Utilizzare IDEM per controllare l'accesso wireless

Case Study:
la rete wireless dell'Università di Ferrara

Introduzione

- Wi-Fe è il servizio di connettività wireless dell'Università di Ferrara
- Il sistema è attivo da febbraio 2004
- Offre la copertura di tutte le aree didattiche e parte dei dipartimenti: installati oltre 250 access point
- Circa 3000 differenti utenti, su una popolazione di 17000 studenti, accedono quotidianamente al servizio

Obiettivi

- Offrire il servizio alla federazione IDEM
- Integrare l'infrastruttura esistente con l'architettura Shibboleth
- Rispettare le normative in tema di sicurezza informatica (logging degli accessi e del traffico)

Architettura di Wi-Fe

Wi-Fe: infrastruttura

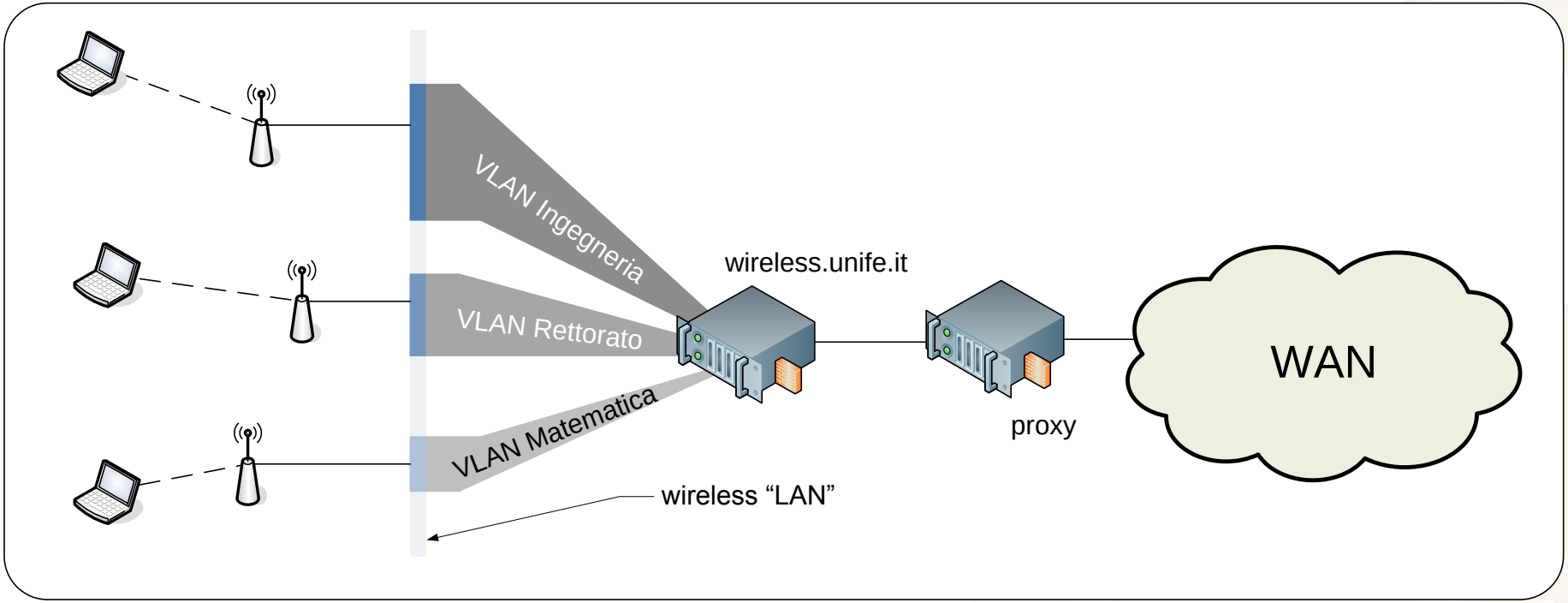
- Non vengono utilizzati sistemi di tipo proprietario (no WLAN controller o lightweight access point)
- Un sistema opensource (Linux based) gestisce in modo centralizzato autenticazione, politiche di autorizzazione e controllo del traffico.

Wi-Fe: infrastruttura di rete

- E' stato scelto di non utilizzare chiavi WEP/WPA per aumentare al massimo la fruibilità del sistema da parte degli utenti.
- Tutto il traffico wifi è convogliato verso il server centrale che adempie a:
  - DHCP server
  - Bridge
  - Firewall
  - Router

Wi-Fe: infrastruttura di rete

- Ciascuna sede ha una VLAN dedicata al traffico wireless. Tutte le VLAN sono riunite in un bridge.



Wi-Fe: autenticazione e log

- Il servizio è autenticato tramite un sistema "captive portal". Esso effettua il logging di tutti gli accessi.
- Il traffico WEB viene monitorato attraverso un transparent proxy.
- Il traffico non WEB viene monitorato attraverso l'uso di iptables.

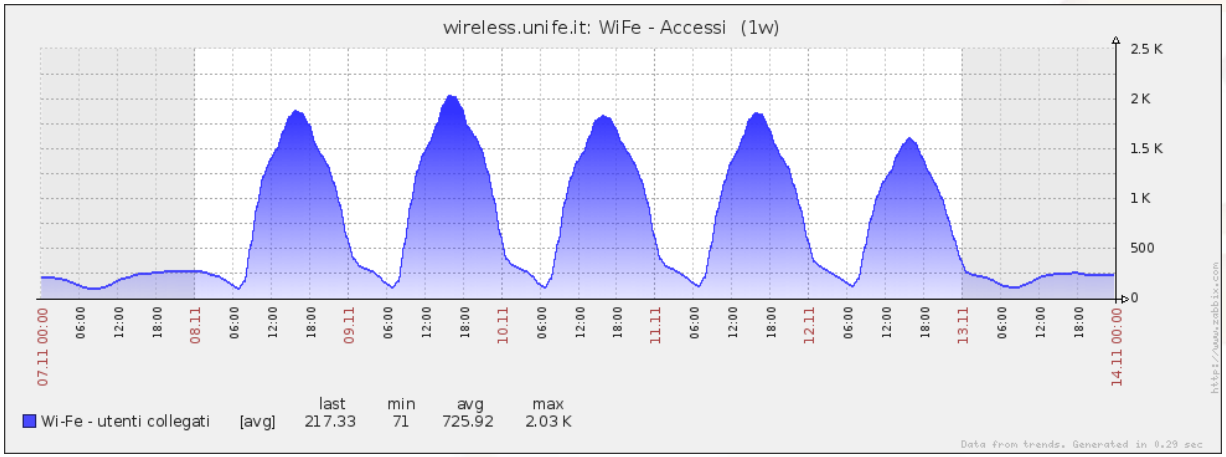
Prestazioni

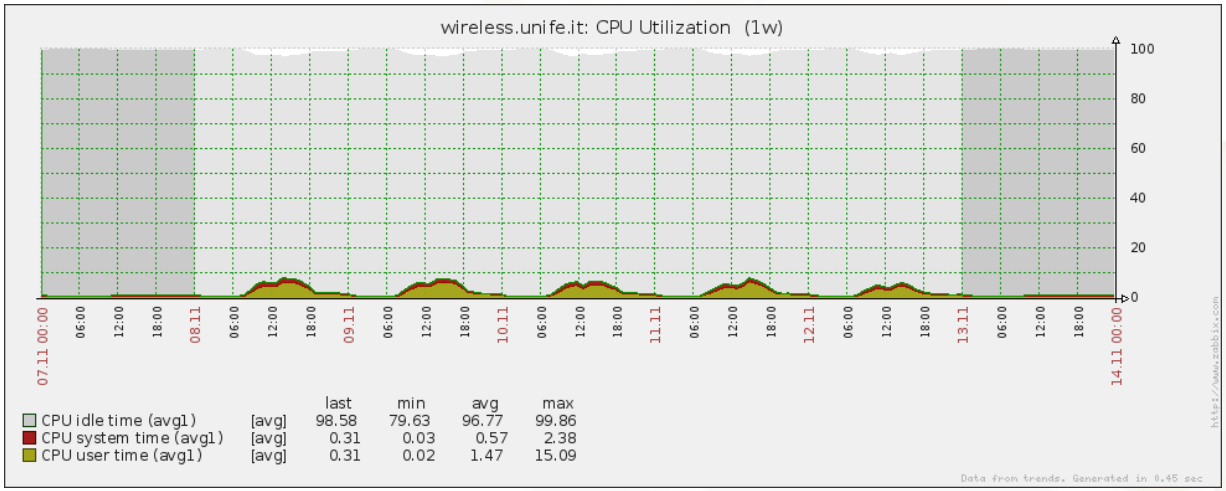
Hardware:

- CPU: 2 Xeon 2Ghz a 4 core
- RAM: 12Gb

Prestazioni:

- Circa 1500 accessi contemporanei nelle ore di picco
- Carico CPU < 20%





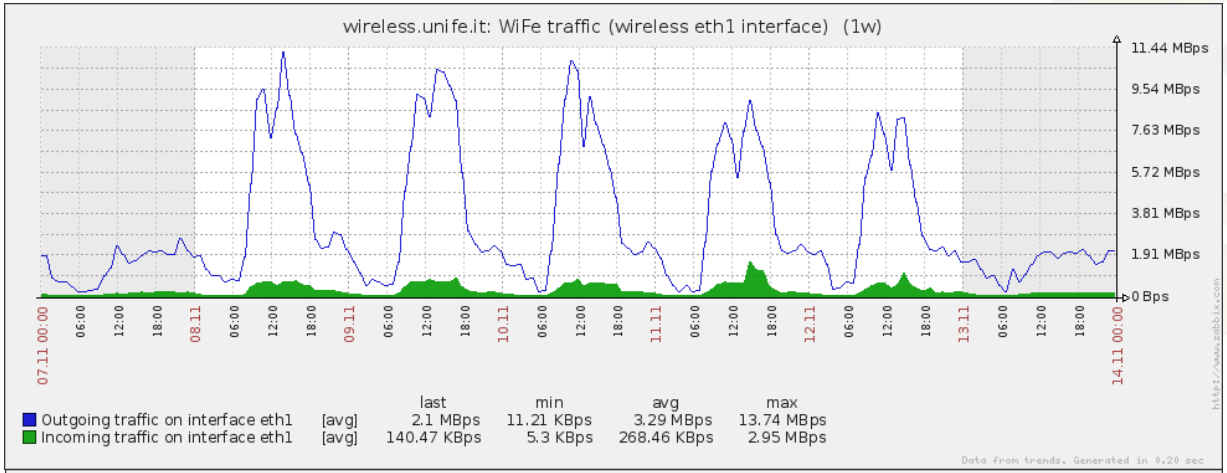
Prestazioni

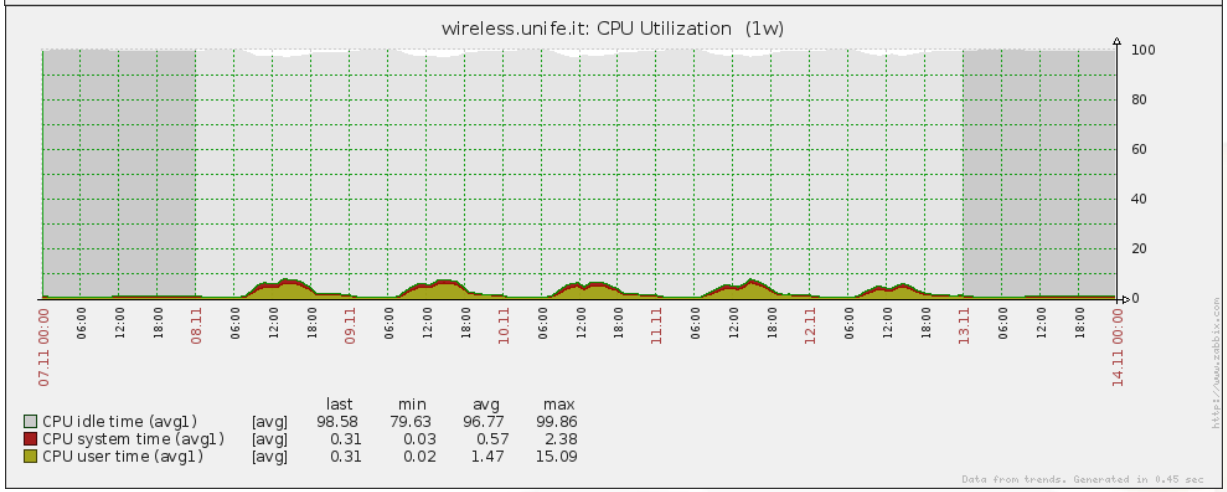
Hardware:

- CPU: 2 Xeon 2Ghz a 4 core
- RAM: 12Gb

Prestazioni:

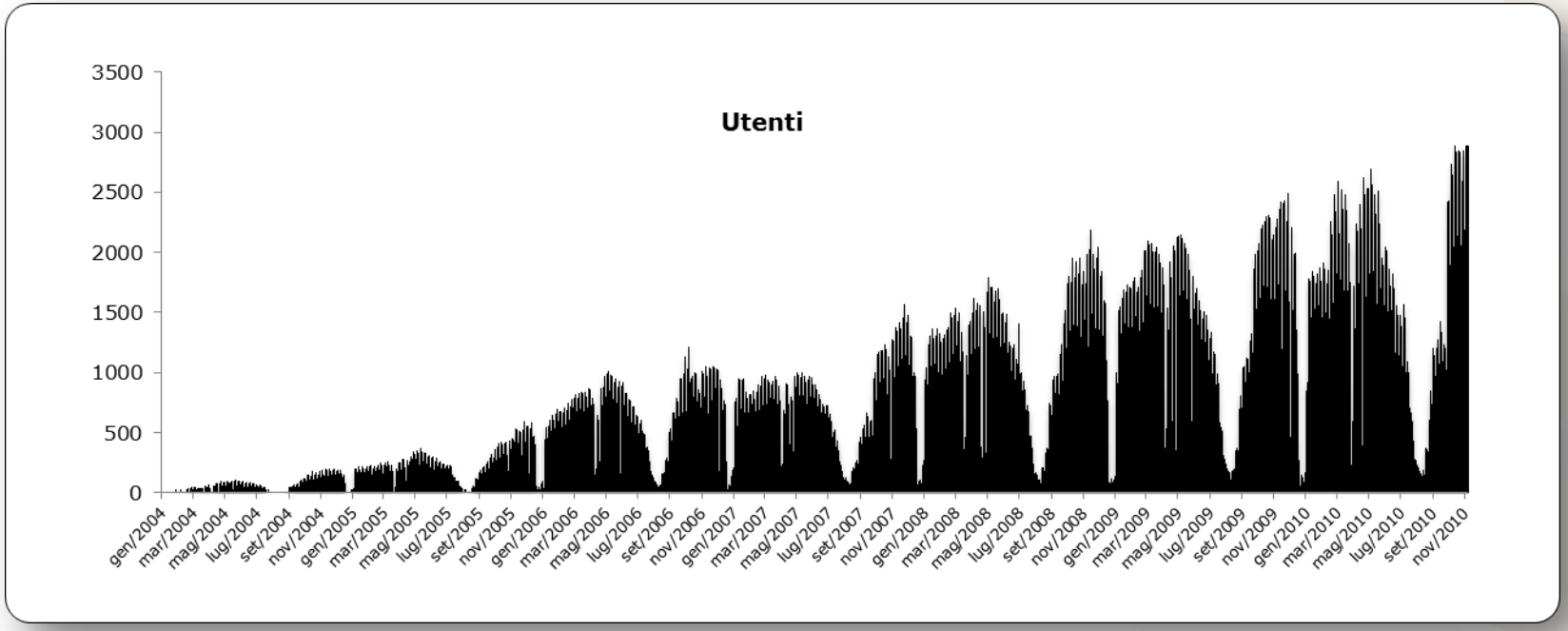
- 100 Mbit/s (12MByte/s) di download complessivo





Prestazioni

- Quasi 3000 differenti utenti accedono quotidianamente al servizio



Il captive portal

Captive portal: panoramica

- Il panorama è vasto: esistono molteplici soluzioni e tecnologie (opensource o commerciali, embedded o server)

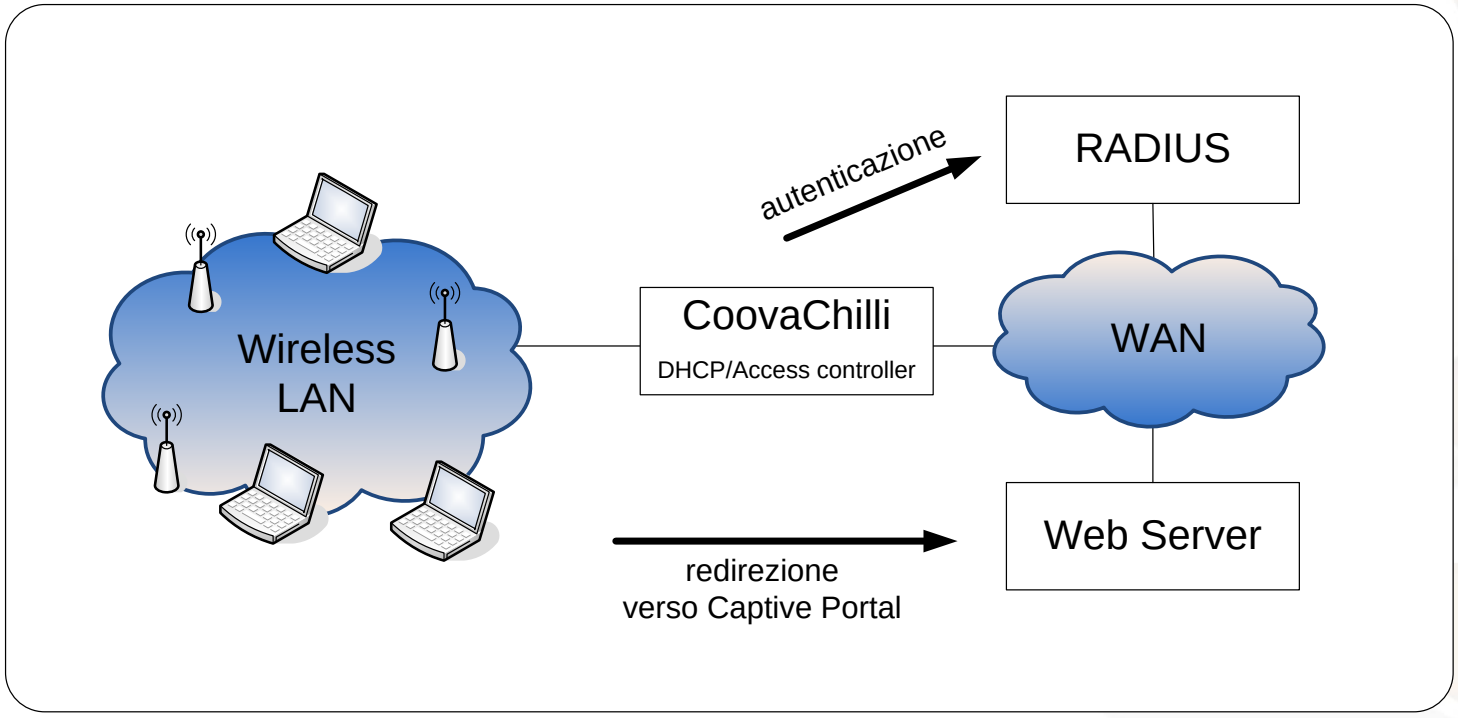
- Si è scelto valutando:
  - Flessibilità (sistema server su ambiente Linux)
  - Costi (opensource)
  - Scalabilità e prestazioni

Captive portal: coovachilli

- Utilizzato dalla comunità "Fon"
- Disponibile per sistemi embedded (CoovaAP)
- Autenticazione via RADIUS
- Gestione integrata di redirezione controllo del traffico
- Integra il DHCP server, utilizzato come keepalive di sessione
- **Possibilità di gestione con comandi/script esterni**

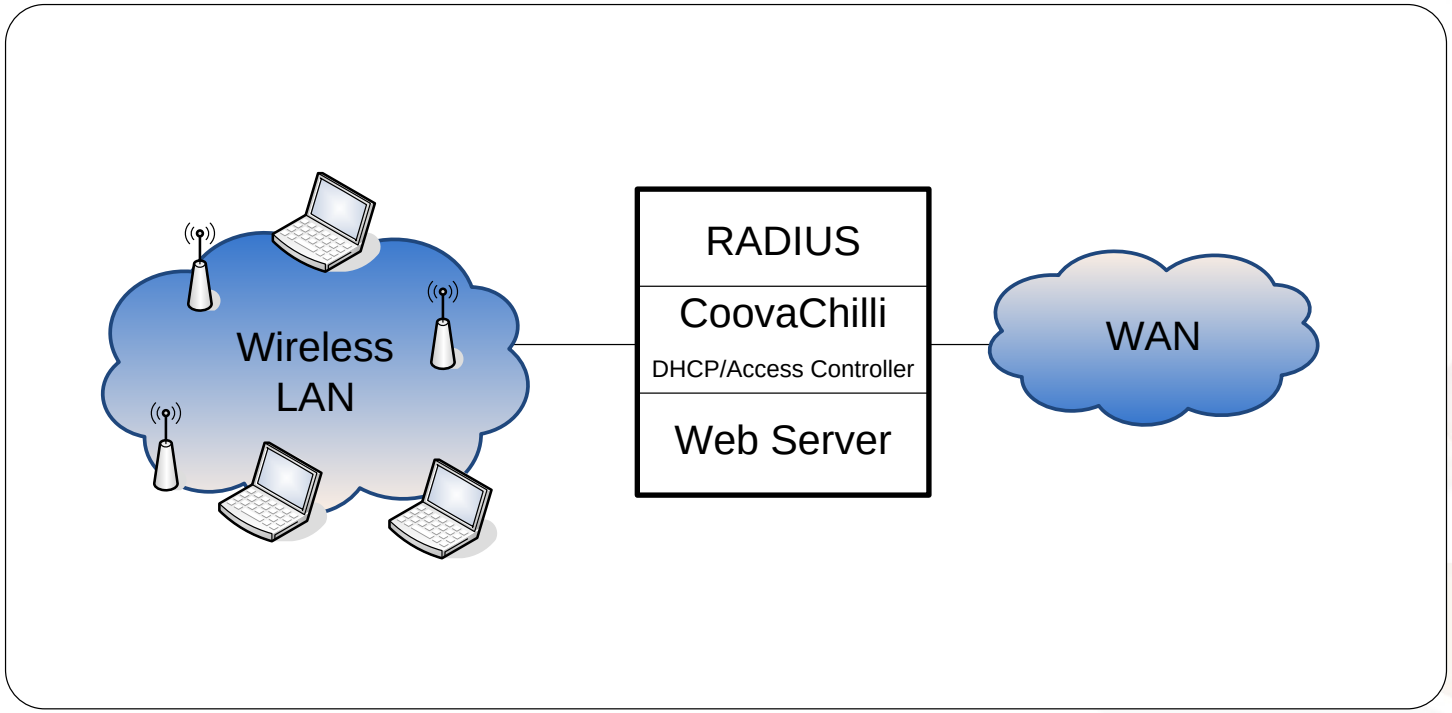
Coovachilli

- Schema di funzionamento

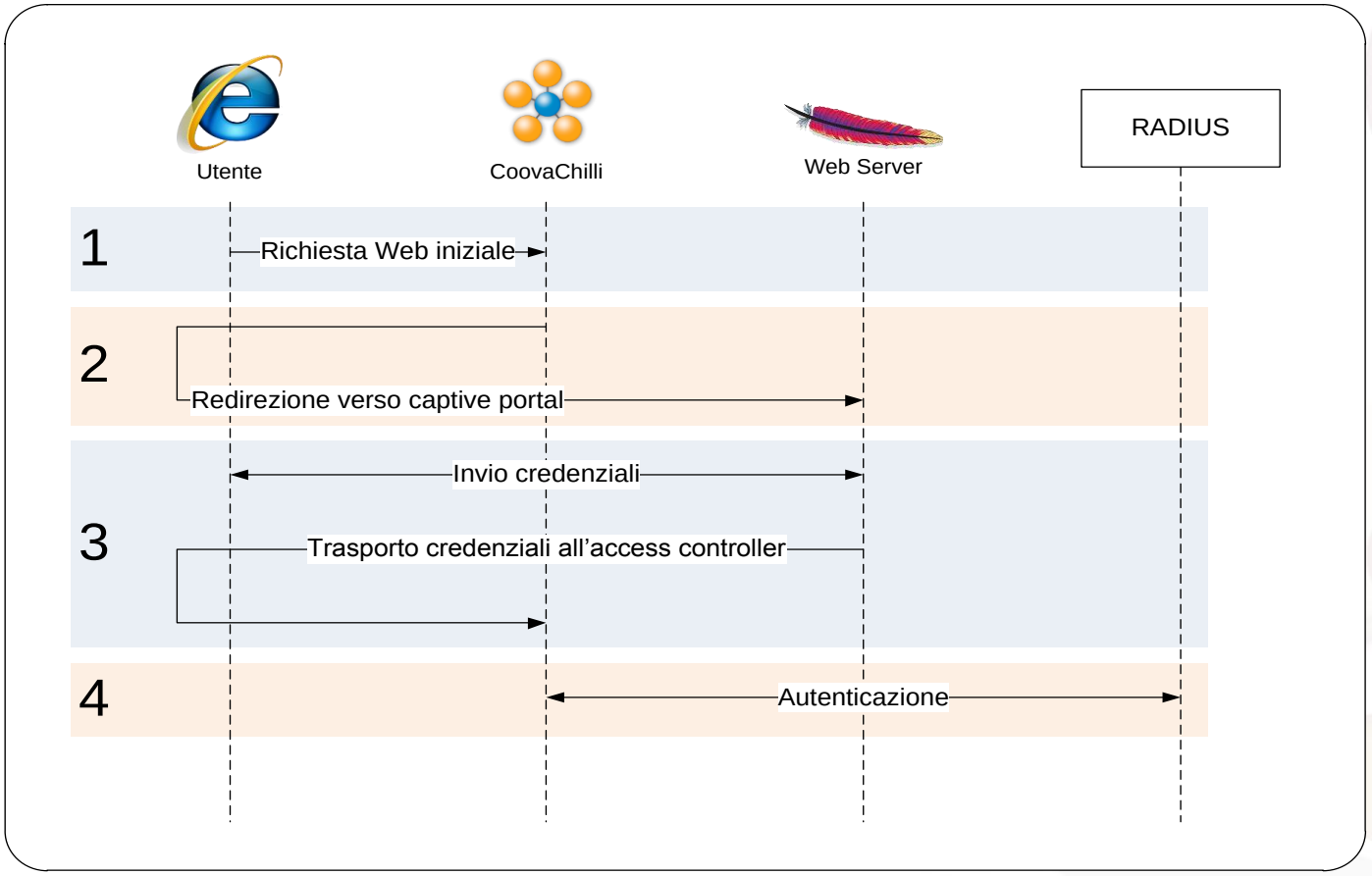


Coovachilli

- Schema di funzionamento in UniFe



Coovachilli: autenticazione



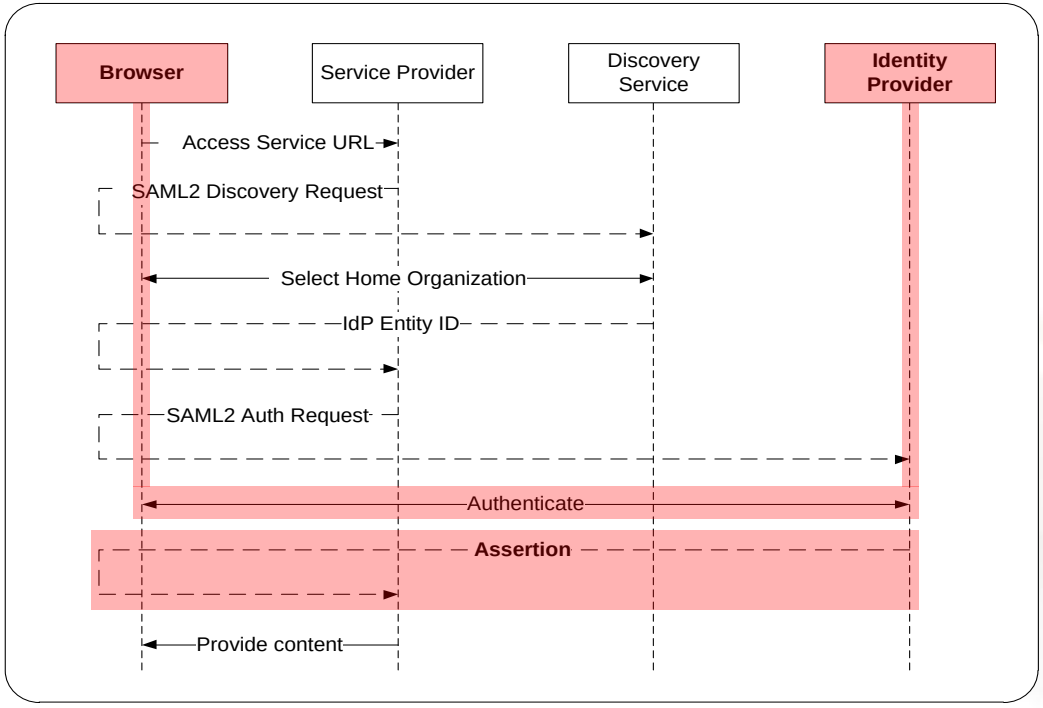
L'integrazione di Wi-Fe con IDEM

Integrazione Wi-Fe/IDEM

- Obiettivo: si vuole mantenere il sistema di Captive Portal in uso, per motivi di affidabilità e costi.

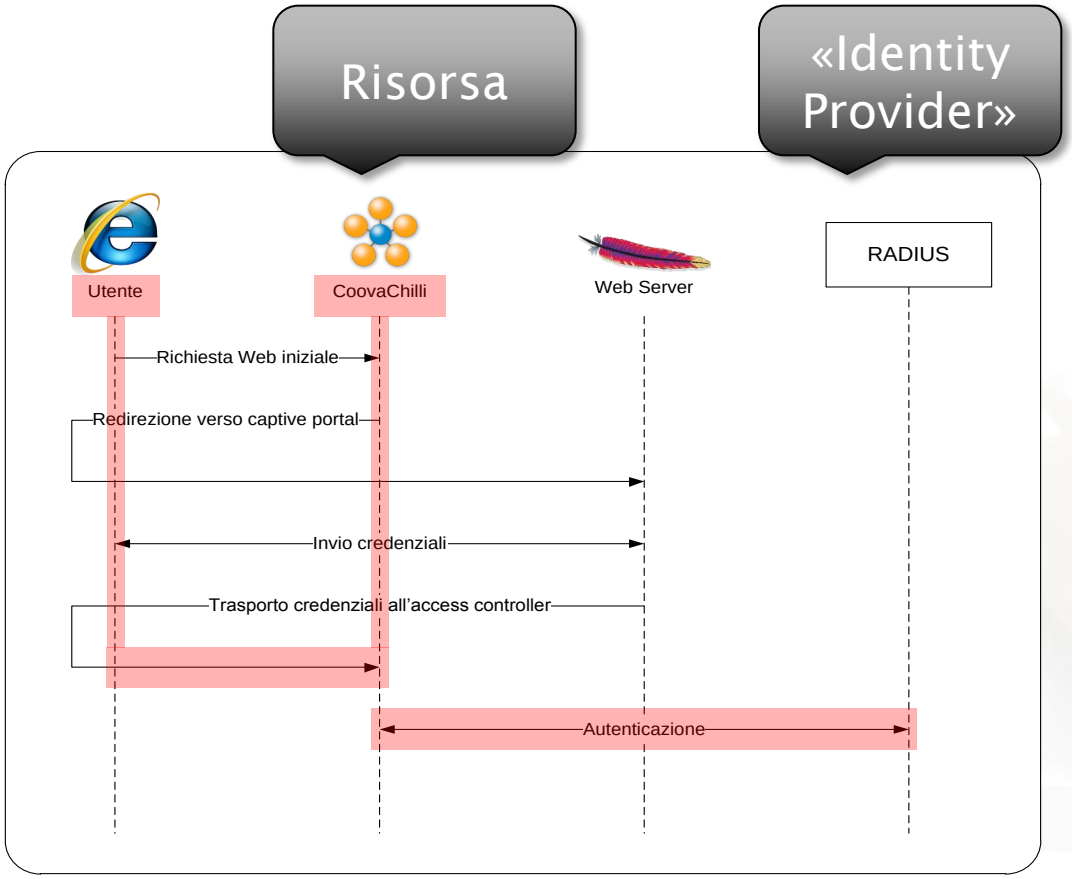
- Problemi: il sistema fa uso del protocollo RADIUS e l'architettura non è compatibile con Shibboleth

Autenticazione Shibboleth



- L'utente invia le credenziali all'Identity Provider, non alla risorsa a cui si vuole accedere.
- L'IdP comunica l'esito dell'autenticazione tramite una «authentication assertion».

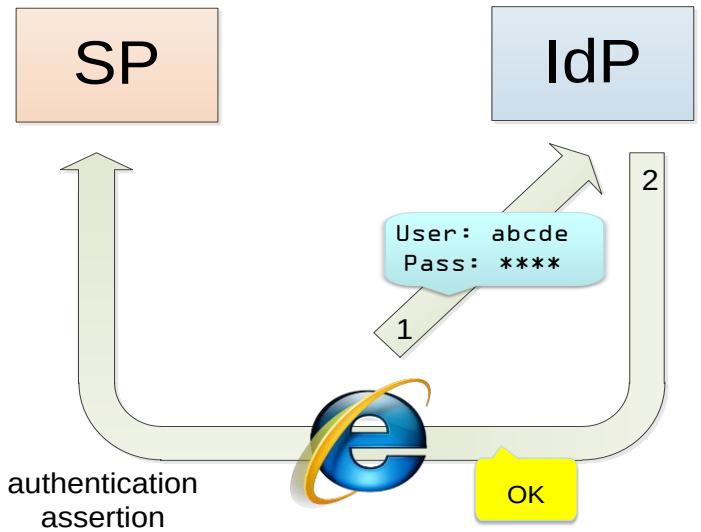
Autenticazione Coovachilli



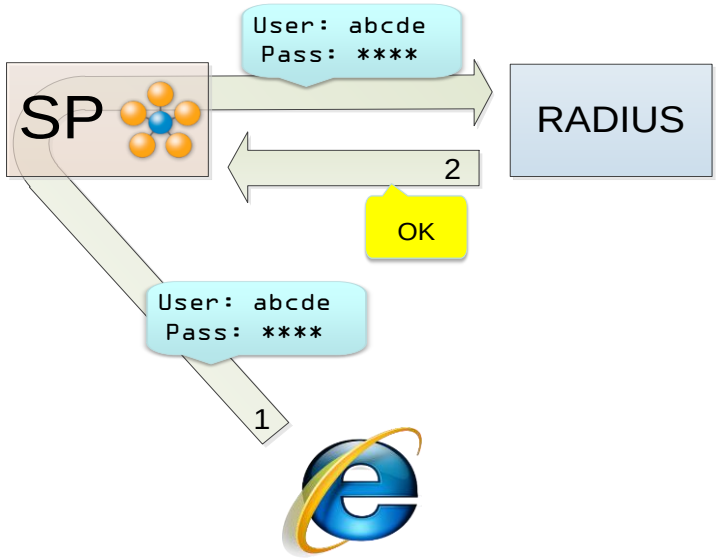
- L'utente invia le credenziali alla risorsa stessa (l'access controller) che poi effettua l'autenticazione
- Non esiste la fase di «authentication assertion»

Confronto Shibboleth/Chilli

Autenticazione in Shibboleth



Autenticazione negli access controller

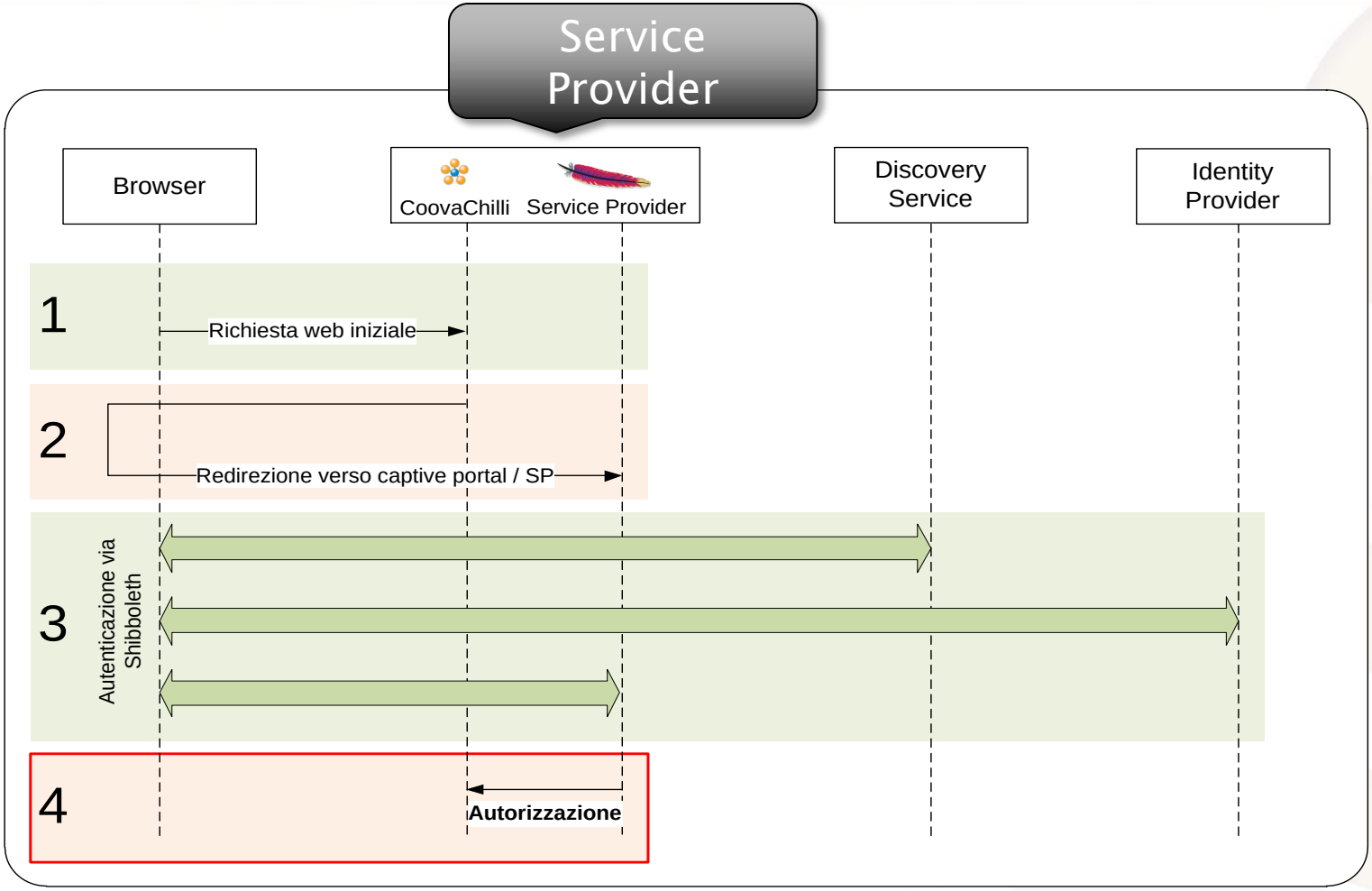


Integrazione

Soluzione di tipo «sistemistico»:

- Nella macchina che ospita il captive portal viene installato un SP Shibboleth (Apache + mod shibboleth)
- L'accesso alla risorsa web protetta invoca una richiesta di autorizzazione all'access controller
- Service Provider e access controller devono risiedere sullo stesso server

Flusso


Service Provider
Browser
CoovaChilli
Service Provider
Discovery Service
Identity Provider
1
Richiesta web iniziale
2
Redirezione verso captive portal / SP
3
Autenticazione via Shibboleth
4
Autorizzazione

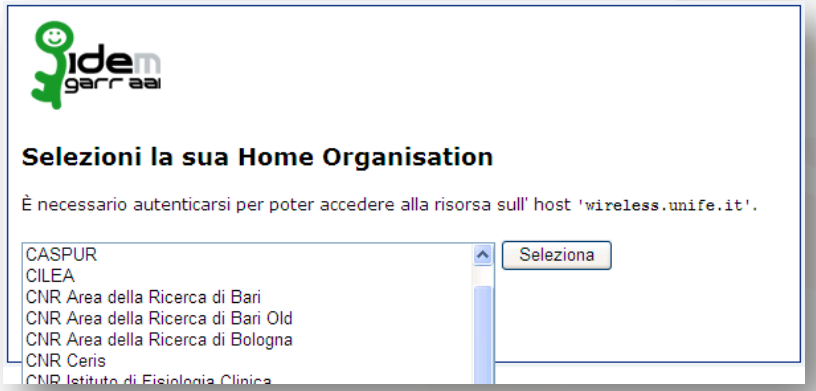
Flusso: passo per passo

1. La navigazione web viene intercettata dal captive portal come di consueto

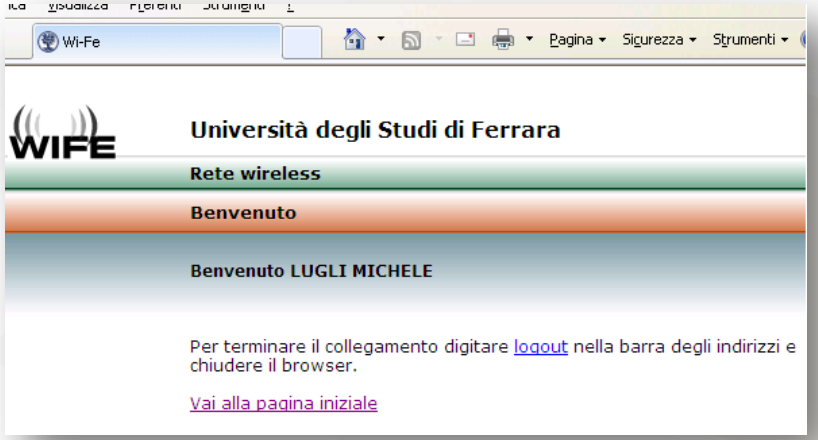
2. Dalla pagina di accesso si seleziona il «Login via IDEM»: è un collegamento alla pagina protetta con Shibboleth

Flusso: passo per passo

3. Si effettua l'autenticazione via IDEM



4. L'accesso alla pagina protetta autorizza l'IP dell'utente, sbloccando il captive portal



Implementazione

Implementazione

- Installato service provider (apache - libmodshib) nella macchina che ospita il controllore degli accessi
- Una pagina, protetta con shibboleth, effettua l'autorizzazione su coovachilli
- Predisposto il walled garden
- Configurati privilegi

Autorizzazione

1. prelievo parametri di sessione

```php
<?php
        # parametri
        $ip   = $_SERVER['REMOTE_ADDR'];
        $user = $_SERVER['eduPersonPrincipalName']

        # autorizzazione su chilli
        $homepage=shell_exec("sudo /usr/sbin/chilli_query authorize ip $ip username $user");

        # scrivo nel file di log di coovachilli
        openlog("Shibboleth-SP", LOG_CONS | LOG_NDELAY, LOG_LOCAL6);
        syslog(LOG_NOTICE, "login from username=".$user." IP=".$ip);
        closelog();
?>
<html>
        <head>
                <title>Wi-Fe Login</title>
    ..................
```

2. autorizzazione

3. logging

Walled garden

```
1  uamallowed "idp.dir.garr.it"
2  uamallowed "shibidp.unipr.it"
3  uamallowed "idp.unimore.it"
4  uamallowed "shibidp.polimi.it"
5  uamallowed "ip1.rettorato.unito.it"
6  uamallowed "fire.rettorato.unito.it"
7  uamallowed "shidp.caspur.it"
8  uamallowed "idp2.cilea.it"
9  uamallowed "idp.uniroma3.it"
```

- E' necessario permettere la navigazione verso gli IdP senza autenticazione (walled garden)
- La configurazione del walled garden viene periodicamente aggiornata sulla base dei metadati della Federazione

Privilegi

- Il server web deve poter autorizzare una sessione di coovachilli

```
1  # /etc/sudoers
2  #
3  # This file MUST be edited with the 'visudo' command as root.
4  #
5  # See the man page for details on how to write a sudoers file.
6  #
7
8  Defaults        env_reset
9
10 # User privilege specification
11 root    ALL=(ALL) ALL
12
13 # Members of the admin group may gain root privileges
14 %admin ALL=(ALL) ALL
15
16 www-data wireless.unife.it = NOPASSWD: /usr/sbin/chilli_query list,\
17                                         /usr/sbin/chilli_query authorize
18
```

Considerazioni finali

- Logout?
- Collaborazione con Lepida SpA al sistema "FedERa"
- Collaborazione con il Comune di Ferrara
- Sviluppo di CoovaChilli