The WiFi password is in the beacon

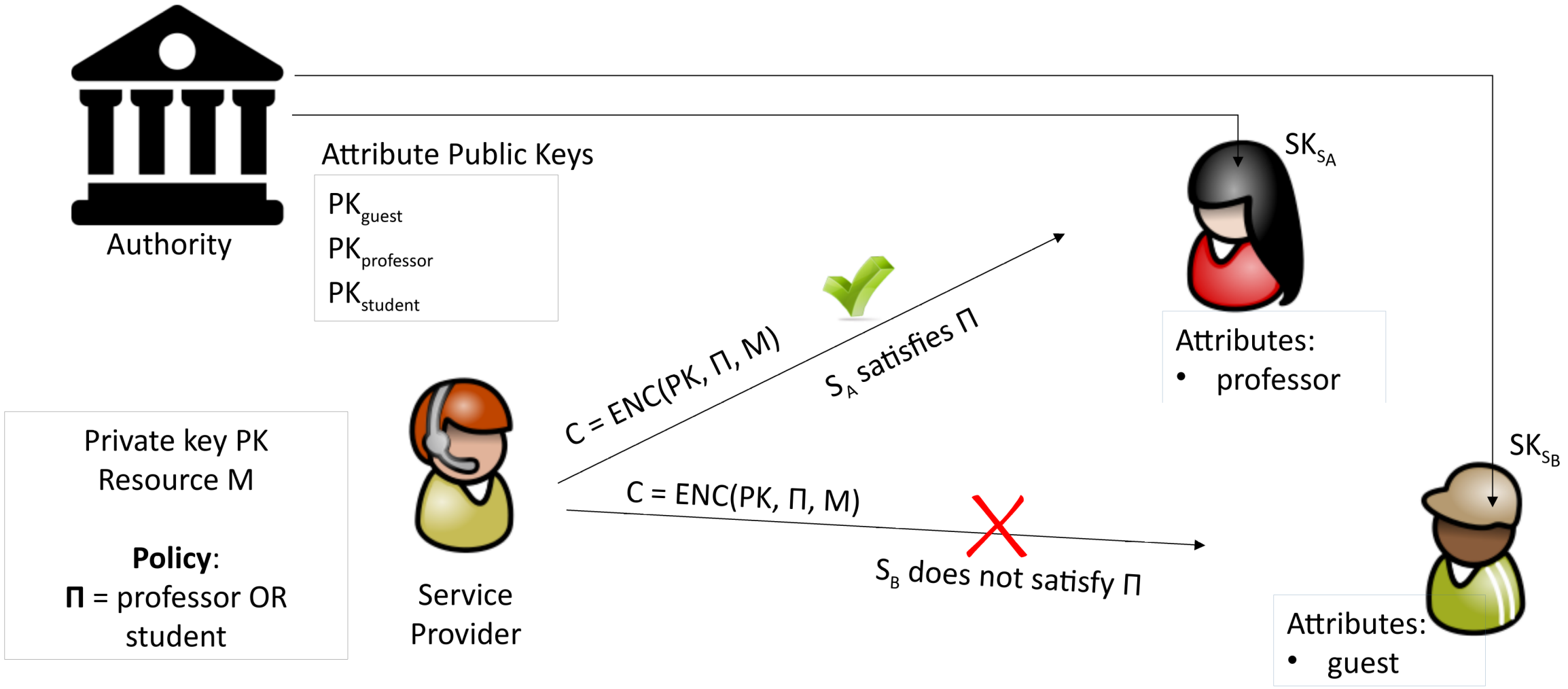
Claudio Pisa
clauz@ninux.org

Wireless Battlemesh v13

AA in Protected WLANs

- Access control for Large WLAN deployments is non-trivial
  - AA Infrastructure deployment is non-trivial
    - e.g. RADIUS-based federations
  - User authentication hinders the privacy of the users

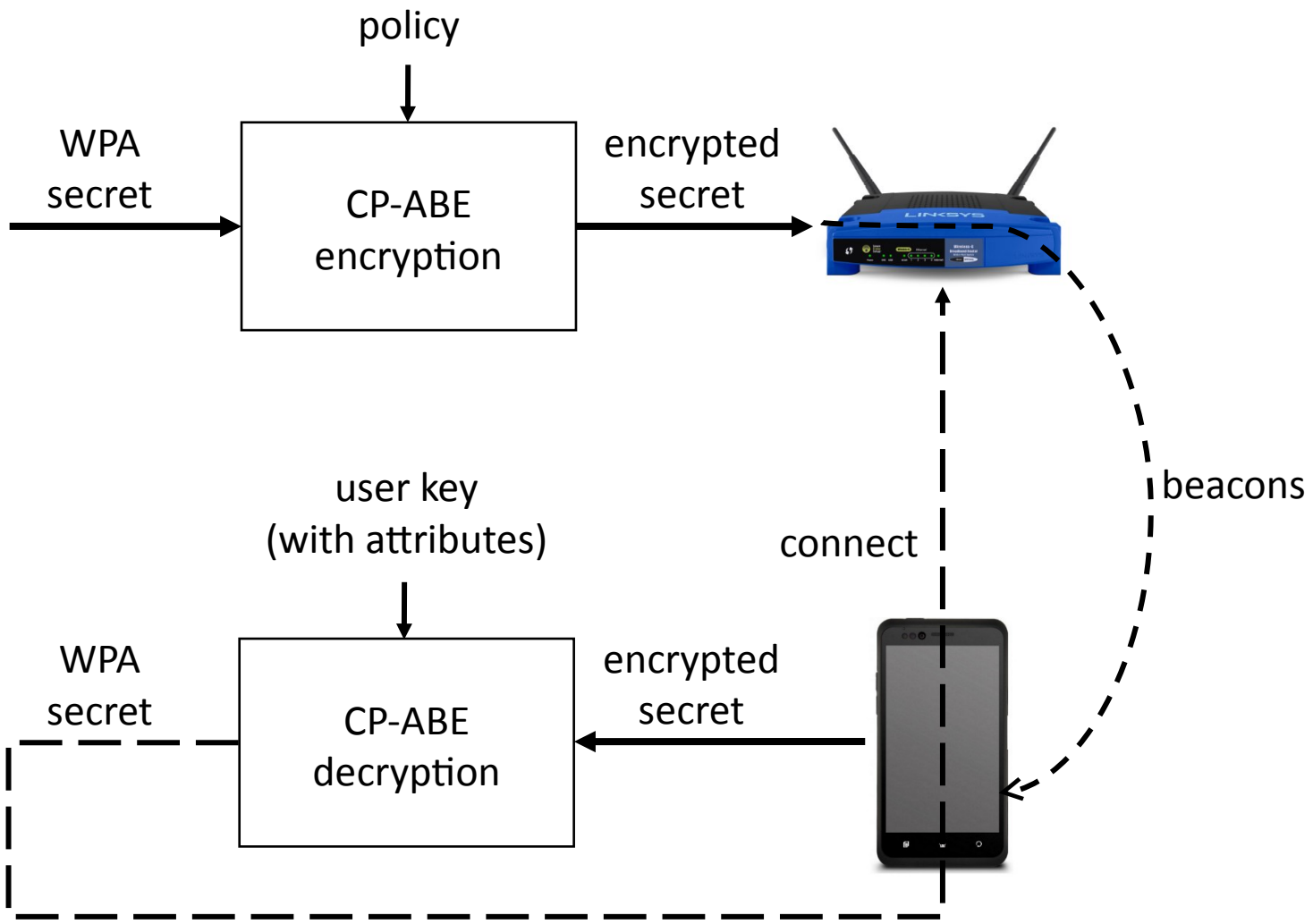
Ciphertext Policy Attribute-Based Encryption (CP-ABE)



Fountain Codes

- Fountain codes aka rateless erasure codes
  - class of erasure codes with the property that a potentially limitless sequence of encoding symbols can be generated from a given set of source symbols
    - the original source symbols can ideally be recovered from any subset of the encoding symbols
- Collect enough encoded chunks until you can decode the message

An ABAC Scheme for Protected WLANs



- Random WPA2 secret generated and changed very often
  - e.g. every 20 seconds
- Fountain Coding used to transmit encrypted secrets
  - Too big to fit in the beacons' information elements

Demo


[root@AccessPoint]# ./ap.sh
Access Point
[root@Station]# ./sta.sh
Station

Components

- AP side
  - hostapd modified to:
    - receive the WPA2 password from a named pipe
    - receive the information elements for the beacons from a named pipe
    - not disconnect STAs on configuration change / SIGHUP
  - daemon providing random passwords + encoded chunks for information elements
    - feeding the named pipes
- STA side
  - “iw scan” wrapper
  - daemon taking chunks from a named pipe and trying to decode and decrypt

More information

- C. Pisa, A. Caponi, T. Dargahi, G. Bianchi, and N. Blefari-Melazzi. “WI-FAB: attribute-based WLAN access control, without pre-shared keys and backend infrastructures.” HotPOST 2016
- C. Pisa, T. Dargahi, A. Caponi, G. Bianchi, and N. Blefari-Melazzi. “On the feasibility of attribute-based encryption for WLAN access control.” WiMob 2017
- https://bitbucket.org/cnit-recred/wifab/
- https://en.wikipedia.org/wiki/Fountain_code

Thank you

Claudio Pisa
clauz@ninux.org

H2020 – Grant Agreement no. 653417 - ReCRED