UNIVERSITÀ DEGLI STUDI DI CAMERINO

SCUOLA DI SCIENZE E TECNOLOGIE

Corso di Laurea in Informatica

“OpenWrt per un Access Point per il WiFi di UNICAM”

Elaborato Finale

Candidato:

Filippo Nardi

Relatore:

Prof. **Fausto Marcantoni**

Correlatore:

Prof. **Francesco De Angelis**

ANNO ACCADEMICO 2013/2014

Sommario

INTRODUZIONE

OBIETTIVI DELLA TESI

Questa tesi analizza gli aspetti affrontati per la progettazione di un sistema che consenta di integrare OpenWrt per gli Access Point per il WiFi Unicam.

Già da un po' di tempo, infatti, all'interno dell'Università è operativo il servizio WiFi. Per connettersi con il servizio wireless che Unicam mette a disposizione degli utenti bisogna collegarsi alla rete UnicamEasyWiFi l'accesso avverrà tramite username e password (nello specifico):

- gli utenti del dominio amministrazione (docenti, ricercatori, personale TA, dottorandi con e-mail ...@unicam.it) dovranno scrivere username (esempio: giuseppe.bianchi) e password
- gli studenti e i dottorandi con e-mail ...@studenti.unicam.it dovranno scrivere: studenti\ prima del proprio username (esempio: studenti\mario.rossi) e la password Unicam in loro possesso
- per gli ospiti è necessario un account ad hoc per consentire il collegamento alla rete UnicamEasyWiFi.

Nella realizzazione di sistemi informativi distribuiti sicuri sorge spesso l'esigenza di verificare che il proprio interlocutore sia effettivamente chi dichiara di essere. Sono tre i principi fondamentali su cui si basa parte della sicurezza informatica, in particolar modo per quando riguarda l'accesso ai dati: l'identificazione (identity), l'autenticazione (authentication) e l'autorizzazione (authorization).

Identificazione: risponde alla domanda "Chi sei tu?". L'utente che vuole accedere ad un sistema deve "rispondere" a questa domanda. Nella maggior parte dei casi questa risposta è il login, username, user-id o la chiave pubblica. Questa informazione può anche essere pubblica, non coperta cioè da nessun vincolo di segretezza.

Autenticazione: è il processo che verifica l'identità, ovvero risponde alla domanda:

"l'utente è chi dice di essere?" una volta stabilità l'identità di una persona, il sistema deve essere sicuro che l'utente sia quello che dice di essere. Per questo motivo, il sistema chiede "Come puoi dimostrare la tua identità?". Questa informazione deve essere assolutamente tenuta segreta e conservata con la massima accuratezza.

Autorizzazione: una volta che il sistema ha acconsentito all’accesso, ora si tratta di stabilire “cosa puoi fare?”, ossia a quali risorse, a quali dati puoi accedere? Il tutto viene garantito con un controllo agli accessi, in base alle autorizzazioni precedentemente date al profilo dell’utente. Il sistema è pertanto in grado di stabilire quali operazioni consentire e quali vietare all’utente.

Una **politica di controllo di accesso** è un insieme di regole che definiscono la protezione di risorse, generalmente stabilendo la possibilità o meno che persone o altre entità possano accedere a tali risorse. Una delle forme più usate di politiche di controllo di accesso è l'access control list.

Quando una richiesta inizia ad essere processata da un server, deve essere associata ad una grande varietà di fattori di sicurezza. Il server usa questi fattori per determinare, quando e come processare queste richieste.

Tali fattori sono chiamati fattori di controllo di accesso (ACFs, Access Control Factors). Essi possono includere, ad esempio, gli indirizzi IP sorgente e destinazione, una stringa cifrata, il tipo di operazione inizialmente richiesta, l'ora del giorno.

Alcuni fattori possono essere specificati dalla stessa richiesta, altri possono essere associati dalla connessione all’atto della trasmissione della richiesta, altri possono essere fattori riferiti all'ambiente.

Le credenziali di autenticazione sono la prova che una parte fornisce l'altra, asseriscono cioè l’identità di una parte (ad esempio un utente), che sta cercando di stabilire una comunicazione con un’altra parte (tipicamente un server).

L’autenticazione è il processo di generazione, trasmissione e verifica di queste credenziali per stabilire l'identità delle parti.

Un'identità di autenticazione (Authentication Identity) è il nome presentato in una credenziale.

Ci sono diverse forme di credenziali di autenticazione e la forma usata dipende dai particolari meccanismi di autenticazione negoziati dalle parti in questione. E' importante ricordare che un meccanismo di autenticazione può imporre la forma dell'identità di autenticazione da usare.

I metodi tramite i quali si può autenticare una persona, sono divisi in tre classi, in base a ciò che:

- **è**: per esempio impronte digitali, impronta vocale, calligrafia o altri identificatori biometrici;
- **ha**: tesserino identificativo, certificato;
- **conosce**: password, parola chiave o numero d'identificazione personale (PIN).

Per l'autenticazione in rete è normalmente usata la terza classe, che non richiede né l'utilizzo di hardware speciali né la presenza dell'utente.

Per ragioni di sicurezza, la password scelta dall'utente, o a lui assegnata, deve soddisfare certi requisiti ed essere:

- non ovvia e di lunghezza minima data;
- non comunicata in chiaro;
- cambiata con una certa frequenza.

L'autenticazione tramite password è oggi il sistema più utilizzato per l'accesso ai vari servizi della rete. Questo modo di procedere causa però alcuni disagi, sia agli utenti che usufruiscono del servizio stesso e sia a chi li fornisce. Infatti, gli utenti si trovano spesso a dover gestire molti codici di accesso; Ad esempio uno per accedere alle riviste elettroniche di un certo editore, un altro per accedere alla posta elettronica, e un altro ancora per accedere ai corsi online. Considerando che la password deve essere cambiata spesso, ne discende che per l'utente può diventare gravoso ricordare una coppia username/password. Ancora di più se poi deve ricordare i loro cambiamenti nel tempo per ciascun servizio che usa. Anche i sistemi che forniscono un'identità singola per accedere alle risorse locali, richiedono comunque un login distinto nel caso che si voglia accedere a risorse remote. Perciò, recentemente si tende a sviluppare dei sistemi di autenticazione che possano essere riutilizzabili per varie funzioni, o servizi. In alcuni casi, tali sistemi si occupano di gestire un'unica coppia username/password associata a un utente, che deve comunque fornirla per ciascun servizio che intende utilizzare. In altri casi, almeno per quanto riguarda servizi diversi accessibili all'interno di un'unica università/azienda, esistono dei veri e propri sistemi di Single Sign On, in cui un utente si autentica una volta soltanto e accede poi liberamente a tutti i servizi per cui possiede l'autorizzazione. Di tali sistemi ne esistono ormai vari e la scelta non è sempre facile e spesso dipende da molte variabili.

TECNOLOGIA ADOTTATA

Nella realizzazione del lavoro sono stati utilizzati tecnologie hardware come (Access Point, Pc), che software (IAS, Apache, ecc.), firmware open source (OpenWrt, DD-Wrt) e macchine virtuali(Windows Server 2003).

Nella prima parte della tesi verranno introdotti i concetti teorici a cui si è fatto riferimento per poter realizzare le varie configurazioni, mentre nella seconda parte del teso verranno trattati gli strumenti, le guide, i software, i firmware e i comandi utilizzati per completare le configurazioni.

Di seguito sarà indicato come ogni strumento è stato configurato in modo tale da creare due possibili tipologie di autentificazione per la rete wifi in Unicam.:

- identificazione tramite Server Radius/Ldap
- identificazione tramite Server Shibboleth

MACCHINA VIRTUALE WINDOWS SERVER 2003

L'installazione di Windows Server 2003 offre ad un computer le capacità di un sistema server, pur mantenendo la funzionalità del suo sistema operativo padre.

Virtualizzazione è un termine che può essere utilizzato non solo per l'installazione di sistemi operativi in uno stesso computer; altri tipi di virtualizzazioni sono le virtualizzazioni software che permettono, ad esempio, di installare reti private virtuali, cd virtuali, stampanti virtuali e tanto altro.

La tecnologia di virtualizzazione viene utilizzata per espandere la capacità delle risorse hardware senza dover spostare software e dati da un computer o server ad un altro, è utilizzata anche per abbassare i costi dell'hardware così da poter installare più server virtuali su un'unica macchina.

Nella virtualizzazione di sistemi operativi, c'è l'Host, il computer principale dove viene installato il programma di gestione e il guest, il sistema operativo che viene eseguito all'interno del programma di virtualizzazione. VMWare Server è un software di virtualizzazione gratuito di VMWare che può ancora essere scaricato e utilizzato gratuitamente. Supporta quasi tutti i sistemi operativi come host, ma non può virtualizzare un sistema 64-bit su un host a 32-bit. VMWare Server ha il supporto per dispositivi USB e interfacce di rete.

IAS

Il servizio IAS costituisce l'implementazione Microsoft di un server e un proxy RADIUS (Remote Authentication Dial-in User Service), che consente la gestione centralizzata dell'autenticazione, dell'autorizzazione, nonché degli account utente. IAS può essere utilizzato per autenticare gli utenti in database contenuti in controller di dominio basati su Windows Server 2003, Windows NT 4.0 o Windows 2000. IAS supporta inoltre una vasta gamma di server NAS (Network Access Server) e RRAS (Routing and Remote Access). Il meccanismo impiegato da RADIUS per nascondere impiega il segreto condiviso RADIUS, l'autenticatore di richieste e l'algoritmo di hash MD5 per crittografare User-Password e altri attributi, ad esempio Tunnel-Password e MS-CHAP-MPPE-Keys. RFC 2865 rileva la potenziale necessità di valutare i pericoli e determinare se è opportuno adottare ulteriori protezioni.

APACHE

Apache HTTP Server, o più comunemente Apache è il nome della piattaforma server Web sviluppata dalla pahe Software Foundation. È la piattaforma server Web modulare più diffusa, in grado di operare su una grande varietà di sistemi operativi, tra cui UNIX/Linux, Microsoft e OpenVMS, Apache è un software che realizza le funzioni di trasporto delle informazioni, di internetwork e di collegamento, ha il vantaggio di offrire anche funzioni di controllo per la sicurezza come quelli che compie il proxy. Il progetto Apache nacque nel 1995. A quel tempo, il server Web più diffuso era il daemon HTTP pubblico sviluppato da Rob McCool al NCSA (National Center for Supercomputing Application), Università dell'Illinois. Dal momento però che a partire dal 1994 lo sviluppo di questo server si era fermato (anche perché il suo autore aveva lasciato l'NCSA) un gruppo di webmaster aveva iniziato a sviluppare patch in maniera autonoma. Fu creata una mailing list e verso la fine di febbraio del 1995 si costituì il primo gruppo di lavoro dell'Apache Group. Otto persone (Brian Behlendorf, Roy T. Fielding, Rob Hartill, David Robinson, Cliff Skolnick, Randy Terbush, Robert S. Thau, Andrew Wilson) presero come punto di partenza la versione 1.3 del demone HTTP NCSA e aggiunsero una serie di patch e di correzioni.

La prima release pubblica di Apache, la 0.6.2, venne rilasciata nell'aprile del 1995. Il nome Apache, secondo la leggenda, nasce dal fatto che inizialmente il server era semplicemente una raccolta di patch da applicare al server NCSA. Dal nome amichevole "a patchy server" sarebbe quindi nato Apache.

In realtà, il nome fu scelto in onore della tribù di nativi americani Apache, come riportano le domande frequenti sul sito ufficiale del progetto. Una nuova architettura server fu integrata poco dopo nella versione 0.8.8 a cui fu dato il nome in codice di Shambala. La versione 1.0 fu pubblicata il 1º dicembre 1995. Nel giro di un anno, la sua diffusione aveva già superato quella del server NCSA da cui era derivato. La versione 2.0 di Apache venne rilasciata durante la conferenza ApacheCon, tenutasi nel marzo 2000 a Orlando, in Florida. Il grande successo di diffusione di questo software è l'indicatore più chiaro della qualità e dell'affidabilità di questo prodotto:. Operativamente, è composto da un demone, in ambiente UNIX, o da un servizio, in ambiente Microsoft, che sulla base delle impostazioni contenute nel file di configurazione httpd.conf permette l'accesso a uno o più siti, gestendo varie caratteristiche di sicurezza e potendo ospitare diverse estensioni per pagine attive (o dinamiche), come PHP o Jakarta/Tomcat. Il Web Server Apache presenta un'architettura modulare, quindi ad ogni richiesta del client vengono svolte funzioni specifiche da ogni modulo di cui è composto, come unità indipendenti. Ciascun modulo si occupa di una funzionalità, ed il controllo è gestito dal core.
I moduli:

- **Core**: programma principale composto da un ciclo sequenziale di chiamate ai moduli.
- **Translation**: traduce la richiesta del client
- **Access Control**: controlla eventuali richieste dannose
- **MIME Type**: verifica il tipo di contenuto
- **Response**: invia la risposta al client e attiva eventuali procedure
- **Logging**: tiene traccia di tutto ciò che è stato fatto

Il core suddivide la richiesta ai vari moduli in modo sequenziale, usando i parametri di uscita di un modulo come parametri di accesso per l'altro, creando così l'illusione di una comunicazione orizzontale fra i moduli (Pipeline software). Sopra il ciclo del core c'è un ulteriore ciclo di polling svolto da un demone che interroga continuamente le linee logiche da cui possono pervenire messaggi di richiesta.

Gli amministratori del server possono usare il file httpd.conf, che è situato nella subdirectory conf della directory indicata durante la installazione. Questo file mette a disposizione tutta la libertà offerta dal server, quindi aggiungere moduli, estensioni, nuovi mime-type ed altro ancora.

NETBEANS

NetBeans è un progetto open-source mirato a fornire un solido prodotto per sviluppare software (come NetBeans IDE e NetBeans Platform) che guida le necessità di sviluppatori, utenti e commerciali che si affidano a NetBeans per i loro prodotti. Il progetto NetBeans è anche una fervida comunità dove chiunque da qualsiasi paese può pensare, porre domande, esprimere pareri, contribuire in tanti modi e in ultimo condividere il successo dei nostri prodotti. Nel mese di giugno 2000 NetBeans è diventato open source grazie a Sun Microsystems che rimane lo sponsor del progetto.

OPENWRT/DD-WRT

Per realizzare la configurazione di cui andremo ad esporre il procedimento più avanti è stato utilizzato un firmware open source basato su kernel linux denominato OpenWrt. Come suggerisce la pagina di Wikipedia:" *OpenWrt è una distribuzione Linux specifica per dispositivi embedded come router CPE, Smartphone, pocket computer* ", in grado di far "*sprigionare*" la potenza nascosta di un qualsiasi router, anche quelli più datati. Tutto questo è reso possibile dal fatto stesso che il firmware non ci viene dato con delle funzioni prestabilite come quelli delle case produttrici nei loro prodotti, ma è composto da un filesystem completamente riscrivibile con una gestione dei pacchetti aggiuntivi molta ampia. Con questa filosofia il progetto OpenWrt libera dai blocchi imposti dai firmware vendo distribuiti dalle case di produzione, ampliando le possibilità di utilizzo, sia nel privato che nel campo aziendale, di macchine ormai datate per i tempi attuali.

ACCESS POINT

Un Access Point (AP) è un dispositivo elettronico di telecomunicazioni che, collegato ad una rete cablata, o anche, per esempio, ad un router, permette all'utente mobile di accedervi in modaltà wireless direttamente tramite il suo terminale, se dotato di scheda wireless.

Se esso viene collegato fisicamente ad una rete cablata (oppure via radio ad un altro access point)[senza fonte], può ricevere ed inviare un segnale radio all'utente grazie ad antenne e apparati di ricetrasmissione, permettendo così la connessione sotto forma di accesso radio.

La funzionalità di Access Point è anche normalmente integrata nei più moderni router.

È possibile collegare più access point alla stessa rete cablata e/o tra di loro per creare in questo modo una rete più grande che permetta l'handover tra terminali e rete wireless (WLAN).

Se collegato ad una rete cablata, esso fa da interfaccia tra la parte wireless di accesso radio da parte degli utenti e la parte cablata di trasporto implementamdo un cambiamento di protocollo per il trasferimento dell'informazione tra le due sezioni di rete; se invece trasmette informazione tramite collegamento wireless agli altri access point (Wireless Distribution System) funziona come un semplice bridge, pur con una perdita di efficienza spettrale nel sistema. L'AP comunica in broadcast alle stazioni riceventi nel proprio raggio di copertura l'SSID della rete o delle reti locali wireless che sta servendo. Le celle di copertura degli AP sono spesso parzialmente sovrapposte per evitare buchi di copertura del segnale mentre la parte cablata è generalmente una rete Ethernet che può essere a bus condiviso oppure commutata ovvero in switch mode. Un Access Point IEEE 802.11 può normalmente comunicare con circa 30 client nel raggio di circa 100 m, anche se il range di copertura può scendere sensibilmente in presenza di ostacoli fisici nella linea di vista. La banda di comunicazione può variare molto in funzione di diverse variabili come il posizionamento interno o esterno, l'altezza dal suolo, la presenza di ostacoli vicini, il tipo di antenna, le attuali condizioni meteo, la frequenza radio su cui opera e la potenza di output del dispositivo. La banda dell'Access Point può essere estesa attraverso l'utilizzo di ripetitori e riflettori del segnale, i quali possono far rimbalzare e amplificare i segnali radio che altrimenti non potrebbero essere ricevuti ordinariamente. Sono stati effettuati alcuni esperimenti in questo senso, al fine di permettere l'estensione delle portate delle reti wireless a distanze di diversi chilometri. I protocolli Wi-Fi consentono anche di adattare la velocità di trasmissione nella parte di accesso wireless in funzione della distanza dalla stazione ricetrasmittente. L'uso tipico di un Access Point è quello di collegarlo ad una LAN e consentire così ad utenti muniti di dispositivi wireless di usufruire dei servizi di rete LAN con in aggiunta il vantaggio della mobilità. In questa configurazione l’Access Point agisce da gateway per i client wireless. Un altro utilizzo è quello di collegare due LAN distinte; ad esempio, se due uffici di una azienda sono separati da una strada pubblica, può risultare economicamente più vantaggioso sfruttare l'etere attraverso due Access Point (uno per ogni sede) anziché portare dei cavi sotto terra (con possibili problematiche di permessi comunali).

In questo caso gli Access Point verranno configurati in modalità bridge (ponte). Reti senza fili a basso costo sono divenute rapidamente popolari verso la fine degli anni '90 e primi anni 2000 poiché permettono di minimizzare il cablaggio dei cavi di collegamento di rete usati nelle reti ethernet tradizionali, riducendo drasticamente i costi d'impianto oltre ad offrire una certa mobilità nell'accesso.

Queste reti si sono diffuse, grazie allo standard Wi-Fi, nelle reti domestiche senza fili e nelle reti pubbliche di accesso a Internet.

Le reti senza fili permettono inoltre maggiore mobilità agli utenti, liberandoli dal vincolo di impiego del terminale nei pressi di una presa di rete (es. RJ-45): nell'industria e nel commercio, ad esempio, terminali portatili o palmari wireless consentono agli utenti maggiore operatività, quali la possibilità di compiere operazioni di registrazioni o rettifica di dati, carichi e scarichi di merce, direttamente ed in tempo reale negli archivi degli elaboratori centrali.

Esistono diversi standard wireless per la trasmissione:

- 802.11a con trasmissione max a 54 Mb/s a 5 GHz
- 802.11b con trasmissione max a 11 Mb/s a 2,4 GHz
- 802.11g con trasmissione max a 54 Mb/s a 2,4 GHz
- 802.11n con trasmissione max a 300 Mb/s a 2,4 GHz e 5 GHz

Gli standard più diffusi sono B/G/N e va anche sottolineato che un Access Point di nuova generazione trasmette in 3 tipologie di segnali e non 4, poiché lo standard A è ormai obsoleto e troppo lento. Inoltre per ricevere da un Access Point un segnale di nuova generazione ovvero di standard N è necessario avere la scheda wireless adeguata altrimenti essa si connetterà in modalità compatibile. Naturalmente se accade il contrario, ovvero che lo standard adottato non è di ultima generazione ma il dispositivo di connessione del PC lo è, esso si connetterà con lo standard più veloce compatibile a quell'Access Point. Le novità più rilevanti dell'802.11n consistono oltre che nella maggiore velocità possibile attraverso il raddoppio della banda utilizzata quando possibile (con un canale secondario , antecedente o conseguente) cioè 20 + 20 MHz, nell'utilizzo della tecnica MIMO, cioè di più coppie di antenne in diversity e un potente DSP che permette di rifasare i segnali corrotti in arrivo dalle differenti antenne e sommarli in una "interferenza additiva". È implementato anche una sorta di "car pooling" dei pacchetti in transito.

In pratica tutto ciò si traduce in una maggiore velocità reale complessiva del collegamento rispetto all'11g, e un dimezzamento dei tempi di latenza (ping). In questo momento la velocità di trasferimento delle reti wireless è certamente minore rispetto alle reti cablate. Esiste solo un numero limitato di bande di frequenze (suddivise in canali) legalmente utilizzabili in ambiente wireless. Al fine di evitare interferenze, normalmente più Access Point adiacenti utilizzano canali differenti per comunicare con i relativi client.

Le periferiche wireless sono in grado di operare su tutti i canali nella banda a loro assegnata, e possono rapidamente settarsi da uno all'altro per ottenere il collegamento migliore.

Comunque, il numero limitato di frequenze/canali disponibili rende solitamente problematico il funzionamento delle apparecchiature wireless in aree ad alte concentrazioni di Access Point, dove diventa quindi difficile trovare un canale libero da interferenze (in linea teorica solo 3 canali contemporanei non interferiscono tra loro se opportunamente spaziati). Un altro problema presente nelle reti wireless è la necessità di misure di sicurezza contro l'accesso abusivo alla rete e l'intercettazione dei dati in transito. Mentre nella rete cablata la sicurezza del mezzo è racchiusa "tra le proprie mura" (e quindi relativamente più semplice gestire il controllo degli accessi al mezzo fisico), in un sistema di reti wireless il "mezzo fisico" è l'etere, per cui un malintenzionato potrebbe ottenere l'accesso alla rete senza fisicamente introdursi all'interno delle mura. Sono quindi stati studiati diversi sistemi di sicurezza. Una delle tecniche più semplici è quella di consentire l'accesso al proprio Access Point solo ai dispositivi aventi MAC address determinato, ma poiché i MAC address possono essere facilmente clonati, è diventato necessario introdurre sistemi di sicurezza più efficaci. La maggior parte degli Access Point implementano un sistema di cifratura dei dati denominato Wired Equivalent Privacy (WEP), ma anche questo sistema si è rivelato debole (l'informazione è allo stato, decodificabile in pochi minuti attraverso la crittoanalisi). I più recenti Access Point implementano sistemi di crittografia denominati Wi-Fi Protected Access (WPA e WPA2), fornendo più robusti criteri di sicurezza.(la sicurezza assoluta non esiste, per robustezza s’intende l'ordine di grandezza del tempo necessario per decodifica: minuti/ore/anni).

CAPTIVE PORTAL

La tecnica di Captive Portal forza un client http connesso ad una rete di telecomunicazioni a visitare una speciale pagina web (usualmente per l'autenticazione) prima di poter accedere alla navigazione. Ciò si ottiene intercettando tutti i pacchetti, relativi a indirizzi e porte, fin dal momento in cui l'utente apre il proprio browser e tenta l'accesso a Internet.

In quel momento il browser viene indirizzato verso una pagina web la quale può richiedere l'autenticazione oppure semplicemente l'accettazione delle condizioni d'uso del servizio. È possibile trovare software Captive Portal in uso presso gli hotspot WiFi. Può essere altresì usato anche per controllare accessi su rete cablata (es: alberghi, hotel, centri commerciali, ecc.). I Captive Portal hanno avuto un incremento presso i free open wireless network dove al posto dell'autenticazione degli utenti viene semplicemente mostrata una pagina di accettazione delle condizioni d'uso.

Anche se la legge al riguardo è ancora poco chiara (specialmente negli USA) è pensiero comune che forzare gli utenti ad accettare le condizioni d'uso del provider sollevi quest'ultimo da particolari obblighi di legge. La maggior parte di queste implementazioni richiede agli utenti di superare una pagina SSL di login criptata, dopo la quale il loro indirizzo IP e MAC sono abilitati a passare attraverso il gateway.

È stato dimostrato che questo tipo di accesso è facilmente attaccabile tramite un semplice packet sniffer. Una volta ottenuti IP e MAC address di altri computer già autenticati e connessi, chiunque dotato di mezzi e conoscenza tecnica può superare i controlli falsificando le proprie credenziali con quelle degli utenti autorizzati e attraversare indisturbato il gateway. Una soluzione a questo problema consiste nell'utilizzare una finestra di controllo che continuamente rinnova l'autenticazione mandando al gateway un pacchetto criptato.

PHP

PHP (acronimo ricorsivo di "PHP: Hypertext Preprocessor", preprocessore d'ipertesti; originariamente acronimo di "Personal Home Page") è un linguaggio di programmazione interpretato, originariamente concepito per la programmazione di pagine web dinamiche. L'interprete PHP è un software libero distribuito sotto la PHP License. In questo momento è principalmente utilizzato per sviluppare applicazioni web lato server, ma può essere usato anche per scrivere script a riga di comando o applicazioni stand-alone con interfaccia grafica. Nato nel 1994 per opera del danese Rasmus Lerdorf, PHP era in origine una raccolta di script CGI che permettevano una facile gestione delle pagine personali. Il significato originario dell'acronimo era Personal Home Page. Il pacchetto originario venne in seguito esteso e riscritto dallo stesso Lerdorf in C, aggiungendo funzionalità quali il supporto al database MySQL e prese a chiamarsi PHP/FI, dove FI sta per Form Interpreter (interprete di form), prevedendo la possibilità di integrare il codice PHP nel codice TML in modo da semplificare la realizzazione di pagine dinamiche. In quel periodo, 50.000 domini Internet annunciavano di aver installato PHP.

A questo punto il linguaggio cominciò ad avere una certa popolarità tra i progetti open source del web, e venne così notato da due giovani programmatori: Zeev Suraski e Andi Gutmans. I due collaborarono nel 1998 con Lerdorf allo sviluppo della terza versione di PHP (il cui acronimo assunse il significato attuale) riscrivendone il motore che fu battezzato Zend da una contrazione dei loro nomi. Le caratteristiche chiave della versione PHP 3.0, frutto del loro lavoro, erano la straordinaria estensibilità, la connettività ai database e il supporto iniziale per il paradigma a oggetti.

Verso la fine del 1998 PHP 3.0 era installato su circa il 10% dei server web presenti su Internet. PHP diventò a questo punto talmente maturo da competere con ASP, linguaggio lato server analogo a PHP sviluppato da Microsoft, e cominciò ad essere usato su larga scala. La versione quattro di PHP fu rilasciata nel 2000 e prevedeva notevoli migliorie. Attualmente siamo alla quinta versione, sviluppata da un team di programmatori, che comprende ancora Lerdorf, oltre a Suraski e Gutmans. La popolarità del linguaggio PHP è in costante crescita grazie alla sua flessibilità. Nel 2005 la configurazione LAMP (Linux, Apache, MySQL, PHP) supera il 50% del totale dei server sulla rete mondiale. Nel 2008 PHP 5 è diventata l'unica versione stabile in fase di sviluppo. A partire da PHP 5.3.0, PHP implementa una funzione chiamata "late static binding" che può essere utilizzata per fare riferimento alla classe chiamata in un contesto di eredità statica. A partire dal 5 febbraio 2008, a causa dell'iniziativa GoPHP5, sostenuta da una serie di sviluppatori PHP, molti dei progetti open-source di alto profilo cessano di supportare PHP 4 nel nuovo codice e promuovono il passaggio da PHP 4 a PHP 5.

RADIUS

RADIUS (Remote Authentication Dial-In User Service) è un protocollo AAA (authentication, authorization, accounting) utilizzato in applicazioni di accesso alle reti o di mobilità IP. RADIUS è attualmente lo standard de-facto per l'autenticazione remota, prevalendo sia nei sistemi nuovi che in quelli già esistenti ed è implementato in appositi server di autenticazione in una comunicazione con un client che vuole autenticarsi (protocollo client-server). RADIUS è un protocollo che utilizza pacchetti UDP per trasportare informazioni di autenticazione e configurazione tra l'autenticatore e il server RADIUS. L'autenticazione è basata su username, password e, opzionalmente, una risposta a una richiesta di riconoscimento (una sorta di "parola d'ordine"). Se l'autenticazione ha successo, il server RADIUS invia le informazioni di configurazione al client, inclusi i valori necessari a soddisfare il servizio richiesto, come un indirizzo IP e una maschera di sottorete per PPP o un numero di porta TCP per Telnet.

Uno dei limiti del protocollo RADIUS è l'autenticazione basata esclusivamente su password: la password è trasmessa o in forma hash (utilizzando l'algoritmo di hashing MD5), oppure sotto forma di risposta a una richiesta di identificazione (CHAP-password). L'Extensible Authentication Protocol (EAP) rende RADIUS capace di lavorare con una varietà di schemi di autenticazione, inclusi chiave pubblica, Kerberos e smart card. L'access point, ad esempio, agisce da traduttore EAP-RADIUS tra il client wireless e il RADIUS server, utilizzando il protocollo EAP per la comunicazione con il client e il protocollo RADIUS per la comunicazione con il server RADIUS.

L’Access Point incapsula quindi le informazioni (come lo username o la chiave pubblica) in un pacchetto RADIUS che inoltra al server RADIUS. Quando il server rimanda una delle possibili risposte (Access-Accept/Reject/Challenge), l’access point spacchetta il pacchetto RADIUS ed inoltra la risposta al client in un pacchetto EAP. La RFC 2869 (RADIUS Extensions) specifica gli attributi opzionali da impostare sui pacchetti RADIUS per indicare al server RADIUS che si sta utilizzando il protocollo EAP.

Poiché il pacchetto EAP include un campo per specificare quale metodo di autenticazione è in uso, il server RADIUS implementa l’autenticazione richiamando un’apposita procedura. L’identificatore è un ottetto che permette al client RADIUS di associare una risposta RADIUS con la relativa richiesta. Nella sezione degli attributi, infine, è conservato un numero arbitrario di campi. RADIUS è un protocollo ampiamente utilizzato negli ambienti distribuiti ed è comunemente usato per dispositivi di rete integrati come router, server modem, switch ecc., per svariate ragioni: I sistemi integrati generalmente non riescono a gestire un gran numero di utenti con informazioni di autenticazione distinte, poiché questo richiederebbe molta più memoria di massa di quanta ne possiedano la maggior parte di essi. RADIUS facilita l’amministrazione utente centralizzata, che è importante per diverse applicazioni. Molti ISP hanno decine di migliaia, centinaia di migliaia o anche milioni di utenti, aggiunti e cancellati di continuo durante una giornata, e le informazioni di autenticazione cambiano costantemente. L’amministrazione centralizzata degli utenti è un requisito operativo. RADIUS fornisce alcuni livelli di protezione contro attacchi attivi e di sniffing. Altri protocolli di autenticazione remota offrono una protezione intermittente, inadeguata o addirittura inesistente. Un supporto RADIUS è quasi onnipresente. Altri protocolli di autenticazione remota non hanno un consistente supporto da parte dei fornitori di hardware, quando invece RADIUS è uniformemente supportato. Poiché le piattaforme sulle quali è implementato RADIUS sono spesso sistemi integrati, vi sono limitate possibilità di supportare protocolli addizionali. Qualsiasi cambiamento al protocollo RADIUS dovrebbe quantomeno avere una compatibilità minima con client e server RADIUS preesistenti (e non modificati). RADIUS offre la possibilità di eseguire l’autenticazione di utenti remoti anche per particolari siti web che richiedono la protezione dall’accesso del pubblico generale. In particolare c’è un modulo che prevede l’integrazione con il server web Apache, ovviando all’uso dei file .htaccess e .htpasswd con le istruzioni Allow e Deny, che rende l’accesso alle risorse web protetto con le caratteristiche AAA viste precedentemente. Il modulo è stato sviluppato per Apache e si chiama mod_auth_radius. In questo modo Apache diventa un client del server RADIUS, sostituendosi al NAS nell’usuale schema di autenticazione, e dando in out-sourcing la gestione dell’autorizzazione e dell’accounting.

AAA

Il framework AAA ha come scopo quello di fornire le risposte alle seguenti domande:

- "Chi sei?"
- "Dove sei?"
- "Quali servizi sono autorizzato ad offrirti?"
- "Per cosa hai utilizzato i miei servizi, come e per quando tempo?"

Queste sono le domande che il nostro AAA server pone a tutti i client che chiedono un servizio valutando le risposte fornite e tenendo traccia delle loro attività.

AAA significa Autenticazione, Autorizzazione e Accounting.

Autentificazione

L'autenticazione è la richiesta che fa il server AAA al client di identificarsi, attraverso combinazione di username e password o attraverso l'impiego di certificati digitali. La conoscenza della password, o il possesso di un certificato riconosciuto valido dal server AAA, implicano l'accesso allo stadio successivo di autorizzazione. Esistono numerosi modi per comunicare gli estremi di autenticazione. Tra questi figurano: pap, chap, peap, eap/tls, eap/ttls.

Autorizzazione

Una volta identificato l'utente i suoi dati vengono passati al modulo di autorizzazione, che deve stabilire cosa l'utente può fare o meno. Ad esempio, nel caso di una connessione internet si può stabilire quale indirizzo IP utilizzare, qual è la durata massima di una sessione e il traffico massimo che può essere fatto. Un gestore di telefonia mobile GSM/UTMS potrebbe addirittura sfruttare un server AAA per non consentire ulteriori chiamate a chi ha esaurito il proprio credito.

Da un punto di vista prettamente di networking é possibile impostare regole di filtraggio di indirizzi IP, impostazione del routing, limitazioni di ampiezza della banda, utilizzo di comunicazioni con crittografia, e tanto altro ancora. In alcune implementazioni di AAA, tra cui (free)RADIUS è possibile andare a prelevare i dati di autorizzazione direttamente dai database, rendendo infinite le possibilità di utilizzo di questo modulo. Una volta che l'utente é autorizzato, prende il controllo il modulo di accounting.

Accounting

Il modulo di accounting è il terzo ed ultimo del framework AAA, cui non spetta il compito di autorizzare o negare l'accesso a un utente o a un servizio: spetta il compito di tenere traccia delle sue attività. Volendo, si potrebbe pensare come un ragioniere che tiene traccia della quantità di dati scambiati, la natura del servizio erogato, il tempo in cui un utente resta collegato al servizio, e il luogo da cui si é connesso. Questi dati possono essere utilizzati per presentare all'utente la bolletta che sarà proporzionata alla quantità di servizio di cui ha usufruito. L'accounting é una operazione che viene fatta in realtime, offrendo ai gestori dei servizi di sapere informazioni fondamentali, quali il numero di utenti connessi e valutare se ci sono anomalie nell'erogazione dei servizi. Anche per il modulo di accounting é previsto in alcune implementazioni la possibilità di memorizzare le informazioni raccolte in un database.

LDAP

LDAP è un acronimo che sta per LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL. E' un protocollo leggero per accedere ai servizi di directory, basati sul protocollo X.500. LDAP opera su TCP/IP o su altre connessioni orientate ai servizi di trasferimento. I dettagli del protocollo e dell'implementazione sono definiti nel RFC2251: "The Lightweight Directory Access Protocol" e su altri documenti, come ad esempio le specifiche tecniche nel RFC3377.

Prima di LDAP, per accedere a dati memorizzati in una directory X.500 un client doveva supportare il DAP(DIRECTORY ACCESS PROTOCOL),il quale imponeva una notevole penalizzazione delle risorse in gioco in quanto richiedeva l'utilizzo della specifica OSI(Open System Interconnection) che oggi é sostituita largamente dalla suite di protocolli TCP/IP ed altri protocolli. LDAP nasce proprio per sostituire DAP in quanto molto oneroso dal punto di vista dell'impiego delle risorse. LDAP è client-sever: un client LDAP invia una richiesta ad un server LDAP, che processa la richiesta ricevuta, accede eventualmente ad un directory database e ritorna dei risultati al client. Il modello di informazioni di LDAP è basato sulle entry. Un'entry è una collezione di attributi aventi un unico nome globale: il Distinguished Name (DN). Il DN è usato per riferirsi ad una particolare entry, senza avere ambiguità. Ogni attributo dell'entry ha un tipo ed uno o più valori. I tipi di solito sono stringhe mnemoniche, come cn per i common name (i nomi comuni), oppure mail per gli indirizzi di posta elettronica. In LDAP, le entry di una directory sono strutturate come in una struttura gerarchica ad albero. L'entry rappresenta il paese si trova alla radice dell'albero.

Al di sotto di essa ci sono quelle che rappresentano stati e organizzazioni nazionali. Seguono poi altri tipi di entry che possono rappresentare organizzazioni, persone, stampanti, documenti, ecc. Un'entry è indicata con il suo Distinguished Name, che è costruito prendendo il nome stesso dell'entry (chiamato Relative Distinguished Name, RDN) e concatenandolo ai nomi delle entry dei suoi predecessori nell'albero. Il protocollo LDAP definisce servizi per accedere e aggiornare una directory. Fornisce, infatti, operazioni per aggiungere o cancellare un'entry da una directory o modificarne una già esistente, oppure cambiarne il nome. LDAP fornisce un efficiente algoritmo di ricerca. Molti servizi di directory non prevedono protezione per i dati e le informazioni, cosi chiunque può accederci. Al contrario LDAP include un meccanismo nel quale un client si può autenticare, o provare la sua identità ad una directory server; inoltre consente servizi di privacy e di integrità delle informazioni. Il servizio di directory LDAP è basato su un modello client – server. Uno o più server LDAP contengono i dati che servono a costruire l'albero delle informazioni di una directory, il DIT (Directory Information Tree). La radice di un DIT è una DSA-Specific Entry (DSE) e non una parte dei nomi del contesto: ogni server ha differenti attributi e valori nella root DSE. Il client si connette al server e gli chiede informazioni. Il server replica con risposte precise e/o con un puntatore che indica dove il client può accedere ad informazioni addizionali, tipicamente un altro server LDAP. Il client LDAP può comunicare sia con un server X.500 sia con un server LDAP.

ACTIVE DIRECTORY

Active Directory è un database integrato nei server Windows 2008 e Windows 2012 che funge da domain controller, consente di catalogare e gestire in modo centralizzato risorse diverse come: utenti, gruppi di lavoro, stampanti, cartelle condivise, ecc. La struttura del database è di tipo gerarchico con contenitori che contengono oggetti e altri contenitori.

Active directory è una struttura che deve essere presente in ogni ambiente informatico in cui il server o i server devono avere il controllo su tutti i client della rete. Active Directory è un insieme di servizi di rete meglio noti come directory service adottati dai sistemi operativi Microsoft a partire da Windows 2000 Server. Si fonda sui concetti di dominio e di Directory, che in inglese sta a significare "elenco telefonico".

Active Directory può essere immaginato come un "elenco telefonico" e il Dominio come un mondo in cui vengono concentrate tutte le risorse della rete a partire da:

- account utente,
- account computer,
- cartelle condivise,
- stampanti ecc. ecc.

L'insieme dei servizi di rete di Active Directory, ed in particolare il servizio di autenticazione Kerberos realizzano un'altra delle caratteristiche importanti il Single Sign-On (SSO). Tramite tale meccanismo un utente, una volta entrato nel dominio ed effettuato quindi il login da una qualsiasi delle macchine di dominio, può accedere a risorse disponibili in rete (condivisioni, mailbox, intranet ecc.) senza dover effettuare nuovamente l'autenticazione facilitando la gestione degli utenti a differenza di quanto accade nelle reti peer to peer. Active Directory è il nome utilizzato da Microsoft per riferirsi alla sua implementazione della sicurezza in una rete distribuita di computer. Utilizza vari protocolli (principalmente LDAP, DNS, DHCP, Kerberos.). In Active Directory LDAP viene usato come un data base che memorizza in forma centralizzata tutte le informazioni di un dominio di rete, relativamente ad autenticazioni ed accesso ai servizi, col vantaggio di mantenere tutte queste informazioni sincronizzate tra i vari server di autenticazione di accesso alla rete. Le reti Active Directory possono variare da una singola installazione con poche centinaia di oggetti a grandi installazioni con milioni di oggetti.

Diversamente dai vecchi sistemi di gestione account e server come User manager for domain e Server manager for domain, AD include in un unico sistema di monitoraggio tutti gli oggetti in tre ampie categorie del dominio: risorse (es. stampanti), servizi (es. email) e utenti (account utenti e gruppi). AD fornisce informazioni sugli oggetti, li organizza, controlla gli accessi e imposta le security. AD è un raggruppamento logico di utenti e computer in un dominio, gestito centralmente da server detti "controllori di dominio". Una struttura 'Active Directory' è un framework gerarchico di oggetti. AD fornisce informazioni sugli oggetti, li organizza, controlla l'accesso e ne imposta la sicurezza. Ciascun oggetto rappresenta una singola entità - magari un utente, un computer, una stampante oppure un gruppo - con i propri attributi. Alcuni oggetti possono anche essere contenitori di altri oggetti. Un oggetto è identificato univocamente dal suo nome e ha un insieme di attributi - le caratteristiche e l'informazione che l'oggetto può contenere - definiti da uno schema, che determina anche il tipo di oggetti che possono essere immagazzinati in Active Directory. Ciascun oggetto attributo può essere utilizzato in differenti classi d'oggetto di uno schema. L'oggetto schema esiste per permettere allo schema di essere esteso o modificato quando necessario. Comunque, poiché ogni oggetto schema è esso stesso parte della definizione degli oggetti Active Directory, la disattivazione o la modifica dell'oggetto schema può avere serie conseguenze poiché modificherà in maniera fondamentale la struttura di Active Directory stesso. Un oggetto schema se modificato, si propagherà automaticamente attraverso Active Directory e, una volta creato potrà soltanto essere disattivato o non cancellato. I Siti contengono oggetti detti Sottoreti e possono essere usati per assegnare oggetti "politica di gruppo", semplificare l'individuazione delle risorse, gestire la replicazione della AD e gestire il traffico di collegamento alla rete. I Siti possono essere collegati ad altri Siti. Agli oggetti di un sito collegato possono essere assegnati costi che rappresentano la velocità, affidabilità, disponibilità o altre proprietà reali di una risorsa fisica. Ai Collegamenti ai Siti può essere anche assegnata una pianificazione. La struttura di AD può essere suddivisa logicamente in tre diverse entità: i domini, gli alberi e le foreste. Un dominio rappresenta un insieme di macchine connesse fra di loro e che condividono un directory database comune in cui sono inseriti gli oggetti. I domini sono identificati in base alla struttura del loro nome DNS, il namespace. Un albero è l'insieme di uno o più domini che condividono uno spazio di nomi contiguo. Tali domini sono collegati fra loro in modo gerarchico e i diversi controllori di dominio possono scambiarsi informazioni reciprocamente in quella che viene definita una relazione di fiducia transitiva (transitive trust relationship). Al livello più alto della struttura viene definita la foresta, ovvero l'insieme di alberi presenti nella directory. Questi alberi condividono fra loro un catalogo globale, uno schema di directory, una struttura logica ed una configurazione.

La foresta rappresenta perciò l'area in cui utenti, computer, gruppi ed altri oggetti sono accessibili. In una foresta composta da più domini, il server che esegue il dominio su cui è stata installata la struttura di AD principale (ovvero quella di più alto livello) viene chiamato controller di dominio primario (Primary Domain Controller). Gli oggetti Active Directory presenti all'interno di un dominio possono essere raggruppati fra loro in unità organizzative (Organizational Units - OU). Utilizzando queste strutture è possibile formare una gerarchia all'interno del dominio e facilitarne così l'amministrazione, ad esempio suddividendo la struttura in termini geografici. Le unità organizzative sono il livello raccomandato per realizzare politiche di gestione dei gruppi, le quali sono oggetti Active Directory chiamati ufficialmente Group Policy Objects (GPOs) per la delegazione di poteri amministrativi. Le unità organizzative rappresentano in ogni caso una semplice astrazione realizzabile per scopi amministrativi e non forniscono perciò un contenitore fisico di oggetti. In questo contesto non è possibile definire ad esempio un account utente con lo stesso nome in due unità organizzative appartenenti allo stesso dominio poiché la visibilità effettiva è sempre limitata al dominio prescelto e non alle unità organizzative in esso contenute. Active Directory fu distribuita come beta nel 1996 per la prima volta con Windows 2000 e, in Windows Server 2003 venne distribuita una versione con nuove funzionalità. Ulteriori miglioramenti sono stati fatti in Windows Server 2003 R2. E' stata ulteriormente raffinata con Windows Server 2008 e Windows Server 2008 R2 ed è stata rinominata in Active Directory Domain Services.

SHIBBOLETH

SITUAZIONE DI PARTENZA

Già da un po' di tempo all'interno dell'Università è operativo il servizio di gestione delle identità Shibboleth IdP, testato, configurato e aggiornato secondo le varie esigenze. Partire da un servizio già presente ha facilitato lo sviluppo del progetto, poiché si è usufruito di un servizio di Identity Provider già esistente Il problema è stato quello di integrare una risorsa web nella rete universitaria, in cui è già presente il servizio di IDP. Il CAS (Central Authentication Service) è il servizio di autenticazione centrale che offre il sistema del Single-Sign-On per l'autenticazione sul web degli utenti. L'IDP si rivolge a questo servizio per autenticare gli utenti secondo il meccanismo del SSO. L'LDAP è il database centrale dell'Università sul quale sono memorizzate le informazioni di tutti gli utenti o affiliati dell'Università. L'IDP interagisce con l'LDAP per la richiesta ed il rilascio delle informazioni (o attributi) utente. Questi attributi saranno successivamente inviati dall'IdP alle risorse (Service Providers) che ne fanno richiesta.

I SP utilizzeranno questi attributi per conoscere meglio l'identità dell'utente ed assegnarli di conseguenza i giusti privilegi d'autorizzazione. Shibboleth è un progetto del gruppo Internet2 e rappresenta una soluzione open source per l'accesso a risorse e servizi web, condivisi tramite credenziali di autorizzazione. Shibboleth consente ai suoi utenti di poter inviare in sicurezza informazioni fidate riguardanti la loro identità a risorse remote. Queste informazioni possono essere poi utilizzate per l'autenticazione, l'autorizzazione, la personalizzazione di contenuti e per l'abilitazione del Sign-on su tutta un'ampia sfera di servizi provenienti da differenti Providers. E' un sistema federato, poiché supporta accesso sicuro a risorse distribuite tra diversi domini di sicurezza.

Le informazioni sugli utenti sono inviate da un Identity Provider ad un Service Provider che prepara le informazioni per la protezione di dati sensibili che sono utilizzate dalle applicazioni. Le federazioni, che non sono soltanto delle strutture puramente tecniche, possono spesso essere utilizzate per aiutare i providers a fidarsi reciprocamente. Implementa il Web-SSO all'interno di una organizzazione o tra organizzazioni diverse e di conseguenza garantisce sicurezza e privacy.

Usa SAML (Security Assertion Markup Language), un protocollo basato su XML per lo scambio d'informazioni di autenticazione e d'autorizzazione codificate da Shibboleth in attributi ed esportati sottoforma di asserzioni SAML.

IDEM FEDERATO

Si può definire come l'insieme di tecnologie standard ed accordi che permettono a un insieme di Service Providers (SP) di accettare come validi gli identificatori utente gestiti da un altro insieme di Providers, detti Identity Provider (IDP). Tale comunità di providers (SP e IDP) viene tipicamente denominata **federazione**. Il ruolo principale della federazione è quello di gestire le relazioni tra gli Identity Provider e i Service Provider, i quali risultano "federati" tra loro. Un tale approccio *implementa implicitamente il Single Sign On (SSO)*, ovvero la possibilità per un utente di autenticarsi presso uno qualsiasi dei providers della federazione e, successivamente, di accedere ai servizi di tutti gli altri. Questo approccio alla gestione delle identità è stato sviluppato per rispondere ad un bisogno di gestione decentralizzata degli utenti: ogni gestore federato mantiene il controllo della propria politica di sicurezza.

I vantaggi sono evidenti:

- Gli utenti si registrano in un unico punto (l'IDP) e accedono facilmente a molte risorse che la federazione mette loro a disposizione
- C'è un controllo fine riguardo al rilascio di informazioni che riguardano l'utente
- L'accesso è basato su standard e prodotti open source
- Nuove risorse e nuovi utenti possono essere aggiunti velocemente
- Il diritto di accesso è regolato dagli attributi e non dalle credenziali o dall' IP
- Efficienza e qualità: federare migliora la gestione degli utenti ed attenua ridondanza di credenziali.

Una **Federazione** è un *accordo* tra organizzazioni, che decidono di fidarsi reciprocamente delle informazioni che si scambiano (condividendo risorse, servizi o applicazioni), sulla base di regole e attraverso una infrastruttura (AAI) che è certificata e sicura.

Richiede l'attuazione di politiche comuni e linee di condotta, al fine di gestire le relazioni di fiducia tra i vari partecipanti. Un **membro** di una federazione rappresenta una qualsiasi organizzazione, (università, ente di ricerca, etc.) che ha sottoscritto il contratto con la Federazione. I membri concordano l'aspetto legale, le policies e la tecnologia da adottare. Ogni federazione mette a disposizione un particolare tipo di servizio oltre a quelli di IDP e SP, ed è il servizio WAYF, che consente agli utenti di scegliere la propria Organizzazione di Appartenenza.

Esiste un solo WAYF (Where Are You From) per tutta la federazione ed è un servizio centrale, che, interpellato dai Service Provider della federazione, ha il compito di ridirigere l'utente sull'IDP di appartenenza (Home Organization) che contiene l'identità digitale dell'utente, in modo che questo si possa autenticare correttamente e usufruire poi del servizio scelto.

La stessa federazione svolge inoltre i seguenti compiti:

- valida i nuovi enti che intendono aderire;
- gestisce la lista di tutti i partecipanti;
- aderisce agli standard tecnici tradizionali;
- assicura una politica di controllo;
- fornisce supporto e consigli.

Sono nate federazioni su base nazionale in oltre quindici paesi europei, tra cui Gran Bretagna, Spagna e Svizzera, oltre che negli Stati Uniti, in Canada, in Nuova Zelanda e in Australia.

In Italia GARR promuove il progetto IDEM, per la realizzazione di un'Infrastruttura di Autenticazione e Autorizzazione federata per l'accesso ai servizi, al quale partecipano Università e Centri di Ricerca, che dà la possibilità a ricercatori, docenti e studenti di utilizzare lo stesso sistema standard di gestione degli accessi. All'interno di una federazione diventa possibile la *gestione federata dell'accesso alle risorse* (IDEM federato), che riduce il carico di lavoro per gestire le credenziali, e facilita il controllo stesso degli accessi alle risorse. Esistono varie *soluzioni tecnologiche* per realizzare la gestione federata degli accessi. Esse definiscono un insieme di protocolli per lo scambio sicuro delle informazioni riguardanti le identità tra istituzioni e fornitori di servizi, e adottano un approccio conforme allo standard SAML. Shibboleth è un esempio di questa tecnologia, adottata dalla Federazione IDEM del GARR. All'interno della federazione, l'informazione su ciascun utente è detenuta soltanto dall'organizzazione alla quale l'utente stesso è affiliato, detta Home Organization, o organizzazione d'appartenenza (IDP), che fornisce l'identità digitale dell'utente.

Ciò significa che per ogni istituzione (es. università, biblioteche, ospedali ...) esiste un singolo punto centrale di gestione delle identità. I fornitori di risorse devono fidarsi della dichiarazione dell'organizzazione di appartenenza sull'identità dell'utente, e quest'ultime devono accordarsi sulla rappresentazione di queste identità, attraverso uno schema degli attributi. Per garantire la privacy dell'utente, le informazioni scambiate tra fornitore d'identità e fornitore di risorsa non riguardano informazioni personali sull'utente (ad es. dati d'autenticazione) ma solo gli attributi, per i quali c'è stato già un accordo, che servono ad assegnare il diritto o meno di accesso alla risorsa.

Queste informazioni dette "attributi" possono semplicemente consistere nella dichiarazione che un utente è un insegnante o uno studente di una particolare università o scuola. Le varie istituzioni, come scuole, università, centri di ricerca, strutture del settore pubblico e partner commerciali, possono tutte trarre vantaggio dall'approccio federato per l'accesso alle risorse. Il sistema, infatti, permette agli utenti di autenticarsi presso la propria istituzione e ciò offre opportunità di collaborazione e di gestione degli accessi.

Per poter adottare il sistema federato di accesso, le istituzioni dovranno intraprendere varie attività, come:

1) definire quali necessità si vogliono soddisfare tramite la gestione degli accessi e valutare la propria capacità di gestione delle identità attraverso una verifica istituzionale;
2) sviluppare dei sistemi di directory (LDAP) per effettuare la gestione delle identità all'interno della istituzione;
3) scegliere un adeguato sistema di autenticazione;
4) implementare il sistema di gestione delle identità, l'Identity Provider (IDP);
5) aderire ad una federazione (es. federazione IDEM del GARR);
6) predisporre corsi di formazione per il personale, manuale per gli utenti e relativo supporto.

SINGLE SIGN ON

Nell'approccio tradizionale alla gestione delle identità (l'IDEM isolato), i sistemi distribuiti sono costituiti da più componenti, ciascuno con un proprio dominio di sicurezza, da cui la necessità per l'utente di autenticarsi presso ogni componente con cui deve interagire.

Considerazioni legate all'usabilità e alla sicurezza suggeriscono di coordinare e integrare i servizi di autenticazione e la gestione degli account degli utenti attraverso un sistema di Single Sign On. Il **Single-Sign-On** (**SSO**, traducibile come *autenticazione unica* o *identificazione unica*) è un particolare sistema di autenticazione centralizzata, che consente a un utente di fornire le proprie credenziali una sola volta e di accedere a tutte le risorse informatiche o applicazioni alle quali è abilitato, all'interno di una rete locale (LAN) o della rete Internet.

Gli obiettivi sono multipli:

- semplificare la gestione delle password: maggiore è il numero delle password da gestire, maggiore è la possibilità che saranno utilizzate password simili le une alle altre e facili da memorizzare, abbassando così il livello di sicurezza;
- semplificare la gestione degli accessi ai vari servizi;
- semplificare la definizione e la gestione delle politiche di sicurezza.

Un sistema di SSO apporta benefici sia lato client sia server:

- lato client, aumenta l'usabilità delle applicazioni dal punto di vista utente, il quale impiega meno tempo nelle operazioni di login e non è costretto a ricordare numerose credenziali per accedere a sistemi diversi;
- lato server, gli amministratori possono gestire gli account utente e i diritti di accesso (autorizzazioni) in modo centralizzato; tutto ciò consente una gestione semplificata degli account e rafforza la sicurezza del sistema.

Shibbboleth è un pacchetto software open source e standardizzato che ha per obiettivo il web Single Sign On, molto ricco per quello che riguarda lo scambio degli attributi, basato su standard aperti tra cui prevalentemente SAML2. E' un sistema federato che supporta l'accesso sicuro a risorse attraverso il web, le informazioni relative all'utente vengono trasmesse dall'Identity Provider (IDP) aziendale al Service Provider (SP) esterno che le prepara per la protezione dei dati sensibili e per l'utilizzo da parte delle applicazioni.

Il processo di autenticazione avviene nei seguenti termini:

1. l'utente richiede la connessione ad una risorsa protetta facendo sì che il SP intercetti la richiesta (la risorsa da proteggere è definita nei files di configurazione del web server che la ospita);
2. il SP basandosi sulla configurazione descritta, determina a quale IdP far riferimento e quale protocollo utilizzare attraverso un meccanismo di "scoperta" noto come servizio WAYF (acronimo di Where Are You From). La richiesta di autenticazione è così delegata al WAYF che la passa all'IDP selezionato dall'utente;
3. come risultato delle azioni precedenti, una richiesta di autenticazione via browser è inviata dal SP all'IDP selezionato dall'utente. L'IDP decide se l'utente può autenticarsi e quali attributi inviare al SP, scelta basata prevalentemente sulle caratteristiche del SP che fornisce il servizio e su quelle dell'attributo principale dell'utente;
4. l'IDP impacchetta e firma i dati che trasmette sotto forma di asserzione SAML al SP che li spacchetta e li decodifica eseguendo poi una serie di controlli di sicurezza per decidere se il richiedente ha o meno diritto di accesso alla risorsa desiderata;
5. infine, se il processo di verifica ha dato esito positivo, l'utente viene finalmente indirizzato alla risorsa richiesta.

IL PROTOCOLLO SAML

Negli ultimi anni il mondo degli ambienti federati sta ingrandendosi sempre più. Di pari passo, sono nati diversi software che offrono un supporto alla gestione di sistemi federati. In questi ambienti, dove i governi e le istituzioni potrebbero utilizzare differenti software “federati” l’interoperabilità diventa estremamente importante. Shibboleth offre questo supporto grazie all’adozione di alcuni protocolli che assicurano l’interoperabilità con le altre implementazioni commerciali.

Tra questi il protocollo SAML, che è diventato ormai uno standard nei sistemi Web-SSO per lo scambio dei dati di autenticazione/autorizzazione; tale protocollo definisce anche il formato dei metadati utilizzati ampiamente da Shibboleth per rappresentare informazioni di vario tipo, come ad esempio ruoli o certificati, legati ai membri di una federazione. Molte soluzioni per la gestione d’identità federate (es. Shibboleth) si fondano sul Security Assertion Markup Language (SAML), un framework basato su XML per comunicare tra un’autorità e terze parti fidate l'avvenuta autenticazione dell'utente, il diritto ad usufruire di una risorsa/servizio e i suoi attributi.

Shibboleth supporta diversi formati, protocolli e versioni di SAML, con lo scopo di combinare unitamente sicurezza e informazioni personali, per scambiarle tra diversi domini.

La funzione essenziale di SAML è fornire un sistema standard per passare le informazioni di sign-on e di autorizzazione tra due domini federati, attraverso messaggi che prendono il nome di asserzioni.

SAML abilita lo scambio d’informazioni di autenticazione e autorizzazione su utenti, dispositivi o qualsiasi entità identificabile (chiamata in gergo subject). Nel caso di una AAI, un SP è in grado di contattare un IdP per autenticare gli utenti che stanno cercando di accedere a contenuti protetti.

SAML definisce tre tipi di asserzioni:

- **Authentication** (Autenticazione), indica che un soggetto è stato autenticato precedentemente (per es. password o chiave pubblica X.509).
- **Authorization** (Autorizzazione), indica se un soggetto può essere concesso o negato l'accesso alle risorse.
- **Attribution** (Assegnazione), indica che il soggetto è associato con attributi.

Utilizzando un sottoinsieme di istruzioni XML, SAML definisce il protocollo di richiesta-risposta attraverso i quali i sistemi accettano o rifiutano le asserzioni del subject. SAML non specifica il grado di fiducia che dovrebbe essere inserito all'interno di un'asserzione.

Sono i sistemi locali a decidere se i livelli o le politiche di sicurezza di una determinata applicazione siano sufficienti a proteggere un'azienda in caso occorrano problemi per una decisione di autorizzazione basata su un'asserzione inaccurata e mal formulata.

Questa caratteristica di SAML migliorerà relazioni e accordi fiduciari tra attività basate su Web, in cui prima di accettare un' asserzione si decide di aderire a un livello base di verifica.

METADATI

I metadati possono essere visti come le "Carte d'Identità" (in formato XML) dei partecipanti "fidati", cioè appartenenti alla federazione e sono utilizzati come strumento con cui si costruiscono le relazioni di fiducia fra i membri.

Shibboleth utilizza i metadati per comunicare informazioni agli IdP fidati, ai Service Provider e per distribuire informazioni relative alle CA (autorità di certificazione).

Sono raccolti in un file basato su standard SAML 2.0 e contengono:

- Certificati
- Scope degli IDP
- Descrizione testuale dei partecipanti

È consentito l'utilizzo di certificati self-signed per la comunicazione SP-IdP (back-channel).

Ogni partecipante per verificare l'identità della controparte e comunicare, utilizza il relativo certificato contenuto nei metadati. Il SP parla solo con un IdP noto (i cui dati siano nel file dei MD).

STRUTTURA DI SHIBBOLETH

Il sistema Shibboleth è costituito di due componenti, che disaccoppiano la gestione delle identità degli utenti dalla gestione delle risorse:

- Shibboleth IDP (Identity Provider);
- Shibboleth SP (Service Provider);

L'Identity Provider amministra le informazioni degli utenti che ha in carico e può essere utilizzato per:

1) registrare gli Utenti e mantenerne le informazioni.
2) gestire le sessioni di autenticazione. Dev'essere in grado di autenticare i propri utenti, come la richiesta di credenziali, oppure verificare che l'utente abbia già una sessione valida (di Single Sign On).
3) rilasciare attributi degli utenti autenticati di cui ne è responsabile e del loro rilascio a chi ne fa richiesta, proteggendo i dati personali. Tali informazioni sono richieste dai Service Provider per autorizzare l'utente all'uso di un servizio. Non tutti i dati sono inviati al SP ma solo quelli ritenuti necessari.

Ogni risorsa, il cui accesso deve essere protetto, richiede un Service Provider (SP), che si occupa di indirizzare l'utente nella propria Organizzazione d'Appartenenza (o Identity Provider) per il suo riconoscimento ed il rilascio di attributi. Il SP raccoglie le informazioni sull'utente inviategli dall'IdP e le usa per proteggere il servizio e concedere l'autorizzazione all'utente che ne ha fatto richiesta.

IDENTITY PROVIDER

L'IDP shibboleth è un'applicazione web java composta da una serie di parti legate fra loro come:

- **Handler Manager**: servizio che gestisce i vari endpoint dai quali l'IDP può ricevere messaggi.
- **Attribute Resolver**: responsabile del recupero di attributi da un database e della loro combinazione e trasformazione in un insieme di dati relativi a uno stesso utente. Lo scopo è quello di rimandarli indietro tutti o in parte al client dell'IdP.
- **Attribute Filter**: creare una collezione filtrata di attributi sulla base di un insieme di regole. Queste regole riflettono quali attributi e valori un client può ricevere.
- **Attribute Authority (AA)**: servizio che dipende dall'Attribute Resolver, prende un insieme di attributi e li codifica in un'asserzione di attributi SAML. Se i metadati SAML del richiedente contengono informazioni sul tipo di attributi richiesti, il servizio effettuerà una operazione di filtraggio rilasciando solo quelli necessari.

PROCEDIMENTO

In questa parte della trattazione verrà esposto il metodo che è stato usato per creare ben tre configurazioni differenti di autentificazione per il WiFi Unicam.

La prima autentificazione avviene tramite immissione di dati in una pagina Captive Portal dove gli utenti sono gestiti da un server Radius/Ldap riassumibile nelle seguenti fasi:

- Flash e configurazione firmware OpenWRT
- Installazione pacchetto NoDogSplash su OpenWRT
- Installazione macchina virtuale Windows Server 2003 SP2
- Configurazione IAS
- Installazione Certificati
- Installazione Server Web Apache con Xampp
- Configurazione Active Directory
- Scrittura Pagina Captive Portal con PHP
- Collegamento Access Point al server IAS
- Test Configurazione

La seconda consiste nella creazione di due reti Wireless separate con entrambe autentificazione WPA2 ma con server Radius/Ldap distinti.

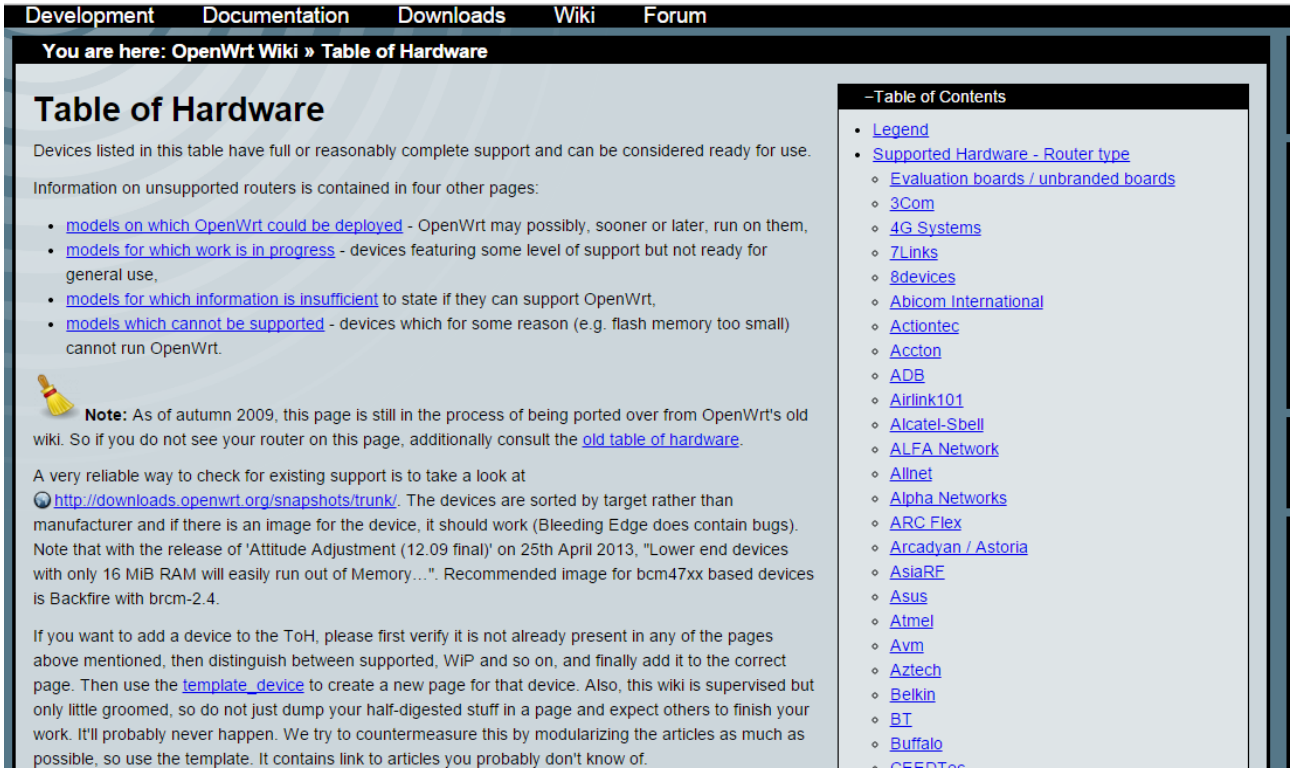
Nella terza è stato implementato il pacchetto Shibboleth, comportando l'installazione di un Service Provider collegato all'Identity Provider di Unicam. Le fasi del procedimento posso essere riassunte nel seguente modo:

- Installazione Partizione Ubuntu 14,04 LTS
- Installazione Service Provider
- Collegamento all' Identity Provider Unicam
- Collegamento Access Point al Service Provider tramite RedirectUrl
- Test configurazione

PRIMA CONFIGURAZIONE

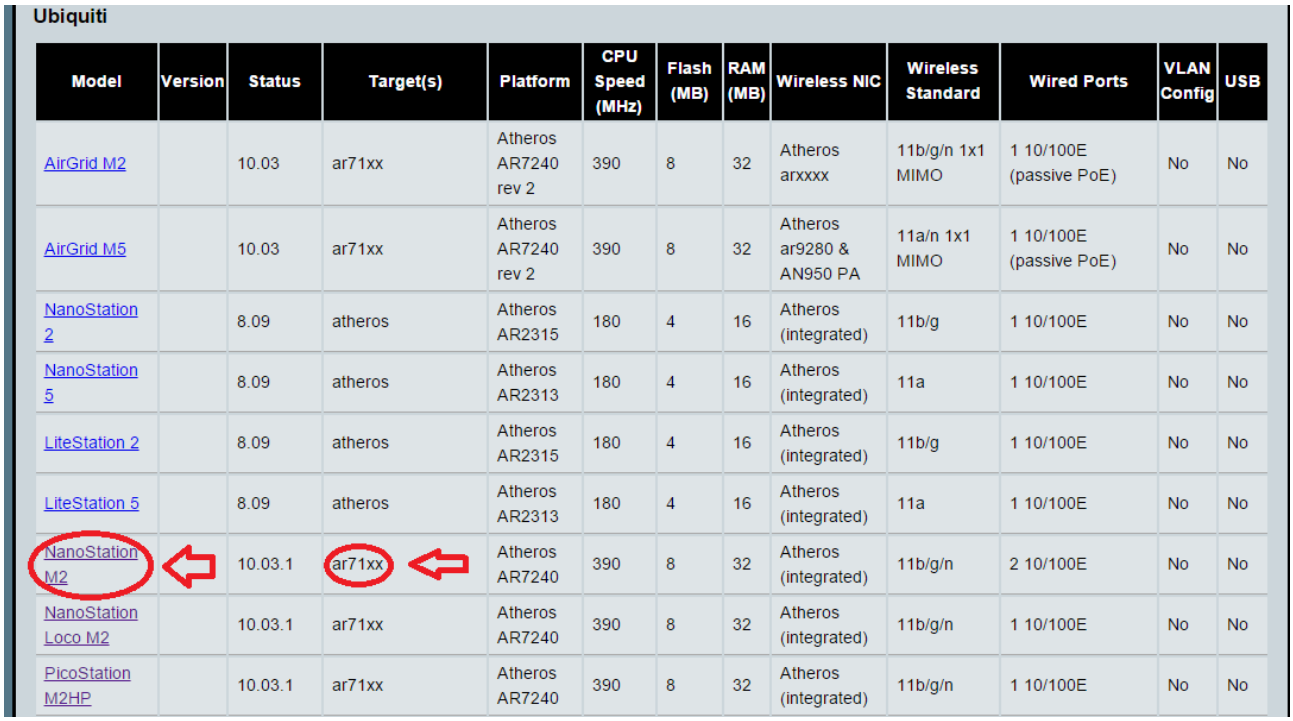
FLASH FIRMWARE OPENWRT

Per poter flashare il firmware, occorre controllare nella wiki di OpenWrt se il modello di Access Point a disposizione è supportato e ,in caso affermativo, quale versione del firmware va installata e come. Per poter determinare ciò basterà collegarsi alla "Table Of Hardware" presente nel seguente link: *http://wiki.openwrt.org/toh/start,* dove si presenterà una schermata come questa:



Nel lato destro sono indicate le case produttrici degli Access Point supportate dal firmware, ad esempio 3Com, Dlink, Huawei, NetComm, Ubiquiti, ZyZEL, ecc. .

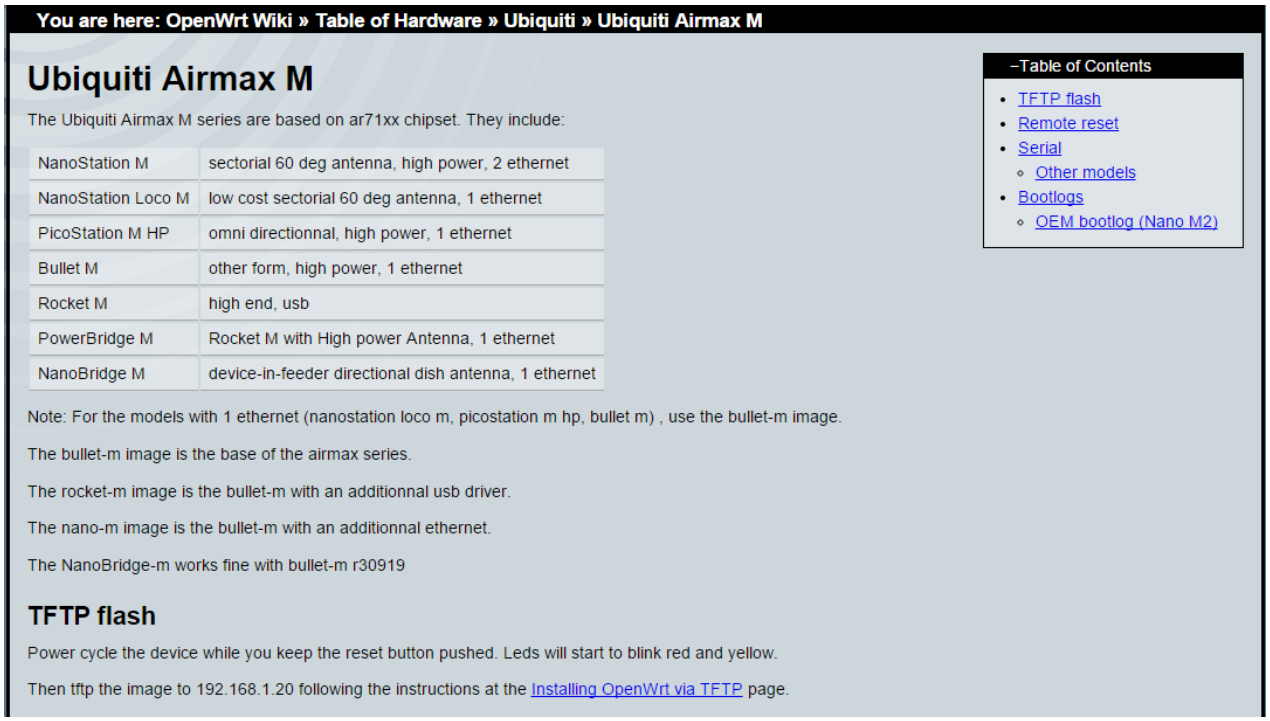
Visto che gli Access Point disponibili sono degli “Ubiquiti” , si cliccherà sul nome della casa per essere indirizzati alla sezione dedicata:



Ubiquiti

Model	Version	Status	Target(s)	Platform	CPU Speed (MHz)	Flash (MB)	RAM (MB)	Wireless NIC	Wireless Standard	Wired Ports	VLAN Config	USB
AirGrid M2		10.03	ar71xx	Atheros AR7240 rev 2	390	8	32	Atheros arxxxx	11b/g/n 1x1 MIMO	1 10/100E (passive PoE)	No	No
AirGrid M5		10.03	ar71xx	Atheros AR7240 rev 2	390	8	32	Atheros ar9280 & AN950 PA	11a/n 1x1 MIMO	1 10/100E (passive PoE)	No	No
NanoStation 2		8.09	atheros	Atheros AR2315	180	4	16	Atheros (integrated)	11b/g	1 10/100E	No	No
NanoStation 5		8.09	atheros	Atheros AR2313	180	4	16	Atheros (integrated)	11a	1 10/100E	No	No
LiteStation 2		8.09	atheros	Atheros AR2315	180	4	16	Atheros (integrated)	11b/g	1 10/100E	No	No
LiteStation 5		8.09	atheros	Atheros AR2313	180	4	16	Atheros (integrated)	11a	1 10/100E	No	No
NanoStation M2		10.03.1	ar71xx	Atheros AR7240	390	8	32	Atheros (integrated)	11b/g/n	2 10/100E	No	No
NanoStation Loco M2		10.03.1	ar71xx	Atheros AR7240	390	8	32	Atheros (integrated)	11b/g/n	1 10/100E	No	No
PicoStation M2HP		10.03.1	ar71xx	Atheros AR7240	390	8	32	Atheros (integrated)	11b/g/n	1 10/100E	No	No

In questa sezione ci vengono delineate le caratteristiche fondamentali dell’apparecchio, come ad esempio la scheda di rete, la memoria Flash, la RAM, se sono disponibili porte USB, il Wireless NIC, lo standard Wireless e la tipologia del firmware supportata.



You are here: OpenWrt Wiki » Table of Hardware » Ubiquiti » Ubiquiti Airmax M

Ubiquiti Airmax M

Table of Contents

- TFTP flash
- Remote reset
- Serial
  - Other models
- Bootlogs
  - OEM bootlog (Nano M2)

The Ubiquiti Airmax M series are based on ar71xx chipset. They include:

NanoStation M	sectorial 60 deg antenna, high power, 2 ethernet
NanoStation Loco M	low cost sectorial 60 deg antenna, 1 ethernet
PicoStation M HP	omni directionnal, high power, 1 ethernet
Bullet M	other form, high power, 1 ethernet
Rocket M	high end, usb
PowerBridge M	Rocket M with High power Antenna, 1 ethernet
NanoBridge M	device-in-feeder directional dish antenna, 1 ethernet

Note: For the models with 1 ethernet (nanostation loco m, picostation m hp, bullet m) , use the bullet-m image.

The bullet-m image is the base of the airmax series.

The rocket-m image is the bullet-m with an additionnal usb driver.

The nano-m image is the bullet-m with an additionnal ethernet.

The NanoBridge-m works fine with bullet-m r30919

TFTP flash

Power cycle the device while you keep the reset button pushed. Leds will start to blink red and yellow.

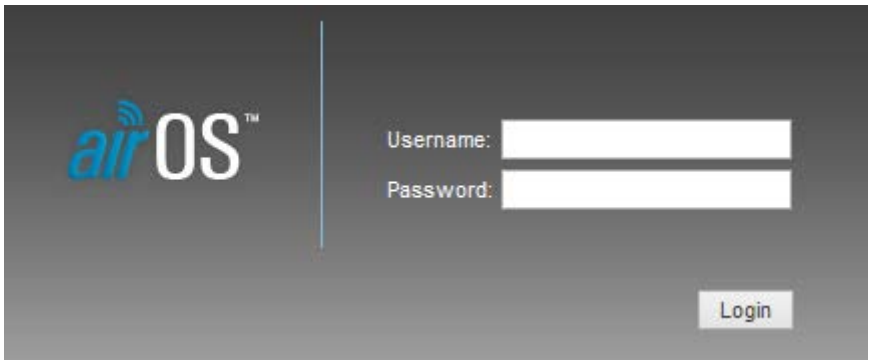
Then tftp the image to 192.168.1.20 following the instructions at the Installing OpenWrt via TFTP page.

Nell'immagine è possibile notare il modello NanoStation M2 supporta la tipologia air7xx. Identificato il firmware, questo può essere scaricato, basterà cliccare sul nome del modello per essere indirizzati su una nuova pagina dedicata a quest'ultimo.

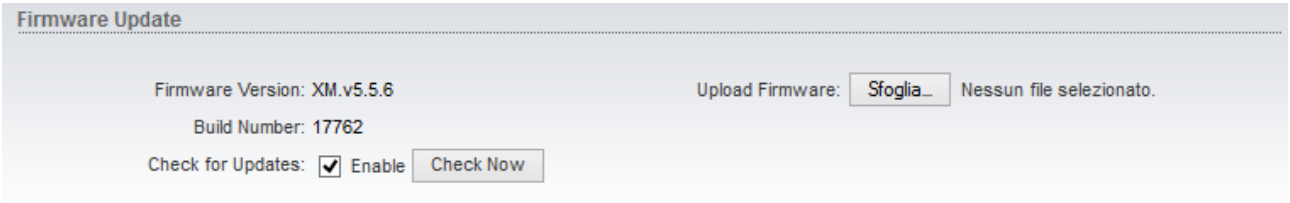
Nella pagina vengono indicati in modo molto sintetico le componenti hardware dei dispositivi della famiglia Ubiquiti Airmax M, su quale chipset sono basati (air7xx come precedentemente accennato) e delle note riguardanti il firmware da installare.

Dato che si possiede un NanoStation Loco M, con solo una porta Ethernet, si dovranno flashare le immagini "bullet-m". La procedura di flashing può essere eseguita sia tramite TFTP o semplice Firmware Upgrade collegandosi al Access Point sul browser.

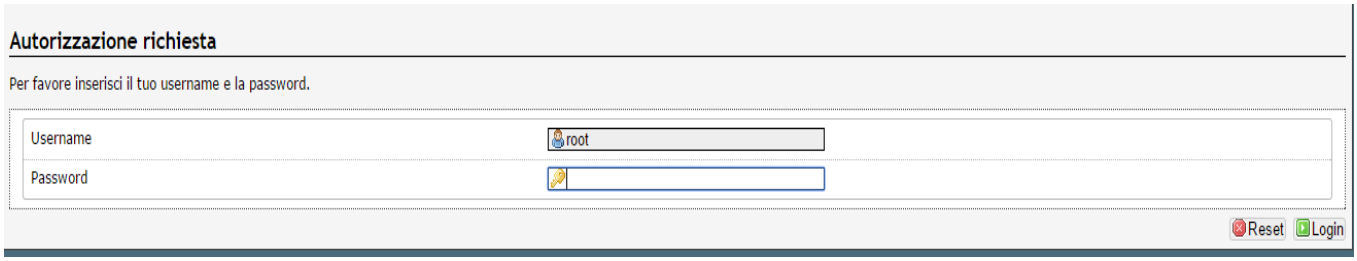
Per rendere possibile il flashing sarà necessario collegare il dispositivo tramite cavo Ethernet, ad un pc dove è stato precedentemente impostato un IP fisso così da creare una sottorete a cui si possa interfacciare il dispositivo. Una volta fatto ciò basterà scrivere nella barra degli indirizzi il seguente indirizzo: 192.168.1.20. Ci apparirà la seguente schermata



Si accede con le dovute credenziali e nella sezione dedicata al sistemata si sceglie la dicitura "Upgrade Firmware" e si carica il firmware OpenWrt precedentemente scaricato.



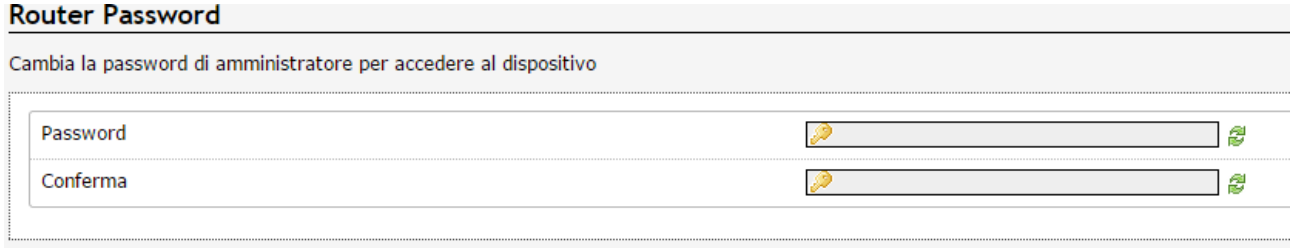
Una volta fatto ciò basterà attendere qualche minuto e digitare l'indirizzo 192.168.1.1 per accedere nuovamente all'Access Point con il nuovo firmware. Dopo di che verrà presentata la seguente schermata di accesso



CONFIGURAZIONE NUOVO FIRMWARE

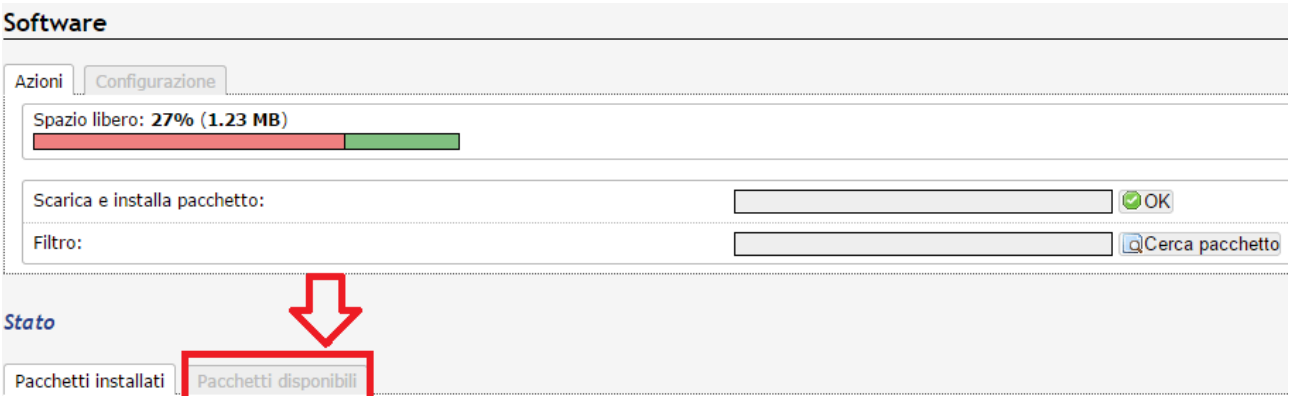
Una volta che è stato installato OpenWrt sarà necessario configurare dei parametri necessari al suo corretto funzionamento come ad esempio la password di accesso, le regole del firewall, il dhcp per le interfaccie, il wifi ecc.

Per prima cosa è necessario impostare la password per l'accesso; per farlo basterà andare nella sezione dedicata ed inserirla nel campo dedicato come illustrato in questa immagine

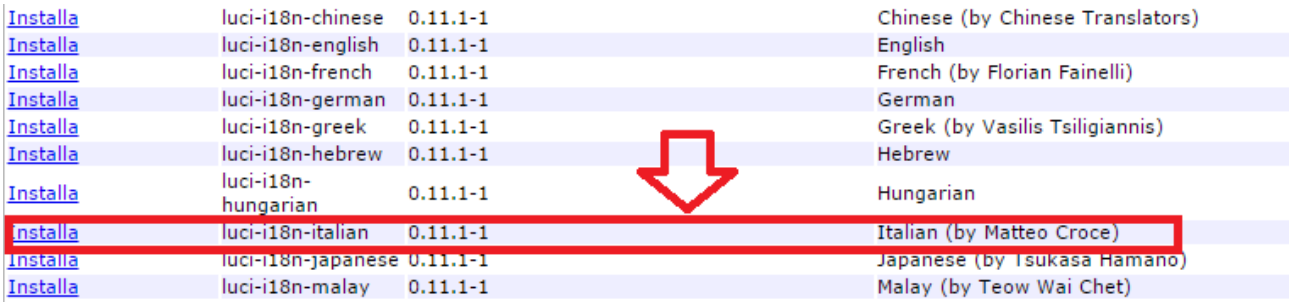


Una volta impostata la password si procede con l'installazione e la configurazione dei pacchetti base necessari per poter ultimare la configurazione.

Si procede con l'installazione del pacchetto di traduzione alla lingua italiana delle varie sezioni così da rendere più semplice la configurazione. Come tutti pacchetti basterà andare nella sezione Sistema→Software, cliccare su "update list", poi andare alla sezione dei pacchetti disponibili.

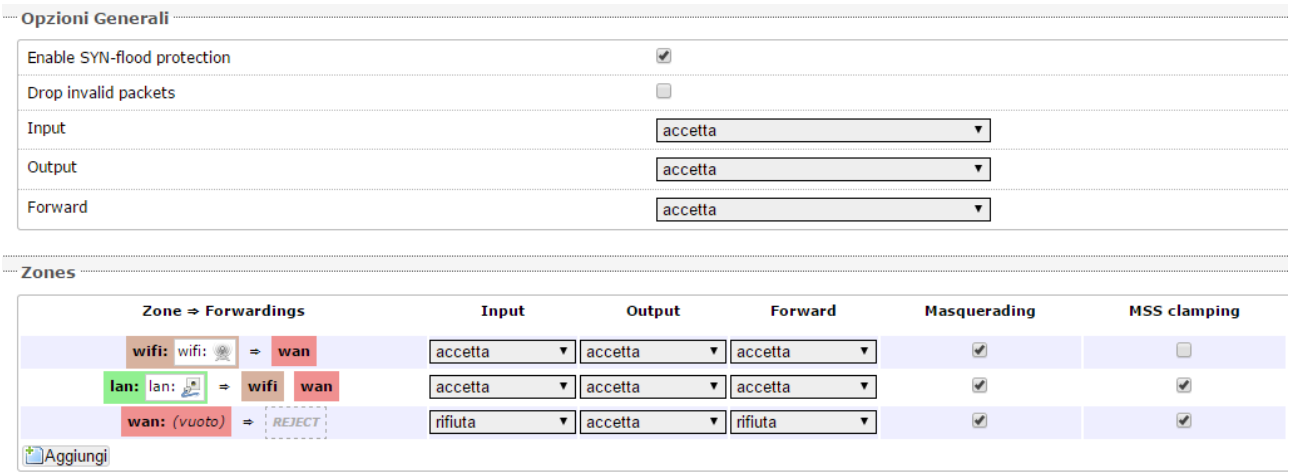


Il pacchetto che ci interessa è denominato “luci-i18n-italian”. Una volta trovato basterà cliccare su “Install” come illustrato nella seguente immagine



Dopo l’installazione si passa alla configurazione del firewall. Dato che per accedere alla rete wifi il client deve essere rimandato ad una pagina Captive Portal locata in un web server nella stessa rete dell’Access Point, sarà necessario abilitare il forwarding.

Per farlo basterà andare nella sezione Rete→Firewall ed abilitare la regola impostando su “ACCEPT” in modo da ottenere la seguente configurazione:

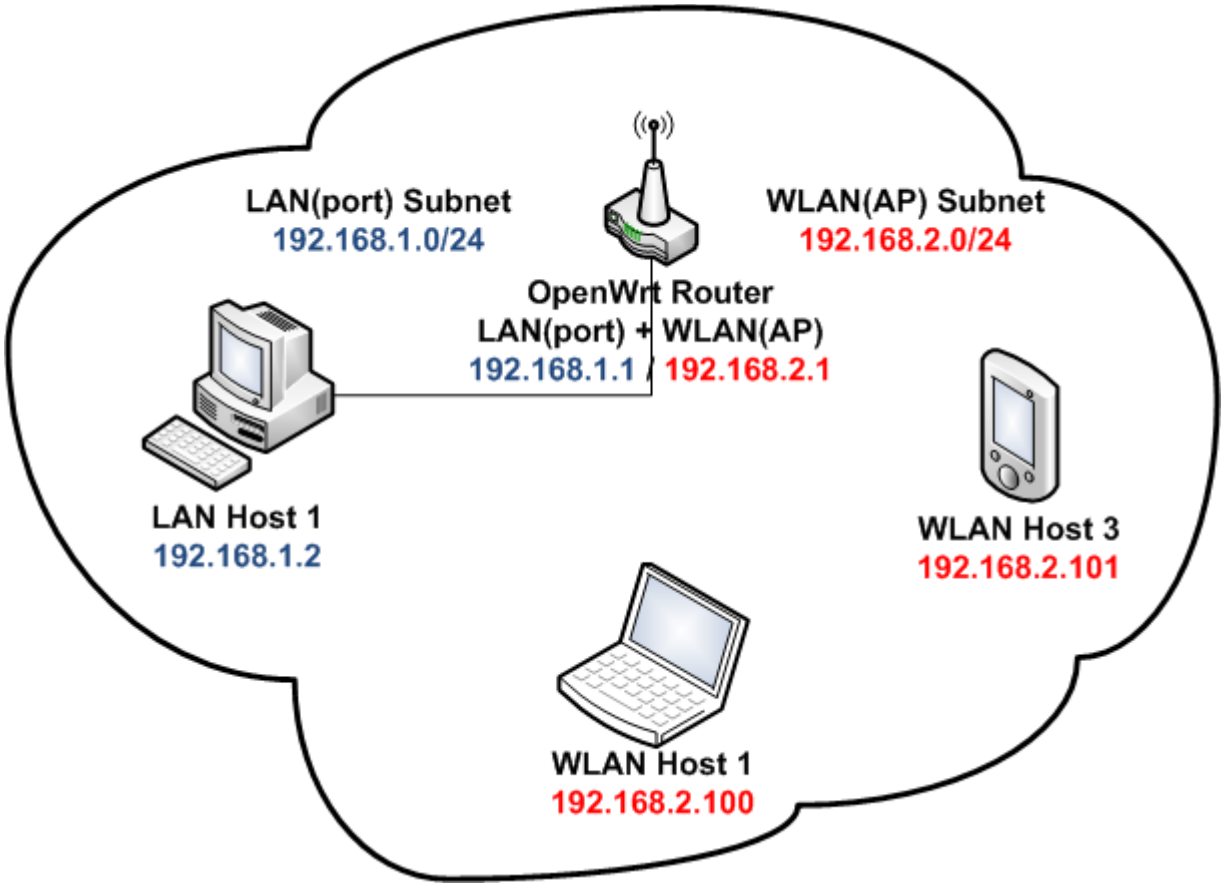


CREAZIONE RETE WIFI SEPARATA

Per la configurazione era stato richiesto di creare una rete WiFi separata dalla Lan dell'Access Point, in modo che ai client collegati venisse assegnato un indirizzo IP di una rete privata differente da quella in cui siano collegati i dispositivi. E' sufficiente seguire la guida fornita dalla pagina ufficiale del progetto OpenWrt al seguenti link:

http://wiki.openwrt.org/doc/recipes/routedap

Per comprendere ciò che si è deciso di creare è sufficiente vedere l'immagine qui sotto riportata:



Per poter apportare le modifiche richieste è necessario un collegamento tramite SSH al dispositivo e anche un editor di testo per la shell Linux. Se si sta usando una macchina con sistema operativo Windows basterà fornirsi di un programma che possa fare da client SSH come Putty o F-Secure SSH client, digitare l'indirizzo IP dell'Access Point e l'username utilizzato per accedere con relativa password mentre per Linux basterà digitare il comando *ssh + indirizzo_ip_del_dispositivo*. In OpenWrt è già presente di default l'editor di testo "vi" ,ma data la sua grande personalizzazione, è anche possibile installarne altri come Nano, Kwrite, Joe ecc.

Nel caso in cui non si conosca alla perfezione i comandi di “vi” e si voglia avere un edito semplice ma completo basterà digitare i seguenti comandi per installare “nano”:

 # opkg install nano

Per poter modificare i file da shell di comando basterà digitare il comando *nano* seguito dal percorso del file interessato. A questo punto si procede con l’elencare i file di configurazione e il relativo percorso:

- */etc/config/network*
- */etc/config/wireless*
- */etc/config/dhcp*
- */etc/config/firewall*

Per prima si procede con la modifica del file *network* sarà necessario creare una nuova interfaccia che si chiamerà *wifi* e avrà i seguenti parametri:

config 'interface' 'wifi'

option 'proto' 'static'

option 'ipaddr' '192.168.2.1'

option 'netmask' '255.255.255.0'

Ora che è stata creata la nuova interfaccia dedicata al WiFi è necessario eliminare la voce *bridged* nella sezione dedicata alla Lan così da avere due interfacce indipendenti l’una dall’altra. Successivamente si passa al file wireless dove si dovrà modificare la già esistente sezione *wifi-iface* cambiando le opzioni *network* con la connessione dall’interfaccia che è stata precedentemente, in modo da ottenere la seguente sezione del file:
config 'wifi-iface'

option 'device' 'wl0'

option 'network' 'wifi'

option 'mode' 'ap'

option 'ssid' 'OpenWrt'

option 'encryption' 'none'

Opzionalmente è anche possibile settare la sicurezza della rete a proprio piacimento inserendo il parametro nella sezione *encryption*. Ora che il WiFi è stato staccato dal bridge con la Lan la nuova rete è sprovvista di un proprio DHCP e per risolvere tale problematica sarà necessario affidargliene uno tramite la modifica del file di configurazione "*dhcp*" inserendo i seguenti parametri

config 'dhcp' 'wifi'

option 'interface' 'wifi'

option 'start' '100'

option 'limit' '150'

option 'leasetime' '12h'

Dato che di default in OpenWrt il traffico dati generato dalla rete wireless non è abilitato a raggiungere la WAN o l'interfaccia LAN sarà necessario applicare delle modifiche al firewall e alle sue "zones". Si procede alla modifica del file di configurazione del firewall e si aggiunge la sezione dedicata alla nuova interfaccia:

config zone

option name wifi

list network 'wifi'

option input ACCEPT

option output ACCEPT

option forward REJECT

Ora che la zone è stata configurata è possibile implementare il controllo di forwarding del traffico dati.

Per permettere ai client Wireless di usare l'interfaccia WAN va inserita la seguente sezione di *forwarding*:

config 'forwarding'

option 'src' 'wifi'

option 'dest' 'wan'

Se i client LAN debbano essere abilitati a contattare i client wireless vanno aggiunte le seguenti righe:

config 'forwarding'

option 'src' 'lan'

option 'dest' 'wifi'

Per permettere ai client Wireless di raggiungere la rete LAN è necessario aggiungere la seguente regola:

config 'forwarding'

option 'src' 'wifi'

option 'dest' 'lan'

Potrebbe capitare la situazione in cui non sia possibile ai client Wireless di connettersi alle rete Internet esterna; per ovviare a tale problematica bisogna abilitare il NAT sulla propria LAN aggiungendo la seguente riga di testo: "*option masq 1*" alla propria zone LAN così da avere una configurazione di questo tipo:

config zone

option name 'lan'

option network 'lan'

option input 'ACCEPT'

option output 'ACCEPT'

option forward 'REJECT'

option masq '1'

Ora che sono state apportate le dovute modifiche manca solo applicarle. Per farlo basterà eseguire i comandi

 # ifup wifi

 # wifi

per abilitare la nuova rete wireless.

Infine si riavvia il servizio del firewall
 # /etc/init.d/firewall restart e quello del DHCP
 # /etc/init.d/dhcp restart.

Qualora i client Wireless continuano a non essere abilitati alla connessione alla rete Internet, basterà aggiungere la seguente opzione: "*option mtu_fix 1*'' alla zone dedicata al LAN.

INSTALLAZIONE PACCHETTO PER IL CAPTIVE PORTAL

Per permettere l'autentificazione tramite Captive Portal dei client che si vorrebbero collegare, è necessario installare un pacchetto aggiuntivo presente nella lista dei software disponibili. Esistono vari pacchetti che permettono la gestione di un Captive Portal, tra questi vi sono Coovachilli ,Chillispot, Wifidog, NoDogSplash, NoCatSPlash, NoCatAuth ecc… .

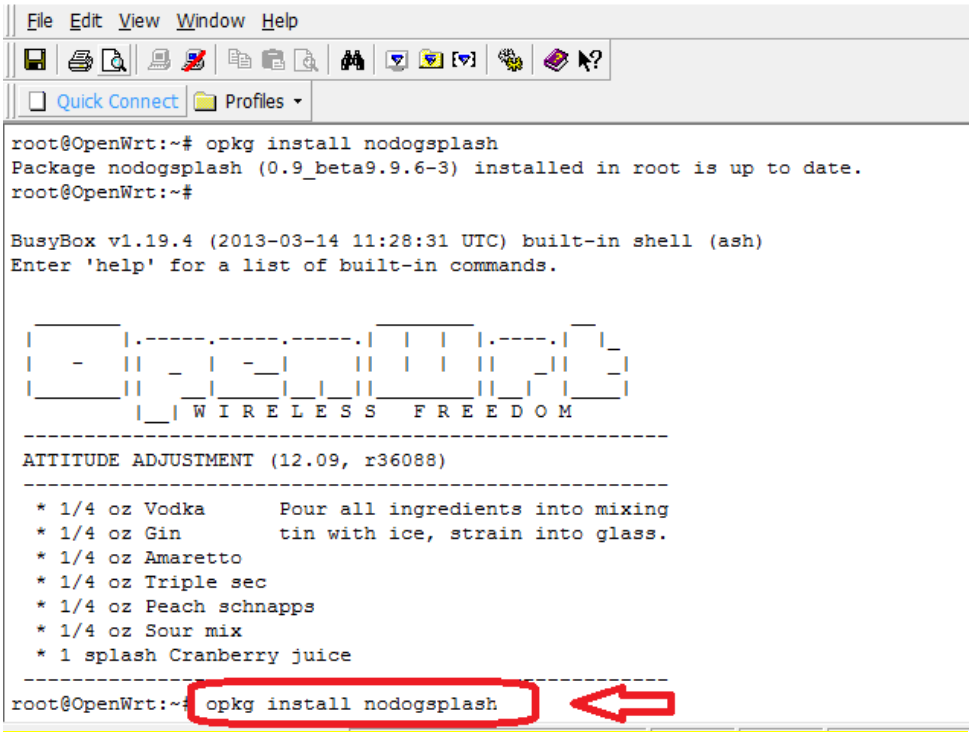
Per poter decidere quale usare bisogna valutare la tipologia di Captive Portal che si voglia implementare. Se si vuole gestire l'autentificazione con un server Radius/Ldap collegato si deve scegliere Coovachilli o Chillispot, mentre se si desidera avere una semplice pagina html splash, dove l'utente o immette semplicemente il proprio indirizzo e-mail o accetta delle particolari regole poste nella pagina, si devono scegliere pacchetti come NoDogSplash o NoCatSplash.

Dato che la pagina Captive Portal, che è stata implementata per la configurazione, gestisce già il controllo su un server Radius/Ldap, non è necessario affidarsi a pacchetti come Coovachilli o Chillispot.

Per la configurazione si è deciso di affidarsi al pacchetto "NoDogSplash" dato che la pagina Captive Portal si connette già di perse al server Radius/Ldap, perciò è solamente necessario avere un redirect a quest'ultima. Per poter fare ciò basterà cercare tale pacchetto nella lista di quelli disponibili all'installazione e cliccare su "Install". Se non si vuole usufruire dell'interfaccia web è anche possibile utilizzare la command line tramite collegamento ssh.

Tutto ciò sarà possibile tramite l'utilizzo di un qualsiasi programma che faccia da client ssh su Windows, ad esempio F-secure, oppure tramite shell digitando il comando *ssh indirizzo_ip_del_dispositivo* su sistemi operativi con architettura Linux.

Installa	nmap	6.01-4	Utility for network explorati
Installa	nmap-ssl	6.01-4	Nmap (with OpenSSL suppo
Installa	nocatauth	nightly-2	NoCatAuth is the original "c clients on your network, as
Installa	nocatsplash	0.93pre2-2	NoCatSplash is an Open Pul gateway/router on a networ gateway daemon then chan address).
Installa	nodogsplash	0.9_beta9.9.6-3	Nodogsplash offers a simple use on wireless access poin
Installa	noping	1.6.2-1	Ncurses application to send
Installa	nping	6.01-4	Network packet generation
Installa	nprobe	4.1-2	nprobe
			The NRPE addon is designe



Una volta installato il pacchetto verrà creata una cartella al seguente percorso /etc/nodogsplash contenente il file di configurazione del pacchetto (nodogsplash.conf) e un'altra cartella contenente il file nascosto html della pagina splash.

Per modificare i valori del file basterà utilizzare l'editor di testo *nano* o *vi* a seconda delle proprie necessità. Il pacchetto distingue in due categorie gli utenti: autenticati e in attesa di autentificazione. Gli utenti autentificati vengono definiti tali perché sono passati attraverso la splash page che, nel caso di questa configurazione, rimanda alla pagina Captive Portal collegata al server Radius/Ldap.

Le potenzialità del pacchetto non si limitano al semplice redirect del client, dato che è in grado, tramite opportune modifiche del file di configurazione, di limitare le porte per gli utenti autentificati o in attesa, limitare il flusso di rete ed impostare un eventuale redirect ad una pagina web dopo che l'utente si sia autentificato. Per una corretta configurazione di NoDogSplash si è deciso di seguire la guida dettagliata presente nella wifi del sito ufficiale di OpenWrt raggiungibile al seguente link:

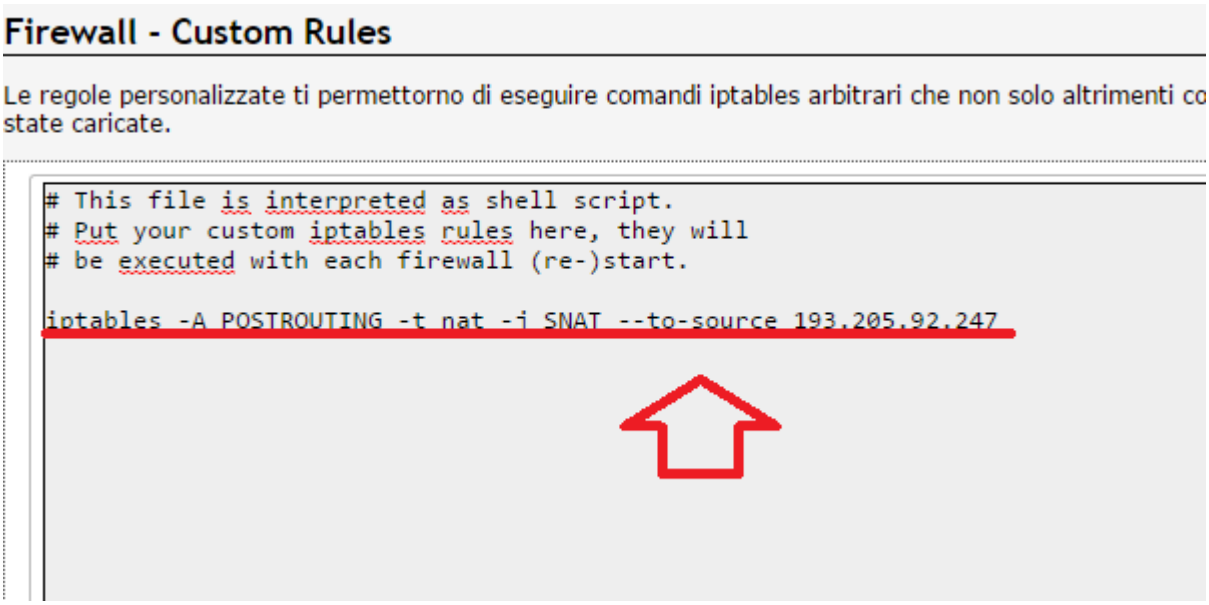
http://wiki.openwrt.org/doc/howto/wireless.hotspot.nodogsplash,

con l'aggiunta delle opportune modifiche a seconda delle proprie necessità.

Dopo aver installato il pacchetto sarà necessario aggiungere una comando personalizzato da eseguire nel firewall di OpenWrt; per farlo basterà andare nella sezione Rete→Firewall→Regole Personalizzate ad aggiungere la seguente istruzione:

iptables -A POSTROUTING -t nat -j SNAT --to-source <indirizzo_ip_statico_dell'interfaccia ethernet>

così da ottenere il seguente risultato:



Ora che si è impostata la regola custom essa verrà eseguita ad ogni reboot del dispositivo portando il traffico rete al seguente indirizzo e facendo uscire i client con un loro proprio indirizzo ip globale e non uno fisso locale.

Si procede con la creazione di una rete Wireless libera ma su una sottorete differente a quella della lan e di eventuali altre reti WiFi che siano state create; dato che è già stato fatto precedentemente sarà omessa la ripetizione del procedimento.

Si passa alla modifica del file di configurazione cambiando la riga di codice inerente la GatewayInterface su cui andrà a lavorare il pacchetto in questo modo:

GatewayInterface wlan0

Per poi aggiungere, tra le porte abilitate per gli utenti in attesa di autentificazione, l'abilitazione di connessione, tramite la porta 443, al Web server esterno su cui risiede la pagina Captive Portal con la seguente dicitura:

FirewallRule allow tcp port 443 to <indirizzo_ip_Web_Server>

Ora sarà possibile indirizzare l'utente alla pagina così da far gestire al server Radius le autentificazioni. Si conclude con la modifica della pagina splash.html contenuta nel percorso /etc/nodogsplash/htdocs inserendo la seguente stringa di codice che permette il redirect:

<meta http-equiv="refresh" content="0;

URL=https://indirizzo_ip_Web_server/percorso_pagina_Captive_Portal">

Una qualsiasi utente che si vuole connettere verrà istantaneamente reindirizzato alla pagina che più si preferisce per gestire l'autentificazione dei client. Infine si abilita l'esecuzione del pacchetto e si riavvia affinchè rilevi le modifiche apportate al suo file di configurazione:

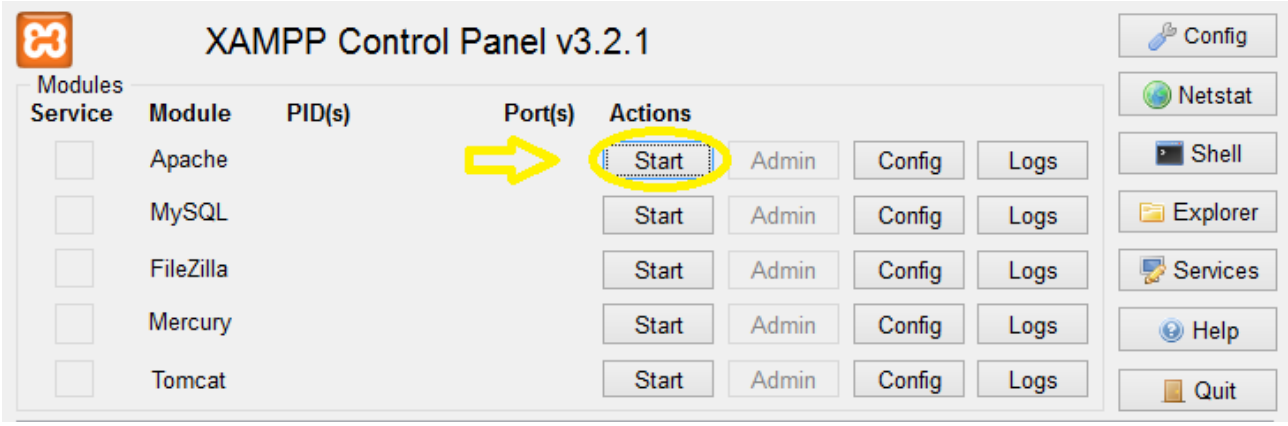
 # /etc/init.d/nodogsplash enable

 # /etc/init.d/nodogsplash restart

INSTALLAZIONE WEB SERVER (APACHE CON XAMPP)

Per poter autentificare i client con un Captive Portal è necessario avere un Server Web che possa contenere tale pagina.

Nella progettazione si è scelto di usare la piattaforma di controllo server Xampp con la quale è possibile gestire Apache, Tomcat, Mysql, Php, Filezilla. Per poter avviare il Server Web Apache basterà avere libere le porte 80 e 443 e premere sul pulsante "start" come in figura



Viene testato il funzionamento del server digitando nella barra degli indirizzi del browser "*http://localhost/*" per essere ridiretti al Server Web locale . Poiché ci sono scambi di credenziali è necessario abilitare la connessione protetta tramite SSL sulla porta 443. modificando il file di configurazione di Apache "httpd.conf" ed aggiungere la richiesta di connessione tramite SSL.

<Directory />

AllowOverride none

Require all denied

SSlRequireSSL

</Directory>

Dopo di che si passa alla modifica del file “httpd-xampp.conf”, in cui verranno inseriti i parametri che permettano la connesione SSL.

<IfModule mod_rewrite.c>

RewriteEngine On

 # Redirect /xampp folder to https

RewriteCond %{HTTPS} !=on

RewriteCond %{REQUEST_URI} xampp

RewriteRule ^(.) https://%{SERVER_NAME}$1 [R,L]*

 # Redirect /phpMyAdmin folder to https

RewriteCond %{HTTPS} !=on

RewriteCond %{REQUEST_URI} phpmyadmin

RewriteRule ^(.) https://%{SERVER_NAME}$1 [R,L]*

 # Redirect /security folder to https

RewriteCond %{HTTPS} !=on

RewriteCond %{REQUEST_URI} security

RewriteRule ^(.) https://%{SERVER_NAME}$1 [R,L]*

 # Redirect /webalizer folder to https

RewriteCond %{HTTPS} !=on

RewriteCond %{REQUEST_URI} webalizer

RewriteRule ^(.) https://%{SERVER_NAME}$1 [R,L]*

</IfModule>

In alcuni casi potrebbe non esserci il file di configurazione nella cartella di Apache di Xampp, denominato “openssl.cnf” che permetta il corretto funzionamento di OpenSSL. Si può scaricarlo dalla cartella del progetto OpenSSL situata su “SorceForge”.

Configurato correttamente il Web Server locale, sarà necessario creare dei certificati self signed per richiedere la connessione https, in modo che possa avvenire la connessione SSL.

Si apre una scheda di prompt dei comandi, ci si sposta nella cartella di configurazione di apache tramite il comando "*cd C://xampp/apache/*" e si digita "*makecert*"; quest'ultimo comando farà apparire una schermata di immissione dati riguardanti la società che rilascia il certificato.

Si inizia creando una password per il certificato, si passa allo stato in cui è localizzata la società, la regione, il nome della società per intero ed il nome dell'ufficio. In quest'ultimo dovrà essere inserito o l'indirizzo IP statico assegnato alla macchina in cui il Web Server è presente o la dicitura "localhost" così che, anche se si cambia indirizzo IP della macchina, il certificato resti funzionante.

Le altre informazioni non sono importanti per la configurazione che si andrà a creare, perciò basterà lasciare i campi vuoti e non si verrà a creare alcun problema. Un problema a cui non si potrà risolvere, a meno che si disponga di certificati rilasciati da una società certificata, è quello del mancato riconoscimento della "società" che rilascia i certificati.

Ciò comporterà, ogni volta che avverrà il redirect al Web Server, che il browser avvertirà l'utente che la pagina che si sta visualizzando non è sicura e verrà chiesta una conferma se si voglia proseguire la navigazione o se si voglia abbandonare la pagina.

CREAZIONE CAPTIVE PORTAL IN PHP

Una volta che sono stati configurati i vari server Radius e Ldap, resta soltanto creare una pagina web che funga da Captive Portal in cui l'utente immetta le proprie credenziali.

Ci sono vari linguaggi come ad esempio Asp.net, Aspx, Dot.net ed infine PHP. La scelta è caduta su PHP dato che è più dinamico ed ha le librerie Ldap integrate. Per poter attivare le librerie è necessario, oltre ad aver installato nel proprio pc PHP, modificare il file .ini presente nella cartella che si viene a creare dopo l'installazione rimuovendo la spunta di commento sulle librerie Ldap.

In alcuni casi potrebbe dare errore una volta che si tenta di compilare una pagina PHP con un qualsiasi compilatore, ad esempio Netbeans.

Per evitare il problema sarà necessario cambiare il percorso che assume PHP nelle variabili d'ambiente del PC.

In questa immagine possiamo notare come venga richiamato il server Ldap, quale porta usare per la connessione e anche l'inizializzazione delle variabili "auth_user" e "auth_pass", le quali corrisponderanno alle credenziali che immetterà l'utente.

```
if($_POST){
$ldap['user']   = $_POST['auth_user'];
$ldap['pass']   = $_POST['auth_pass'];
$ldap['host']   = '192.168.1.40';
$ldap['port']   = 389;
```

In questo Screenshot è racchiuso il cuore della nostra pagina Captive Portal. Si inizia con l'inizializzazione della connessione al server mediante la funzione ldap['conn'], dove viene eseguito un controllo sull'indirizzo IP del server e sulla porta scelta per la connessione. Se il controllo ha esito positivo si passa al bind della connessione dove vengono controllate le credenziali immesse dall'utente. Naturalmente, alla prima esecuzione, non ci sono ancora dati quindi il bind non viene effettuato ma viene semplicemente presentata la seguente pagina per l'immissione delle credenziali.

```
    <?php
if($_POST){
    $ldap['user']   = $_POST['auth_user'];
    $ldap['pass']   = $_POST['auth_pass'];
    $ldap['host']   = '192.168.1.22';
    $ldap['port']   = 389;
    $ldap['conn'] = ldap_connect($ldap['host'], $ldap['port'])
    or die("Could not conenct to {$ldap['host']}" );
    $ldap['bind'] = ldap_bind($ldap['conn'], $ldap['user'], $ldap['pass']);

    //--------------------
    if( !$ldap['bind'] )
    {
    //echo 'Login Fallito';
    echo ldap_error( $ldap['conn'] );
    exit;
    }
    //---------------------------
    elseif($ldap['bind'])
    {
    ldap_close($ldap['conn']);
    header("Location: http://www.google.it");
    }
}
else
{

echo '
```

Nella seguente immagine si può avere un'idea di come sia strutturata a livello grafica la pagina Captive Portal

Questa pagina non è una semplice pagina scritta in Html, ma è composta anche da codice Javascript che ci permette di fare un controllo dei form di inserimento dello Username e della password utente. Dall' immagine successiva è possibile notare l'applicazione della funzione "Controlla Form" per impedire che, nel caso ci si voglia connettere, non sia possibile lasciare i form vuoti:

```
    <script language = "Javascript">
function ControllaForm()
{
    var controllo =false;
    if (document.form1.auth_user.value == "")
    {
        controllo = true;
        alert("Attenzione, non è stato inserito alcun utente");
    }
    if (document.form1.auth_pass.value == "")
    {
        controllo = true;
        alert("Attenzione, non è stata inserita una password");
    }
}
    </script>
```

Il tutto è reso possibile anche tramite la comparsa di due finestre che ci avvertono che abbiamo lasciato dei campi vuoti:

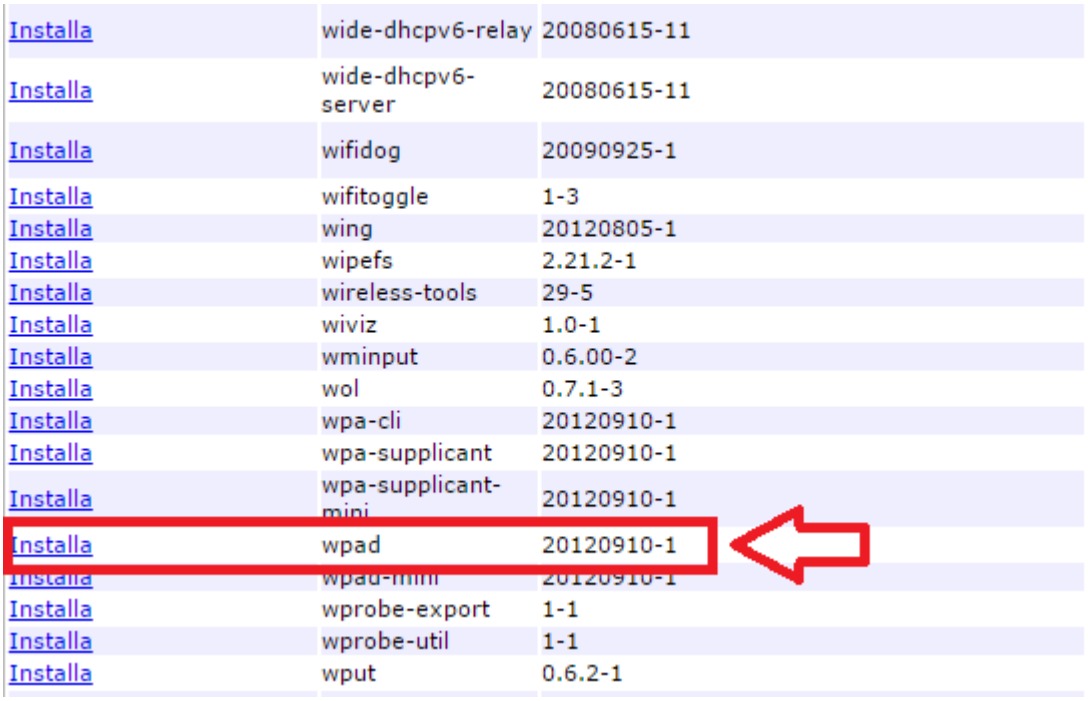
Se non si trascrivono le credenziali si verrà continuamente rimandati alla pagina di compilazione dei form e non si avrà l'accesso per la connessione WiFi. Dopo aver immesso le credenziali inizierà un controllo sul server Radius/Ldap per determinare se i dati immessi siano presenti; se le credenziali sono corrette l'utente verrà reindirizzato ad una pagina, in questo caso a Google.com.

SECONDA CONFIGURAZIONE

Per questa configurazione è stato usato OpenWrt come nella precedente ma questa volta sono stati utilizzati differenti pacchetti.

INSTALLAZIONE WPAD

Il firmware OpenWrt non dispone della autentificazione tramite Wpa2 EAP di default, perciò è stato necessario installare il pacchetto “Wpad” sostituendo il precedente “Wpad-mini”. Esistono due possibili procedure per poter installare il pacchetto, la prima consiste nel collegarsi all’interfaccia grafica di Openwrt e nella sezione Sistema-->Software, aggiornare l’elenco dei pacchetti disponibili cliccando su “aggiorna lista software”, cliccare sulla sezione “sofitware disponibili”, cercare sotto la lettera w e cliccare cliccare su “install” come illustrato nella seguente immagine:



Installa	wide-dhcpv6-relay	20080615-11
Installa	wide-dhcpv6-server	20080615-11
Installa	wifidog	20090925-1
Installa	wifitoggle	1-3
Installa	wing	20120805-1
Installa	wipefs	2.21.2-1
Installa	wireless-tools	29-5
Installa	wiviz	1.0-1
Installa	wminput	0.6.00-2
Installa	wol	0.7.1-3
Installa	wpa-cli	20120910-1
Installa	wpa-supplicant	20120910-1
Installa	wpa-supplicant-mini	20120910-1
Installa	wpad	20120910-1
Installa	wprobe-export	1-1
Installa	wprobe-util	1-1
Installa	wput	0.6.2-1

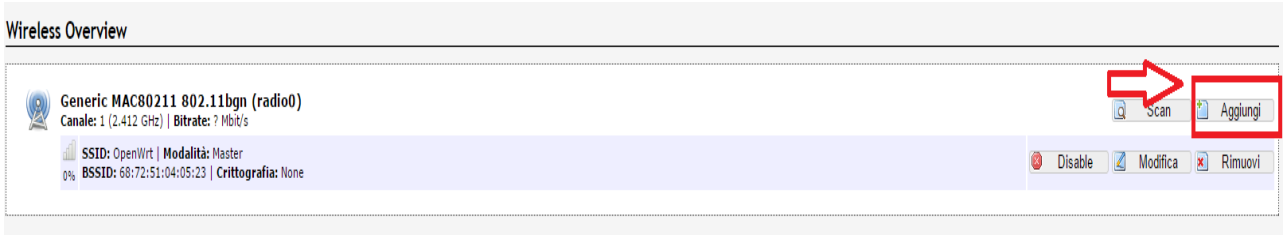
Una volta installato e configurato il pacchetto , occorre solamente rimuovere il vecchio pacchetto scorrendo la lista dei pacchetti installati e cliccare su “Remove” come illustrato in figura:

Il secondo metodo prevede invece il collegamento tramite SSH all’Access Point ed una volta collegato digitare i seguente comandi:

 # opkg remove wpad-mini

 # opkg install wpad.

Ad installazione avvenuta si passa alla sezione Rete→Wifi e si aggiunge una nuova rete



Ora che sono state create le reti Wireless basterà impostare la crittografia su WPA2 ed inserire nei campi richiesti l’indirizzo ip del server Radius/Ldap, la relativa password e la porta utilizzata tralasciando la sezione inerente l’accounting.

TERZA CONFIGURAZIONE

INSTALLAZIONE PARTIZIONE UBUNUTU 14.04 LTS

Per poter installare il Service Provider si è deciso di lavorare su di una macchina dove era stato precedentemente installato un sistema operativo basato su kernel Linux, in particolare Ubuntu 14.04 LTS.

Tale scelta è giustificata dal fatto che la versione del Service Provider per Linux è più aggiornata e seguita dalla comunity rispetto a quella per altri sistemi operativi come Windows o Mac.

Per poter installare il sistema operativo si è ricorsi ad una installazione mediante un device usb in cui, tramite il software "Universal Usb Installer", è stato creato un installer. Modificando la priorità di avvio del boot, installerà Ubuntu in una partizione del disco senza sovrascrivere il sistema operativo principale.

INSTALLAZIONE SP DI SHIBBOLETH

La parte pratica svolta in questo lavoro consiste nella realizzazione di un sistema per la "Shibbolettizzazione" del servizio, sfruttando il servizio di IDP già attivo nell'Università. L'obbiettivo principale della configurazione è stato quello di indirizzare l'utente alla pagina Captive Portal con autentificazione Shibboleth per poter accedere al WiFi Unicam.

Avere un servizio di Identity Provider già attivo nell'Università è stato un grande vantaggio ai fini dello svolgimento del progetto, poiché non ci si è dovuti occupare di realizzarne uno per intero.

Questa fase di lavoro è stata dunque molto semplificata, e si è data maggiore importanza al come proteggere la risorsa e a quali scelte compiere per configurare adeguatamente la fase di autenticazione e lo scambio di attributi.

Per poter eseguire al meglio la configurazione ci si è basati su di una guida fornita dal GARR che è possibile scaricare in formato pdf al seguente link:

"*https://www.idem.garr.it/documenti/doc_view/313-installazione-shibboleth-service-provider-su-debian-linux*"

Per prima cosa si installano i seguenti software:

- Openssl
- Ntp
- Apache2
- Libapache2-mod-shib2
- Php5

Per far ciò basterà digitare su terminale il seguente comando con permessi di root:

 # apt-get install Openssl Ntp Apache2 libapache2-mod-shib2 php5 php-commons.

Si testa il corretto funzionamento del Web Server Apache digitando nella barra degli indirizzi del browser *http://localhost* ed attendere la comparsa di una pagina in cui è scritto: "It Works!". Si passa all'installazione vera e propria del Service Provider.

Si inizia creando una cartella *secure* con i permessi di root mediante la digitazione dei seguenti comandi:

- *# Cd /var/www/html*
- *# Mkdir secure*

Dopo di che si crea un file index.php

 # sudo gedit index.php

composto del seguente testo:

```
<?php
foreach($_SERVER as $key_name => $key_value) {
print "<BR>" . $key_name . " = " . $key_value . "
";
}
?>
```

Il quale, in caso di funzionamento della configurazione, restituirà i parametri del client che si sta connettendo e quale Service Provider stia usando.

Si passa ora alla modifica del file */etc/apache2/sites-available/default-ssl* aggiungendo i seguenti parametri prima del *"</VirtualHost>":*

```
<Location /secure>
AuthType shibboleth
ShibRequireSession On
require valid-user
```

</Location>

Si procede con l'attivazione del modulo shib2 di Apache tramite i seguenti comandi:

- *# a2enmod shib2*
- *# service apache2 restart*

COLLEGAMENTO IDP UNICAM

Per poter collegare il SP installato sulla mia macchina è stato necessario modificare il file di configurazione di Shibboleth contenuto nella cartella */etc/shibboleth/* tramite il comando

 # sudo nano /etc/shibboleth/example-shibboleth2.xml.

Il seguente non è altro che un file di esempio per poter creare al meglio la configurazione finale di Shibboleth. In esso sarà solamente necessario cambiare gli url inerenti l'entityID

https://idp.cs.unicam.it/idp/shibboleth,

che va inserito in ogni sezione che venga richiesto ed inserire la sezione inerente il Metadata Provider:

<MetadataProvider type="XML"
uri="https://idp.cs.unicam.it/idp/shibboleth"
reloadInterval="7200">
</MetadataProvider>

Dopo aver apportato le dovute richieste basterà rinominare il file in shibboleth2.xml e sostituire così il vecchio file di configurazione.

INTEGRAZIONE ACCESS POINT

Per poter integrare l'autentificazione tramite Shibboleth occorre solamente modificare i parametri inerenti la pagina di redirect inserita nella splash page del pacchetto NoDogSPlash.

L'indirizzo da inserire è quello che è stato precedentemente impostato al Service Provider di Shibboleth con la seguente dicitura finale: *https://indirizzo_ip_del_SP/secure/index.php*

L'indirizzo dirigerà i client alla pagina di Captive Portal del IDP Unicam dove dovranno inserire email unicam e password per poter accedere. Se l'autentificazione ha avuto successo avverrà il redirect finale che porterà alla pagina di test dove sono mostrate tutte le informazioni account del client che effettuato l'accesso.

Per poter permettere all’utente di navigare basterà inserire un semplice redirect al posto della pagina di test così che non noti il continuo passaggio da una pagina all’altra.

CONCLUSIONI E SVILUPPI FUTURI

Nel lavoro di tesi sono stati realizzati tre possibili configurazioni implementabili per l'autentificazione dei client nel WiFi Unicam. Sono stati utilizzati software basati sia su sistemi operativi della famiglia Windows, sia quelli basati su kernel Linux. Per poter contenere i costi di eventuali hardware aggiuntivi è stato virtualizzato il server Radius/Ldap, utilizzato per la prima e la seconda configurazione. In tutte le configurazioni sono stati impiegati gli Access Point forniti dall'Unicam con firmware OpenWRT. Si è scelto OpenWRT per potenziare ed ampliare le capacità degli Access Point perché il firmware stock non aveva a disposizione funzioni per l'autentificazione tramite Captive Portal, o il controllo della banda sulle varie reti Wireless. Per la terza configurazione è stato utilizzato il pacchetto Shibboleth che permette di avere un'autentificazione federata ed un controllo maggiore degli account collegati. Tramite questo pacchetto è possibile collegare alla propria rete account di diversi IDP, dando la possibilità a docenti o personale di altre facoltà di avere accesso alla rete Unicam. Tramite dei test è stato possibile implementare la prima e la seconda configurazione nello stesso Access Point in modo che un dispositivo possa essere a disposizione per eventuali future applicazioni.

Mi auguro che le configurazioni che sono state trattate e realizzate siano implementate in futuro ed estese all'interno di altre facoltà o ambienti lavorativi.

RINGRAZIAMENTI

Vorrei ringraziare innanzitutto il mio relatore Prof. Fausto Marcantoni ed il correlatore Prof. Francesco De Angelis, per la loro disponibilità e collaborazione nella realizzazione del progetto, per avermi guidato ed aiutato durante questo duro lavoro di tesi e per avermi permesso di lavorare ad un'idea così importante ed innovativa.

Ringrazio la mia famiglia per il supporto datomi e la pazienza avuta durante questi mesi in cui ho praticato continui test sfruttando la rete e quindi intasandola continuamente.

Ringrazio di cuore la mia ragazza che mi è stata vicino in questi ultimi mesi, dalle cui sorprendenti manifestazioni di affetto ho tratto la forza per superare i momenti più difficili, e ho ritrovato gli stimoli per dedicarmi a questa tesi di laurea.

Vorrei anche ringraziare i miei compagni di corso che in questi anni con la loro amicizia mi hanno fatto sempre sentire a mio agio e mi hanno sostenuto a prendere decisioni importanti e delicate sia per la mia carriera universitaria che nella vita privata.

Vorrei ringraziare anche i miei amici d'infanzia che grazie al loro sostegno mi hanno dato la forza necessaria per intraprendere questo progetto e completarlo.

BIBLIOGRAFIA E SITOGRAFIA

[1] John Hornor Jacobs, *The Shibboleth*, Carolrhoda Books

[2] Brian Arkills, *LDAP Directories Explained***,** Addison Wesly

[3] Lopez Daniel Blanco Jesus, *APACHE IN TASCA*, Pearson Education Italia

[4] https://idp.cs.unicam.it/idp/shibboleth,

[5] http://wiki.openwrt.org/doc/recipes/routedap

[6] http://wiki.openwrt.org/doc/howto/wireless.hotspot.nodogsplash

[7] http://wiki.openwrt.org/toh/start

[8] https://www.idem.garr.it/documenti/doc_view/313-installazione-shibboleth-service-provider-su-debian-linux

[9] https://wiki.shibboleth.net/confluence/display/SHIB2/Home

[10] http://www.internet2.edu/

[11] https://wiki.shibboleth.net/confluence/display/SHIB2/Home

[12] https://openwrt.org/