WORKSHOP GARR 2026

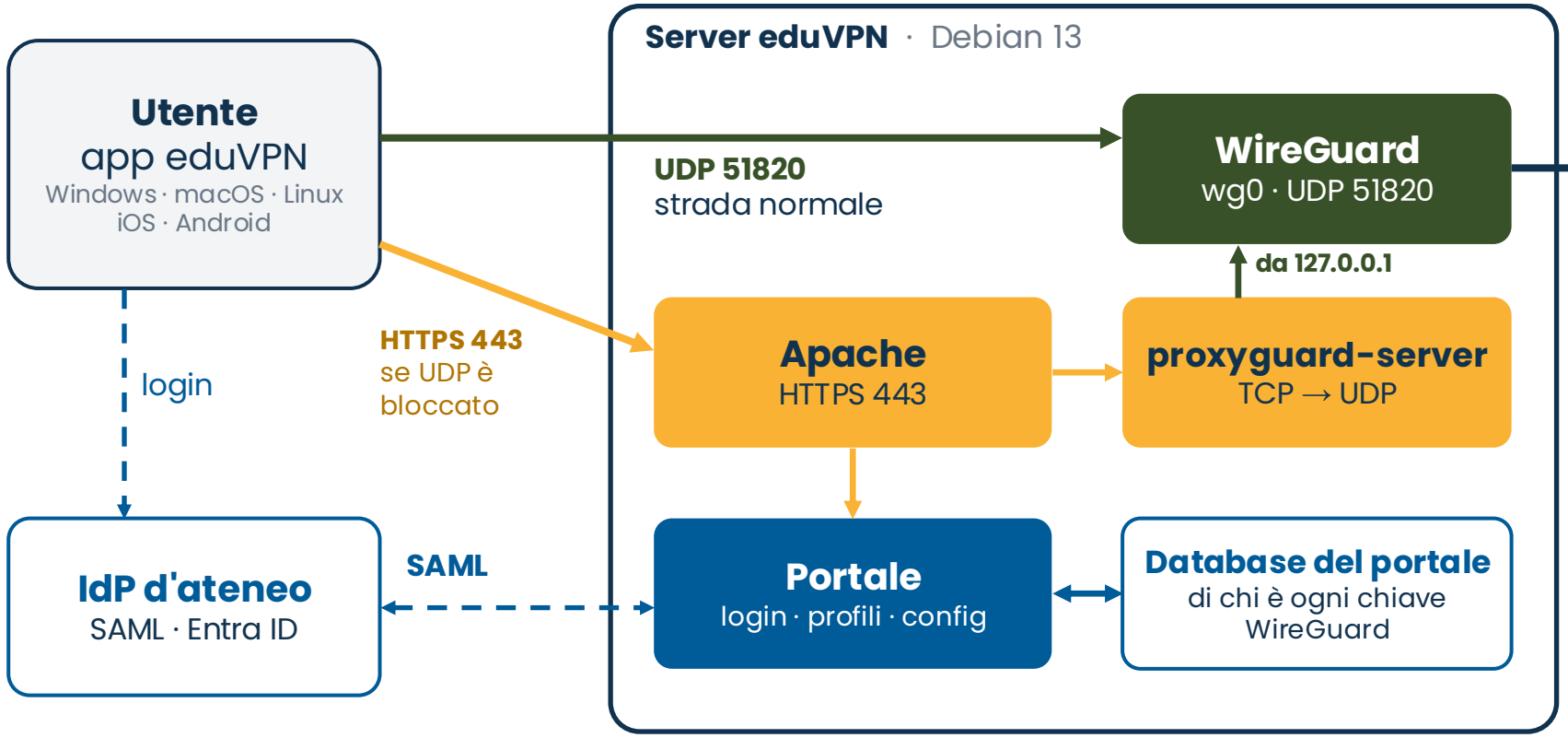
NET MAKERS

eduvpn-logger: chi, da dove e quando nelle sessioni WireGuard di eduVPN

Giacomo Camata

Università degli Studi di Trieste

La mia tesi: eduVPN in produzione a UniTS



Senza NAT

il server instrada soltanto: NAT e policy sul firewall

6 profili

full e split tunnel, in base al ruolo

Institute Access · solo WireGuard · **il limite l'ho trovato mettendolo in produzione**

“Chi c'era dietro 42.42.42.42 ieri alle 22:14?”

IP di fantasia

Indirizzo di provenienza: “**currently only available when clients connect using OpenVPN**”
docs.eduvpn.org · server v3 · logging

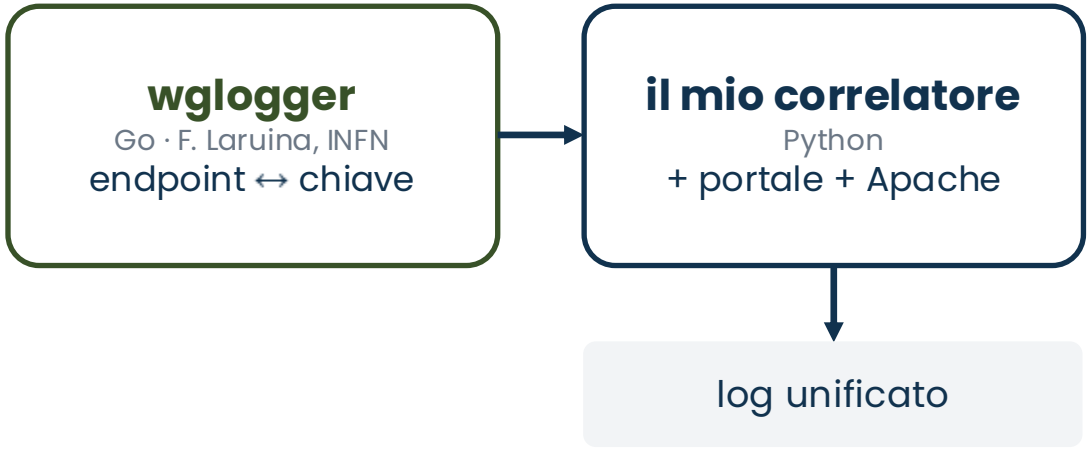
Obiettivo: l'IP di ogni utente nel log, appena si connette

I dati ci sono, ma sparsi

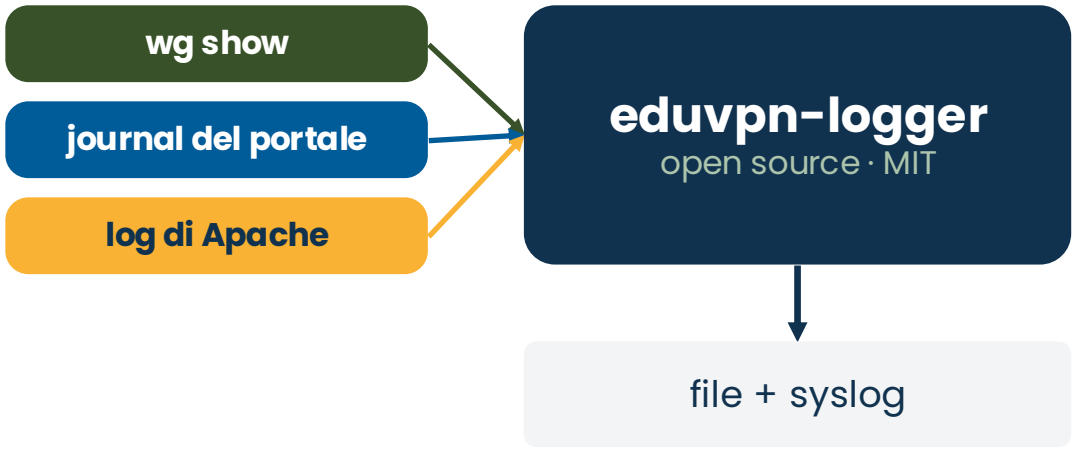
	Portale	WireGuard	Apache · ProxyGuard
Chi è utente	✓ utente e profilo	✗ solo una chiave	✗ mai, nemmeno con ProxyGuard
Da dove IP pubblico	✗ solo per OpenVPN	◐ sì, ma 127.0.0.1 dietro ProxyGuard	◐ solo per chi usa ProxyGuard (TCP)
Quando inizio e fine	◐ solo se l'app si disconnette	✗ nessuna connessione, nessun log	✗ scrive a fine tunnel, giorni dopo

Da dove sono partito

Prima versione: due programmi



Oggi: un solo servizio



Da script interno a software pubblico:

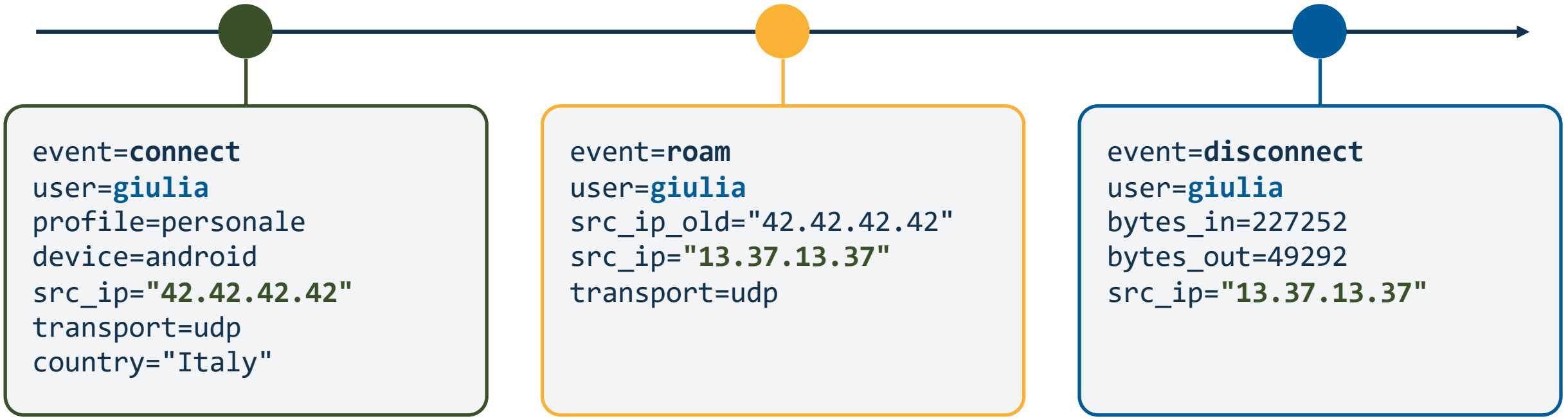
- abbinamento ProxyGuard
- sessioni ai riavvii
- anti-falsificazione
- installer
- test su 8 distro

Una riga per evento, pronta per il SIEM

09:58 da casa

10:41 rete mobile

11:02 fine sessione



```
event=connect
user=giulia
profile=personale
device=android
src_ip="42.42.42.42"
transport=udp
country="Italy"
```

```
event=roam
user=giulia
src_ip_old="42.42.42.42"
src_ip="13.37.13.37"
transport=udp
```

```
event=disconnect
user=giulia
bytes_in=227252
bytes_out=49292
src_ip="13.37.13.37"
```

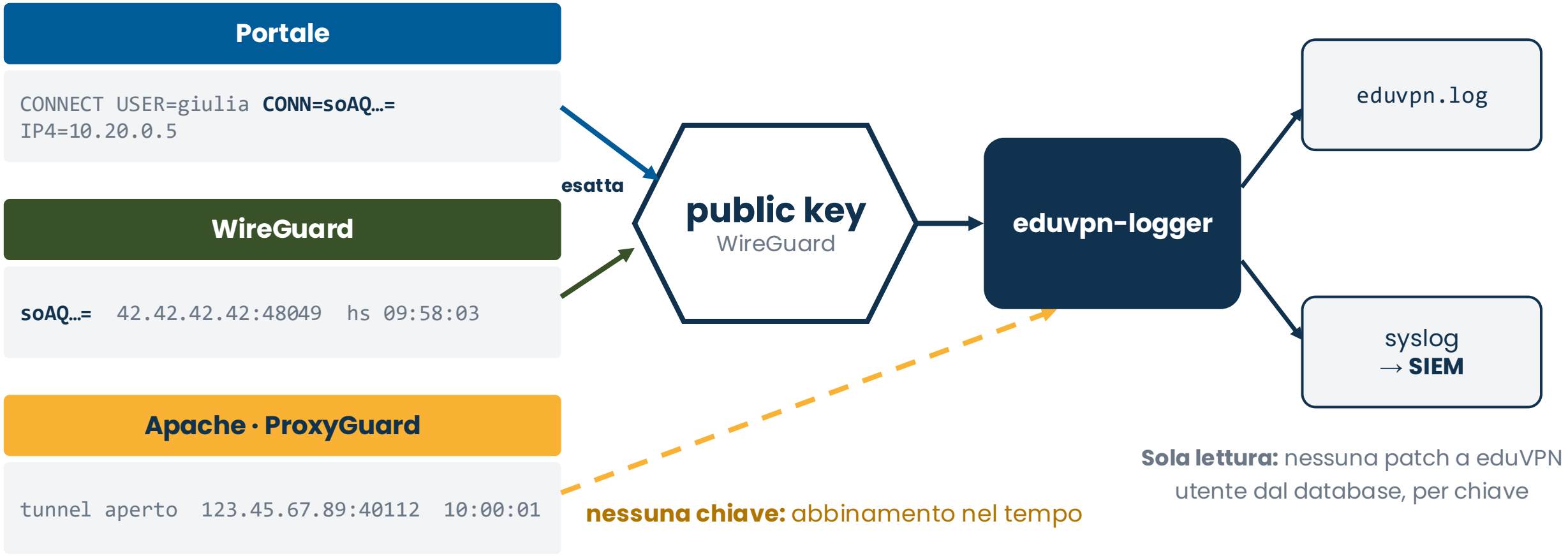
chiave=valore → file + syslog → qualunque SIEM

Ogni campo ha la sua fonte

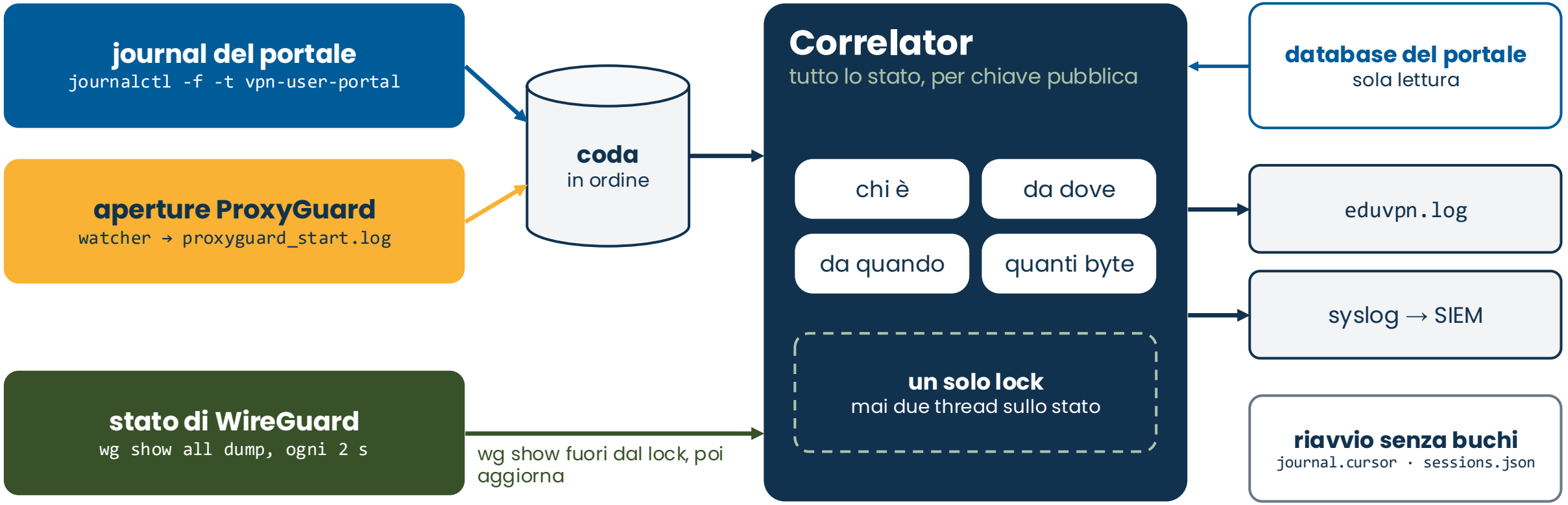
```
2026-04-15T12:10:04+02:00 event=connect user=marco profile=studenti device=windows
conn=GUUepz8z…= tunnel_ip4="10.20.1.9" src_ip="123.45.67.89" src_port=40112
transport=tcp tcp_match=time tcp_candidates=1 country="Austria" city="Vienna"
```

Portale	WireGuard	Apache · ProxyGuard	GeoIP	eduvpn-logger
user · profile · device conn = chiave pubblica tunnel_ip4 / ip6	src_ip · src_port inizio, roam, fine byte trasferiti	src_ip dei tunnel TCP tcp_match tcp_candidates	country · city facoltativo	ts · event transport inferred=1 se dedotto
fonte: **journal + database**	fonte: **wg show ogni 2 s**	fonte: **log di Apache**	fonte: **MaxMind, offline**	fonte: **calcolati**

Due fonti si uniscono sulla chiave, la terza sul tempo

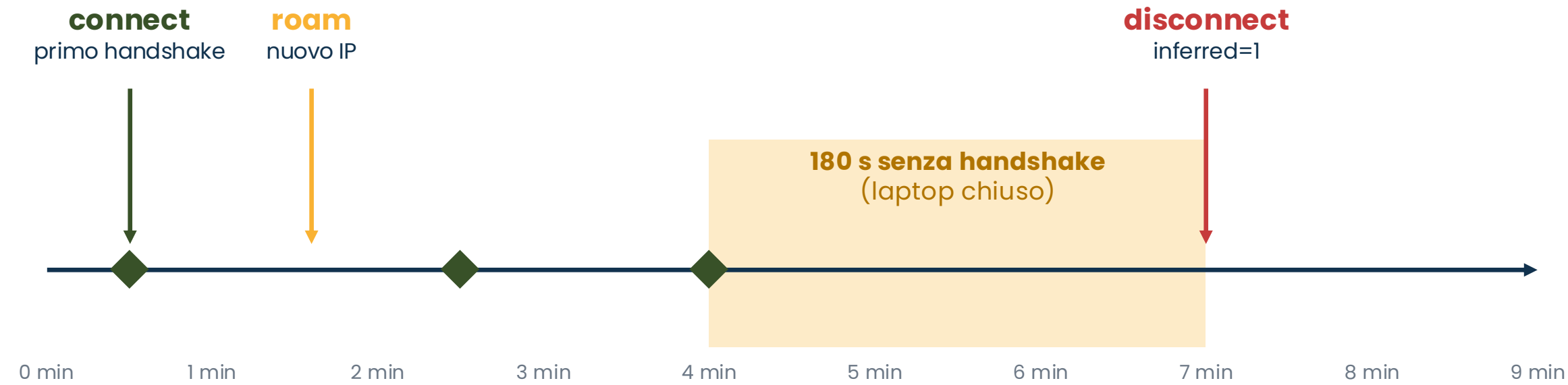


Dentro il logger: tre lettori, uno stato



Solo libreria standard Python · GeoIP facoltativo

WireGuard non ha connessioni: le ricavo dallo stato



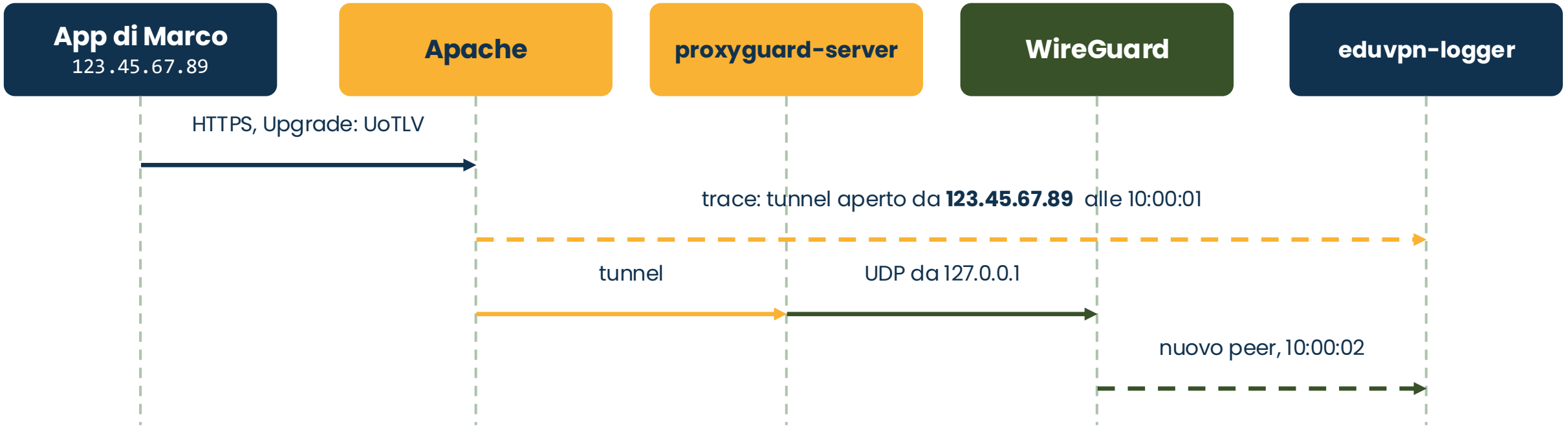
◆ handshake WireGuard (circa ogni 2 min se c'è traffico), stato letto con `wg show` ogni 2 s

connect
primo handshake recente: la riga aspetta fino a 10 s di sapere chi è l'utente

roam
cambia l'IP. Se cambia solo la porta è il NAT: nessuna riga. Al massimo uno ogni 30 s

disconnect
dal portale, oppure dopo 180 s senza handshake: WireGuard ha scartato le chiavi

Dietro ProxyGuard, WireGuard vede solo 127.0.0.1

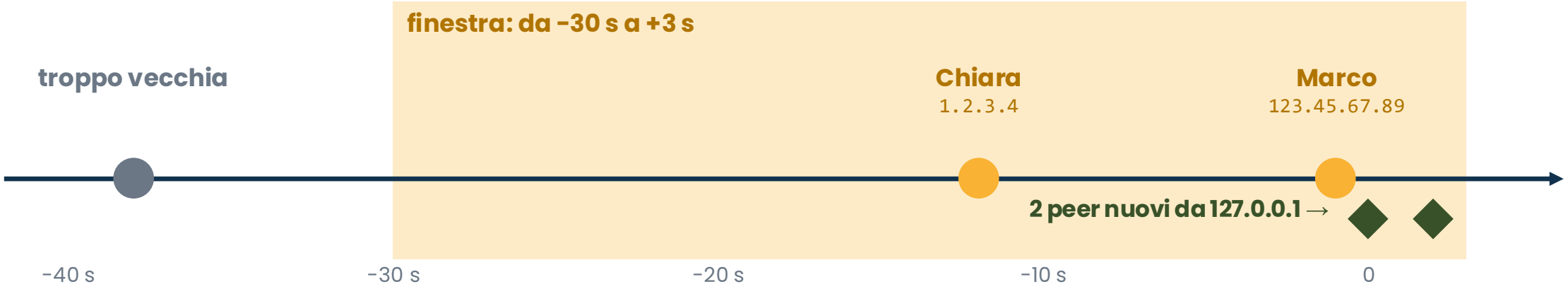


Apache: sa l'IP vero, non la chiave

WireGuard: sa la chiave, non l'IP vero

In comune c'è solo il tempo

L'abbinamento nel tempo, dichiarato nel log



peer delle 10:00:02 · la più vicina; c'era anche Chiara

```
src_ip="123.45.67.89"  tcp_candidates=2
```

peer delle 10:00:04 · Marco è già preso: resta Chiara

```
src_ip="1.2.3.4"  tcp_candidates=1
```

① l'apertura più vicina nel tempo

② ogni apertura assegnata una volta sola

③ tcp_candidates = quante erano possibili

Costruito sul comportamento reale di eduVPN

Comportamenti reali

Modifiche alla configurazione
apply-changes ricrea i peer: la sessione continua

NAT rebinding
cambia solo la porta: non è uno spostamento

Rete mobile instabile
al massimo un roam ogni 30 s, l'ultimo IP mai perso

Chiave riusata
chiusa e riaperta via API: sessione nuova, nulla ereditato

Sicurezza e privacy

Anti-falsificazione
eventi del portale solo da account di sistema: l'UID lo scrive journald

User ID ostili
chi imita il formato del log viene scartato; l'utente si prende dal DB

Sandbox systemd
gira come root, ma con 4 capability e senza rete

Dati personali
log leggibili solo dagli amministratori, poi cancellati

Verificato con **un eduVPN 3 vero in laboratorio**, test su 8 distribuzioni e un fuzzer da 2.500 scenari

Un comando per installarlo

```
$ sudo ./install.sh
```

Debian 13

Ubuntu 22.04 · 24.04 · 26.04

AlmaLinux 9 · 10

Fedora 43 · 44

1 Pacchetti

installa solo quelli mancanti, senza aggiornare gli altri

2 Servizi

logger e watcher in sandbox, log conservati 180 giorni, syslog

3 Portale

accende il log delle connessioni, dopo php -l e backup

4 Apache

trace solo per ProxyGuard; se configtest fallisce, torna indietro

5 Controllo

alla fine stampa Ready, oppure cosa sistemare

Idempotente: rilanciato aggiorna · `--uninstall` toglie tutto · GeoIP facoltativo

In produzione all'Università di Trieste

Dove si ferma, e come andare oltre

Limite	Perché	Come si supera
⚠ **Un solo nodo**	journal, database e wg show letti sulla stessa macchina	→ un agente per ogni nodo VPN e un correlatore centrale vicino al portale
⚠ **Campionamento**	WireGuard letto ogni 2 s: uno spostamento più breve non si vede	→ si può scendere a 0,5 s; per un log di audit 2 s bastano

Il log dice sempre come ha ottenuto il dato: `transport · tcp_match · inferred`

ProxyGuard sta cambiando, e il logger con lui

Fino a 2.1.0

Abbinamento nel tempo

Apache vede l'IP, WireGuard la chiave: in comune c'è solo il tempo

`tcp_match=time`

resta come riserva

→

ProxyGuard 2.1.1

Abbinamento esatto

Con -logip il logger segue le porte da Apache fino al peer

(in eduVPN: ProxyGuard 2.2.0)

`tcp_match=port`

✓ già nel logger

→

TLS sui nodi

Senza Apache davanti

Una sola riga di ProxyGuard, sul nodo, contiene già l'IP del client

`2.2.0 · portale 3.16`

→ serve il multi-nodo

Grazie

Domande dal pubblico
www.partici.fi · codice **83995288**

github.com/giacomocamata/eduvpn-logger
open source · licenza MIT

Giacomo Camata · Università degli Studi di Trieste

WORKSHOP GARR 2026
NET MAKERS