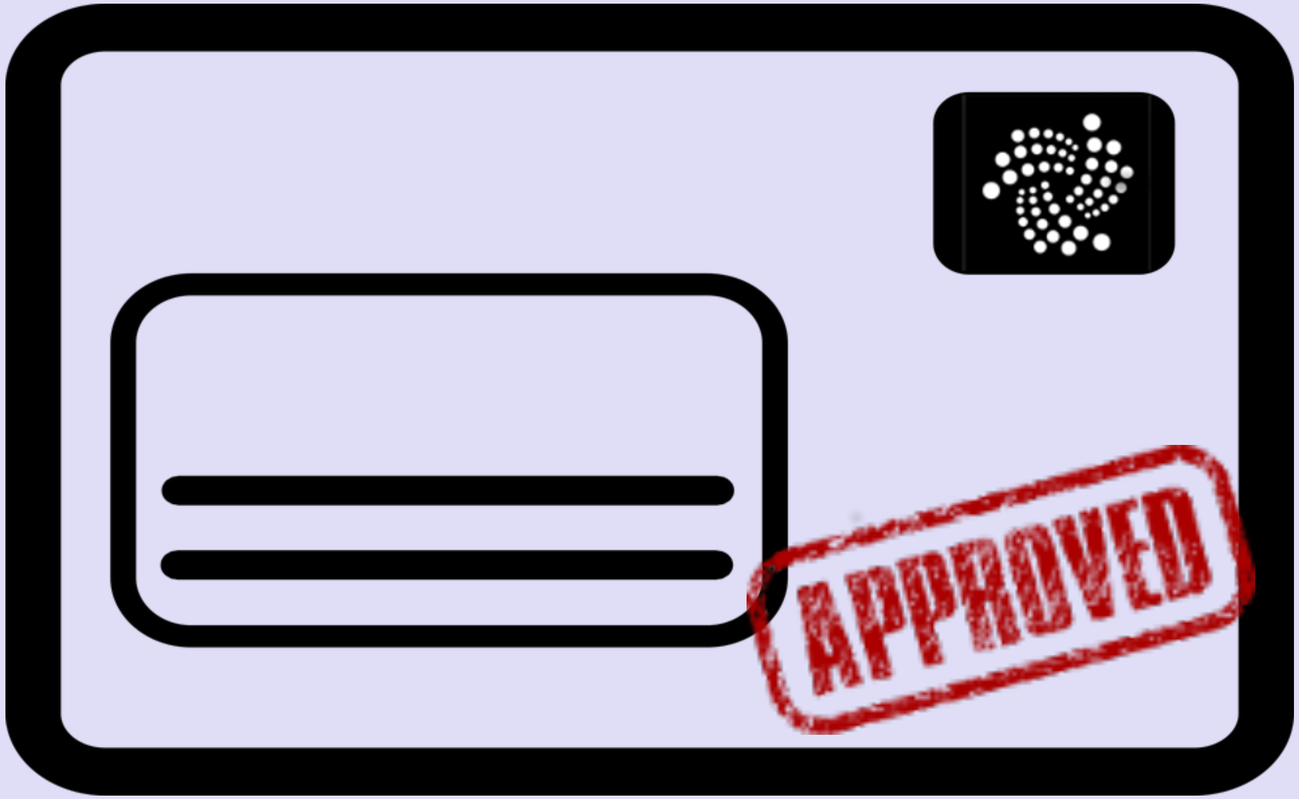
IOTA TUTORIAL 9

Address & Checksum



INTRO

- In this video I will explain how IOTA addresses are generated and how address checksums are calculated.

ADDRESS

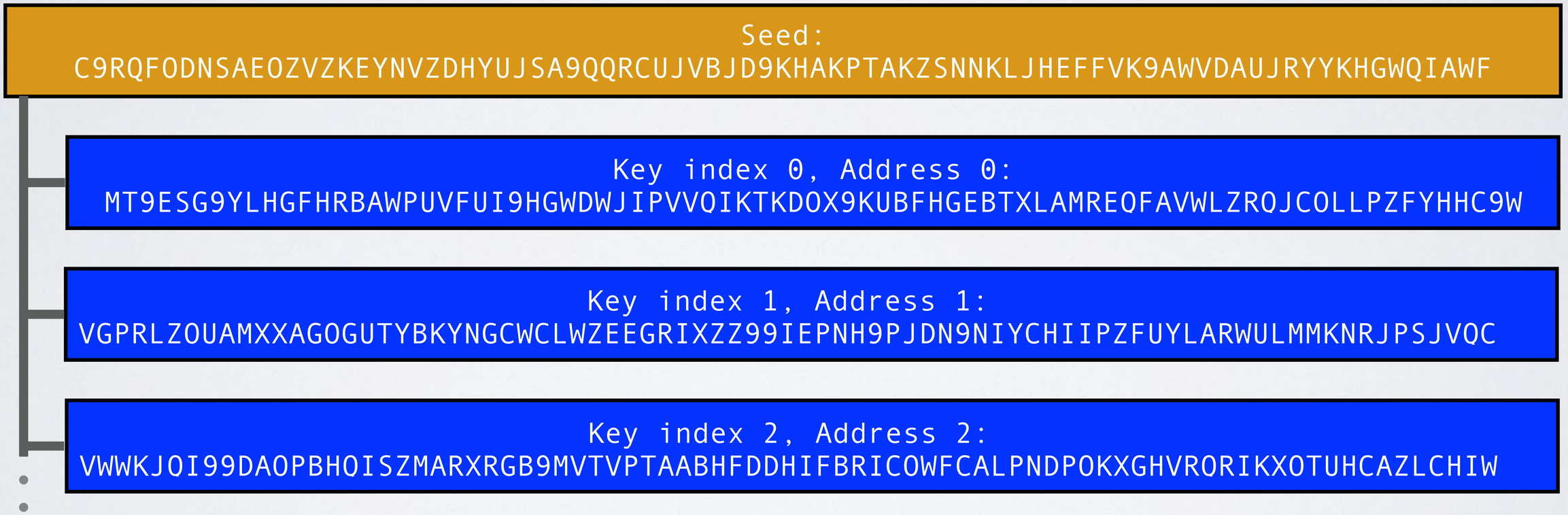
- The procedure to generate IOTA addresses is as follows:

- IOTA addresses are deterministically generated starting with the seed (81 trytes).
  Seed (trytes): **C9RQF ... QIAWT**

- Convert the seed (81 trytes) to trits (= 81 x 3 = 243 trits)
  Seed (trits): **0,1,0,0,0,0 ... -1,-1,0,-1,1,-1**

ADDRESS

- Every address has a corresponding key index number. A key index number is an integer starting from 0. Address 0 has key index number 0, address 1 has key index number 1, etc.



ADDRESS

- They key index number always starts with integer 0, and is simply incremented in order to get the next address.

- The largest key index number allowed is 9007199254740991 (9,007,199,254,740,991).

- This largest key index number is the same as $2^{53} - 1$, which is the same as the Javascript constant: Number.MAX_SAFE_INTEGER

- An IOTA seed can generate in total **9007199254740992** addresses.

- The decimal key index number must be converted to trits.
  For example the key index number 1 converted to trits looks like: **1,0,0**

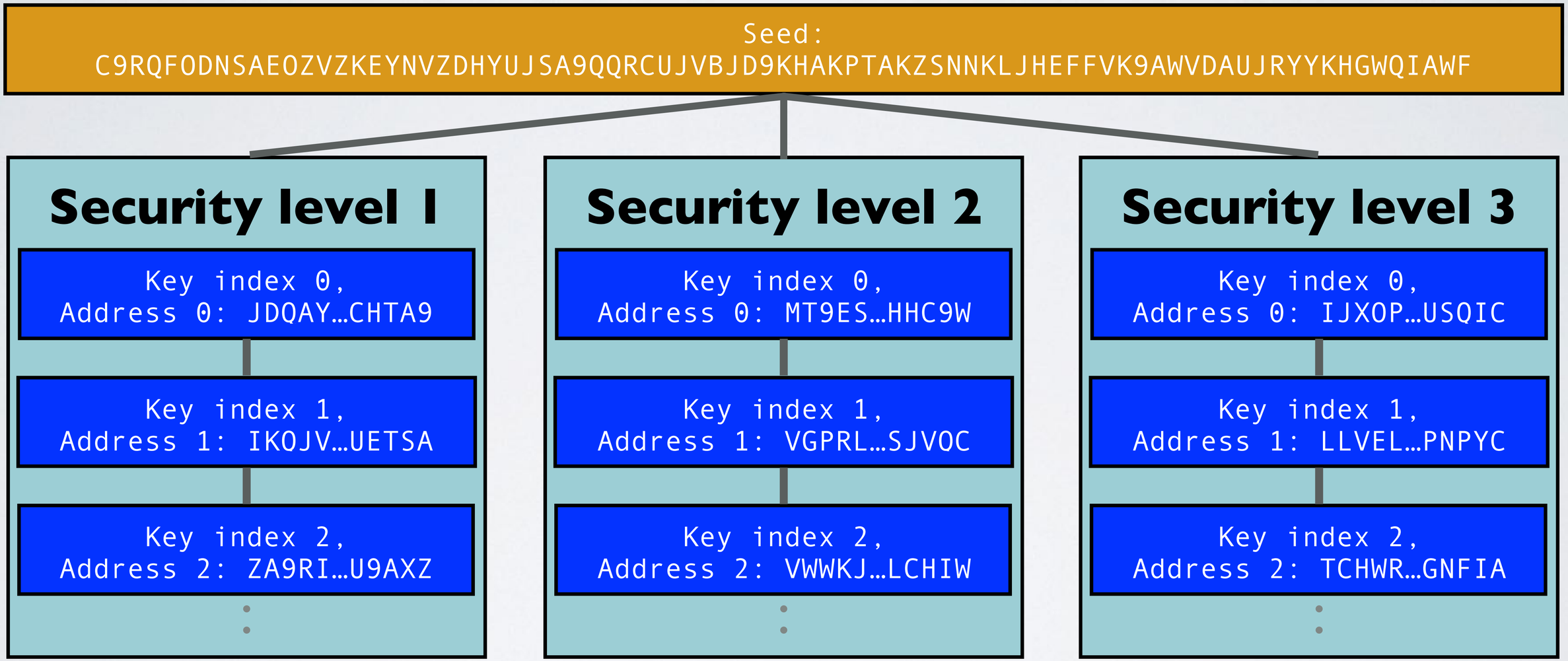
ADDRESS

- Next create a subseed by adding the key index number and seed together.
  subseed = seed + key index number

```
Seed:                0,1,0,0,0,0 ... -1,-1,0,-1,1,-1
Key index number:    1,0,0
                     ------------------------------- +
Subseed:             1,1,0,0,0,0 ... -1,-1,0,-1,1,-1
```

- IOTA provides 3 security levels: 1, 2 or 3.
  A security level determines the number of rounds for hashing, which means that a single seed can have 3 different accounts.
  A different security level with the same index number, means that you will get a different address.

ADDRESS


Seed:
C9RQFODNSAEOZVZKEYNVZDHYUJSA9QQRCUJVBJD9KHAKPTAKZSNNKLJHEFFVK9AWVDAUJRYYKHGWQIAWF
Security level 1
Key index 0,
Address 0: JDQAY…CHTA9
Key index 1,
Address 1: IKOJV…UETSA
Key index 2,
Address 2: ZA9RI…U9AXZ
Security level 2
Key index 0,
Address 0: MT9ES…HHC9W
Key index 1,
Address 1: VGPRL…SJVOC
Key index 2,
Address 2: VWWKJ…LCHIW
Security level 3
Key index 0,
Address 0: IJXOP…USQIC
Key index 1,
Address 1: LLVEL…PNPYC
Key index 2,
Address 2: TCHWR…GNFIA

ADDRESS

Security Level	Key size (trits)	Remark
1	6561 x 1	Used for low security (for very high efficiency). Best for tiny IoT devices that only transact/store small amounts of value.
2	6561 x 2	Used for standard security (for medium performance). Best for regular people's wallets and devices that store higher amounts of value.
3	6561 x 3	Used for full blown quantum proof security that conforms to National Security Agency's (NSA) recommendations for sensitive material. Good for big value transactions and paranoids.

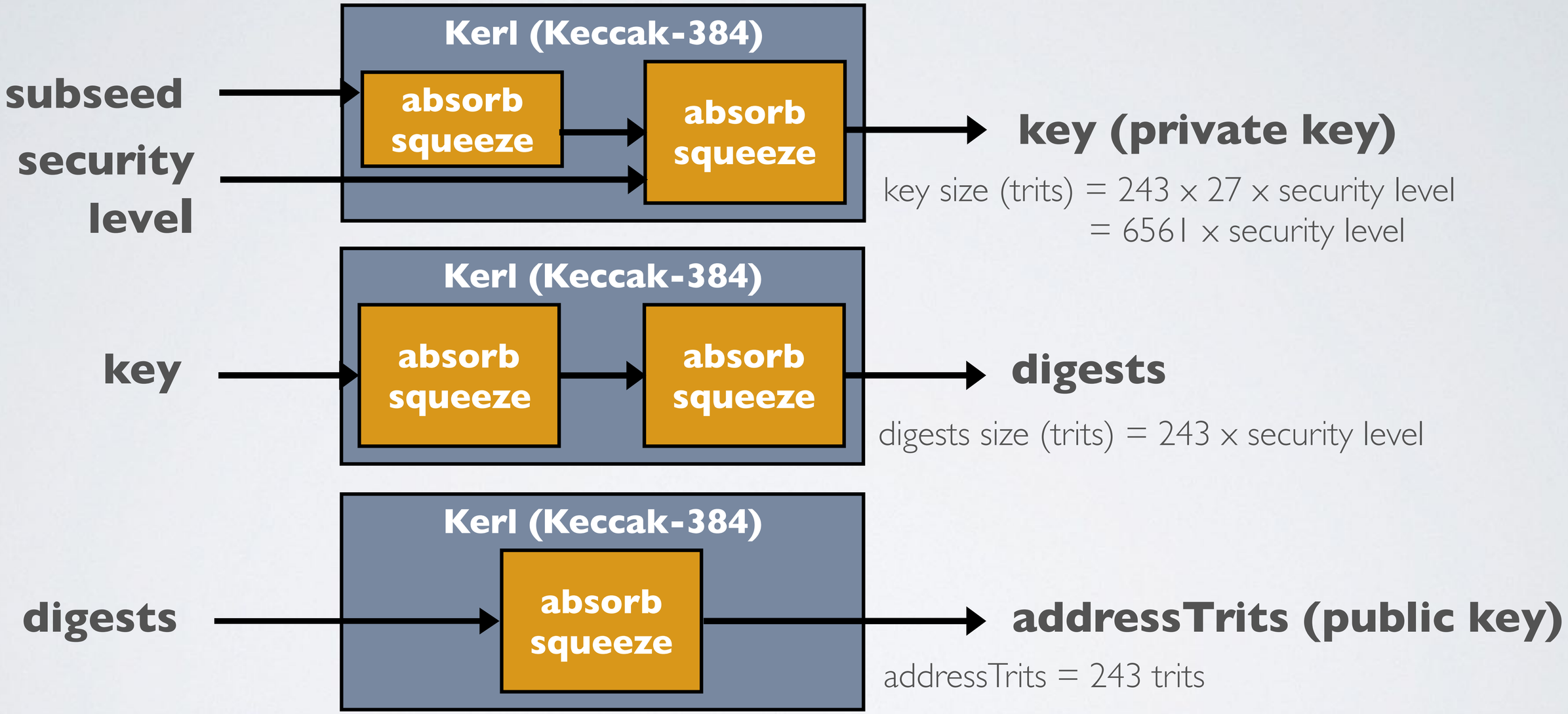
- Client libraries, such as iota.lib.js makes it possible to choose another security level.
  See: https://www.mobilefish.com/services/cryptocurrency/iota_wallet.html

ADDRESS

- By default the IOTA light wallet uses security level 2 and you can not change its security level. If you created an address using security level 1 or 3 this address will not appear in the IOTA light wallet using the same seed.

- In the next slide a simplistic explanation is given how the subseed is hashed multiple times using the Keccak-384 hash algorithm. The hashing is done in a wrapper class called Kerl.

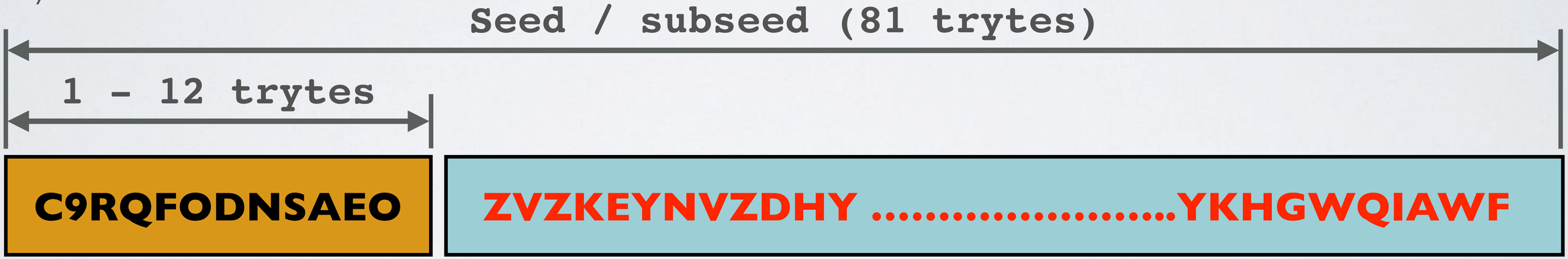
ADDRESS


Kerl (Keccak-384)
absorb
squeeze
absorb
squeeze
subseed
security
level
key (private key)
key size (trits) = 243 x 27 x security level
= 6561 x security level
Kerl (Keccak-384)
absorb
squeeze
absorb
squeeze
key
digests
digests size (trits) = 243 x security level
Kerl (Keccak-384)
absorb
squeeze
digests
addressTrits (public key)
addressTrits = 243 trits

ADDRESS

- As mentioned earlier the key index number is added to the seed to create the subseed.

- The seed and subseed can differ between the first 1 tryte up to and including 12 trytes.



ADDRESS

- If someone else has exactly the last 69 (= 81 - 12) trytes up to and including 81 trytes of your seed they can see the balance of one or more of your addresses.

- The probability that someone else happens to have the same last 69 trytes of your seed is very small. Here is the proof:
  IOTA seed with only 69 trytes has $27^{69} = 5.80 \times 10^{98}$ possible combinations.

- For comparison:
  A Bitcoin private key with 256 bits has $2^{256} = 1.15 \times 10^{77}$ possible combinations.
  This means, even if you have an IOTA seed with only 69 trytes it has more possible combinations than a Bitcoin private key.

ADDRESS CHECKSUM

- A checksum is an additional 9 trytes added to an address (81 trytes) which can be used to validate the integrity and validity of the address.

- An address with checksum is 90 trytes long, 81 trytes for the address itself and 9 trytes for the checksum. For example:
  9AMLQAQURNSXWHCMZYDTSSXKF9M9EIFERLHJRNTKYYRTFFWGRPNCWSC
  CHOBQTQX9UBKMDASIKCYSPSNI9**Y9WFPBMPP**

- The iota.api.getNewAddress API function (see library iota.lib.js) makes it possible to directly return checksum'ed addresses.

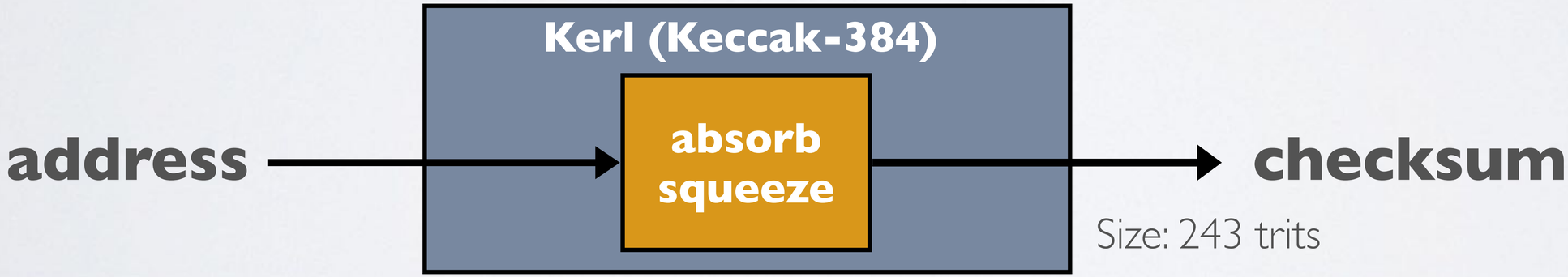
ADDRESS CHECKSUM

- The procedure to calculate an address checksum is as follows:

- Start with an IOTA address (81 trytes).
  Address (trytes): **`FSAFM ... NVDZC`**

- Convert the address (81 trytes) to trits (= 81 x 3 = 243 trits)
  Address (trits): **`1,0,-1,1,0,-1 ... -1,0,0,0,1,0`**

- The address is hashed using the Keccak-384 hash algorithm.



ADDRESS CHECKSUM

- Convert the address checksum (243 trits) to trytes (81 trytes):
  ...PJFNYWVUGKPRTRV

- Get the last 9 trytes: **VUGKPRTRV**

- Append the last 9 trytes to the original address:
  FSAFM ... NVDZCVUGKPRTRV

- The address including checksum has a length of 81 + 9 = 90 trytes.

ADDRESS CHECKSUM

- The IOTA light wallet:
  - Always creates addresses including the checksum. The addresses are always 90 trytes long.
  - Always requires receive addresses, with valid checksums when making a transaction. The receive addresses must be 90 trytes long.