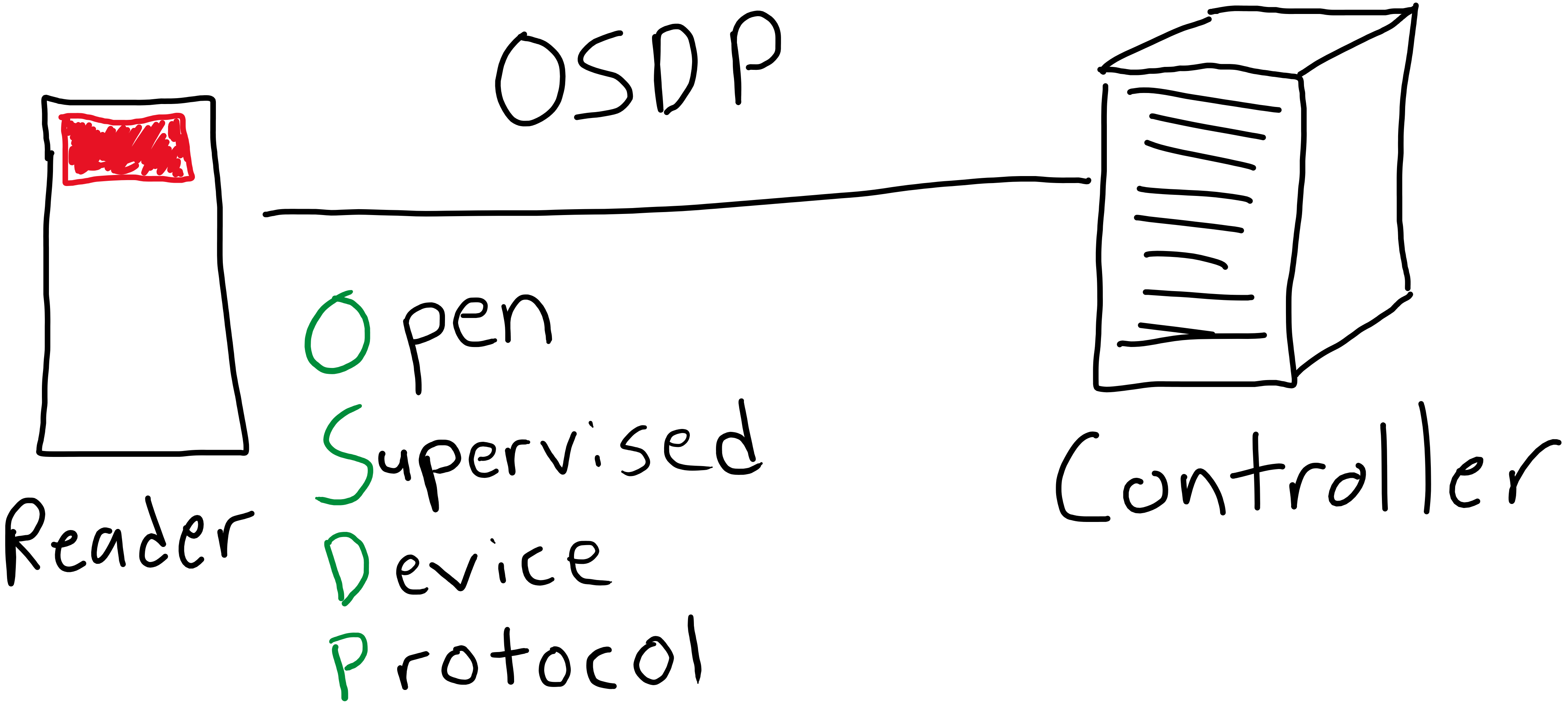
RFID Badge Setup



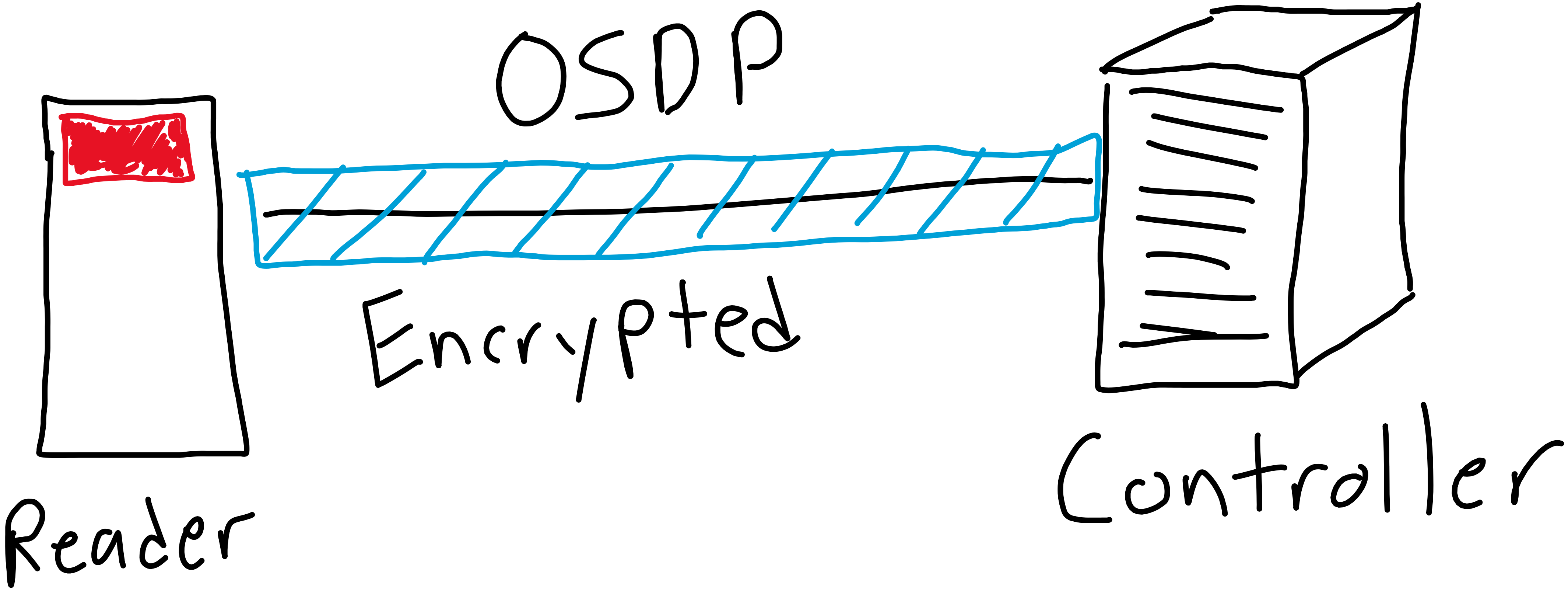
RFID Badge Setup



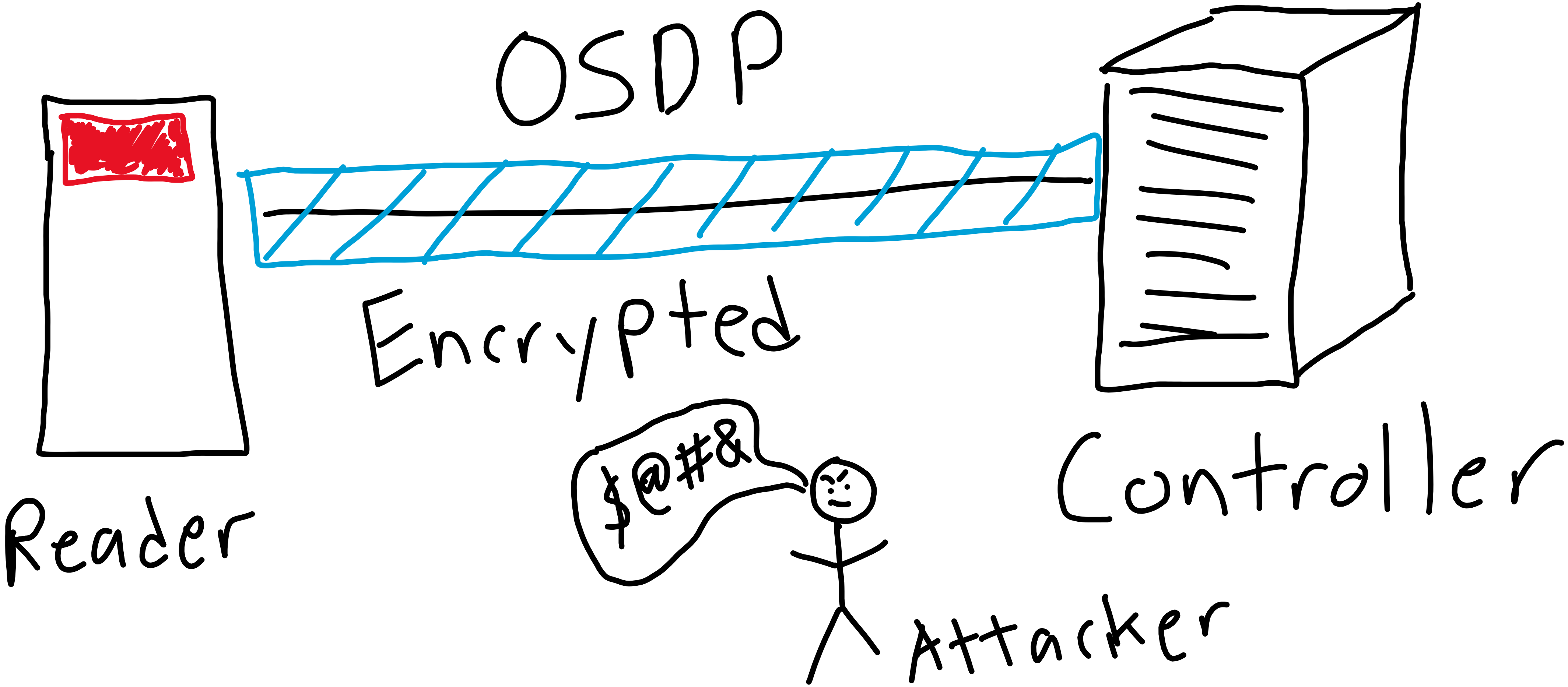
RFID Badge Setup



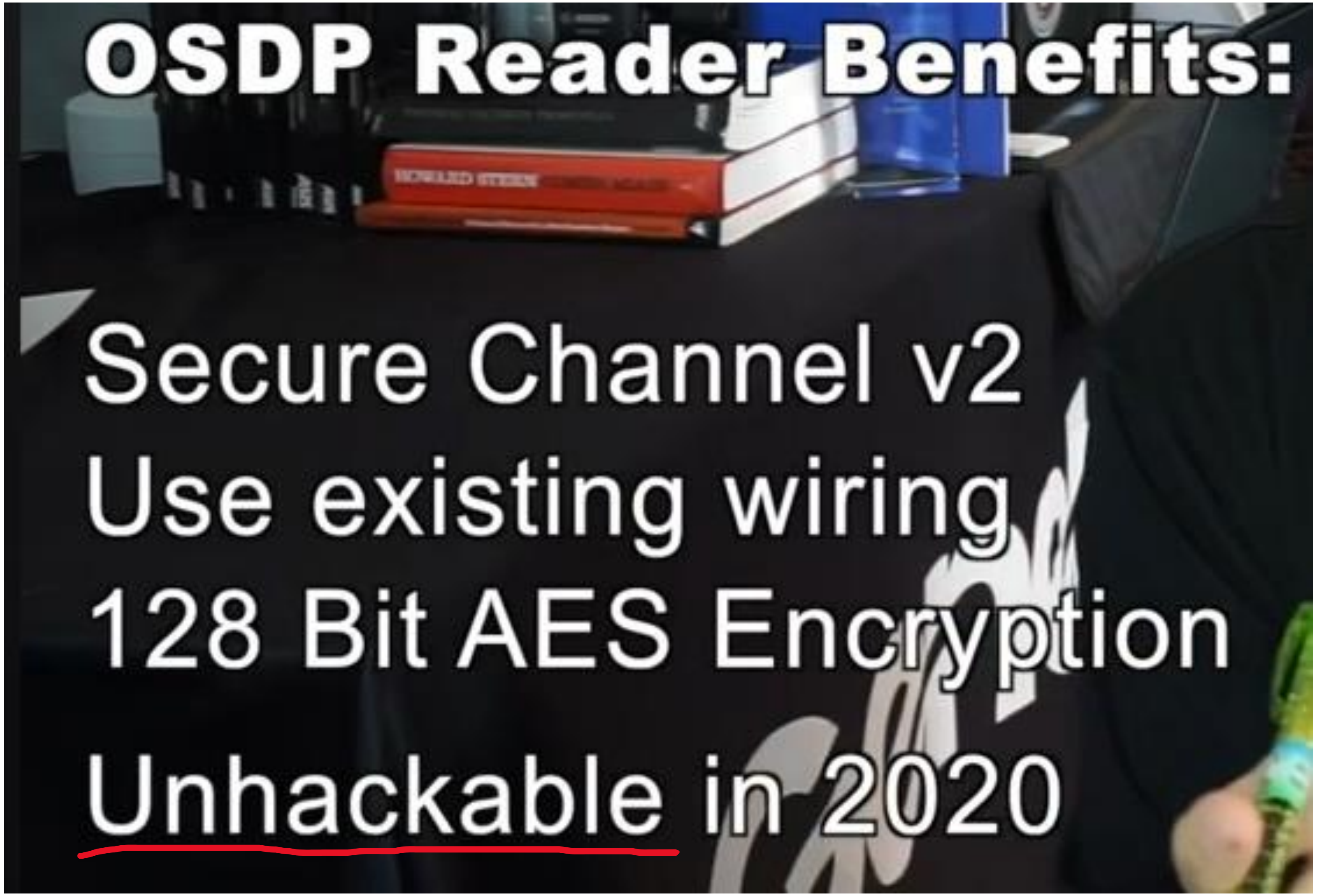
RFID Badge Setup



RFID Badge Setup



Bold Claims



OSDP Reader Benefits:

Secure Channel v2

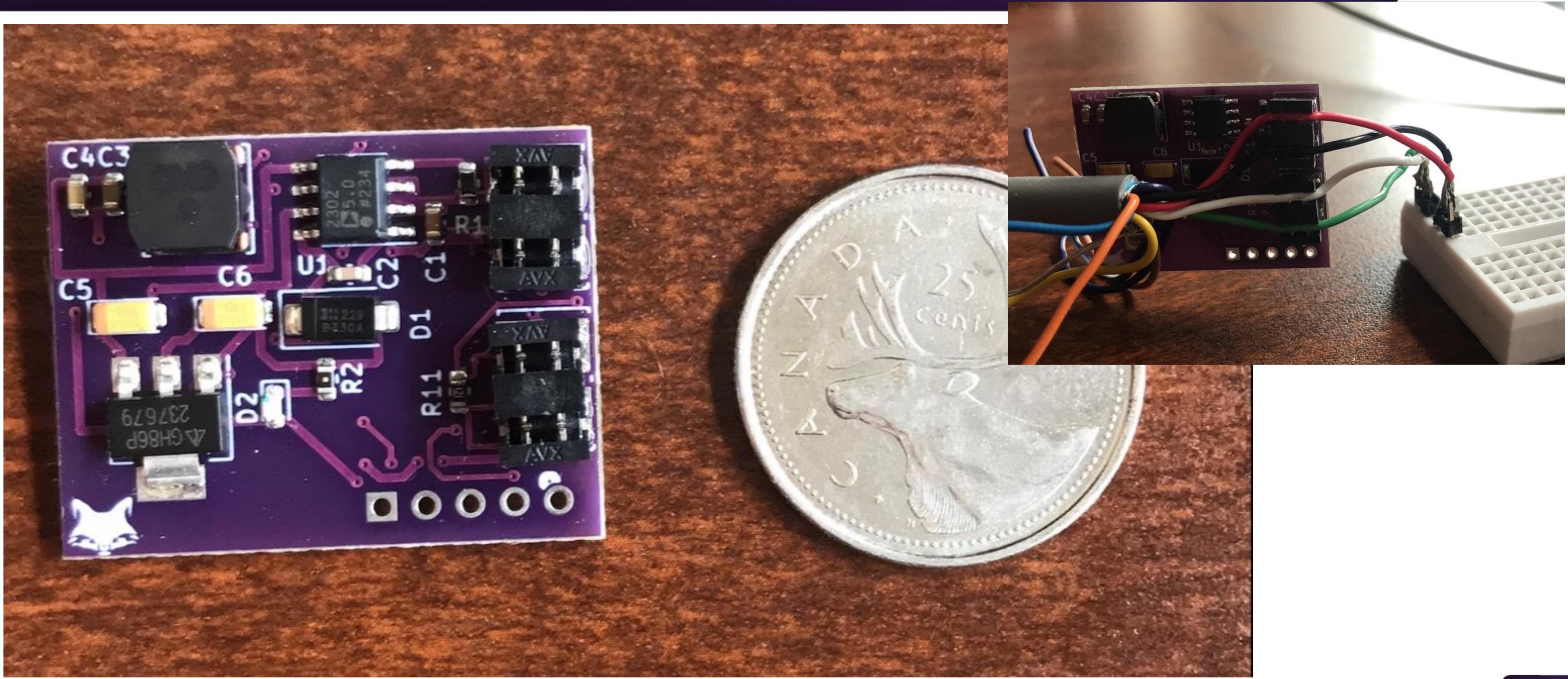
Use existing wiring

128 Bit AES Encryption

Unhackable in 2020

THE ELVISH WORD FOR FRIEND

Mellon



LET'S JUST TAKE A LOOK

Demo #1

Axis A1001

The Protocol

OSDP Supports

The Protocol

OSDP Supports

but doesn't require

The Protocol

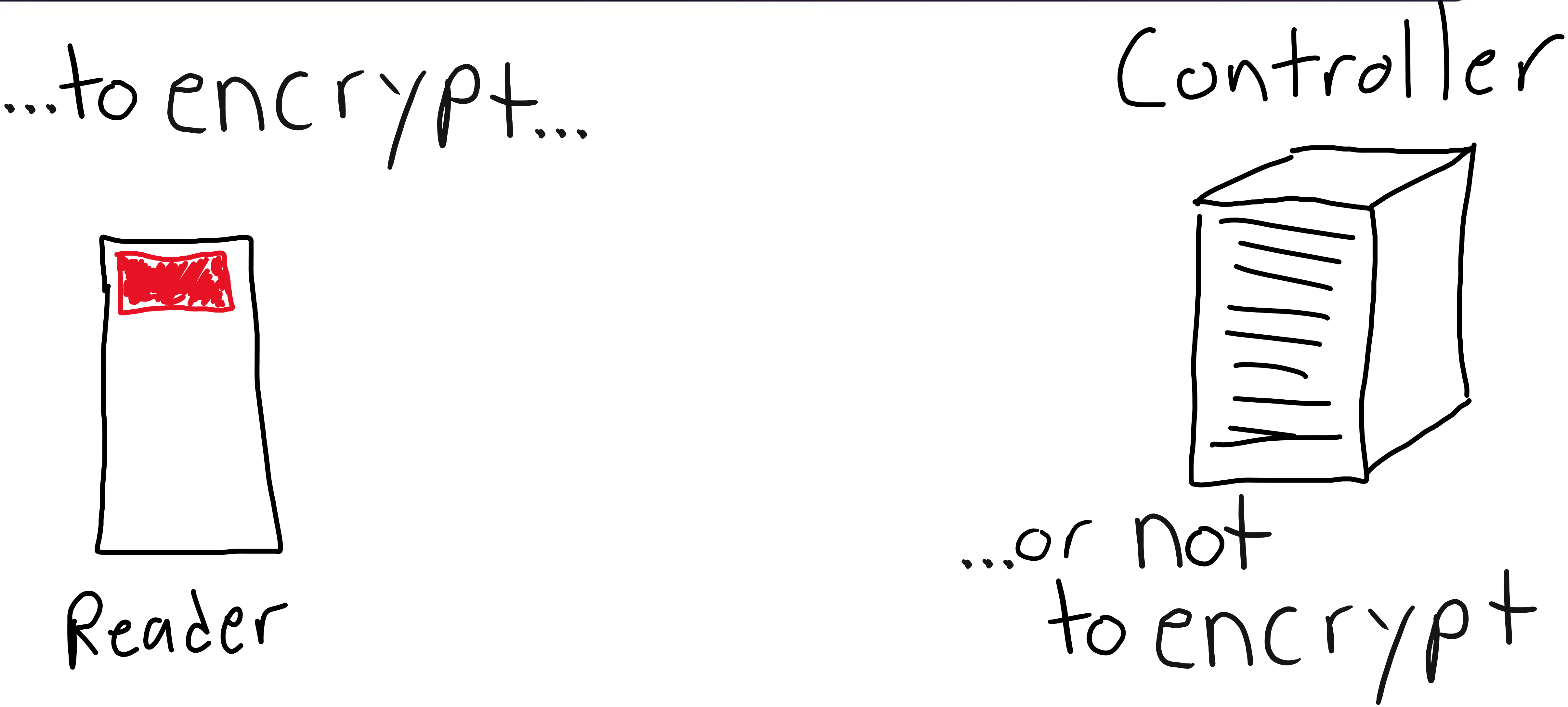
OSDP Supports but doesn't require encryption

The Protocol

OSDP-SC ("secure" chanel)

is an OSDP extension

The Protocol



OSDP

Protocol

Basics

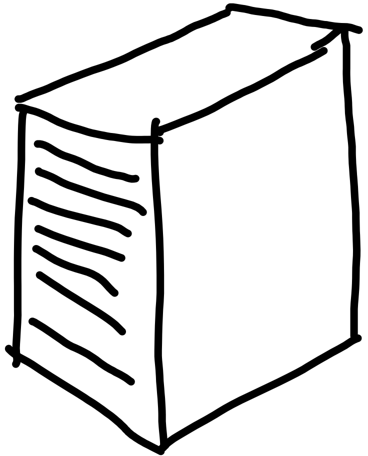
The Protocol

RS-485 (the <u>other</u> serial)

- multi-drop

The Protocol

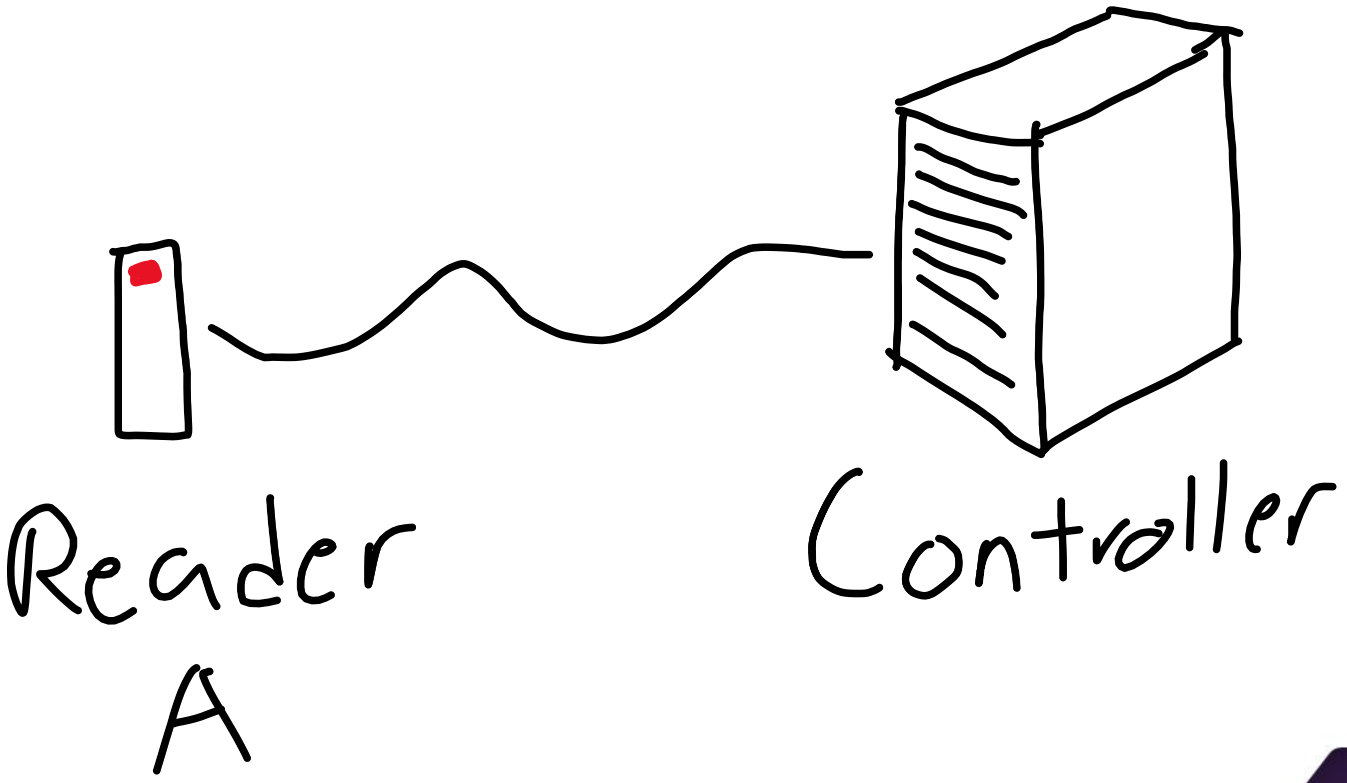
RS-485 (the <u>other</u> serial)



Controller

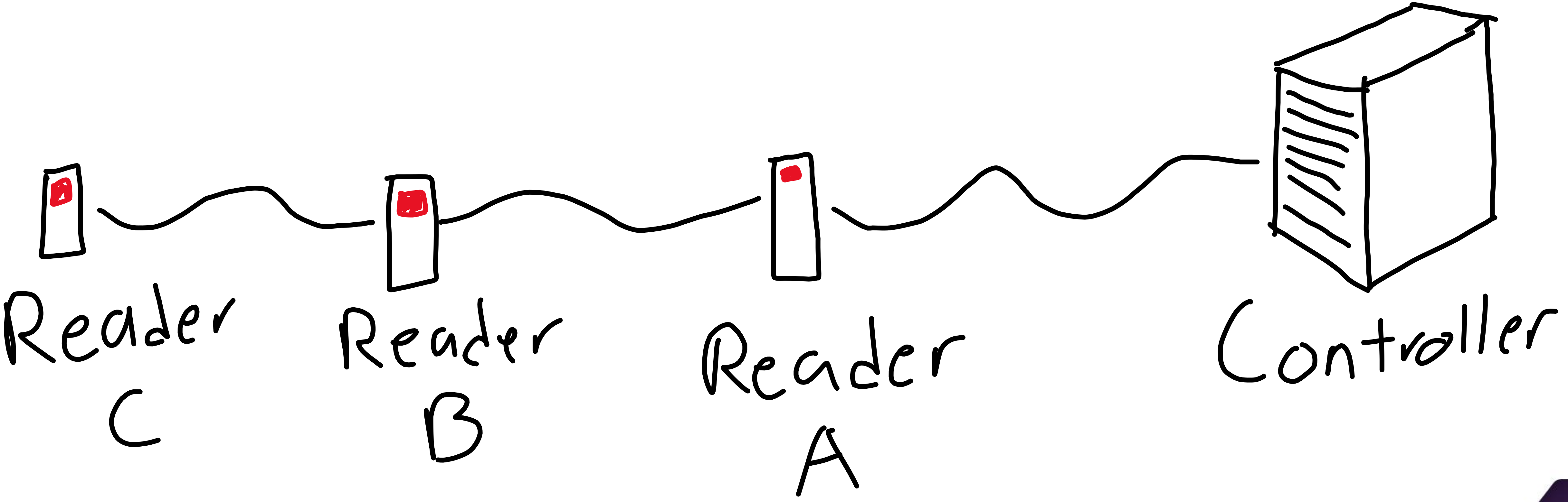
The Protocol

RS-485 (the _other_ serial)



The Protocol

RS-485 (the other serial)

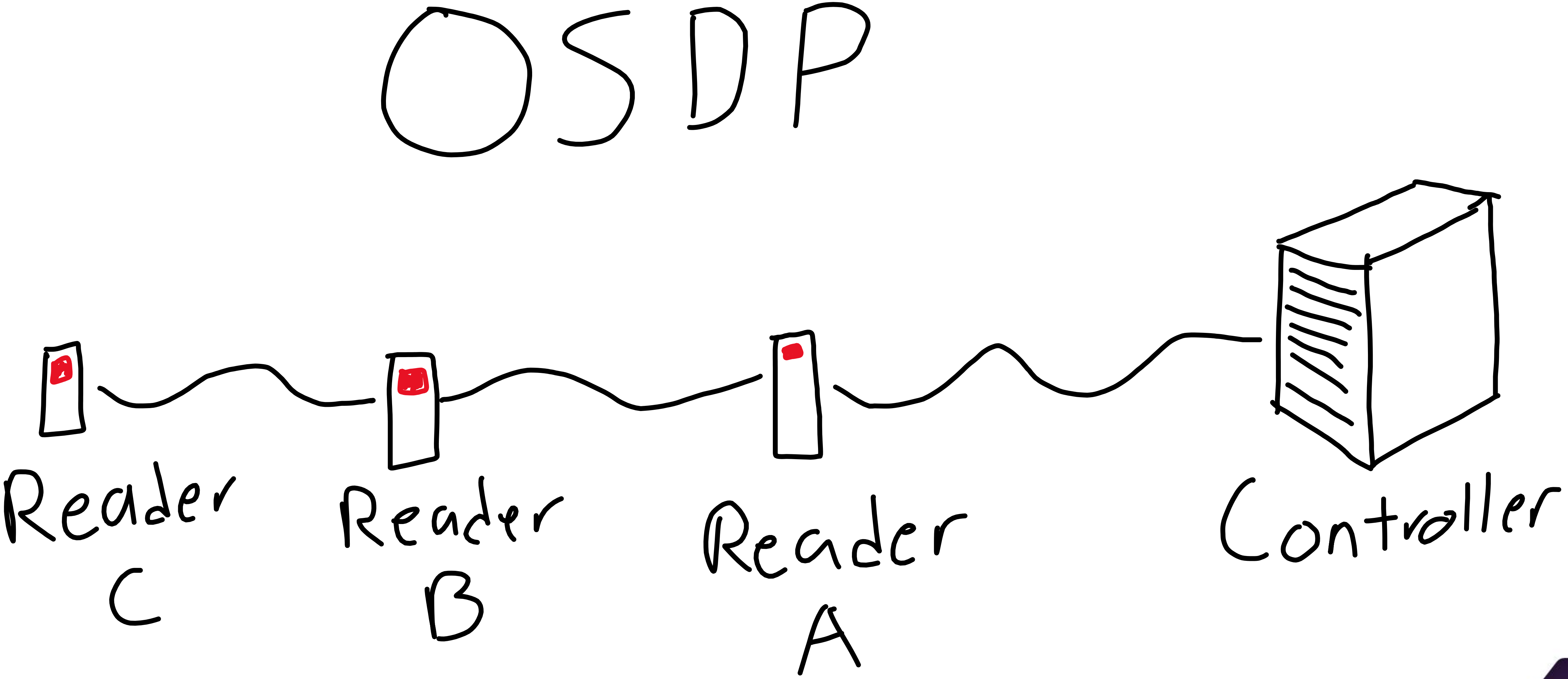


The Protocol

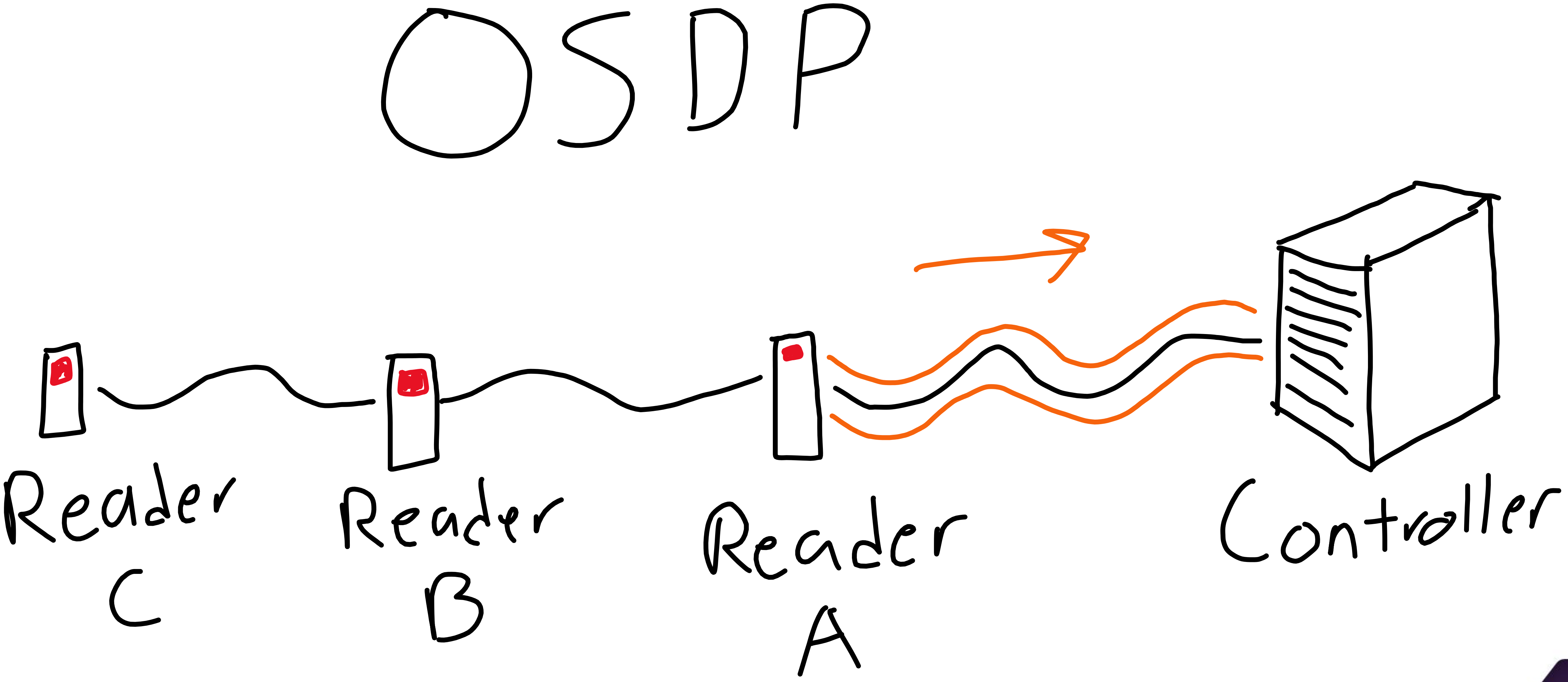
OSDP:

All messages are broadcast

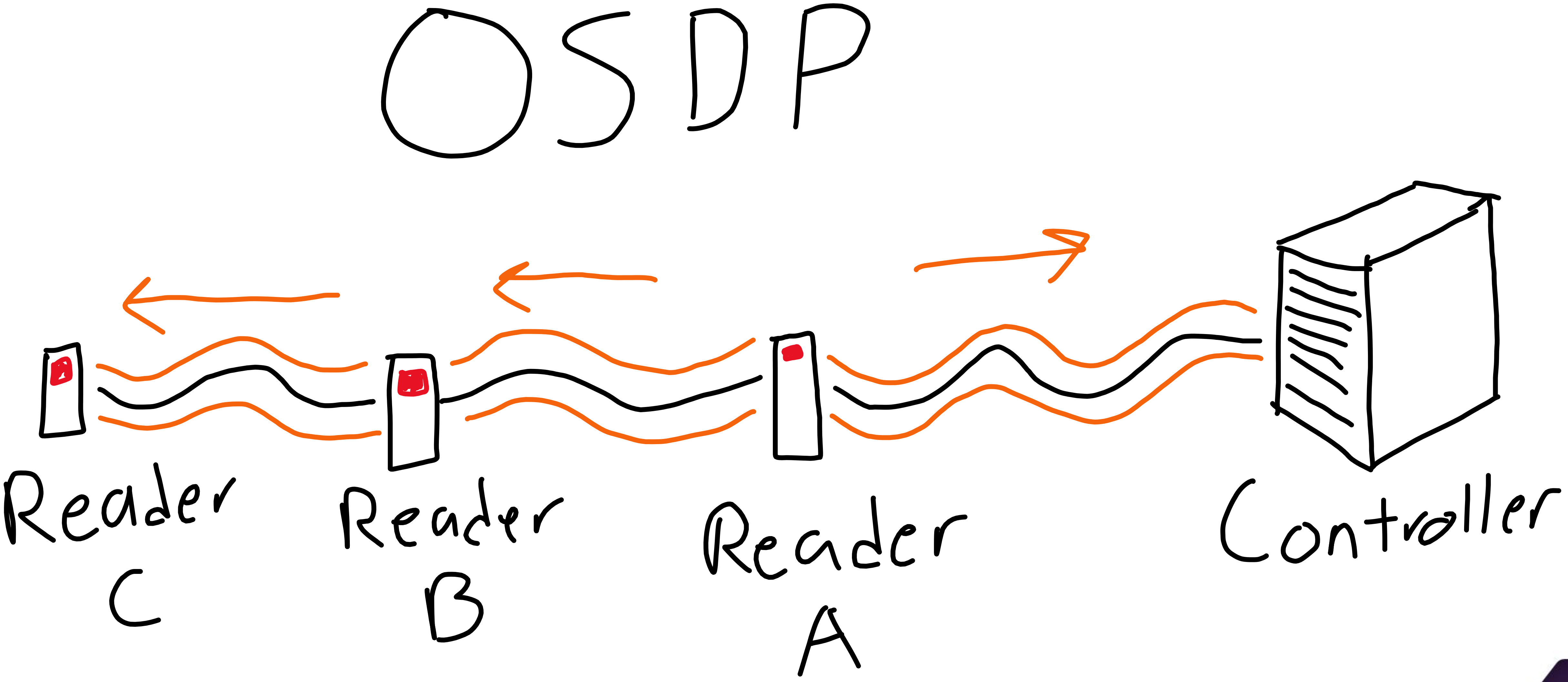
The Protocol



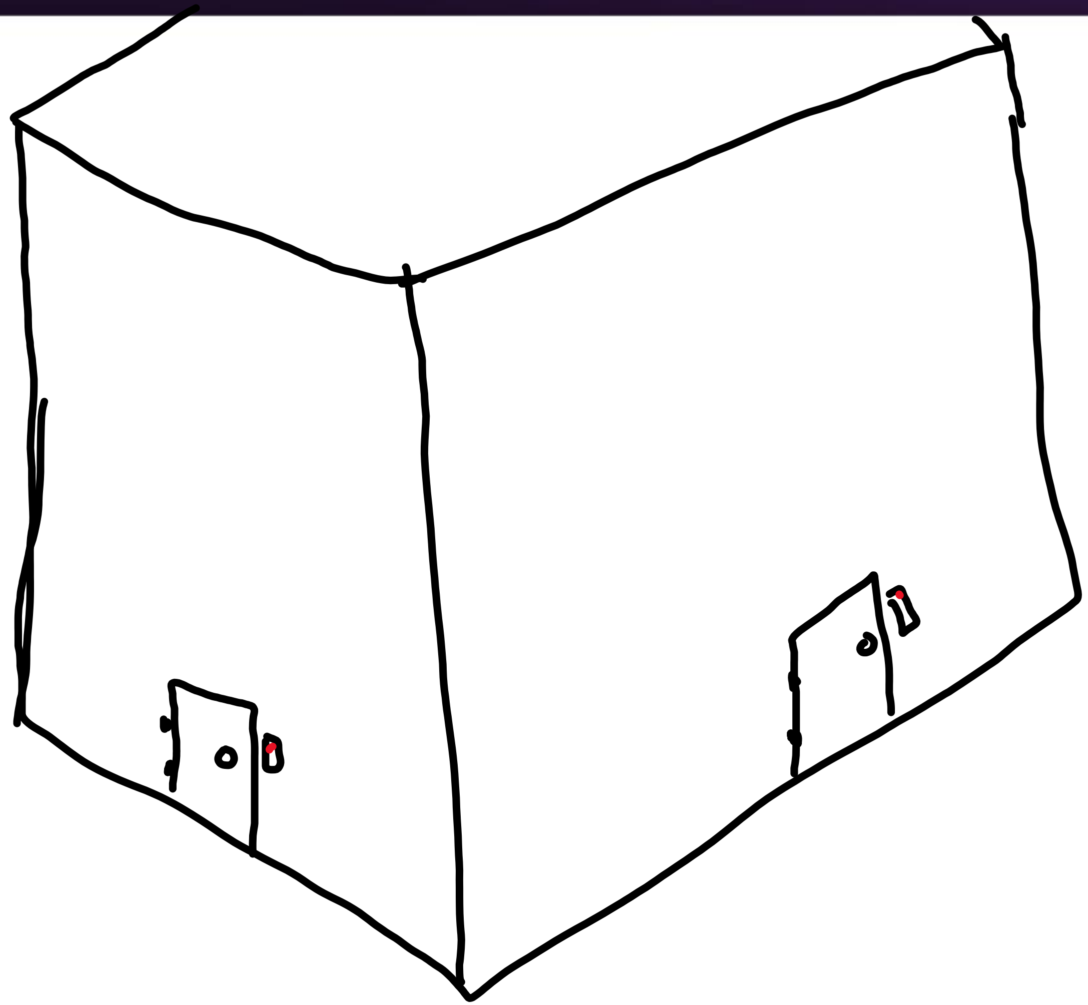
The Protocol



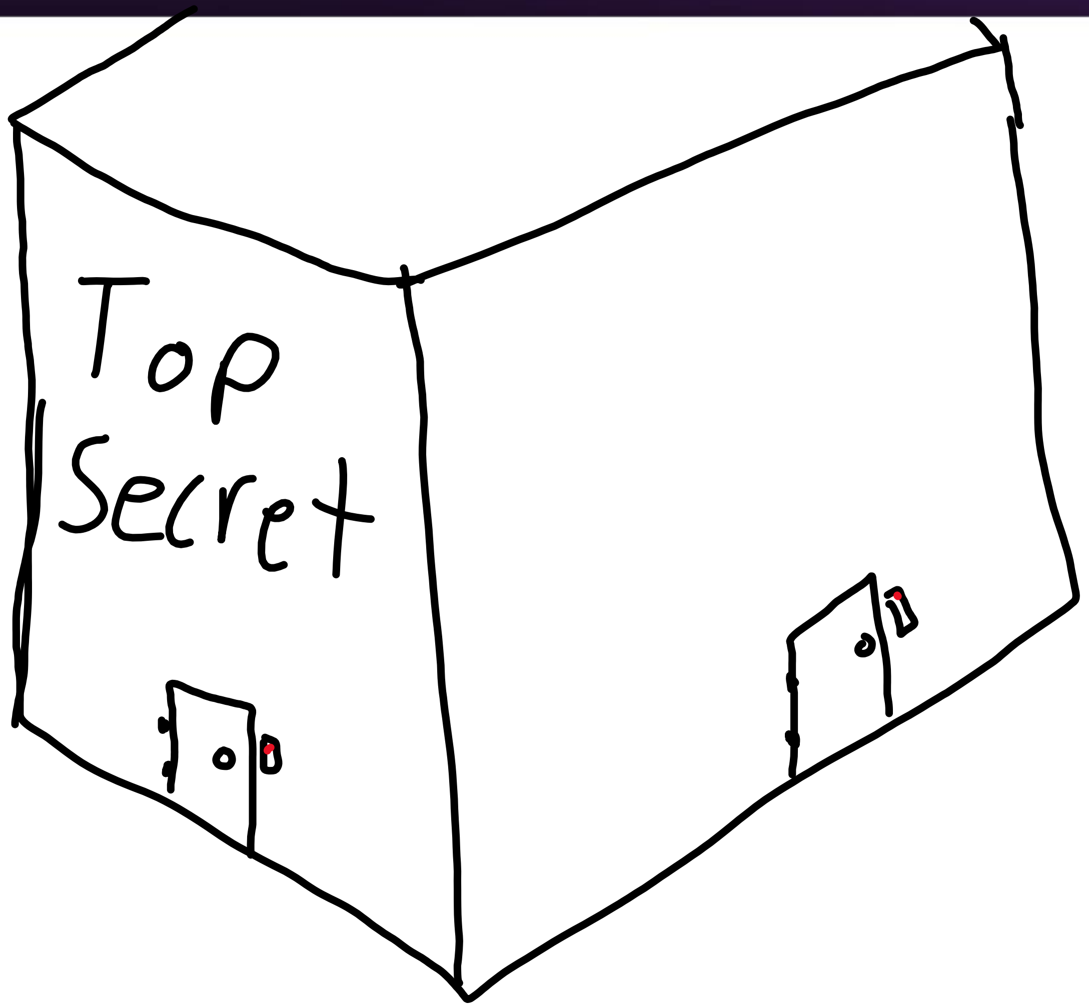
The Protocol



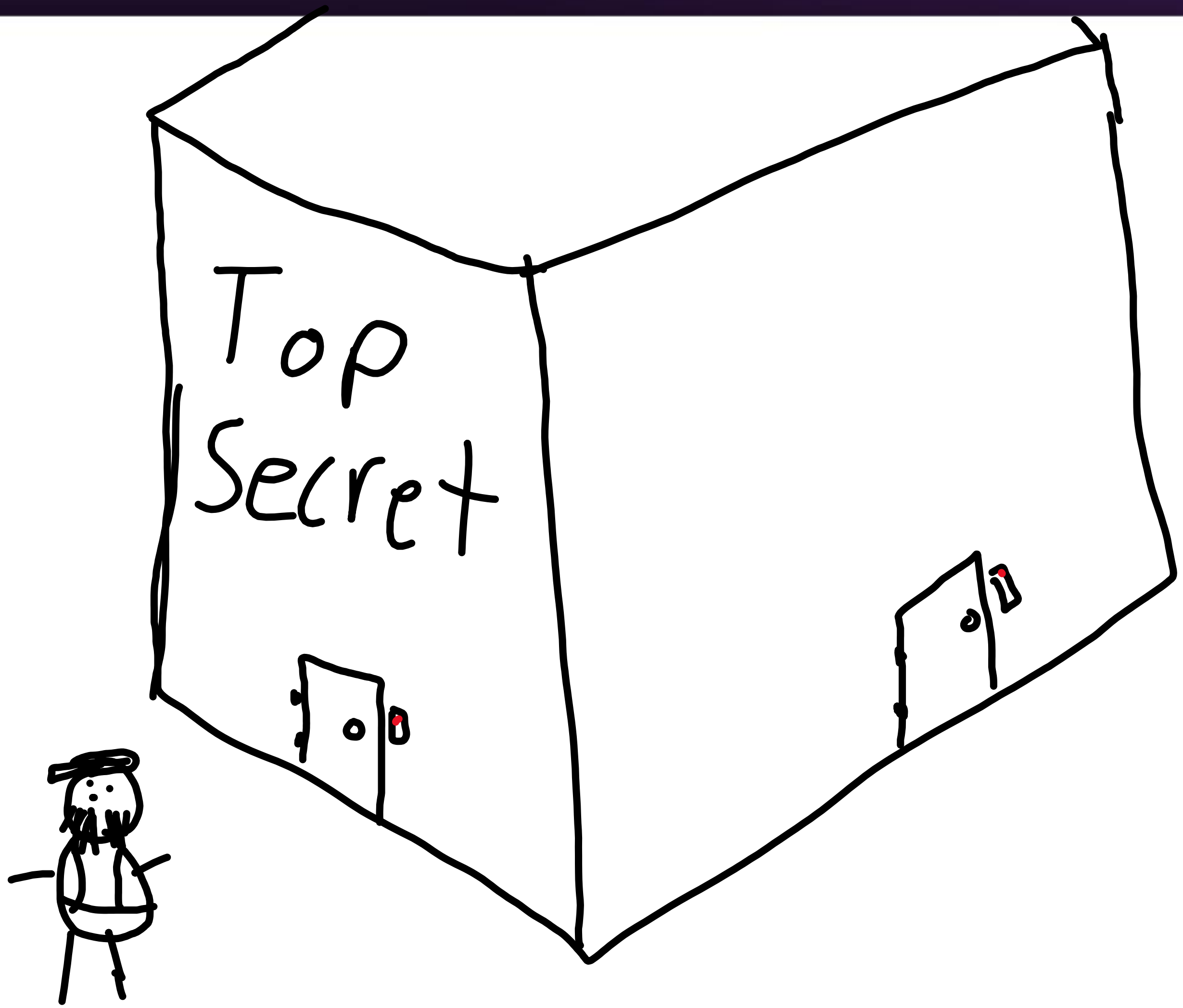
Undisclosed Location in Santa Fe, NM



Undisclosed Location in Santa Fe, NM



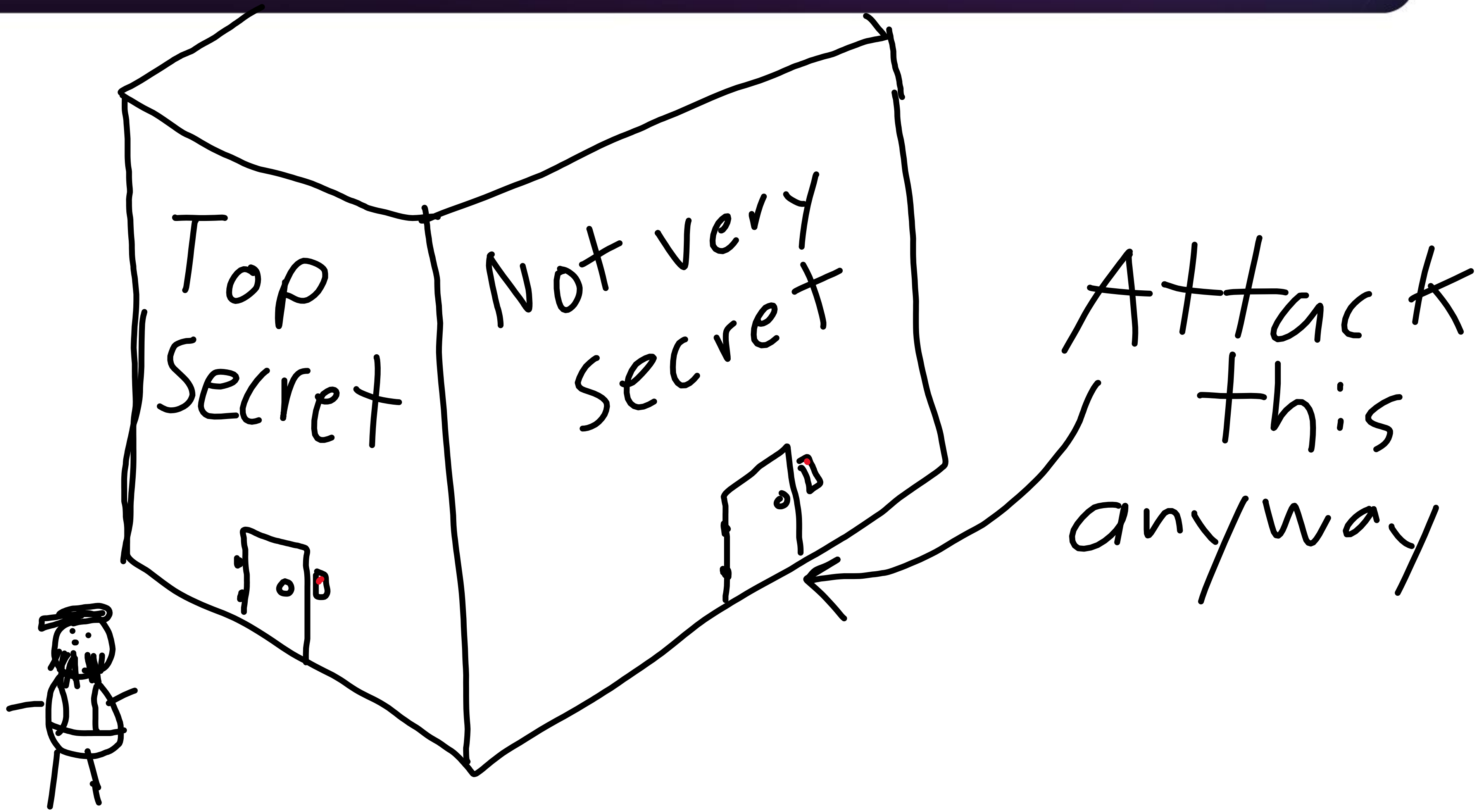
Undisclosed Location in Santa Fe, NM



Undisclosed Location in Santa Fe, NM

Top Secret
Not very secret

Undisclosed Location in Santa Fe, NM

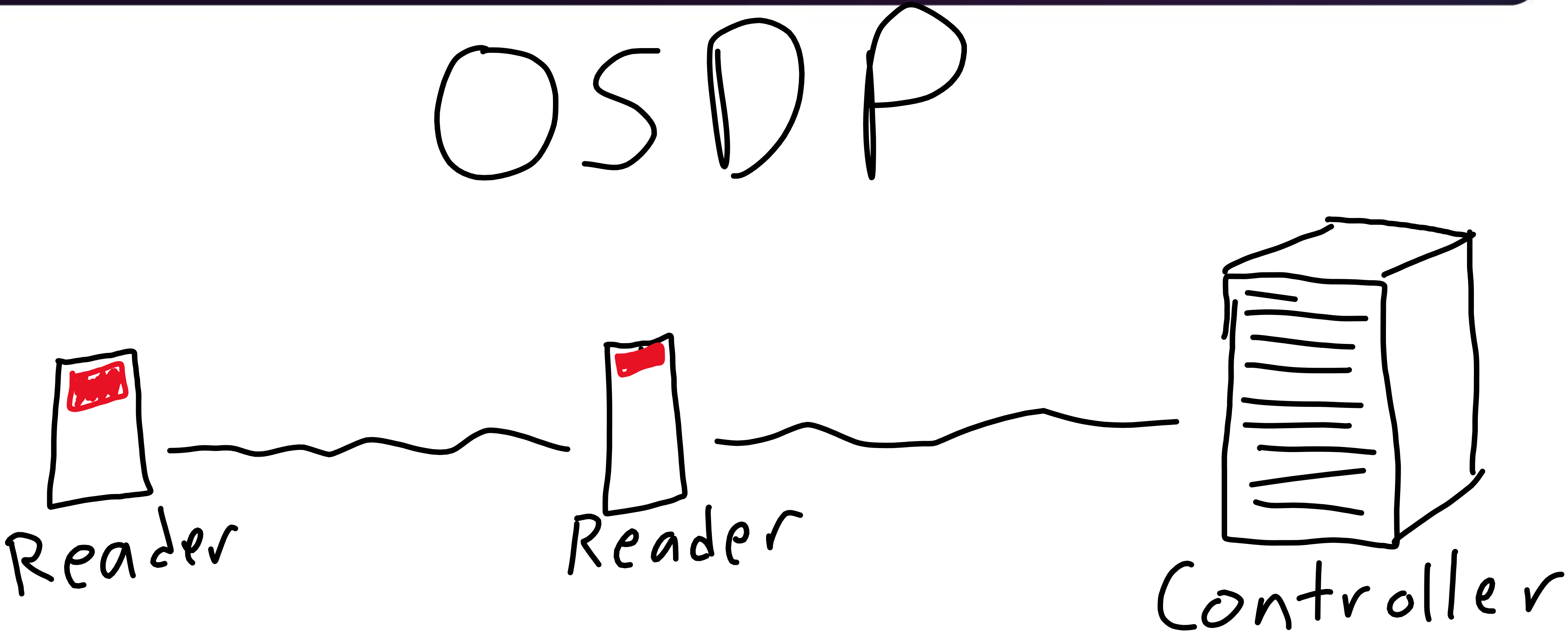


The Protocol

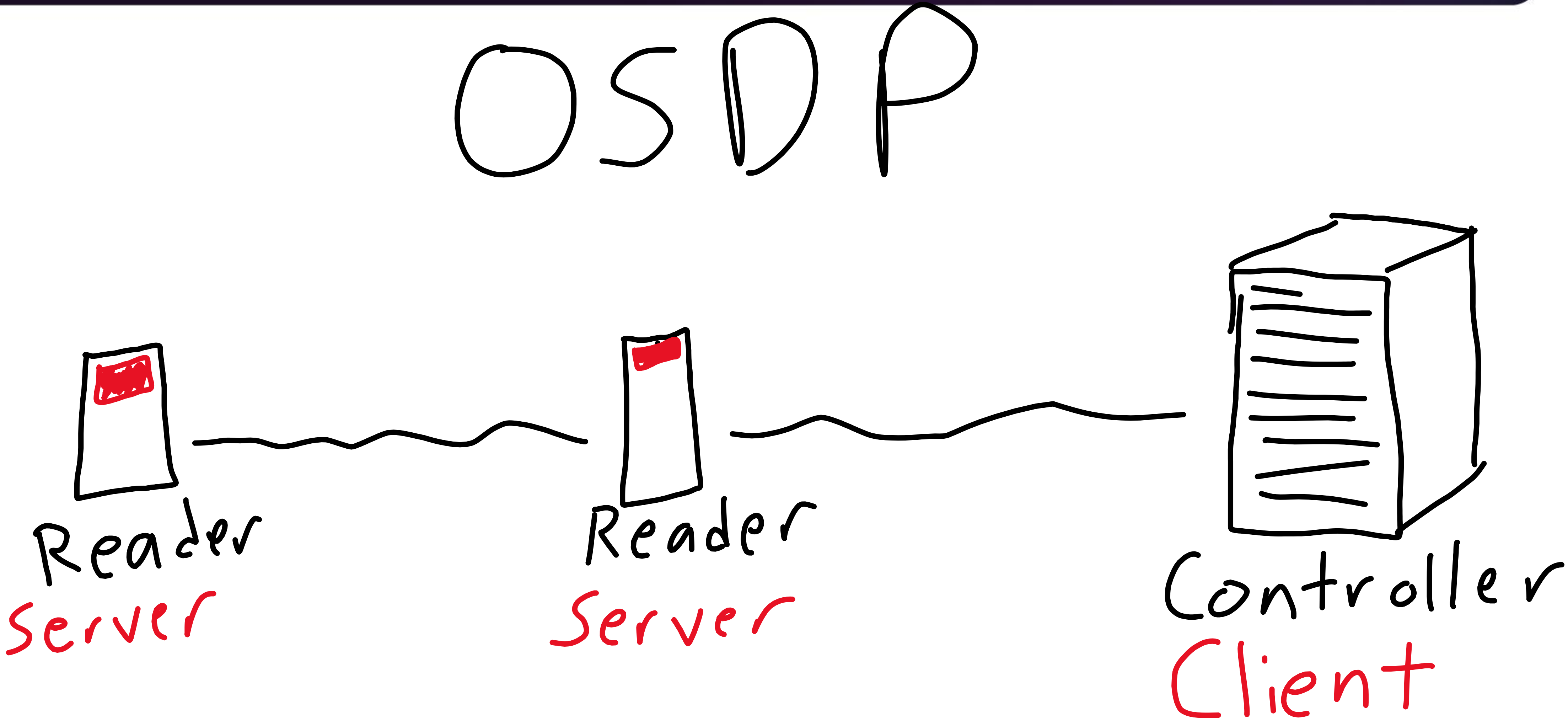
OSDP:

Client-Server
Model

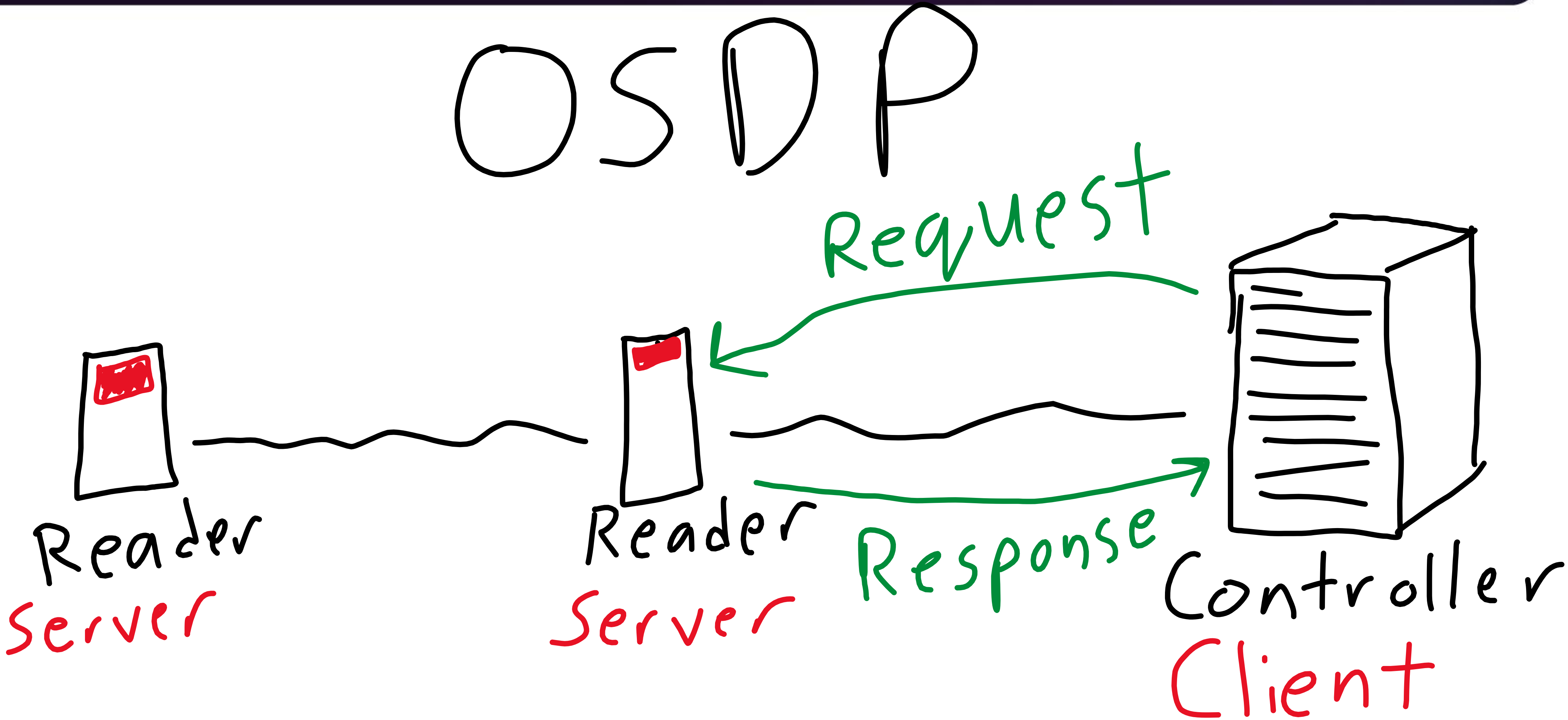
The Protocol



The Protocol



The Protocol

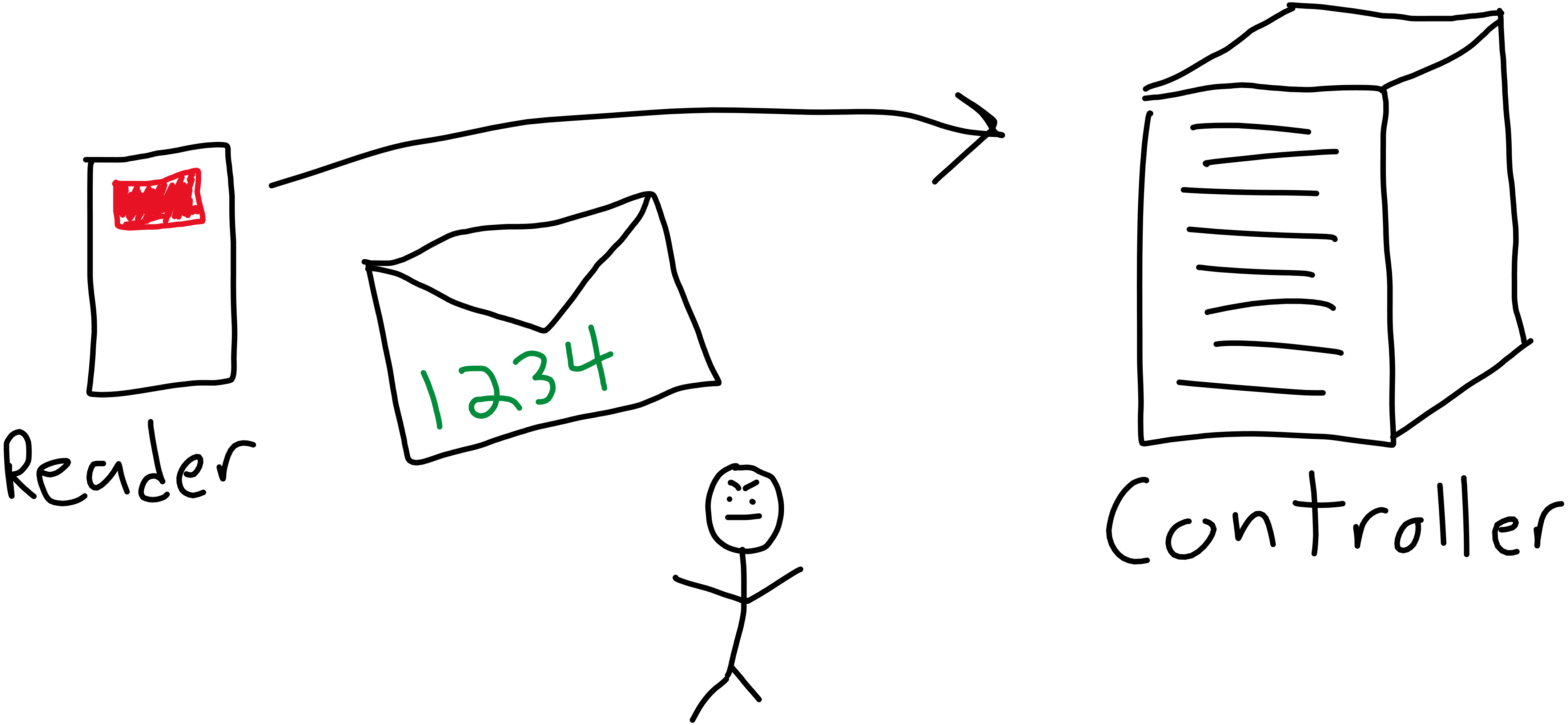


Protocol WTF #1

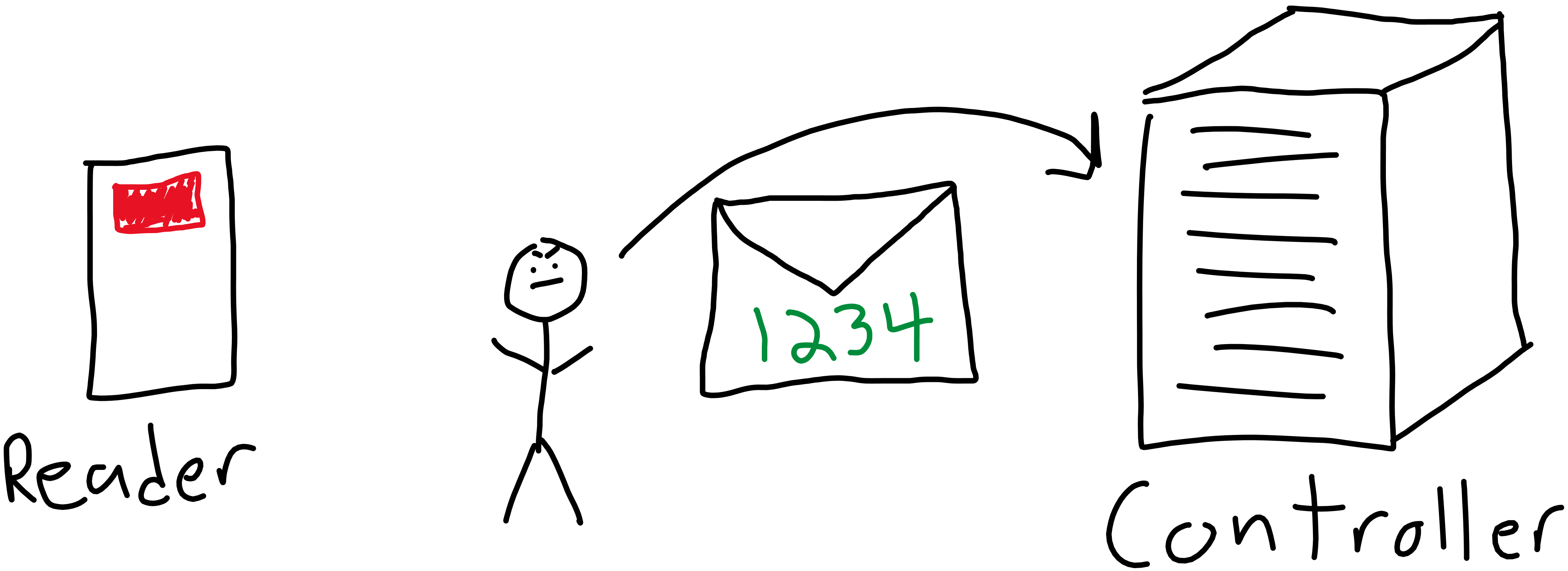
Protocol WTF #1

What, are we paying by the bit now?

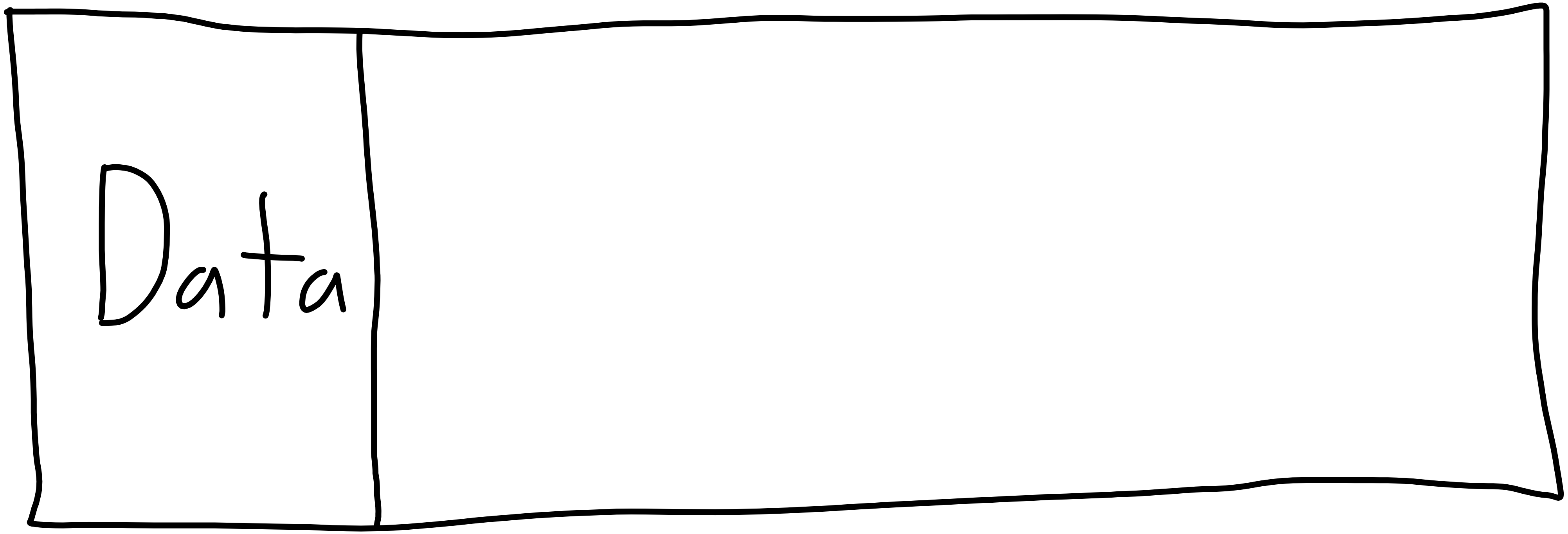
Replay Attacks



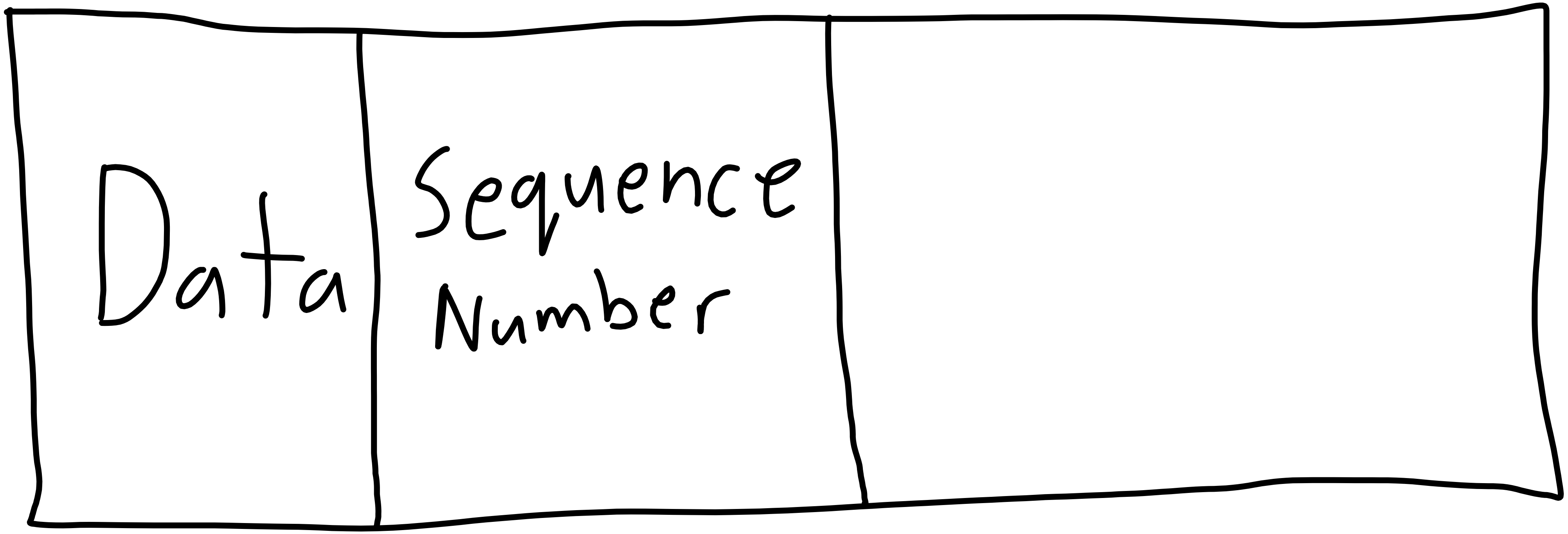
Replay Attacks



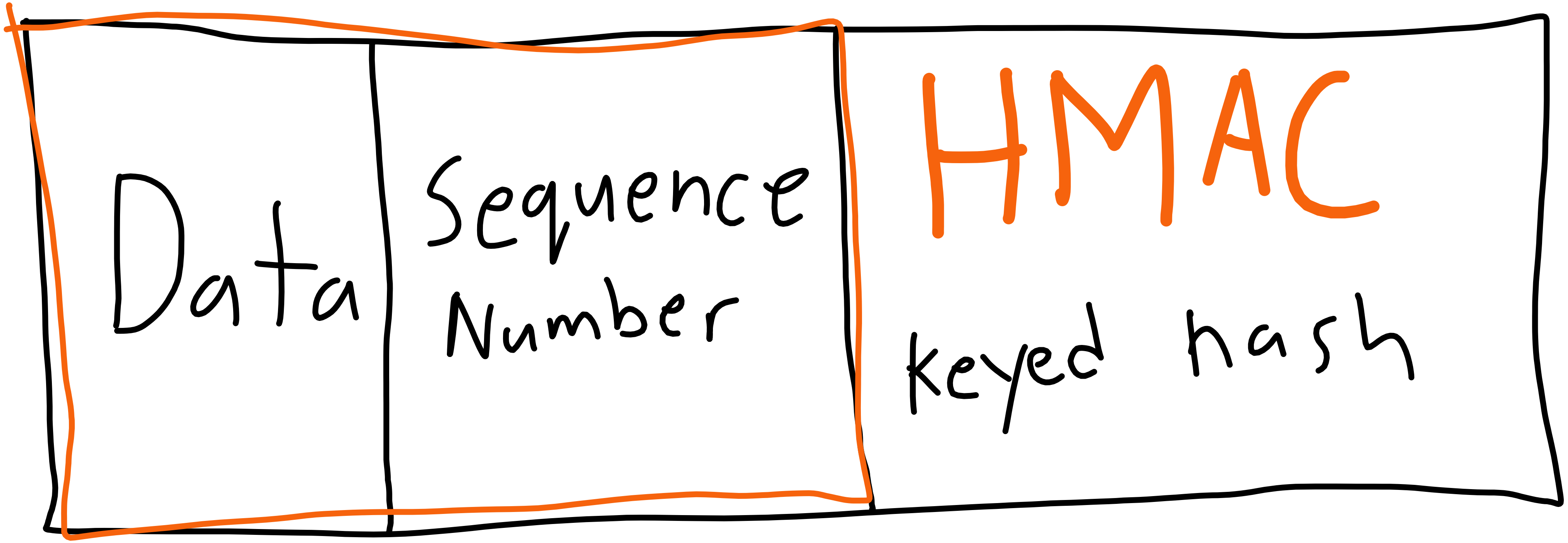
Sequence Number + HMAC



Sequence Number + HMAC



Sequence Number + HMAC



SEQUENCE NUMBER

How Many Bits is Enough?

128-bits? - Cryptographic Strength

SEQUENCE NUMBER

How Many Bits is Enough?

~~128-bits?~~ - Cryptographic Strength

64-bits? - Edge of Enumeration

How Many Bits is Enough?

~~128-bits?~~ - Cryptographic Strength

~~64-bits?~~ - Edge of Enumeration

32-bits? - Fine, maybe?

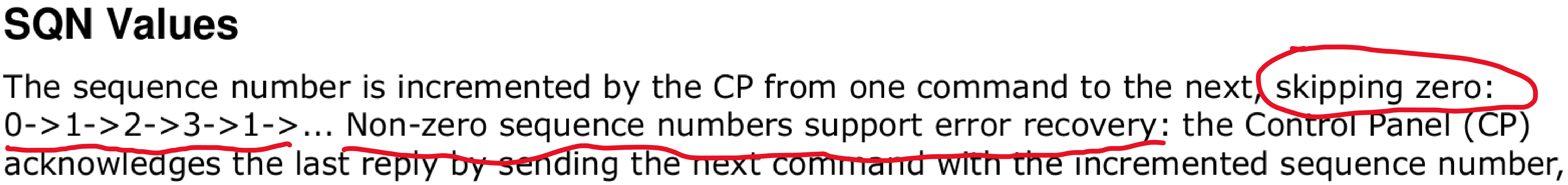
How Many Bits is Enough?

2-bits

How Many Bits is Enough?

~~2~~ 1½ -bits

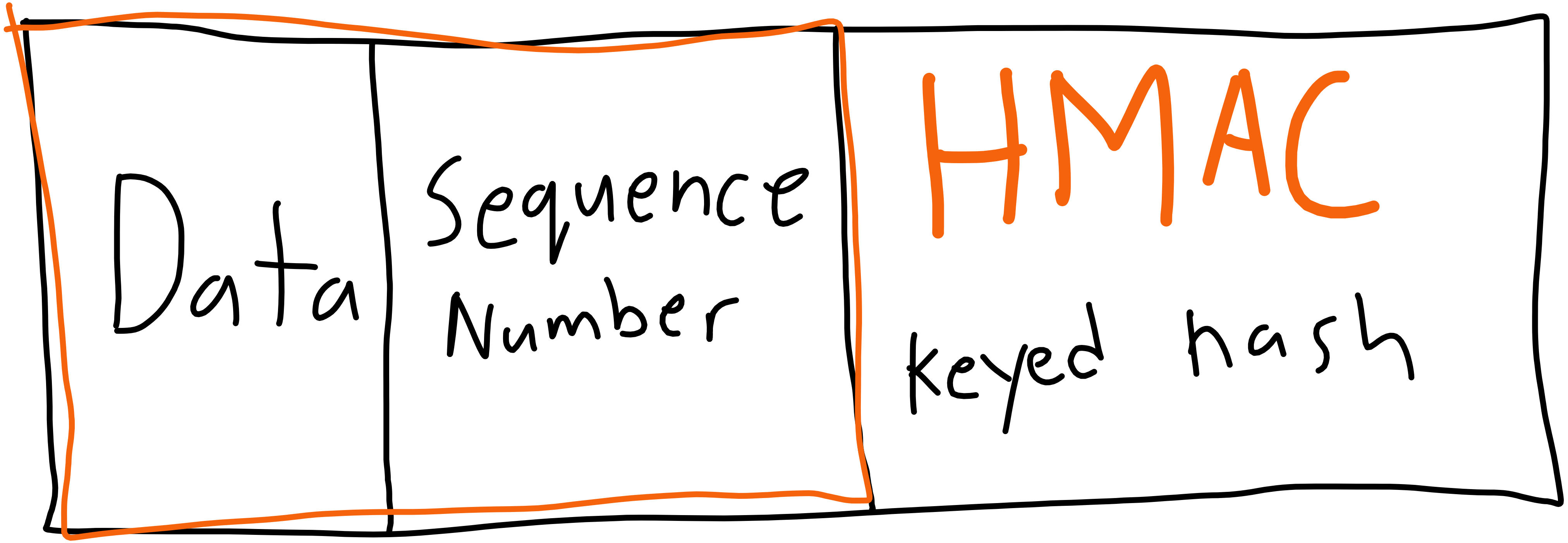
The Protocol



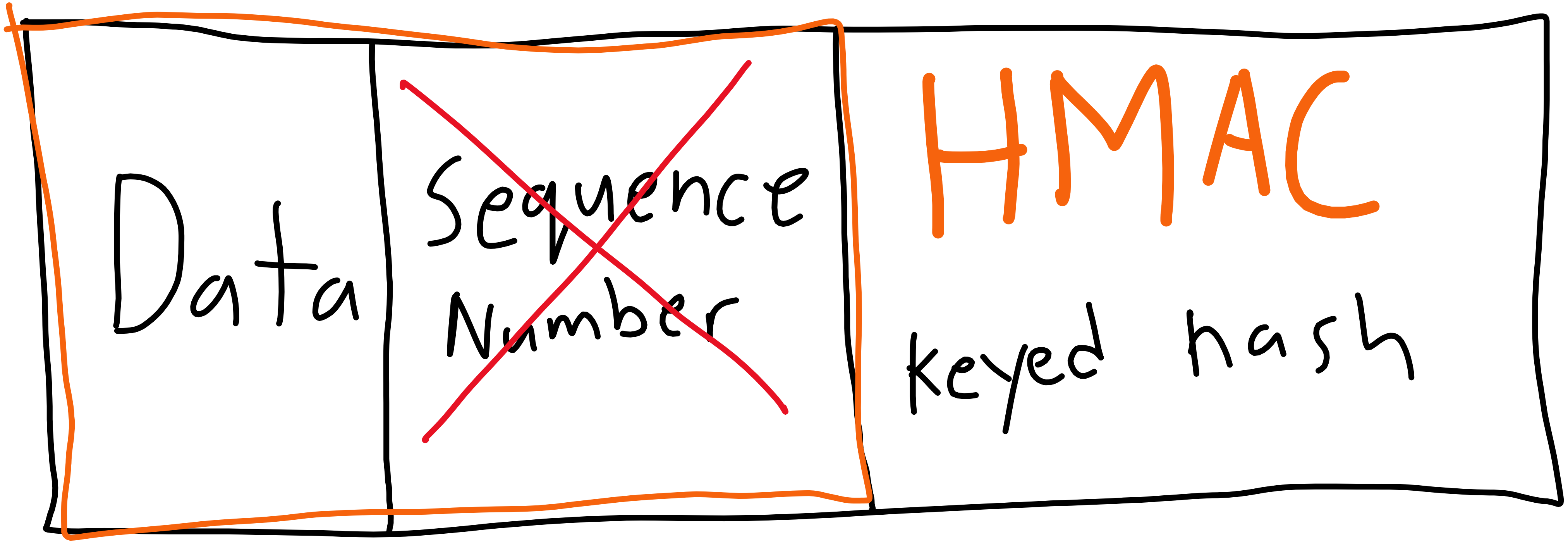
SQN Values

The sequence number is incremented by the CP from one command to the next, skipping zero: 0->1->2->3->1->... Non-zero sequence numbers support error recovery: the Control Panel (CP) acknowledges the last reply by sending the next command with the incremented sequence number,

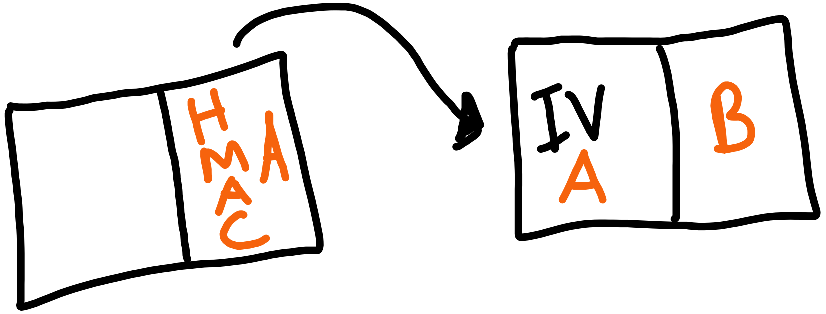
Sequence Number + HMAC



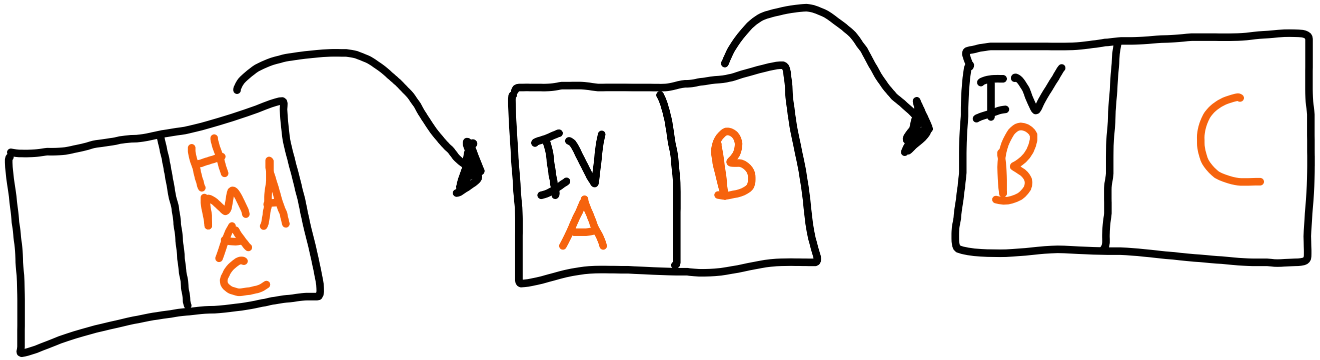
Sequence Number + HMAC



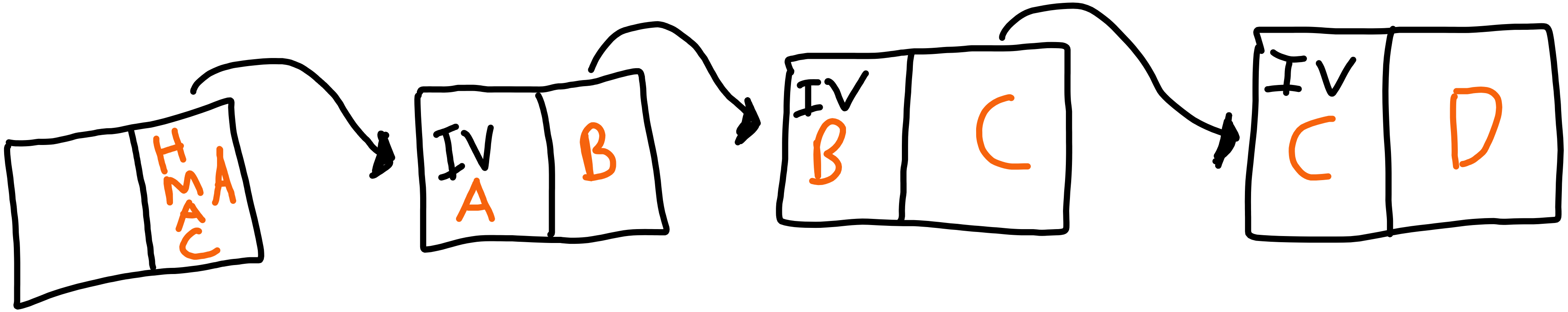
IV Chaining



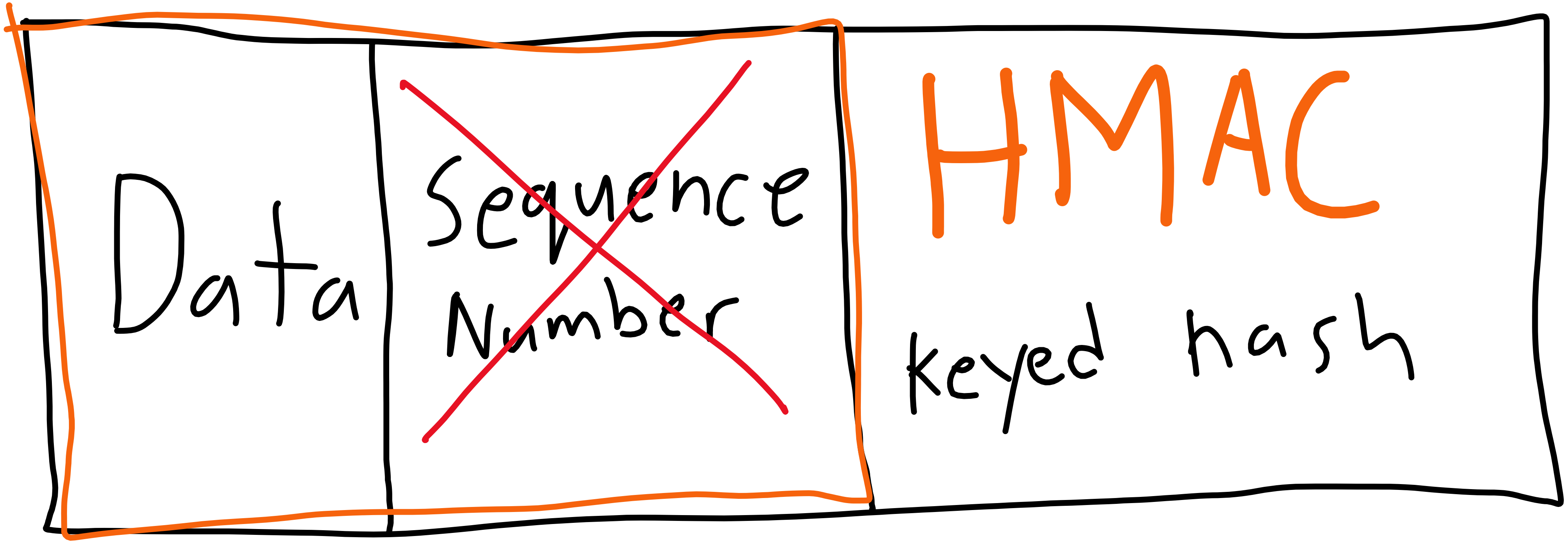
IV Chaining



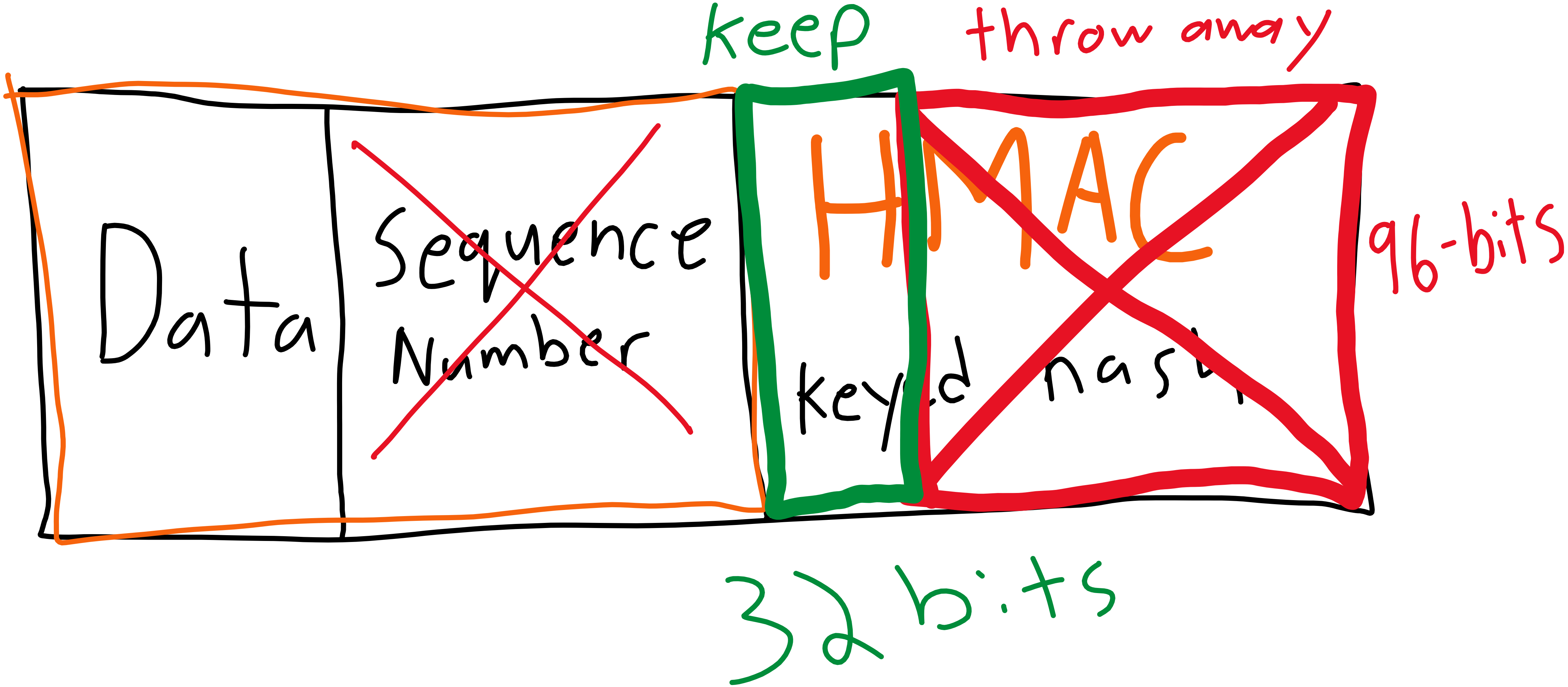
IV Chaining



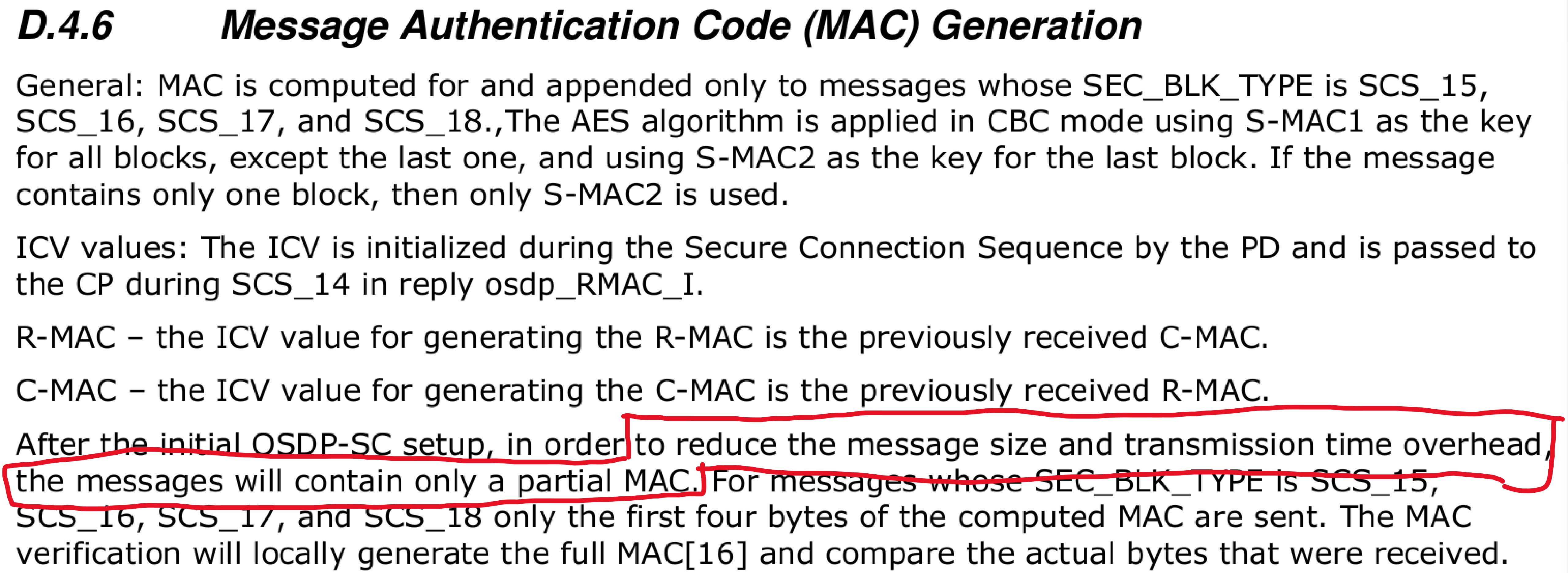
Sequence Number + HMAC



Sequence Number + HMAC



Overhead



D.4.6 Message Authentication Code (MAC) Generation

General: MAC is computed for and appended only to messages whose SEC_BLK_TYPE is SCS_15, SCS_16, SCS_17, and SCS_18.,The AES algorithm is applied in CBC mode using S-MAC1 as the key for all blocks, except the last one, and using S-MAC2 as the key for the last block. If the message contains only one block, then only S-MAC2 is used.

ICV values: The ICV is initialized during the Secure Connection Sequence by the PD and is passed to the CP during SCS_14 in reply osdp_RMAC_I.

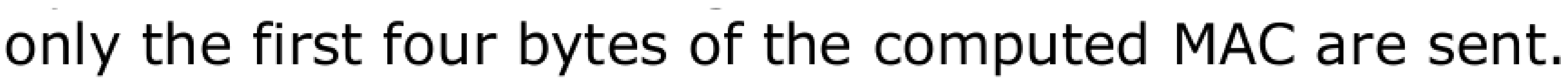
R-MAC – the ICV value for generating the R-MAC is the previously received C-MAC.

C-MAC – the ICV value for generating the C-MAC is the previously received R-MAC.

After the initial OSDP-SC setup, in order to reduce the message size and transmission time overhead, the messages will contain only a partial MAC. For messages whose SEC_BLK_TYPE is SCS_15, SCS_16, SCS_17, and SCS_18 only the first four bytes of the computed MAC are sent. The MAC verification will locally generate the full MAC[16] and compare the actual bytes that were received.

Overhead

Wat

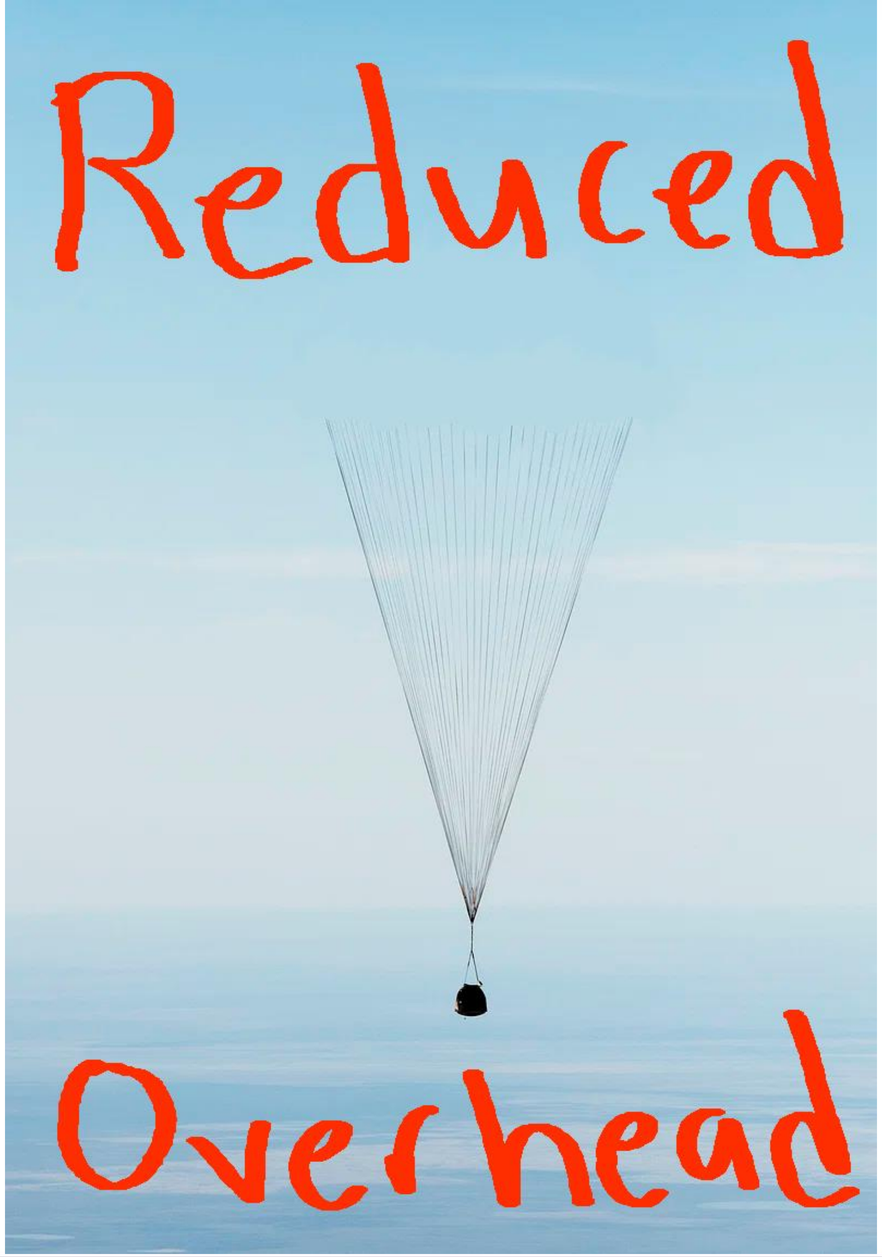


only the first four bytes of the computed MAC are sent.

only the first four bytes

BRITISH AIRWAYS
G-ZBKA

Reduced
BRITISH AIRWAYS
G-ZBKA
Overhead


Reduced
Overhead

Math

2^{31} attempts (on average)

Math

2^{31} attempts (on average)

115,200 baud (RS-485)

Math

2^{31} attempts (on average)

115,200 baud (RS-485)

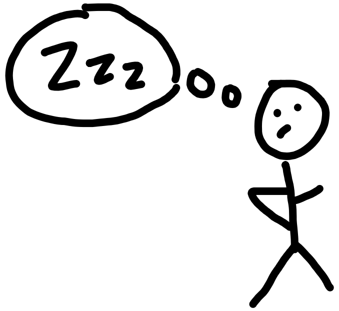
~ success after

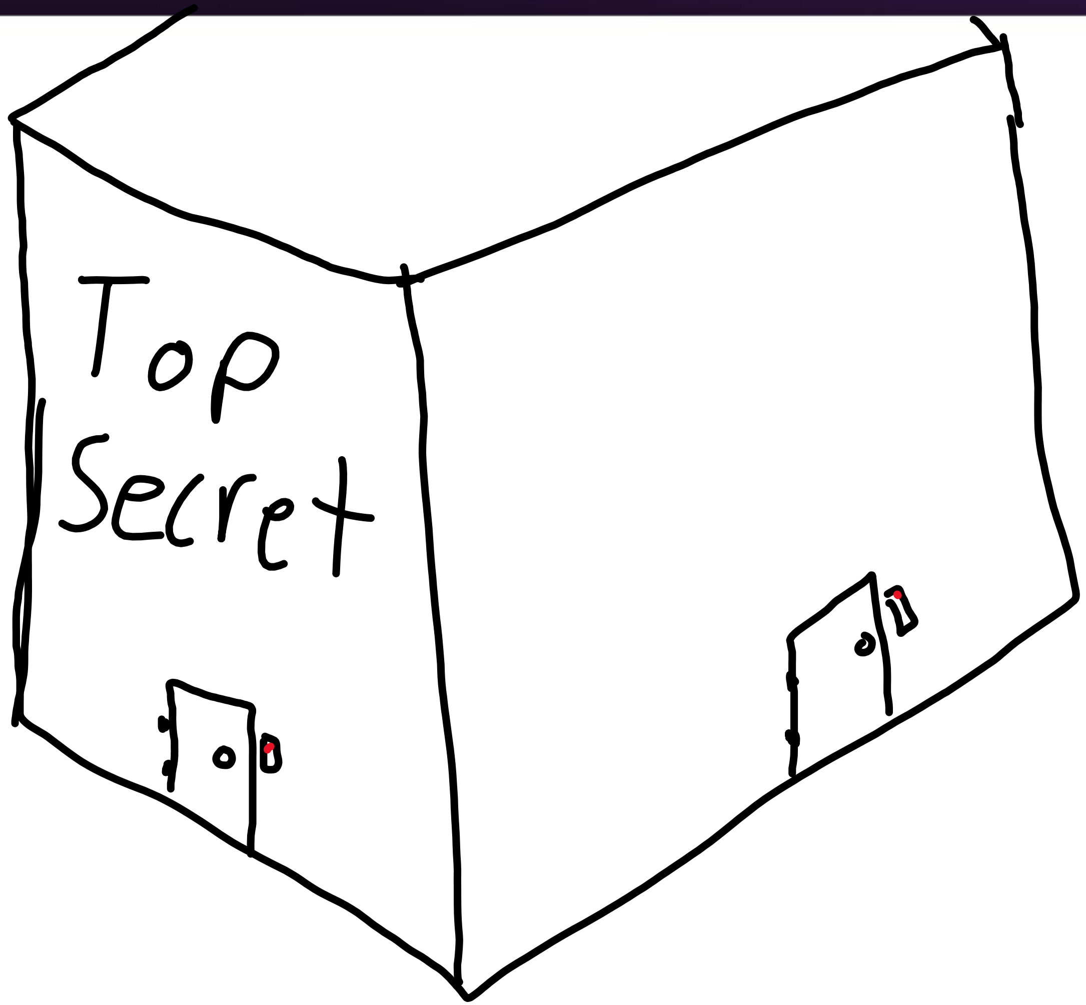
35 days

Math

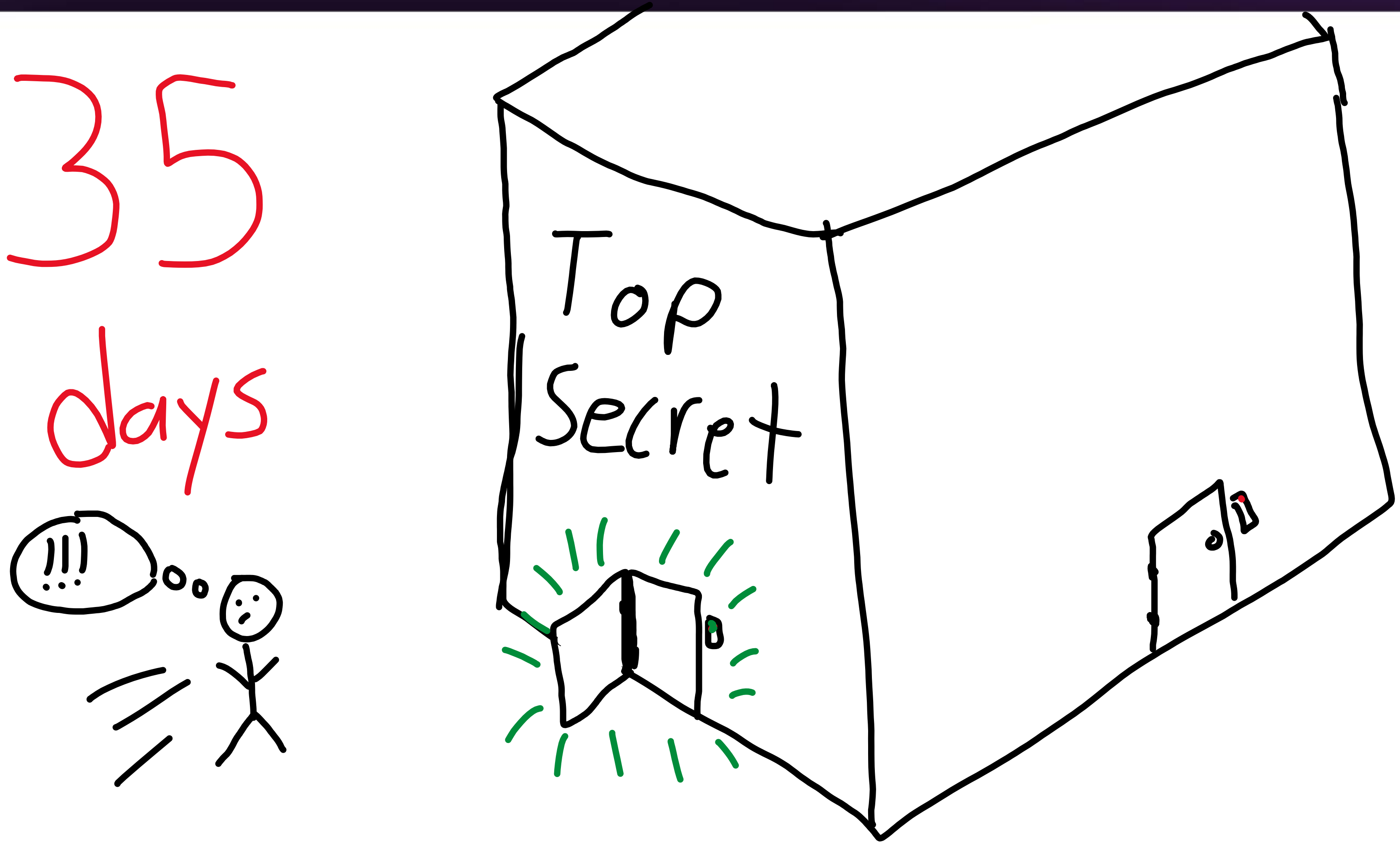
35

days

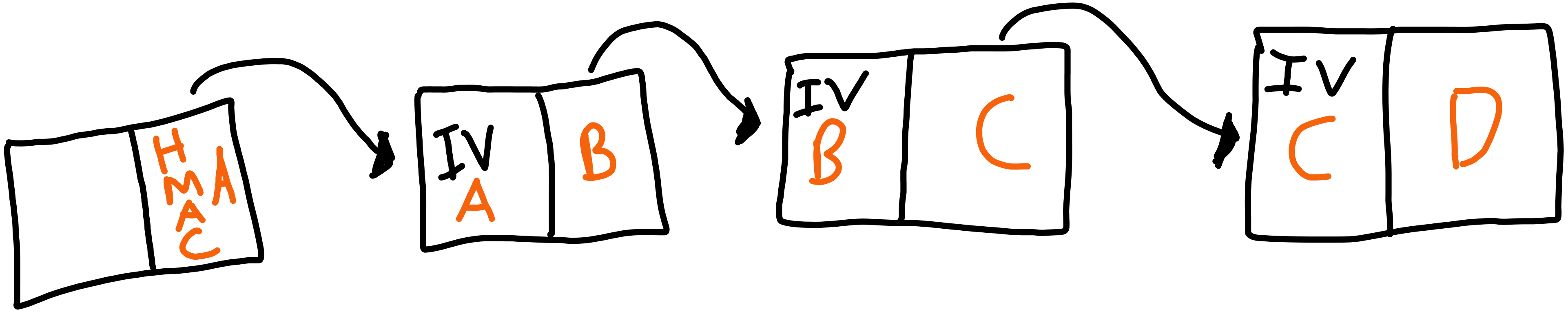




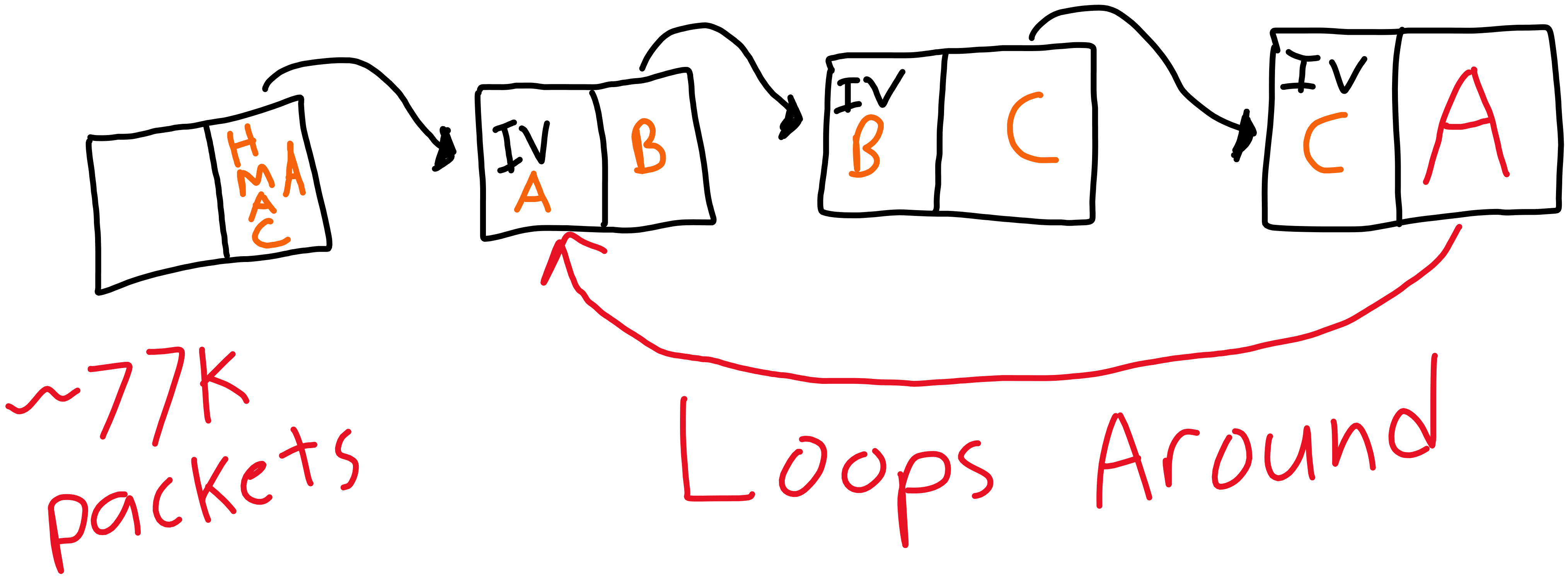
Math



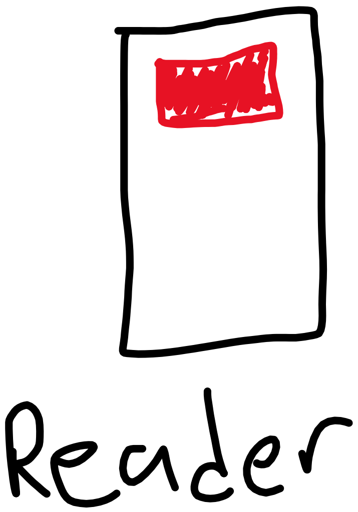
Happy Birthday

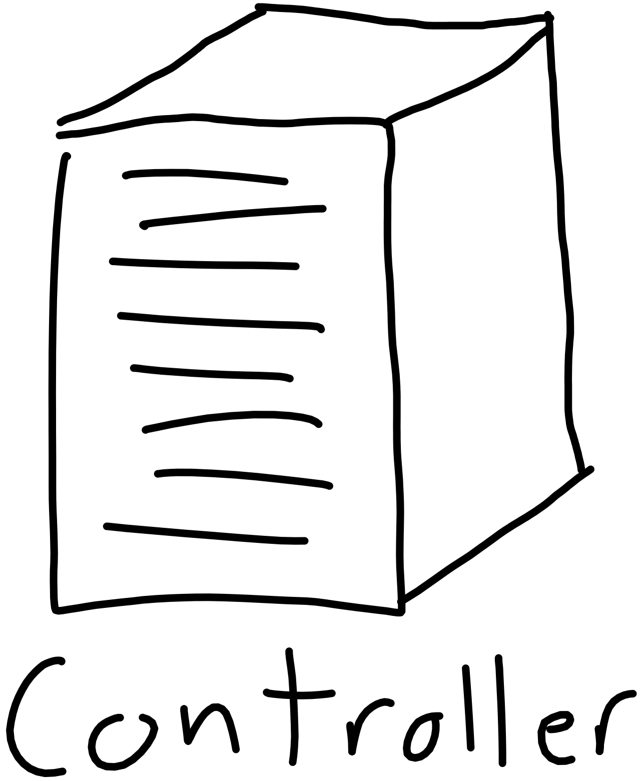


Happy Birthday

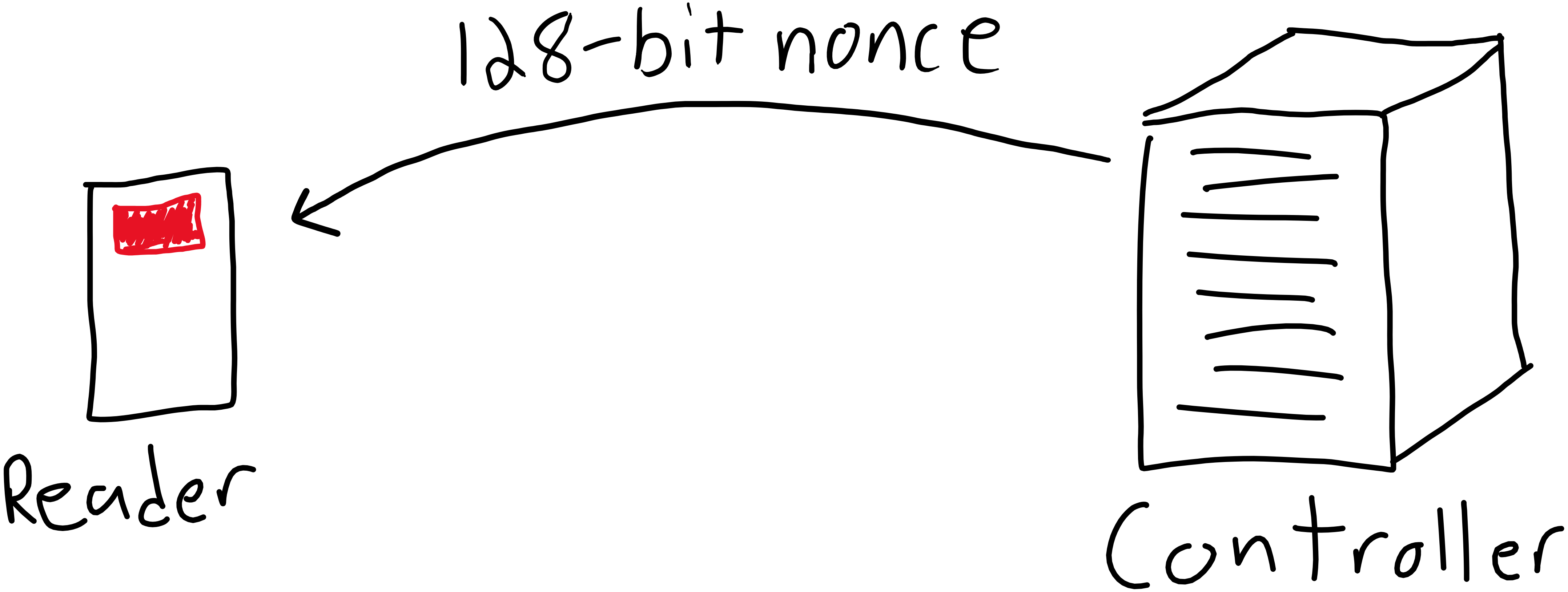


Session Keys

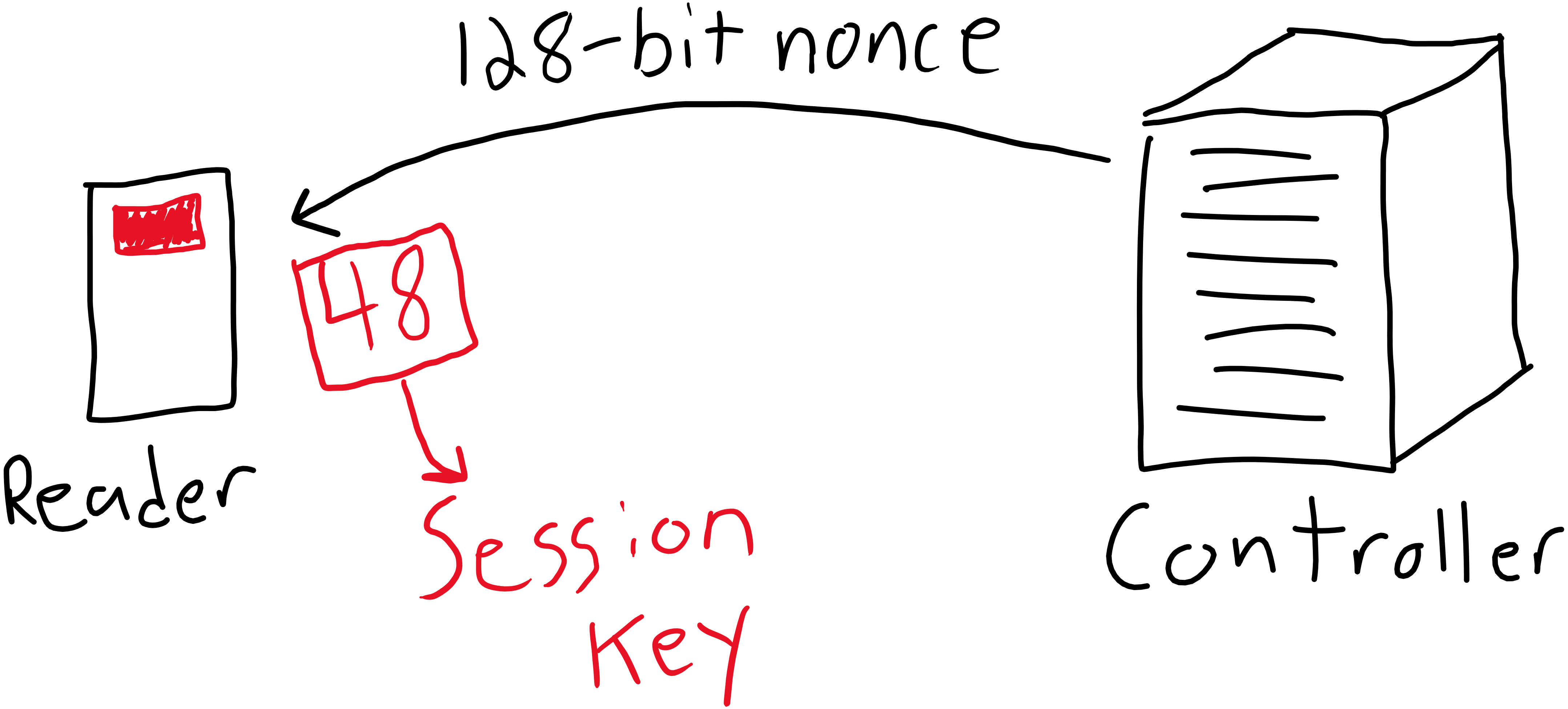




Session Keys



Session Keys

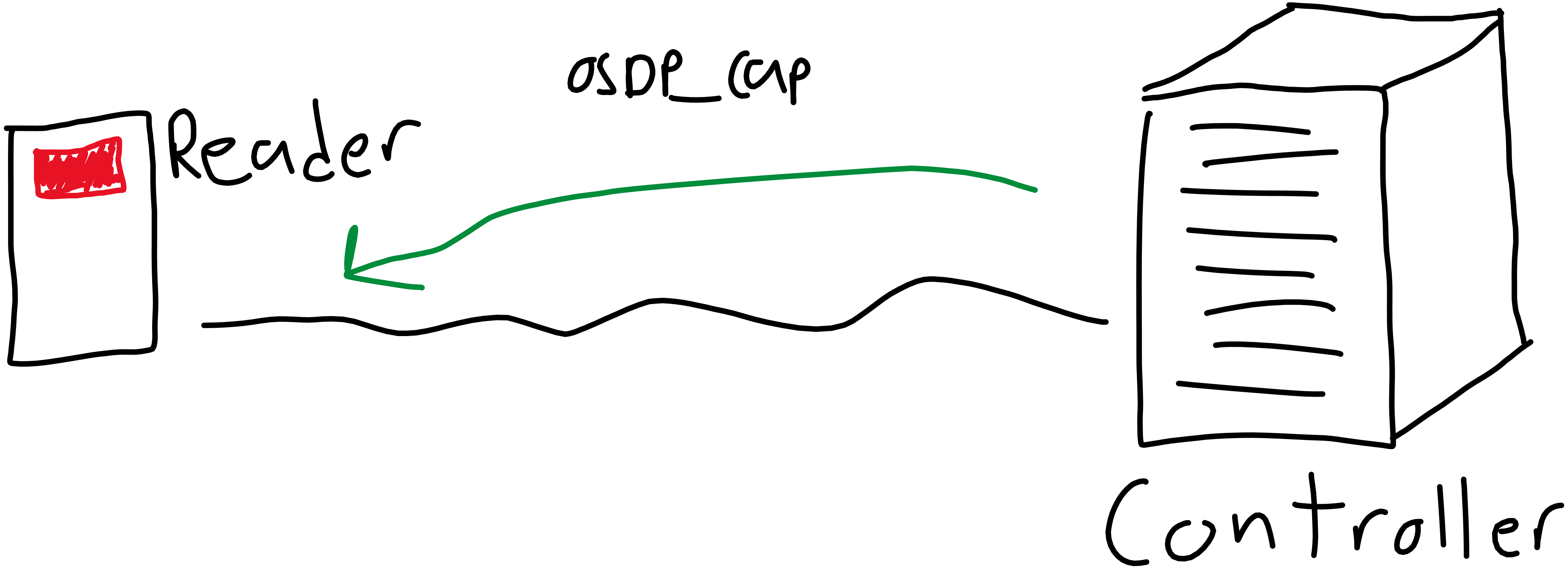


Demo #2

Downgrade

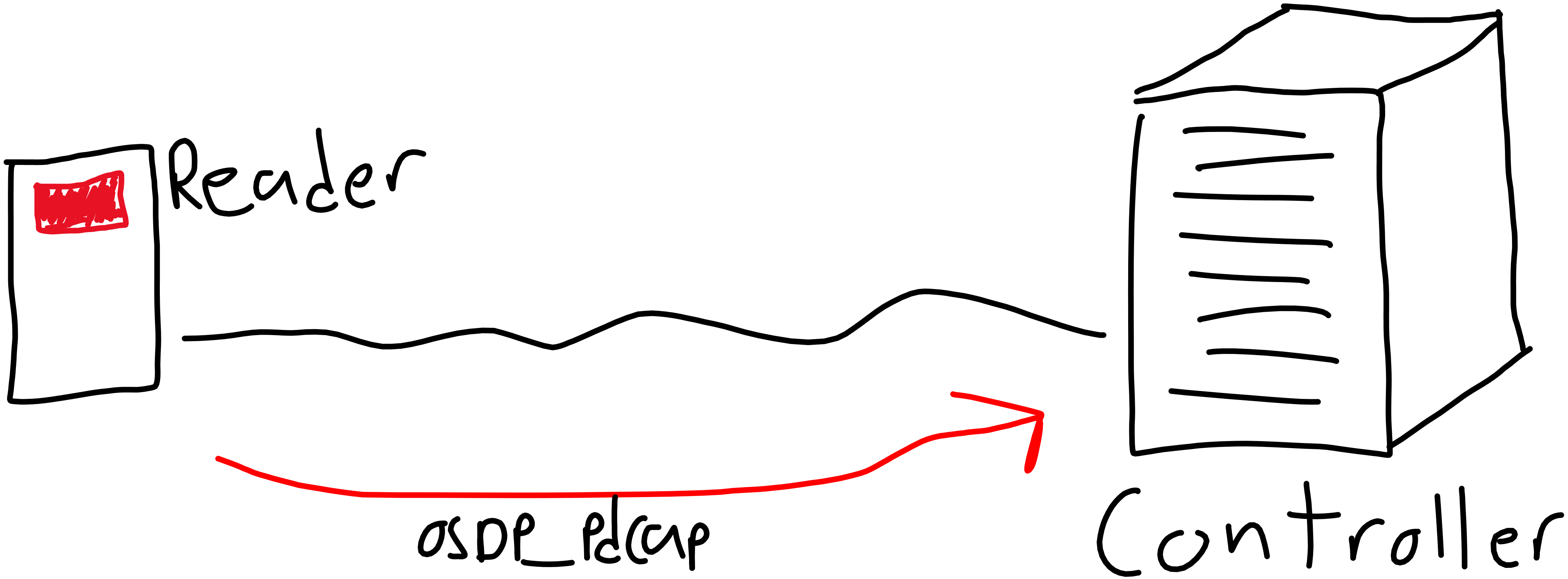
Attack

Me no speak AES



THERE'S ALWAYS ONE...

Me no speak AES



THERE'S ALWAYS ONE...

Me no speak AES

osDP_PDCaP= biometrics

THERE'S ALWAYS ONE...

Me no speak AES

OSDP_PDCaP=KeYPad

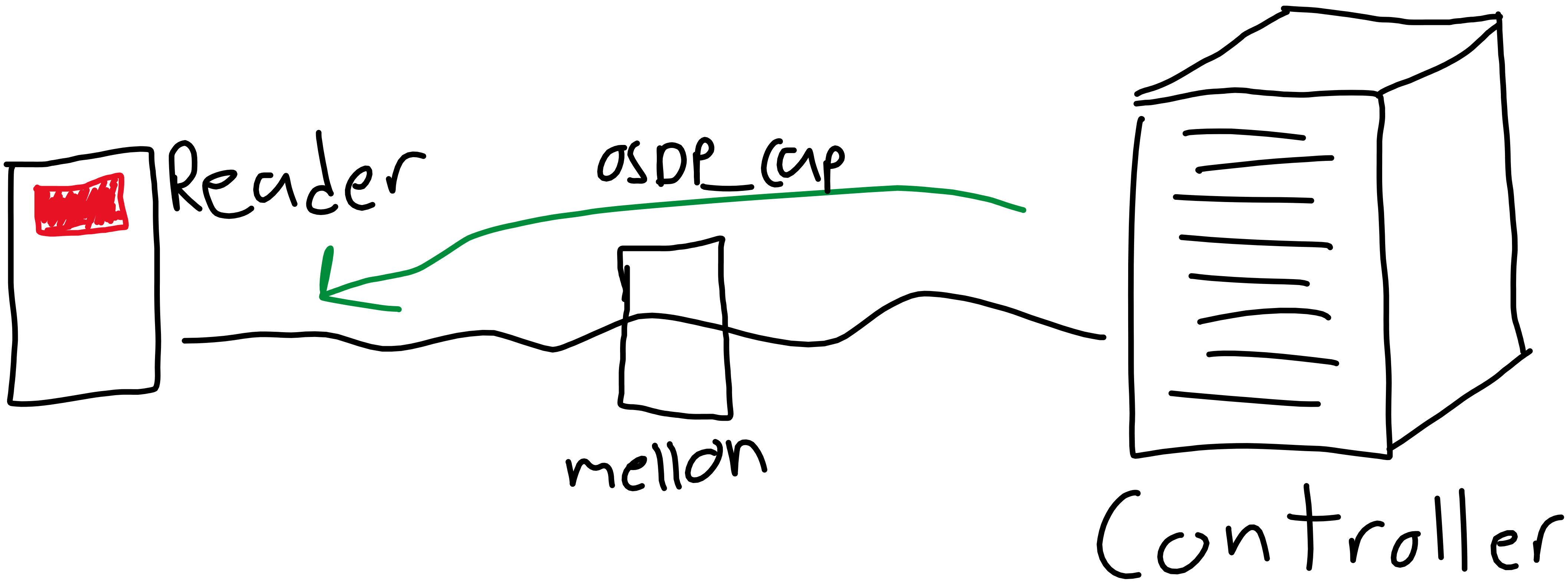
THERE'S ALWAYS ONE...

Me no speak AES

OSDP_PDCAP = communication security

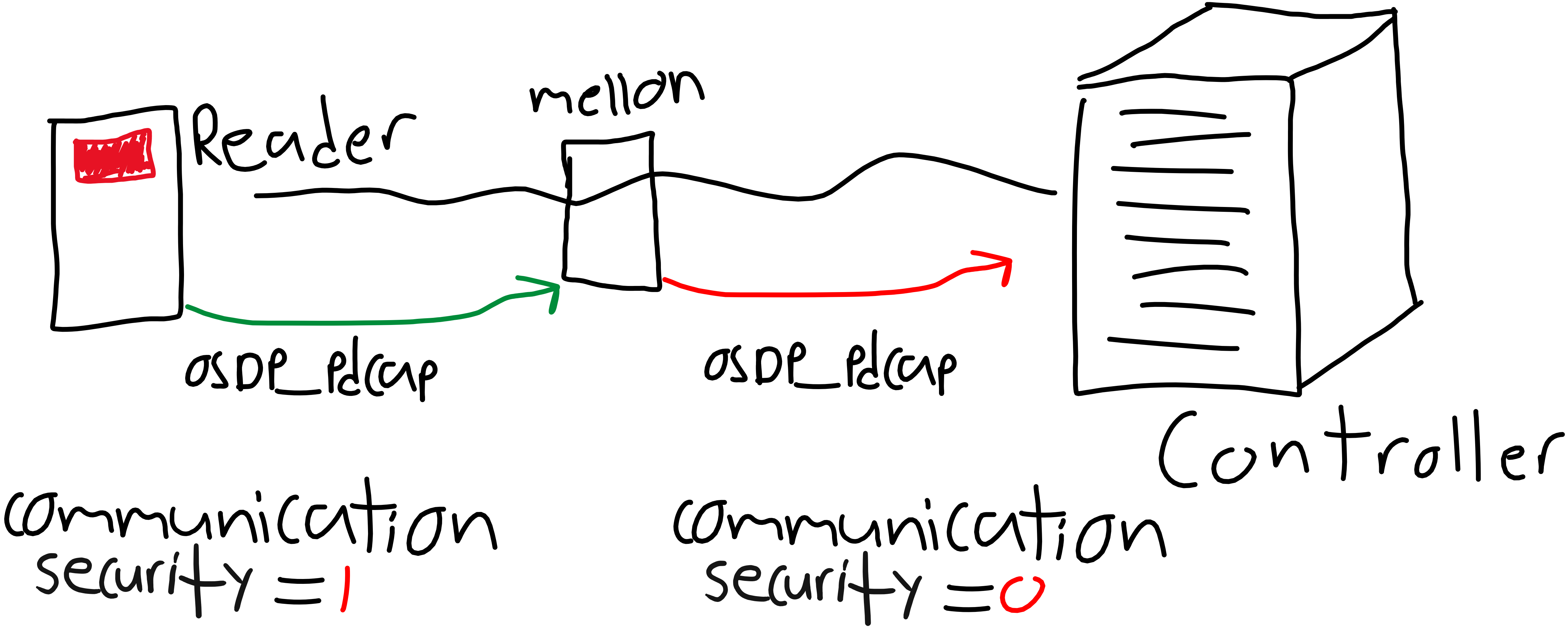
THERE'S ALWAYS ONE...

Me no speak AES



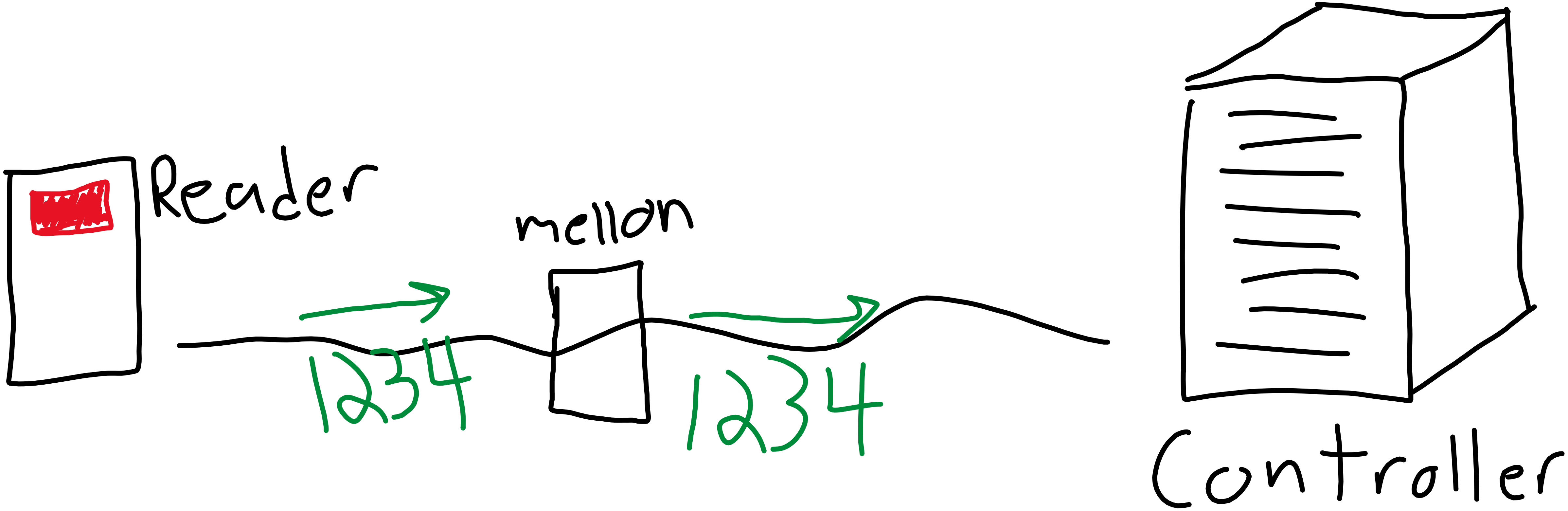
THERE'S ALWAYS ONE...

Me no speak AES



THERE'S ALWAYS ONE...

Me no speak AES



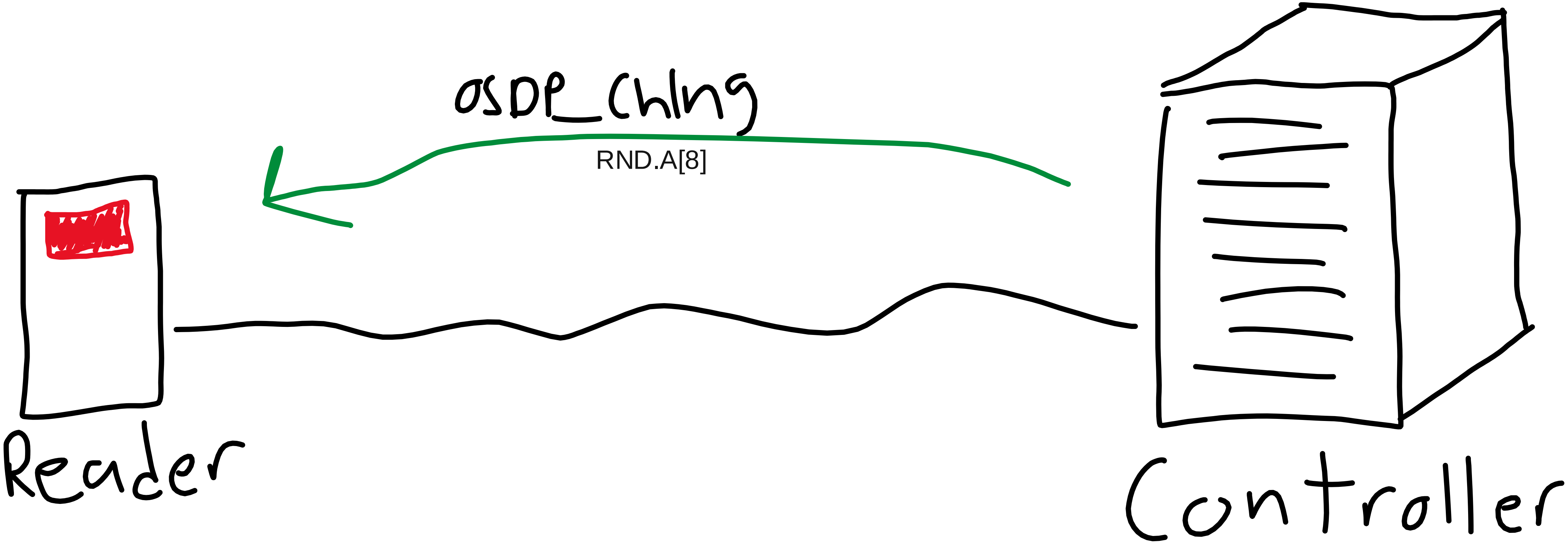
Protocol WTF #2

Stop Making

Null Ciphers

YOU SUPPORT ENCRYPTION...I SUPPORT ENCRYPTION...LETS DO SOMETHING ABOUT THAT

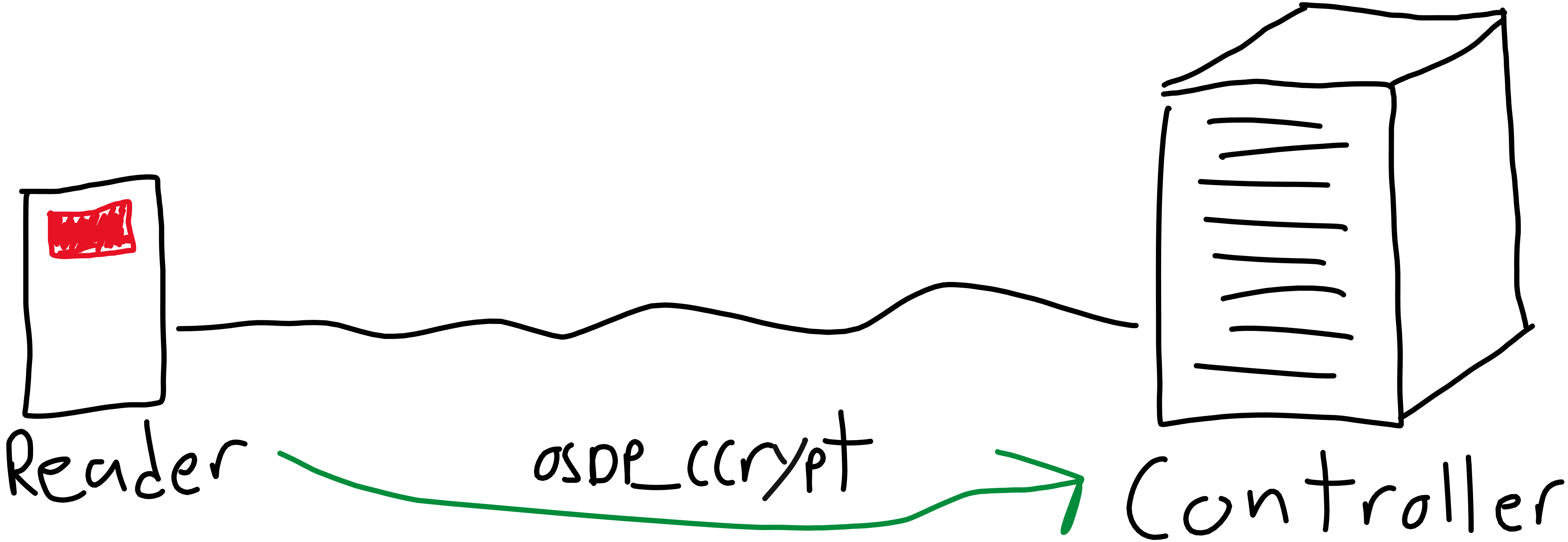
"Secure" Channel - Connection Sequence: SCS_11



RND.A[8]

YEA OK, LETS PARTY.

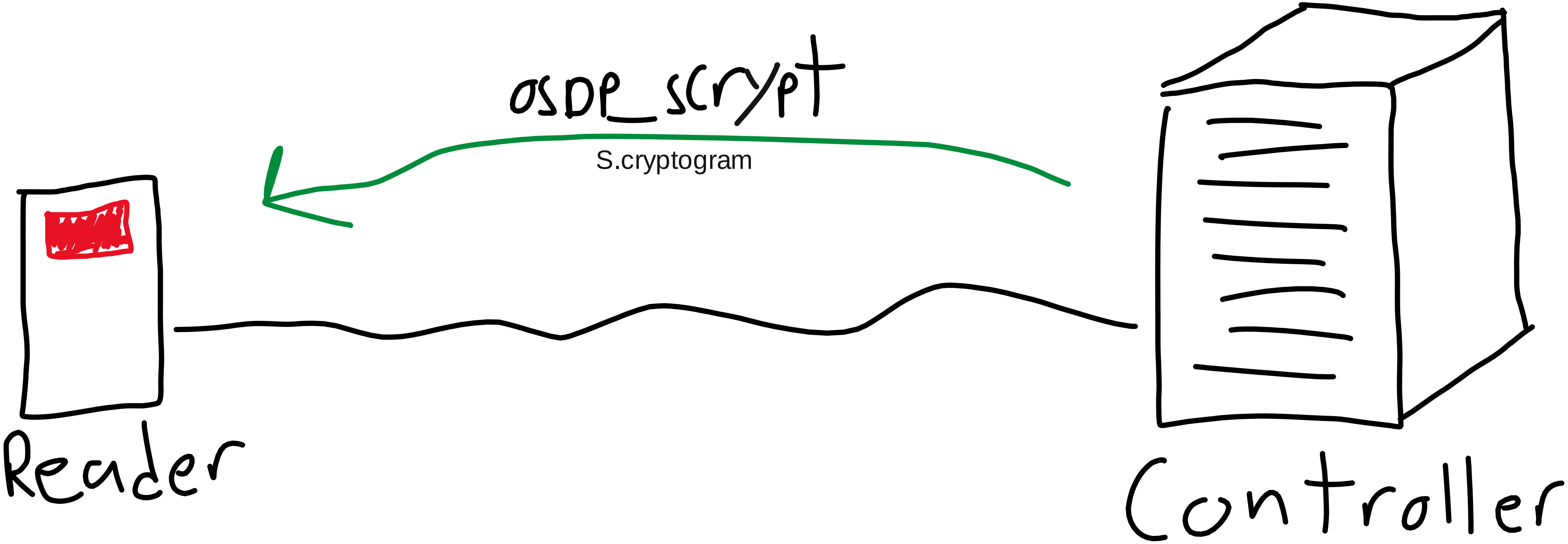
"Secure" Channel - Connection Sequence: SCS_12



RND.B[8]
C.Cryptogram
PD's cUID

YOU SEEM LEGIT

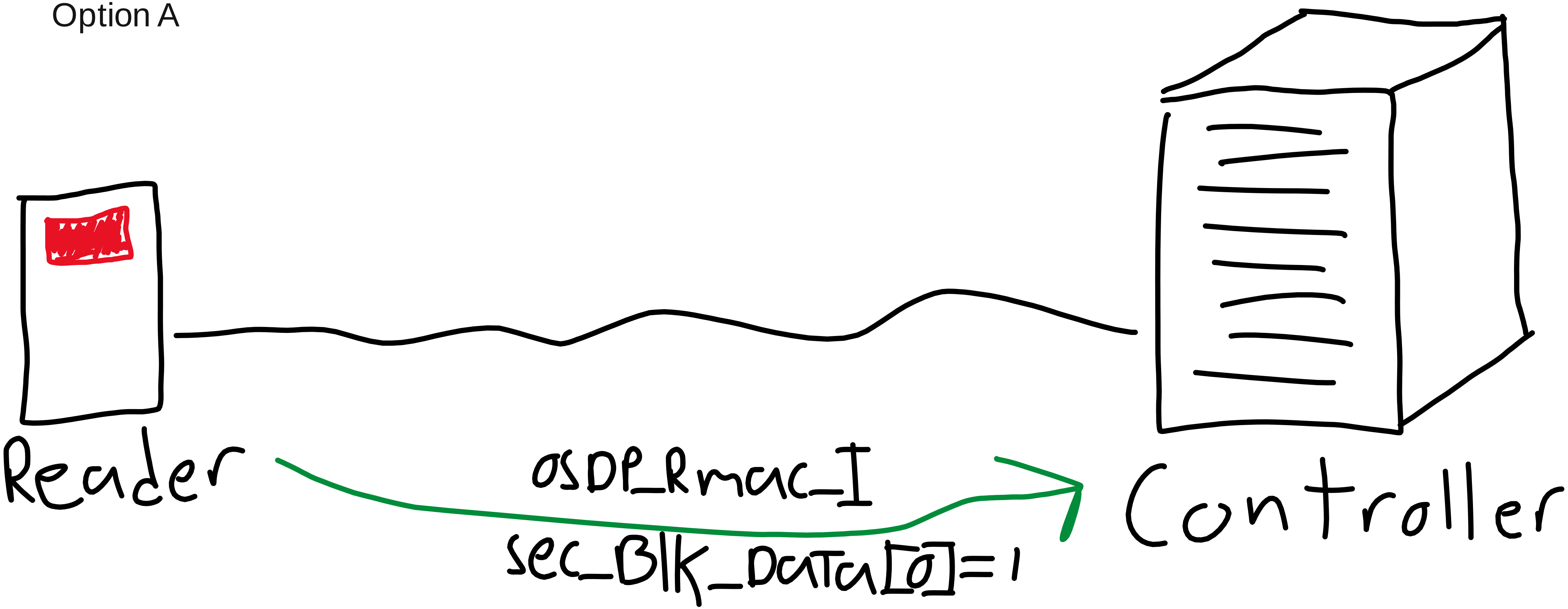
"Secure" Channel - Connection Sequence: SCS_13



YOU SEEM LEGIT AS WELL

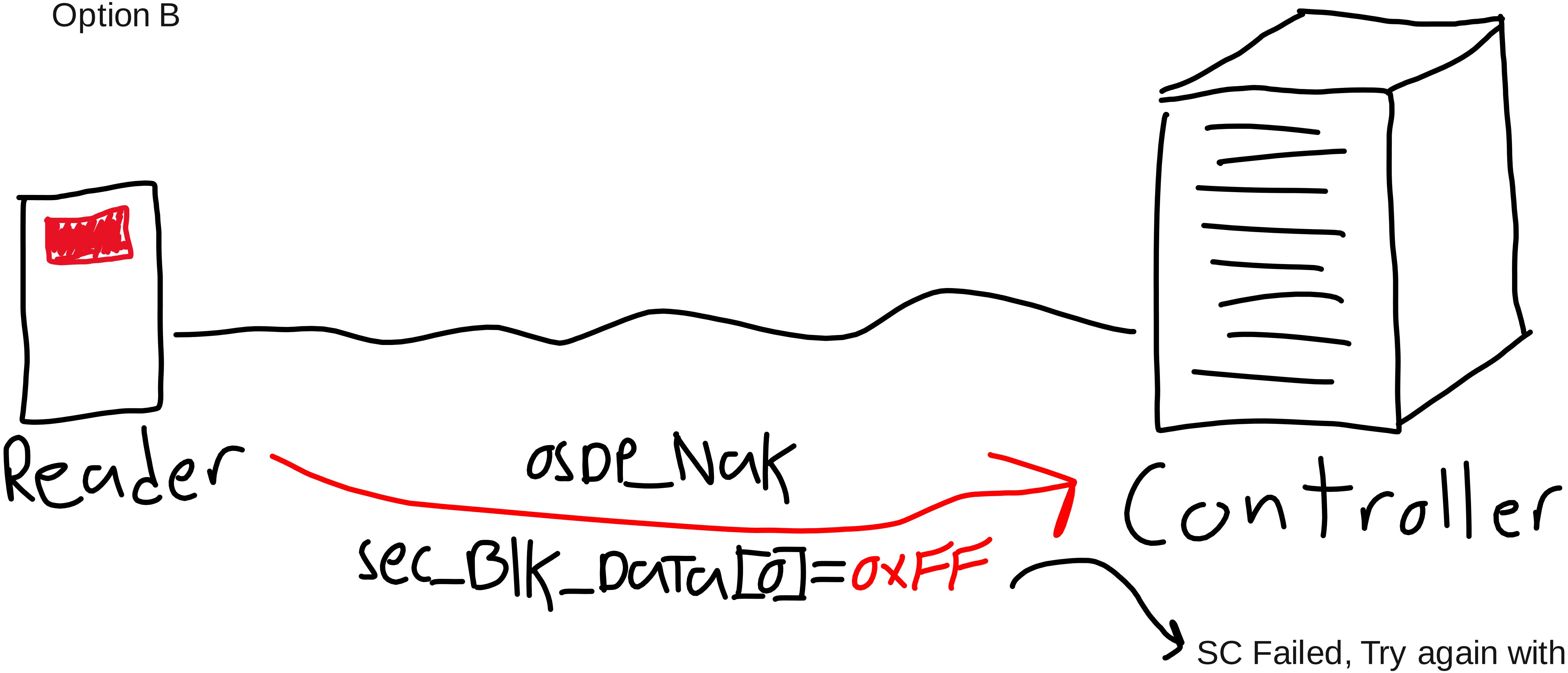
"Secure" Channel - Connection Sequence: SCS_14

Option A



"Secure" Channel - Connection Sequence: SCS_14

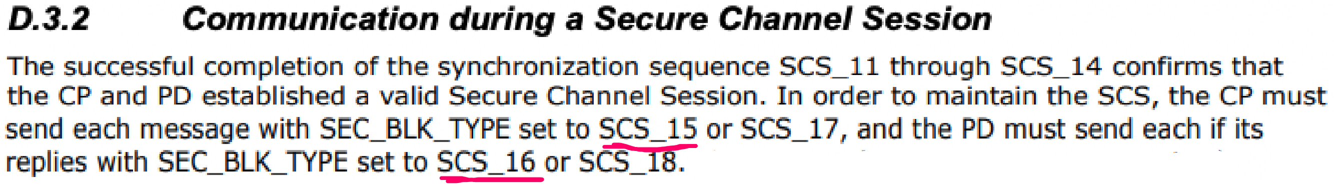
Option B



SC Failed, Try again with SCBK-D

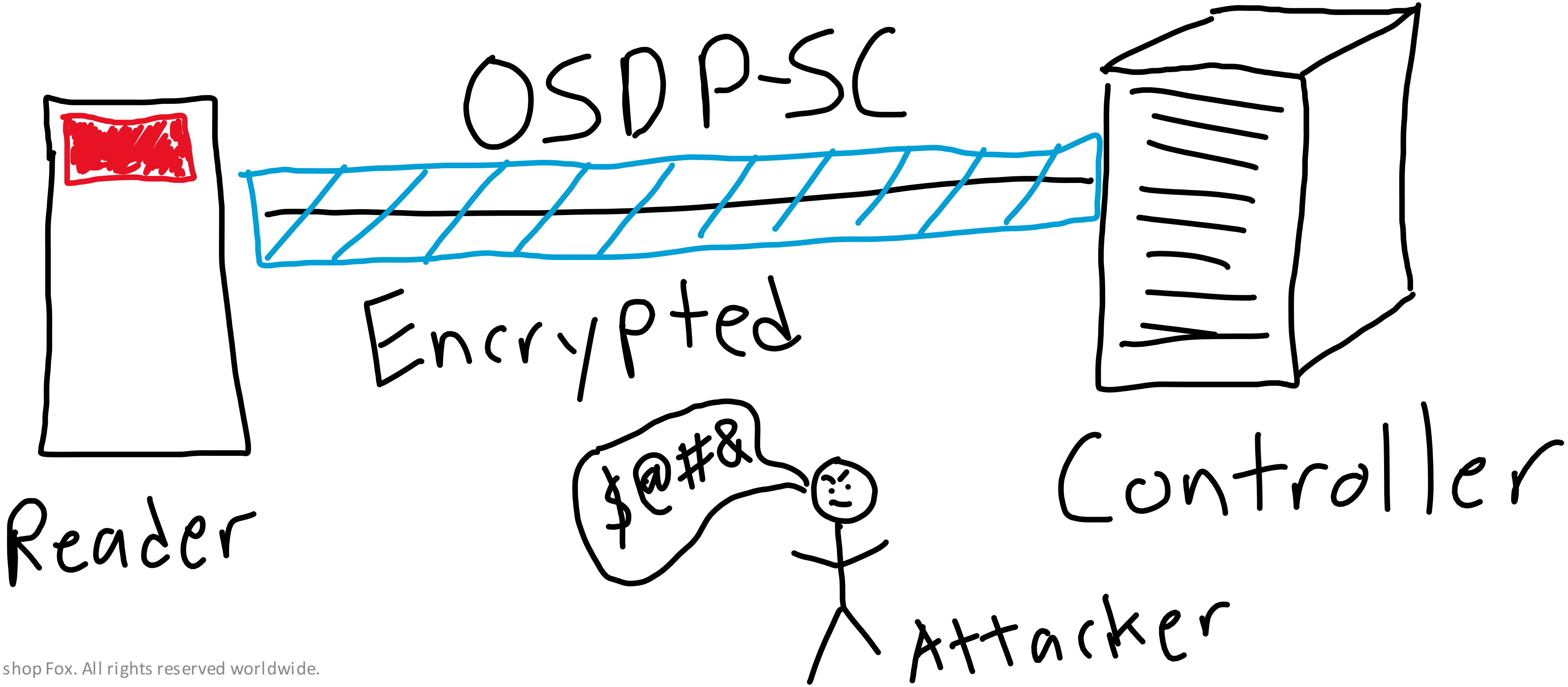
YAY!!!

"Secure" Channel - Established



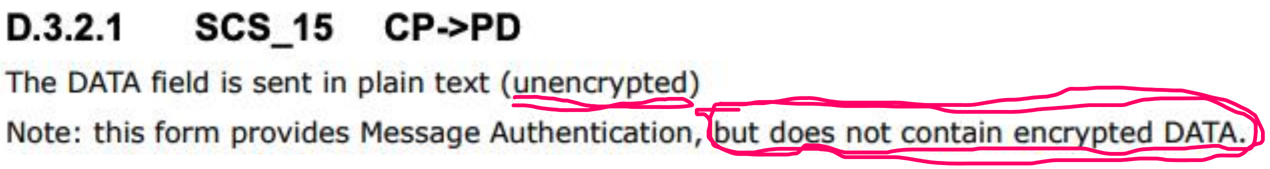
D.3.2 Communication during a Secure Channel Session

The successful completion of the synchronization sequence SCS_11 through SCS_14 confirms that the CP and PD established a valid Secure Channel Session. In order to maintain the SCS, the CP must send each message with SEC_BLK_TYPE set to SCS_15 or SCS_17, and the PD must send each if its replies with SEC_BLK_TYPE set to SCS_16 or SCS_18.



WAIT A SECOND

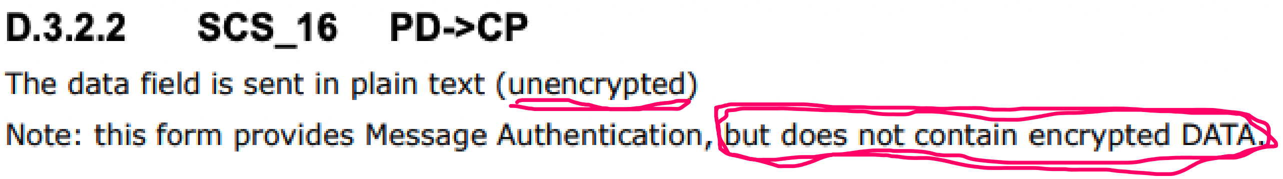
"Secure" Channel - Connection Sequence: SCS_15 & SCS_16



D.3.2.1 SCS_15 CP->PD

The DATA field is sent in plain text (unencrypted)

Note: this form provides Message Authentication, but does not contain encrypted DATA.

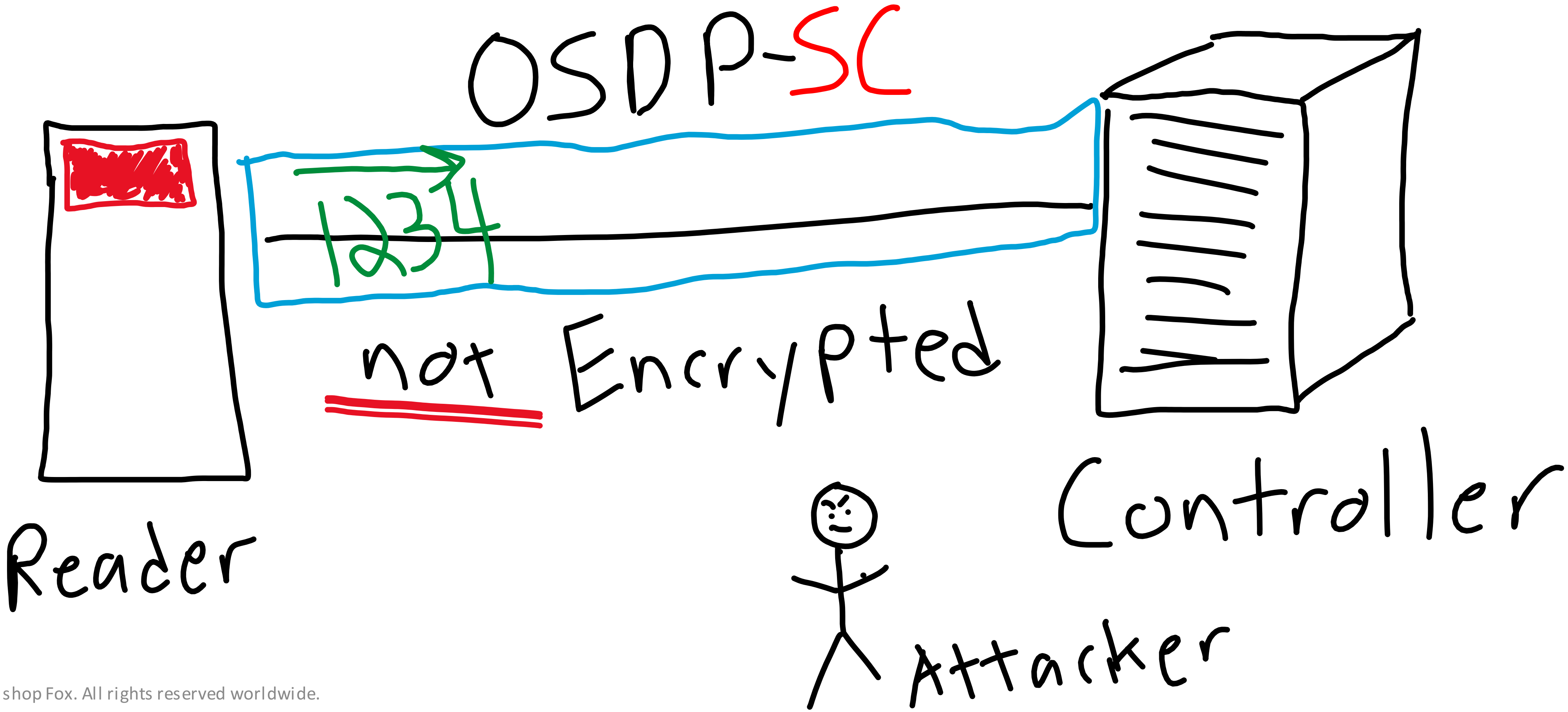


D.3.2.2 SCS_16 PD->CP

The data field is sent in plain text (unencrypted)

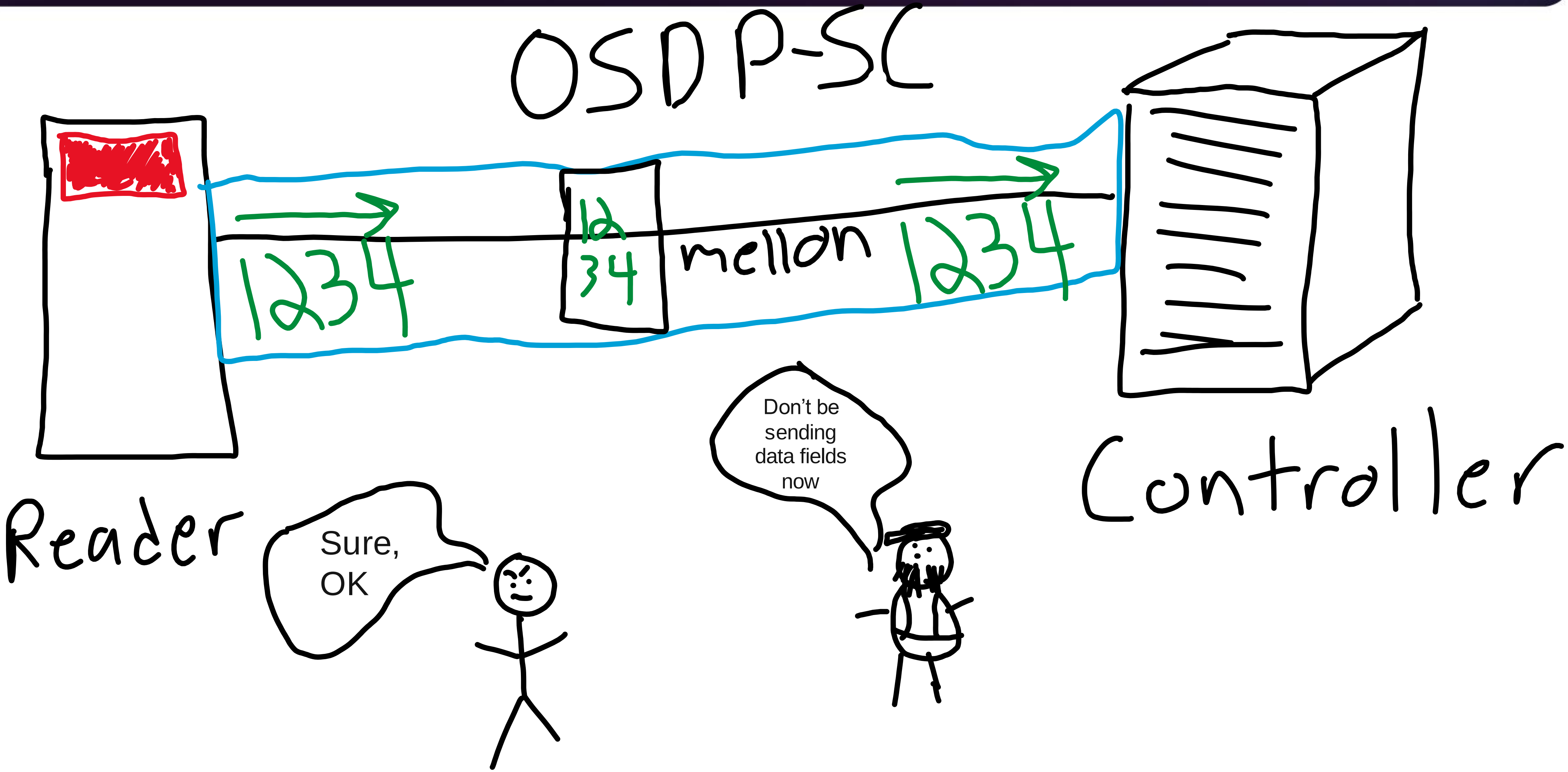
Note: this form provides Message Authentication, but does not contain encrypted DATA.

"Secure" Channel - Established



WHYYYYYYY

"Secure" Channel - SCS_15 & SCS_16



Demo #3

Install-mode

Attack

Install-Mode

SSH Security Model

Install-Mode

SSH Security Model

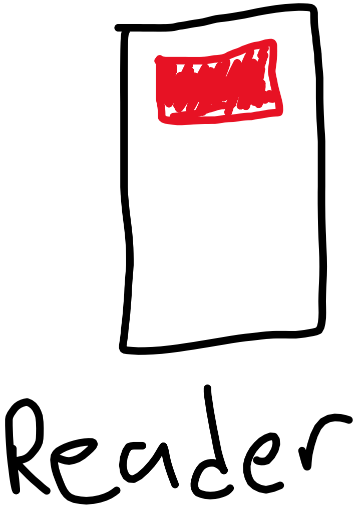
One-time Insecure Setup

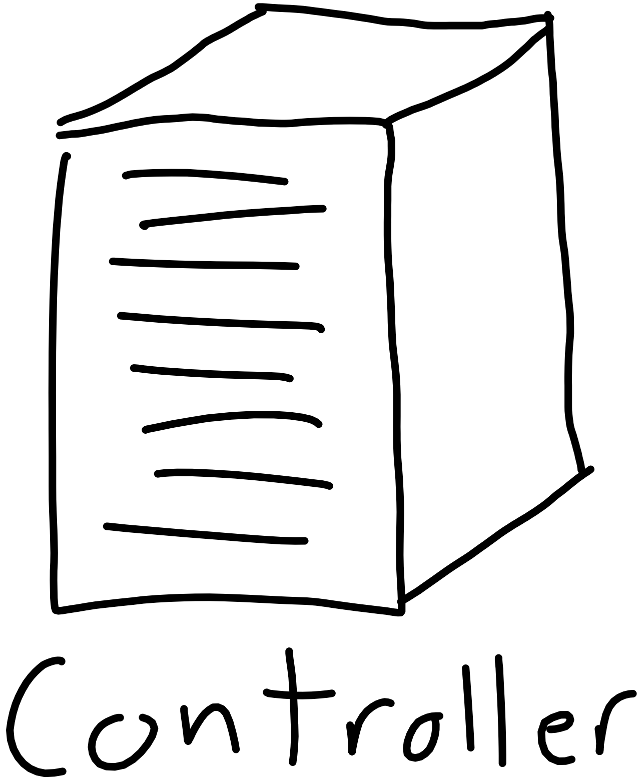
Install-Mode

SSH Security Model

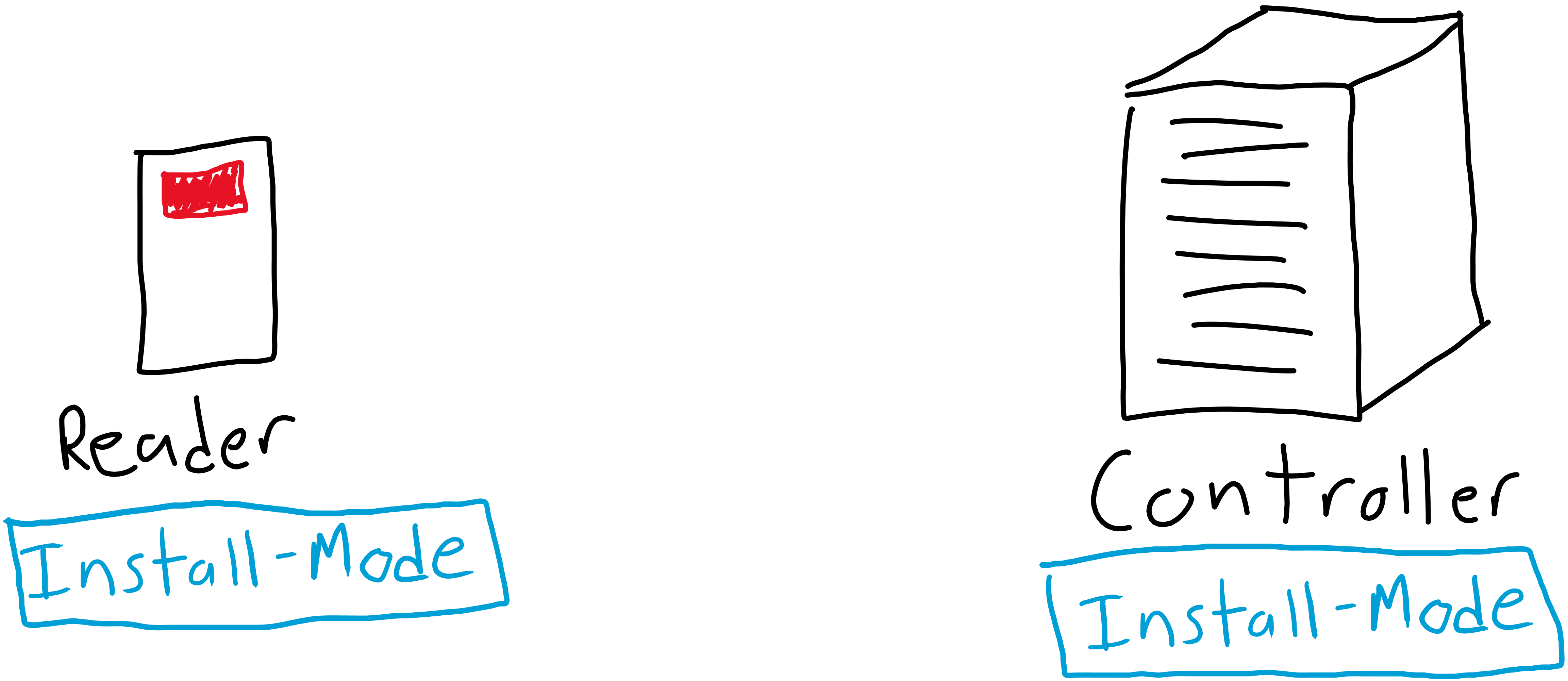
One-time Insecure Setup

Install-Mode

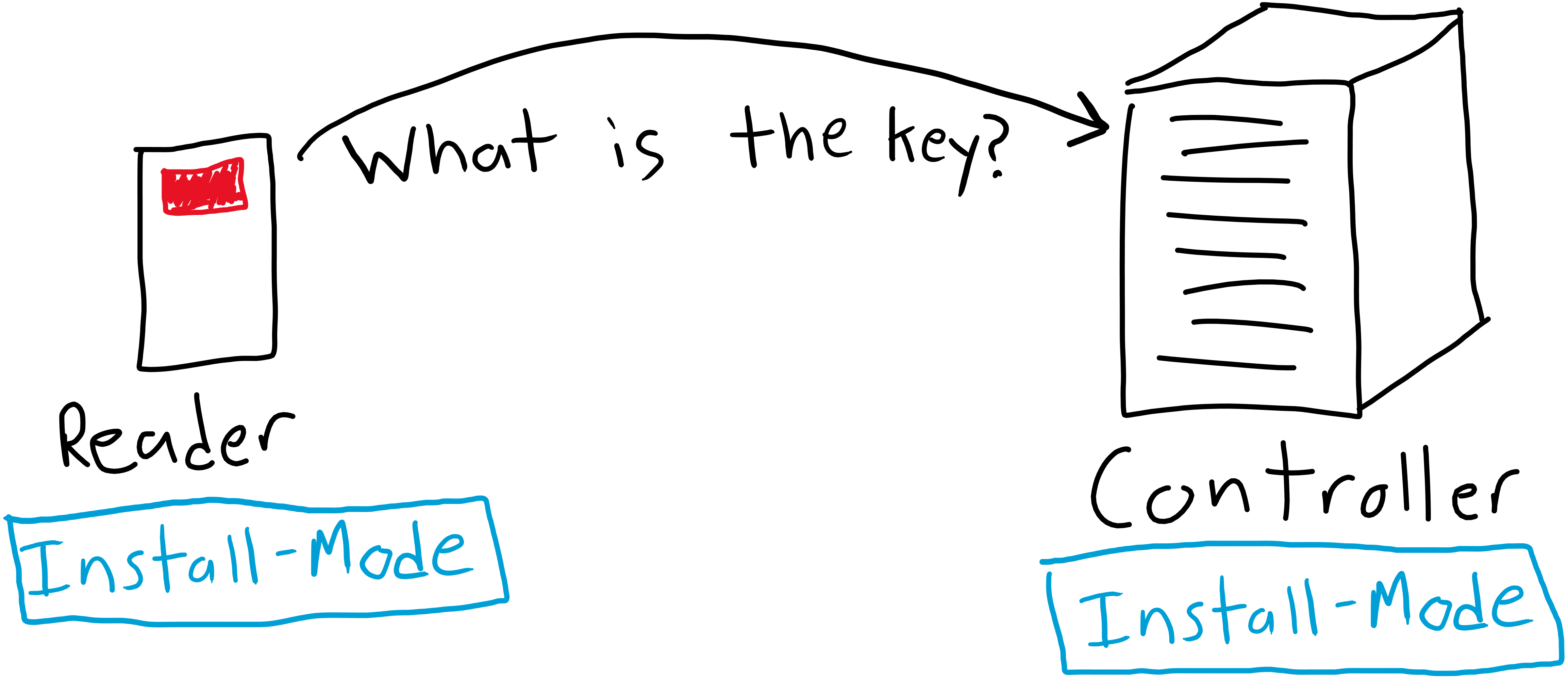




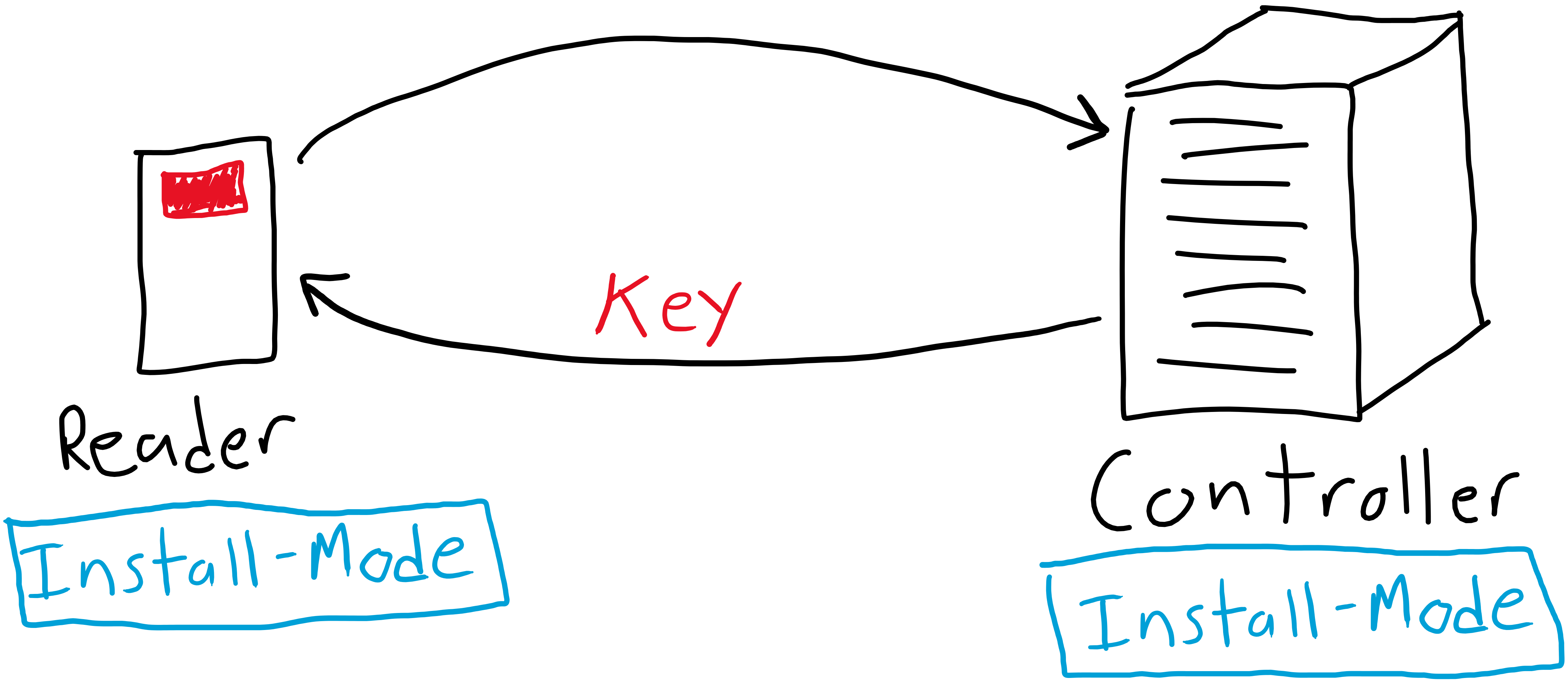
Install-Mode



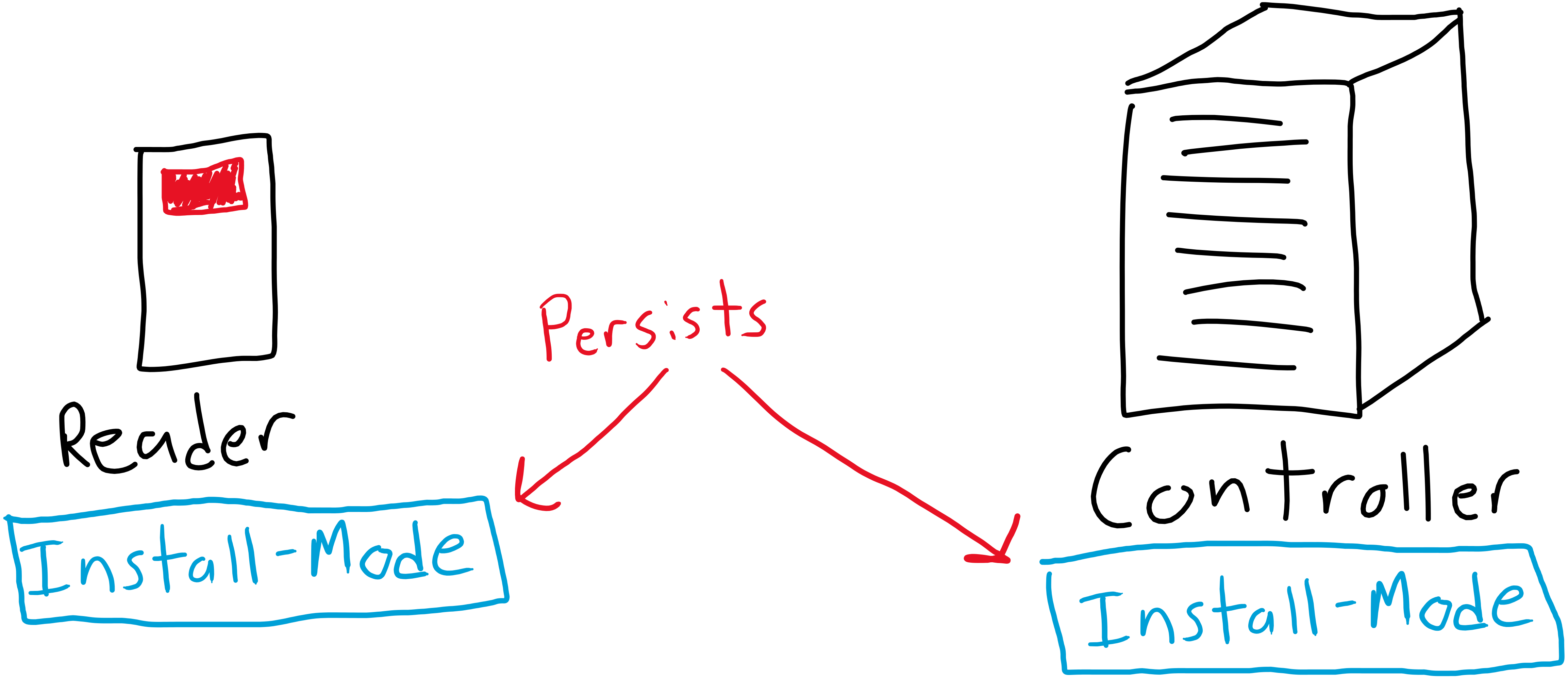
Install-Mode



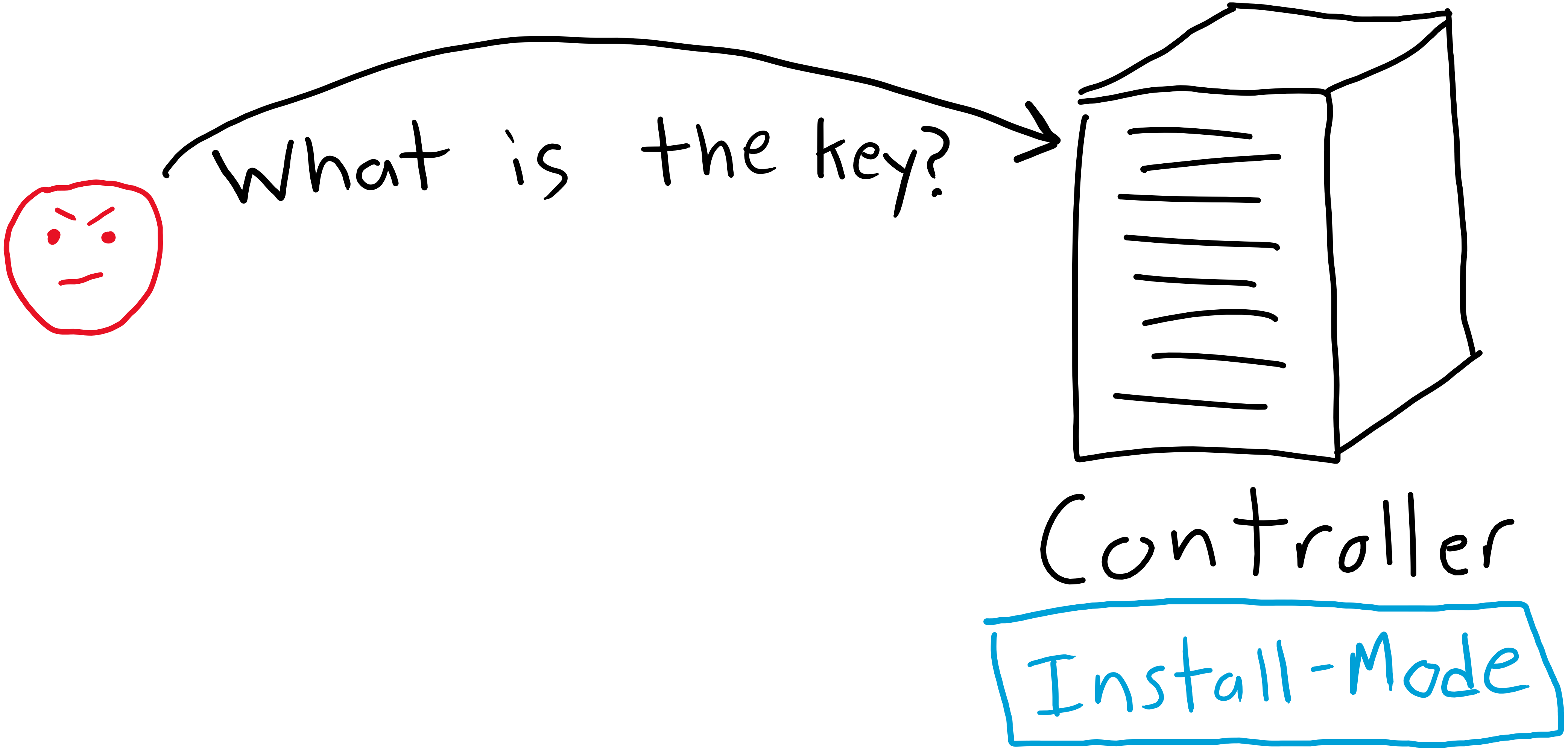
Install-Mode



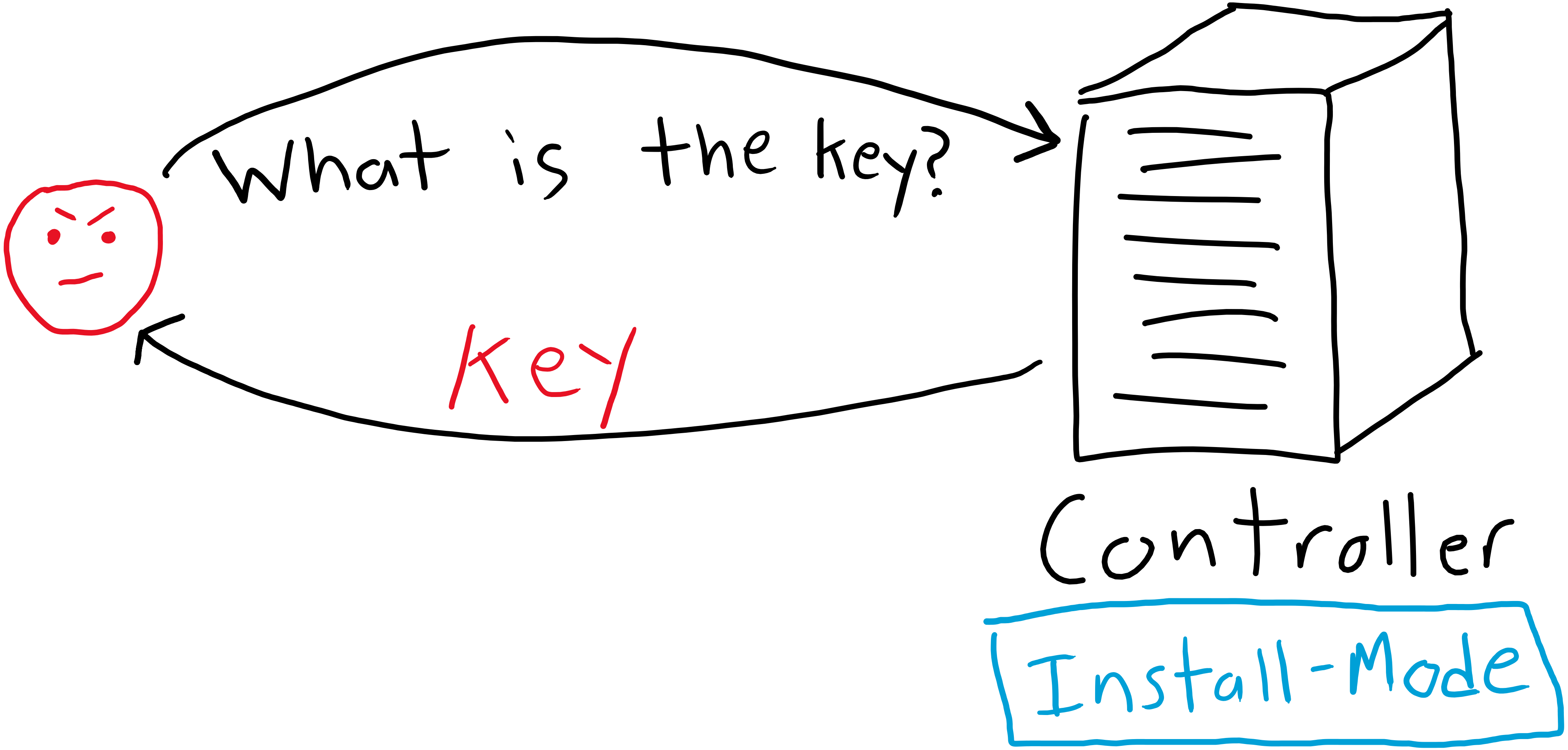
Install-Mode



Install-Mode



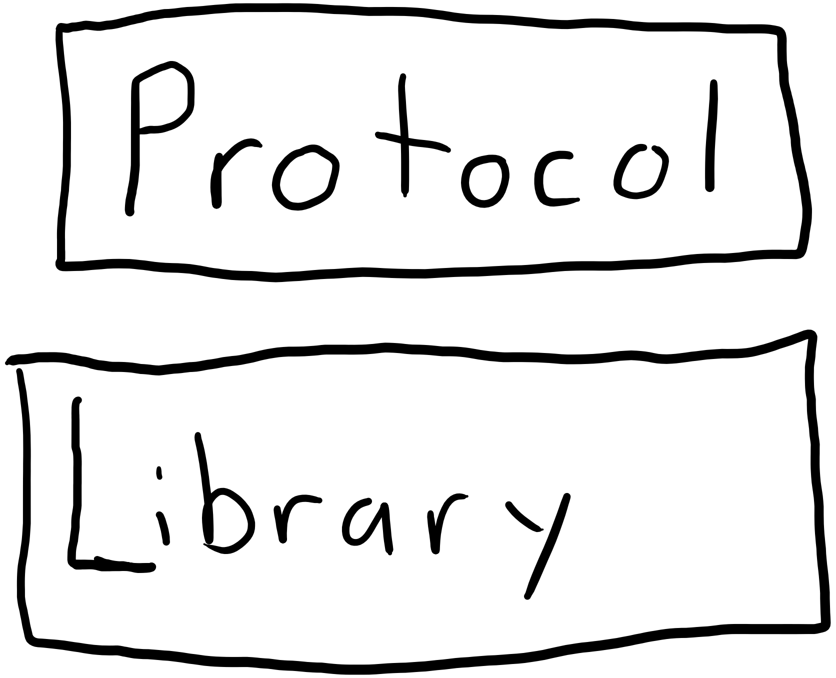
Install-Mode



Where does the vuln lie?

Protocol

Where does the vuln lie?



Where does the vuln lie?

Protocol

Library

Config

Where does the vuln lie?

Protocol

Documentation

Library

Config

Where does the vuln lie?

Protocol

Documentation

Library

Implementation

Config

Where does the vuln lie?

Protocol

Library

Config

Documentation

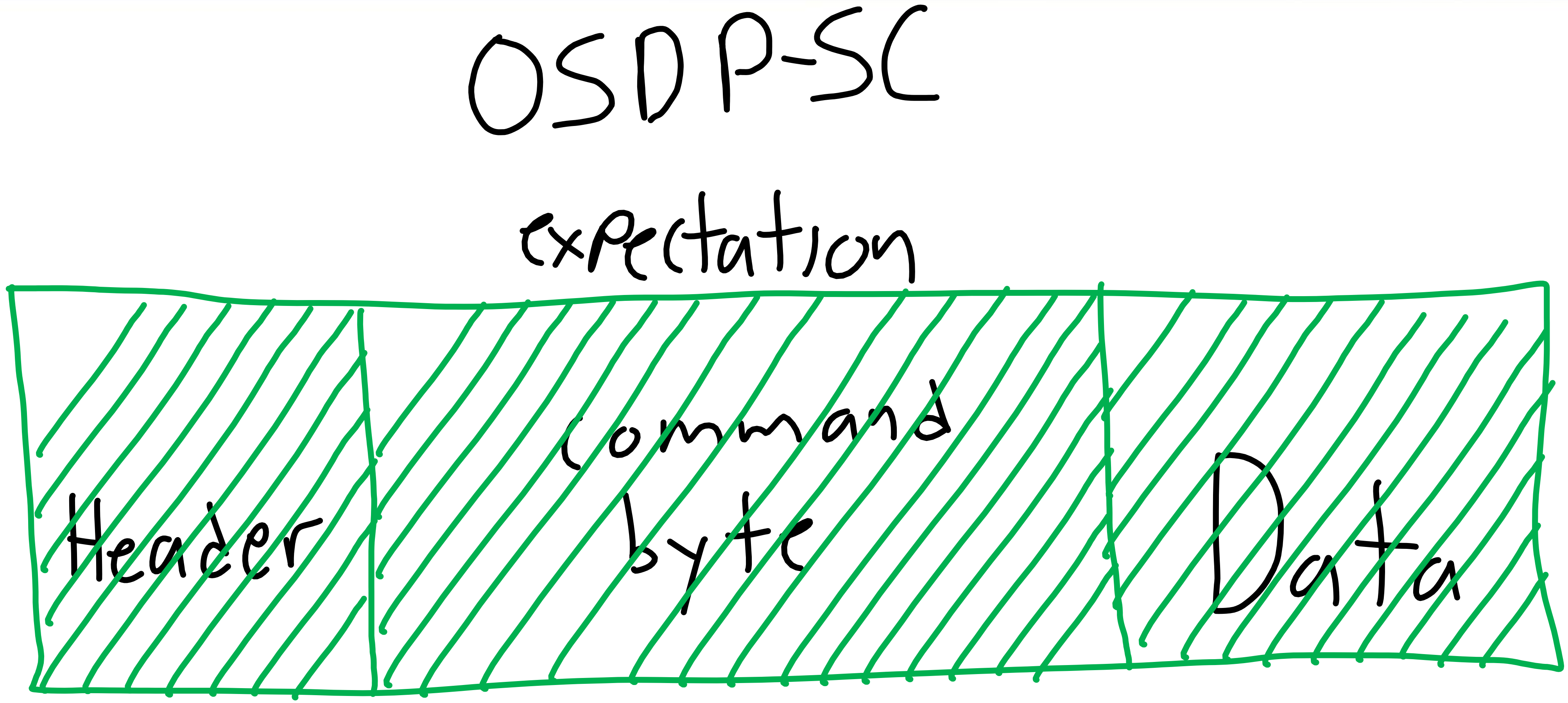
Implementation

Marketing

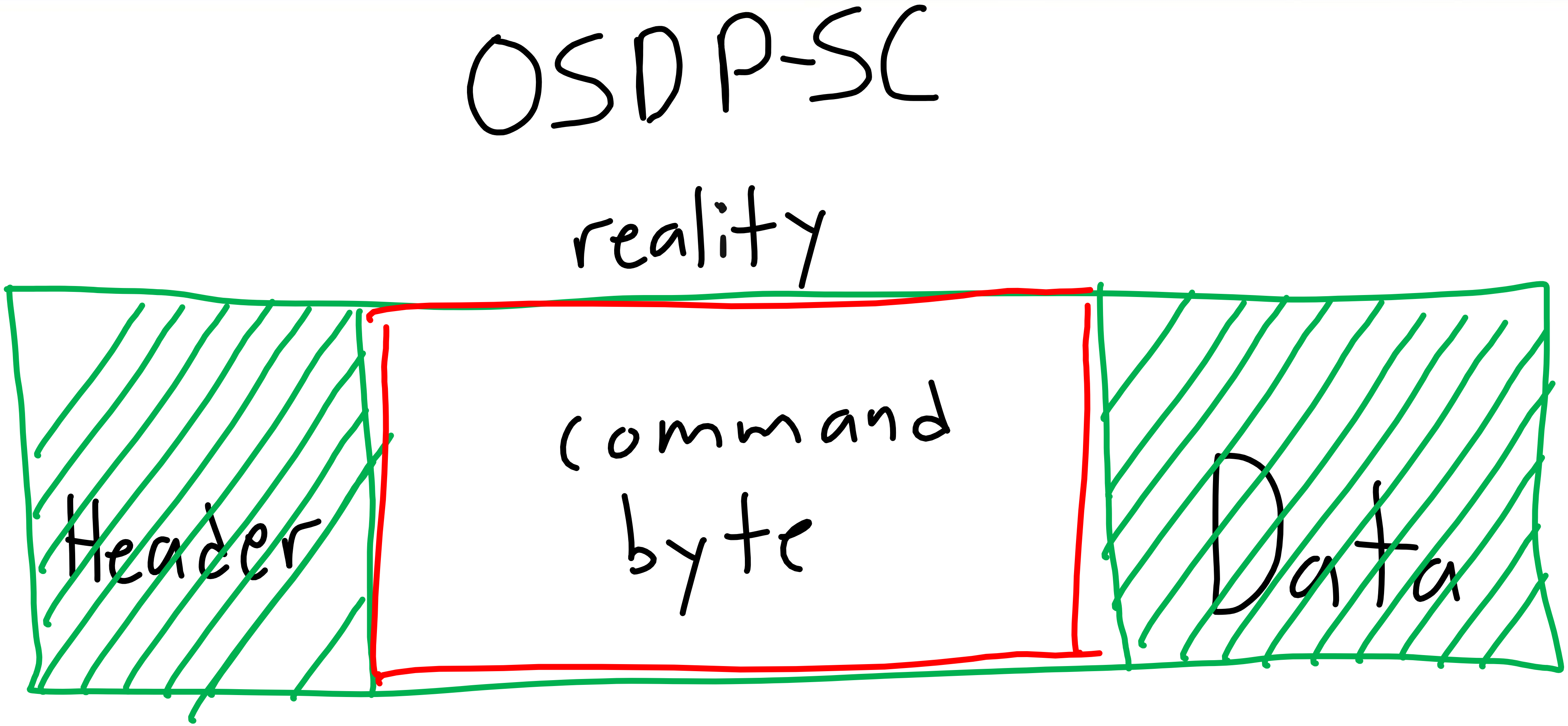
Protocol WTF #3

We never said we'd encrypt ALL of the data...

SCS_17& SCS_18: The whole packet is encrypted...right?



SCS_17& SCS_18: The whole packet is encrypted...right?



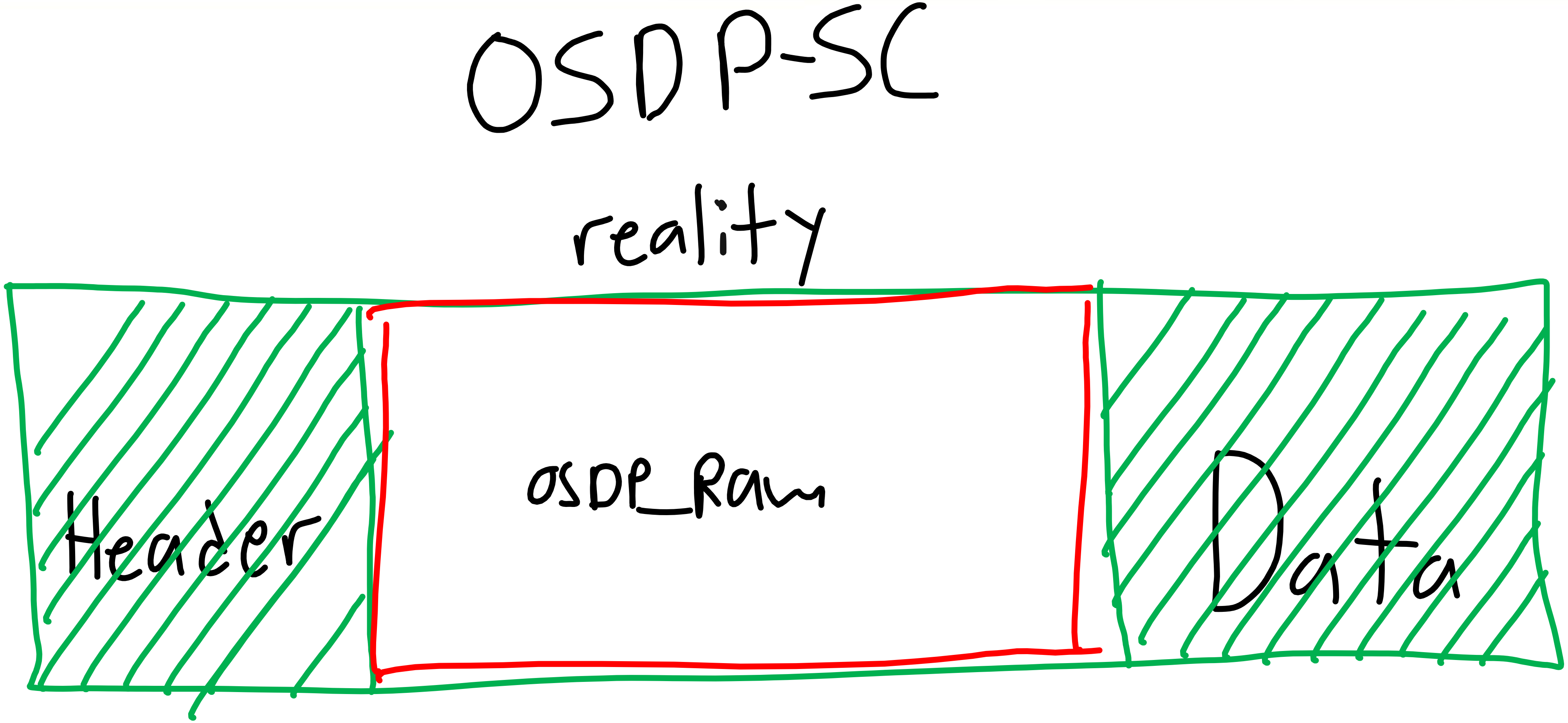
PROTOCOL WTF #…?

SCS_17& SCS_18: The whole packet is encrypted...right?

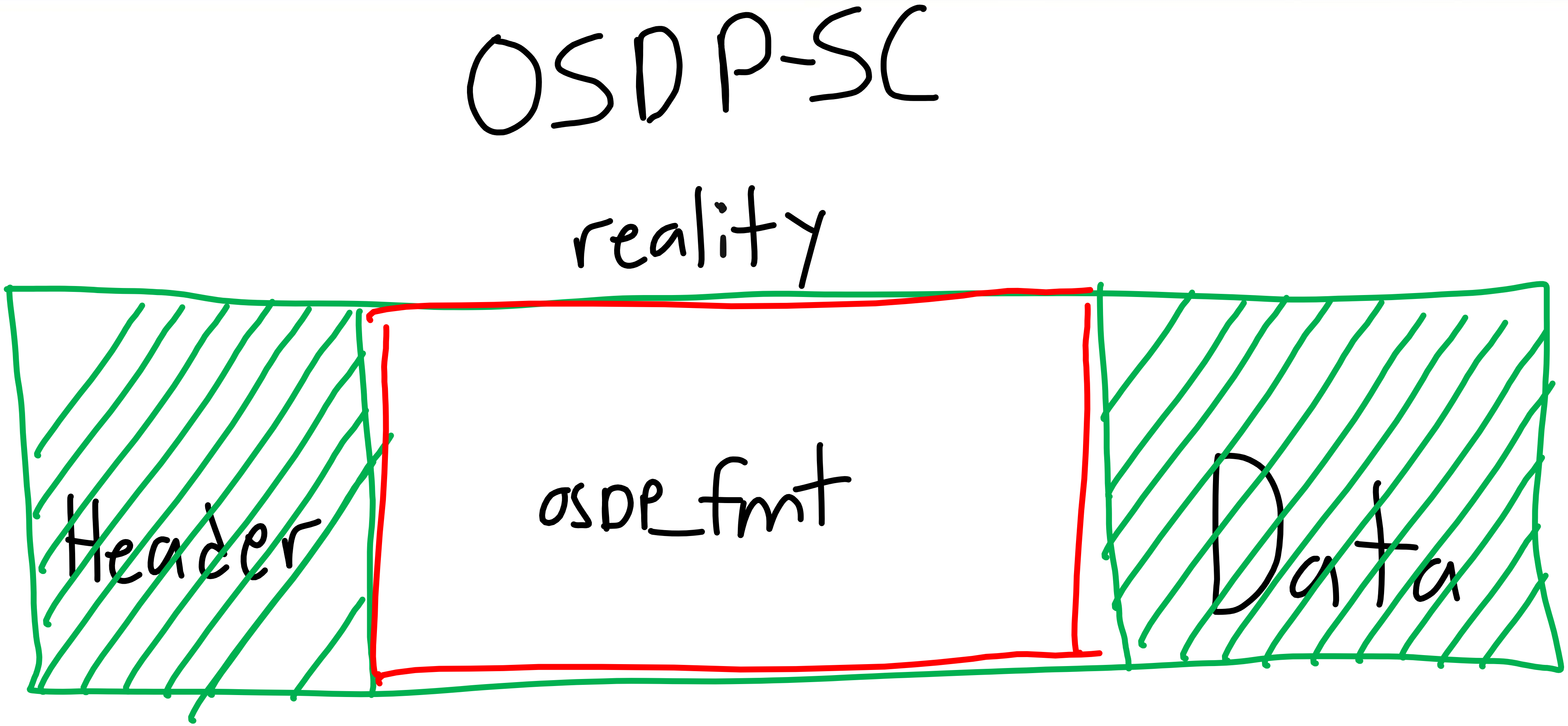
command byte = Pd/CP data transmitted

PROTOCOL WTF #…?

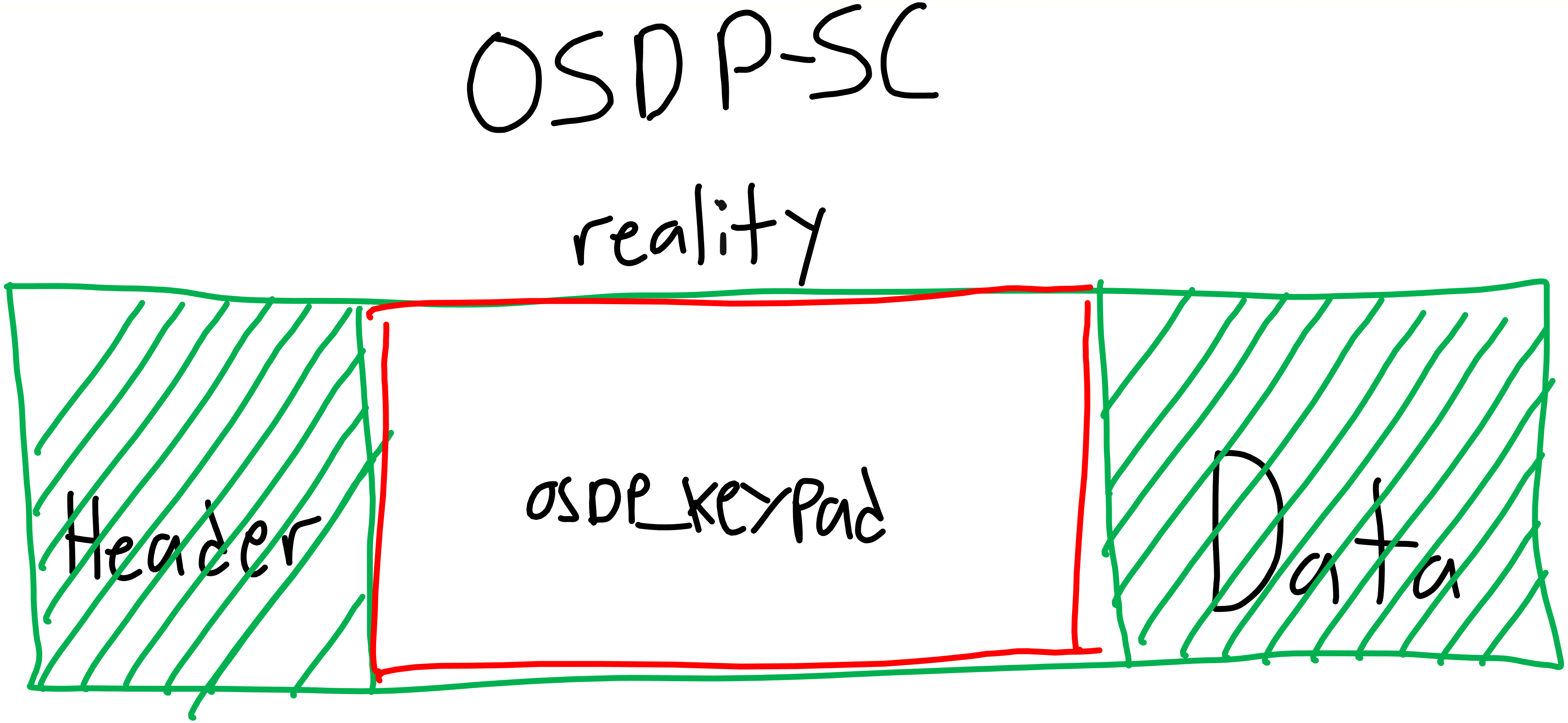
osdp_RAW



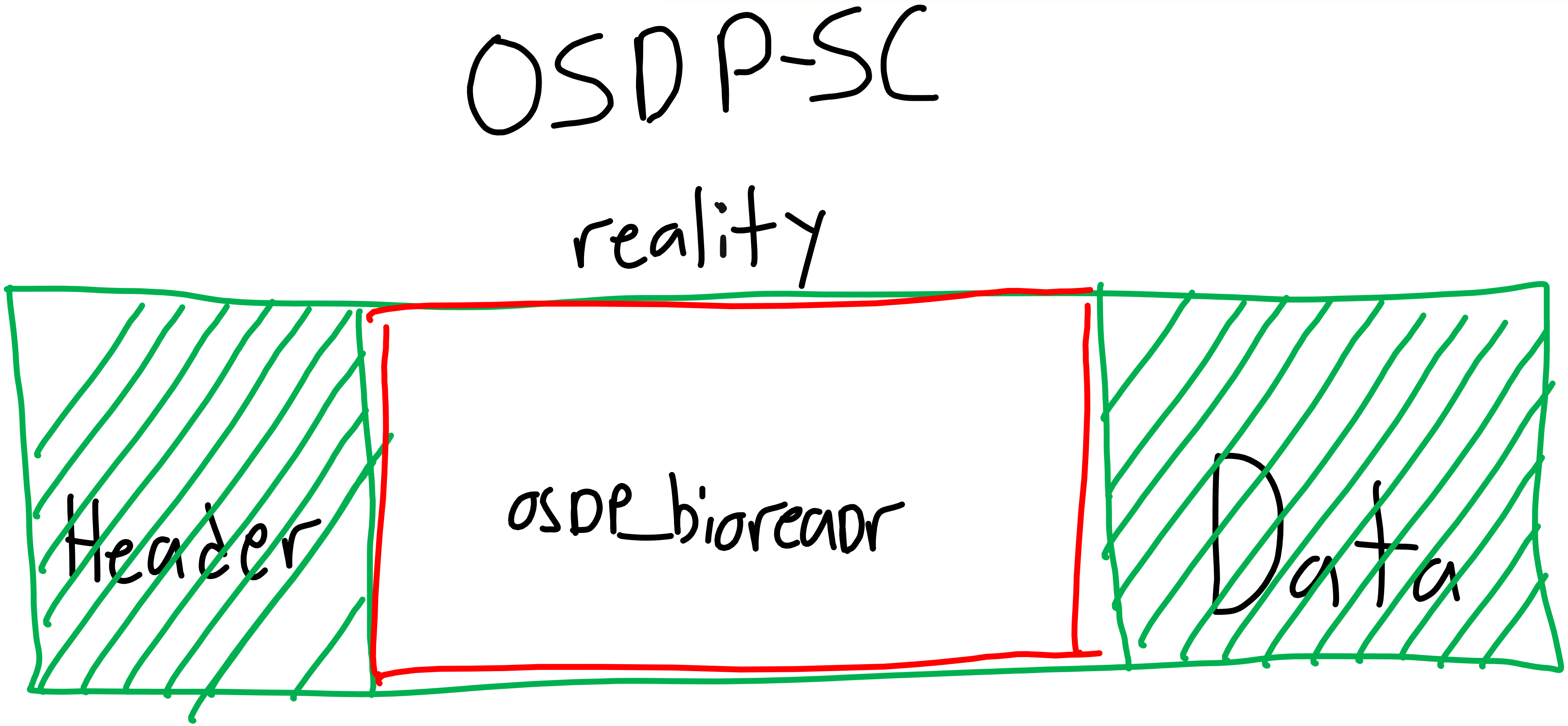
osdp_FMT



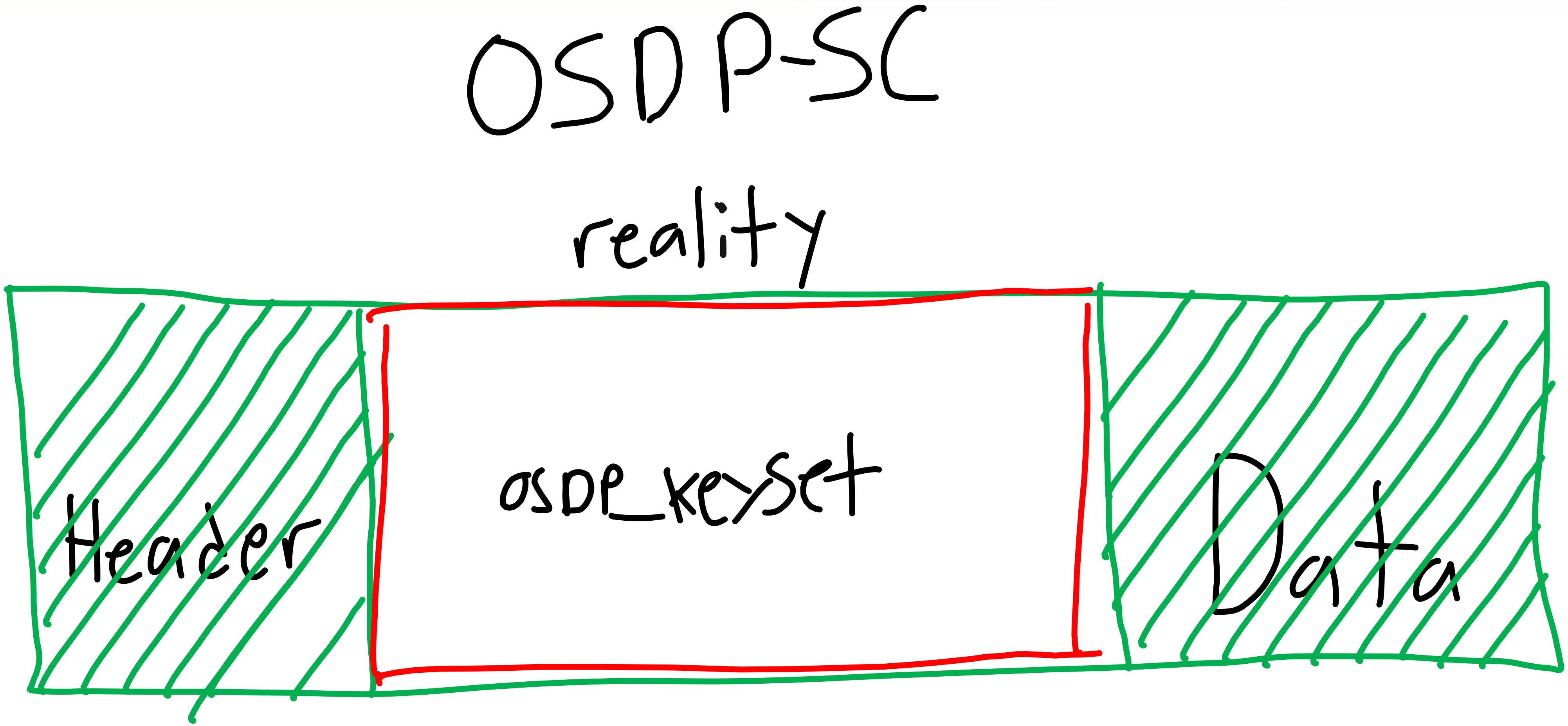
osdp_KEYPAD



osdp_BIOREADR



osdp_KEYSET



Demo #4

Weak

Keys

Attack

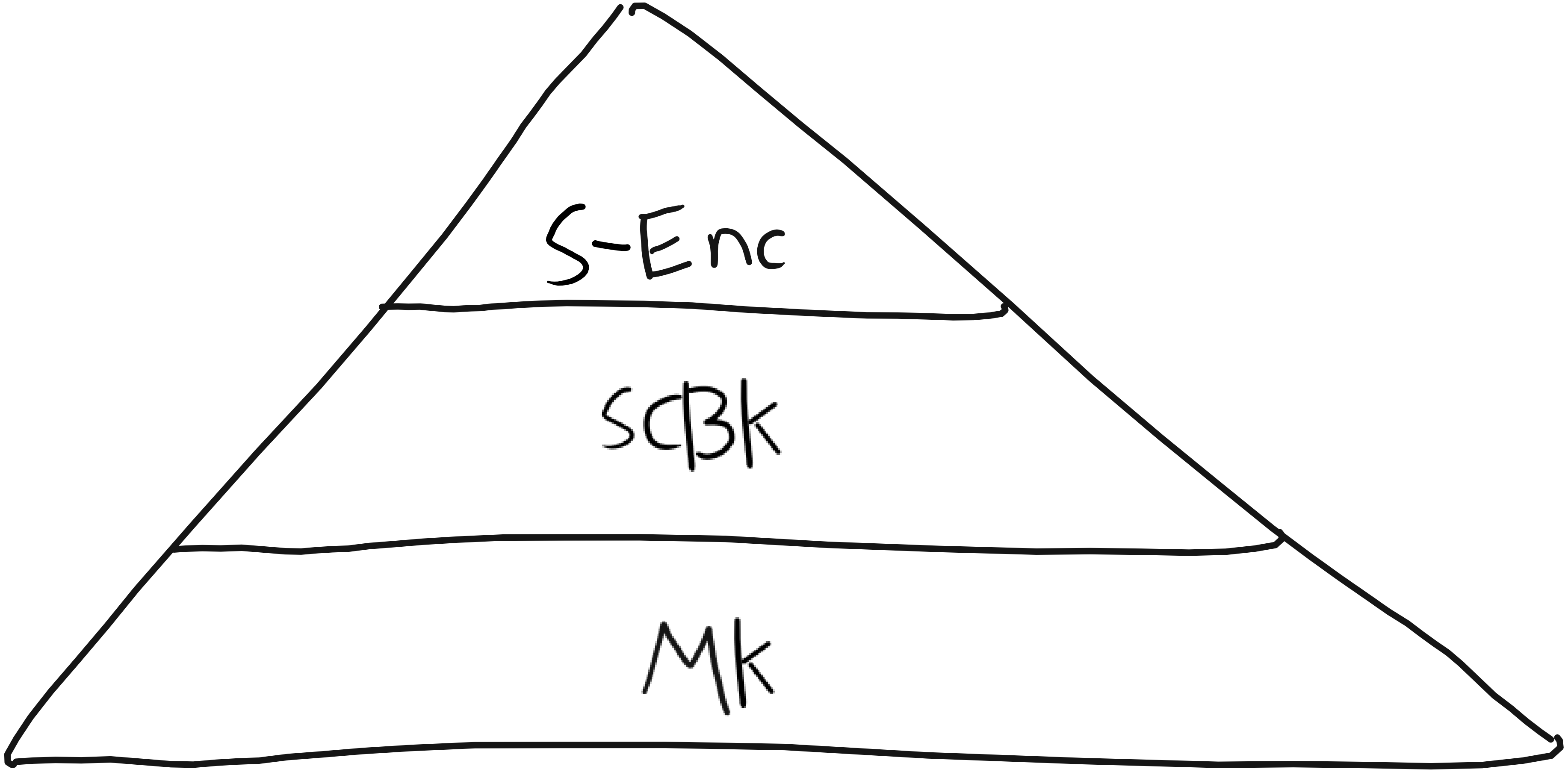
INTRODUCING..

Weak Keys

not protocol specific

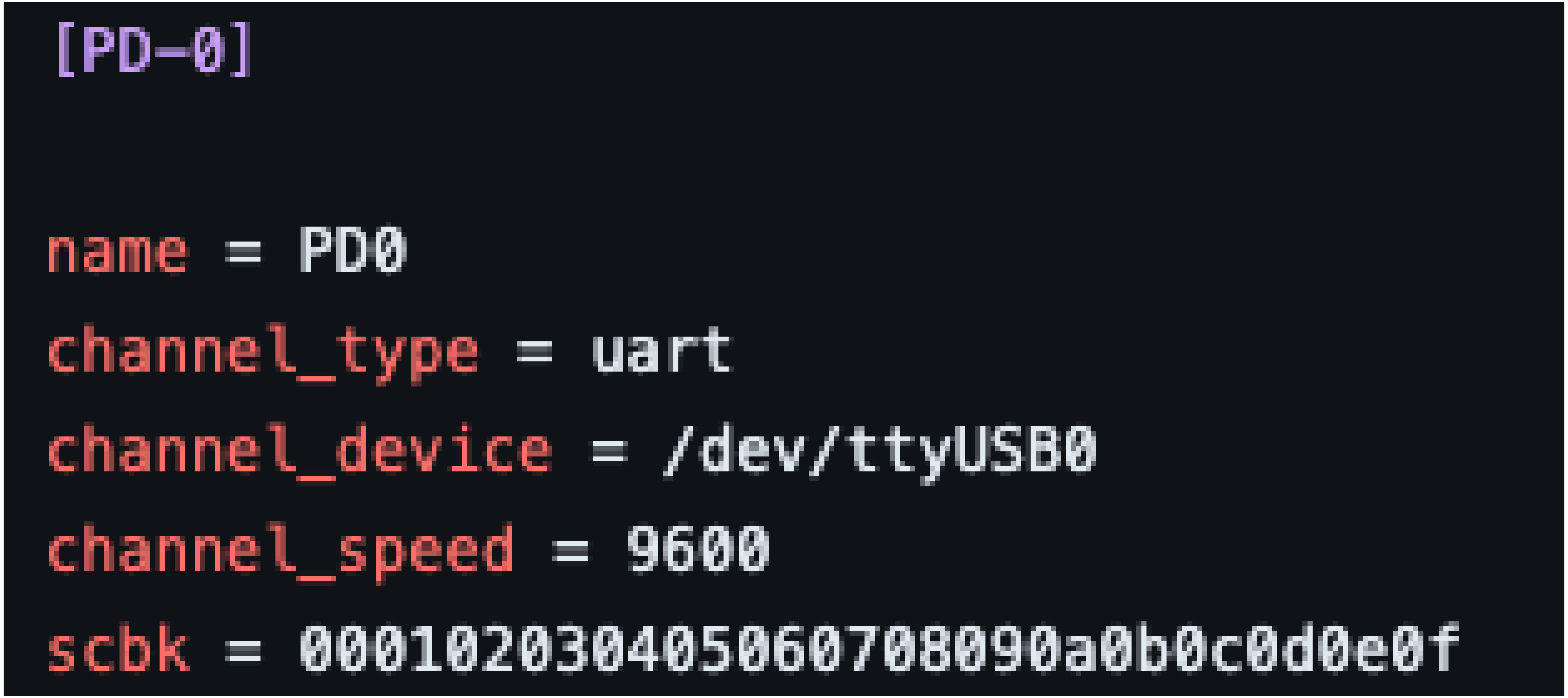
HIERARCHY

The keys



HARD CODED CREDS…

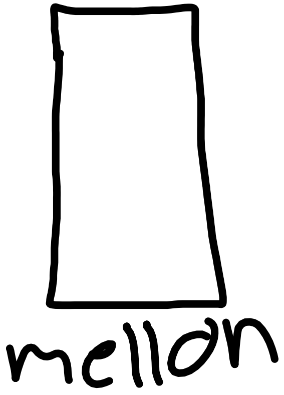
Introducing: Weak keys



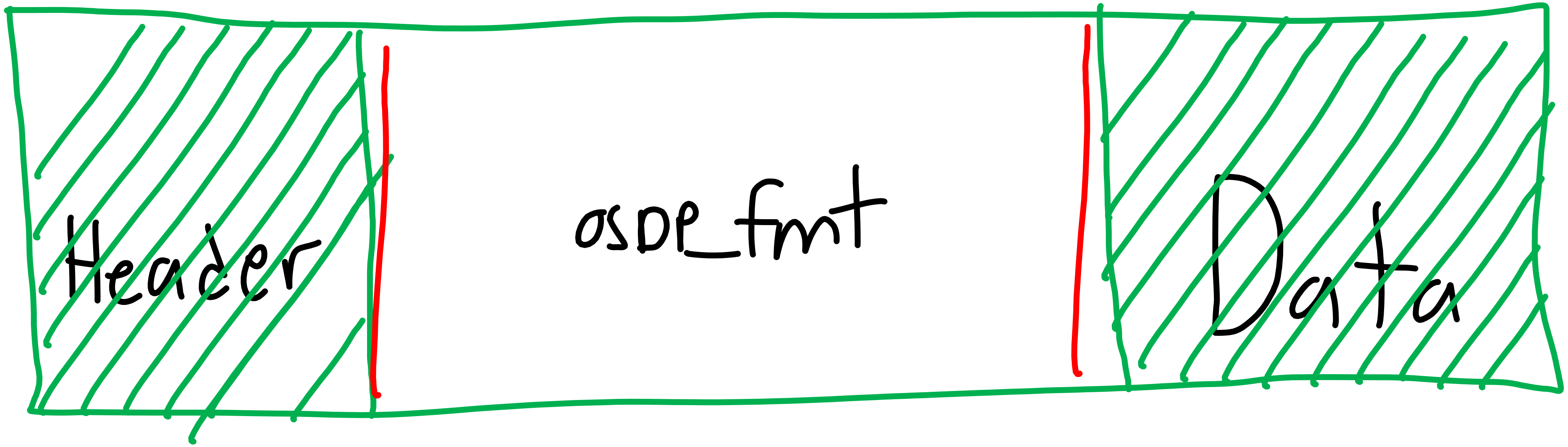
```
[PD-0]


name = PD0
channel_type = uart
channel_device = /dev/ttyUSB0
channel_speed = 9600
scbk = 000102030405060708090a0b0c0d0e0f
```

Using weak keys can't possibly backfire...



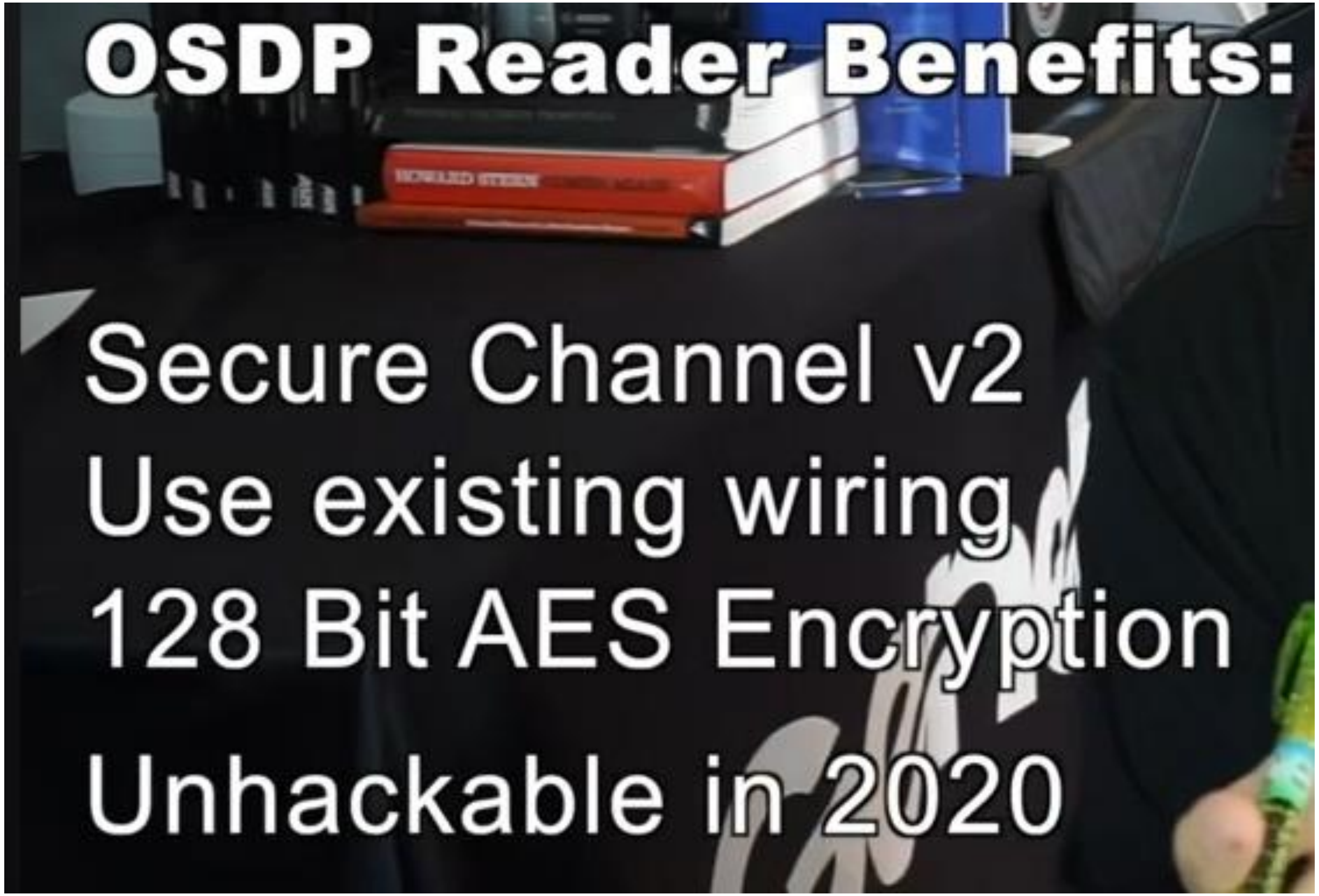
Key 1 = 0x00, 0x00, 0x00…
Key 2 = 0x01, 0x02, 0x03…
Key 3 = 0x30, 0x31, 0x32…
Key 4 = …



Protocol WTF #4

Encryption is not magic fairy dust

Bold Claims



OSDP Reader Benefits:

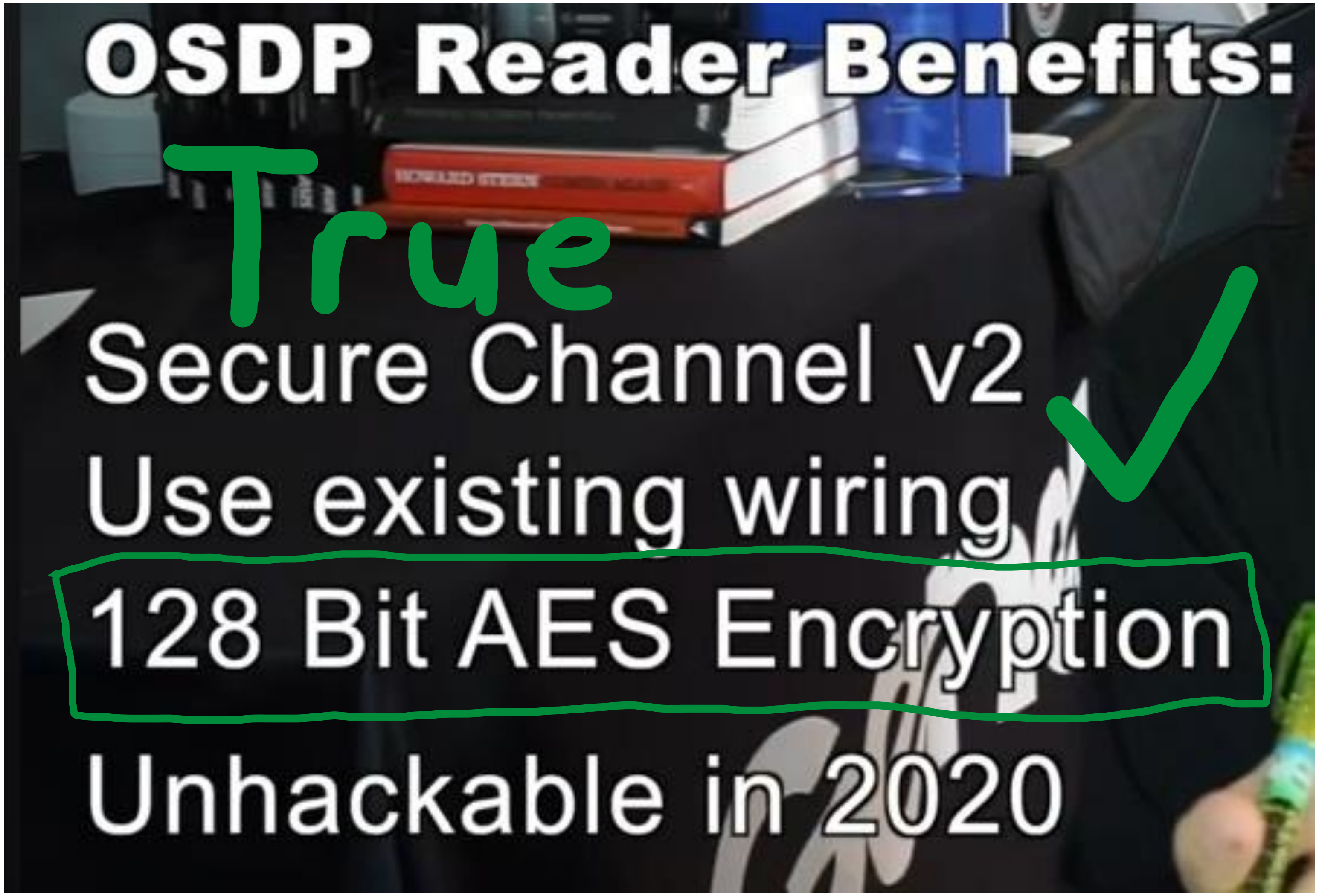
Secure Channel v2

Use existing wiring

128 Bit AES Encryption

Unhackable in 2020

Bold Claims



OSDP Reader Benefits:

True

Secure Channel v2 ✓

Use existing wiring

128 Bit AES Encryption

Unhackable in 2020

AES

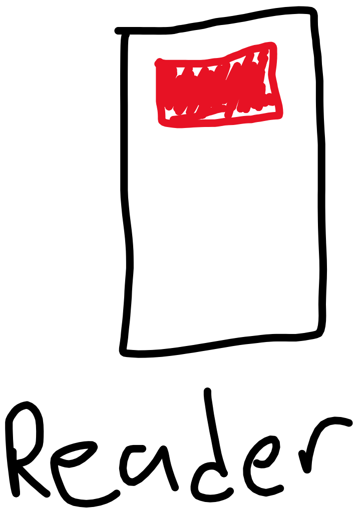
But ONLY AES

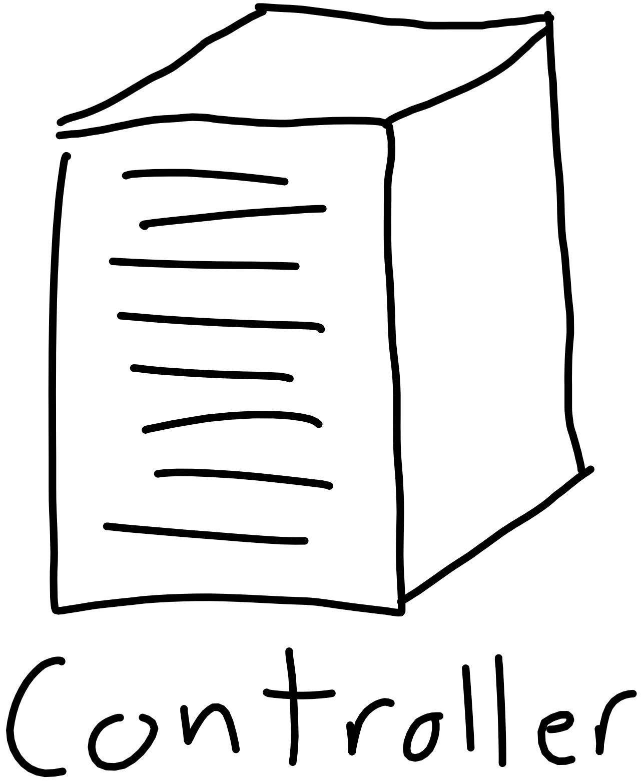
AES

But ONLY AES

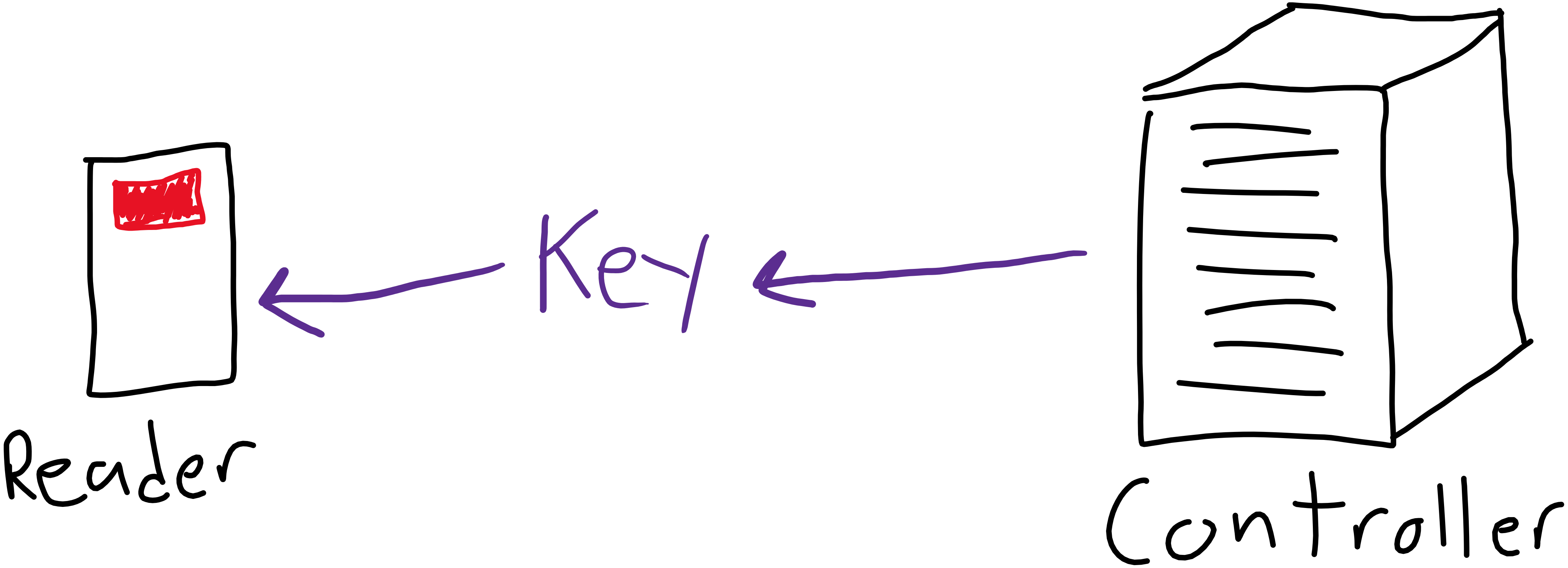
No asymmetric crypto

Key Exchange

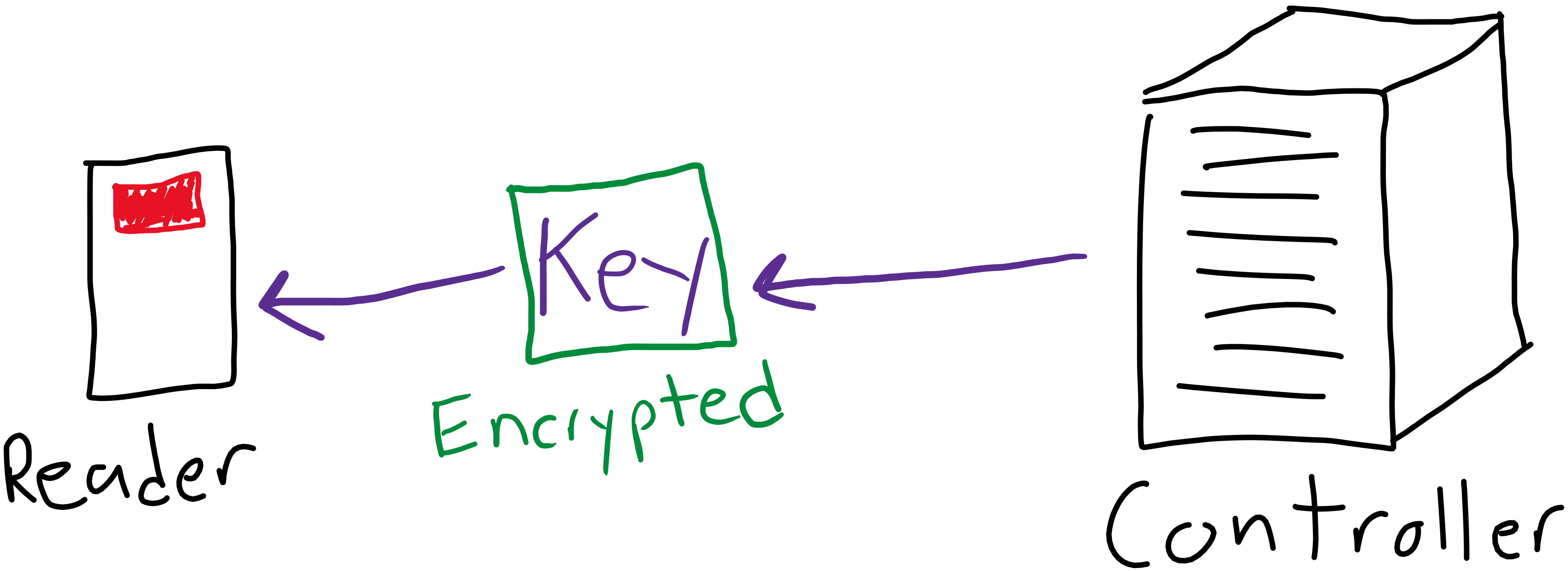




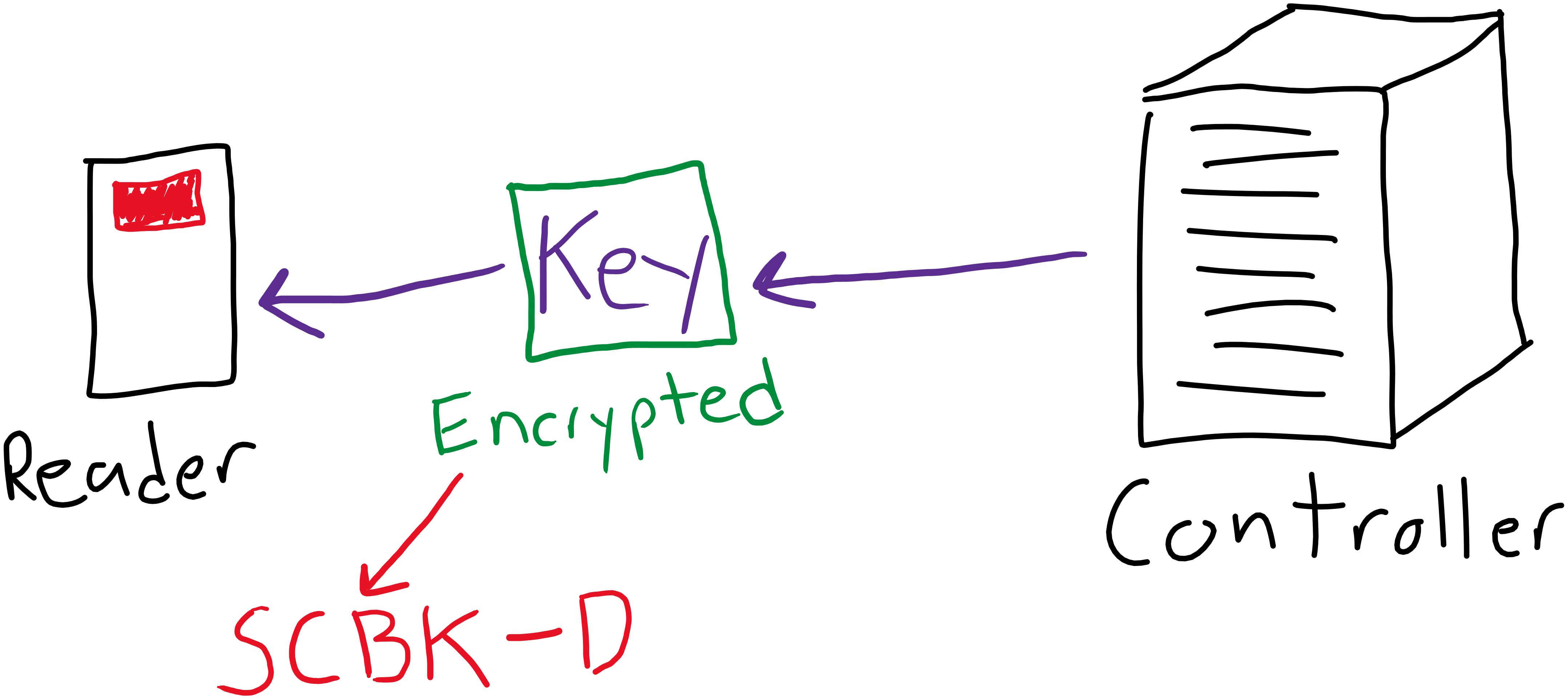
Key Exchange



Key Exchange



Key Exchange



Key Exchange

S C B K - D

Key Exchange

S C B K - D
e
c
u
r
e

Key Exchange

S C B K - D

e h
c a
u n
r n
e e
  l

Key Exchange

S	C	B	K	-	D
e	h	a			
c	a	s			
u	n	e			
r	n				
e	e				
	l				

Key Exchange

SCBK-D

Secure

Channel

Base

Key

Key Exchange

S C B K - D

Secure

Channel

Base

Key

Default

Key Exchange

SCBK - D

Secure
Channel
Base
Key
-
Default

0x30, 0x31,
0x32, 0x33,
. . .
0x3e, 0x3f

Demo #5

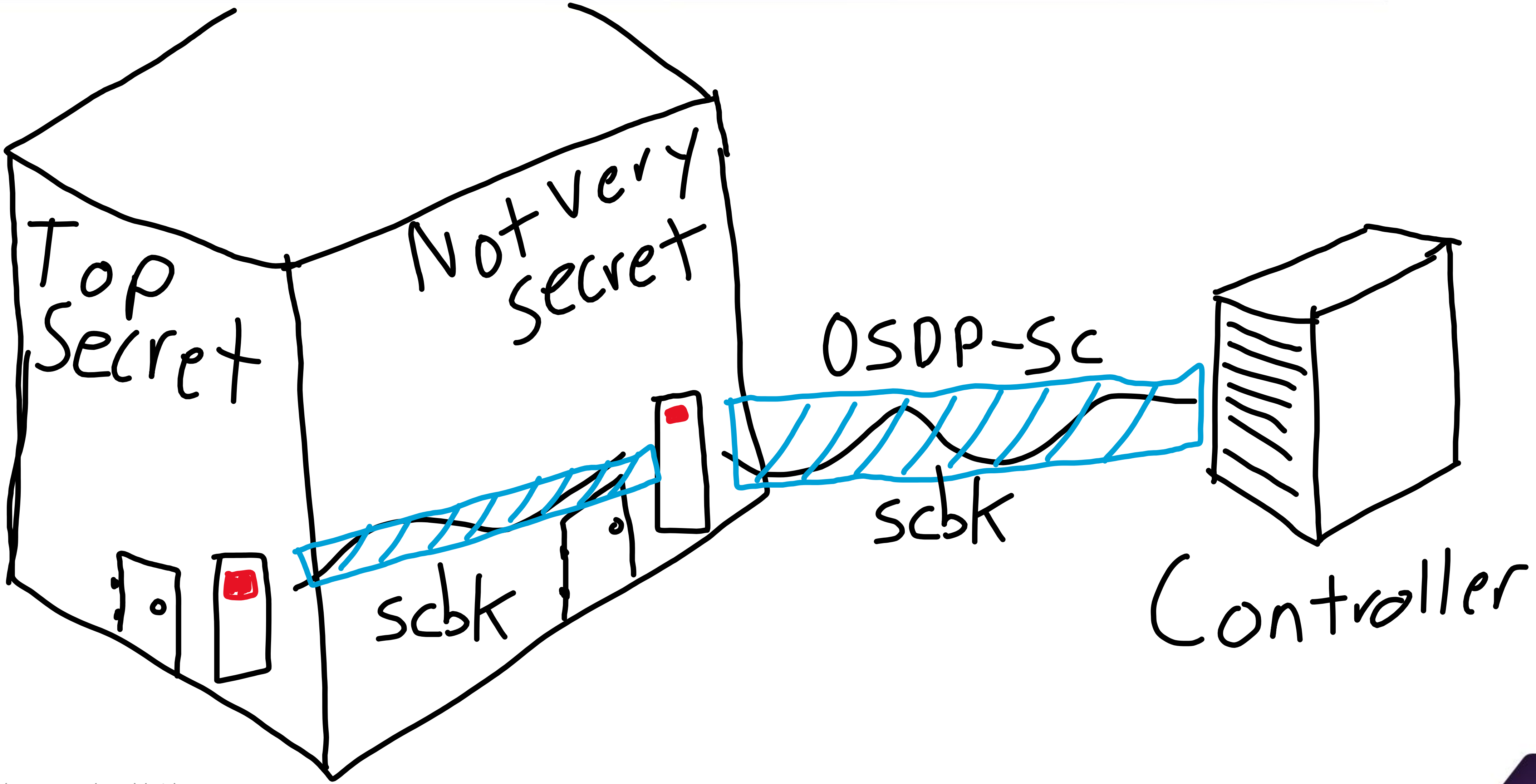
Keyset

Capture

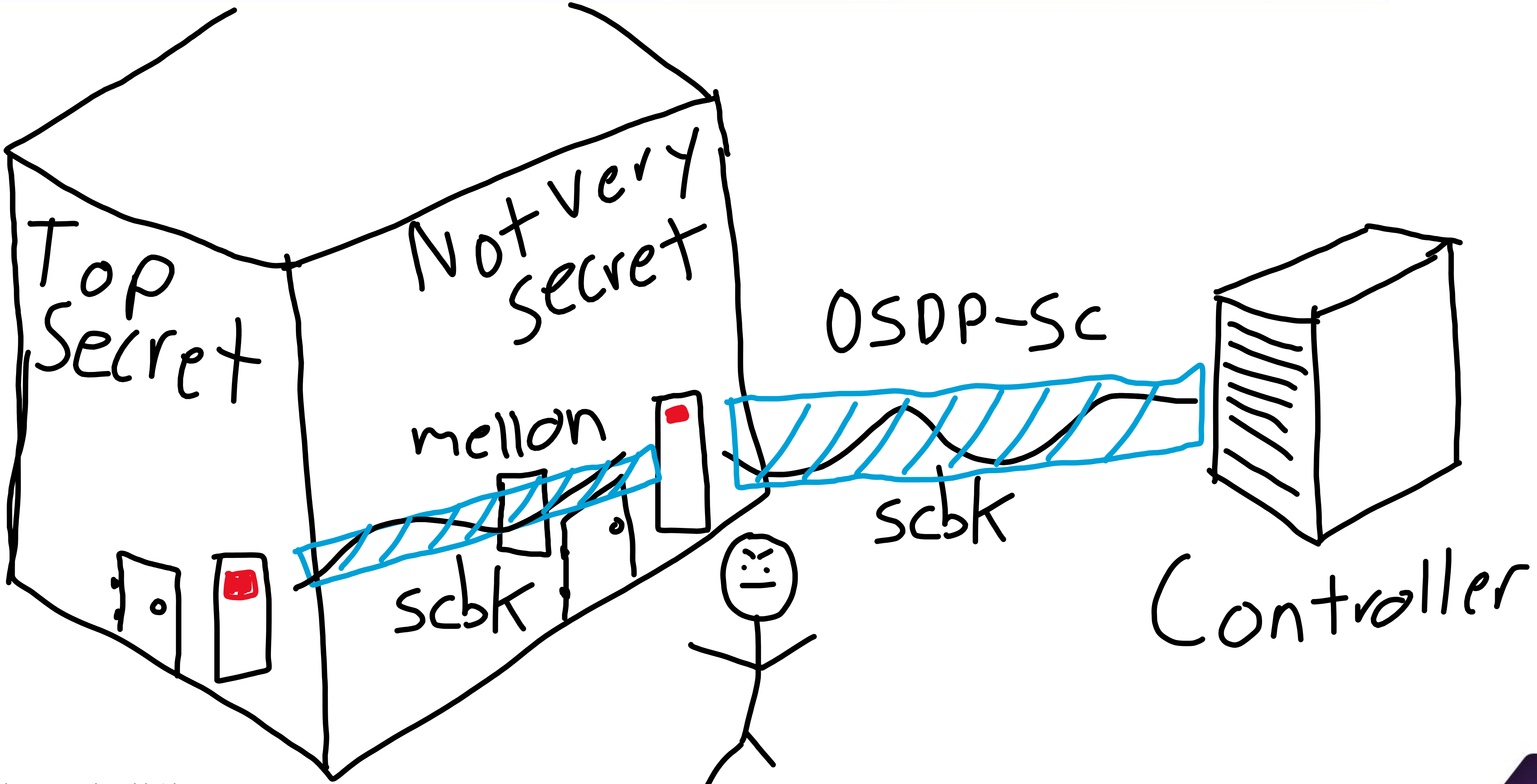
I've set everything up securely, I'm not affected by any of this

- "But, a lot of this can be avoided by configuring the device properly"
- If best practices were normally followed and devices were set up securely, many of us would be out of a job.
- Remember the broadcast nature of the protocol?

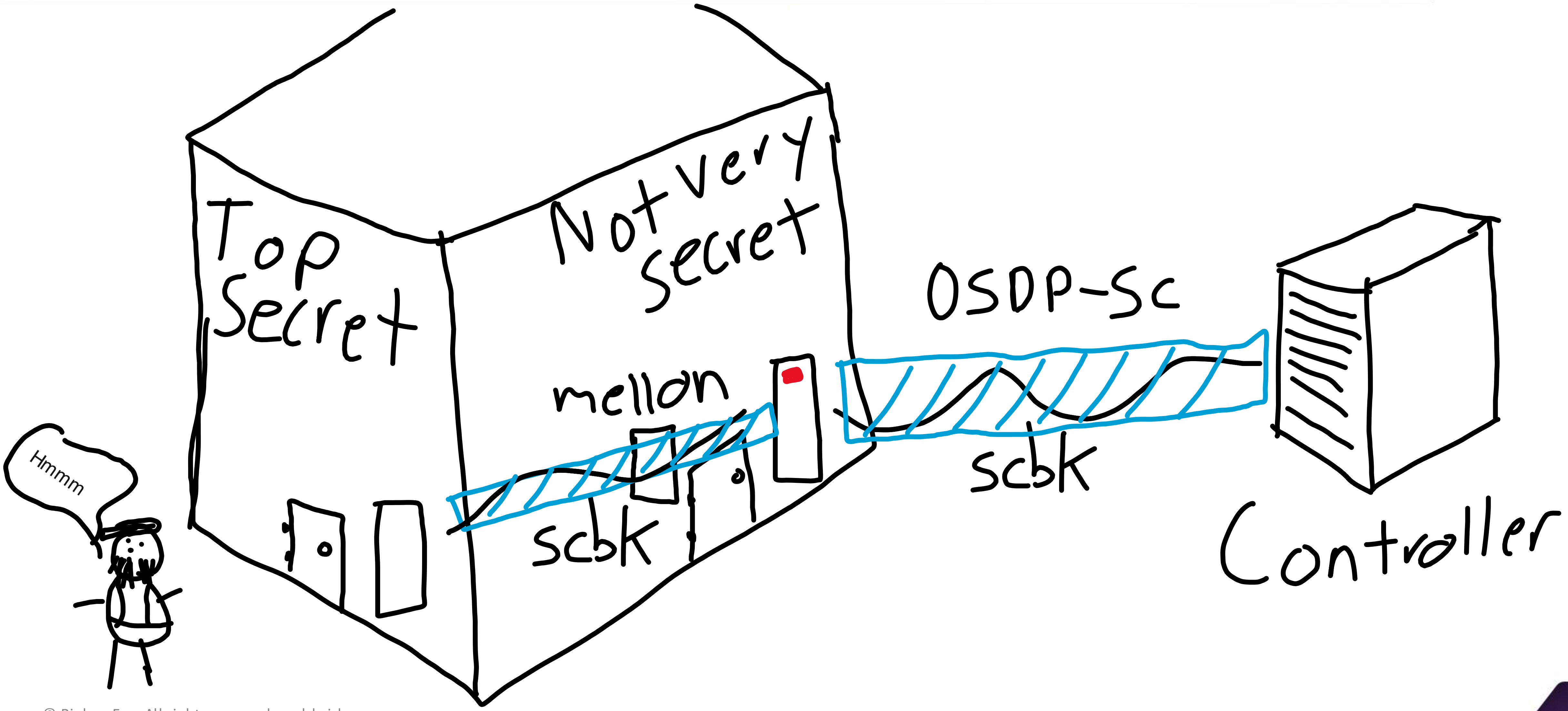
RS-485



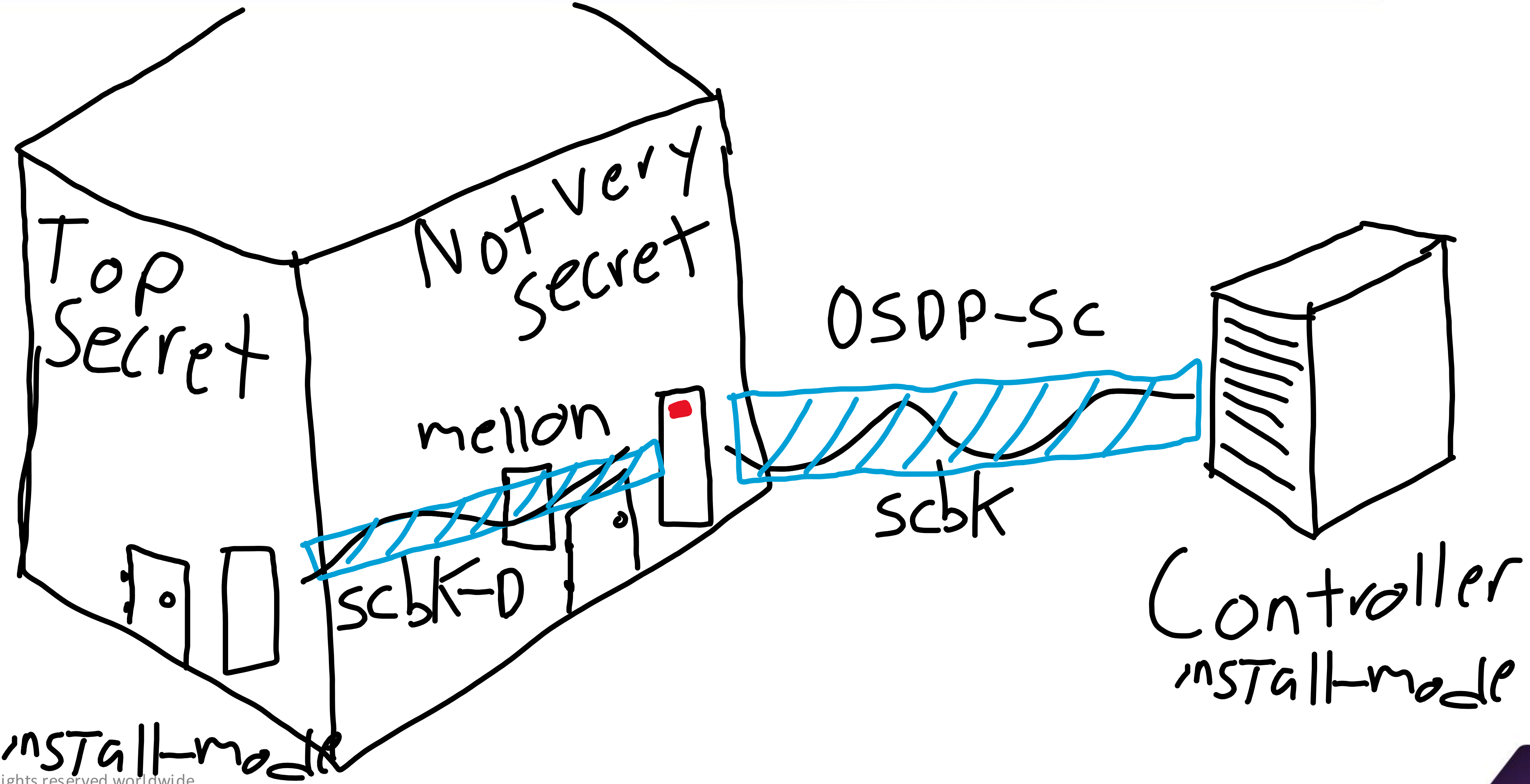
RS-485


Top Secret
Not very secret
mellon
scbk
OSDP-Sc
scbk
Controller

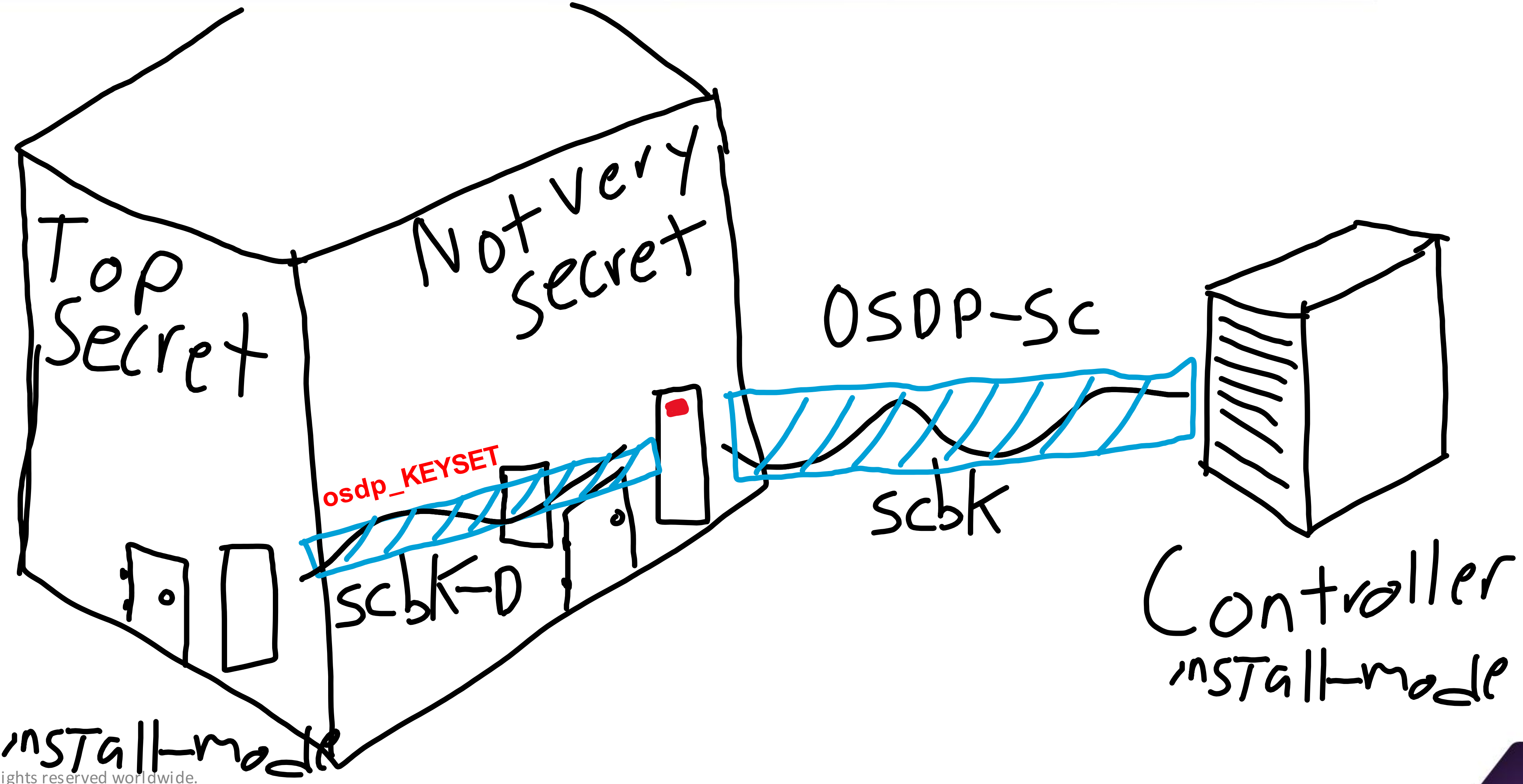
RS-485


Top Secret
Not very secret
mellon
scbk
OSDP-Sc
scbk
Controller
Hmmm

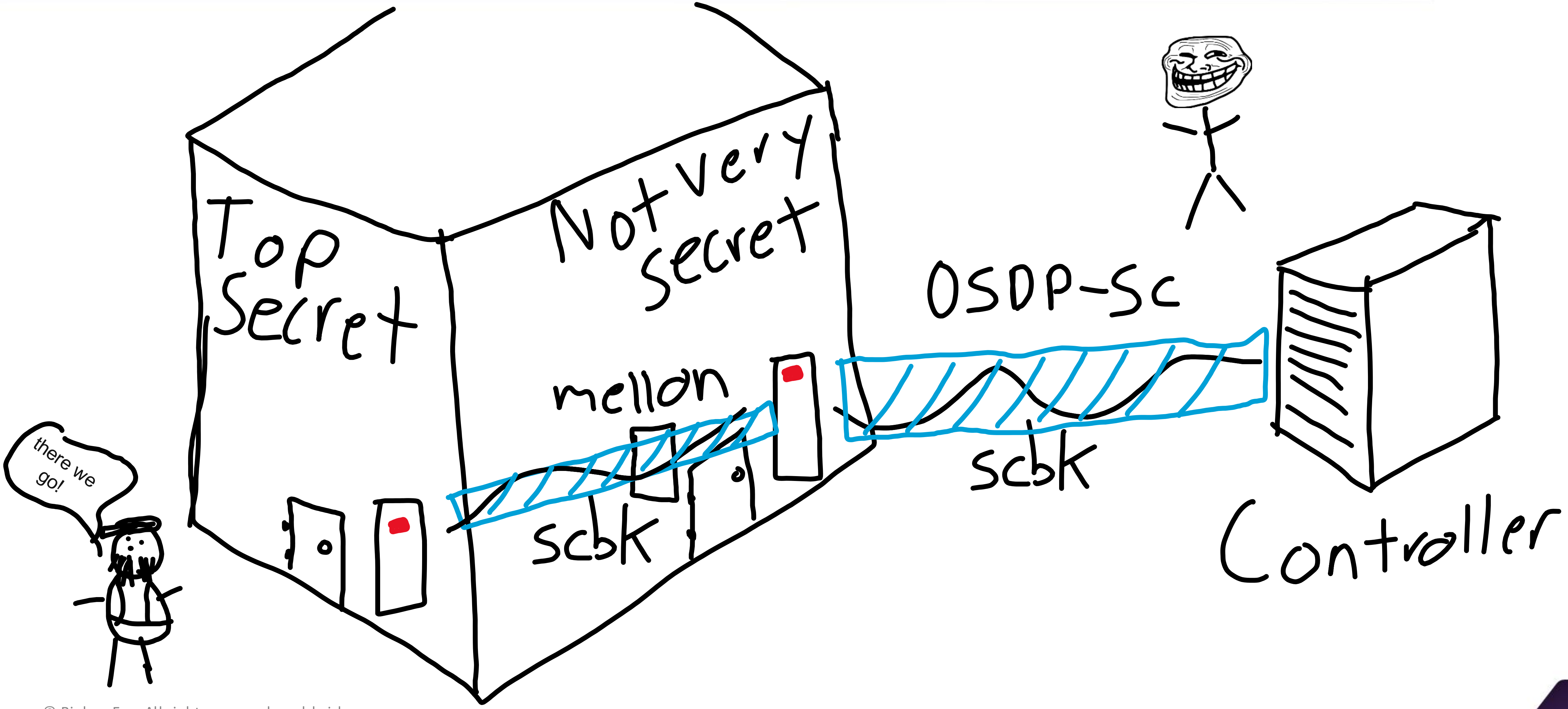
RS-485


Top Secret
Not very secret
mellon
scbk-D
OSDP-Sc
scbk
Controller
install-mode
install-mode

RS-485


Top Secret
Not very secret
osdp_KEYSET
scbk-D
OSDP-Sc
scbk
installmode
Controller
installmode

RS-485


Top Secret
Not very secret
mellon
scbk
OSDP-Sc
scbk
Controller
there we go!

Conclusion

. . . what do I do?!

ADVICE FOR DEFENDERS

Conclusion

Check your configs

ADVICE FOR DEFENDERS

Conclusion

Check your configs

✓ use encryption

Conclusion

Check your configs

✓ use encryption

✓ Require encryption

ADVICE FOR DEFENDERS

Conclusion

Check your configs

- ✓ use encryption
- ✓ Require encryption
- ✓ Disable Install Mode

Conclusion

Never configure a reader in production.

✗

ADVICE FOR DEFENDERS

Conclusion

Don't ignore tamper alerts

✗

ADVICE FOR DEFENDERS

Conclusion

Buy OSDP Verified Devices

✓

ADVICE FOR DEFENDERS

Conclusion

Don't trust

"It's encrypted"

Thanks!