Shamir Secret Sharing with no ID numbers

Jorge Arce

Blockchain and Cryptography Researcher @Nethermind

Shamir Secret Sharing: review

Problem description

You just got the private key of a wallet holding 1000 ETH!

(volcano laptop monster decide october blue now drastic laptop slow effort collect)

For secure long-term storage, you want to:

- Split the key into several 12-word seedphrases, such that two of them are needed for reconstruction.
- You want the secret to be accessible even if one piece gets lost.

You will create 3 pieces. This is called a (2,3) threshold scheme.

Share generation:

volcano laptop monster decide october
blue now drastic laptop slow effort collect

Share generation:

volcano laptop monster decide october
blue now drastic laptop slow effort collect

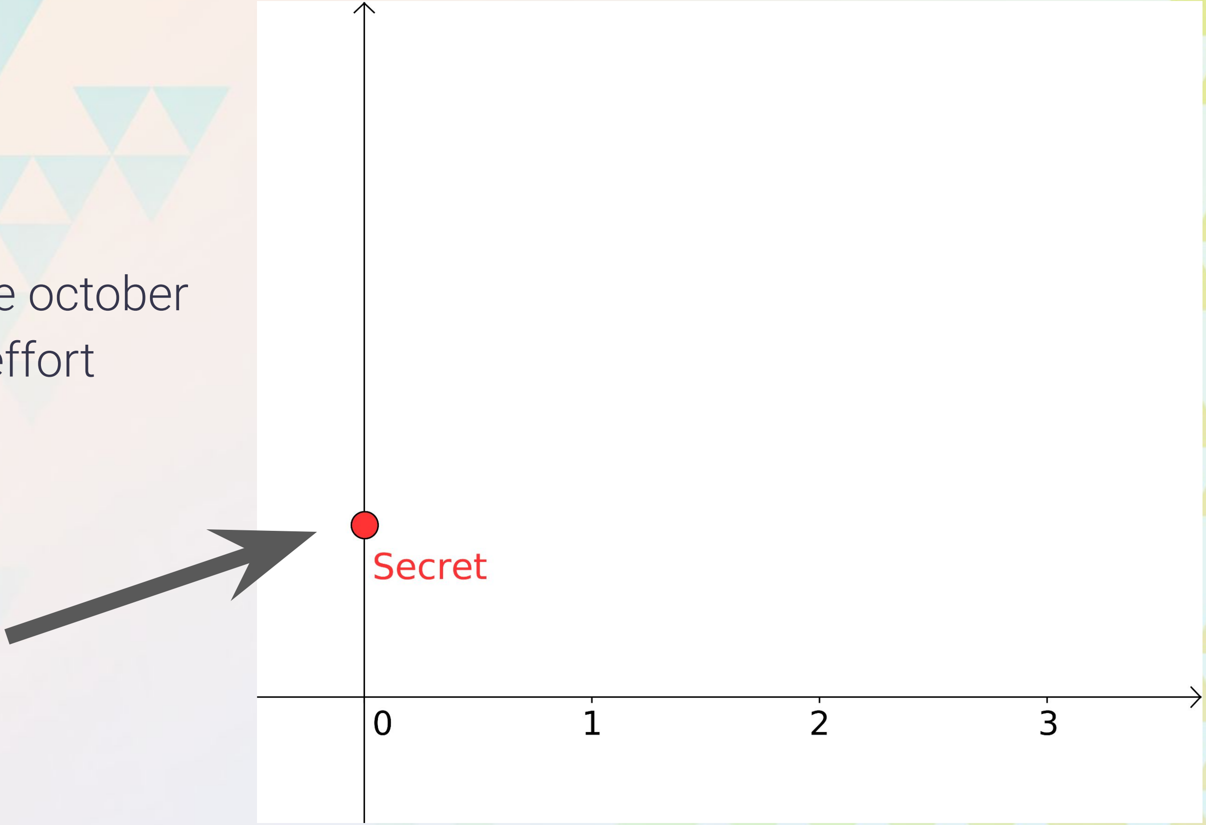
11110101101 01111101000
10001111010 00111000110
10011001000 00011000010
10010111001 01000010010
01111101000 11001100001
01000110101 00101101100

(Encoding with BIP-39’s word dictionary)

Share generation:

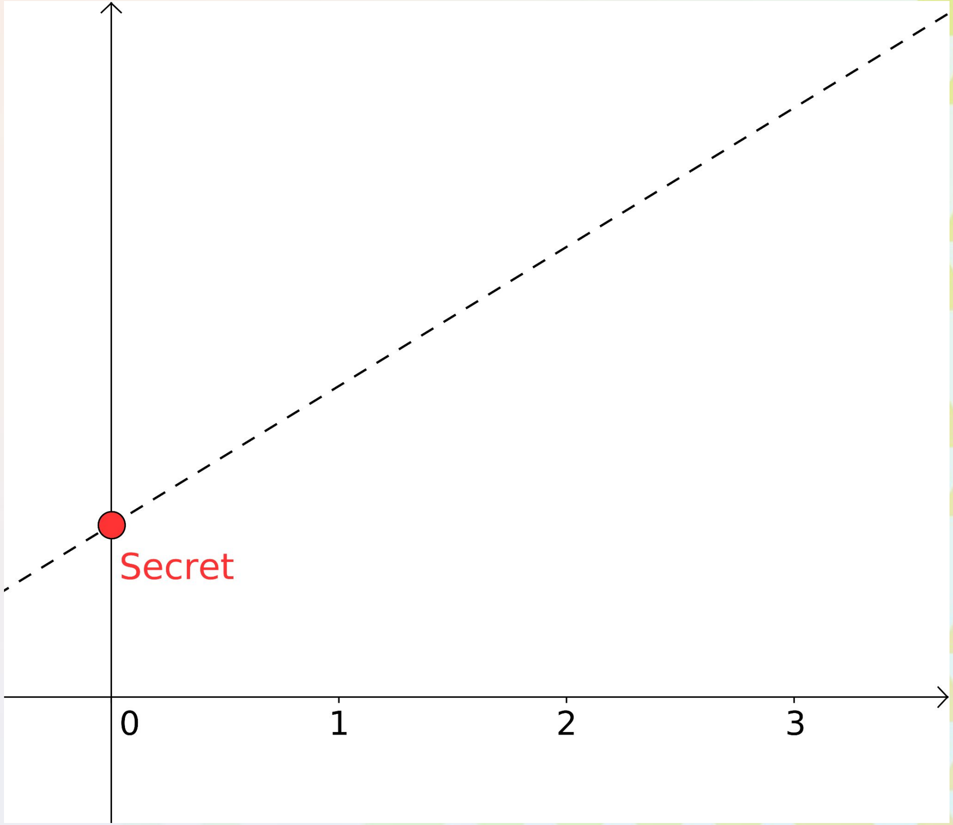
volcano laptop monster decide october blue now drastic laptop slow effort collect

11110101101 01111101000
10001111010 00111000110
10011001000 00011000010
10010111001 01000010010
01111101000 11001100001
01000110101 00101101100



Share generation:

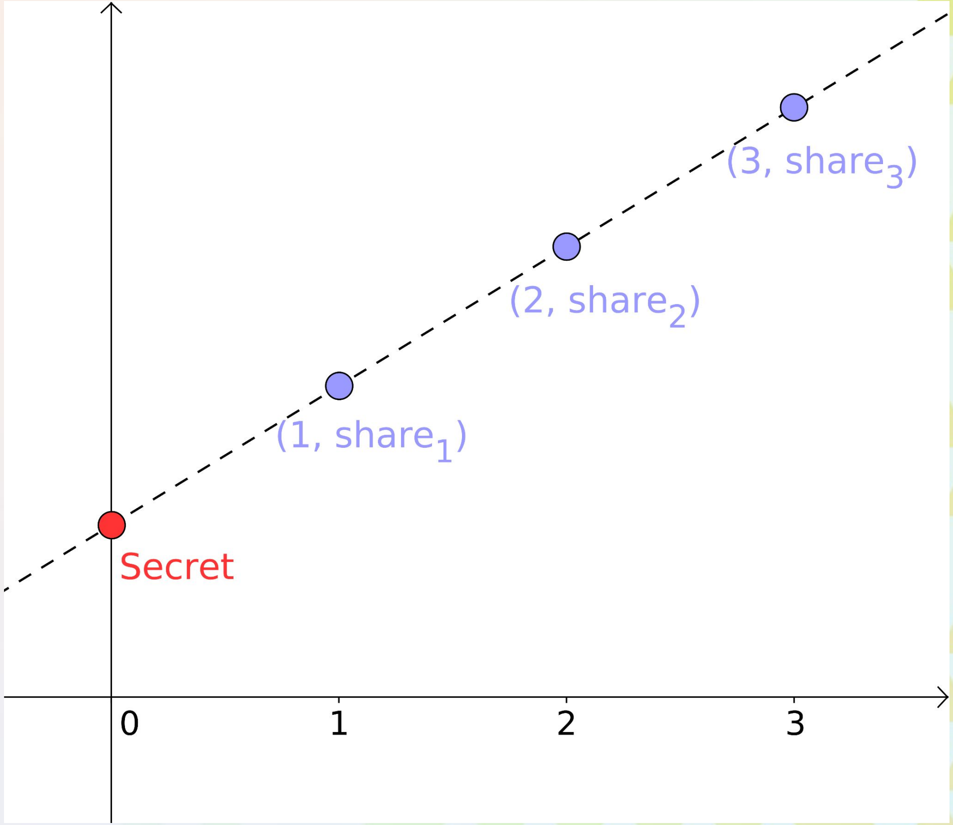
Choose a random straight line passing through the secret.



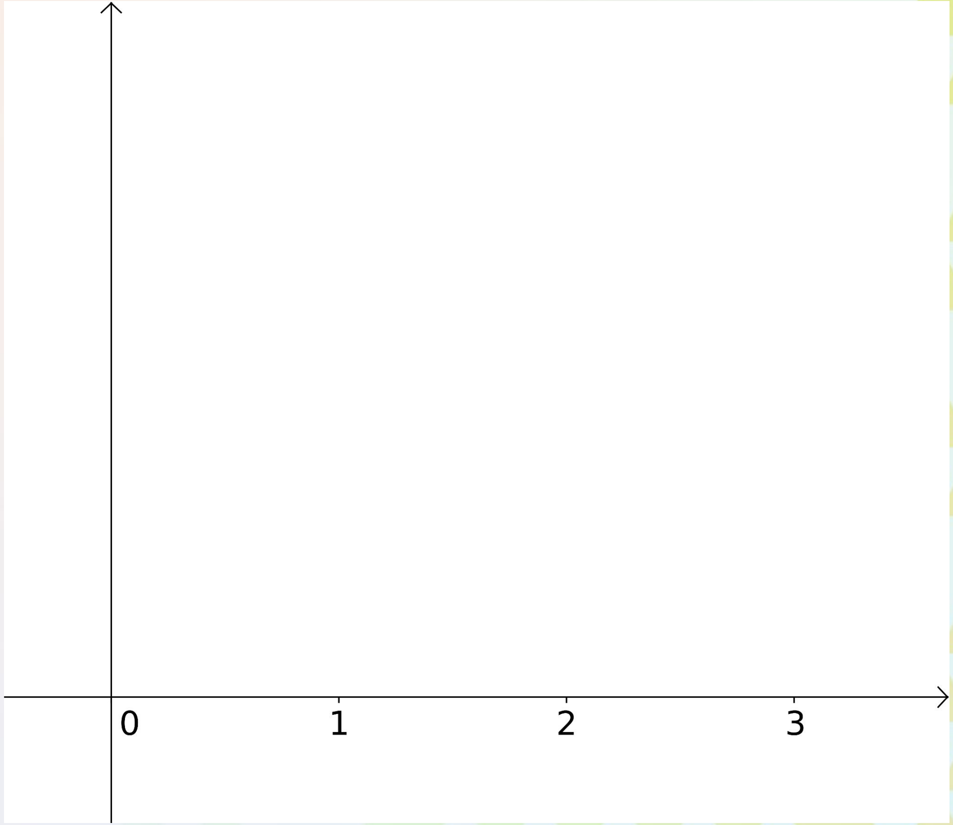
Share generation:

Pick three points on the line.

These are your shares.

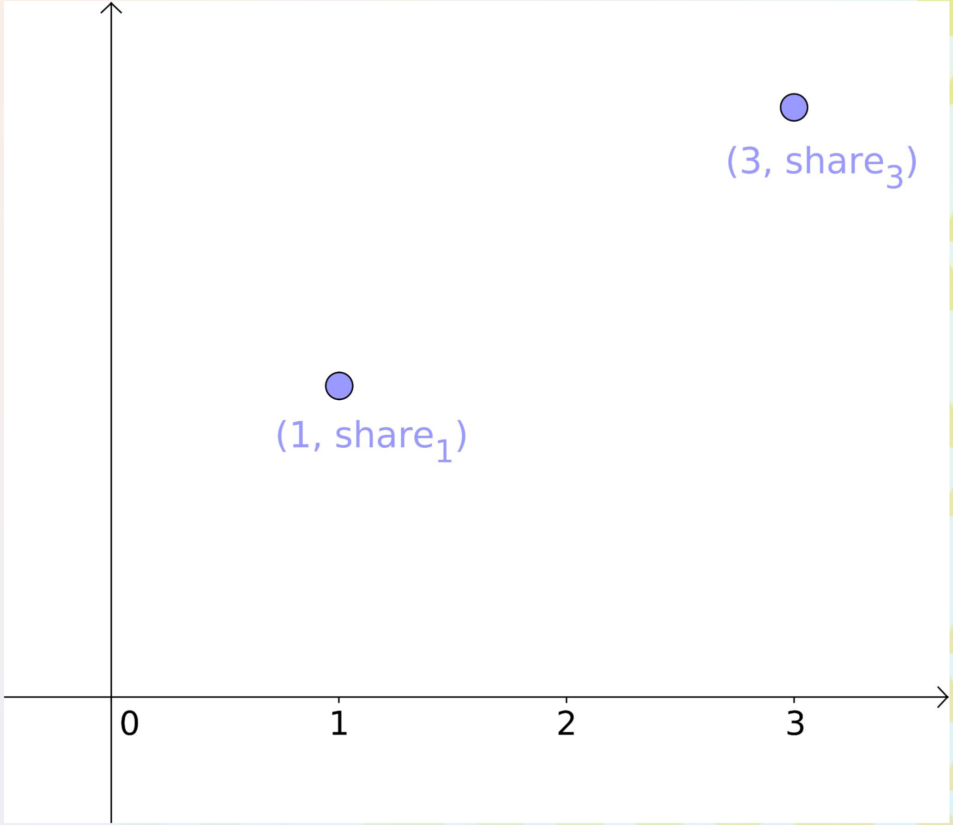


Share generation:



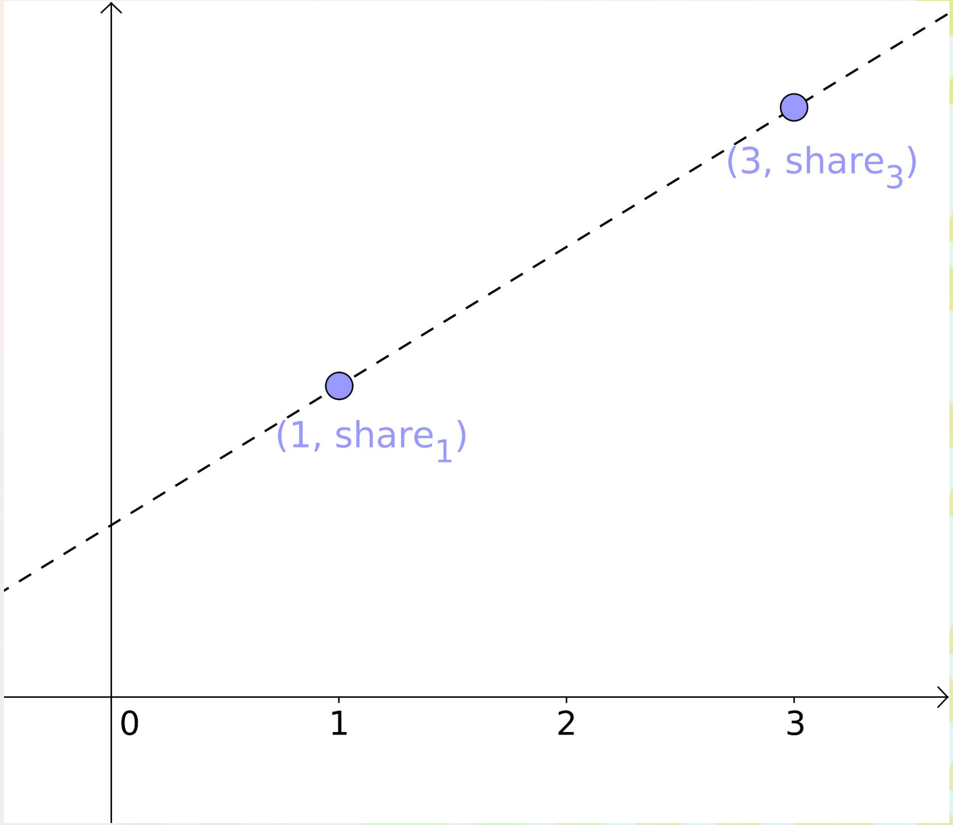
Share generation:

Any two shares generate the correct secret.



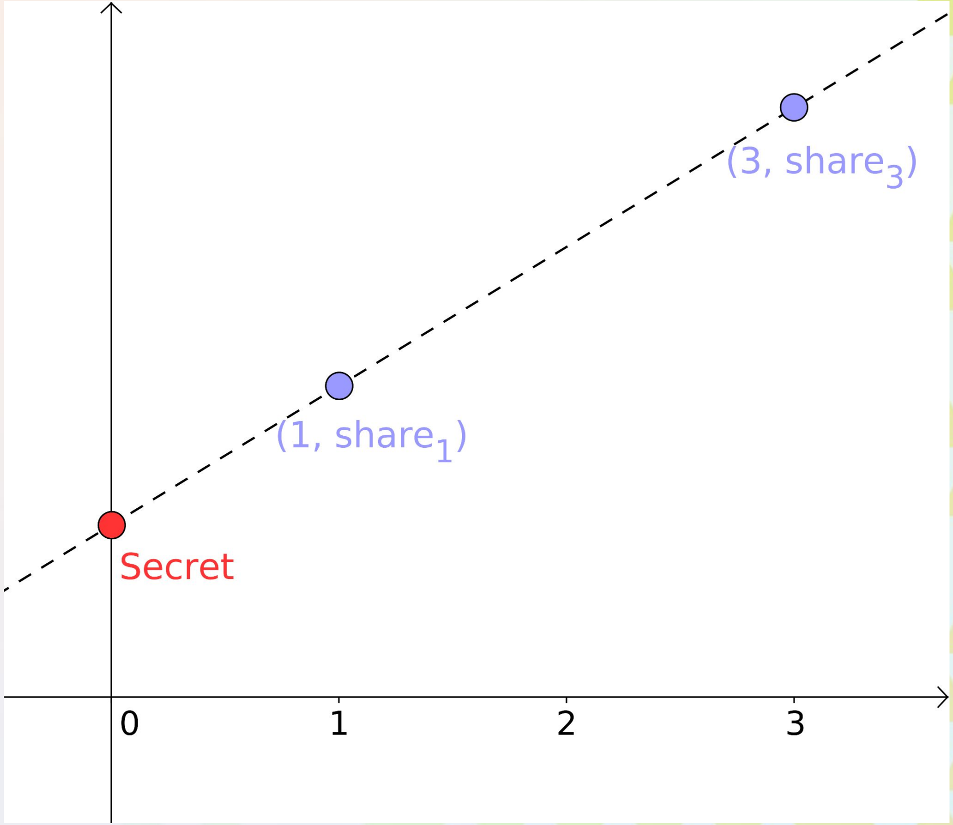
Share generation:

Any two shares generate the correct secret.



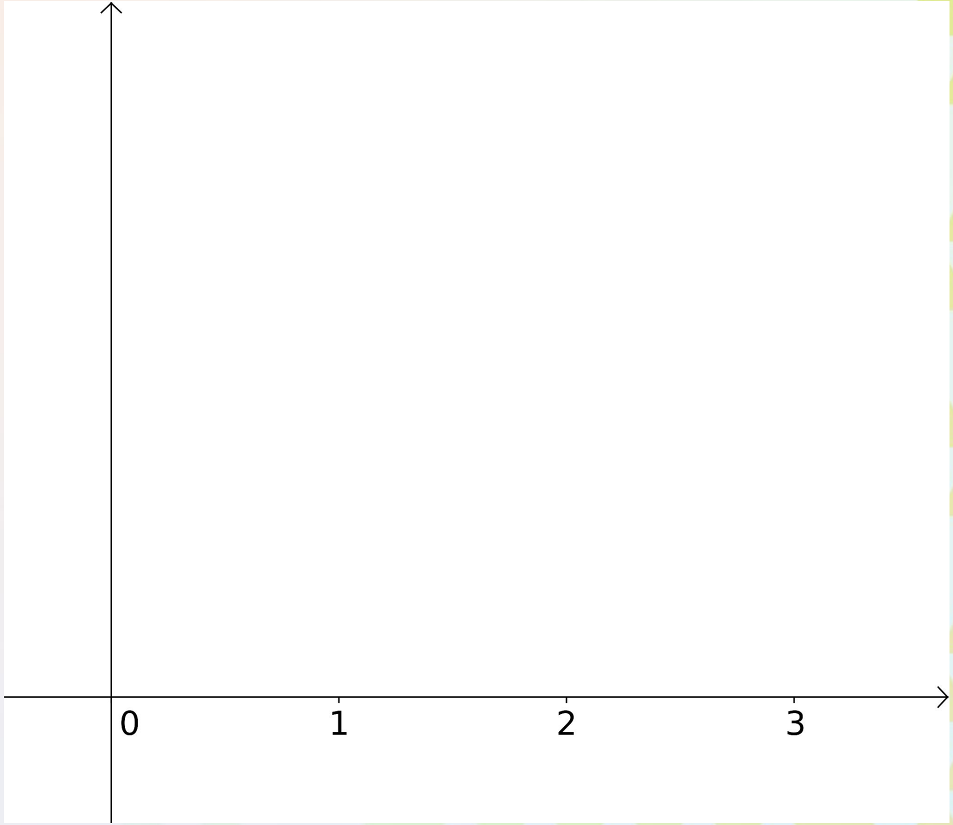
Share generation:

Any two shares generate the correct secret.



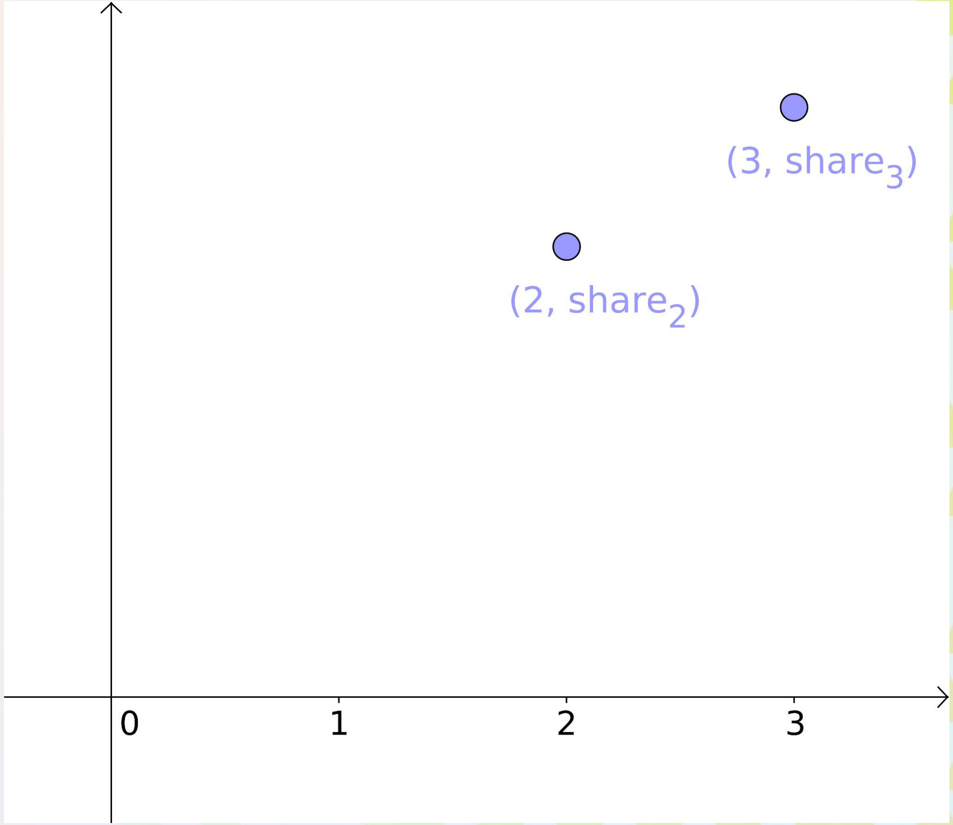
Share generation:

Another example.



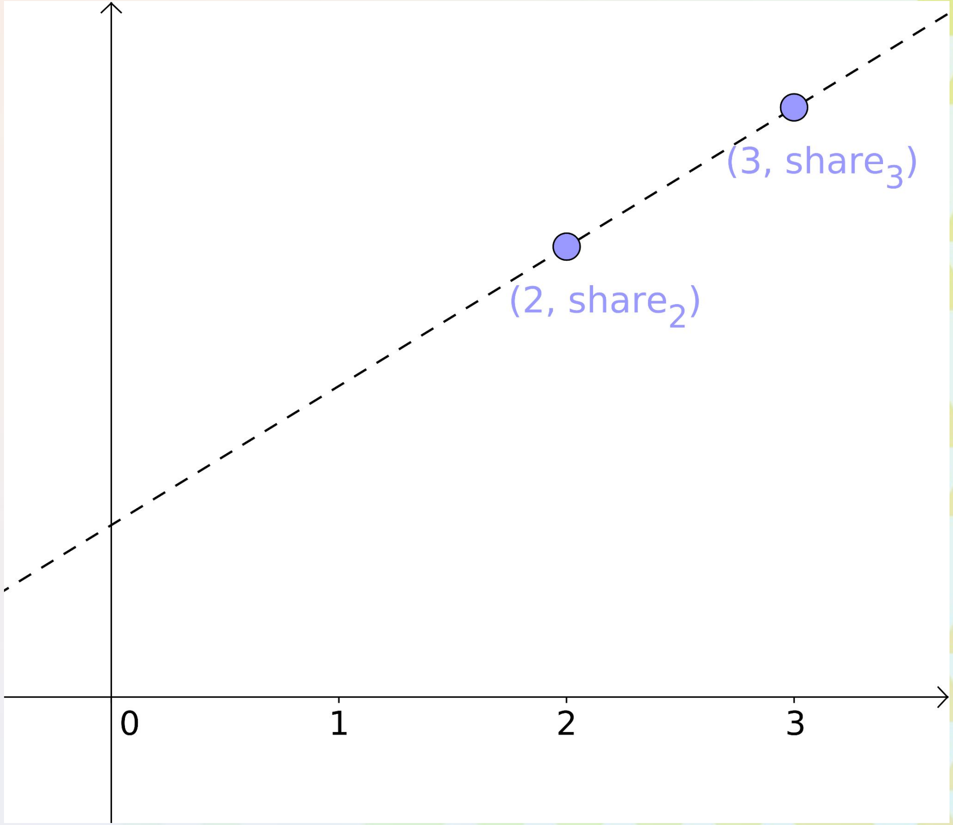
Share generation:

Another example



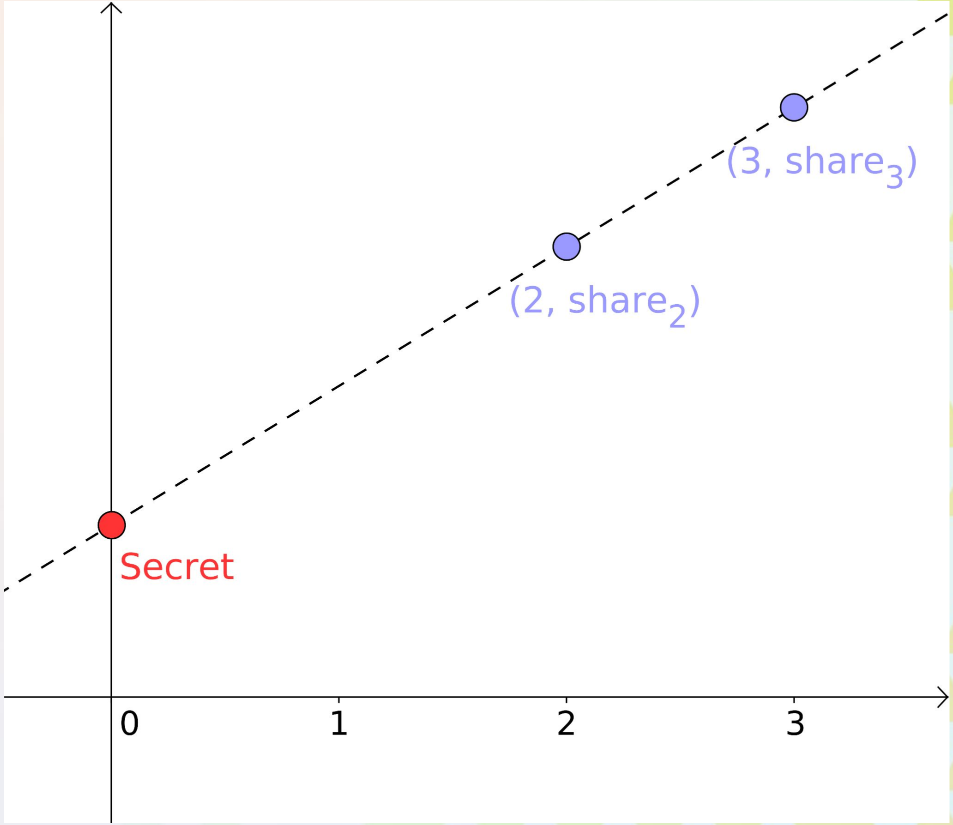
Share generation:

Another example



Share generation:

Another example



In this example, the shares are

(**1**, 1001100011 1111001110 0100111111 1100011010 0101111111 0011010000 1100001101 0001110011 1001010000 0010000010 0010100100 1011001100 0101)

(**2**, 0010111100 0011000000 0010000111 0001110001 0001011010 0100111110 0000110100 0110100001 0000100010 0100111011 0100011001 0100000101 0001000111 110)

(**3**, 0100001001 0111011101 0111001110 1000011101 0000111100 1001010001 0000000011 1001011101 0110111100 0010100101 0000011011 0110000010 0111100101 11)

The x-values are called **ID numbers.**

Encoding the binary into seedphrases, we get:

(**1**, “ocean vicious exit shoot save half artist transfer expand animal pigeon obvious”)

(**2**, “congress gasp athlete session stand wealth person ancient cherry edge little elephant”)

(**3**, “drastic upgrade soldier deliver venture marine defense pupil apart lock gauge very”)

Remark: ID numbers are crucial

What if we miswrote the ID number of:

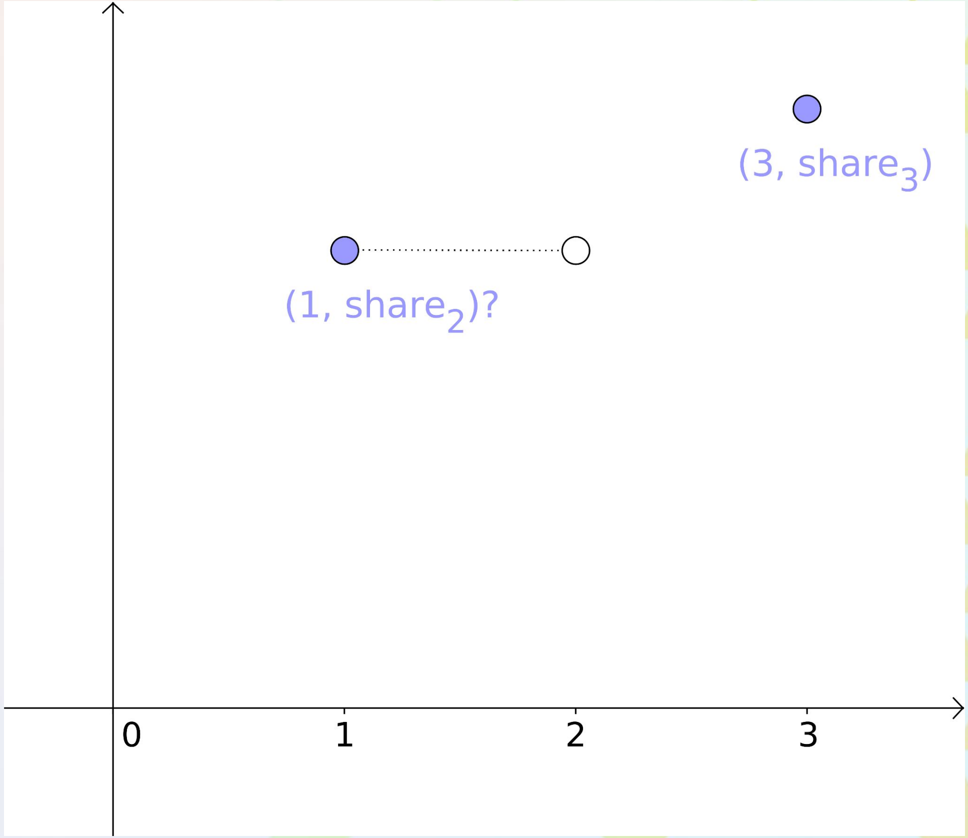
(**2**, “congress gast athlete session stand wealth person ancient cherry edge little elephant”)

Remark: ID numbers are crucial

What if we miswrote the ID number of:

(**1**, “congress gast athlete session stand wealth person ancient cherry edge little elephant”)

…and used a 1 instead?

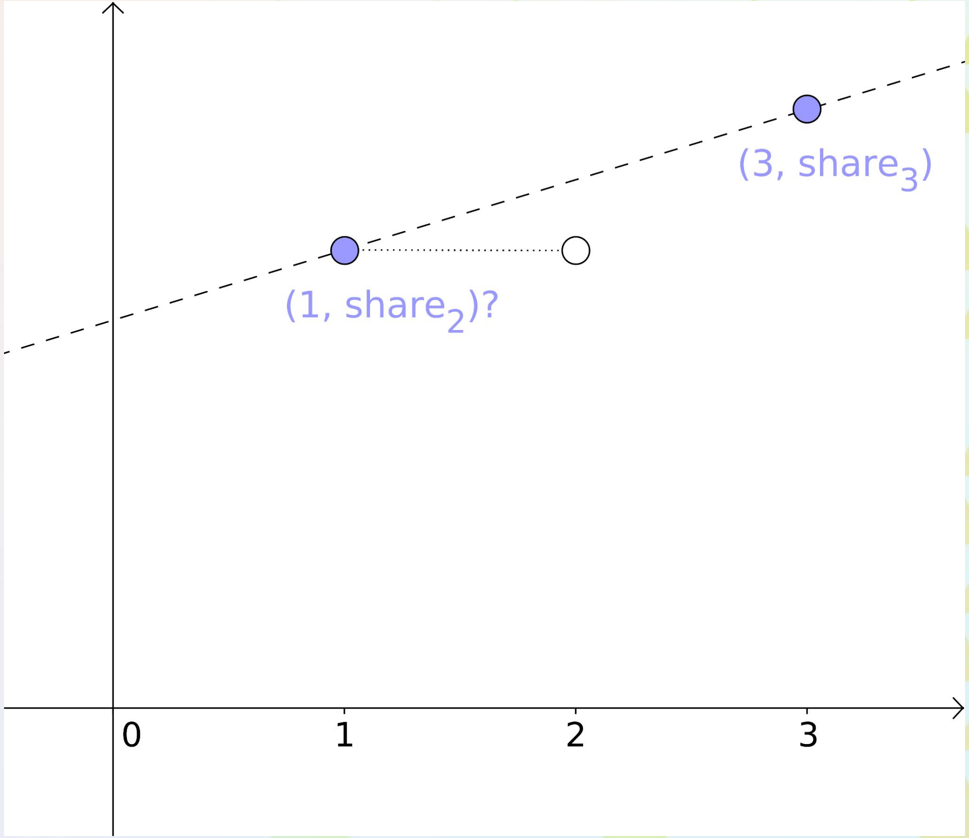


Remark: ID numbers are crucial

What if we miswrote the ID number of:

(**1**, “congress gast athlete session stand wealth person ancient cherry edge little elephant”)

…and used a 1 instead?

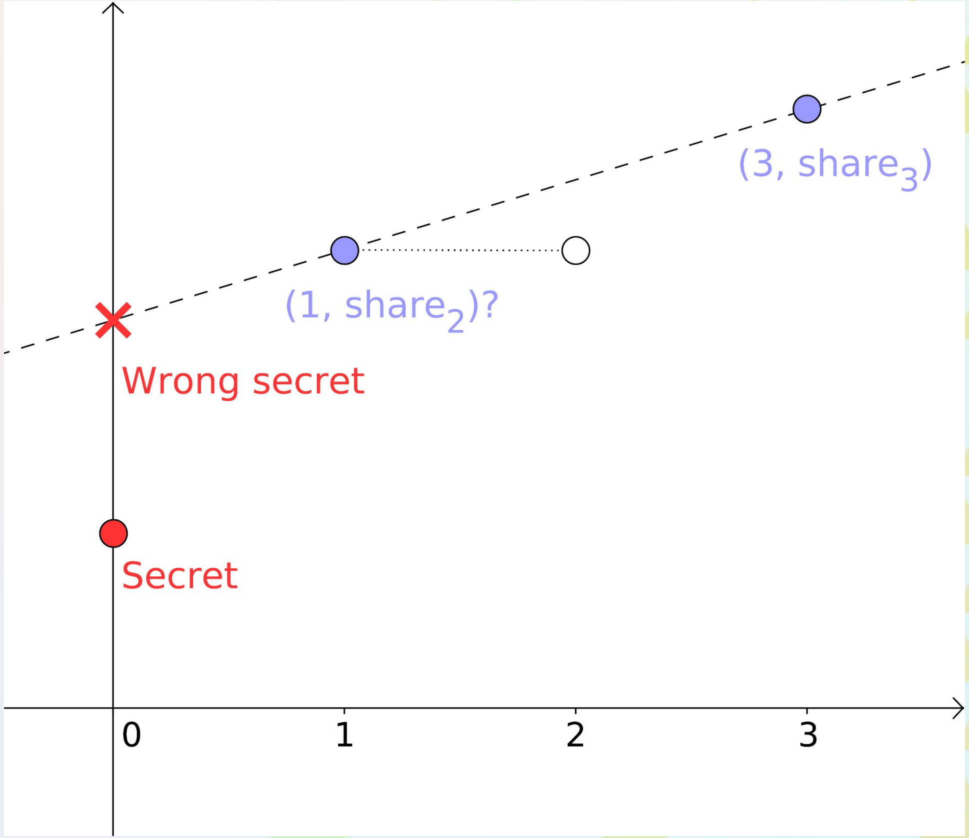


Remark: ID numbers are crucial

What if we miswrote the ID number of:

(**1**, “congress gast athlete session stand wealth person ancient cherry edge little elephant”)

…and used a 1 instead?



Can we improve this?

(Goal: encode the ID numbers in the seedphrase itself, without adding extra words)

BIP-39 seedphrase standard review

Not all the bits corresponding to the seed phrase carry independent information.

11110101101 01111101000 10001111010 00111000110
10011001000 00011000010 10010111001 01000010010
01111101000 11001100001 01000110101 00101101100

BIP-39 seedphrase standard review

Not all the bits corresponding to the seed phrase carry independent information.

11110101101 01111101000 10001111010 00111000110
10011001000 00011000010 10010111001 01000010010
01111101000 11001100001 01000110101 0010110**1100**

BIP-39 seedphrase standard review

Not all the bits corresponding to the seed phrase carry independent information.

11110101101 01111101000 10001111010 00111000110
10011001000 00011000010 10010111001 01000010010
01111101000 11001100001 01000110101 0010110**1100**

sha256(11110101101…) = 1100…

No need to include the checksum bits for share generation/reconstruction

volcano laptop monster decide october blue now drastic laptop slow effort collect

11110101101 01111101000 10001111010 00111000110
10011001000 00011000010 10010111001 01000010010
01111101000 11001100001 01000110101 0010110~~1100~~

Do SSS on the non-checksum bits, get new shares.

(**1**, 00010101010 11110100001 01010010100 10101111001
00100010111 11110111001 10110101001 00000101110
11111111001 00101110111 10111001100 1010100)

(**2**, 00110100010 01101111011 00110110111 00010111001
11101110111 11000110101 11010011001 11001101011
01111011011 00001001100 10111000110 0010011)

(**3**, 01000010010 11101110101 11001110100 00111010000
11110010010 10001000000 00111001011 10101101111
00001010010 10000011011 01100000100 1111001)

Complete the last 4 bits with ID numbers in binary

(**1**, 0001010101O 1111010000l 0101001010O 1010111100l
0010001011l 1111011100l 1011010100l 0000010111O
1111111100l 0010111011l 1011100110O 101010**0001**)

(**2**, 0011010001O 0110111101l 0011011011l 0001011100l
1110111011l 1100011010l 1101001100l 1100110101l
0111101101l 0000100110O 1011100011O 001001**0010**)

(**3**, 0100001001O 1110111010l 1100111010O 0011101000O
1111001001O 1000100000O 0011100101l 1010110111l
0000101001O 1000001101l 0110000010O 111100**0011**)

Convert to seedphrase.

best vintage family quality carry warm release alarm you confirm ridge popular

crowd hunt dad blame upon shop spring sniff kiwi another rhythm chaos

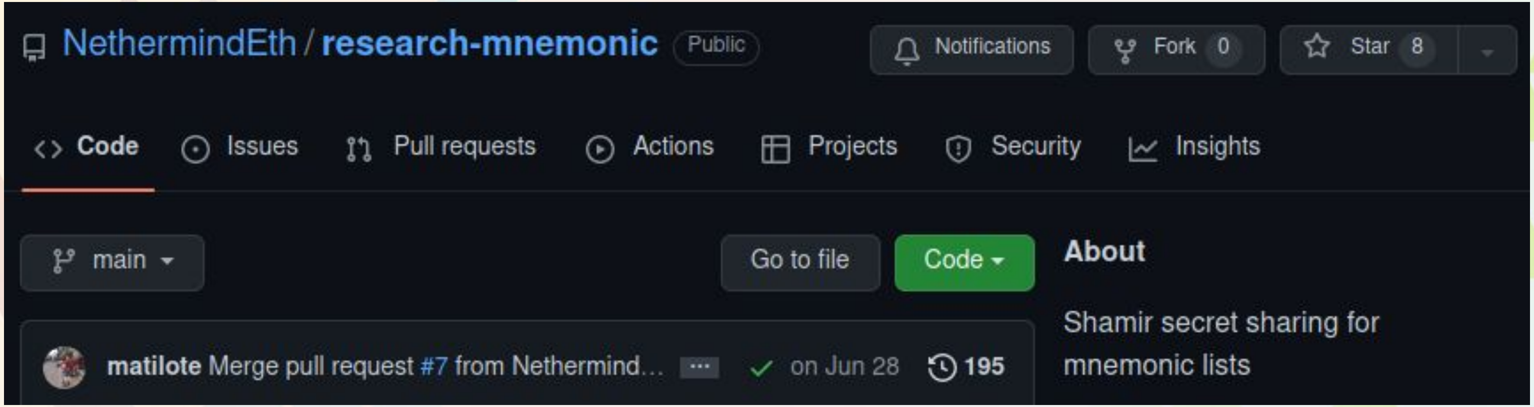
start town twenty liar fence clap van memory welcome twice elevator pen

Last 4 bits in the last word encode the ID number!

Popular = 1010100**0001**

Chaos = 0010011**0010**

Pen = 1111001**0011**



NethermindEth / research-mnemonic Public

matilote Merge pull request #7 from Nethermind… ✓ on Jun 28 195

About

Shamir secret sharing for mnemonic lists

Thank you!

Jorge Arce

Blockchain and Cryptography Researcher

jorge.arce-garro@nethermind.io

@0xjorgeth