Consiglio Nazionale delle Ricerche

WiFi@PDCnr
Manuale d'uso
e
Guida all'implentazione

P.Bison, C.Cavaggion
ISIB-CNR

Rapporto Tecnico 01/11, ISIB-CNR
Padova, Ottobre 2011

ISIB-CNR
Corso Stati Uniti 4
35100 Padova IT

SOMMARIO

WiFi@PDCnr è un servizio gestito dall'Area della Ricerca di Padova per fornire l'accesso alla rete internet attraverso un collegamento wireless ed è offerto sia a dipendenti che a ospiti delle istituzioni afferenti all'area della ricerca stessa. L'accesso è consentito solo ad utenti in possesso di credenziali rilasciate dall'Area della Ricerca di Padova e dal CNR di Roma o ad utenti appartenenti ad istituzioni afferenti alla federazione GARR IDEM.

ABSTRACT

WiFi@PDCnr is a service managed by the CNR Research Area in Padua (ADRPD) to allow personel and guests of an istitution belonging to the ADRPD to connect to the internet through a wireless access. In ordert to connect to the network a user must have login credentials issued by ADRPD or by the CNR headquarter in Rome or be an employee of an istitution member of the GARR IDEM federation.

INDICE

Introduzione

Presso l'Area della Ricerca di Padova è attiva una rete wireless che nelle zone di copertura, pubblica tre reti SSID:

- *adr_wifi*
- *adr_voip*
- *adr_guest*

La rete *adr_wifi* è riservata al personale permanente afferente alle istituzioni che risiedono presso l'area della ricerca in possesso di credenziali di accesso rilasciate dall'area stessa. La rete *adr_voip* è dedicata a dispositivi voip, quali telefoni wifi, e permette solamente l'uso dei protocolli relativi ai sistemi voip. La rete *adr_guest* è a disposizione sia del personale che di ospiti temporaneamente presenti presso l'Area della Ricerca di Padova, offre un insieme limitato di protocolli di rete e il suo accesso è vincolato al possesso di credenziali di accesso.

Per una completa descrizione delle modalità e sui vincoli inerenti l'uso del servizio WiFi@PDCnr si veda il sito http://wifi.pd.cnr.it/.

Questo rapporto nella sezione "Manuale d'uso" descrive le procedure che utente deve eseguire per collegarsi alla rete *adr_guest*, mentre la sezione "Guida all'implementazione" illustra i dettagli implementativi riguardanti tali procedure.

La struttura della rete wireless e le modalità di accesso alle reti *adr_wifi* e *adr_voip* sono descritte nei rapporti tecnici elencati in bibligrafia.

Manuale d'uso

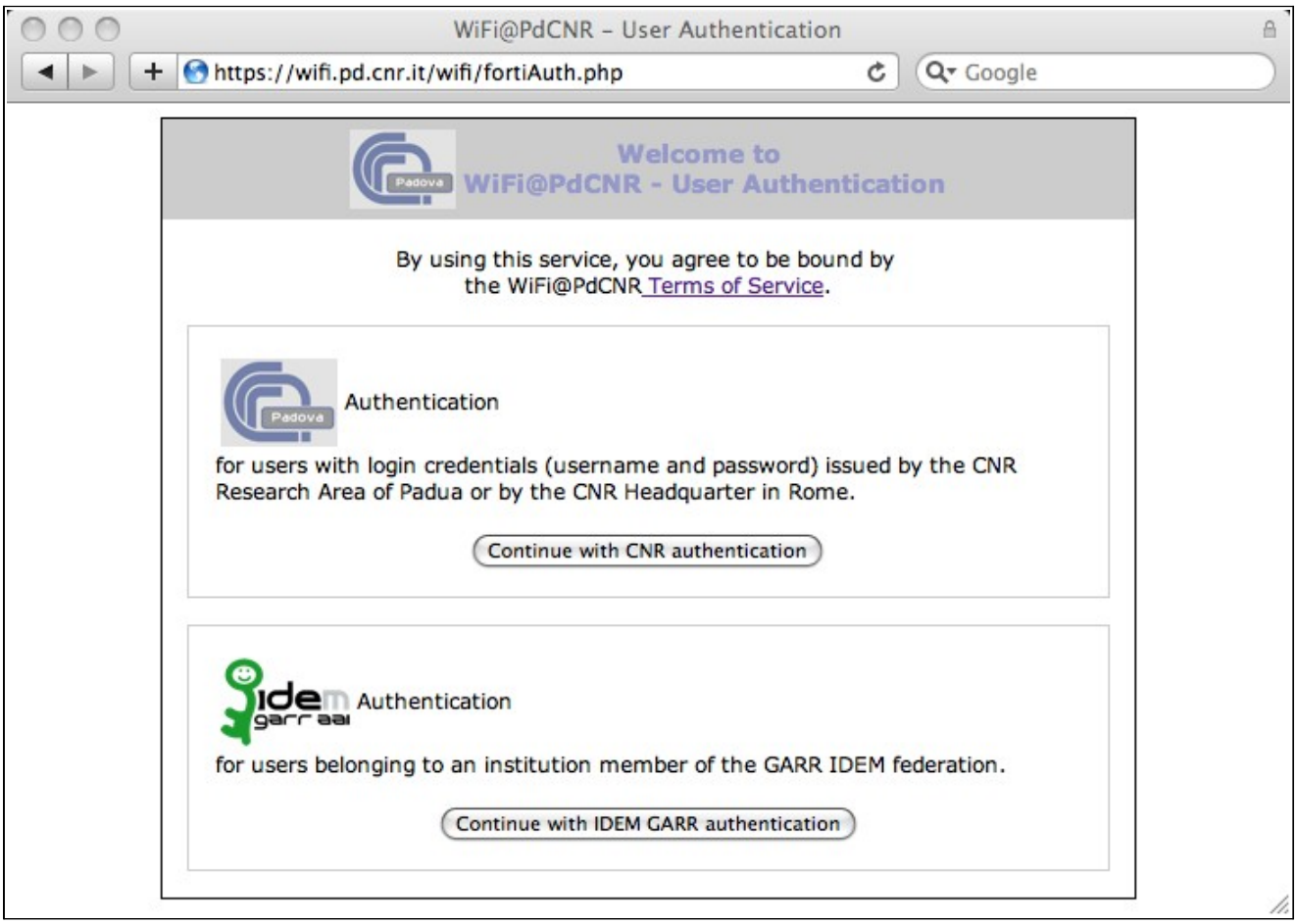
In questa sezione si descrive la procedura che un utente deve seguire per potersi collegare alla rete internet attraverso la rete wireless a*dr_guest*.

Collegamento iniziale

L'utente deve scegliere la rete wireless *adr_guest*, attivare un browser WWW con la funzioniltà javascript abilitata e accedere ad un sito qualsiasi.

Se il browser WWW non ha javascript abilitato appare la seguente pagina che invita l'utente ad abilitare tale funzionalità e riattivare il brower dato che senza javascript non è possibile ottenere l'accesso a internet:

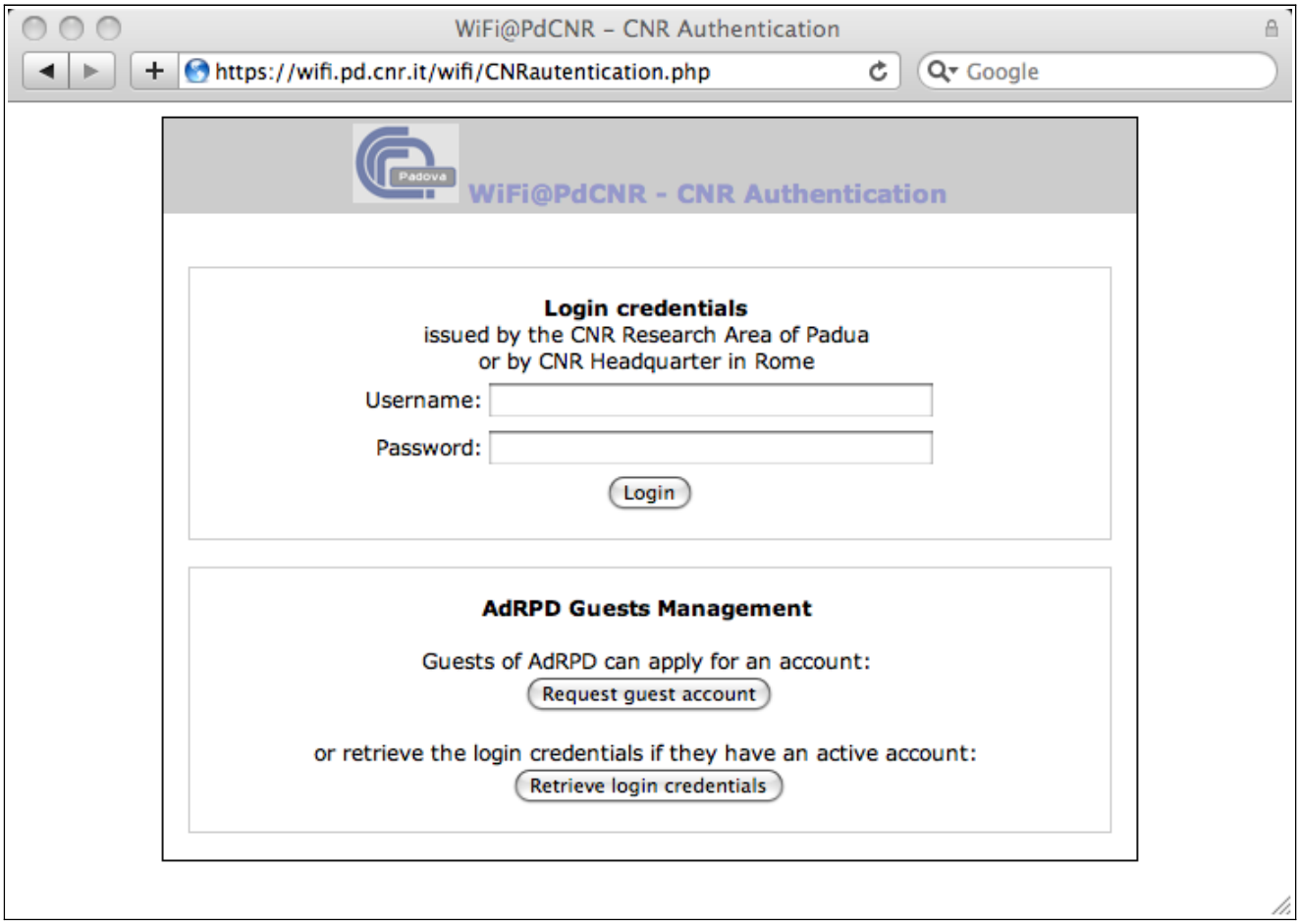
Con javascript attivo l'utente viene ridirezionato su un server web che gli presenta la seguente pagina contenente due possibili scelte per la modalità di autenticazione:



Se l'utente ha credenziali fornite dall'Area della Ricerca di Padova oppure dal CNR di Roma sceglie la prima possibilità, mentre se appartiene ad una istituzione afferente alla federazione GARR IDEM la seconda.

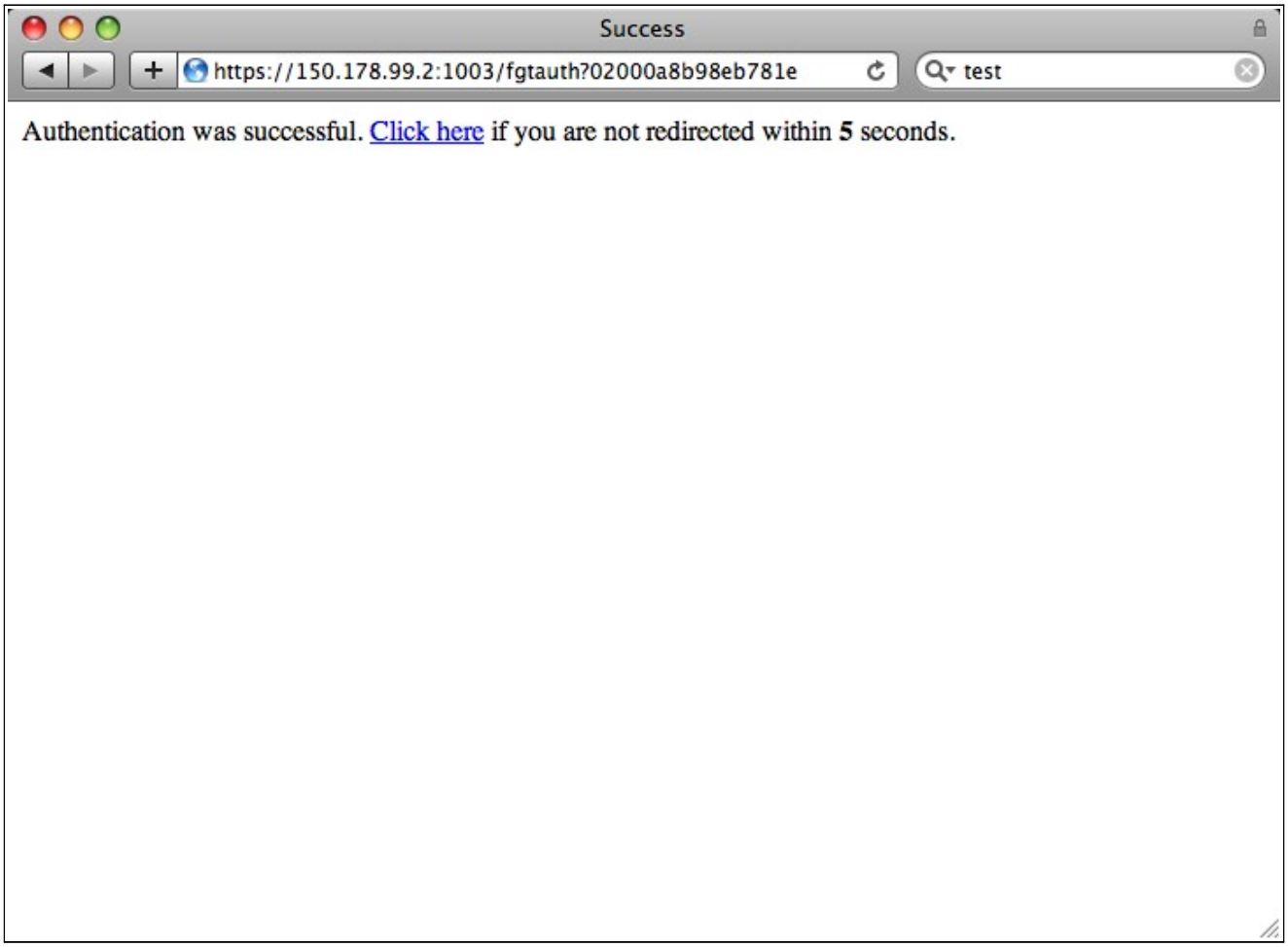
Autenticazione CNR

Una volta scelta questa opzione viene presentata la seguente pagina

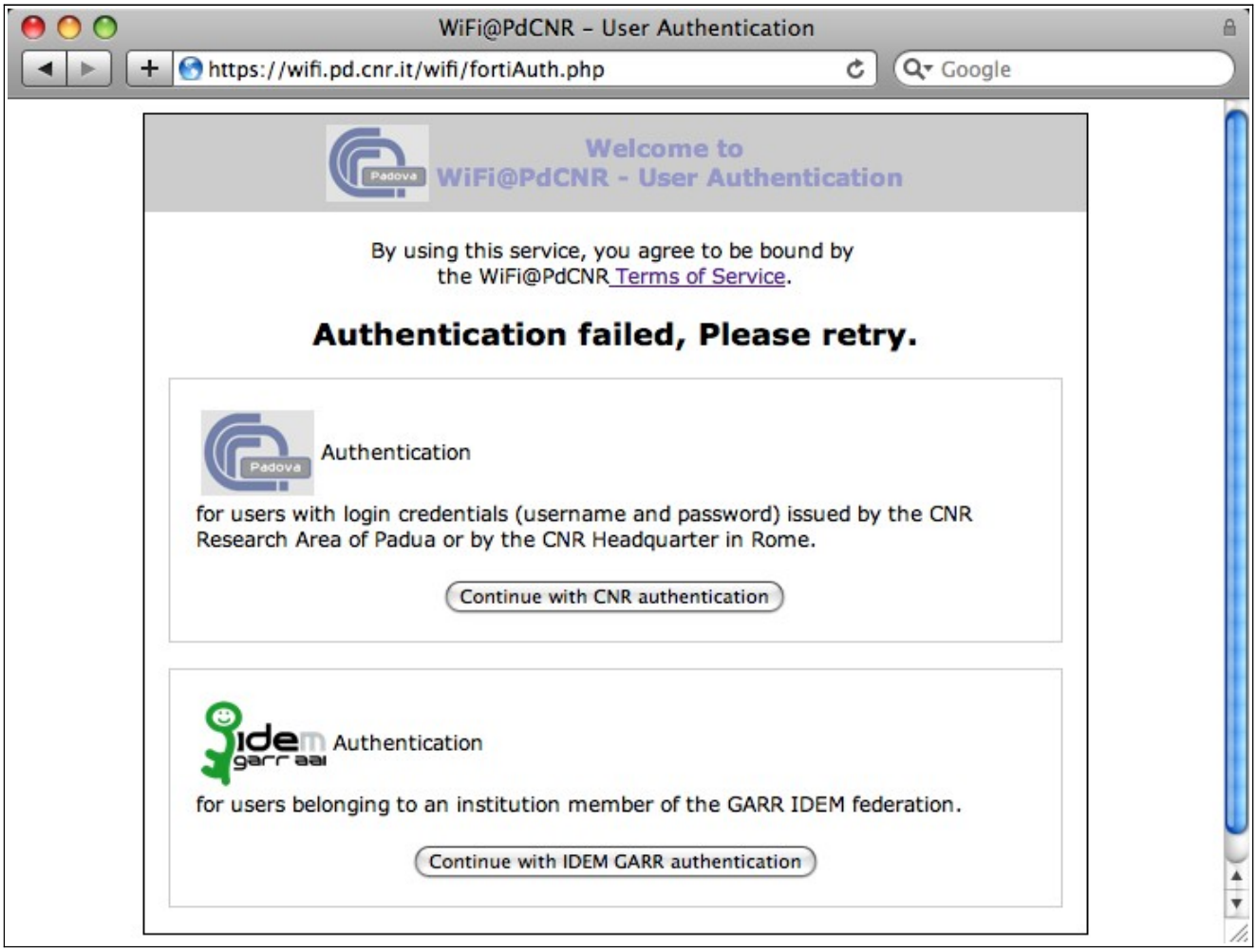


Se l'utente è già in possesso di credenziali, username e password, fornite dall'Area della Ricerca di Padova oppure dal CNR di Roma può digitarle nei due campi Username: e Password: e inviarle con il pulsante Login.

Se le credenziali risulteranno corrette il sistema abiliterà l'accesso alla rete e l'utente potrà continuare la navigazione web e/o utilizzare gli altri protocolli abilitati:



altrimenti verrà riproposta la pagina di autenticazione CNR con un messaggio di errore per una eventuale riprova.



Se il sistema dovesse continuare a rifiutare l'accesso anche in presenza di credenziali ritenute corrette, rivolgersi al proprio referente d'istituto.

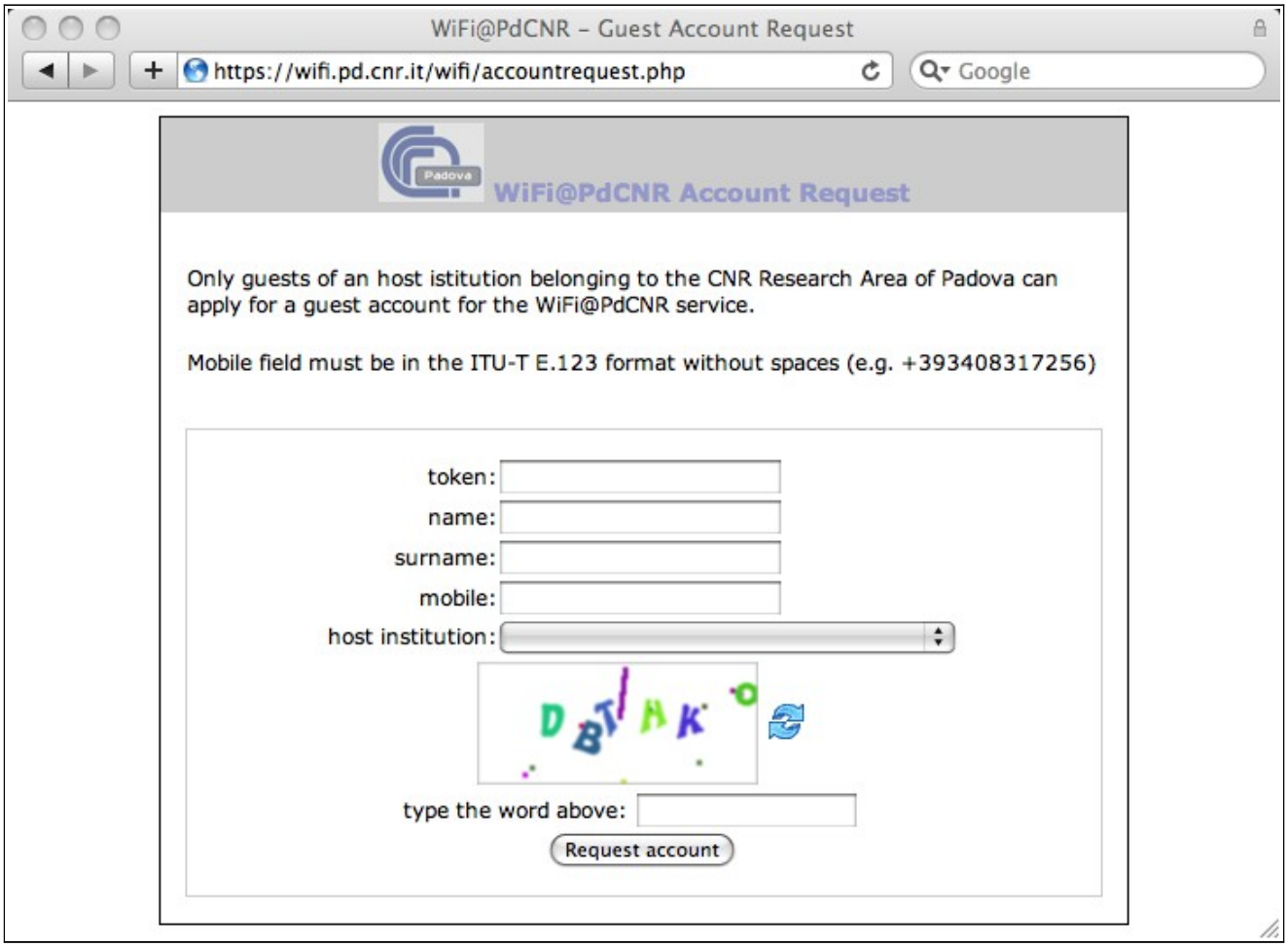
Credenziali temporanee

Utenti che sono ospiti temporanei di istituzioni afferenti all'Area della Ricerca di Padova possono richiedere via web credenziali temporanee che scadono la mezzanotte del sabato successivo alla creazione. **Le credenziali assegnate sono valide fino alla mezzanotte del sabato successivo alla loro creazione e non è possibile modificare né lo username né la password.**

Creazione credenziali temporanee

Per far questo gli utenti temporanei devono richiedere al referente di istituto di cui sono ospiti un codice (token) da utilizzarsi nella procedura di richiesta che viene attivata con il pulsante "Request guest account" presente nella pagina di autenticazione CNR.

La prima pagina presentata da questa procedura è un modulo web utilizzato per la raccolta dei dati personali dell'utente necessari per la creazione delle credenziali di accesso:

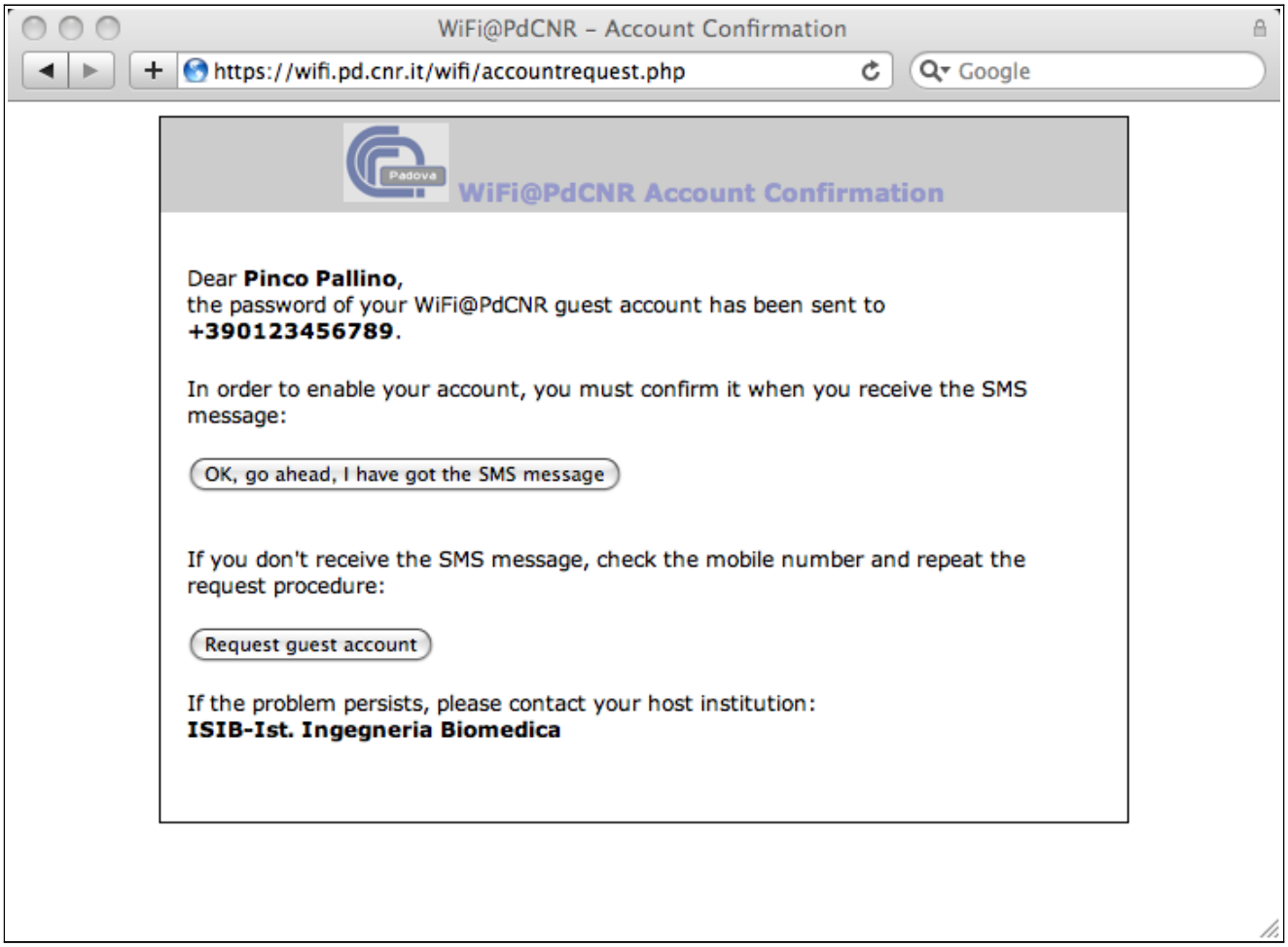


dove:

token	è il codice ottenuto dal referente di istituto
name	è il nome dell'utente
surname	è il cognome dell'utente
mobile	è il numero di telefono cellulare utilizzato per individuare in maniera univoca la persona fisica titolare delle credenziali di accesso, inoltre a tale numero verrà inviata la password per l'accesso al servizio
host istitution	è un menù che permettere di scegliere l'istituzione che ospita l'utente

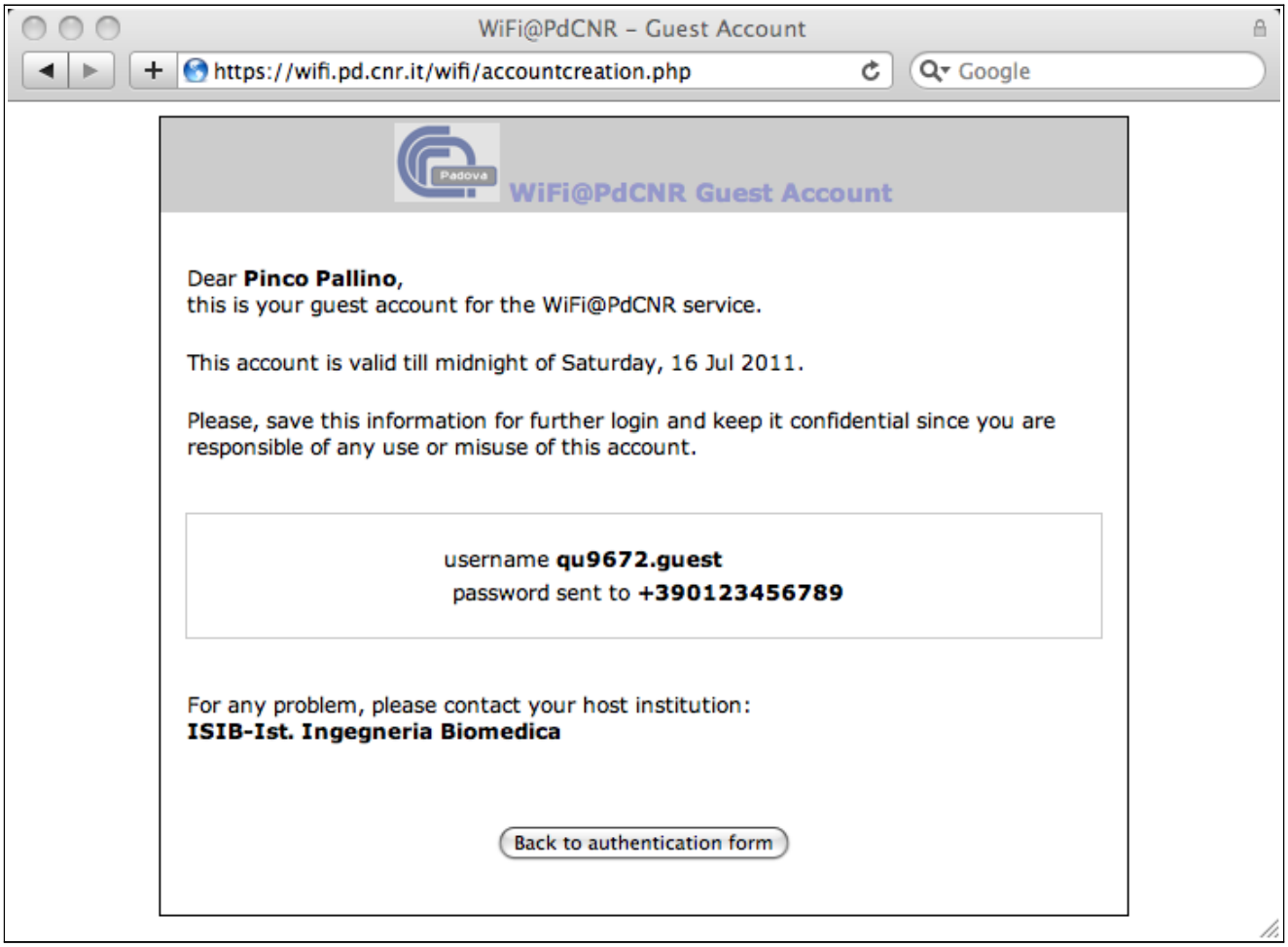
Una volta che l'utente ha compilato tutti i suddetti campi e inserito il codice di controllo può inviare la richiesta con il pulsante “Request account”.

Se i dati risulteranno incompleti o errati, il sistema riproporrà il modulo segnalando gli errori commessi, altrimenti invierà via SMS la password e visualizzerà la seguente pagina:



Se l'utente non riceve un SMS con la password in un tempo ragionevole (qualche minuto), può ripetere la procedura di richiesta con il pulsante "Request guest account" dopo aver controllato se il numero di telefono inserito è corretto. Se persiste l'impossibilità di creare un account temporaneo, contattare il referente d'istituto.

Una volta che l'utente ha ricevuto il messaggio SMS contente la password deve confermarlo attivando il pulsante "OK, go ahead, ..." che fornirà i dati completi (eccetto la password) relativi alle credenziali appena create:

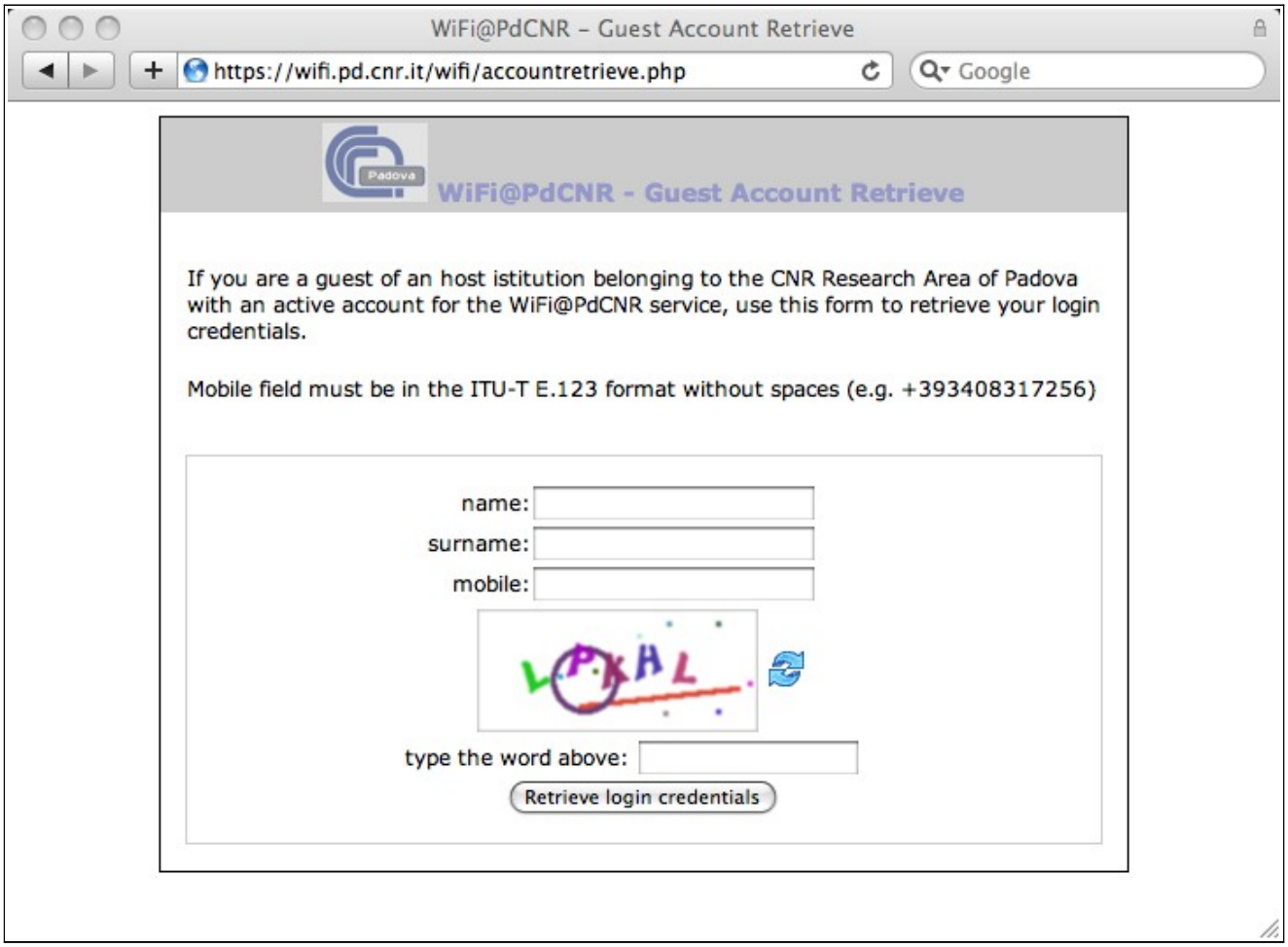


Si suggerisce di salvare la pagina contente le informazioni relative alle proprie credenziali, ad esempio stampandola come file, per poterle riutilizzare in seguito.

Le credenziali temporanee sono valide fino alla mezzanotte del sabato successivo alla loro creazione e non è possibile modificare né lo username né la password.

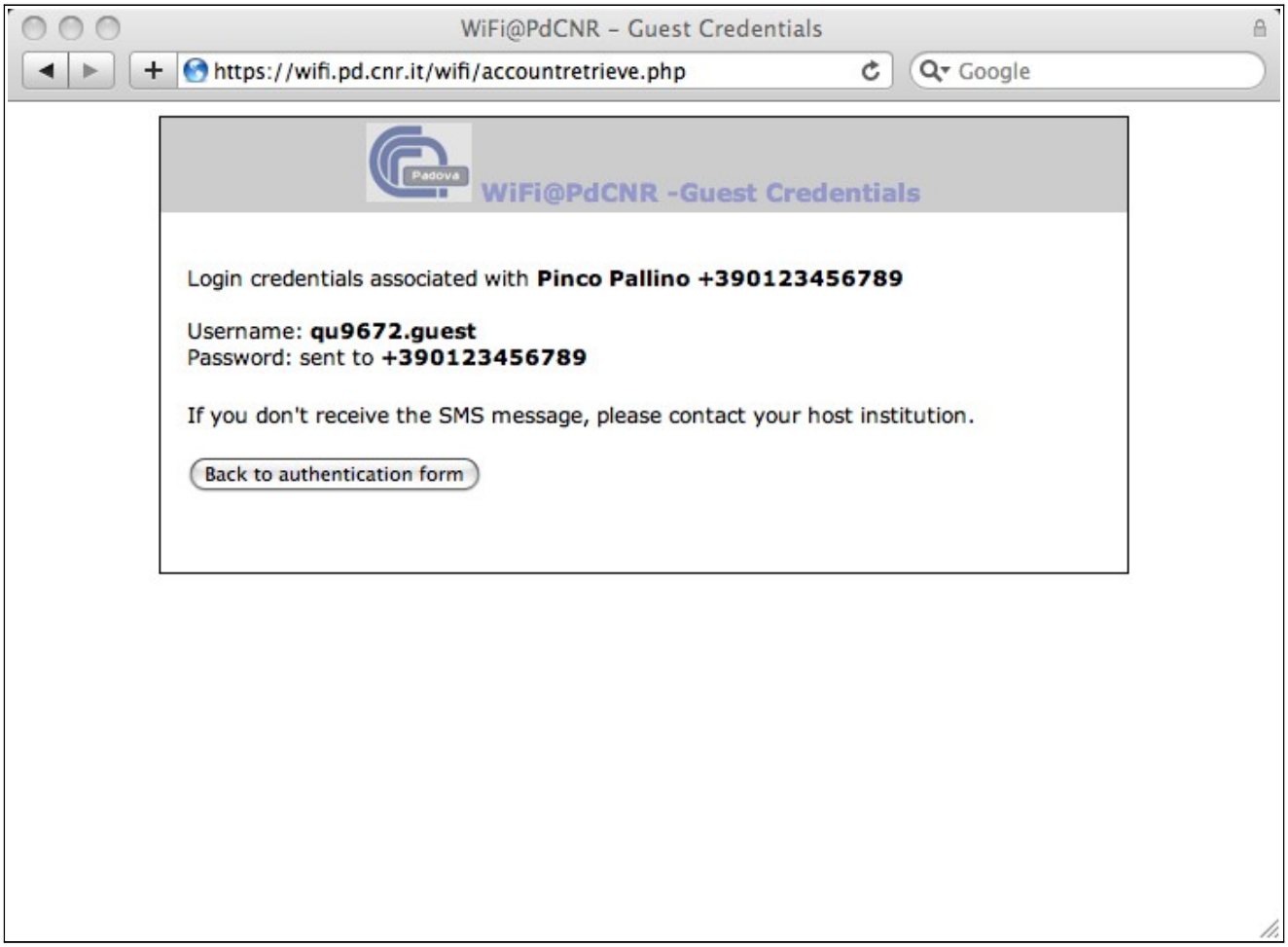
Recupero credenziali temporanee

Se un utente ospite ha perso i dati relativi alle sue credenziali temporanee può ricuperarle utilizzando il bottone ”Retrieve login credentials” presente nella pagina relativa all'autenticazione CNR che lo invia alla seguente pagine web:



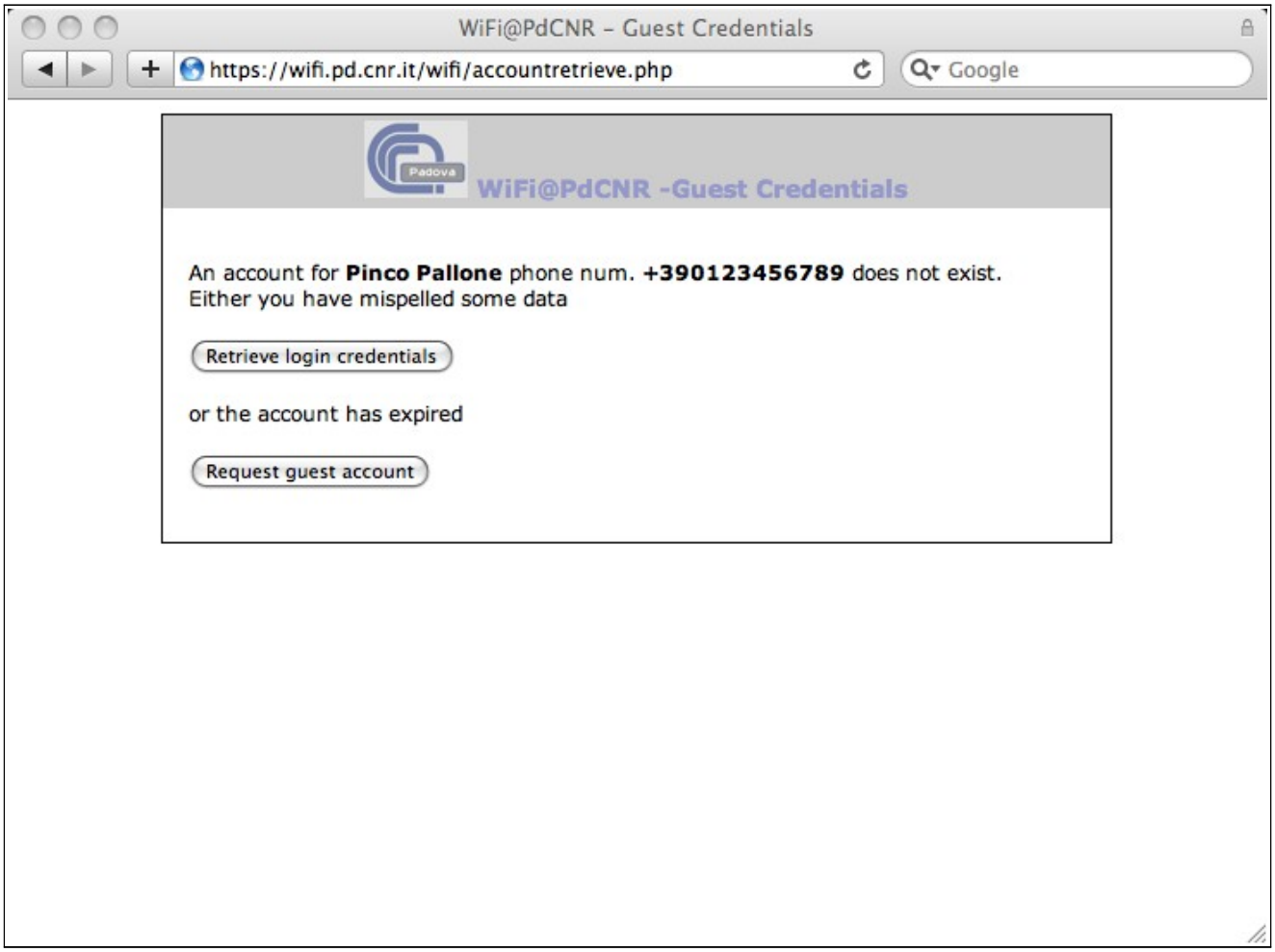
Un volta inseriti i dati relativi all'account (nome,cognome e numero di cellulare) e il codice di controllo l'utente può richiedere le relativi credenziali di accesso utilizzando il pulsane “Retrieve login credentials”.

Se l'account esiste e non è scaduto il sistema invia la password al numero di cellulare e visualizza la seguente pagina contenente la username:



Si suggerisce di salvare la pagina contente le informazioni relative alle proprie credenziali, ad esempio stampandola come file, per poterle riutilizzare in seguito.

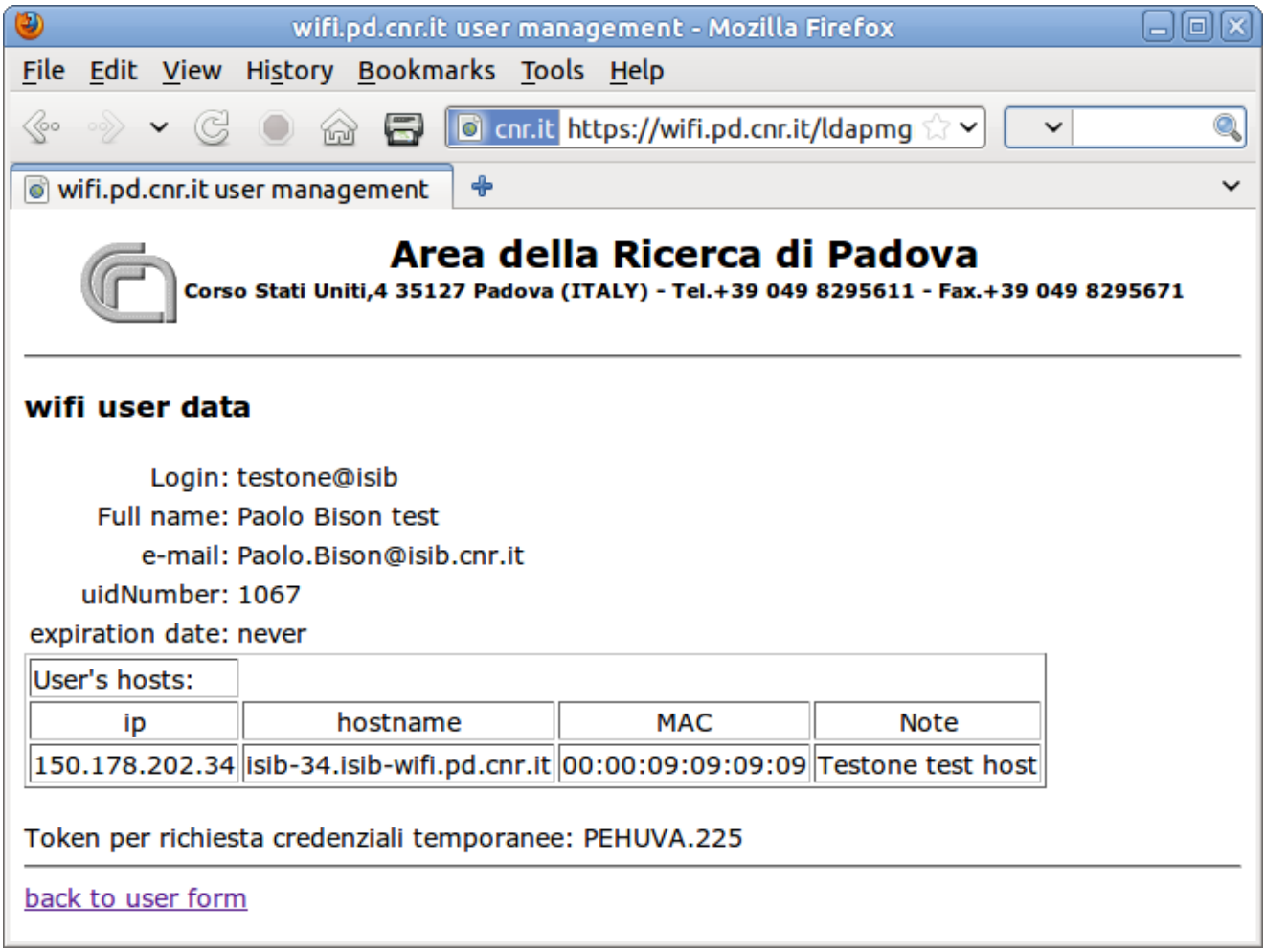
Se invece l'account non esiste o è scaduto l'utente riceve il seguente messaggio:



che lo invita a riprovare se ritiene di aver sbagliato ad introdurre i dati, oppure a richiedere delle nuove credenziali temporanee.

Richiesta token

Utenti con credenziali permanenti rilasciate dall'Area della Ricerca di Padova per l'accesso alla rete wireless possono ottenere un token con validità giornaliera attraverso l'interfaccia web per la visualizzazione dei dati associati alle proprie credenziali. Collegandosi con un browser al URL https://wifi.pd.cnr.it/ldapmgm/usermgmt.php e utilizzando il pulsante "Show user data" dopo aver digitato le proprie credenziali, si ottiene una pagina che visualizza un token attivo per il giorno corrente, come visualizzato nell'immagine seguente che in questo caso indica come token la stringa **PEHUVA.225**:

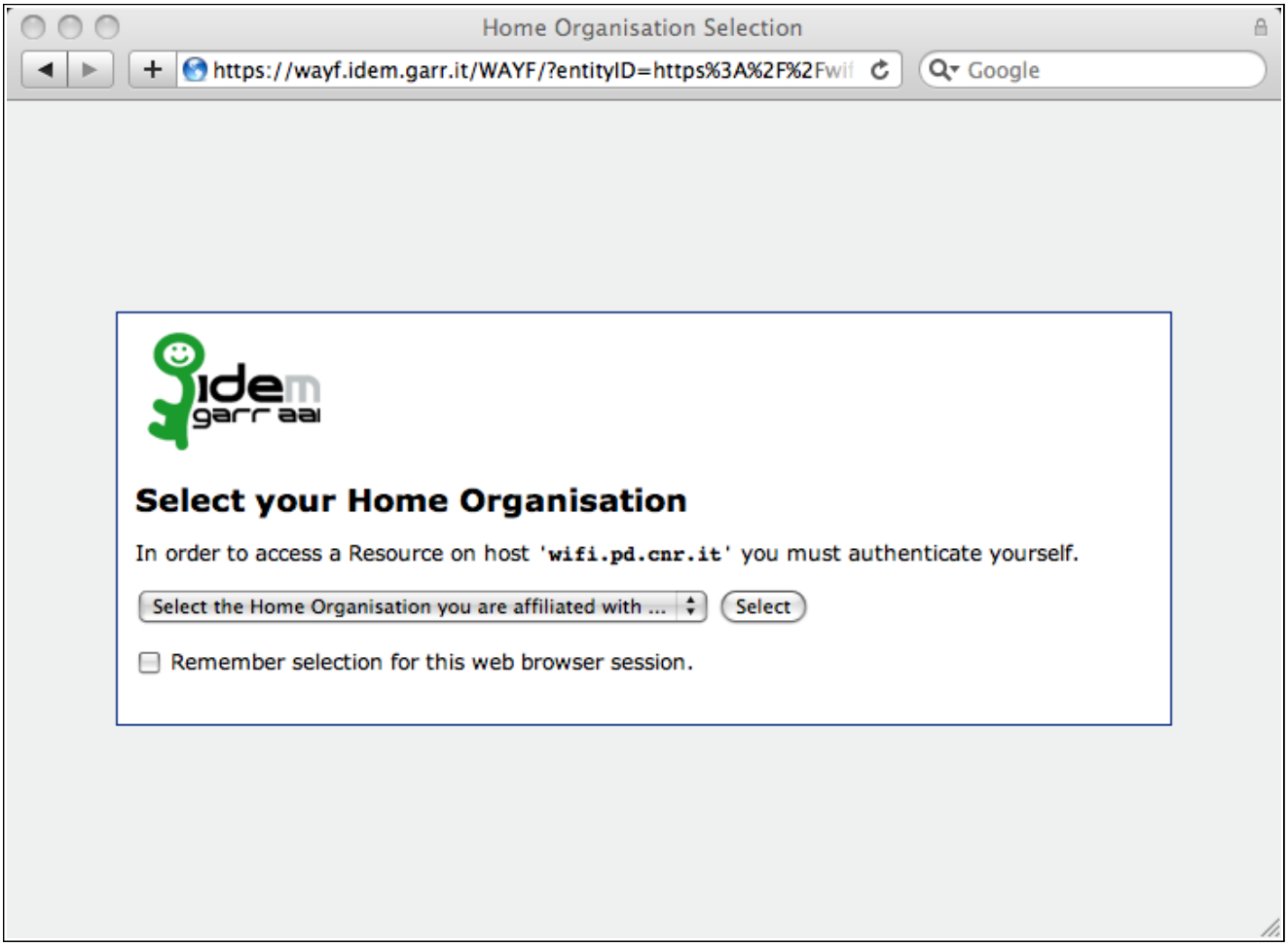


Se vi fosse la necessità di avere token validi per più giorni, ad esempio in occasione di congressi o riunioni, si può inviare una richiesta via e-mail a supporto@pd.cnr.it specificando il motivo e il periodo di validità del token richiesto.

Autenticazione IDEM

Un ospite presente nell'Area del Ricerca di Padova che abbia un account presso una qualunque istituzione afferente alla federazione GARR IDEM, può utilizzare questa modalità di autenticazione per accedere alla rete internet.

Attraverso il pulsante "Continue with IDEM GARR authentication" l'utente viene indirizzato su un server gestito dal GARR che gli presenta la seguente (o similare) pagina



dove deve scegliere il server relativo alla propria istituzione di appartenenza attraverso il menu presente nella pagina stessa.

Una volta fatta questa scelta premendo il pulsante "select" il browser viene indirizzato sulla pagina di autenticazione relativa alla propria istituzione (ad esempio l'Università di Ferrara come mostrato nella figura seguente:):

A questo punto l'utente deve introdurre le credenziali di accesso relative all'account che ha presso tale istituzione e, se sono corrette, ottiene l'autorizzazione all'accesso alla rete attraverso il servizio WiFi@PDCnr.

Nel caso sia errate o inesistenti, riceve un messaggio di errore direttamente dal server di autenticazione della propria istituzione di appartenenza. Se le credenziali continuano ad essere rifiutate, l'utente deve contattare la propria istituzione e non l'Area della Ricerca di Padova dato che i server di autenticazione relativi alla federazione IDEM non sono gestiti dall'area stessa.

Guida all'implementazione

Dispositivi hardware

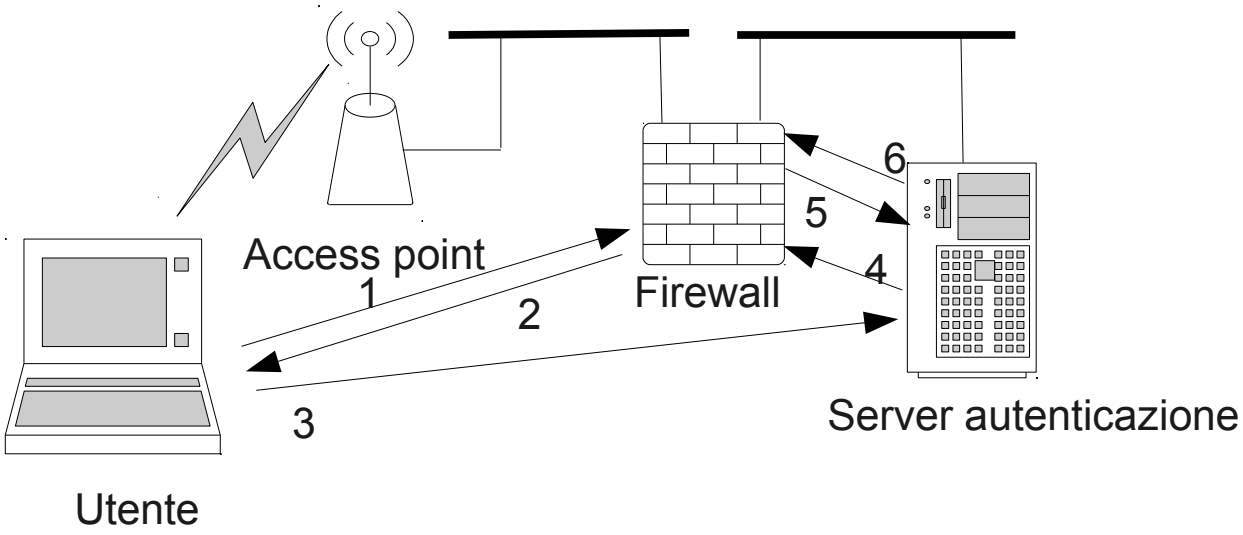
Il servizio WiFi@PDCnr viene gestito usando i seguenti dispositivi hardware:

controllore wireless Hpprocurve 5304XL
gestisce le reti wireless e fornisce il servizio DHCP per i nodir della rete ospiti

firewall Fortinet FG310B
firewall in configurazione ridondante gestisce l'autenticazione

server autenticazione
due macchine virtuali basate su Centos 5.5 in configurazione HA che gestiscono il server LDAP, RADIUS e HTTP utilizzati nelle procedure di autenticazione.

Procedura di autenticazione



Al primo tentativo di accesso con un browser WWW ad un sito qualunque della rete una regola presente nel firewall indirizza l'utente sul captive portal (1) del firewall stesso che a sua volta (2) attraverso una funzione javascript rimanda (3) al servizio web presente nel server di autenticazione.
Acquisite le credenziali d'accesso (username e password) dell'utente il servizio web le invia (4) al firewall che provvede alla loro autenticazione contattando (5) il server RADIUS presente nel server di autenticazione che validerà o meno (6) tali credenziali. Se le credenziali sono valide il firewall permetterà l'accesso alla rete connettendo l'utente al sito web richiesto altrimenti riproporrà la procedura di login.

Il server web gestisce due modi di autenticazione: CNR e IDEM. Nel caso di autenticazione CNR acquisisce in maniera esplicita le credenziali di login (username e password) e le invia come richiesta http al firewall che provvederà alla fase di autenticazione via il server radius.
Per l'autenticazione IDEM una volta che l'utente ha ricevuto l'abilitazione da parte della propria istituzione all'accesso del servizio IDEM WiFi@PDCnr il sistema crea delle credenziali temporanee basandosi sui dati identificativi fornitegli da IDEM, le inserisce nel file dedicato agli utenti IDEM

presente nel server radius e le invia come richiesta http al firewall che procederà all'autenticazione sempre via radius server.

L'autenticazione avviene sempre attraverso il server radius a cui è collegato il firewall. Il server radius utilizza i seguenti metodi per autenticare gli utenti:

- file per utenti temporanei e IDEM
- server LDAP per utenti permanenti gestiti dall'Area della Ricerca di Padova
- collegamento al server radius del CNR di Roma per utenti CNR

Configurazione server radius

File users definisce le politiche di gestione dei vari tipi di utente ed include i due file con i dati relativi a utenti temporanei e utenti IDEM.
Il file **users.guests** contiene le credenziali di utenti temporanei individuati da nome e numero di telefono :

mentre il file **users.idem** contiene quelle relativa agli utenti IDEM che sono identificati attraverso l'ID fornito dal nodo di autenticazione dell'istituzione a cui appartengono:

```
# https://idp.cnr.it/idp/shibboleth!https://wifi.pd.cnr.it/IDEMauth!NRP+6NknRoq7Svdf77DIpx2Y/pc=staff@cnr.it
NRP+6NknRoq7Svdf77DIpx2Y/pc=.idem Cleartext-Password := "d5sU1ri4JI6hU6gy6"
        Fall-Through = yes
```

file **users**

```
$INCLUDE users.guests
$INCLUDE users.idem

DEFAULT User-Name =~ "^([^,]+).guest", Fortinet-Vdom-Name=~ "ospiti"
       Tunnel-Type = 13,
       Tunnel-Medium-Type = 6,
       Tunnel-Private-Group-ID = 199,
       Reply-Message = "Welcome to fortinet, %u!",
       Reply-Message = "%{Auth-Type}",
       Fall-Through = no

DEFAULT User-Name =~ "^([^,]+).idem"
       Tunnel-Type = 13,
       Tunnel-Medium-Type = 6,
       Tunnel-Private-Group-ID = 199,
       Reply-Message = "Welcome to local GUESTS, %u!",
       Fall-Through = no

DEFAULT User-Name =~ "^([^,]+).local", Fortinet-Vdom-Name=~ "ospiti"
       Tunnel-Type = 13,
       Tunnel-Medium-Type = 6,
       Tunnel-Private-Group-ID = 199,
       Reply-Message = "Welcome to local GUESTS, %u!",
       Fall-Through = no

#DEFAULT     User-Name =~ "^([^,]+)@cnr.it", Proxy-To-Realm := "cnr.it"

#DEFAULT Auth-Type = LDAP, Simultaneous-Use := 1
#      Reply-Message := "%{User-Name} USER NAME",
#      Reply-Message := "%{NAS-Port-Type} NAS PORT",
#      Reply-Message := "%{Service-Type} SERVICE TYPE",
#      Reply-Message := "%{Called-Station-Id} ID",
#      Fall-Through = yes


DEFAULT NAS-Port-Type == Virtual, Service-Type == Administrative-User, Auth-Type
:=CONSOLE_ADM, Autz-Type:=CONSOLE_ADM
       Fall-Through = no

DEFAULT NAS-Port-Type == Virtual, Service-Type == Login-User, Auth-Type:=CONSOLE
_ADM, Autz-Type:=CONSOLE_ADM
        Service-Type = NAS-Prompt-User,
        cisco-avpair = "shell:priv-lvl=15",
        Fall-Through = no

DEFAULT NAS-Port-Type == Virtual, Service-Type == NAS-Prompt-User, Auth-Type:=CO
NSOLE, Autz-Type:=CONSOLE
        Fall-Through = no

DEFAULT NAS-Port-Type == Ethernet
        Fall-Through = no

DEFAULT User-Name =~ "^([^,]+)@guests", Connect-info=~ "vpn-ssl"
```

```
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 199,
        Reply-Message = "Welcome to GUESTS, %u!",
        Fall-Through = no

DEFAULT User-Name =~ "^([^,]+)@guests", Called-Station-Id=~ "^([^:]+):adr_guest"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 199,
        Reply-Message = "Welcome to GUESTS, %u!",
        Fall-Through = no

#DEFAULT Called-Station-Id=~ "^([^:]+):adr_guest", Auth-Type := Reject
#       Reply-Message := "%{Ldap-UserDn} BASE DN",
#       Reply-Message = "You cannot access adr_guest",
#       Fall-Through = no

DEFAULT User-Name =~ "^([^,]+)@guests", Called-Station-Id=~ "^([^:]+):adr_test"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 199,
        Reply-Message = "Welcome to GUESTS, %u!",
        Fall-Through = no

#DEFAULT Called-Station-Id=~ "^([^:]+):adr_test", Auth-Type := Reject
#       Reply-Message := "%{Ldap-UserDn} BASE DN",
#       Reply-Message = "You cannot access adr_guest",
#       Fall-Through = no

# ISIB users
DEFAULT User-Name =~ "^([^,]+)@isib"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 202,
        Reply-Message = "Welcome to ISIB, %u!",
        Fall-Through = no

# ITC users
DEFAULT User-Name =~ "^([^,]+)@itc"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 204,
        Reply-Message = "Welcome to ITC, %u!",
        Fall-Through = no

# ISTM users
DEFAULT User-Name =~ "^([^,]+)@istm"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 210,
        Reply-Message = "Welcome to ISTM, %u!",
        Fall-Through = no

# ICIS users
DEFAULT User-Name =~ "^([^,]+)@icis"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 212,
        Reply-Message = "Welcome to ICIS, %u!",
        Fall-Through = no

# IENI users
DEFAULT User-Name =~ "^([^,]+)@ieni"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 214,
        Reply-Message = "Welcome to IENI, %u!",
        Fall-Through = no

# IRPI users
DEFAULT User-Name =~ "^([^,]+)@irpi"
        Tunnel-Type = 13,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID = 215,
```

```
       Reply-Message = "Welcome to IRPI, %u!",
       Fall-Through = no

# ADR users
DEFAULT User-Name =~ "^([^,]+)@adr"
       Tunnel-Type = 13,
       Tunnel-Medium-Type = 6,
       Tunnel-Private-Group-ID = 216,
       Reply-Message = "Welcome to ADR, %u!",
       Fall-Through = no

# USP users
DEFAULT User-Name =~ "^([^,]+)@usp"
       Tunnel-Type = 13,
       Tunnel-Medium-Type = 6,
       Tunnel-Private-Group-ID = 217,
       Reply-Message = "Welcome to USP, %u!",
       Fall-Through = no

# ISAC users
DEFAULT User-Name =~ "^([^,]+)@isac"
       Tunnel-Type = 13,
       Tunnel-Medium-Type = 6,
       Tunnel-Private-Group-ID = 219,
       Reply-Message = "Welcome to ISAC, %u!",
       Fall-Through = no

DEFAULT        Proxy-To-Realm := "cnr.it", Fortinet-Vdom-Name=~ "ospiti"
DEFAULT        Proxy-To-Realm := "cnr.it", Fortinet-Vdom-Name=~ "area"
#DEFAULT       Proxy-To-Realm := "cnr.it", NAS-IP-Address=~ "150.178.100.15"
DEFAULT        Proxy-To-Realm := "cnr.it",  Called-Station-Id=~ "^([^:]+):adr_guest"

DEFAULT Auth-Type := Reject
       Reply-Message := "%{Ldap-UserDn} BASE DN",
       Reply-Message = "You cannot access here"
```

Nel file proxy.conf si deve definire come realm cnr.it i dati relativi al server radius del CNR di Roma:

```
realm cnr.it{
       type   =      radius
       authhost=     150.146.204.10:1812
       acchost=      150.146.204.10:1813
       secret =      xxxxxx
}
```

Comandi shell per la gestione della configurazione radius

checkRadiusGuest
data una stringa che descrive l'utente, lo script ritorna username e password concatenati da | relativi all'utente se esiste nel file users.guests altrimenti il messaggio nouser

```
#!/bin/bash
# $1 guest_description
USERFILE="/usr/local/etc/raddb/users.guests"
USERID=`grep -w "$1" $USERFILE| cut -d "|" -f 1 | cut -d "=" -f 2`
if [ -z $USERID ];then
echo "nouser"
else
PASSWORD=`grep -w $USERID.guest $USERFILE | cut -d "\"" -f 2`
echo "$USERID.guest|$PASSWORD"
fi
```

checkRadiusUniqueID
Dati un identificativo d'utente (username), una descrizione dell'utente e l'estensione del file che contiene i dati degli utenti, ritorna FALSE se esiste un utente con il medesimo identificativo ma con una differente descrizione, altrimenti ritorna TRUE (l'identificativo non esiste nel file oppure l'utente che lo possiede è quello cercato).

```
#!/bin/bash
# $1 uniqueID $2 full_description $3 user_type_extension
USERFILE="/usr/local/etc/raddb/users.$3"
FULLDESCR=`grep -w $1 $USERFILE | cut -d "|" -f 2`
[ -z $FULLDESCR ] && echo "TRUE" && exit 0
if [ "$2" == "$FULLDESCR" ]
then
echo "TRUE"
else
echo "FALSE"
fi
```

createRadiusUser

date le credenziali (username e password) relative ad un nuovo utente, un commento e l'estensione del file in cui si deve memorizzare il nuovo utente, se non esiste un utente con il medesimo username lo script inserisce i dati relativi al nuovo utente e fa ricaricare la configurazione al server radius, nel caso l'utente esista non fa niente. Alla fine ritorna il messaggio yes se ha inserito l'utente, no in caso contrario.

```
#!/bin/bash
# $1 username $2 password $3 comment $4 user_type
USERFILE="/usr/local/etc/raddb/users.$4"
INSERTED="no"
PASSWORD=`grep -w $1 $USERFILE | cut -d "\"" -f 2`
if [ -z $PASSWORD ];then
PASSWORD=$2
echo -e "# $3\n$1 Cleartext-Password := \"$PASSWORD\"\n          Fall-Through = yes" >>
$USERFILE 2>>/tmp/tmplog
kill -1 `cat /usr/local/var/run/radiusd/radiusd.pid`
INSERTED="yes"
fi
echo $INSERTED
```

clearRadiusUser

elimina tutti gli utenti presenti nei file users.guests e users.idem cancellando tali file e ricreandoli vuoti, dopo la cancellazione fa ricaricare la configurazione al radius server. Da attivare via cron per cancellare alla mezzanotte (per la precisione alle 23:55) di ogni sabato tutti gli account di tipo guest e idem:

```
55 23 * * 6 /usr/local/sbin/clearRadiusUser
```

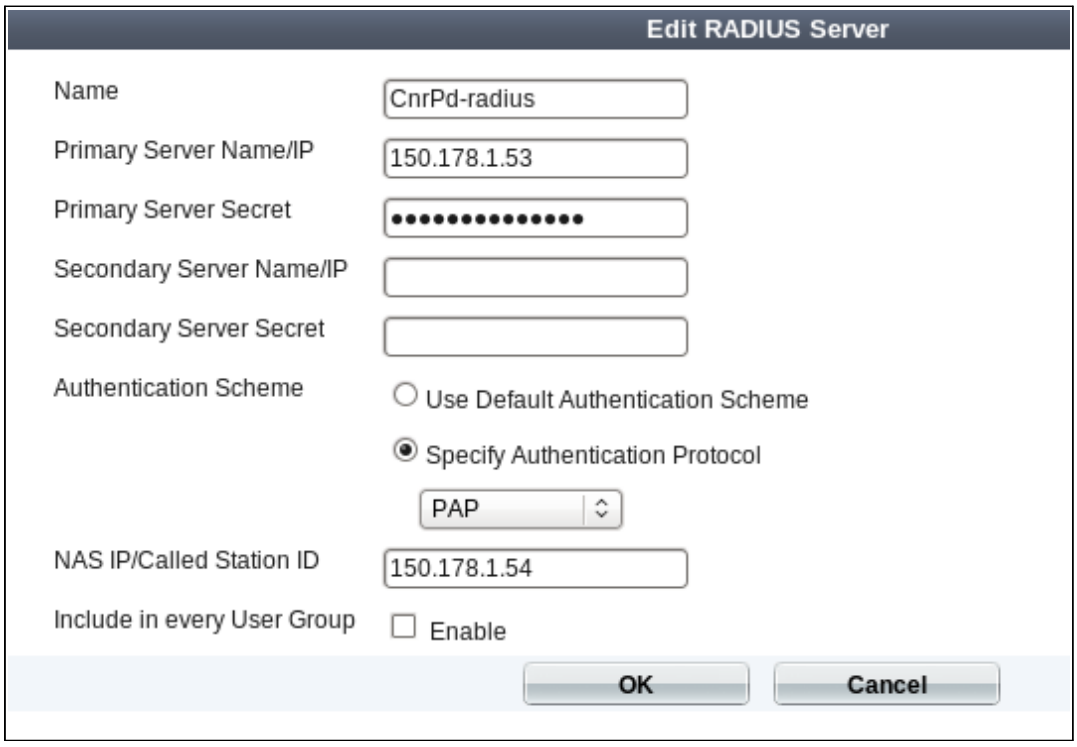
```
#!/bin/bash
RADIUSFILE="/usr/local/etc/raddb/users."
for ext in guests idem
do
/bin/rm $RADIUSFILE$ext
touch $RADIUSFILE$ext
done
kill -1 `cat /usr/local/var/run/radiusd/radiusd.pid`
```

Configurazione Fortinet

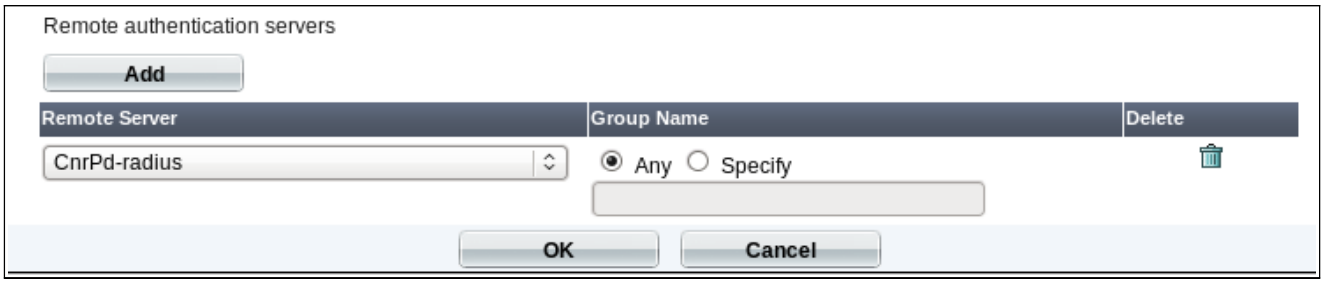
Nel dominio VDOM **ospiti**

autenticazione utenti

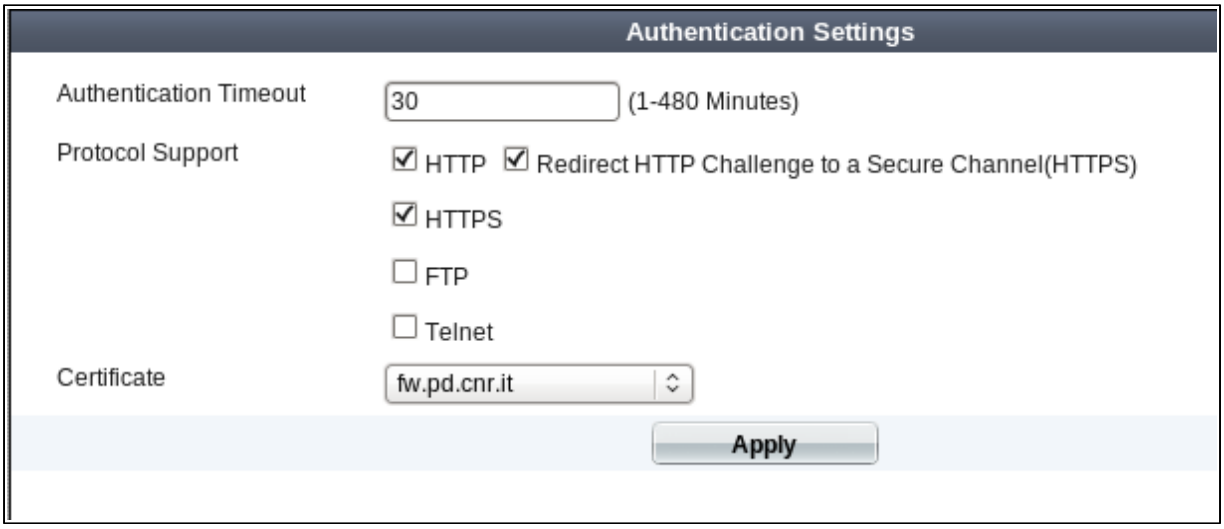
settaggio di un radius server
User → Remote → RADIUS



creazione gruppo utenti
User → User Group → User Group
crea un nuovo gruppo utenti di nome wifi-guest che abbia come membro il server remoto di tipo RADIUS CnrPd-radius:



settaggio modalità di autenticazione del captive portal
User → Authentication



modifica file captive portal
si devono modificare i messaggi relativi alla fase di login per poter gestire l'autenticazione con server web remoto.

I messaggi di default devono essere sostituiti con il contenuto del file **FortiLoginPage:**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN">
<html><head>
<title>Firewall Authentication</title>
</head>
<body>
<noscript>
 <h2>You must enable javascript in your browser to access the net.<br>
Please, enable it and restart the browser</h2>
</noscript>
<div class="pageContent">
<form action="https://wifi.pd.cnr.it/wifi/fortiAuth.php" method="post">
<input type="hidden" id="url" name="AUTH_POST_URL" value="%%AUTH_POST_URL%%">
<input type="hidden" name="REDIRID" value="%%REDIRID%%">
<input type="hidden" name="PROTURI" value="%%PROTURI%%">
<input type="hidden" name="MAGICID" value="%%MAGICID%%">
<input type="hidden" name="MAGICVAL" value="%%MAGICVAL%%">
<input type="hidden" name="QUESTION" value="%%QUESTION%%">
<input type="hidden" name="USERNAMEID" value="%%USERNAMEID%%">
<input type="hidden" name="PASSWORDID" value="%%PASSWORDID%%">
<input type="hidden" name="failed" value="no"> <!-- //yes in loginFailedPage -->
<!-- <input type="submit" value= "Continue">-->
</form>
     </div>
       <script type="text/javascript">
        //<![CDATA[
           function Process()
            {
              document.getElementById("url").value =document.location.href;
              document.forms[0].submit();
            }
            Process();
        //]]>
        </script>
</body></html>
```

che se nel browser è abilitata la funzionalità javascript invia in maniera automatica il browser all'URL `https://wifi.pd.cnr.it/wifi/fortiAuth.php` passandogli i parametri utilizzati dal firewall per gestire l'autenticazione altrimenti visualizza il messaggio definito dal tag `<noscript>` .

Per modificare i messaggi si deve andare in System → Config → Replacement Messages e sotto la voce Authentication modificare i messaggi relativi alla Login page e alla Login failed page sostituendo il contenuto originale con il contenuto del file FortiLoginpage. Inoltre nel caso della

Login failed page bisogna modificare il valore del campo `failed` in `yes` in modo da avere la seguente linea:

```
<input type="hidden" name="failed" value="yes"> <!-- //yes in loginFailedPage -->
```

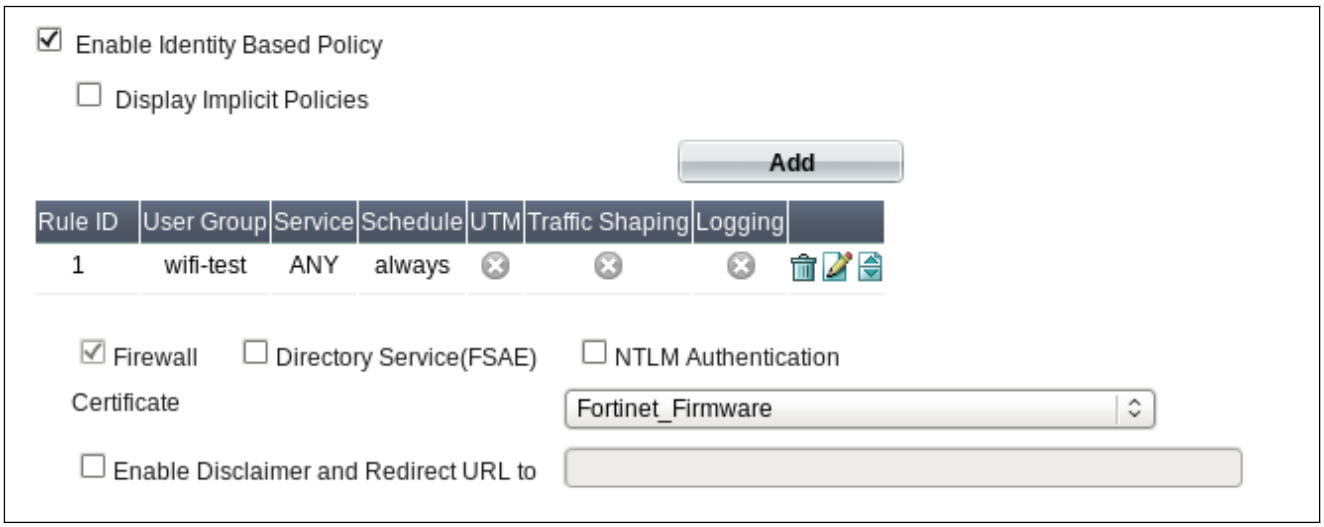
regole di firewall
Prima della definizione delle regole si devono definire in Firewall → Address → Address gli indirizzi dei nodi coinvolti nel processo di autenticazione:

- il server web
- tutti i nodi utilizzati nell'autenticazione IDEM GARR

e in Firewall → Address → Group il gruppo IDP-idem che raggruppa tutti i nodi IDEM.

in Firewall → Policy definire due regole:

1. la prima regola serve per permettere ad un nodo cliente di accedere direttamente ai nodi coinvolti nell'autenticazione, quindi ha come destinatari il server web e il gruppo IDP-idem, servizi abilitati solamente HTTP e HTTPS e nessun blocco per cui i nodi cliente possono accedervi senza alcun problema.
2. la seconda regola che ha come destinatari qualunque nodo e permette qualunque servizio, richiede al nodo cliente di autenticarsi prima di permettergli l'accesso; questa funzionalità viene abilitata utilizzando il flag "Enable Identity Based Policy"



Identificazione dei nodi IDP IDEM

È necessario ottenere l'indirizzo IP di tutti i nodi utilizzati per l'autenticazione degli utenti idem per poterli inserie nel firewall e permettere ai nodi utente di accedervi durante la procedura di autenticazione.

Inoltre esistendo nodi IDP che rinviano ad altri nodi, di solito sistemi CAS, che eseguono effettivamente l'autenticazione, non è sufficiente l'indirizzo del nodo IDP, ma bisogna individuare anche l'indirizzo IP del nodo a cui si è rinviati.

Trovati gli IP, sia dei nodi IDP sia di quelli a cui si è rinviati durante la fase di autenticazione, si devono definire gli indirizzi nel firewall e inserirli nel gruppo che raggruppa tutti i nodi IDEM di autenticazione.
Convenzione nomi

Sono stati sviluppati due script presenti in /usr/local/sbin per ricavare gli indirizzi IP di tutti i nodi coinvolti nell'autenticazione IDEM e per monitorare eventuali cambiamenti nella definizione di tali nodi:

listIDEMIDP

Script bash che genera un elenco (IP, FQDN) di tutti i nodi coinvolti nella autenticazione IDEM compresi gli eventuali nodi CAS. Utilizzando wget simula l'uso del browser per accedere ad un servizio IDEM per poi provare tutti gli URL dei nodi IDP presenti nel menu select.

```
#!/bin/bash
# listIDEMIDP 0.0
# elenca i nodi IDP della federazione IDEM con gli eventuali nodi CAS
# vuole come parametro l'URL di un servizio IDEM
# ad esempio
#    listIDEMIDP http://vconf.garr.it/econfportal-shib/www-aai/mcu_reservation_manager
# testato su:
#   ubuntu 10.10 con GNU Wget 1.12
#   CentOS release 5.5 (Final) con GNU Wget 1.11.4 Red Hat modified
#
[ -z "$1" ] && echo "usage: $0 URL" && exit
# file temporanei utilizzati dallo script
# contenuto pagina di autenticazione del servizio IDEM
servicepage=`mktemp -q /tmp/listIDEMIDP.XXX`
# elenco URL degle IDP
idplist=`mktemp -q /tmp/listIDEMIDP.XXX`
#
idppage=`mktemp -q /tmp/listIDEMIDP.XXX`
# ottiene la pagina di autenticazione fornita dal servizio IDEM
wget --no-check-certificate -S -O - "$1" >$servicepage 2>&1
# estrae il nome del host utilizzato come wayf e il protocollo usato
 wayfhost=`grep Location $servicepage | tail -n 1 | cut -d "/" -f 3| cut -d "?" -f 1`
 protocol=`grep Location $servicepage | tail -n 1 | cut -d ":" -f 2|tr -d " "`
# estrae dalla form l'URL a cui inviare le richieste
action=`grep action $servicepage | cut -d '"' -f 10 | sed -e 's\amp;\|\g' | tr -d "|"`
#estrae gli URL degli IDP dai campi option del menu select
grep "option value" $servicepage| grep http | cut -d '"' -f 2 >$idplist
# per ogni IDP
while read url
do
 # estrae da URL del IDP  il nome FQDN del host e calcola il corrispondente IP
 idphost=`echo $url |cut -d "/" -f 3`
 idpip=`host $idphost | grep -v IPv6 |grep address | cut -d " " -f 4`
 echo "$idpip $idphost"
# ottieni la pagina di autenticazione dal nodo IDP
 wget --no-check-certificate -S -O - "$protocol://$wayfhost$action" --post-
data="user_idp=$url" >$idppage 2>&1
# estrai dall'ultimo header di tipo Location l'indirizzo del nodo autenticatore CAS
# presupponendo che wget abbia seguito tutte le ridirezioni
 cashost=`grep Location $idppage | tail -n 1 | cut -d "/" -f 3| cut -d "?" -f 1| cut -d ":"
-f 1`
 casip=`host $cashost | grep -v IPv6 |grep address | cut -d " " -f 4`
# stampa i dati del nodo CAS se è differente dal nodo IDP
 if [ "$idpip" != "$casip" ]
 then
  echo "$casip $cashost"
 fi
done <$idplist
rm -f $servicepage $idplist $idppage
exit
```

Ad esempio con il comando

```
listIDEMIDP http://vconf.garr.it/econfportal-shib/www-aai/mcu_reservation_manager
```

si ottiene la lista:

```
193.204.6.243 shidp.caspur.it
131.175.1.68 idp2.cilea.it
131.175.1.145 caoss.cilea.it
130.186.28.14 idp-staf-prod.cineca.it
150.145.80.71 idp.ba.cnr.it
192.167.161.15 biblio.bo.cnr.it
150.145.48.155 shib2.to.cnr.it
146.48.68.189 idea.ifc.cnr.it
```

```
146.48.98.166 idp.iit.cnr.it
146.48.92.50 idem-idp.ilc.cnr.it
150.145.48.156 shibidp.to.cnr.it
150.146.205.36 idp.cnr.it
193.206.158.60 idp.dir.garr.it
193.206.190.48 idp2.idem.garr.it
192.167.173.4 idem.ced.inaf.it
193.204.90.137 identity.istat.it
131.175.187.34 shibidp.polimi.it
131.175.187.35 aunicalogin.polimi.it
193.206.102.29 idem.unina2.it
212.189.128.20 idem.unisalento.it
150.217.6.133 shibboleth.unifi.it
157.138.7.177 idp.unive.it
157.138.251.11 idp.iuav.it
193.205.128.200 idem.univpm.it
193.204.176.33 shibidp.uniba.it
137.204.24.76 idp.unibo.it
192.146.242.54 idp.unica.it
192.167.219.13 identity.unife.it
149.132.2.23 idp.unimib.it
149.132.2.49 bridge.si.unimib.it
155.185.254.93 idp.unimore.it
143.225.200.126 idemshibb.unina.it
192.167.9.168 idp.uniparthenope.it
147.162.199.12 shibidp.cca.unipd.it
160.78.48.193 shibidp.unipr.it
160.78.48.140 cas.unipr.it
193.204.35.139 idp.unipv.it
193.205.139.35 idp.uniroma3.it
130.192.112.2 ip1.rettorato.unito.it
193.205.207.19 idp.unitn.it
193.205.2.19 idp.uniurb.it
```

checkIDEMIDP

Script bash per segnalare via e-mail eventuali cambiamenti nella lista dei nodi IDP. Esecuzione temporizzata da cron alle 20.00 di ogni sabato con la definizione:
0 20 * * 6 /usr/local/sbin/checkIDEMIDP

```
#!/bin/bash
# checkIDEMIDP 0.0
# esecuzione temporizzata da cron alle 20.00 di ogni sabato
# 0 20 * * 6 /usr/local/sbin/checkIDEMIDP

# elenco e-mail dove inviare modifiche
emails="paolo@isib.cnr.it claudio@isib.cnr.it"
# file usati
# elenco attuale dei nodi IDP
idplist="/etc/shibboleth/idplist"
# nuovo elenco
newlist="/etc/shibboleth/newidplist"
# dati da inviare
sendlist=`mktemp /tmp/checkIDEMIDP.XXX`
send="NO"
echo "Subject: Modifica elenco nodi IDEM IDP su firewall">$sendlist
echo >>$sendlist
echo "Nodi da inserire:" >>$sendlist
echo "" >>$sendlist
/usr/local/sbin/listIDEMIDP https://wifi.pd.cnr.it/IDEMauth >$newlist
if [ -f "$idplist" ]
then
 while read ip name
 do
  if ! grep $ip $idplist >/dev/null; then
   echo "$ip $name" >>$sendlist
   send="YES"
  fi
 done <$newlist
echo >>$sendlist
echo "Nodi da cancellare:" >>$sendlist
echo "" >>$sendlist
 while read ip name
 do
  if ! grep $ip $newlist >/dev/null; then
```

```
   echo "$ip $name" >>$sendlist
   send="YES"
  fi
 done <$idplist
else
 cat $newlist >>$sendlist
 send="YES"
fi
if [ "$send" == "YES" ]
then
  echo -e "\n\nDopo la modifica del firewall eseguire il comando:\n\ncp    $newlist
$idplist\n\nper disabilitare invio di ulteriori mail" >>$sendlist
  sendmail $emails <$sendlist
fi
rm -f $sendlist
```

Configurazione server web

server web apache configurato con ssl nella directory /var/www/html.ssl

installazione di shibboleth

creazione di due directory:

/var/www/html.ssl/wifi
contiene il file d'ingresso e tutti i file utilizzati per il processo di autenticazione CNR

/var/www/html.ssl/IDEMauth
è la directory il cui accesso è controllato via shibboleth utilizzata per l'autenticazione di utenti IDEM

Autenticazione CNR

La directory /var/www.html.ssl/wifi contiene i file che gestiscono l'autenticazione CNR.

I seguenti file (hiddenFortiParams.inc, extractFortiParams.inc e style.css) file definiscono elementi utilizzati da altri file e vengono da questi inclusi:

hiddenFortiParams.inc
contiene le definizioni di tipo input per le form html delle variabili utilizzate dal firewall per individuare la connessione da abilitare:

```
<input type="hidden" name="AUTH_POST_URL" value="<?php echo $AUTH_POST_URL;?>">
<input type="hidden" name="REDIRID" value="<?php echo $REDIRID;?>">
<input type="hidden" name="PROTURI" value="<?php echo $PROTURI;?>">
<input type="hidden" name="MAGICID" value="<?php echo $MAGICID;?>">
<input type="hidden" name="MAGICVAL" value="<?php echo $MAGICVAL;?>">
<input type="hidden" name="QUESTION" value="<?php echo $QUESTION;?>">
<input type="hidden" name="USERNAMEID" value="<?php echo $USERNAMEID;?>">
<input type="hidden" name="PASSWORDID" value="<?php echo $PASSWORDID;?>">
```

extractFortiParams.inc
contiene le istruzioni php per assegnare a variabili php i valori delle variabili utilizzate dal firewall per individuare la connessione da abilitare che sono memorizzate in campi prestabiliti delle form html:

```
<?php
$AUTH_POST_URL=$_POST["AUTH_POST_URL"];
$REDIRID=$_POST["REDIRID"];
$PROTURI=$_POST["PROTURI"];
$MAGICID=$_POST["MAGICID"];
$MAGICVAL=$_POST["MAGICVAL"];
$QUESTION=$_POST["QUESTION"];
$USERNAMEID=$_POST["USERNAMEID"];
$PASSWORDID=$_POST["PASSWORDID"];
?>
```

style.css
definisce l'aspetto grafico comune a tutti i file html utilizzati per l'autenticazione:

```
body{font-family: verdana;font-size: 12px; text-align: center;}

p{padding: 3px 0px;}
```

```
h1,h2,h3{margin: 0;}
h1{font-family: verdana; font-size: 20px; font: bold}
h2{font-size: 1.5em}
h3{padding:3px; color:#9598cb; background-color:#cccccc}
h3{font-size: 1.2em}
li{padding: 3px 0px;}

div#content{text-align:left; padding: 15px;}
div#piccolo{font-size: 10px}
div#login{margin-left:auto; margin-right:auto; width:520px; border: 1px solid #cccccc}
div#form{margin-left:auto; margin-right:auto; width:520px; border: 1px solid #cccccc}
div#centrato{margin:auto; width:550px; border: 1px solid black}
```

fortiAuth.php

file principale per la gestione dell'autenticazione che viene attivato in maniera automatica attraverso il firewall. Presenta due form: una che indirizza all'autenticazione CNR associata all URL `https://wifi.pd.cnr.it/wifi/CNRautentication.php` mentre l'altra indirizza all'autenticazione IDEM `https://wifi.pd.cnr.it/IDEMauth/`

```
<?php

//header('P3P: CP="IDC DSP COR CURa ADMa OUR IND PHY ONL COM STA"');
header('P3P: CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS OUR IND CNT"');
session_start();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<head>
 <title> WiFi@PdCNR - User Authentication</title>
 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
 <meta http-equiv="P3P" content='CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS
OUR IND CNT"'>
 <link rel="stylesheet" href="style.css" type="text/css">
</head>
<body >
<div id="centrato">
<h3>
<center><table>
<tr><td style="vertical-align:middle"><img src="/wifi/images/cnr-padova-logo-small.png"
width="60" alt="ISIB CNR - Istituto di Ingegneria B
iomedica"></td> <td align="center">
 Welcome to <br>WiFi@PdCNR - User Authentication</td></tr>
</table>
</center></h3>
<br>
By using this service, you agree to be bound by <br>the WiFi@PdCNR<a
href="http://wifi.pd.cnr.it/description.php#termsofservice" target="_b
lank">
Terms of Service</a>.<br>
<br />
<?php
include "extractFortiParams.inc";
if (isset($_POST["failed"]))
  if ($_POST["failed"]=="yes")
    echo "<h2>Authentication failed, Please retry.</h2><br>";
?>

<div id="login">
<div id="content">
<table>
<tr><td style="vertical-align:middle"><img style="border: 0px solid ;"
src="/wifi/images/cnr-padova-logo-small.png" height="50" alt="CNR Pa
dova logo"></td> <td>Authentication</td></tr>
</table>
for users with login credentials (username and password) issued by the CNR Research Area of
Padua
or by the CNR Headquarter in Rome.
</div>
<form action="https://wifi.pd.cnr.it/wifi/CNRautentication.php" method="post">
 <?php include "hiddenFortiParams.inc";?>
 <input type="submit" value="Continue with CNR authentication">
</form>
```

```
<br>
</div>
<br>
<div id="login">
<div id="content">
<table>
<tr><td style="vertical-align:middle"><img style="border: 0px solid ;"
src="/wifi/images/IDEM_logo.png" height="50" alt="IDEM GARR logo"></
td> <td>Authentication </td></tr>
</table>
for users belonging to an istitution member of the GARR IDEM federation.<br>
</div>
<form action="https://wifi.pd.cnr.it/IDEMauth/" method="get">
 <?php include "hiddenFortiParams.inc";?>
 <input type="submit" value="Continue with IDEM GARR authentication">
</form>
<br>
</div>
<br>
</div>
<br>
</body>
</html>
```

CNRautentication.php

php script che gestisce le procedure relative all'autenticazione di utenti con credenziali CNR (Area della Ricerca di Padova o CNR di Roma) e gestione degli utenti temporanei che sono ospiti dell? area di padova Lo script genera tre form: una per l'autenticazione, le altre due senza campi per la gestione degli account temporanei.

La form relativa all'autenticazione richiede le credenziali (username e password) e le invia attraverso la URL presente nella variabile $AUTH_POST_URL al firewall che eseguirà l'autenticazione attraverso il server radius.

Per quanto riguarda la gestione degli account temporanei la form utilizzante lo URL https://wifi.pd.cnr.it/wifi/accountrequest.php permette di iniziare la procedura di richiesta di un nuovo account temporaneo mentre la form con URL https://wifi.pd.cnr.it/wifi/accountretreive.php permette di ottenere i dati di un account esistente.

```
<?php

//header('P3P: CP="IDC DSP COR CURa ADMa OUR IND PHY ONL COM STA"');
header('P3P: CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS OUR IND CN
T"');
session_start();
if (!empty($_SESSION['WIFIPdCNRuser'])) {
 $guest=$_SESSION['WIFIPdCNRuser'];
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.or
g/TR/html4/loose.dtd">
<head>
 <title> WiFi@PdCNR - CNR Authentication</title>
 <meta http-equiv="P3P" content='CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS
OUR IND CNT"'>
 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
 <link rel="stylesheet" href="style.css" type="text/css">
</head>
<body >
<?php
include "extractFortiParams.inc";
?>
<div id="centrato">
<h3><img src="/wifi/images/cnr-padova-logo-small.png" width="60" alt="ISIB CNR -
 Istituto di Ingegneria Biomedica">
 WiFi@PdCNR - CNR Authentication</h3>
<br>
<br />
<div id="login">
<br />
<b>Login credentials</b>
```

```
<br> issued by the CNR Research Area of Padua <br>or by CNR Headquarter in Rome

<br />

<form action=<?php echo $AUTH_POST_URL;?> method="post">
 <input type="hidden" name="<?php echo $REDIRID;?>" value="<?php echo $PROTURI;? >">
 <input type="hidden" name="<?php echo $MAGICID;?>" value="<?php echo $MAGICVAL; ?>">
 <center>
 <table>
 <tr><td align="right"> Username:</td><td>
 <input name=<?php echo $USERNAMEID;?> id="un" style="width:245px"
 <?php
  if (isset($guest))
    echo 'value="'.$guest.'"';
 ?>
 /></td></tr>
 <tr><td align="right">Password:</td><td>
 <input name=<?php echo $PASSWORDID;?> id="pd" type="password" style="width:245p x">
 </td></tr>
 </table>
 </center>
 <input type="submit" value="Login" />
</form>
<br />
</div>
<br>
<div id="login">
<br />
<b>AdRPD Guests Management</b>
<br />
<br />
Guests of AdRPD can apply for an account:
<br />
<form action="https://wifi.pd.cnr.it/wifi/accountrequest.php" method="post">
 <?php       include "hiddenFortiParams.inc";?>
 <input type="submit" value="Request guest account">
</form>
<br />
or retrieve the login credentials if they have an active account:
<br />
<form action="https://wifi.pd.cnr.it/wifi/accountretrieve.php" method="post">
 <?php       include "hiddenFortiParams.inc";?>
 <input type="submit" value="Retrieve login credentials">
</form>
<br />
</div>
<br>
</div>
</body>
</html>
```

accountrequest.php

Script per la gestione della form di richiesta di un account temporaneo.
Se non vengono passati campi oppure vi sono errori nei dati immessi presenta la form con eventuali messaggi di errore, altrimenti include il file confirmform.inc che crea una richiesta di conferma. Le variabili $institution e $description contengono rispettivamente gli acronimi e le descrizioni delle istituzioni presenti in area e vengono utilizzate per la creazione del menu per la scelta dell'istituto a cui afferisce il richiedente.

```
<?php

//header('P3P: CP="IDC DSP COR CURa ADMa OUR IND PHY ONL COM STA"');
header('P3P: CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS OUR IND CNT"');
session_start();

$cryptinstall="./crypt/cryptographp.fct.php";
 include $cryptinstall;

function is_invalid($token)
{
  $result=true;
  if ($fh = fopen("/usr/local/etc/raddb/tokens.lst", "r")) {
    while (!feof($fh)) {
```

```
      $line = fgets($fh);
      $token_data=explode("|",$line);
      if ($token==$token_data[0]) $result=false;
    }
    fclose($fh);
  }
  return $result;
}

function bad_format($str){
$replace_pairs=array(
'"'=>'',
'@'=>'',
'#'=>'',
'|'=>'',
'&'=>'',
'%'=>'',
'$'=>'',
'^'=>'',
'£'=>'',
'['=>'',
']'=>''
);
if ($str==strtr ( $str , $replace_pairs )) return FALSE;
else return TRUE;
}

if(isset($_POST['word']))
{  if (chk_crypt($_POST['word']))
     $word_ok = "yes";
     else $word_ok = "no" ;
} else {
      $word_ok = false;
}

include "extractFortiParams.inc";

$institution=array(
"",
"icis",
"idpa",
"ieni",
"igi",
"irpi",
"isac",
"isib",
"istc",
"istm",
"itc",
"spp"
);

$description=array(
""=>"",
"icis"=>"ICIS-Ist. Chimica Inorganica e Superfici",
"idpa"=>"IDPA-Ist. Dinamica Processi Ambientali",
"ieni"=>"IENI-Ist. Energetica e Interfasi",
"igi"=> "IGI -Ist. Gas Ionizzati",
"irpi"=>"IRPI-Ist. Ricerca Protezione Idrogeologica",
"isac"=>"ISAC-Ist. Scienze Atmosfera e Clima",
"isib"=>"ISIB-Ist. Ingegneria Biomedica",
"istc"=>"ISTC-Ist. Scienze e Tecnologie Cognizione",
"istm"=>"ISTM-Ist. Scienze e Tecnologie Molecolari",
"itc"=> "ITC -Ist. Tecnologie della Costruzione",
"spp"=> "SPP -Ser. Prevenzione e Protezione"
);

$check="yes";
if (! isset($_POST["word"])) {
$word='';
$check="no";
}
else $word=$_POST["word"];
if (! isset($_POST["surname"])) {
$surname='';
```

```
$check="no";
}
else $surname=$_POST["surname"];
if (! isset($_POST["name"])) {
$name='';
$check="no";
}
else $name=$_POST["name"];
if (! isset($_POST["token"])) {
$token='';
$check="no";
}
else $token=$_POST["token"];

if (! isset($_POST["phone"])) {
$phone='';
$check="no";
}
else $phone=$_POST["phone"];

if (! isset($_POST["istituto"])) {
$istituto='';
$check="no";
}
else $istituto=$_POST["istituto"];

$error_msg="";
$surname_err="";
$name_err="";
$token_err="";
$phone_err="";
$word_err="";
$istituto_err="";
if ($check=="yes"){
if (empty($surname) or bad_format($surname)){
$surname_err="<font color=\"red\">*</font>";
}
if (empty($name) or bad_format($name)){
$name_err="<font color=\"red\">*</font>";
}
if (empty($token)or is_invalid($token)){
$token_err="<font color=\"red\">*</font>";
}
if (empty($phone) or ! ereg ("^\+[0-9]+$", $phone)){
$phone_err="<font color=\"red\">*</font>";
}
if (empty($istituto)){
$istituto_err="<font color=\"red\">*</font>";
}
if (!(empty($surname_err) and empty($name_err) and empty($token_err) and empty($phone_err)
and empty($istituto_err))){
$error_msg="* empty or bad format field ";
}
if (empty($word) or $word_ok=="no"){
$word_err="<font color=\"red\">#</font>";
$error_msg=$error_msg."<br># empty or wrong word";
}

}//endif ($check=="yes")

if($word_ok!==false)
{
       if($word_ok=="yes" and $error_msg=='')
       {
              include "confirmform.inc";
exit(0);
       }
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title> WiFi@PdCNR - Guest Account Request</title>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
```

```
<link rel="stylesheet" href="style.css" type="text/css">
</head>

<body >

<div id="centrato">
<h3><img src="/wifi/images/cnr-padova-logo-small.png" width="60" alt="CNR Padova logo">
 WiFi@PdCNR Account Request</h3>
<div id="content">
<p>Only guests of an host istitution belonging to the CNR Research Area of Padova can apply
for a guest account
for the WiFi@PdCNR service.
<p>Mobile field must be in the ITU-T <acronym title="+&lt;country code&gt;&lt;mobile
number&gt;">E.123</acronym> format without spaces
(e.g. +393408317256)
 </div>
<div id="form">
<br>
<center>
<?
  if ($error_msg) {
echo "<font color=\"red\">".$error_msg."</font>";
}
?>
<form action="<?=$_SERVER['PHP_SELF']?>" method="post">
<?php        include "hiddenFortiParams.inc";?>
<table cellpadding="0" cellspacing="0">
<tr><td align="right">token:</td><td><input type="text" name="token" size="27" value="<?
=htmlspecialchars($token)?>">
<?php
  if ($token_err) echo $token_err;
?>
</td></tr>
<tr><td align="right">name:</td><td><input type="text" name="name" size="27" value="<?
=htmlspecialchars($name)?>">
<?php
  if ($name_err) echo $name_err;
?>
</td></tr>
<tr><td align="right">surname:</td><td><input type="text" name="surname" size="27"
value="<?=htmlspecialchars($surname)?>">
<?php
  if ($surname_err) echo $surname_err;
?>
</td></tr>
<tr><td align="right">mobile:</td><td><input type="text" name="phone" size="27" value="<?
=htmlspecialchars($phone)?>">
<?php
  if ($phone_err) echo $phone_err;
?>
</td></tr>
</td></tr>
<tr><td align="right">host institution:</td><td>
<?php
    echo "<select name=\"istituto\">";
    foreach ($institution as $labname){ // lista istituti
    echo "<option value=\"".$labname."\"";
if ($istituto==$labname) echo " selected";
    echo "> ".$description[$labname];
    echo "</option>";
    }
    echo " </select>";
  if ($istituto_err) echo $istituto_err;
?>
</td></tr>
<tr><td colspan="2" align="center">
<?php dsp_crypt(0,1); ?>
</td></tr>
<tr><td colspan="2" align="center">type the word above: <input type="text" name="word">
<?php
  if ($word_err) echo $word_err;
?>
  </td></tr>
<tr><td colspan="2" align="center"><input type="submit" value="Request account"></td></tr>
</table><br />
```

```
</form>
</center>
</div>
<br>
</div>
</body>
</html>
```

confirmform.inc

Questo script gestisce la creazione di un account temporaneo generando una richiesta di conferma.

Se esiste già un utente con i dati introdotti lo segnala ed invita l'utente ad eseguire la procedura per riottenere le credenziali altrimenti crea le nuove credenziali, invia la password all'utente via SMS e genera un form di conferma con URL https://wifi.pd.cnr.it/wifi/accountcreation.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
 <title> WiFi@PdCNR - Account Confirmation</title>
 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
 <link rel="stylesheet" href="style.css" type="text/css">
</head>

<body >

<div id="centrato">
<h3>
<img src="/wifi/images/cnr-padova-logo-small.png" width="60" alt="CNR Padova logo">
 WiFi@PdCNR Account Confirmation
</h3>
<div id="content">
<p>

<?php

function make_password($length,$strength=0) {
  $vowels = 'aeiouy';
  $consonants = 'bdghjmnpqrstvwxz';
    $numbers = '23456789';
  srand(time());
  // inizia con una lettera
   if ( rand()% 2 == 1) {
      $password = $consonants[(rand() % strlen($consonants))];
    } else {
        $password = $vowels[(rand() % strlen($vowels))];
    }
  for ($i = 1; $i < $length; $i++) {
    switch (rand() % 3) {
      case 0:
      $password .= $consonants[(rand() % strlen($consonants))];
        break;
      case 1:
       $password .= $vowels[(rand() % strlen($vowels))];
         break;
      case 2:
        $password .= $numbers[(rand() % strlen($numbers))];
        break;
       }
  }
  return $password;
}

function make_id(){
  $vowels = 'aeiouy';
  $consonants = 'bdghjmnpqrstvwxz';
    $numbers = '0123456789';
  srand(time());
```

```
  $id=$consonants[(rand() % strlen($consonants))];
  $id .= $vowels[(rand() % strlen($vowels))];
  for ($i = 1; $i < 5; $i++) {
        $id .= $numbers[(rand() % strlen($numbers))];
      }
  return $id;

}


function compute_uniqueID($par){
//$fullID = hash("sha512", $par);
  do{
    $i=0;
    //$shortID=substr($fullID, $i, 7);
    $shortID=make_id();
    $test=exec('sudo flock /var/lock/localUser /usr/local/sbin/checkRadiusUniqueID "id='.
$shortID.'" "'.$par.'" guests');
    $i=$i+1;
  } while ($test=="FALSE");
  return $shortID;
}

//$local_user= preg_replace('/[^(\x20-\x7F)]*/','', $local_user); // elimina caratteri non-
ascii
// normalizza name e surname
$name=ucwords(strtolower($name));
$surname=ucwords(strtolower($surname));
$guestDescription=$name.",".$surname.",".$phone;
/**/
//controlla esistenza account nel file users.guests del radius server
$password=exec('sudo flock /var/lock/localUser /usr/local/sbin/checkRadiusGuest "'.
$guestDescription.'"');
if ($password!="nouser") { //esiste account
  echo "An account for <b>$name $surname</b> phone num. <b>$phone</b> already exits.";
  echo "<p>Use this button to retrieve username and password";
  echo '<form action="https://wifi.pd.cnr.it/wifi/accountretrieve.php" method="POST">';
  include "hiddenFortiParams.inc";
  echo '<input type="submit" name="submit" value="Retrieve login credentials" />';
  echo "</form>";
  echo "</div></div></body></html>";
  exit(0);
}

$password=make_password(7,4);
$uniqueID=compute_uniqueID($guestDescription);
$guest=$uniqueID.".guest";
$guestInfo='id='.$uniqueID.'|'.$guestDescription;
$now=time();
$dw = date( "w", $now); //giorno corrente della settimana: 0 domenica 6 sabato
$limit=$now + (6-$dw)*3600*24; // sabato seguente
$deaddate=date("l, j M Y",$limit);
//salva dati su file per passarli a accountcreation
$tmpdir="/tmp/uniqueIDs/";
system("mkdir -p $tmpdir");
$cmd='echo -e "<?php\n '.
     '\$requesttime=\"'.$now.'\";\n'.
     '\$uniqueID=\"'.$uniqueID.'\";\n'.
     '\$surname=\"'.$surname.'\";\n'.
     '\$name=\"'.$name.'\";\n'.
     '\$phone=\"'.$phone.'\";\n'.
     '\$guest=\"'.$guest.'\";\n'.
     '\$password=\"'.$password.'\";\n'.
     '\$deaddate=\"'.$deaddate.'\";\n'.
     '\$guestInfo=\"'.$guestInfo.'\";\n'.
     '\$hostInstitution=\"'.$description[$istituto].'\";\n'.
     '?>" > '.$tmpdir.$uniqueID;
system($cmd);

// invia sms in maniera diretta
//$smscmd='echo -e "WiFi@PdCNR:\n'.$password.'" | /usr/bin/gnokii --sendsms '.$phone;
//system("sudo flock /var/lock/sendSMS ssh root@mercurio.isib.cnr.it  '".$smscmd."'
>/dev/null");

// invia sms in maniera batch
```

```
$smscmd='/usr/local/bin/invioPSW "'.$phone.'" "'.$password.'"';
system("sudo ssh root@mercurio.isib.cnr.it  '".$smscmd."' >/dev/null");
/**/
?>
Dear <b><?php echo $name;?> <?php echo $surname;?></b>,<br>
the password of your WiFi@PdCNR guest account has been sent to <b><?php echo $phone;?></b>.
<p>
In order to enable your account, you must confirm it when you receive the SMS message:
<form action="https://wifi.pd.cnr.it/wifi/accountcreation.php" method="POST">
 <?php include "hiddenFortiParams.inc";?>
 <input type="hidden" name="uniqueID" value="<?php echo $uniqueID;?>">
 <input type="submit" name="submit" value="OK, go ahead, I have got the SMS message" />
</form>
<br>
<p>If you don't receive the SMS message, check the mobile number and repeat
the  request procedure:
<form action="https://wifi.pd.cnr.it/wifi/accountrequest.php" method="post">
 <?php include "hiddenFortiParams.inc";?>
 <input type="submit" value="Request guest account">
</form>
<p>If the problem persists, please contact your host institution:<br>
<b><?php echo $description[$istituto];?></b>
<br><br>
</div>
</div>
</body>
</html>
```

accountcreation.php

questo script viene attivato per confermare un account temporaneo. Se non è trascorso più di mezz'ora dalla richiesta inserisce i dati relativi all'account nel server radius, li visualizza ad esclusione della password ed invita l'utente adf accedere alla form di autenticazione. Se il tempo è scaduto mostra un messaggio che indica all'utente di ripetere la procedura di richiesta.

```
<?php

//header('P3P: CP="IDC DSP COR CURa ADMa OUR IND PHY ONL COM STA"');
header('P3P: CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS OUR IND CNT"');
session_start();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
 <title> WiFi@PdCNR - Guest Account</title>
 <meta http-equiv="P3P" content='CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS
OUR IND CNT"'>
 <link rel="stylesheet" href="style.css" type="text/css">
</head>

<body >
<div id="centrato">
<h3><img src="/wifi/images/cnr-padova-logo-small.png" width="60" alt="Area Ricerca CNR
Padova">
 WiFi@PdCNR Guest Account</h3>
<div id="content">
<?php
$ip=$_SERVER['REMOTE_ADDR'];
include "extractFortiParams.inc";

$uniqueID=$_POST["uniqueID"];

$tmpdir="/tmp/uniqueIDs/";
if (! file_exists( $tmpdir.$uniqueID)){
  echo "FATAL ERROR<br>Contact your system administrator";
  echo "</div></div></body></html>";
  exit(0);
}
```

```
include $tmpdir.$uniqueID;
$now=time();
//verifica che non sia passato troppo tempo dalla richiesta (max 30')
if ($now-$requesttime>30*60){ //TIMEOUT
  echo "REQUEST TIMEOUT<br>Please repeat the account request procedure";
  echo "</div></div></body></html>";
  system("/bin/rm -f $tmpdir$uniqueID");
  exit(0);
}
//abilita account inserendolo nel radius server
$inserted=exec('sudo flock /var/lock/localUser /usr/local/sbin/createRadiusUser "'.
$guest.'" "'.$password.'" "'.$guestInfo.'" guests');
if ($inserted=="yes"){
  // log il nuovo utente
  $dw = date( "r", $now);
  $cmd='sudo flock /var/lock/localUser echo "'.$now.
       ' | '.$dw.
       ' | '.$guest.
       ' | '.$name.
       ' | '.$surname.
       ' | '.$phone.
       ' | '.$hostInstitution.
       ' | '.$ip.
       '" >> /var/log/radiusLocalUsers';
  system($cmd);
}

$_SESSION['WIFIPdCNRuser'] = $guest;
system("/bin/rm -f $tmpdir$uniqueID");

?>
<p>Dear <b><?php echo $name;?> <?php echo $surname;?></b>,<br>
this is your guest account for the WiFi@PdCNR service.
<p> This account is valid till midnight of <?php echo $deaddate;?>.
<p> Please, save this information for further login and keep it confidential
since you are responsible of any use or misuse of this account.
</div>
<div id="login">
<br>
<center>
<table>
<tr><td align="right">username</td><td><b><?php echo $guest;?></b></td></tr>
<tr><td align="right">password</td><td>sent to <b><?php echo $phone;?></b></td></tr>
</table>
<br>
</center>
</div>
 <div id="content">
<p>For any problem, please contact your host institution:<br>
<b><?php echo $hostInstitution;?></b>
</div>
<br>
<FORM>
 <INPUT TYPE="BUTTON" VALUE="Back to authentication form" ONCLICK="window.location.href='<?
php echo $PROTURI;?>'">
</FORM>
<br><br>
</div>
</body>
</html>
```

accountretrieve.php

Questo script viene attivato quando si desidera recuperare i dati relativi ad un account esistente. Se non vengono passati campi oppure vi sono errori nei dati immessi presenta la form di richiesta dei dati utente con eventuali messaggi di errore, altrimenti include il file retrieveform.inc che gestisce la procedura di recupero.

```
<?php

//header('P3P: CP="IDC DSP COR CURa ADMa OUR IND PHY ONL COM STA"');
header('P3P: CP="IDC DSP COR ADM DEVi TAIi PSA PSD IVAi IVDi CONi HIS OUR IND CNT"');
```

```
session_start();

$cryptinstall="./crypt/cryptographp.fct.php";
include $cryptinstall;


function bad_format($str){
  $replace_pairs=array(
    '"'=>'',
    '@'=>'',
    '#'=>'',
    '|'=>'',
    '&'=>'',
    '%'=>'',
    '$'=>'',
    '^'=>'',
    '£'=>'',
    '['=>'',
    ']'=>''
  );
  if ($str==strtr ( $str , $replace_pairs )) return FALSE;
  else return TRUE;
}

if(isset($_POST['word']))
{  if (chk_crypt($_POST['word']))
     $word_ok = "yes";
     else $word_ok = "no" ;
} else {
      $word_ok = false;
}

include "extractFortiParams.inc";


$check="yes";
if (! isset($_POST["word"])) {
  $word='';
  $check="no";
}
else $word=$_POST["word"];
if (! isset($_POST["surname"])) {
  $surname='';
  $check="no";
}
else $surname=$_POST["surname"];
if (! isset($_POST["name"])) {
  $name='';
  $check="no";
}
else $name=$_POST["name"];

if (! isset($_POST["phone"])) {
  $phone='';
  $check="no";
}
else $phone=$_POST["phone"];

$error_msg="";
$surname_err="";
$name_err="";
$phone_err="";
$word_err="";
if ($check=="yes"){
  if (empty($surname) or bad_format($surname)){
    $surname_err="<font color=\"red\">*</font>";
  }
  if (empty($name) or bad_format($name)){
    $name_err="<font color=\"red\">*</font>";
  }
  if (empty($phone) or ! ereg ("^\+[0-9]+$", $phone)){
    $phone_err="<font color=\"red\">*</font>";
  }
  if (!(empty($surname_err) and empty($name_err) and empty($phone_err))){
    $error_msg="* empty or bad format field ";
```

```
  }
  if (empty($word) or $word_ok=="no"){
    $word_err="<font color=\"red\">#</font>";
    $error_msg=$error_msg."<br># empty or wrong word";
  }

}//endif ($check=="yes")

if($word_ok!==false)
{
  if($word_ok=="yes" and $error_msg=='')
  {
    include "retrieveform.inc";
    exit(0);
  }
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
 <title> WiFi@PdCNR - Guest Account Retrieve</title>
 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
 <link rel="stylesheet" href="style.css" type="text/css">
</head>

<body >

<div id="centrato">
<h3><img src="/wifi/images/cnr-padova-logo-small.png" width="60" alt="CNR Padova logo">
 WiFi@PdCNR - Guest Account Retrieve</h3>
<div id="content">
<p>If you are a guest of an host istitution belonging to the CNR Research Area of Padova
with an active account
for the WiFi@PdCNR service, use this form to retrieve your login credentials.
<p>Mobile field must be in the ITU-T <acronym title="+&lt;country code&gt;&lt;mobile
number&gt;">E.123</acronym> format without spaces
(e.g. +393408317256)
 </div>
<div id="form">
<br>
<center>
<?php
if ($error_msg) {
  echo "<font color=\"red\">".$error_msg."</font>";
}
?>
<form action="<?=$_SERVER['PHP_SELF']?>" method="post">
 <?php include "hiddenFortiParams.inc";?>
 <table cellpadding="0" cellspacing="0">
 </td></tr>
 <tr><td align="right">name:</td><td><input type="text" name="name" size="27" value="<?
=htmlspecialchars($name)?>">
 <?php
  if ($name_err) echo $name_err;
 ?>
 </td></tr>
 <tr><td align="right">surname:</td><td><input type="text" name="surname" size="27"
value="<?=htmlspecialchars($surname)?>">
 <?php
  if ($surname_err) echo $surname_err;
 ?>
 </td></tr>
 <tr><td align="right">mobile:</td><td><input type="text" name="phone" size="27" value="<?
=htmlspecialchars($phone)?>">
 <?php
  if ($phone_err) echo $phone_err;
 ?>
 </td></tr>
 <tr><td colspan="2" align="center">
 <?php dsp_crypt(0,1); ?>
 </td></tr>
 <tr><td colspan="2" align="center">type the word above: <input type="text" name="word">
 <?php
  if ($word_err) echo $word_err;
```

```
 ?>
   </td></tr>
 <tr><td colspan="2" align="center"><input type="submit" value="Retrieve login
credentials"></td></tr>
 </table><br />
</form>
</center>
</div>
<br>
</div>
</body>
</html>
```

retrieveform.inc

script che partendo dai dati identificativi di un utente ne visualizza le credenziali di accesso eccetto la password che viene inviata via SMS se esiste un account associato a tale utente, altrimenti visualizza un messaggio di errore.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
 <title> WiFi@PdCNR - Guest Credentials</title>
 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
 <link rel="stylesheet" href="style.css" type="text/css">
</head>

<body >

<div id="centrato">
<h3>
<img src="/wifi/images/cnr-padova-logo-small.png" width="60" alt="CNR Padova logo">
 WiFi@PdCNR -Guest Credentials
</h3>
<div id="content">
<p>

<?php

//controlla esistenza account nel file users.guests del radius server
$name=ucwords(strtolower($name));
$surname=ucwords(strtolower($surname));
$guestDescription=$name.",".$surname.",".$phone;
$guestdata=exec('sudo flock /var/lock/localUser /usr/local/sbin/checkRadiusGuest "'.
$guestDescription.'"');
if ($guestdata=="nouser") { //esiste account
  echo "An account for <b>$name $surname </b> phone num. <b>$phone</b> does not exist.";
  echo "<br>Either you have mispelled some data<br>";
 echo '<form action="https://wifi.pd.cnr.it/wifi/accountretrieve.php" method="POST">';
  include "hiddenFortiParams.inc";
  echo '<input type="submit" name="submit" value="Retrieve login credentials" />';
  echo "</form>";
  echo "<br>or the account has expired<br><br>";
  echo '<form action="https://wifi.pd.cnr.it/wifi/accountrequest.php" method="POST">';
  include "hiddenFortiParams.inc";
  echo '<input type="submit" name="submit" value="Request guest account" />';
  echo "</form>";
  echo "<br>";
  echo "</div></div></body></html>";
  exit(0);
}

$data=explode("|",$guestdata);

// invia sms in maniera batch
$smscmd='/usr/local/bin/invioPSW "'.$phone.'" "'.$data[1].'"';
system("sudo ssh root@mercurio.isib.cnr.it  '".$smscmd."' >/dev/null");

?>
```

```
Login credentials associated with <b><?php echo $name;?> <?php echo $surname;?> <?php echo
$phone;?></b><br><br>
Username: <b><?php echo $data[0];?></b><br>
Password: sent to <b><?php echo $phone;?></b>
<p>If you don't receive the SMS message, please contact your host institution.
<br>
<FORM>
 <INPUT TYPE="BUTTON" VALUE="Back to authentication form" ONCLICK="window.location.href='<?
php echo $PROTURI;?>'">
</FORM>
<br><br>
</div>
</div>
</body>
</html>
```

Autenticazione IDEM

Questa modalità di autenticazione permette ad utenti afferenti ad istituzioni che appartengono alla federazione IDEM GARR di accedere alla rete wireless.

La procedura di autenticazione viene attivata mediante un accesso web alla directory /var/www/html.ssl/IDEMauth dove risiede lo script php dedicato a questa procedura. Questo script viene eseguito solamente se l'autenticazione IDEM va a buon fine e l'indirizzo IP del nodo da cui proviene la richiesta appartiene alla rete wireless guest.

/etc/httpd/conf.d/shib.conf
file di configurazione per il server apache utilizzato per definire la directory web /IDEMauth protetta in accesso dal modulo Shibboleth

```
# https://spaces.internet2.edu/display/SHIB2/NativeSPApacheConfig

# RPM installations on platforms with a conf.d directory will
# result in this file being copied into that directory for you
# and preserved across upgrades.

# For non-RPM installs, you should copy the relevant contents of
# this file to a configuration location you control.

#
# Load the Shibboleth module.
#
LoadModule mod_shib /usr/lib64/shibboleth/mod_shib_22.so

#
# Used for example logo and style sheet in error templates.
#
<IfModule mod_alias.c>
  <Location /shibboleth-sp>
    Allow from all
  </Location>
  Alias /shibboleth-sp/main.css /usr/share/doc/shibboleth-2.4.2/main.css
  Alias /shibboleth-sp/logo.jpg /usr/share/doc/shibboleth-2.4.2/logo.jpg
</IfModule>

#
# Configure the module for content.
#
# You MUST enable AuthType shibboleth for the module to process
# any requests, and there MUST be a require command as well. To
# enable Shibboleth but not specify any session/access requirements
# use "require shibboleth".
#
<Location /IDEMauth>
  AuthType shibboleth
```

```
  ShibRequestSetting requireSession 1
  require valid-user
</Location>
```

index.php

Script PHP per l'autenticazione IDEM installato nella directory /var/www/html.ssl/IDEMauth. L'esecuzione di questo script è subordinata all'autenticazione IDEM via Shibboleth. Una volta attivato lo script ottenuti i parametri associati all'utente IDEM e verificato che l'indirizzo IP del nodo da cui proviene la richiesta appartenga alla rete guest, genera se non esiste un nuovo account radius associato a tale utente IDEM e invia in maniera automatica con un javascript le credenziali al firewall per l'autenticazuione effettiva.

```
<?php
function make_password($length,$strength=0) {
  $vowels = 'aeiouy';
  $consonants = 'bdghjlmnpqrstvwxz';
  if ($strength & 1) {
    $consonants .= 'BDGHJLMNPQRSTVWXZ';
  }
  if ($strength & 2) {
    $vowels .= "AEIUYO";
  }
  if ($strength & 4) {
    $numbers = '0123456789';
  }
  if ($strength & 8) {
    $specials = '@#$%^';
  }
  $alt = time() % 2;
  srand(time());
  $alt=rand() % 3; // un numero tra 0 e 2
  // inizia con una lettera
   if ($alt % 2 == 1) {
      $password = $consonants[(rand() % strlen($consonants))];
    } else {
       $password = $vowels[(rand() % strlen($vowels))];
    }
      $alt = ($alt + 1)%3;
  for ($i = 1; $i < $length; $i++) {
    switch ($alt) {
      case 0:
      $password .= $consonants[(rand() % strlen($consonants))];
        break;
      case 1:
       $password .= $vowels[(rand() % strlen($vowels))];
         break;
      case 2:
        $password .= $numbers[(rand() % strlen($numbers))];
        break;
       }
      $alt = ($alt + 1)%3;
  }
  return $password;
}

$ip=$_SERVER['REMOTE_ADDR'];$ePTI=$_SERVER['persistent-id'];$ePSA=$_SERVER['affiliation'];
$comp=explode("!",$ePTI);
$hex=bin2hex($comp[2]);
?>
<html>
<head>
 <title>CNR Padova - WiFi IDEM Authentication</title>
 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
</head>

<body >
<?php
  $AUTH_POST_URL=$_GET["AUTH_POST_URL"];
  $REDIRID=$_GET["REDIRID"];
  $PROTURI=$_GET["PROTURI"];
  $MAGICID=$_GET["MAGICID"];
  $MAGICVAL=$_GET["MAGICVAL"];
```

```
  $QUESTION=$_GET["QUESTION"];
  $USERNAMEID=$_GET["USERNAMEID"];
  $PASSWORDID=$_GET["PASSWORDID"];
  if (strstr($ip,"150.178.99.")){
    $idem_user=$comp[2].".idem";
    $password=exec('sudo flock /var/lock/localUser /usr/local/sbin/checkRadiusUser "'.
$idem_user.'" idem');
    if ($password=="nouser")
      $password=make_password(17,7);
    $comment=$ePTI.$ePSA;
    $inserted=exec('sudo flock /var/lock/localUser /usr/local/sbin/createRadiusUser "'.
$idem_user.
                   '" "'.$password.'" "'.$comment.'" idem');
    if ($inserted=="yes"){
      // log il nuovo utente
      $now=time();
      $dw = date( "r", $now);
      $cmd='sudo flock /var/lock/localUser echo "'.$now.
           ' | '.$dw.
           ' | '.$idem_user.
           ' | '.$ePTI.
           ' | '.$ePSA.
           ' | '.$ip.
           '" >> /var/log/radiusLocalUsers';
      system($cmd);
    }
  }else {
    echo "Autenticazione utenti IDEM <p> Servizio disponibile solamente per i nodi della
VLAN ospiti<br></body></html>";
    return;
  }
?>
<br>
<br>
Redirecting to authentication service ...
<br>

<form action=<?php echo $AUTH_POST_URL;?> method="post">
 <input type="hidden" name="<?php echo $REDIRID;?>" value="<?php echo $PROTURI;?>">
 <input type="hidden" name="<?php echo $MAGICID;?>" value="<?php echo $MAGICVAL;?>">
 <input name=<?php echo $USERNAMEID;?> id="un" type="hidden" value=<?php echo '"'.
$idem_user.'"';?>/>
 <input name=<?php echo $PASSWORDID;?> id="pd" type="hidden" value=<?php echo '"'.
$password.'"';?>/>
</form>

<script type="text/javascript">
        //<![CDATA[

             function spedisciForm()
             {
                   document.forms[0].submit();
             }

             spedisciForm();

        //]]>
        </script>

</body>
</html>
```

Gestione token per richiesta credenziali

Per poter richiedere un account temporaneo un utente senza credenziali di accesso deve indicare nel modulo di richiesta un token valido. Un token è una stringa di sei lettere seguita da un carattere tra -.%_/e tre cifre. Senza il token non è possibile ottenere le credenziali temporanee

I token validi sono memorizzati nel file /usr/local/etc/raddb/tokens.lst, uno per linea con il seguente formato:

<token>|<deadline time>|<validity period>|<deadline date>

dove

<token>	è la sequenza di caratteri che individua il token
<deadline time>	è la scadenza temporale del token espressa in unix time
<validity period>	è il numero di giorni di validità oppure * (asterisco) per il token creato in maniera automatica giornalmente
<deadline date>	è la scadenza temporale espressa come data

I seguenti script, che risiedono nella directory /usr/local/sbin, permettono la gestione dei token e possono essere utilizzati sia su linea di comando che attraverso un interfaccia web.

createToken.php
script per la creazione di un token. Ha due modalità di funzionamento: con parametro numerico crea un token con una durata in giorni pari al valore del parametro, senza parametro crea un token di durata giornaliera. Ritorna la codifica del token che dev essere inserita nel file contente i token attivi. Ad esempio, per creare un token della durata di 7 giorni con interfaccia CLI si deve da root dare il seguente comando:
usr/bin/php -q /usr/local/sbin/createToken.php 7 >>/usr/local/etc/raddb/tokens.lst

```
<?php
/*
 /usr/local/sbin/createToken.php

*/
function make_token(){
  $vowels = 'AEIOUY';
  $consonants = 'BCDGHJKLMNPQRSTVWXZ';
  $numbers = '123456789';
  $symbols ="-.%_/";
  srand(time());
  $tk="";
  for ($i = 1; $i <= 3; $i++) {
    $tk .=$consonants[(rand() % strlen($consonants))];
    $tk .= $vowels[(rand() % strlen($vowels))];
  }
    $tk .= $symbols[(rand() % strlen($symbols))];
  for ($i = 1; $i <= 3; $i++) {
        $tk .= $numbers[(rand() % strlen($numbers))];
      }
  return $tk;

}

function compute_deadline($days){
  $dw = date( "j M Y",time()+3600*24*($days-1))." 23:59:00"; // 23.59 del giorno di
scadenza
  return strtotime($dw);
}

if ($argc==1){ // token casuale giornaliero
  $token=make_token();
  $limit=compute_deadline(1); // durata 1 giorno
  echo "$token|$limit|*|".date("r",$limit)."\n";
}
if ($argc==2 ){ // token casuale assegnato con numero di giorni
```

```
  $num = (int)$argv[1];
  if (is_int($num)and $num>0){
    $token=make_token();
    $limit=compute_deadline($num); // durata $argv[1] giorni
    echo "$token|$limit|$num|".date("r",$limit)."\n";
  }
}
// do nothing
?>
```

clearTokens.php

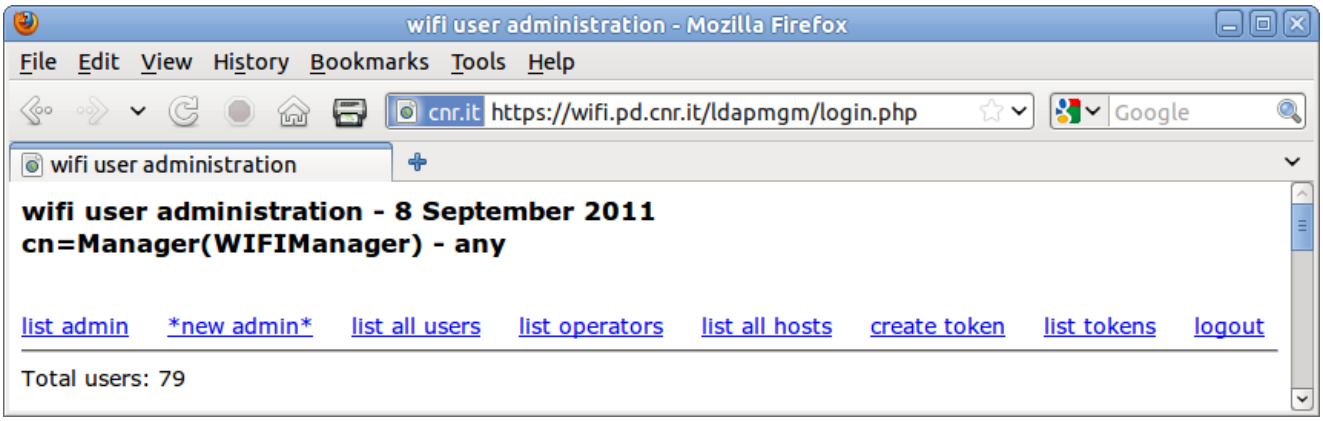
script che elimina dal file il cui nome è passato coma parametro, tutti i token scaduti. Se nel file non è presente un token giornaliero, indicato dall'asterisco nel campo relativo alla durata in giorni, ne crea uno. Viene attivato giornalmente con la seguente definizione cron:

```
4 0 * * * /usr/bin/php -q /usr/local/sbin/clearTokens.php /usr/local/etc/raddb/tokens.lst
```

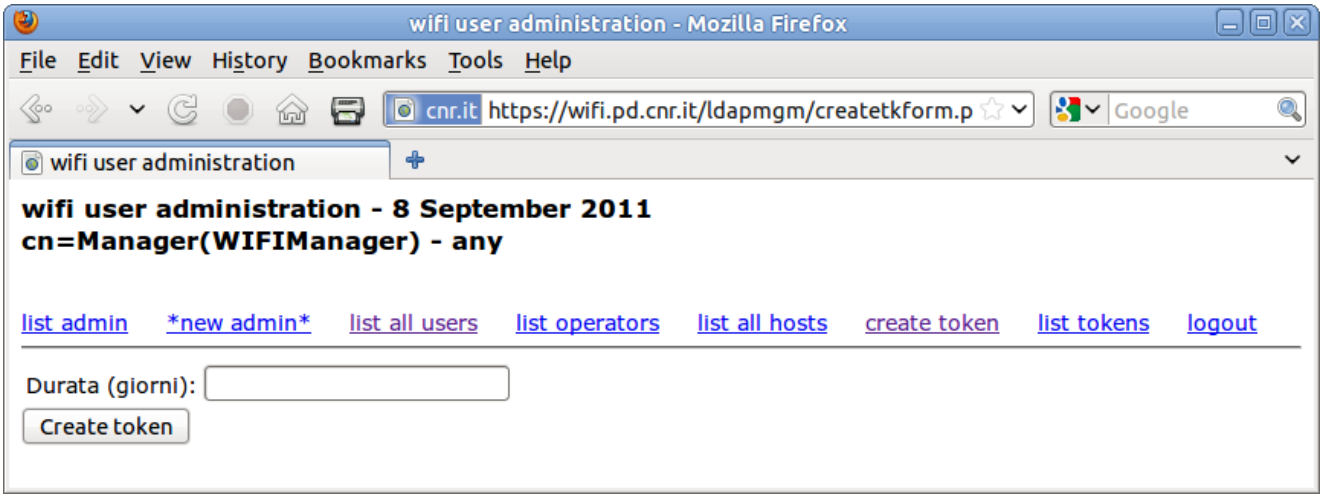
```
<?php
/*
 /usr/local/sbin/clearTokens.php
 4 0 * * * /usr/bin/php -q /usr/local/sbin/clearTokens.php /usr/local/etc/raddb/tokens.lst
*/

if ($argc==2){ // $argv[1] file contenente i tokens
  $now=time();
  $tmp = "/tmp/tokens.tmp";
  $th = fopen($tmp, 'w') or die("can't open file");
  if ($fh = fopen($argv[1], "r")) {
    $notdaily=true;
    while (!feof($fh)) {
      $token_line = fgets($fh);
      $token_data=explode("|",$token_line);
      if (isset($token_data[1]))
        if ($now<$token_data[1]){
          fwrite($th, $token_line);
          if ($token_data[2]=="*") $notdaily=false;
        }
    }
    fclose($fh);
    fclose($th);
    exec("mv $tmp $argv[1]");
    if ($notdaily) // crea token giornaliero
      exec("/usr/bin/php -q /var/www/html.ssl/wifi/createToken.php >>$argv[1]");
  }
}
// do nothing
?>
```

Per eliminare un token valido prima della sua scadenza si deve cancellare con un editor la corrisponente linea nel file /usr/local/etc/raddb/tokens.lst.

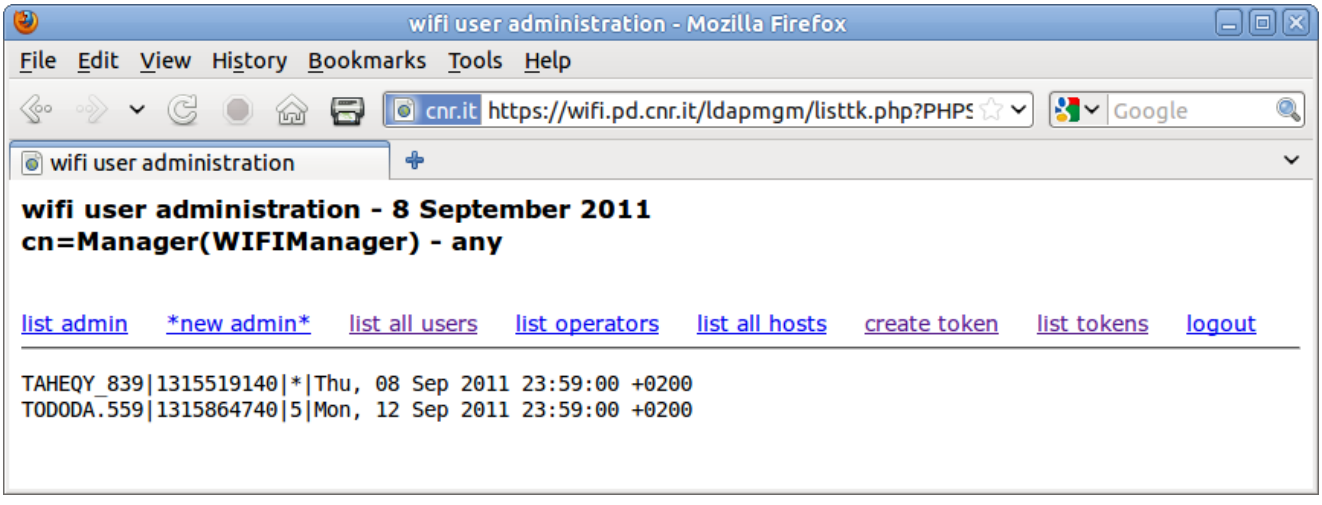
È possibile gestire via web la creazione e visualizzazione dei token attraverso l'interfaccia di amministrazione degli utenti wifi raggiungibile dal URL https://wifi.pd.cnr.it/ldapmgm. Entrando come manager si ottiene una pagina in cui sono presenti i due link, create_token e list_tokens per la gestione dei token:



Con il link create_token si accede al modulo che permette di inserire la durata in giorni del token:



Il link list_tokens permette di visualizzare tutti i token validi:



Log file

Il servizio utilizza alcuni file di log per memorizzare eventi significativi che avvengono durante il funzionamento:

/var/log/radiusd.log
contiene i log generati dal server radius con le informazioni dell'utente autenticato:

```
Sep 14 14:13:27 ldap1 radiusd[3764]: Login OK: [pb@isib] (from client Management Network port 5 cli 150.178.99.91)
Sep 14 14:25:58 ldap1 radiusd[3764]: Login OK: [NRP+6NknRoq7Svdf77DIpx2Y/pc=.idem] (from client Management Network
port 5 cli 150.178.99.91)
```

/var/log/radiusLocalUsers
contiene i record generati alla creazione di un nuovo utente sia temporaneo che IDEM

```
1316003158 | Wed, 14 Sep 2011 14:25:58 +0200 | NRP+6NknRoq7Svdf77DIpx2Y/pc=.idem | https://idp.cnr.it/idp/shibboleth!
https://wifi.pd.cnr.it/IDEMauth!NRP
+6NknRoq7Svdf77DIpx2Y/pc= | staff@cnr.it | 150.178.99.91
```

BIBLIOGRAFIA

P.Bison, C.Cavaggion, D.Galiazzo, *Struttura ed uso della rete WIFI nell'area della Ricerca di Padova*, rapporto tecnico 07/09, ISIB-CNR, Padova, Dicembre 2009

P.Bison, C.Cavaggion, D.Galiazzo, *Realizzazione di una rete WIFI con autenticazione 802.1x nell'area della Ricerca di Padova*, rapporto tecnico 01/10, ISIB-CNR, Padova, Aprile 2010