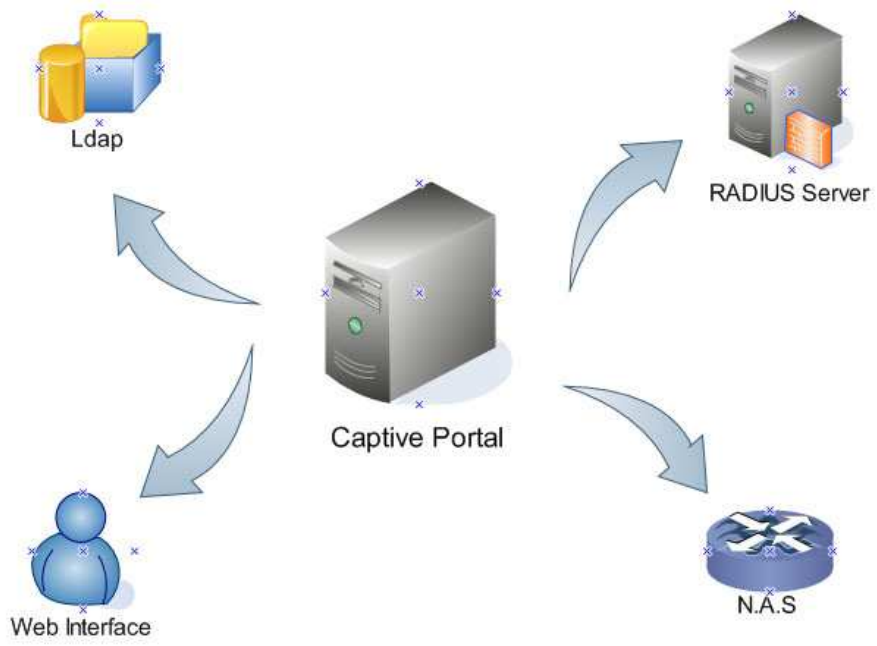
ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ

ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ



‘‘Υλοποίηση Captive Portal και Radius Server για εξουσιοδοτημένη πρόσβαση στο ασύρματο δίκτυο του Τ.Ε.Ι Κρήτης Παρατήματος Χανίων μέσω υπηρεσίας καταλόγου LDAP.’’

Κυρικάκης Ν. Στέφανος

Επιβλέπων Καθηγητής: Γεώργιος Σ. Λιοδάκης

Καθηγητής Εφαρμογών

Χανιά

Μάιος 2009

Πρόλογος

Στα σύγχρονα τηλεπικοινωνιακά δίκτυα η ανάγκη για ασφάλεια ολοένα και αυξάνεται. Αυτή η ανάγκη οδήγησε στην διεξαγωγή της συγκεκριμένης υλοποίησης στο δίκτυο του ΤΕΙ Κρήτης Παραρτήματος Χανίων και αφορά τους Χρήστες του ασύρματου δικτύου που συμπεριλαμβάνονται στον κατάλογο LDAP (Ελαφρύ Πρωτόκολλο Πρόσβασης Καταλόγου).

Όλες οι λεπτομέρειες της υλοποίησης που αφορούν το software που χρησιμοποιήθηκε (Coova-Chilli, Freeradius Server), ζητήματα παραμετροποίησης και προβλήματα που προέκυψαν κατά την δοκιμαστική περίοδο εξετάστηκαν και παρουσιάζονται. Αξίζει να αναφερθεί ότι η όλη υλοποίηση είναι σε χρήση στο δίκτυο του ΤΕΙ Κρήτης Παραρτήματος Χανίων και συνεισφέρει σημαντικά στην πολιτική ασφαλείας για την παροχή εξουσιοδοτημένης χρήσης του δικτύου σε όλα τα μέλη της ακαδημαϊκής κοινότητας.

ABSTRACT

Provision of Authenticated Network Services for WLAN users of TEI of Crete / Branch of Chania through a Captive Portal technique and Radius Implementation

A Network Access Control (NAS) strategy for remediation is that of captive portals. A captive portal technique forces an http client on a network to see a special web page (usually for authentication purposes) before using the internet normally.

Security concerns for WLAN users at the Technological Educational Institute of Crete / Branch of Chania who are included in the associated LDAP (Lightweight Directory Access Protocol) directory guided the conduction of the current thesis. Varius implementation details concerning the software used (Coova-Chilli, Freeradius Server) parametrications issues, problems faced during test period, etc, are examined and presented. It is to be mentioned that the overall implementation is in use at TEIoC / BoC and contributes significantly to the security policy for the provision of authenticated network services to all members of academic community.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 4 Υλοποίηση Captive Portal

ΚΕΦΑΛΑΙΟ 5 Εγκατάσταση και παραμετροποίηση Freeradius Server

ΚΕΦΑΛΑΙΟ 6

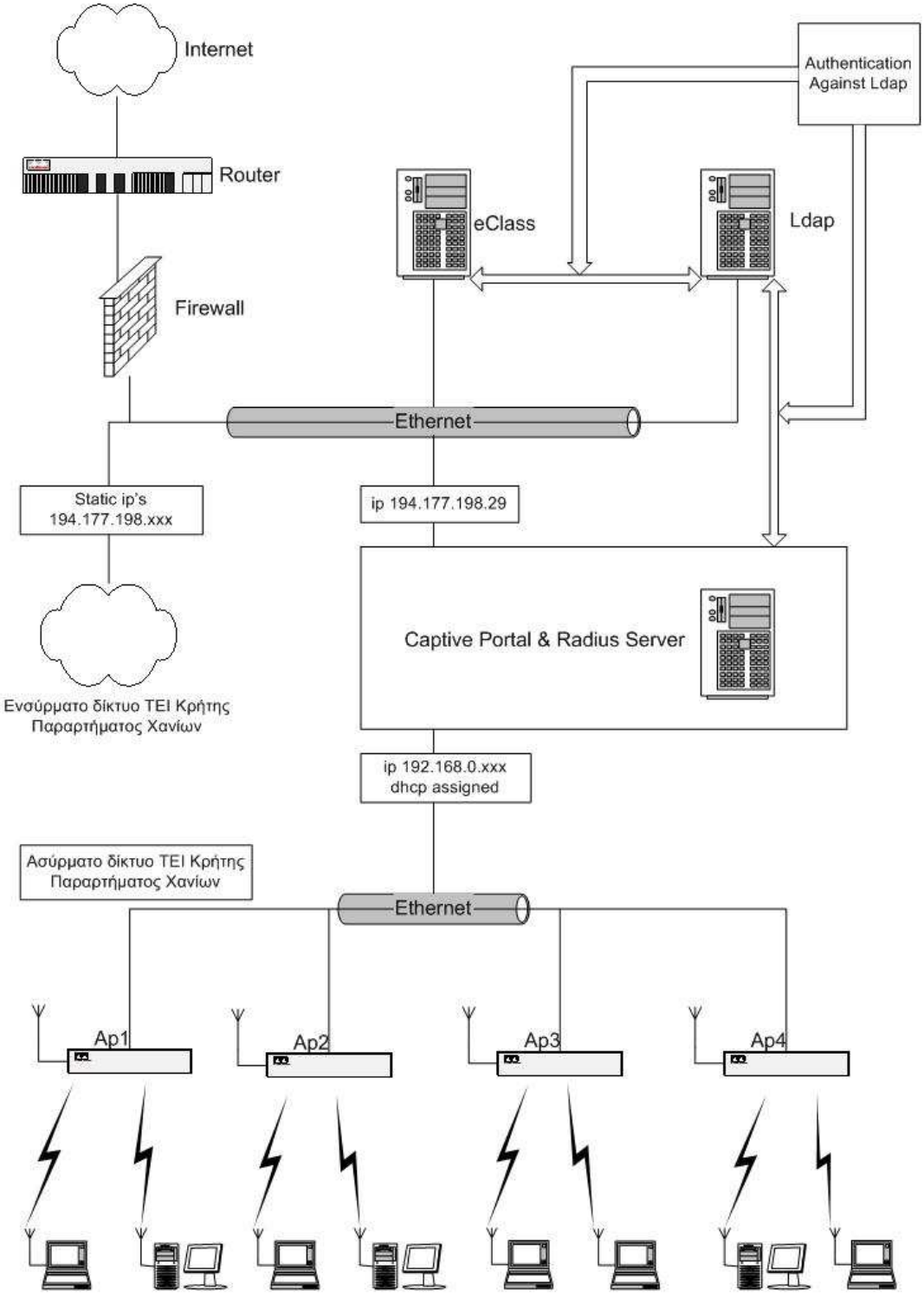
Έλεγχος λειτουργίας Προτάσεις - Συμπεράσματα για την υλοποίηση στο Τ.Ε.Ι Κρήτης Παράρτημα Χανίων

ΚΕΦΑΛΑΙΟ 1 Η Τεχνική Captive Portal

1.1 Τι είναι το Captive Portal

Η τεχνική του Captive Portal ή NAS [1] (διακομιστής πρόσβασης δικτύου) δεν κάνει χρήση των κοινών πρωτοκόλλων κρυπτογράφησης σε επίπεδο καναλιού, αλλά χρησιμοποιεί λύσεις λογισμικού που κρυπτογραφούν τα στοιχεία της σύνδεσης σε υψηλότερο δικτυακό επίπεδο. Παρεμβάλλεται μεταξύ του κοινόχρηστου δικτύου και του δικτύου που θέλουμε να περιορίσουμε την πρόσβαση και εξαναγκάζει το χρήστη να δει μία ειδική σελίδα πιστοποίησης στον web browser χωρίς να του επιτρέπει να έχει πρόσβαση στους πόρους του δικτύου μέχρι να πιστοποιηθεί (σχήμα 1.1).

Ένα captive portal, μεταμορφώνει τον web browser σε μια μηχανή πιστοποίησης. Αυτό γίνεται με το να παρεμποδίζει όλα τα πακέτα πληροφοριών ανάλογα με την διεύθυνση του χρήστη ή την πόρτα ωσότου ο χρήστης ανοίξει τον web browser και προσπαθήσει να μπει στο internet. Σε αυτό το σημείο ο browser ανακατευθύνεται από το captive portal σε μια σελίδα όπου μπορεί να του ζητηθεί η εισαγωγή ονόματος χρήστη και password (πιστοποίηση) ή και πληρωμή ή μπορεί απλά να του εμφανίζεται η πολιτική χρήσης και να απαιτεί από το χρήστη να συμφωνήσει με αυτήν. Η τεχνική captive portal χρησιμοποιείται κατά κόρον στα Wi-Fi hotspot, επίσης μπορεί να υλοποιηθεί ταυτόχρονα και για τον έλεγχο της ενσύρματης δικτύωσης πχ. σε ενοικιαζόμενα διαμερίσματα, δωμάτια ξενοδοχείων, επαγγελματικούς χώρους.


Internet
Router
Firewall
Authentication Against Ldap
eClass
Ldap
Ethernet
Static ip's 194.177.198.xxx
ip 194.177.198.29
Captive Portal & Radius Server
Ενσύρματο δίκτυο ΤΕΙ Κρήτης Παραρτήματος Χανίων
ip 192.168.0.xxx dhcp assigned
Ασύρματο δίκτυο ΤΕΙ Κρήτης Παραρτήματος Χανίων
Ethernet
Ap1
Ap2
Ap3
Ap4

Σχήμα 1.1

1.2 Τεχνικά χαρακτηριστικά

Οι εφαρμογές Captive Portal [5] ως client χρησιμοποιούν οποιοδήποτε από τους συνηθισμένους web browser, έτσι ώστε ο χρήστης να έχει την δυνατότητα να εισάγει τα στοιχεία του (username/pass) χωρίς να χρειαστεί να εγκαταστήσει κάποιο πρόγραμμα client στον υπολογιστή του, ανεξάρτητα από το λειτουργικό σύστημα που διαθέτει. Σαν δεύτερη χρήση, μπορούν να παρεμβάλουν κάποιες πληροφορίες πριν δώσουν πρόσβαση στον χρήστη (όπως π.χ. ότι πρέπει να συμφωνήσει σε μια αποδεκτή χρήση της υπηρεσίας).

Η χρήση των captive portals είναι ιδιαίτερα διαδεδομένη σε ανοιχτά δίκτυα που δεν έχουν ενεργοποιημένες άλλες μεθόδους πιστοποίησης (όπως WEP ή MAC filters). Σε μερικές περιπτώσεις, τα πρωτόκολλα για την προστασία του layer 2 επιπέδου σε ένα ασύρματο δίκτυο, όπως τα WPA και WPA2, δεν είναι κατάλληλα λόγω ότι δεν είναι εύκολο να ρυθμιστούν από τον τελικό χρήστη και λόγω ασυμβατότητας. Όπως για παράδειγμα, στην περίπτωση ενός μεγάλου ασύρματου δικτύου αποτελούμενο από πολλά access points προερχόμενα από διαφορετικούς κατασκευαστές με διαφορές στις προδιαγραφές και στα υποστηριζόμενα πρωτόκολλα. Σε άλλες περιπτώσεις, δεν χρειάζεται να περιοριστεί η πρόσβαση μόνο στο ασύρματο δίκτυο αλλά και στο ενσύρματο δίκτυο Ethernet.

Η λύση σε αυτά τα προβλήματα μπορεί να είναι η μετακίνηση του ελέγχου πρόσβασης από το OSI Layer2 στο Layer3 του δικτύου, χρησιμοποιώντας ένα Captive Portal. Με αυτή την τεχνική, ο χρήστης εισάγει τα πιστοποιητικά του (username, password) στον περιηγητή του ιστού για εξουσιοδοτημένη πρόσβαση στο δίκτυο.

Το Captive Portal [5] παρεμβάλλεται μεταξύ της κεντρικής πύλης (router) και του υποδικτύου στο οποίο θέλουμε να περιορίσουμε την πρόσβαση , έτσι λειτουργεί σαν κεντρική πύλη για το υποδίκτυο αναλαμβάνοντας να διανείμει εκ νέου δυναμικές IP διευθύνσεις στους χρήστες. Για να περιορίσει την πρόσβαση στο υποδίκτυο, μπλοκάρει όλα τα εξερχόμενα πακέτα IP που βλέπει. Μόνο όταν ο χρήστης κάνει ένα http ή https αίτημα στην πόρτα 80 και 443, ανακατευθύνοντας τον σε έναν web server (στην προκειμένη περίπτωση τον αποκαλούμε Authentication Server) προβάλλοντας στο χρήστη μια σελίδα πιστοποίησης επιτρέποντάς του να εισάγει τα πιστοποιητικά του. Αν ο χρήστης εισάγει τα σωστά πιστοποιητικά τότε το Captive Portal προωθεί τα εξερχόμενα πακέτα IP έξω από το προστατευόμενο υποδίκτυο.

Όλες οι αλληλεπιδράσεις μεταξύ του web browser του χρήστη και του Authentication Server που προαναφέραμε, είναι κρυπτογραφημένες για να αποφύγουμε την υποκλοπή τους στο δίκτυο. Οι χρήστες πιστοποιούνται μέσω της IP και της MAC διεύθυνσης τους. Ωστόσο, αυτές οι δύο παράμετροι είναι εύκολο να αλλαχτούν από κακόβουλους χρήστες. Έτσι, για να προληφθεί κάτι τέτοιο, απαιτείται από το web browser του χρήστη να έχει ένα authenticator, αυτό δεν είναι τίποτα άλλο από ένα αναδυόμενο παράθυρο του web browser του χρήστη που μένει ανοικτό ανανεώνοντας την σύνδεση στέλνοντας περιοδικά ένα κρυπτογραφημένο μήνυμα στον authentication server χωρίς να γίνεται αντιληπτό από τη μεριά του χρήστη. Επίσης, μέσω αυτού του παραθύρου, ο χρήστης μπορεί να διακόψει την σύνδεση με το Captive Portal.

1.3 Συνεργασία Captive portal και Radius server

Για την λειτουργία της πιστοποίησης (Authentication) σε ένα Captive Portal είναι απαραίτητη η χρήση ενός Radius Server [2] (διακομιστή πιστοποίησης). Ένας διακομιστής πιστοποίησης αναλαμβάνει την διαμεσολάβηση μεταξύ του Captive Portal και της υπηρεσίας καταλόγου LDAP όπου είναι αποθηκευμένα τα πιστοποιητικά των χρηστών.

Όταν ο χρήστης εισάγει τα πιστοποιητικά του (όνομα χρήστη, κωδικό) στην σελίδα πιστοποίησης του Captive Portal αυτά αποστέλλονται κρυπτογραφημένα στον διακομιστή πιστοποίησης (Radius Server). Τότε ο Radius ψάχνει σε μια βάση δεδομένων για το όνομα χρήστη (username) που υπάρχει στην αίτηση και αν το βρει επιστρέφει ένα αίτημα έγκρισης (Authorization) προς το Captive Portal.

Μια άλλη πολύ σημαντική λειτουργία του Radius Server είναι αυτή της παρακολούθησης (Accounting) του δικτύου. Όλα τα αιτήματα για σύνδεση (επιτυχημένα ή μη) καταγράφονται λεπτομερώς σε μια βάση δεδομένων έτσι ώστε να μπορεί να ελεγχθεί ανά πάσα στιγμή ο χρήστης για τυχόν παράβαση της πολιτικής ασφαλείας. Επίσης μετά το πέρας μιας επιτυχημένης σύνδεσης καταγράφεται στη βάση ο χρόνος και ο όγκος δεδομένων που έστειλε και έλαβε ένας χρήστης και επιτρέπει στους διαχειριστές του δικτύου να επιβάλουν περιορισμούς στη χρήση ή και χρεώσεις ανάλογα με το χρόνο χρήσης ή τον όγκο δεδομένων.

Ένα πλεονέκτημα του Radius Server είναι ότι μπορεί να εγκατασταθεί οπουδήποτε ακόμα και εκτός του τοπικού δικτύου, αυτή η υλοποίηση μας δίνει την δυνατότητα να έχουμε πολλά Captive Portal ελεγχόμενα κεντρικά από ένα και μοναδικό Radius Server. Αυτό προσφέρει μεγάλη ευελιξία στην

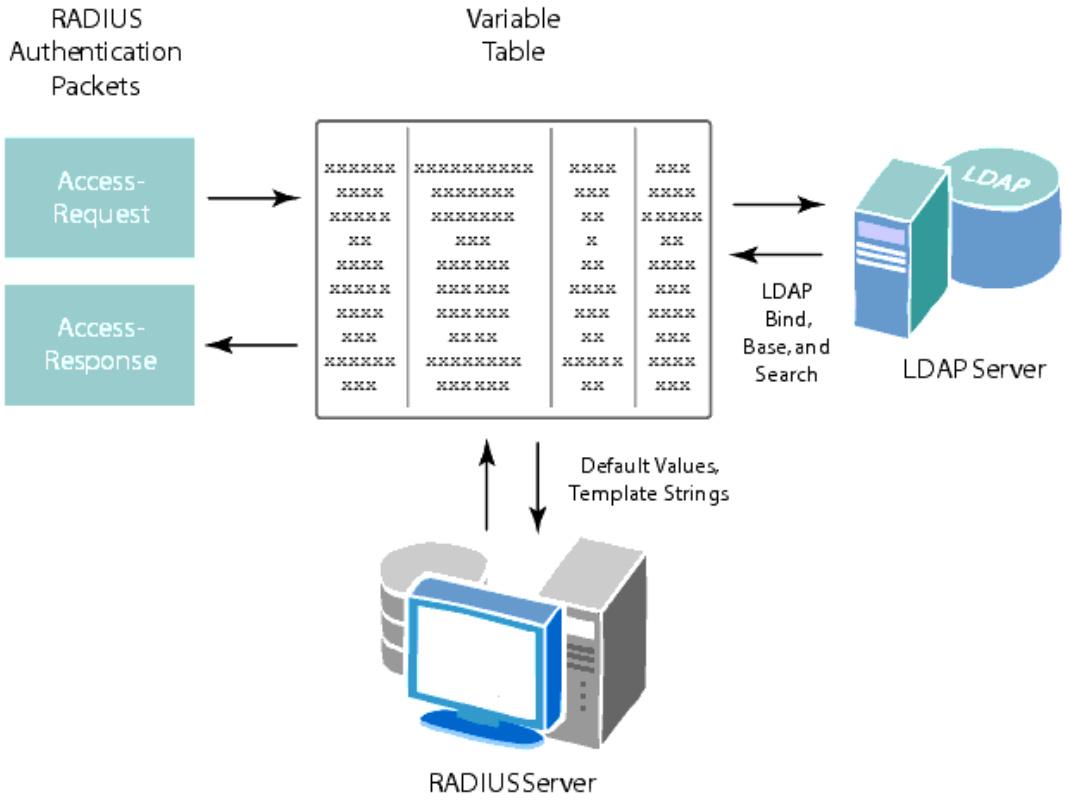
ανάπτυξη της υλοποίησης και επεκτασιμότητα, π.χ. θα μπορούσαμε να έχουμε δύο ή περισσότερα σημεία πρόσβασης (Captive Portals) σε διαφορετικές πόλεις που θα στέλνουν αιτήματα έγκρισης μέσω του δικτύου σε έναν απομακρυσμένο Radius Server που αναλαμβάνει να εξουσιοδοτήσει και να παρακολουθήσει ολόκληρο το δίκτυο. Έτσι μπορούμε να καταλάβουμε ότι το μεγαλύτερο κομμάτι της υλοποίησης βρίσκεται στον Radius Server .

1.4 Επισκόπηση Υπηρεσίας καταλόγου LDAP

Ένας Radius Server μπορεί να παρέχει πιστοποίηση με βάση μητρώα χρηστών που είναι αποθηκευμένα σε μια εξωτερική υπηρεσία καταλόγου LPAP. Μια υπηρεσία καταλόγου LDAP είναι ένα σύνολο αντικειμένων με παρόμοιες ιδιότητες που οργανώνονται κατά τρόπο λογικό και ιεραρχικό. Όπως και ο τηλεφωνικός κατάλογος, που αποτελείται από μια σειρά ονομάτων (είτε των προσώπων είτε των οργανώσεων) που οργανώνονται ιεραρχικά και αλφαβητικά, με κάθε όνομα που ανήκει σε μια πόλη και η πόλη με τη σειρά της σε ένα νομό, έχει μια διεύθυνση και έναν τηλεφωνικό αριθμό συνημμένους. Στην δική μας υλοποίηση για κάθε αντικείμενο (χρήστη) προστέθηκαν και άλλες ιδιότητες π.χ. ο κωδικός του, το ίδρυμα στο οποίο ανήκει την ιδιότητα του σαν χρήστης (σπουδαστής, καθηγητής) σύμφωνα με το σχήμα της υπηρεσίας καταλόγου του ΕΔΕΤ , της υπηρεσίας διασύνδεσης για όλα τα εκπαιδευτικά ιδρύματα της τριτοβάθμιας εκπαίδευσης. (Βλέπε αναλυτικά στο κεφάλαιο 3).

Αρχικά, πριν από κάθε αίτηση πιστοποίησης στον LDAP, ο Radius Server δημιουργεί ένα πίνακα μεταβλητών. Τα χαρακτηριστικά της αίτησης, καθώς και άλλες πληροφορίες σχετικά με αυτήν, εισάγονται στον πίνακα ώστε να χρησιμοποιηθούν από τον LDAP για αναζήτηση μέσα στον κατάλογο. Όταν

βρεθεί ο χρήστης σύμφωνα με τα χαρακτηριστικά της αναζήτησης, όλες οι ιδιότητές του εισάγονται στον ίδιο αυτό πίνακα μεταβλητών και τελικά οι επιλεγμένες πληροφορίες σχετικά με αυτόν δίνονται στον Radius σε ένα πακέτο απάντησης, (Βλέπε σχήμα 1.2).



Σχήμα 1.2

ΚΕΦΑΛΑΙΟ 2. Λογισμικό Radius Server

2.1 Εισαγωγή

Καθώς τα δίκτυα εξαπλώνονται πέρα από το φυσικό χώρο των επιχειρήσεων, η έννοια της ασφάλειας γίνεται πιο σημαντική και σύνθετη. Οι εταιρίες, πρέπει να προστατέψουν τα δίκτυά και τους δικτυακούς τους πόρους από απομακρυσμένους χρήστες που μπαίνουν παράνομα στο σύστημα αποκτώντας πρόσβαση με κάποιο τρόπο. Τα συστήματα της Cisco χρησιμοποιούν μία στρατηγική που είναι γνωστή σαν Πιστοποίηση, Έγκριση και Παρακολούθηση (authentication, authorization, accounting-**AAA**) για να εκτελέσει τις λειτουργίες της πιστοποίησης της ταυτότητας του χρήστη, τη παροχή ή όχι πρόσβασης και την παρακολούθηση των κινήσεων των απομακρυσμένων χρηστών αντίστοιχα. Στα σημερινά δίκτυα χρησιμοποιούνται τα πρωτοκολλά0 TACACS+ (Terminal Access Controller Access Control System plus) και RADIUS (Remote Access Dial-In User Service) για τη παροχή AAA λύσεων. Η υποστήριξη των RADIUS και TACACS+ δίνει τη δυνατότητα στη Cisco να προτείνει μία πολύ ευέλικτη και αποδοτική AAA λύση.

2.2 Ανάλυση των Απαιτήσεων Ασφάλειας (AAA)

Authentication - Πιστοποίηση

Η Πιστοποίηση είναι η διαδικασία με την οποία καθορίζεται ποιός έχει πρόσβαση στο LAN. Απλές μέθοδοι έγκρισης χρησιμοποιούν μια βάση δεδομένων που αποτελείται από usernames και passwords στον server πρόσβασης. Πιο εξελιγμένα συστήματα χρησιμοποιούν μεθόδους όπως το TACACS και το Kerberos.

Ωστόσο, το ότι πιστοποιείται η ταυτότητα κάποιου χρήστη δε σημαίνει ότι αυτός έχει αποκτήσει πρόσβαση σε όλες τις υπηρεσίες του δικτύου . Είναι πιθανό να του ζητηθεί εκ νέου κάποιος κωδικός από κάποια συγκεκριμένη υπηρεσία UNIX, NetWare ή AppleShare. Ένας καλός NAS server υποστηρίζει μία πλειάδα επιλογών πιστοποίησης.

Authorization - Έγκριση

Η Έγκριση είναι η ικανότητα του περιορισμού των δικτυακών υπηρεσιών σε διαφορετικούς χρήστες βάση μιας δυναμικά εφαρμοζόμενης λίστας πρόσβασης (access list) που μερικές φορές αναφέρεται και ως "προφίλ χρήστη" και που βασίζεται στο δίδυμο username/password. Αυτό το χαρακτηριστικό είναι σημαντικό για δύο λόγους: βοηθάει στη μείωση της έκθεσης του εσωτερικού δικτύου στον έξω κόσμο και απλοποιεί τη μορφή του δικτύου για τον τελικό χρήστη που αγνοεί τις τεχνικές του λεπτομέρειες.

Το χαρακτηριστικό της έγκρισης επιτρέπει στους χρήστες να κινούνται. Κινούμενοι και προσωρινοί χρήστες (χρήστες με φορητά από ξενοδοχεία και τηλεργαζόμενοι με modems και ISDN συνδέσεις από το σπίτι) θέλουν να συνδεθούν στη πιο κοντινή τοπική σύνδεση διατηρώντας ωστόσο όλα τα προνόμιά των LAN τους.

Ο Διαχειριστής του δικτύου (Network Administrator) πρέπει να είναι σε θέση να περιορίζει τη πρόσβαση στο δίκτυο για όλα τα πρωτόκολλα και τις υπηρεσίες (Telnet, IP, IPX και AppleTalk) όσο οι χρήστες συνδέονται (dial-in) από τη την ίδια "πηγή" modem (pool). Η διαδικασία έγκρισης με τη χρήση access list για κάθε χρήστη δεν περιορίζεται σε συγκεκριμένα interfaces αλλά ανατίθεται δυναμικά στη συγκεκριμένη πόρτα στην οποία συνδέεται ο χρήστης. Για παράδειγμα όταν ο χρήστης Α συνδέεται στη πόρτα 1, μπορεί να δει τα υπο-δίκτυα 1, 2, 3 και τις AppleTalk ζώνες bldg D, bldg E και bldg F. Όταν ο χρήστης 2 συνδέεται στη πόρτα 1, τότε το προφίλ του τον περιορίζει στο υπο-δίκτυο 1 και στη ζώνη bldg D.

Από τη στιγμή που ο NAS υποστηρίζει πολύ περισσότερους απομακρυσμένους χρήστες από τις φυσικές γραμμές που έχει στη διάθεσή του κάθε χρήστης ή group, μπορεί να τηλεφωνήσει στο ίδιο περιστροφικό κέντρο και να αποκτήσει πρόσβαση στο δίκτυο. Αυτή η access list βασίζεται στο username και σαν τέτοια κάθε NAS μπορεί να υποστηρίξει χιλιάδες χρήστες στη βάση δεδομένων που έχει για τα usernames και passwords.

Accounting-Παρακολούθηση

Η παρακολούθηση είναι το τρίτο κύριο συστατικό ενός ασφαλούς συστήματος. Οι διαχειριστές του συστήματος μπορεί από το να θέλουν να χρεώσουν τους πελάτες τους για την ώρα που παρέμειναν συνδεδεμένοι στο δίκτυο μέχρι να παρακολουθήσουν ύποπτες προσπάθειες σύνδεσης.

2.3 Το Πρωτόκολλο RADIUS

Το πρωτόκολλο RADIUS αναπτύχθηκε από την Livingston Enterprises ως ένας server πρόσβασης, πιστοποίησης και παρακολούθησης. Από τότε έχει υλοποιηθεί από διάφορους άλλους πωλητές και έχει κερδίσει ευρεία υποστήριξη ανάμεσα ακόμα και στους παροχείς υπηρεσιών ίντερνετ(ISPs).

Το RADIUS είναι βασισμένο στο client/server μοντέλο. Οι servers πρόσβασης (**NAS**-Network Access Servers) λειτουργούν σαν clients του RADIUS. Ο client είναι υπεύθυνος για την προώθηση της πληροφορίας του χρήστη στον αρμόδιο RADIUS server και την εκτέλεση των εντολών που θα του σταλούν πίσω από το server.

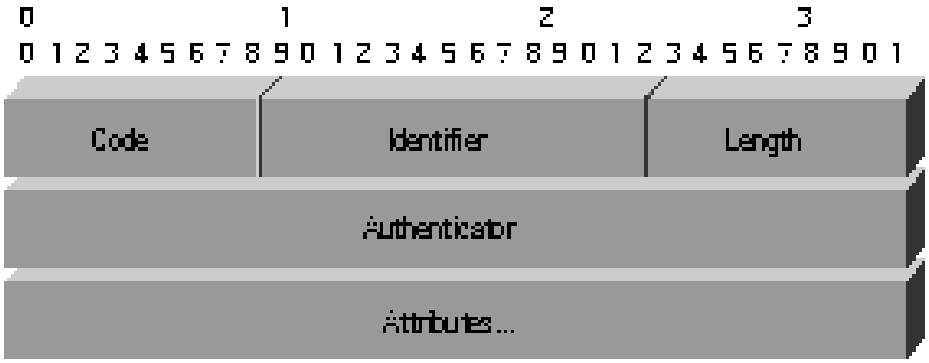
Ο RADIUS server ή daemon παρέχει υπηρεσίες πιστοποίησης και παρακολούθησης σε έναν ή περισσότερους RADIUS clients δηλαδή συσκευές NAS. Οι RADIUS servers είναι υπεύθυνοι για το να λαμβάνουν τις αιτήσεις σύνδεσης των χρηστών, να τους πιστοποιούν και τέλος να επιστρέφουν όλη τη πληροφορία με τις απαιτούμενες ρυθμίσεις για τους clients ώστε να δοθούν οι αιτούμενες υπηρεσίες στους χρήστες. Ο RADIUS server πρόσβασης είναι συνήθως ένας αφιερωμένος σταθμός εργασίας συνδεδεμένος με το δίκτυο.

2.3.1 Λειτουργία Πρωτοκόλλου

Η επικοινωνία μεταξύ ενός NAS [5] και ενός RADIUS SERVER [2] βασίζεται στο User Datagram Protocol (UDP). Το σχήμα 2.1 παρουσιάζει τη μορφή ενός πακέτου RADIUS.

Οι δημιουργοί του RADIUS επέλεξαν το UDP ως το πρωτόκολλο μεταφοράς για τεχνικούς λόγους. Γενικά, το RADIUS θεωρείται μία υπηρεσία άνευ συνδέσεως (connectionless). Θέματα που σχετίζονται με τη διαθεσιμότητα του server, την επανεκπομπή και τα timeouts, διαχειρίζονται από διάφορες συσκευές του RADIUS και όχι από το πρωτόκολλο μεταφοράς.

A summary of the RADIUS data format is shown below.
The fields are transmitted from left to right.



Code

The Code field is one octet, and identifies the type of RADIUS packet.
When a packet is received with an invalid Code field, it is silently discarded.
Radius Codes (decimal) are assigned as follows:

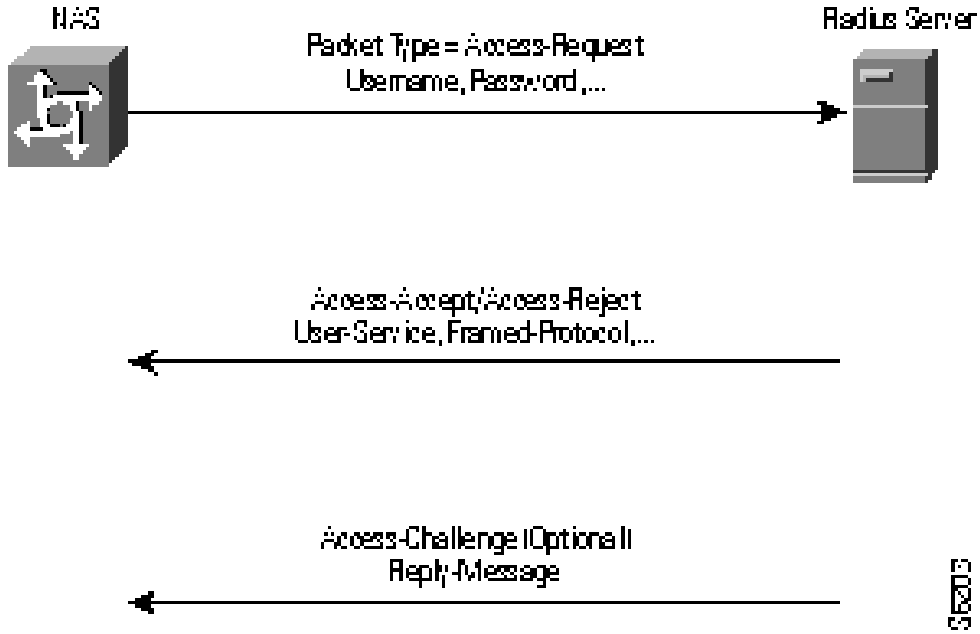
1	Access-Request
2	Access-Accept
3	Access-Reject
4	Accounting-Request
5	Accounting-Response
11	Access-Challenge
12	Status-Server (experimental)
13	Status-Client (experimental)
255	Reserved

Σχήμα 2.1:RADIUS Packet Format from RFC 2058

Τυπικά, μία αίτηση για login αποτελείται από μία αίτηση (Access Request) από το NAS server στον RADIUS server και μια απάντηση, θετική ή αρνητική, του τελευταίου (Access-Accept ή Access-Reject). Το πακέτο αίτησης που στέλνει ο NAS server περιέχει το username, το κρυπτογραφημένο password, την IP διεύθυνση του NAS server και τη πόρτα. Η μορφή της αίτησης παρέχει επιπλέον πληροφορίες για τον τύπο της σύνδεσης την οποία ο χρήστης θέλει να ξεκινήσει. Για παράδειγμα εάν η αίτηση παρουσιάζεται σε mode χαρακτήρων τότε το "Service-Type = Exec-User" αλλά εάν παρουσιάζεται σε mode PPP πακέτου τότε το "Service-Type = Framed User" και "Framed-Type = PPP"

Όταν ο RADIUS server λαμβάνει μια αίτηση από κάποιον NAS, ψάχνει σε μια βάση δεδομένων για το username που υπάρχει στην αίτηση. Εάν το username δεν υπάρχει στη βάση δεδομένων, τότε είτε ένα τυπικό προφίλ φορτώνεται και ο RADIUS server αποστέλλει μήνυμα αποδοχής (Access-Accept), είτε αποστέλλει μήνυμα απόρριψης (Access-Reject), το οποίο μπορεί να συνοδεύεται και από κάποιο επεξηγηματικό μήνυμα του λόγου απόρριψης.

Στην περίπτωση που το username βρεθεί και το password είναι σωστό, ο RADIUS server επιστρέφει μία Access-Accept απάντηση η οποία περιλαμβάνει μια λίστα των χαρακτηριστικών των ρυθμίσεων που πρέπει να χρησιμοποιηθούν από τη μεριά του NAS για τη σύνδεση. Τυπικές παράμετροι περιλαμβάνουν το τύπο της υπηρεσίας (shell ή framed), το τύπο του πρωτοκόλλου, την IP διεύθυνση που θα δοθεί στο χρήστη (στατική ή δυναμική), την access list που πρέπει να εφαρμοστεί ή τη στατική διεύθυνση που πρέπει να εγκατασταθεί στον πίνακα δρομολογίων του NAS. Το σχήμα 2.2 δείχνει τη διαδικασία του RADIUS login και authentication.



Σχήμα 2.2:RADIUS Login and Authentication Process

2.3.2 Χαρακτηριστικά Διαδικασίας Πιστοποίησης και Έγκρισης

Η πιστοποίηση είναι η πιο απαιτητική πλευρά της ασφάλισης απομακρυσμένων χρηστών, λόγω της δυσκολίας που σχετίζεται με τη σίγουρη αναγνώριση του χρήστη. Για τη διασφάλιση της ταυτότητας ενός απομακρυσμένου χρήστη το πρωτόκολλο RADIUS υποστηρίζει πολλές μεθόδους πιστοποίησης περιλαμβανομένων των Password Authentication Protocol (PAP), Challenge Handshake Authentication Protocol (CHAP) και token cards. Προς το παρόν όλες οι εκδόσεις του RADIUS απαιτούν να τρέχει ένας server για τα token cards επιπρόσθετα του RADIUS server. Όταν δημοσιευθεί η έκδοση υποστήριξης του RADIUS, CiscoSecure, θα περιέχει OEM υποστήριξη για CryptoCard token κάρτες και έτσι δεν θα είναι απαραίτητος επιπλέον server για τα token cards.

2.3.3 Ενεργοποίηση της Πιστοποίησης, Έγκρισης και παρακολούθησης RADIUS

Για κάθε τύπο login που χρειάζεται πιστοποίηση και έγκριση, πρέπει να εισαχθεί μια γραμμή εντολών. Αυτή η γραμμή είναι η λίστα που χρησιμοποιείται για login μέσω του RADIUS εκτός αν υπάρχει κάποια άλλη λίστα που έχει ρυθμιστεί. Η παρακολούθηση μπορεί να χρησιμοποιηθεί ανεξάρτητα από τις άλλες διαδικασίες και επιτρέπει την αποστολή δεδομένων στην αρχή και στο τέλος των συνδέσεων, καταδεικνύοντας τη ποσότητα των πόρων που χρησιμοποιήθηκαν κατά τη σύνδεση. Ένας ISP θα μπορούσε να χρησιμοποιήσει το RADIUS για να καλύψει ειδικές απαιτήσεις ασφάλειας και χρέωσης.

ΚΕΦΑΛΑΙΟ 3 Υπηρεσία καταλόγου LDAP.

3.1 Το ελαφρύ πρωτόκολλο λειτουργίας LDAP

Το **ελαφρύ πρωτόκολλο πρόσβασης καταλόγου**, ή **LDAP**(Lightweight Directory Access Protocol) [3], είναι πρωτόκολλο εφαρμογής για ερώτηση και τροποποίηση σε υπηρεσίες καταλόγου που τρέχουν πάνω σε TCP/*IP.

Ένας κατάλογος είναι ένα σύνολο αντικειμένων με παρόμοιες ιδιότητες, που οργανώνονται κατά τρόπο λογικό και ιεραρχικό. Όπως σε ένα σύστημα αρχείων το οποίο αποτελείτε από καταλόγους με υποκαταλόγους στους οποίους βρίσκονται τα αρχεία. Λόγω αυτού του βασικού σχεδίου (μεταξύ άλλων παραγόντων), ο LDAP χρησιμοποιείται συχνά από άλλες υπηρεσίες για την επικύρωση, παρά τα προβλήματα ασφάλειας αυτό που προκαλεί.

Το δέντρο καταλόγου LDAP , απεικονίζει συχνά τα διάφορα πολιτικά, γεωγραφικά, ή/και οργανωτικά όρια, ανάλογα με το πρότυπο που επιλέγεται. Οι επεκτάσεις LDAP τείνουν σήμερα να χρησιμοποιήσουν Domain Name Servers (dns) για τη δόμηση των κορυφαίων επιπέδων της ιεραρχίας. Πιο μέσα στον κατάλογο υπάρχουν καταχωρήσεις που αντιπροσωπεύουν τους ανθρώπους, τις οργανωτικές μονάδες, τους εκτυπωτές, τα έγγραφα, τις ομάδες ανθρώπων ή οτιδήποτε άλλο που αντιπροσωπεύει μια δεδομένη είσοδο δέντρων (ή τις πολλαπλάσιες καταχωρήσεις).

Η τρέχουσα έκδοσή της είναι LDAPv3, το οποίο διευκρινίζεται σε μία σειρά Τυποποιημένη διαδρομή ομάδας εργασίας εφαρμοσμένης μηχανικής Διαδικτύου (IETF) Αιτήματα για σχόλια (RFCs) όπως εκτίθεται λεπτομερώς μέσα στο RFC 4510.

3.2 Προέλευση και επιρροές

Οι επιχειρήσεις τηλεπικοινωνιών, εισήγαγαν την έννοια των υπηρεσιών καταλόγου στην τεχνολογία της πληροφορίας και στα δίκτυα υπολογιστών, καθώς είχαν κατανοήσει ότι η χρήση των υπηρεσιών καταλόγου ήταν αρκετά ανεπτυγμένη μετά από περίπου 70 έτη παραγωγής και διαχείρισης των τηλεφωνικών καταλόγων. Το αποκορύφωμα αυτής της εισαγωγής ήταν η περιεκτική προδιαγραφή X.500 , μια ακολουθία των πρωτοκόλλων που αναπτύχθηκαν από την Διεθνής Ένωση Τηλεπικοινωνιών (ITU) τη δεκαετία του '80.

Οι υπηρεσίες καταλόγου X.500, προσεγγίστηκαν παραδοσιακά μέσω του X.500. Πρωτόκολλο πρόσβασης καταλόγου (DAP), το οποίο απαίτησε Λίστα πρωτοκόλλου διασύνδεσης ανοικτών συστημάτων (OSI). Ο LDAP προορίστηκε αρχικά να είναι ένα "ελαφρύ" εναλλακτικό πρωτόκολλο για την πρόσβαση των υπηρεσιών καταλόγου X.500 μέσω του απλούστερου (και τώρα πλέον διαδεδομένου) πρωτοκόλλου TCP / IP. Αυτό το πρότυπο της πρόσβασης καταλόγου δανείστηκε από το DIXIE και άλλα Πρωτόκολλα υπηρεσιών βοήθειας καταλόγου.

Σύντομα ακολούθησαν οι αυτόνομοι κεντρικοί υπολογιστές καταλόγου LDAP, όπως οι κεντρικοί υπολογιστές καταλόγου που υποστηρίζουν και DAP και LDAP. Το τελευταίο έχει γίνει δημοφιλές στις επιχειρήσεις, ως LDAP απομακρύνοντας οποιαδήποτε ανάγκη να αναπτυχθεί ένα δίκτυο OSI. Σήμερα, τα πρωτόκολλα καταλόγου X.500 συμπεριλαμβανομένου του DAP μπορούν επίσης να χρησιμοποιηθούν άμεσα πάνω στο TCP / IP.

Το πρωτόκολλο δημιουργήθηκε αρχικά από τον Tim Howes, στο Πανεπιστήμιο του Michigan, Steve Kille από την Isode Limited, και ο Wengyik Yeong από την Performance Systems International. Η περαιτέρω ανάπτυξη πραγματοποιήθηκε από την ομάδα εργασίας εφαρμοσμένης μηχανικής Διαδικτύου.

Στα αρχικά στάδια εφαρμογής του ο LDAP ήταν γνωστός ως *ελαφρύ πρωτόκολλο ξεφυλλίσματος καταλόγου*, ή *LDBP*. Αργότερα, μετονομάστηκε δεδομένου ότι το πεδίο του πρωτοκόλλου επεκτάθηκε για να περιλάβει είτε τις λειτουργίες καταλόγου που κοιτάζουν βιαστικά και ψάχνουν τις λειτουργίες, είτε λόγω των αναπροσαρμογών του καταλόγου.

Ο LDAP έχει επηρεάσει τα επόμενα πρωτόκολλα Διαδικτύου, συμπεριλαμβανομένων των πιο πρόσφατων εκδόσεων του X.500, XML κατάλογος (XED), Γλώσσα σήμανσης υπηρεσιών καταλόγου (DSML), Γλώσσα σήμανσης υπηρεσιών (SPML), και Πρωτόκολλο θέσης υπηρεσιών (SLP).

3.3 Επισκόπηση πρωτοκόλλου

Ένας πελάτης αρχίζει μια σύνοδο LDAP με τη σύνδεση με έναν κεντρικό υπολογιστή LDAP, εξ ορισμού πάνω σε μια TCP πόρτα 389. Έπειτα, ο πελάτης στέλνει ένα αίτημα λειτουργίας στον κεντρικό υπολογιστή, και στη συνέχεια ο κεντρικός υπολογιστής στέλνει τις απαντήσεις. Με μερικές εξαιρέσεις, ο πελάτης δεν χρειάζεται την αναμονή για μια απάντηση πριν στείλει το επόμενο αίτημα, ενώ ο κεντρικός υπολογιστής μπορεί να στείλει τις απαντήσεις σε οποιαδήποτε διαταγή.

Ο πελάτης μπορεί να ζητήσει τις ακόλουθες διαδικασίες:

- Δεσμεύστε - επικυρώστε και διευκρινίστε την έκδοση πρωτοκόλλου LDAP.
- Έναρξη TLS - χρησιμοποιήστε το LDAPv3 Επέκταση ασφάλειας στρώματος μεταφορών (TLS) για μια ασφαλή σύνδεση.
- Η αναζήτηση - αναζήτηση ή/και ανακτά τις καταχωρήσεις καταλόγου.

- Συγκρίνετε - εξετάστε εάν μια ονομασμένη είσοδος περιέχει μια δεδομένη αξία ιδιοτήτων.
- Προσθέστε ένα νέο λήμμα.
- Διαγράψτε ένα λήμμα.
- Τροποποιήστε μια είσοδο.
- Τροποποιήστε το διακεκριμένο όνομα (DN) - κινήστε ή μετονομάστε μια είσοδο.
- Εγκαταλείψτε - αποβάλτε ένα προηγούμενο αίτημα.
- Εκτεταμένη λειτουργία - γενική λειτουργία που χρησιμοποιείται για να καθορίσει άλλες διαδικασίες.
- Αποδεσμεύστε - κλείστε τη σύνδεση (όχι το αντίστροφο Bind).

Επιπλέον, ο κεντρικός υπολογιστής μπορεί να στείλει τις "εκούσιες ανακοινώσεις" που δεν είναι απαντήσεις σε οποιοδήποτε αίτημα, π.χ. πριν από το τέλος του χρόνο ζωής μιας σύνδεσης.

Μια κοινή μέθοδος ασφαλείας στην επικοινωνία LDAP χρησιμοποιεί SSL. Αυτό χρησιμοποιείται σε LDAP URLs με τη χρησιμοποίηση του σχεδίου URL "ldaps". Η πόρτα προεπιλογής για τον LDAP SSL είναι η 636. Η χρήση LDAP πάνω σε SSL ήταν κοινή στην έκδοση 2 (LDAPv2) αλλά δεν τυποποιήθηκε ποτέ σε οποιαδήποτε επίσημη προδιαγραφή. Αυτή η χρήση έχει αποδοκιμαστεί μαζί με την LDAPv2, το οποίο αποσύρθηκε επίσημα το 2003.

3.4 Περιγραφή σχήματος LDAP στον ds.grnet.gr

Το παρόν έγγραφο αποτελεί μια περιγραφή του σχήματος που υιοθετήθηκε για την διαμόρφωση της Υπηρεσίας Καταλόγου του **ds.grnet.gr**. [9] Ιδιαίτερη έμφαση δίνεται στον ορισμό, την περιγραφή και την τεκμηρίωση αναγκαιότητας του objectclass (κλάσης) **eduPerson** για το σχήμα του ds.grnet.gr.

Πριν περάσουμε στην αναλυτική περιγραφή του σχήματος για το ds.grnet.gr, θεωρούμε χρήσιμο να ορίσουμε συνοπτικά κάποιες έννοιες, οι οποίες είναι αναγκαίες για την κατανόηση του κειμένου. Όπως βλέπουμε στον Πίνακα 3.1

LDAP	**Σύνολο από πρωτόκολλα για την προσπέλαση καταλόγων πληροφοριών. Βασίζεται στο πρωτόκολλο X.500, αλλά είναι απλούστερο. Υποστηρίζει το πρωτόκολλο TCP/IP**
Schema (σχήμα)	Χρησιμοποιείται για να ορίσει την δομή ενός καταλόγου
LDIF (LDAP Data Interchange Format)	Απεικόνιση των δεδομένων ενός καταλόγου σε μορφή κειμένου Χρησιμοποιείται για την εισαγωγή κι εξαγωγή δεδομένων σ' έναν directory server.
Entry (εγγραφή)	Η βασική μονάδα πληροφορίας σ' έναν LDAP. Αντίστοιχο του record (εγγραφή) σε μια Βάση Δεδομένων
Object (αντικείμενο)	Οντότητα, η οποία περιγράφεται με attributes κι έχει ένα μοναδικό όνομα, το οποίο το ξεχωρίζει από τ' άλλα objects. Αντίστοιχο του table (πίνακα) σε μια Βάση Δεδομένων
Objectclass (κλάση)	Χρησιμοποιούνται για την ταξινόμηση παραπλήσιων δεδομένων. Κάθε entry ανήκει σε ένα ή περισσότερα objectclasses.

Attribute (χαρακτηριστικό)	Κάθε entry αποτελείται από ένα ή περισσότερα attributes, τα οποία περιγράφουν κάποια χαρακτηριστικά του, π.χ. cn (το επώνυμο), mail (τη διεύθυνση ηλεκτρονικού ταχυδρομείου) κλπ.. Αντίστοιχο του field (πεδίου) σε μια Βάση Δεδομένων
Distinguished Name (DN)	Το όνομα που ορίζει μοναδικά μια εγγραφή, π.χ.”cn=John Smith, ou=People, o=ACompany, c=gr”

Πίνακας 3.1: Επεξήγηση όρων LDAP

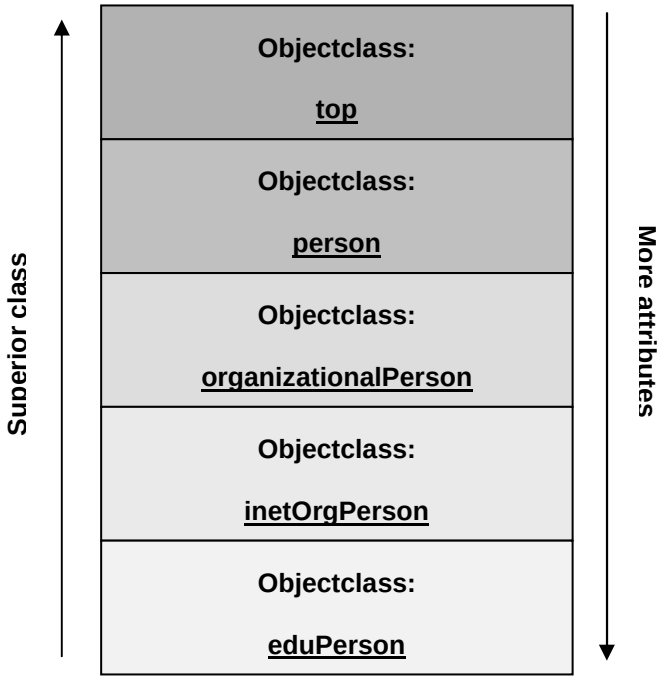
Κάθε LDAP server πρέπει να χρησιμοποιεί - ή να υλοποιεί - ένα συγκεκριμένο **σχήμα** (**schema**), το οποίο καθορίζει ποιά **χαρακτηριστικά** (**attributes**) μπορούν να αποθηκευτούν στα διάφορα **αντικείμενα** (**objects**). Δηλαδή το σχήμα σ’ έναν directory server καθορίζει τα objectclasses και τα attributes που χρησιμοποιούνται.

3.5 Ο Directory Server του ΕΔΕΤ

Ο Directory Server του ΕΔΕΤ [9], έχει σαν βάση του το δέντρου **c=GR**, ενώ ιεραρχικά κάτω από αυτό βρίσκονται το "**o=grnet, c=gr**", το "**o=hua, c=gr**", καθώς και **referrals** προς τους αντίστοιχους καταλόγους των συνεργαζομένων ιδρυμάτων. Παρέχεται επίσης η δυνατότητα στα συνεργαζόμενα ιδρύματα, να φιλοξενήσουν εξ’ ολοκλήρου τον Directory Server τους τοπικά, κατά το πρότυπο του o=grnet, με απομακρυσμένη διαχείριση στους διαχειριστές του. Παρακάτω ακολουθεί μια περιγραφή του σχήματος του DS για το ΕΔΕΤ.

3.5.1 Περιγραφή standard κλάσεων (objectclasses)

Ένα objectclass μπορεί να προκύπτει από κάποιο άλλο, μέσω μιας διαδικασίας που ονομάζεται **κληρονομικότητα** (**inheritance**). Σ' αυτή την περίπτωση, η «κληρονόμος» objectclass κληρονομεί κάποια χαρακτηριστικά της «κληρονομούμενης» objectclass και αυτό ονομάζεται **subclassing** ή **objectclass inheritance**. Αυτό φαίνεται περιγραφικά στο σχήμα 3.2, ενώ ακολουθεί επεξήγηση των κλάσεων:



Σχήμα 3.2: objectclass inheritance

objectClass: eduPerson

Κληρονομεί την κλάση **inetOrgPerson**. Υπάρχει εκτενής αναφορά στη συγκεκριμένη κλάση αργότερα στο κείμενο.

objectClass: inetOrgPerson

Κληρονομεί την κλάση Person μέσω της ενδιάμεσης κλάσης organizationalPerson. Εμπεριέχει συμπληρωματικά attributes, ώστε να μπορεί να περιέχει δεδομένα που έχουν ευρεία χρήση στο Internet και μέσα σε οργανισμούς (organizations).

objectClass: organizationalPerson

Κληρονομεί την κλάση **Person**.

objectClass: person

Κληρονομεί την **αφηρημένη** (**abstract**) κλάση **top**.

objectClass: top

Αποτελεί μια ιδιαίτερη, αφηρημένη κλάση, από την οποία όλες οι άλλες άμεσα ή έμμεσα κληρονομούν χαρακτηριστικά της.

Γενικότερα, η κλάση από την οποία μια άλλη κλάση κληρονομεί χαρακτηριστικά ονομάζεται **superior** ή **superclass**. Θα λέγαμε δηλαδή ότι η κλάση top είναι η superclass της κλάσης Person.

Όταν μια κλάση κληρονομεί μια άλλη, κληρονομεί το σύνολο των υποχρεωτικών attributes, το σύνολο των προαιρετικών attributes, καθώς και τον τύπο της κλάσης την οποία κληρονομεί. Εξυπακούεται ότι μια κλάση κληρονομώντας μια άλλη, αναγκαστικά κληρονομεί και τα χαρακτηριστικά της superclass της.

3.5.2 Περιγραφή υπολοίπων χαρακτηριστικών

Ακολουθεί περιγραφή των attributes που χρησιμοποιούνται από το schema του directory server του ΕΔΕΤ:

cn (commonName, ορισμένο στο person); OID 2.5.4.3

Αντιστοιχεί στο όνομα ενός αντικειμένου. Αν αυτό είναι άτομο, αποτελεί συνήθως το ονοματεπώνυμό του.

description (ορισμένο στο person); OID 2.5.4.13

Αντιστοιχεί σε μια ανεπίσημη περιγραφή του ατόμου. Πρακτικά μπορεί να δεχθεί οποιαδήποτε τιμή, π.χ. τα ενδιαφέροντα ενός ατόμου. Έχει πολυγλωσσική υποστήριξη.

displayName (ορισμένο στο inetOrgPerson) OID 2.16.840.1.113730.3.1.241

Αντιστοιχεί στο όνομα που θέλει ο χρήστης να εμφανίζεται στην εγγραφή του, π.χ. το όνομα και το επώνυμό του, ωστόσο κάτι τέτοιο δεν είναι δεσμευτικό. Έχει πολυγλωσσική υποστήριξη.

givenName (ορισμένο στο inetOrgPerson); OID 2.5.4.42

Αντιστοιχεί σ’ εκείνο το μέρος του ονόματος του ατόμου που δεν είναι επώνυμο ή πατρώνυμο (δηλαδή το όνομα). Έχει πολυγλωσσική υποστήριξη.

homePhone (ορισμένο στο inetOrgPerson); OID 0.9.2342.19200300.100.1.20

Αντιστοιχεί στον αριθμό τηλεφώνου της οικίας του ατόμου.

homePostalAddress (ορισμένο στο inetOrgPerson); OID 0.9.2342.19200300.100.1.39

Αντιστοιχεί στην ταχυδρομική διεύθυνση της οικίας του ατόμου. Έχει πολυγλωσσική υποστήριξη.

l (ορισμένο στο organizationalPerson); OID 2.5.4.7

Αντιστοιχεί στην γεωγραφική περιοχή με την οποία συνδέεται το άτομο, π.χ. πόλη στην οποία εργάζεται το άτομο. Έχει πολυγλωσσική υποστήριξη.

labeledUri (ορισμένο στο inetOrgPerson); OID 1.3.6.1.4.1.250.1.57

Αντιστοιχεί στο url που συνδέεται με το άτομο, π.χ. την προσωπική του ιστοσελίδα.

mobile (ορισμένο στο inetOrgPerson); OID 0.9.2342.19200300.100.1.41

Αντιστοιχεί στον αριθμό του κινητού τηλεφώνου του ατόμου

o (ορισμένο στο inetOrgPerson); OID 2.5.4.10

Αντιστοιχεί στην κορυφή της ιεραρχίας του ιδρύματος, με το οποίο το άτομο συνδέεται.

ou (ορισμένο στο inetOrgPerson); OID 2.5.4.11

Αντιστοιχεί στην μονάδα εκείνη του ιδρύματος, οργανισμού, κλπ., με την οποία το άτομο συνδέεται.

postalAddress (ορισμένο στο orgPerson); OID 0.9 2.5.4.16

Αντιστοιχεί στην ταχυδρομική διεύθυνση της εργασίας του ατόμου. Έχει πολυγλωσσική υποστήριξη.

preferredLanguagel (ορισμένο στο inetOrgPerson); OID 2.16.840.1.113730.3.1.39

Αντιστοιχεί στην προτιμώμενη γλώσσα του ατόμου. Έχει πολυγλωσσική υποστήριξη.

sn (ορισμένο στο person); OID 2.5.4.4

Αντιστοιχεί στο επίθετο του ατόμου. Έχει πολυγλωσσική υποστήριξη.

telephoneNumber (ορισμένο στο person); OID 2.5.4.20

Αντιστοιχεί στον πρωτεύοντα υπηρεσιακό αριθμό τηλεφώνου του ατόμου.

title (ορισμένο στο inetOrgPerson); OID 2.5.4.12

Προσδιορίζει την λειτουργία, ασχολιά ή υπευθυνότητα του αντικειμένου (ατόμου) στα πλαίσια ενός ιδρύματος. Έχει πολυγλωσσική υποστήριξη.

uid (ορισμένο στο inetOrgPerson); OID 0.9.2342.19200300.100.1.1

Αντιστοιχεί στο όνομα χρήστη ή άλλο μοναδικό όνομα που συνδέεται με το άτομο.

userCertificate (ορισμένο στο inetOrgPerson); OID 2.5.4.35

Αντιστοιχεί στο πιστοποιητικό του χρήστη, το οποίο είναι base-64 encoded.

userPassword (ορισμένο στο person); OID 2.5.4.35

Αντιστοιχεί στο μυστικό κωδικό του ατόμου. Έχει πολυγλωσσική υποστήριξη.

mail (ορισμένο στο inetOrgPerson); OID 0.9.2342.19200300.100.1.3

Αντιστοιχεί στην διεύθυνση ηλεκτρονικού ταχυδρομείου

objectclass eduPerson

Η κλάση **eduPerson** αποτελεί καρπό της συνεργασίας πανεπιστημιακών

ιδρυμάτων, κυβερνητικών φορέων και ιδιωτών, στα πλαίσια του internet2 και του **EDUCAUSE**. Η κλάση **eduPerson** (OID: 1.3.6.1.4.1.5923.1.1.2) είναι μια βοηθητική κλάση (objectclass) για τις Υπηρεσίες Καταλόγου των πανεπιστημιακών ιδρυμάτων και εμπεριέχει χαρακτηριστικά τα οποία συναντά κανείς στην πλειοψηφία αυτών.

Απώτερος στόχος είναι να αποτελέσει η κλάση eduPerson ως προδιαγραφή για τις Υπηρεσίες Καταλόγου πανεπιστημιακών ιδρυμάτων και αυτό επειδή είναι σχεδιασμένη με τέτοιο τρόπο, ώστε να διευκολύνει την διαπανεπιστημιακή επικοινωνία κάνοντας χρήση εξελιγμένων υπηρεσιών IT. Με αυτό τον τρόπο, μπορούν πάνω στη κλάση αυτή να βασιστούν μια σειρά από προηγμένες υπηρεσίες, όπως πρόσβαση σε πόρους, ανταλλαγή δεδομένων, κ.α. σε διαπανεπιστημιακό επίπεδο.

Η κλάση eduPerson, περιλαμβάνει attributes για μέλη της πανεπιστημιακής κοινότητας, μαζί με συστάσεις αναφορικά με τη σύνταξη ή τη σημασία των δεδομένων που εισάγονται σε αυτά.

Υπάρχει σύσταση προς τα πανεπιστημιακά ιδρύματα, τα οποία υλοποιούν την κλάση eduPerson, όλες οι εγγραφές τύπου person να κληρονομούν άλλη μία κλάση, με όνομα <localdomain>EduPerson, όπου <localdomain> είναι το όνομα του ιδρύματος. Έτσι για το AUTH λ.χ., η κλάση θα είναι authEduPerson. Σκοπός της σύστασης αυτής, είναι να προστεθούν στην κλάση αυτή όλα εκείνα τα χαρακτηριστικά που θέλει να ενσωματώσει το ίδρυμα, αλλά δεν περιλαμβάνονται στις τυπικές προδιαγραφές του eduPerson. Με αυτό τον τρόπο αποφεύγεται η χρήση διπλών ονομάτων σε παγκόσμιο επίπεδο.

Η ερμηνεία των attributes του eduPerson βρίσκεται στο specification του eduPerson, ωστόσο ποιές εγγραφές θα έχουν συμπληρωμένες τιμές για την κλάση αυτή, καθώς και τί τιμές θα είναι αυτές, είναι ένα θέμα που εναπόκειται στα ίδια τα εκπαιδευτικά ιδρύματα. Πρέπει να σημειωθεί ότι όλα τα χαρακτηριστικά (attributes) της eduPerson είναι προαιρετικά, ωστόσο ενθαρρύνεται η χρήση όσο το δυνατόν περισσοτέρων.

Περιγραφή attributes του objectclass eduPerson

Παρακάτω ακολουθεί μια περιγραφή των attributes του **eduPerson**, τα οποία χρησιμοποιήθηκαν για την δημιουργία του πρότυπου schema (σχήματος) για το ΕΔΕΤ. Συγκεκριμένα:

eduPersonAffiliation (ορισμένο στο eduPerson 1.0); OID: 1.3.6.1.4.1.5923.1.1.1.1

Προσδιορίζει τη σχέση ή τις σχέσεις του ατόμου ως προς το πανεπιστημιακό ίδρυμα σε ευρείες κατηγορίες, π.χ. student (φοιτητής), faculty (διδακτικό προσωπικό), staff (προσωπικό), alum (απόφοιτοι), κλπ. Εδώ πρέπει να περιλαμβάνεται και η τιμή που έχει το χαρακτηριστικό eduPersonPrimaryAffiliation. Δέχεται και περισσότερες από μια τιμές (πλειότιμο), π.χ. student και staff, ενώ υπάρχει πολυγλωσσική υποστήριξη για ελληνικά κι αγγλικά.

eduPersonNickname (ορισμένο στο eduPerson 1.0); OID: 1.3.6.1.4.1.5923.1.1.1.2

Προσδιορίζει το άτυπο όνομα ενός ατόμου (το «χαϊδευτικό») ή το όνομα με το οποίο θέλουν να του απευθυνόμαστε, π.χ. Κώστας, Γιάννης, κλπ. Παρέχει πολυγλωσσική υποστήριξη.

eduPersonOrgDN (ορισμένο στο eduPerson 1.0); OID: 1.3.6.1.4.1.5923.1.1.1.3

Αποτελείται από το DN της εγγραφής που προσδιορίζει το ίδρυμα με το οποίο ο χρήστης συνδέεται (στο οποίο ανήκει). Μέσω του DN ο πάροχος μπορεί να αποκομίσει πληροφορίες για το ίδρυμα από το οποίο προέρχεται ο χρήστης, αλλά και να κατευθύνει κατάλληλα ερωτήματα προς τον directory server, ώστε να συλλέξει παραπάνω πληροφορίες. Το eduPersonOrgDN μαζί με το cn και το sn αποτελούν τον πυρήνα της κλάσης eduPerson, ως προς τη χρηστικότητά της σε εφαρμογές.

eduPersonOrgUnitDN (ορισμένο στο eduPerson 1.0); OID: 1.3.6.1.4.1.5923.1.1.1.4

Αποτελείται από το DN της εγγραφής που προσδιορίζει το Organizational Unit του χρήστη. Επιδέχεται παραπάνω από μια τιμές, καθώς π.χ. ένας φοιτητής σε μια μονάδα (σχολή/τμήμα) μπορεί παράλληλα να είναι και εργαζόμενος σε κάποια άλλη.

eduPersonPrimaryAffiliation (ορισμένο στο eduPerson 1.0); OID: 1.3.6.1.4.1.5923.1.1.1.5

Προσδιορίζει την πρωτεύουσα σχέση ενός ατόμου με το ίδρυμά του σε ευρείες κατηγορίες όπως student, faculty, staff, alum, κλπ. Δέχεται μόνο μια τιμή κι έχει νόημα μόνο εάν υπάρχει τουλάχιστο μια τιμή ορισμένη στο eduPersonAffiliation. Αν για παράδειγμα, ένα μέλος είναι student και staff, τότε εδώ θα μπει student ή staff, πράγμα που εξαρτάται μόνο από το ποια από τις δυο είναι η πρωτεύουσα σχέση του συγκεκριμένου ατόμου προς το ίδρυμά του. Παρέχει πολυγλωσσική υποστήριξη

eduPersonPrincipalName (ορισμένο στο eduPerson 1.0); OID: 1.3.6.1.4.1.5923.1.1.1.6

Η ηλεκτρονική διεύθυνση ενός ατόμου για τις ανάγκες της διαπανεπιστημιακής πιστοποίησης. Θα πρέπει να είναι της μορφής user@univ.edu (ή .gr), όπου univ.edu (ή .gr) είναι το όνομα του πανεπιστημιακού domain. Εάν τεθεί τιμή, ο χρήστης θα πρέπει να μπορεί να κάνει, υπό προϋποθέσεις πιστοποίηση (authentication) σε τοπικές υπηρεσίες του πανεπιστημίου ή ακόμη και σε διαπανεπιστημιακό επίπεδο.

Το πεδίο αυτό μπορεί να συμπίπτει με την ηλεκτρονική διεύθυνση του χρήστη, αλλά αυτό δεν είναι απαραίτητο. Η χρήση του περιορίζεται για πιστοποίηση και δεν πρέπει να μπερδεύεται με την ηλεκτρονική διεύθυνση του χρήστη.

eduPersonEntitlement (ορισμένο στο eduPerson 200210); OID: 1.3.6.1.4.1.5923.1.1.1.7

Αποτελείται από ένα URI (URN ή URL), το οποίο προσδιορίζει ένα σύνολο δικαιωμάτων για συγκεκριμένους πόρους σε συγκεκριμένες πηγές. Η ανάγκη για το χαρακτηριστικό αυτό προέκυψε από το πρόγραμμα Shibboleth του MACE. Το συγκεκριμένο attribute αποδεικνύεται χρήσιμο στην περίπτωση που ένα ίδρυμα έχει συνάψει συνεργασία με κάποιο άλλο ίδρυμα ή φορέα (πάροχο) για την παροχή υπηρεσιών ή πόρων. Σε αυτή την περίπτωση, ο πάροχος μπορεί να διατηρεί μια κατάσταση με τα συνεργαζόμενα με αυτό ιδρύματα. Όταν κάποιος χρήστης αιτηθεί πρόσβαση σε πόρους του πάροχου, ο τελευταίος, αφού ελέγξει την κατάστασή του για να δει εάν ο χρήστης ανήκει στα συνεργαζόμενα με αυτό ιδρύματα, ζητάει από τον LDAP εξυπηρετητή το χαρακτηριστικό eduPersonEntitlement, για να διαπιστώσει κατά πόσο ο συγκεκριμένος χρήστης έχει πρόσβαση στην υπηρεσία.

eduPersonPrimaryOrgUnitDN (ορισμένο στο eduPerson 200210); OID:1.3.6.1.4.1.5923.1.1.1.8

Αποτελείται από το DN της εγγραφής που προσδιορίζει το πρωτεύων Organizational Unit του χρήστη, δηλαδή την κύρια ενασχόλησή του. Αν ένας χρήστης ανήκει σε πολλά τμήματα (OU), τότε εδώ θα μπει το τμήμα εκείνο που θεωρείται πρωτεύον. Δέχεται μόνο μια τιμή κι έχει νόημα μόνο εάν υπάρχει τουλάχιστο μια τιμή στο eduPersonOrgUnitDN.

eduPersonScopedAffiliation (ορισμένο στο eduPerson 200312); OID: 1.3.6.1.4.1.5923.1.1.1.9

Προσδιορίζει τη θέση ενός ατόμου μέσα σ’ ένα συγκεκριμένο χώρο. Η θέση αυτή υποδηλώνεται με πλατύς όρους, όπως student, faculty, κλπ. Οι τιμές που παίρνει αυτό το attribute αποτελούνται από δυο μέρη, το αριστερό και το δεξί, με το σύμβολο @ να τα χωρίζει. Το αριστερό αντλεί τις τιμές του από το eduPersonAffiliation, ενώ το δεξί αποτελείται από τον χώρο για τον οποίο προσδιορίζουμε τη θέση του ατόμου, χωρισμένο με τελείες κατά τα πρότυπα του DNS και συμπίπτει με τις τιμές στο δεξί τμήμα του

eduPersonPrincipalName.

Για παράδειγμα, εάν είχαμε δυο άτομα από το Τμήμα Πληροφορικής, ένα του προσωπικού κι ένα φοιτητή, το πρώτο θα είχε αυξημένα δικαιώματα, σε σχέση με το δεύτερο, στο χώρο computing.<institution_name>.gr, καθώς τα δικαιώματά τους είναι διαφορετικά. Οι αντίστοιχες τιμές των δυο αυτών ατόμων στο attribute eduPersonScopedAffiliation, θα μπορούσαν να είναι staff@computing. <institution_name>.gr και student@computing. <institution_name>.gr. Με αυτό τον τρόπο το attribute αυτό μπορεί να χρησιμοποιηθεί για τον καθορισμό του επιπέδου δικαιωμάτων σε επίπεδο ιδρύματος όσο και σε διαπανεπιστημιακό επίπεδο.

eduPersonTargetedID (ορισμένο στο eduPerson 200312); OID: 1.3.6.1.4.1.5923.1.1.1.10

Προστέθηκε για πρώτη φορά στο eduPerson Specification 200312 και αποβλέπει κυρίως στην υποστήριξη συστημάτων ομοσπονδιακής διαχείρισης ταυτότητας (federated identity management systems), όπως είναι το Shibboleth. Ο σκοπός του είναι να χρησιμοποιηθεί ως identifier μεταξύ φορέων που επιθυμούν να μοιραστούν πόρους. Χρησιμοποιείται για την ταυτοποίηση του χρήστη μεταξύ του φορέα που παρέχει τα πιστοποιητικά (π.χ. ίδρυμα) και του φορέα που διαθέτει την πρόσβαση στους πόρους (π.χ. βιβλιοθήκη). Το πεδίο αυτό απευθύνεται αποκλειστικά στο φορέα του πόρου και δεν πρέπει να αποκαλύπτεται σε κανέναν.

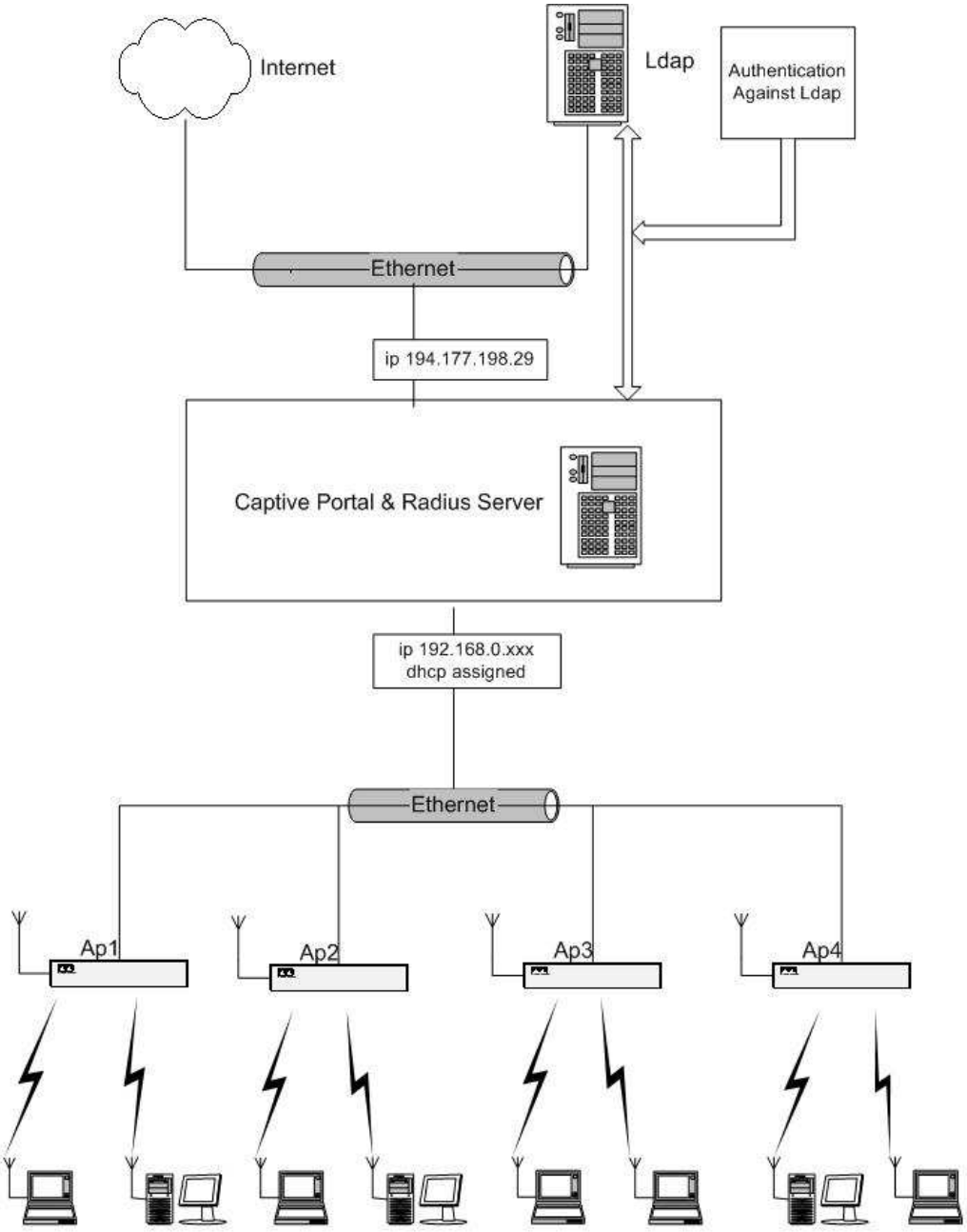
ΚΕΦΑΛΑΙΟ 4

Υλοποίηση Captive portal

Σε αυτό το κεφάλαιο, θα περιγράψουμε τη σχεδίαση και την υλοποίηση ενός ασύρματου δικτύου, με δυνατότητα εξουσιοδοτημένης πρόσβασης στο χώρο του ΤΕΙ Κρήτης Παραρτήματος Χανίων. Στις επόμενες ενότητες, θα αναλυθούν οι λειτουργίες του καθώς και οι μηχανισμοί πρόσβασης και διαχείρισης του.

4.1 Εισαγωγή

Η διάταξή μας αποτελείται από έναν κεντρικό υπολογιστή, που ενώνεται ενσύρματα με το κεντρικό σημείο πρόσβασης της σχολής και αναλαμβάνει το ρόλο του firewall. Με τη χρήση του firewall, η κυκλοφορία των δεδομένων απομονώνεται και διοχετεύεται μέσω ενός σταθερού σημείου εισόδου, όπου εκεί μπορούν να εφαρμοστούν τα πρόσθετα στρώματα ασφαλείας μέσω της χρήσης ενός ιδεατού ιδιωτικού δικτύου όπως φαίνεται και παρακάτω (Σχήμα 4.1).


Internet
Ldap
Authentication Against Ldap
Ethernet
ip 194.177.198.29
Captive Portal & Radius Server
ip 192.168.0.xxx dhcp assigned
Ethernet
Ap1
Ap2
Ap3
Ap4

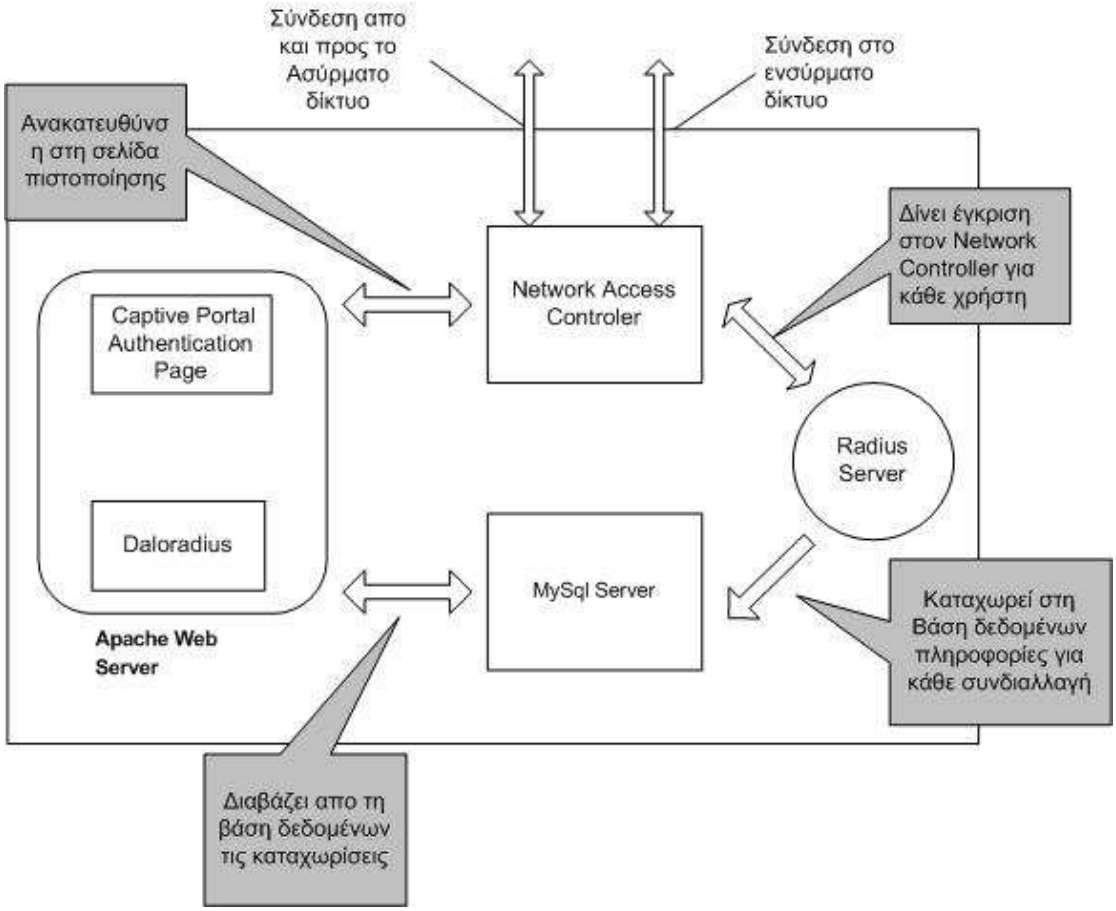
Σχήμα 4.1

4.2 Υλικό

Για την υλοποίηση του Captive Portal, χρησιμοποιήθηκε ένας υπολογιστής ως server. Στον υπολογιστή εγκαταστάθηκαν δύο κάρτες δικτύου. Η μία συνδέει τον server με το εσωτερικό δίκτυο του ΤΕΙ και το internet, ενώ η άλλη χρησιμοποιείται για την ενσύρματη δικτύωση με τον κατανεμητή όπου συνδέονται τα 4 access point τα οποία είναι της εταιρίας Cisco Systems καθώς ένα ήδη υπάρχον στο χώρο της βιβλιοθήκης.

4.3 Λογισμικό

Στον Server εγκαταστάθηκε λειτουργικό σύστημα Linux Ubuntu Server 8.04. Το Coova-chilli 1.0.11 ανέλαβε το κυρίως κομμάτι της υλοποίησης, την παροχή διευθύνσεων στους πελάτες (DHCP Server) και τον έλεγχο της κίνησης (traffic control). Την πιστοποίηση (Authentication), την εξουσιοδότηση (Authorization), καθώς και την παρακολούθηση (Accounting) της χρήσης του δικτύου την ανέλαβε ο Freeradius 2.1.4. Επίσης, εγκαταστάθηκε ο web server Apache 2.2.11 όπου αναλαμβάνει την επικοινωνία με τον τελικό χρήστη, καθώς και ο MySql Server 5.0.51 για την καταγραφή του ιστορικού πρόσβασης σε μια βάση δεδομένων. Για την διαχείριση της βάσης του ιστορικού χρήσης. εγκαταστάθηκε στον web server ο Daloradius ένα σετ από php script που επιτρέπει την διαχείριση της βάσης με γραφικό τρόπο μέσα από τον web browser και απομακρυσμένα. Σχήμα 4.1



Σχήμα4.1

4.4 Περιγραφή του Coova-Chilli

Το Coova-chilli είναι ένα λογισμικό ελέγχου πρόσβασης ανοικτού κώδικα, που χρησιμοποιείται ευρέως στις υλοποιήσεις ασύρματων σημείων πρόσβασης. Προσφέρει πολλές δυνατότητες και χρησιμοποίει το πρωτόκολλο Radius για την παροχή πρόσβασης και παρακολούθησης

Σαν εφαρμογή έχει τρία βασικά υποσυστήματα. Ένα υποσύστημα για την διασύνδεση των χρηστών και την ανάθεση IP διευθύνσεων μέσω DHCP , ένα για την σύνδεση με τον Radius Server για την πιστοποίηση των χρηστών και ένα τρίτο για την σύνδεση με το κοινόχρηστο δίκτυο για την προώθηση της κίνηση από και προς άλλα δίκτυα.

Όλες οι παράμετροι λειτουργίας του Coova-Chilli, είτε μπορούν να διαμορφωθούν από τη γραμμή εντολών του συστήματος Unix, είτε από το βασικό αρχείο διαμόρφωσης, καθώς εκεί αποθηκεύονται οι ρυθμίσεις που φορτώνονται κατά την εκκίνηση του προγράμματος. Το προεπιλεγμένο αρχείο διαχείρισης, βρίσκεται στη θέση: */usr/local/etc/chilli.conf* στην οποία μπορούμε να οδηγηθούμε δίνοντας στην κονσόλα την εντολή **–conf .** Στην ενότητα αυτή παρουσιάζονται οι ρυθμίσεις που έγιναν για το στήσιμο της εφαρμογής.

4.4.1 Ανάθεση IP διευθύνσεων μέσω DHCP Server

Με τον όρο DHCP (Dynamic Host Configuration Protocol) αναφερόμαστε σε ένα μηχανισμό διαχείρισης TCP/IP πρωτοκόλλων.

Το πρωτόκολλο είναι ουσιαστικά ένα λογισμικό που τρέχει σε έναν υπολογιστή και κανονίζει όλα τα θέματα επικοινωνίας με αυτόν τον υπολογιστή και άλλους που χρησιμοποιούν αυτό το πρωτόκολλο σαν γλώσσα. Για να δουλέψει το ίδιο λογισμικό σε τόσους πολλούς υπολογιστές, υπάρχει η ανάγκη να το ξεκινήσουμε σε κάθε υπολογιστή με τις αντίστοιχες παραμέτρους, για αυτόν και για τη θέση του στο δίκτυο. Η αρχικοποίηση μπορεί να γίνει κατά τη διάρκεια του φορτώματος (αν το πρωτόκολλο είναι συγχωνευμένο στο λειτουργικό σύστημα) ή με την κλήση του πρωτοκόλλου από κάποια εφαρμογή. Οι παράμετροι αυτοί μπορούν να οριστούν τοπικά για κάθε υπολογιστή ξεχωριστά. Κάτι τέτοιο όμως δημιουργεί αρκετά προβλήματα:

- Χρειάζεται πάρα πολύ εργασία από τον διαχειριστή του δικτύου η οποία είναι χρονοβόρα και επιρρεπής σε λάθη.

- Το να διατηρούνται οι παράμετροι ενημερωμένες χρειάζεται συνεχή δουλειά η οποία αυξάνεται γεωμετρικά με τις αλλαγές

που συμβαίνουν στο δίκτυο, ειδικά αν υπάρχουν υπολογιστές που αλλάζουν συνεχώς θέση (πχ φορητοί Η/Υ).

- Η αλλαγή μιας κοινής παραμέτρου κοινής για τους υπολογιστές σε ένα subnet (πχ τοπική διεύθυνση ενός router) απαιτεί αλλαγές σε κάθε υπολογιστή.
- Μερικά μηχανήματα μπορεί να λειτουργούν ως τερματικά αυτό σημαίνει ότι δεν έχουν αποθηκευτικό χώρο για να κρατήσουν τις ρυθμίσεις.
- Σε περιπτώσεις έλλειψης διευθύνσεων ή ενός δικτύου που αλλάζει συνέχεια θα ήταν χάσιμο χρόνου να δίνουμε σε έναν μη σταθερό υπολογιστή μόνιμη διεύθυνση. Μια καλύτερη προσέγγιση θα ήταν να χρησιμοποιούνται ομάδες διευθύνσεων από ομάδες υπολογιστών. Η χειροκίνητη ρύθμιση τέτοιου είδους δεν παρέχει εύκολο τρόπο για να γίνει αυτό.

Όλοι αυτοί οι λόγοι οδήγησαν στην ανάγκη για έναν αυτόματο μηχανισμό διαχείρισης των TCP/IP πρωτοκόλλων. Ο DHCP είναι σήμερα ο πιο προηγμένος μηχανισμός για να γίνεται αυτό. Έτσι, για να είναι η σύνδεση όσο το δυνατόν περισσότερο αυτοματοποιημένη και να μην γίνονται πολλές ρυθμίσεις από τη μεριά του πελάτη, ενεργοποιήσαμε τον DHCP Server του Coova-Chilli με την εξής διαδικασία:

Ανοίγουμε το αρχείο διαμόρφωσης που βρίσκεται στη θέση */usr/local/etc/chilli.conf* όπου θα ορίσουμε τις παραμέτρους για την σωστή διαχείριση του δικτύου μας.

```
HS_WANIF
```

Καθώς προσδιορίζει την κάρτα ενσύρματης δικτύωσης που θα συνδεθεί με το WAN (δίκτυο ευρείας περιοχής), βάζουμε την τιμή eth0 που

είναι και η πρώτη κάρτα του Η/Υ.

```
HS_LANIF= eth1
```

Έτσι ορίζουμε την δεύτερη κάρτα ως την «πόρτα» προς το δίκτυο των συνδρομητών.

```
HS_NETWORK=192.168.0.0
```

Εδώ εισάγουμε το δίκτυο με τις διευθύνσεις IP που θα χρησιμοποιήσουμε. Τέλος στην παράμετρο

```
HS_NETMASK
```

Εισάγουμε την «μάσκα» για να δημιουργήσουμε το υποδίκτυο που θα μας δώσει και το τελικό εύρος διευθύνσεων, πχ αν εισάγουμε την μάσκα 255.255.255.0 θα μας δώσει το υποδίκτυο από 192.168.0.1- 192.168.0.254.

```
HS_UAMLISTEN=192.168.0.1
```

Ορίζει την IP του hotspot στο δίκτυο των συνδρομητών. Πρέπει να είναι μέσα στο εύρος διευθύνσεων που ορίσαμε παραπάνω. Αυτή την διεύθυνση θα βλέπουν σαν «gateway» οι συνδρομητές.

```
HS_UAMPORT=3990
```

Η TCP/IP πόρτα του δικτύου των συνδρομητών.

```
HS_DNS1=194.177.198.2
```

Εδώ δίνουμε την τιμή της διεύθυνσης IP του DNS στο δίκτυο μας. Στην παράμετρο *HS_DNS2* ορίζουμε και δευτερεύον αν υφίσταται.

4.2.2 Παραμετροποίηση των υποσυστημάτων

Ο χρήστης στο Coova chilli [5] διακρίνεται σε δύο καταστάσεις: Πιστοποιημένος και μη πιστοποιημένος. Στην κατάστασή μη πιστοποίησης ο χρήστης ανακατευθύνεται στον web server πιστοποίησης. Στην περίπτωση που ο χρήστης είναι μη πιστοποιημένος μπορεί να έχει πρόσβασή σε μια λίστα πόρων πχ το site της εταιρείας ή του ιδρύματος που ανήκει το hotspot ή σε ένα site όπου θα μπορεί να πληρώσει για να του δοθούν συνθηματικά για πρόσβαση στις υπηρεσίες του Captive Portal. Στην δική μας υλοποίηση έχει δοθεί ελεύθερη πρόσβαση στο site του ΤΕΙ που από εκεί μπορεί ο χρήστης να συμπληρώσει μια φόρμα για να του δοθούν στοιχεία πρόσβασης. Αυτή η τεχνική ονομάζεται wallet gargen και χρησιμοποιεί και τον DNS για να δεί αν θέλουμε ολόκληρα domain πχ. **.teicrete.gr**

Η πιστοποίηση των χρηστών όπως προαναφέραμε διενεργείτε από τον Radius Server με την μέθοδο UAM η οποία δεν είναι τίποτα άλλο από μια σελίδα πιστοποίησης που αποστέλλετε από τον web server του Captive στον πελάτη και αντίστροφα με το ασφαλές πρωτόκολλο κρυπτογράφησης SSH. Για την επικοινωνία του Captive με τον Radius Server χρησιμοποιείτε το πρωτόκολλο κρυπτογράφησης CHAP για την αποστολή των στοιχείων του χρήστη και το κλειδί όπως αναφέρεται στην προδιαγραφή RFC 2865. Παρακάτω αναφέρονται οι ρυθμίσεις που δόθηκαν για επιτευχθεί αυτό.

```
HS_UAMSECRET=theuamsecret
```

Το μυστικό κλειδί που είναι κοινό ανάμεσα στον web server και το coova-chilli, για την αποστολή των στοιχείων πρόσβασης του χρήστη.

```
HS_RADIUS=127.0.0.1
```

Εδώ ορίζουμε την IP διεύθυνση του Radius Server, στην περίπτωση μας είναι εγκατεστημένος στο ίδιο μηχάνημα με το Coova-chilli έτσι εισάγουμε την localhost διεύθυνση δηλ την διεύθυνση της ίδιας της κάρτας δικτύου.

```
HS_RADIUS2=127.0.0.1
```

Ο δευτερεύον Radius Server σε περίπτωση FailOver. Δεν χρησιμοποιείται ωστόσο πρέπει να δηλωθεί.

```
HS_RADSECRET=c@ptive
```

Το Μυστικό κλειδί που απαιτείται για την επικοινωνία του Coova-chilli με τον Radius Server και την αποκρυπτογράφιση των μηνυμάτων.

```
HS_UAMALLOW=www.chania.teicrete.gr,www.google.gr,194.177.198.6
```

Εδώ εισάγουμε τις διευθύνσεις ή τα domain names των web server θέλουμε να βλέπουν οι χρήστες που δεν είναι πιστοποιημένοι.

```
HS_UAMSERVER=192.168.0.1
```

Η διεύθυνση του Web Server που αναλαμβάνει την πιστοποίηση που μπορεί να βρίσκεται και σε άλλο μηχάνημα εκτός του τοπικού δικτύου. Εδώ είναι εγκατεστημένος στο ίδιο μηχάνημα.

```
HS_UAMFORMAT=https://\$HS_UAMSERVER/cgi-bin/hotspotlogin.cgi
```

Μαζί με την παράμετρο `HS_UAMSERVER` Ορίζει την τελική διαδρομή του δικτύου για την σελίδα υποδοχής του web server το Captive Portal.

```
HS_DEFSESSIONTIMEOUT=0
```

Μπορούμε Να ορίσουμε το μέγιστο χρόνο που μπορεί ο χρήστης να είναι συνδεδεμένος στο δίκτυο, στο μηδέν αυτός ο χρόνος είναι απεριόριστος.

```
HS_DEFIDLETIMEOUT=600
```

Όταν ένας χρήστης έχει συνδεθεί αλλά η σύνδεση του είναι ανενεργή ορίζουμε ένα ελάχιστο χρόνο αποσύνδεσης για λόγους ασφαλείας.

Κεφάλαιο 5

Εγκατάσταση και παραμετροποίηση Freeradius Server

Το παρόν κείμενο περιέχει την τεκμηρίωση του εξυπηρετητή RADIUS Freeradius. Ο εξυπηρετητής αυτός ανήκει στην κατηγορία του ελεύθερου λογισμικού και παρέχει πολύ μεγάλες δυνατότητες, ευελιξία και ταχύτητα και μπορεί ικανοποιήσει με τον καλύτερο τρόπο και τις μεγαλύτερες απαιτήσεις μεγέθους.

Θεωρείται ότι ο αναγνώστης έχει μία γενική γνώση του πρωτοκόλλου RADIUS καθώς και του πρωτοκόλλου LDAP.

5.1 Δυνατότητες

- **Authentication μέσω LDAP:** Η πιστοποίηση των χρηστών γίνεται με τη χρήση κατάλληλων LDAP ερώτησεων.
- **Authorization LDAP:** Το authorization των χρηστών γίνεται και αυτό μέσω LDAP πρωτοκόλλου. Μέσω ενός attribute στον LDAP καθορίζεται η δυνατότητα πρόσβασης στην υπηρεσία τηλεφωνικής πρόσβασης.
- **Reply-Items στον LDAP:** οι παράμετροι σύνδεσης των χρηστών προσδιορίζονται από attributes που περιέχονται στο entry του χρήστη στον διακομιστή LDAP.
- **Ημερήσια, Εβδομαδιαία Όρια:** Μετρητές χρησιμοποιούνται για τον έλεγχο και επιβολή ημερήσιων και εβδομαδιαίων ορίων χρήσης του dial-up.

- **Υποστήριξη των μηχανισμών PAP, CHAP, MS-CHAP και EAP (EAP-MD5 και EAP-TLS):** Υποστηρίζονται όλες οι παραπάνω μέθοδοι αποστολής του συνθηματικού του χρήστη. Στην περίπτωση του PAP υποστηρίζεται πλήθος δυνατών μεθόδων κωδικοποίησης του συνθηματικού του χρήστη όπως cleartext, crypt, MD5 καθώς και SHA1.
- **Login-Time:** Καθορισμός του επιτρεπόμενου χρόνου σύνδεσης των χρηστών
- **Expiration Date:** Καθορισμός του χρόνου λήξης της ισχύος ενός συγκεκριμένου κωδικού χρήστη.
- **Double logins:** Ανίχνευση double-login (πολλαπλή πρόσβαση) των χρηστών
- **Authorization** με βάση τα Caller-Ids των χρηστών και τα IP addresses του access server στον οποίο συνδέονται (τοπική πρόσβαση)
- **Multi-threaded :** Ο ίδιος ο server καθώς και τα βασικά modules (LDAP,SQL κτλ) είναι πλήρως multithreaded με αποτέλεσμα τη δυνατότητα υποστήριξης σχεδόν απεριόριστου αριθμού αιτήσεων
- **Accounting σε mySQL βάση (όπως επίσης και σε πολλές άλλες βάσεις δεδομένων όπως Orcale, DB2, MSSQL κτλ).**
- **Web-based σύστημα διαχείρισης** (Daloradius)

5.2 Πιστοποίηση μέσω LDAP

Η πιστοποίηση των χρηστών γίνεται μέσω ενός bind request στον διακομιστή LDAP με το login/password που παρέχει ο χρήστης κατά την σύνδεση. Κατά αυτόν τον τρόπο υποστηρίζονται όλοι οι τρόποι κρυπτογράφησης του password του χρήστη τους οποίους υποστηρίζει ο διακομιστής LDAP. Επιπλέον παρέχεται η δυνατότητα η αποστολή του login και του password στον ldap server να γίνει μέσω ασφαλούς σύνδεσης SSL.

5.3 Διαδικασία Έγκρισης

Για το authorization των χρηστών χρησιμοποιείται το objectclass **radiusprofile**, το οποίο παρέχει τα διάφορα attributes που σχετίζονται με το authorization. Ένα τέτοιο attribute για παράδειγμα είναι το dialupaccess, το οποίο εάν είναι FALSE ο χρήστης δεν επιτρέπεται να χρησιμοποιήσει την υπηρεσία. Το objectclass επίσης περιέχει και άλλα attributes, τα οποία είναι είτε check items τα οποία ελέγχονται κατά το authorization του χρήστη, είτε reply items με παραμέτρους σύνδεσης που επιστρέφονται στον access server εάν το authentication επιτύχει (π.χ., session-timeout, idle-timeout).

Ο αναλυτικός ορισμός του radiusprofile παρουσιάζεται παρακάτω στον πίνακα 5.1:

LDAP Attribute	Περιγραφή	Τύπος (check/reply item)
radiusSimultaneousUse	Ορίζει τον μέγιστό αριθμό από ταυτόχρονες συνδέσεις (όχι multilink) που μπορούν να γίνουν από ένα συγκεκριμένο χρήστη. Συνήθως λαμβάνει την τιμή 1 προκειμένου να αποκλείονται περιπτώσεις double login	check
radiusAuthType	Ορίζει τον τύπο του authentication που θα εκτελεστεί από τον radius server	check
radiusExpiration	Ορίζει την ημερομηνία κατά την οποία θα λήξει η πρόσβαση του χρήστη (είναι της μορφής 20 May 2002)	check
dialupaccess	Ορίζει το κατα πόσο ο χρήστης έχει πρόσβαση στην υπηρεσία dialup. Αν έχει την τιμή FALSE τότε η πρόσβαση του χρήστη στην υπηρεσία δεν επιτρέπεται. Αν έχει οποιαδήποτε άλλη τιμή τότε η πρόσβαση επιτρέπεται	check
radiusHint		check

radiusLoginTime	Ορίζει το χρονικό διάστημα (σε UUCP format) κατά το οποίο μπορεί να συνδεθεί ο αντίστοιχος χρήστης στην υπηρεσία	check
radiusarapfeatures		
radiusarapsecurity		
radiusarapzoneaccess		
radiuscallbackid		
radiuscallbacknumber		
radiuscalledstationid	Το τηλέφωνο στο οποίο καλεί ο χρήστης (DNIS)	check
radiuscallingstationid	Το τηλέφωνο από το οποίο γίνεται η κλήση (CLID)	check
radiusclass		
radiusfilterid		
radiusframedappletalklink		
radiusframedappletalknetwork		
radiusframedappletalkzone		
radiusframedcompression	Το πρωτόκολλο συμπίεσης το οποίο θα εφαρμοστεί στη σύνδεση. Το πλέον διαδεδομένο και συνηθισμένο πρστόκολλο είναι το Van-Jacobson-TCP-IP	reply

radiusframedipaddress	Η διεύθυνση IP η οποία θα αποδωθεί στον χρήστη	reply
radiusframedipnetmask	Η IP netmask που θα αποδωθεί στο χρήστη	reply
radiusframedipxnetwork		reply
radiusframedmtu	Το MTU της σύνδεσης	reply
radiusframedprotocol	Το πρωτόκολλο της σύνδεσης (συνήθως είναι PPP)	reply
radiusframedroute		reply
radiusframedrouting		reply
radiusidletimeout	Ο μέγιστος χρόνος για τον οποίο επιτρέπεται η σύνδεση του χρήστη να μείνει ανενεργή (idle)	reply
radiusloginihost		reply
radiusloginlatgroup		reply
radiusloginlatnode		reply
radiusloginlatport		reply
radiusloginlatservice		reply
radiusloginservice		reply

radiuslogintcpport		reply
radiuspasswordretry		reply
radiusportlimit	Ο μέγιστος αριθμός απο διαθέσιμα κανάλια στον access server τα οποία μπορεί να ανοίξει ταυτόχρονα ο χρήστης σε μία mutlink σύνδεση	reply
radiusprompt		reply
radiusservicetype	Ο τύπος της σύνδεσης του χρήστη. Συνηθισμένες τιμές είναι Framed-User εφόσον η σύνδεση είναι τύπου Framed, Outbound-User για εξερχόμενες (από την πλευρά του access server) συνδέσεις και Login-User για συνδέσεις telnet	reply
radiussessiontimeout	Ο μέγιστος χρόνος που μπορεί να διαρκέσει η σύνδεση	reply
radiusterminationaction		reply
radiustunnelassignmentid		reply
radiustunnelclientendpoint		reply
radiustunnelmediumtype		reply
radiustunnelpassword		reply
radiustunnelpreference		reply
radiustunnelprivategroupid		reply

radiustunnelserverendpoint		reply
radiustunneltype		reply
radiusvsa		reply
radiuscheckitem	Γενικής φύσης attribute το οποίο μπορεί να χρησιμοποιηθεί για την αποθήκευση οποιουδήποτε check item. Η τιμή του πρέπει να είναι της μορφής <RADIUS attribute> <operator> <value> (πχ NAS-IP-Address := "194.63.239.238")	check
radiusreplyitem	Γενικής φύσης attribute το οποίο μπορεί να χρησιμοποιηθεί για την αποθήκευση οποιουδήποτε reply item. Η τιμή του πρέπει να είναι της μορφής <RADIUS attribute> <operator> <value> (πχ Cisco-AVPair := "lcp:send-secret=XXXXXX")	reply

Πίνακας 5.1

Οι ρυθμίσεις που αφορούν τον κάθε χρήστη ορίζονται με τον συνδυασμό τεσσάρων προφίλ, του **User-Profile**, του **Default-Profile**, του **Regular-Profile** και του **προσωπικού προφίλ**. Το User-Profile συνήθως ορίζεται μέσα στο αρχείο users με βάση ελέγχους στα radius attributes που περιέχονται στο Access-Request πακέτο. Ορίζει το DN ενός entry το οποίο περιέχει ρυθμίσεις για μία συγκεκριμένη κατηγορία χρηστών. Αν είναι ορισμένο τότε το Default-Profile δεν λαμβάνεται υπόψη. Κατά αυτό τον τρόπο μπορεί να επιλέγεται ένα γενικό profile για τους χρήστες αναλόγως με τον τύπο της αιτούμενης σύνδεσης ή με βάση άλλα κριτήρια τα οποία μπορεί να ορίσει ο διαχειριστής της υπηρεσίας. Έτσι, αιτήσεις με Service-Type=Framed-User θα έχουν διαφορετικές ρυθμίσεις (profiles) από τις αιτήσεις με Service-Type=Login-User. Το Default-Profile είναι ένα ξεχωριστό DN που ορίζεται στο αρχείο διαμόρφωσης του radius server και το οποίο (το default-profile entryt) περιέχει attributes που προσδιορίζουν τις προκαθορισμένες (default) ρυθμίσεις των χρηστών. Το Regular-Profile, είναι ένα attribute που περιέχεται

στα entries των χρηστών και το οποίο δείχνει σε κάποιο DN με ρυθμίσεις που αφορούν την ομάδα χρηστών στην οποία ανήκει ο χρήστης. Τέλος, το entry του κάθε χρήστη μπορεί να περιέχει attributes που διαφοροποιούν το προφίλ του συγκεκριμένου χρήστη από τις default ρυθμίσεις ή τις ρυθμίσεις ομάδας. Προφανώς, κατά το authorization του κάθε χρήστη οι παράμετροι της σύνδεσης προσδιορίζονται από τα τρία αυτά προφίλ, με σειρά φθίνουσας προτεραιότητας: προφίλ χρήστη, Regular-Profile, Default-Profile.

5.4 Παρακολούθηση

Για το accounting συνήθως χρησιμοποιείται μία βάση mysql η οποία αποθηκεύει την σχετική πληροφορία. Πιο συγκεκριμένα, για τον σκοπό αυτό χρησιμοποιείται ο πίνακας **radacct** ο οποίος αποθηκεύει ένα row πληροφορίας για κάθε dial-up session.

Αμέσως μετά την επιτυχή πιστοποίηση του χρήστη ο access server στέλνει ένα Accounting-Start πακέτο στο radius server το οποίο περιέχει βασικές πληροφορίες για τη σύνδεση (την πόρτα στην οποία συνδέθηκε ο χρήστης, ο αριθμός τηλεφώνου από τον οποίο συνδέθηκε κ.τ.λ.) το οποίο και χρησιμοποιείται για τη δημιουργία ενός νέου row στον πίνακα radacct το οποίο και περιέχει τις πληροφορίες αυτές. Μετά την αποσύνδεση του χρήστη, ο access server στέλνει ένα Accounting-Stop πακέτο το οποίο περιέχει πλήθος πληροφοριών για τη σύνδεση όπως την ip address, τα bytes που παραλήφθηκαν και στάλθηκαν κ.τ.λ. Οι πληροφορίες αυτές χρησιμοποιούνται για να ανανεωθεί η αντίστοιχη εγγραφή στον πίνακα radacct. Σε περίπτωση που έχει ενεργοποιηθεί κάτι τέτοιο στον access server μετά την επιτυχή σύνδεση του χρήστη ο access server, μπορεί να στείλει ένα Accounting-Update το οποίο να περιέχει πληροφορίες για το χρήστη οι οποίες δεν ήταν διαθέσιμες κατά την πιστοποίηση του (όπως πχ η IP address που του παραχωρήθηκε).

Σημειώνεται ότι εάν το session-time μιας σύνδεσης είναι 0, το session δεν θεωρείται επιτυχές και το αντίστοιχο row δεν αποθηκεύεται από την accounting διαδικασία στον πίνακα. Αυτό τυπικά συμβαίνει σε περιπτώσεις ανεπιτυχούς authentication. Πάντως, ο πίνακας (όπως αναφέρεται παρακάτω) μπορεί να περιέχει rows με session-time=0, τα οποία εισάγονται από άλλες διαδικασίες (bad login)

Η δομή του πίνακα radacct φαίνεται παρακάτω:

Πεδίο	Τύπος	Περιγραφή
RadAcctId	bigint(21)	Το ID του κάθε row
AcctSessionId	varchar(32)	Το Session ID της κάθε σύνδεσης
AcctUniqueId	varchar(32)	Αν έχει ενεργοποιηθεί το acct_unique module του freeradius το πεδίο αυτό θα περιέχει ένα μοναδικό id για τη σύνδεση αυτή
UserName	varchar(64)	Το username του χρήστη
Realm	varchar(64)	Το realm στο οποίο ανήκει ο χρήστης. Αν δεν ανήκει σε κάποιο τότε το πεδίο αυτό παραμένει κενό
NASIPAddress	varchar(15)	Η IP Address του access server στον οποίο συνδέθηκε ο χρήστης
NASPortId	int(12)	Η πόρτα του access server στην οποία συνδέθηκε ο χρήστης
NASPortType	varchar(32)	Ο τύπος της πόρτας στην οποία έγινε η σύνδεση
AcctStartTime	datetime	Ο χρόνος έναρξης της σύνδεσης

AcctStopTime	datetime	Ο χρόνος λήξης της σύνδεσης
AcctSessionTime	int(12)	Ο συνολικός χρόνος της σύνδεσης
AcctAuthentic	varchar(32)	
ConnectInfo_start	varchar(32)	
ConnectInfo_stop	varchar(32)	
AcctInputOctets	bigint(12)	Ο συνολικός αριθμός των bytes που εισήλθαν στον access server στη σύνδεση
AcctOutputOctets	bigint(12)	Ο συνολικός αριθμος των bytes που αποστάλθηκαν προς τον χρήστη
CalledStationId	varchar(30)	Ο αριθμός τηλεφώνου τον οποίο κάλεσε ο χρήστης
CallingStationId	varchar(30)	Ο αριθμός τηλεφώνου από τον οποίο έγινε η κλήση για την πραγματοποίηση της σύνδεσης
AcctTerminateCause	varchar(32)	Ο λόγος λήξης της σύνδεσης (πχ User-Request, Session-Timeout κτλ)
ServiceType	varchar(32)	Ο τύπος της σύνδεσης (px Framed-User, Outbound-User)
FramedProtocol	varchar(32)	Ο τύπος του πρωτοκόλλου που χρησιμοποιήθηκε εφόσον η σύνδεση ήταν τύπου Framed (συνήθως είναι PPP)
FramedIPAddress	varchar(15)	Η IP Address που αποδώθηκε στο χρήστη
AcctStartDelay	int(12)	Η καθυστέρηση που παρουσιάστηκε κατά την αποθήκευση του Accounting-Start
AcctStopDelay	int(12)	Η καθυστέρηση που παρουσιάστηκε κατά την αποθήκευση του Accounting-Stop

Πίνακας 5.2

5.5 Μετρητές

Το σύστημα παρέχει δυνατότητα μετρητών για authorization και accounting χρήση σε ωριαία, ημερήσια, εβδομαδιαία και μηνιαία βάση. Αυτή την στιγμή χρησιμοποιούνται μόνο ημερήσιοι και εβδομαδιαίοι μετρητές. Ο κάθε ένας από αυτούς τους μετρητές χρησιμοποιεί ένα attribute (**Max-Daily-Session, Max-Weekly-Session** αντίστοιχα) που ορίζει το μέγιστο όριο ημερισίου/εβδομαδιαίου χρόνου που δικαιούται ο κάθε χρήστης, το οποίο και περιέχεται στο entry του χρήστη στον LDAP [3]. Το attribute αυτό θα πρέπει να γίνεται map σε αντίστοιχο attribute στον LDAP το οποίο να έχει προστεθεί και στο LDAP schema του LDAP server.

Οι ημερήσιοι και εβδομαδιαίοι μετρητές χρησιμοποιούνται κατά το authorization για να επιστρέψουν το υπόλοιπο του ημερησίου/εβδομαδιαίου χρόνου του χρήστη. Κατά το authorization, η τιμή του ημερησίου/εβδομαδιαίου μετρητή αφαιρείται από το ημερήσιο/εβδομαδιαίο όριο. Με τον τρόπο αυτό υπολογίζεται και επιστρέφεται το Session-Timeout της σύνδεσης, που δηλώνει το μέγιστο χρόνο που επιτρέπεται να μείνει συνδεδεμένος ο χρήστης. Η τιμή του μετρητή ενημερώνεται όταν ο χρήστης κάνει logout προσθέτοντας στην προηγούμενη τιμή την χρονική διάρκεια του νέου session.

Ειδικές περιπτώσεις αποτελούν οι περιπτώσεις αλλαγής ημέρας/εβδομάδας. Συγκεκριμένα:

- Εάν ένας χρήστης κάνει login κάποια στιγμή πριν τις 00:00 και ο χρόνος που του απομένει είναι πέραν τις αλλαγής μέρας, το Session-Timeout υπολογίζεται ως ο χρόνος που απομένει ως τις 00:00 συν το Max-Daily-Session.
- Εάν ο χρήστης κάνει logout μετά την αλλαγή της ημέρας/εβδομάδας, ο μετρητής της μέρας/εβδομάδας τίθεται στο χρονικό διάστημα από την αλλαγή ως την στιγμή της αποσύνδεσης.
- Στις υπόλοιπες περιπτώσεις, οι μετρητές μηδενίζονται στην αλλαγή της ημέρας/εβδομάδας (στις 00:00 κάθε μέρας και στις 00:00 της Κυριακής αντίστοιχα).

Οι ημερήσιοι και εβδομαδιαίοι μετρητές κρατιούνται σε ξεχωριστό DBM αρχείο. Ουσιαστικά, πρόκειται για διαφορετικά instances του ίδιου module με διαφορετικό configuration προκειμένου το ένα να κάνει reset ανά μήνα και το άλλο ανά εβδομάδα.

5.6 Εγκατάσταση – Παραμετροποίηση

Η εγκατάσταση της υπηρεσίας περιλαμβάνει δύο σκέλη: το compilation του distribution με ενεργοποίηση των απαραίτητων επιλογών για υποστήριξη ldap authentication/authorization και mySQL accounting και την παραμετροποίηση των αρχείων διαμόρφωσης του server.

5.6.1 Compilation

Η διαδικασία compilation του radius server είναι σχετικά απλή. Αφού γίνει μεταφόρτωση του distribution του radius server στον αρχικό κατάλογο εκτελούμε το αρχείο configure ως εξής:

./configure --prefix=/usr/local/radiusd –with-localstatedir=/var/radiusd --with-rlm-ldap-lib-dir=/usr/local/openldap/lib --with-rlm-ldap-include-dir=/usr/local/openldap/include --with-mysql-lib-dir=/usr/local/mysql/lib/mysql --with-mysql-include-dir=/usr/local/mysql/include –enable-snmp

Η αποστολή των παραμέτρων που δίνονται ως ορίσματα προς το configure είναι προφανής. Στη συνέχεια τρέχουμε τις εντολές make και make install. Εφόσον δεν υπάρξει πρόβλημα ο radius server θα έχει εγκατασταθεί κάτω από τον κατάλογο /usr/local/radiusd, με τα log files να περιέχονται στον κατάλογο /var/radiusd.

Στη συνέχεια από τον κατάλογο src/modules/rlm_sql/drivers/rlm_sql/mysql χρησιμοποιούμε το αρχείο db_mysql.sql για να προσθέσουμε τους ορισμούς των πινάκων της SQL βάσης του radius server στον mysql server. Προηγουμένως θα πρέπει να έχει

προφανώς δημιουργηθεί η βάση radius στον SQL server.

5.6.2 Δομή Εγκατάστασης

radiusd/:

radiusd/bin: Περιέχει τα διάφορα προγράμματα αλληλεπίδρασης με το radius server

radiusd/etc:

radiusd/etc/raddb: Περιέχει όλα τα αρχεία διαμόρφωσης του radius server

radiusd/lib: Περιέχει τα modules του radius server με τη μορφή DLLs.

radiusd/man: Περιέχει τις man pages για το radius server καθώς και για τα προγράμματα που περιέχονται στον κατάλογο bin

radiusd/man/man1:

radiusd/man/man5:

radiusd/man/man8:

radiusd/sbin: Στον κατάλογο αυτό περιέχεται το εκτελέσιμο του radius server.

/var/radiusd: Στον κατάλογο αυτό καταγράφονται τα log files της υπηρεσίας.

5.7 Παραμετροποίηση των αρχείων διαμόρφωσης

Η παραμετροποίηση των αρχείων διαμόρφωσης της υπηρεσίας περιλαμβάνει την διαμόρφωση των εξής σημείων στα παρακάτω αρχεία:

clients.conf: Στο αρχείο αυτό προστίθενται εγγραφές για κάθε radius client ο οποίος απαιτείται να κάνει ερωτήσεις στο radius server. Κάθε εγγραφή είναι της μορφής:

client 147.102.220.5 {

secret = mysecret

shortname = client1

nastype = other

}

Το nastype περιέχει τον τύπο του access server. Η παράμετρος αυτή είναι απαραίτητη κατά τον έλεγχο πολλαπλής πρόσβασης (double login detection), καθώς αυτή προσδιορίζει τον τύπο του access server και άρα την μέθοδο με την οποία θα ερωτηθεί προκειμένου να προσδιοριστεί κατά πόσον ο χρήστης είναι ήδη συνδεδεμένος. Σε περίπτωση που ο access server είναι cisco, μπορεί να δωθεί η τιμή cisco στην παράμετρο nastype. Αν ο access server είναι άλλου τύπου, ο διαχειριστής θα πρέπει να συμβουλευτεί τα σχόλια που περιλαμβάνονται στο αρχείο προκειμένου να βρει τον τύπο που πρέπει να χρησιμοποιήσει.

Είναι απαραίτητο να υπάρχει τουλάχιστον μία εγγραφή για κάθε access server, ο οποίος θα χρησιμοποιεί το radius server για AAA. Επιπλέον καλό είναι να προστεθεί ένα entry για το localhost για την εύκολη

πραγματοποίηση πειραμάτων καθώς και ένα entry για τον υπολογιστή στον οποίο είναι εγκατεστημένη η εφαρμογή web διαχείρισης dialup_admin προκειμένου να είναι δυνατή η λειτουργία των σελίδων ‘Check Server’ και ‘Test User’.

snmp.conf: Στο αρχείο αυτό γίνεται η παραμετροποίηση του snmp interface που παρέχει ο radius server. Ουσιαστικά απαιτείται μόνο η προσθήκη μίας εγγραφής για το smux server στον οποίο θα συνδεθεί ο radius server ως εξής:

smux_password = sec

sql.conf: Στο αρχείο αυτό γίνεται η παραμετροποίηση του sql module του radius server. Παρακάτω παρατίθενται οι παράμετροι οι οποίοι θα πρέπει να αλλαχθούν για τη σωστή λειτουργία του sql module:

 # Database type

 # Current supported are: rlm_sql_mysql, rlm_sql_postgresql, rlm_sql_iodbc, rlm_sql_oracle

driver = "rlm_sql_mysql"

 # Connect info

server = "localhost"

login = "root"

password = ""

 # Database table configuration

radius_db = "radius"

 # If you want both stop and start records logged to the

 # same SQL table, leave this as is. If you want them in

 # different tables, put the start table in acct_table1

 # and stop table in acct_table2

acct_table1 = "radacct"

acct_table2 = "radacct"

authcheck_table = "radcheck"

authreply_table = "radreply"

groupcheck_table = "radgroupcheck"

groupreply_table = "radgroupreply"

usergroup_table = "usergroup"

 # Remove stale session if checkrad does not see a double login

deletestalesessions = yes

 # Print all SQL statements when in debug mode (-x)

sqltrace = yes

sqltracefile = ${logdir}/sqltrace.sql

 # number of sql connections to make to server

num_sql_socks = 5

Οι κύριες παράμετροι οι οποίες θα πρέπει να αλλαχθούν είναι το

server, το οποίο ορίζει τον SQL server στον οποίο θα συνδεθεί το module, το *login,* το *password* που ορίζει το login και το password με το οποίο θα γίνει η σύνδεση στον SQL server, το *radius_db* που ορίζει το όνομα της βάσης του radius server (κανονικά θα πρέπει να παραμείνει το όνομα radius) και το *num_sql_socks* που ορίζει τον αριθμό των συνδέσεων τις οποίες θα διατηρεί το sql module με τον SQL server. Όσο μεγαλύτερη κίνηση προβλέπεται να έχει ο radius server, τόσο μεγαλύτερη θα πρέπει να τεθεί η μεταβλητή αυτή. Θα πρέπει να είναι κατ' ελάχιστο ίση με τον αριθμό των radius server threads που έχουν οριστεί στο radiusd.conf.

naspasswd: Στο αρχείο αυτό προστίθεται μία γραμμή για κάθε access server. Το αρχείο αυτό επίσης χρησιμοποιείται κατά τον έλεγχο των double logins και η μορφή του είναι ως εξής:

server- name SNMP community

όπου *server-name* είναι η IP ή το domain name του access server και *community* είναι το SNMP community για RO SNMP κλήσεις στον access server αυτόν.

radiusd.conf: Το βασικό αρχείο διαμόρφωσης του radius server. Το αρχείο περιέχει ένα μεγάλο όγκο από χρήσιμα σχόλια τα οποία βοηθούν στην κατανόηση του προορισμού της κάθε παραμέτρου. Ιδιαίτερη προσοχή θα πρέπει να δοθεί στα εξής σημεία στο αρχείο:

```
ldap ldap1{
        server = "nic450.att.sch.gr"
        identity = "cn=Directory Manager"
        password = "XXXXXX"
```

basedn = "dc=sch,dc=gr"

filter = "(uid=%{Stripped-User-Name:-%{User-Name}}*)"*

default_profile = "uid=default-dialup,ou=people,dc=sch,dc=gr"

access_attr = "dialupAccess"

profile_attribute = "dialupRegularProfile"

dictionary_mapping = ${raddbdir}/ldap.attrmap

timeout = 4

timelimit = 3

net_timeout = 1

ldap_debug = 0x0000

ldap_connections_number = 5

#password_header = "{clear}"

#password_attribute = userPassword

#groupname_attribute= cn

#groupmembership_filter="(|(&(objectClass=GroupOfNames)(member=%{Ldap-UserDn}))(&(objectClass=GroupOfUniqueNames)(uniquemember=%{Ldap-UserDn})))"

#groupmembership_attribute = radiusGroupName

}

Στο σημείο αυτό γίνεται η διαμόρφωση του ldap module. Η χρήση των παραπάνω παραμέτρων είναι η εξής:

- *server:* Ο ldap server στον οποίο θα συνδεθεί το ldap module
- *identity,password:* Το DN/Password με το οποίο θα κάνει bind το ldap

module στον ldap server.

- *basedn:* Το DN το οποίο θα χρησιμοποιηθεί ως base στις αναζητήσεις που πραγματοποιούνται στον ldap.
- *filter:* Το φίλτρο με το οποίο γίνονται οι αναζητήσεις. Η παράμετρος '%{Stripped-User-Name:-%{User-Name}}' αντικαθίσταται κάθε φορά απο το username του χρήστη.
- *default_profile:* Το DN του Default-Profile entry που περιγράφηκε παραπάνω.
- *access_attr:* Το attribute το οποίο ορίζει την δυνατότητα του χρήστη να χρησιμοποιήσει την υπηρεσία dialup. Στην περίπτωση που το attribute αυτό δεν υπάρχει στο entry του χρήστη η πρόσβαση δεν επιτρέπεται.
- *profile_attribute:* Το attribute στο entry του κάθε χρήστη το οποίο και δείχνει στο DN του Regular-Profile entry το οποίο περιγράφηκε παραπάνω. Δεν είναι απαραίτητο το attribute αυτό να υπάρχει στο entry του χρήστη.
- *dictionary_mapping:* Ορίζει που βρίσκεται το αρχείο στο οποίο γίνονται map τα RADIUS attributes σε LDAP attributes. Το αρχείο αυτό είναι της μορφής:
- *ldap_connections_number:* Ο αριθμός των συνδέσεων οι οποίες θα γίνουν με τον εξυπηρετητή LDAP. Θα πρέπει κανονικά να είναι ίσο με τον αριθμό των server threads που έχει οριστεί πιο πριν στο radiusd.conf
- *password_attribute:* Το LDAP attribute το οποίο περιέχει το password του user entry. Αν δεν είναι κενό τότε γίνεται extraction του password προκειμένου να χρησιμοποιηθεί κατά τη φάση του authentication (για παράδειγμα από τα chap και pap modules).
- *password_header:* Αν δεν είναι κενό τότε ορίζει το password header το οποίο θα πρέπει να παραληφθεί κατά το extraction του password.

counter daily{

```
        filename = ${localstatedir}/db.day
        key = User-Name
        count-attribute = Acct-Session-Time
        reset = daily
        counter-name = Daily-Session-Time
        check-name = Max-Daily-Session
        allowed-servicetype = Framed-User
}
```

Στο σημείο αυτό γίνεται η διαμόρφωση του counter module το οποίο και εφαρμόζει τα ημερήσια και εβδομαδιαία όρια χρήσης. Παραπάνω, φαίνεται η διαμόρφωση του module για ημερήσια όρια χρήσης. Η χρήση των παραπάνω παραμέτρων έχει ως εξής:

- *filename:* Το αρχείο στο οποίο διατηρούνται οι μετρητές για κάθε χρήστη.
- *reset:* Κάθε πότε γίνεται η αρχικοποίηση των μετρητών. Μπορεί να πάρει τις τιμές ‘daily’,’weekly’ και ‘monthly’.
- *check-name:* Το attribute το οποίο ορίζει το όριο χρήσης. Μπορεί να πάρει τις τιμές ‘Max-Daily-Session’,’Max-Weekly-Session’,’Max-Monthly-Session’.

```
        # PAP module to authenticate users based on their stored password
        #
        #  Supports multiple encryption schemes
        #  clear: Clear text
        #  crypt: Unix crypt
        #    md5: MD5 ecnryption
```

 # sha1: SHA1 encryption.

 # DEFAULT: crypt

pap {

encryption_scheme = crypt

}

Στο σημείο αυτό γίνεται η διαμόρφωση του pap module το οποίο και επιτρέπει τον έλεγχο του password του χρήστη με ένα ήδη διαθέσιμο κρυπτογραφημένο password (το οποίο έχει εξαχθεί για παράδειγμα μέσω του ldap module από την εγγραφή του χρήστη). Υποστηρίζεται μόνο μία παράμετρος:

- *encryption_scheme:* Το σχήμα κρυπτογράφησης που χρησιμοποιείται. Μπορεί να είναι *clear* για απουσία κρυπτογράφησης, *crypt* για τη *unix* crypt, *md5* για MD5 και *sha1* για SHA1.

chap {

}

Στο σημείο αυτό γίνεται η διαμόρφωση του chap module το οποίο και επιτρέπει την πιστοποίηση χρηστών που συνδέονται με χρήση του πρωτοκόλλου CHAP εφόσον είναι διαθέσιμο το password του χρήστη σε clear text μορφή (έχει εξαχθεί για παράδειγμα μέσω του ldap module από την εγγραφή του χρήστη).

checkval nas-check{

item-name = "NAS-IP-Address"

check-name = "NAS-IP-Address"

```
        data-type = "ipaddr"
    }
```

Στο σημείο αυτό γίνεται η διαμόρφωση του checkval module το οποίο και επιτρέπει την εξουσιοδότηση ενός χρήστη, μόνο αν ένα attribute στο Access-Request από τον access server περιέχει μία συγκεκριμένη τιμή. Κατά αυτόν τον τρόπο είναι δυνατή η εξουσιοδότηση ενός χρήστη μόνο στην περίπτωση που καλεί από ένα συγκεκριμένο τηλέφωνο ή προσπαθεί να συνδεθεί σε ένα συγκεκριμένο access server. Η χρήση των παραμέτρων έχει ώς εξής:

item-name: Το attribute το οποίο θα αναζητηθεί για να γίνει η σύγκριση. Εάν το attribute αυτό δεν περιέχεται στο Access-Request τότε επιτρέπεται η σύνδεση.

check-name: Το attribute με την τιμή του οποίου θα γίνει η σύγκριση. Το attribute αυτό θα πρέπει να περιέχεται στο entry του χρήστη στο ldap και γίνεται διαθέσιμο αυτόματα από το ldap module. Αν το attribute αυτό είναι διαθέσιμο πολλές φορές, ο έλεγχος γίνεται για κάθε τιμή που είναι διαθέσιμη.

data-type: Ο τύπος των δεδομένων. Μπορεί να είναι ‘string’ για κείμενο, ‘integer’ για νούμερο, ‘ipaddr’ για IP address, ‘date’ για ημερομηνία και ‘octets’ για binary πληροφορία.

Πιθανές περιπτώσεις όπου μπορεί να χρησιμοποιηθεί το module αυτό είναι οι παρακάτω:

- Την εξουσιοδότηση ενός χρήστη μόνο αν συνδέεται από ένα (ή περισσότερους) τηλεφωνικό αριθμό. Για τον σκοπό αυτό κάθε user entry θα πρέπει να περιέχει ένα ή περισσότερα attributes radiuscallingstationid τα οποία περιέχουν τα νούμερα από τα οποία επιτρέπεται να συνδέεται ο χρήστης.

- Την εξουσιοδότηση ενός χρήστη μόνο αν συνδέεται σε ένα συγκεκριμένο access server. Για κάθε χρήστη προστίθενται δύο attributes στο entry του:

  radiusCheckItem=NAS-IP-Address := "194.63.239.238"

  radiusCheckItem=NAS-IP-Address := "255.255.255.255"

Το πρώτο attribute περιέχει την IP διέυθυνση του access server από τον οποίο μπορεί να συνδέεται ο χρήστης, ενώ το δεύτερο χρησιμοποιείται για να μπορεί να λειτουργεί η σελίδα ‘Test User’ στο περιβάλλον daloRadius.

```
instantiate{
        daily
        weekly
}
authenticate {
      authtype FailOver {
        redundant{
            ldap1
            ldap2
        }
      }
      authtype LDAP {
        redundant{
            ldap1
            ldap2
        }
```

```
        }
        authtype CHAP {
                chap
        }
}
authorize {
        preprocess
        chap
        suffix
        daily
        weekly
        files
        redundant {
                ldap1
                ldap2
        }
        daily
        weekly
        nas-check
        callerid-check
}

# Pre-accounting. Look for proxy realm in order of realms, then
# acct_users file, then preprocess (hints file).
preacct {
        suffix
```

```
        files
        preprocess
}

# Accounting. Log to detail file, and to the radwtmp file, and maintain
# radutmp.
accounting {
        sql
        daily
        weekly
}

# Session database, used for checking Simultaneous-Use. The radutmp or the sql
# modules handle this
session {
        sql
}
postauth {
          ippool
}
```

Στο σημείο αυτό ορίζεται ποιά modules συμμετέχουν σε κάθε βασική λειτουργία του radius server. Πιο συγκεκριμένα οι λειτουργίες του radius server είναι οι εξής:

- *instantiate:* Η αρχικοποίηση των modules. Στο section αυτό θα πρέπει

να προστεθούν modules τα οποία θα πρέπει να αρχικοποιηθούν πριν από τα υπόλοιπα όπως για παράδειγμα το counter module που πρέπει να κάνει register το check-name attribute.

- *authenticate:* Ο έλεγχος του password του χρήστη. Στην παραπάνω περίπτωση για τον έλεγχο αυτό χρησιμοποιείται το ldap module. Αντί του ldap module είναι δυνατόν να χρησιμοποιείται το pap module εφόσον έχει διαμορφωθεί το ldap module να κάνει εξαγωγή του password του χρήστη κατά την φάση του authorization.
- *authorize:* Στο σημείο αυτό γίνεται ο έλεγχος της πρόσβασης του χρήστη (η πρόσβαση επιτρέπεται από το dialupaccess attribute, ο χρήστης δεν έχει υπερβεί το μέγιστο ημερήσιο/εβδομαδιαίο χρόνο χρήσης κτλ) καθώς και η προσθήκη των απαραίτητων radius attributes που θα περιέχονται στην απάντηση στον access server (session-timeout, service-type κτλ).
- *accounting:* Η καταγραφή πληροφοριών για τα sessions των χρηστών με βάση τις πληροφορίες που περιέχονται στα Accounting-Start και Accounting-Stop πακέτα.
- *session:* Στο σημείο αυτό γίνεται ο έλεγχος αν απαιτείται για πολλαπλή πρόσβαση του χρήστη (ο έλεγχος γίνεται μόνο αν είναι ορισμένο το Simultaneous-Use attribute για το χρήστη).
- *postauth:* Η λειτουργία αυτή εκτελείται μετά την επιτυχή πιστοποίηση του χρήστη. Εδώ προστίθενται modules όπως το ippool module το οποίο χρησιμοποιείται για server side ip pool management.

Το keyword redundant είναι ειδικό και χρησιμοποιείται προκειμένου να υλοποιείται μηχανισμός fallback. Όσα modules περιέχονται σε ένα redundant section θα δοκιμάζονται το ένα μετά το άλλο μέχρι ένα από αυτά να μην αποτύχει.

users: Το αρχείο στο οποίο ορίζονται οι γενικές περιπτώσεις authentication. Προκειμένου να γίνεται επεξεργασία του αρχείου αυτού θα πρέπει να είναι

ενεργοποιημένο το module files και να έχει προστεθεί στο authorization section. Περισσότερες πληροφορίες για τη διαμόρφωση του αρχείου υπάρχουν στο αντίστοιχο manpage καθώς και στο sample αρχείο που περιέχεται στο distribution. Σε γενικές γραμμές το αρχείο θα πρέπει να είναι της μορφής:

DEFAULT Auth-Type = FailOver, Simultaneous-Use := 1

Fall-Through = 1

DEFAULT Service-Type == NAS-Prompt-User

Fall-Through = 1

DEFAULT Service-Type == Framed-User, Framed-Protocol == PPP, User-Profile := "uid=default-dialup,ou=people,dc=sch,dc=gr"

Port-Limit = 1,

Session-Timeout = 14400,

Idle-Timeout = 600,

Service-Type = Framed-User,

Framed-Protocol = PPP,

Framed-Compression = Van-Jacobson-TCP-IP,

Framed-IP-Address = 255.255.255.254,

Framed-IP-Netmask = 255.255.255.255,

Framed-MTU = 1500

DEFAULT Service-Type == Framed-User, Framed-Protocol == SLIP

Framed-Protocol = SLIP

DEFAULT Service-Type == Outbound-User, User-Profile := ""

5.8 Εργαλείο διαχείρισης Radius Server daloRadius

Το daloRadius είναι μια εξελιγμένη web εφαρμογή διαχείρισης που σκοπός της είναι η διαχείριση εφαρμογών hotspot και γενικά σε παροχείς internet. Το χαρακτηριστικό του γνώρισμα είναι η διαχείριση χρηστών, παρουσίαση γραφικών αναφορών, παρακολούθηση με λίγα λόγια ένας μηχανισμός χρέωσης των πελατών συν τις άλλοις ενσωματώνει και την υπηρεσία Googlemaps για γεωγραφικό εντοπισμό των σταθμών.

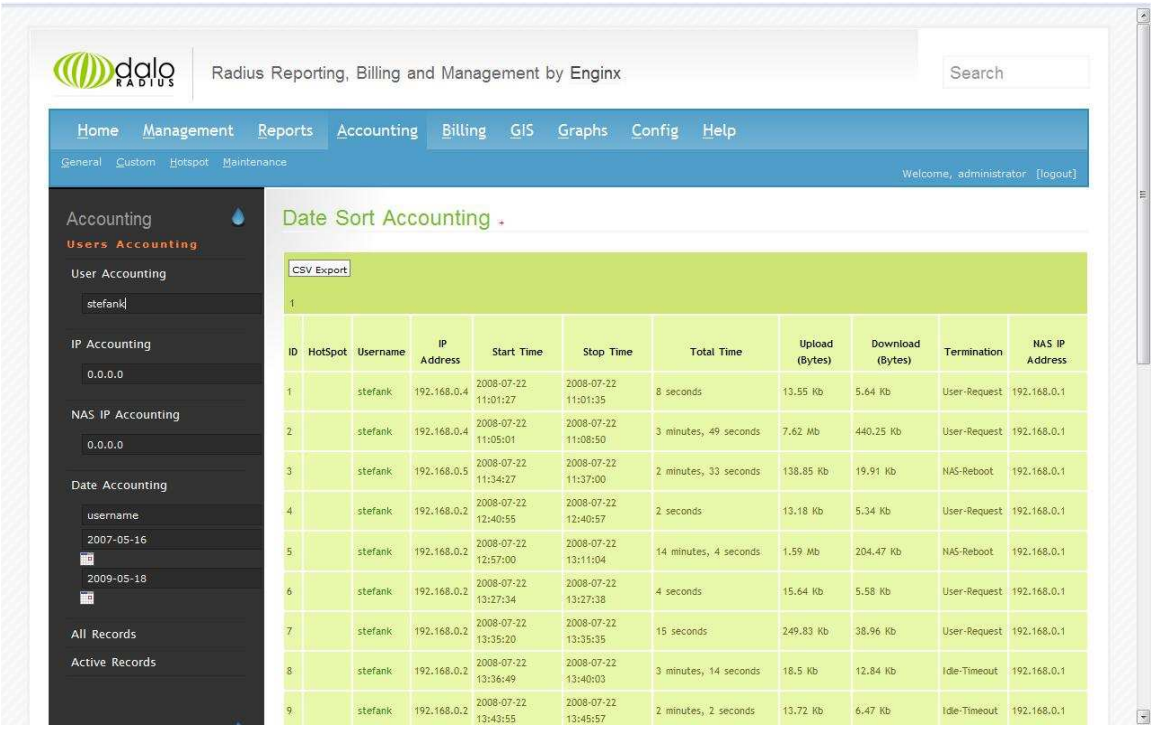
5.8.1 Τεχνικά χαρακτηριστικά

Η εφαρμογή Daloradius είναι γραμμένη με τη γλώσσα προγραμματισμού php και javascript και μπορεί να αξιοποιήσει την βάση δεδομένων Mysql. Η εφαρμογή αναπτύσσεται πάνω σε μια εγκατάσταση freeRadius Server έχοντας ως κορμό μία βάση δεδομένων Mysql.

Στην υλοποίηση που έγινε για το ΤΕΙ Χανίων ορισμένα από τα χαρακτηριστικά της εφαρμογής δεν χρησιμοποιήθηκαν μιας και η χρήστες μας βρίσκονται στην υπηρεσία καταλόγου LDAP και η διαχείριση γίνεται κεντρικά από άλλο τμήμα. Τα χαρακτηριστικά του Daloradius που χρησιμοποιήθηκαν αναφέρονται στην διαχείριση μέσω web της Mysql βάσης του FreeRadius Server και αφορούν την παρακολούθηση της κίνηση των χρηστών. Ορισμένα από αυτά αναφέρονται παρακάτω.

5.8.2 Παρακολούθηση χρηστών μέσω username

Στην παρακολούθηση μέσω username εισάγουμε το username του χρήστη και η εφαρμογή μας επιστρέφει όλο το ιστορικό χρήσης του δικτύου από ένα ή περισσότερα Captive Portal (hotspots) βλέπε σχήμα 5.1



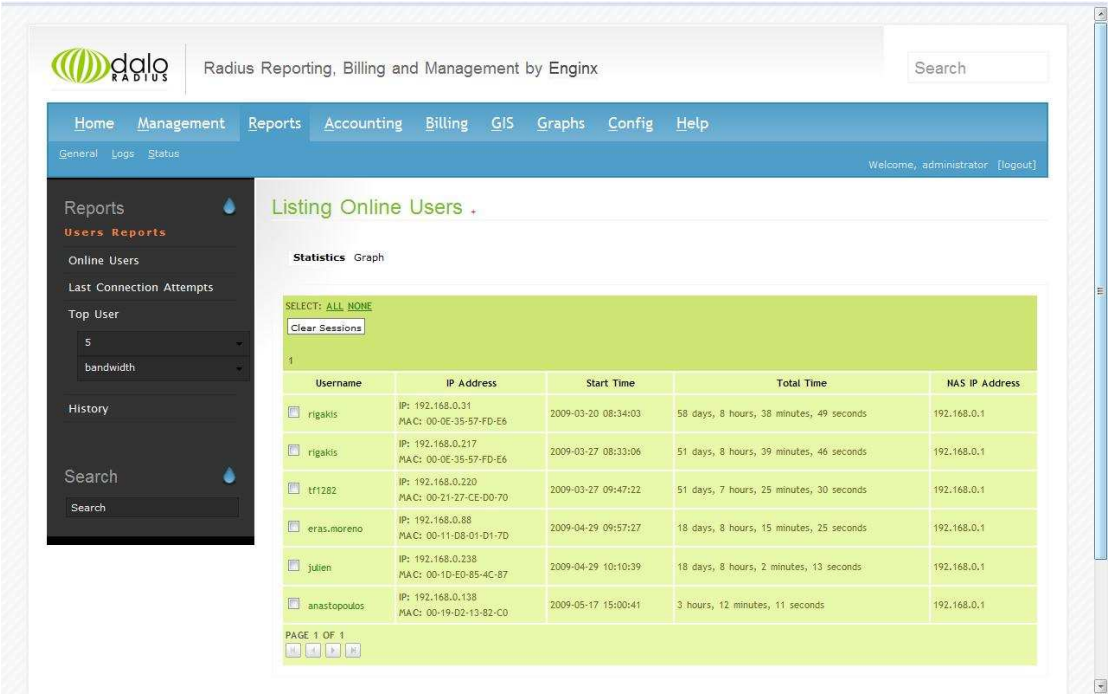
Σχήμα 5.1

Εδώ βλέπουμε ότι εισάγοντας το username του χρήστη μας επιστέφει το χρόνο που άρχισε η σύνδεση του χρήστη, το χρόνο που τελείωσε, το συνολικό χρόνο, τον εισερχόμενο όγκο δεδομένων, τον εξερχόμενο καθώς και

το λόγο που τερματίστηκε η σύνδεση.

5.8.3 Έλεγχος συνδεδεμένων χρηστών

Πηγαίνοντας στο μενού Reports > online users μπορούμε να δούμε ποιοι χρήστες βρίσκονται συνδεδεμένοι και σε ποιο NAS (διακομιστή πρόσβασης) ή Captive Portal. Σχήμα 5.2



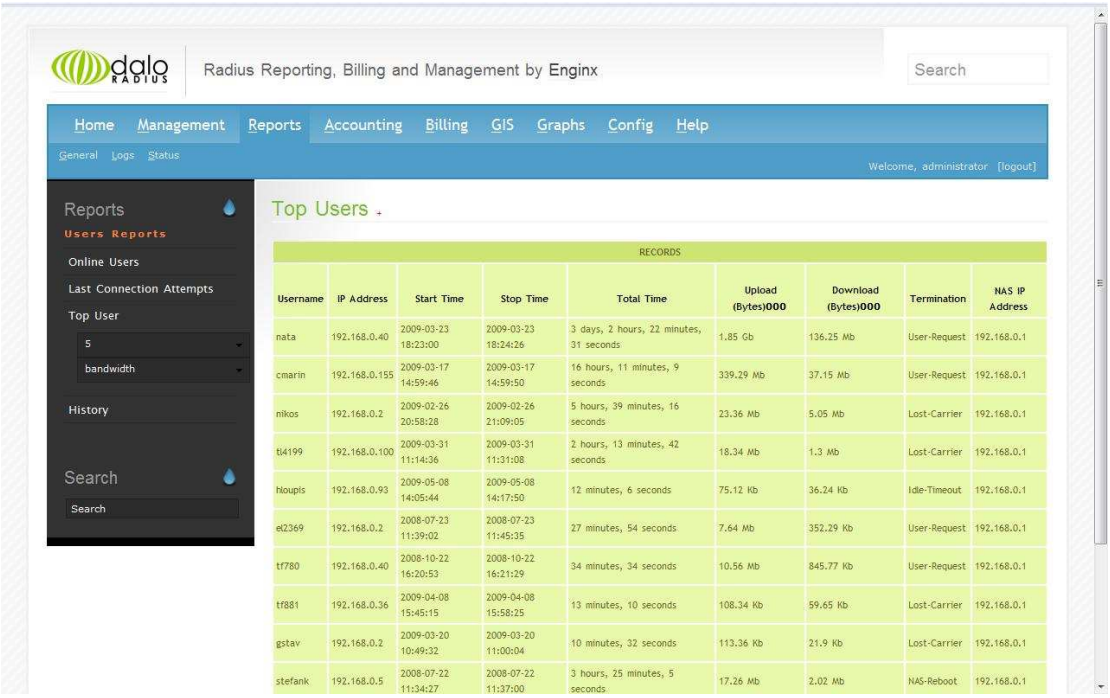
Σχήμα 5.2

Η Σελίδα αυτή μας δείχνει τους online χρήστες, την διεύθυνση IP που έχουν πάρει στο δίκτυο του Captive Portal, το χρόνο έναρξης της σύνδεσης

καθώς και τον συνολικό χρόνο που είναι online.

5.8.4 Αναζήτηση με όγκο χρήσης δεδομένων

Η εφαρμογή Daloradius με βάση τη γλώσσα προγραμματισμού για web εφαρμογές php κάνει επεξεργασία των δεδομένων της βάσης προσφέροντας πολλές δυνατότητες στον διαχειριστή του συστήματος. Μια από αυτές τις δυνατότητες είναι και η επιλογή που δίνει για την παρακολούθηση των χρηστών που κάνουν υπερβολική χρήση των πόρων του δικτύου όπως βλέπουμε στο σχήμα 5.3

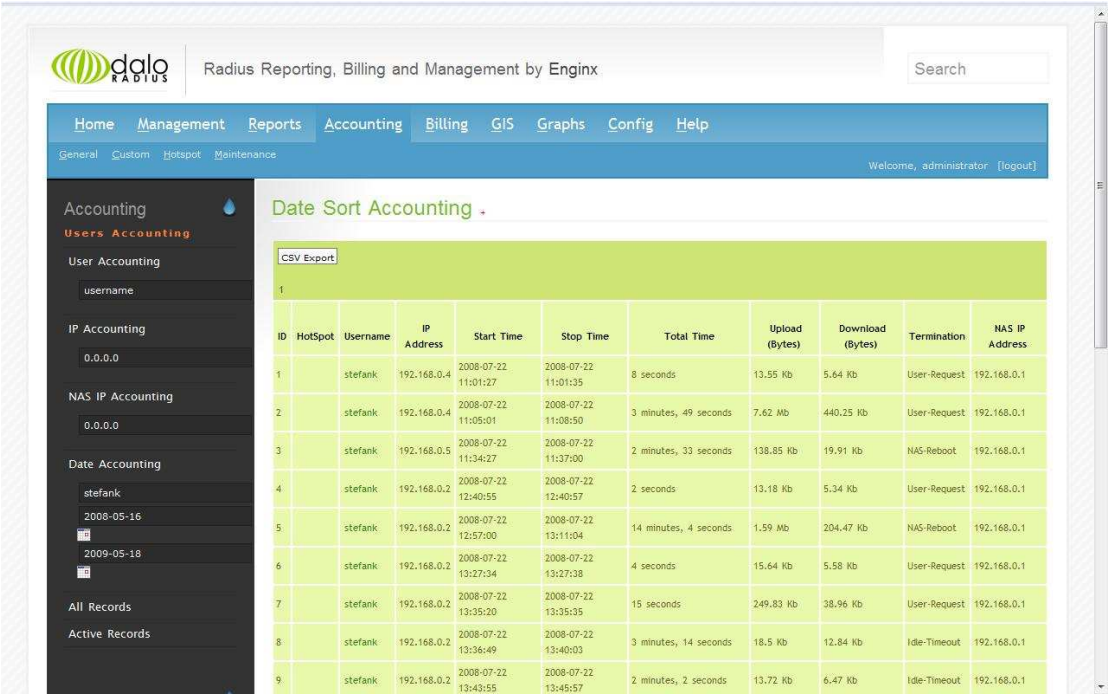


Σχήμα 5.3

Η σελίδα αυτή εκτός από τα στοιχεία που είδαμε στις προηγούμενες σελίδες μας παρέχει τη δυνατότητα να παρακολουθήσουμε το συνολικό όγκο χρήσης.

5.8.5 Έλεγχος με ημερομηνία

Τέλος μια πολύ χρήσιμη δυνατότητα που έχουμε μέσω του daloradius είναι ο έλεγχος ενός χρήστη με βάση την ημερομηνία σύνδεσης, Αυτό είναι πολύ χρήσιμο σε περιπτώσεις που δεν συμφωνεί ο πελάτης με μια χρέωση ή όταν θέλουμε να δούμε ποιοι χρήστες ήταν online σε μια δεδομένη χρονική στιγμή. Σχήμα 5.4



Σχήμα 5.4

Η αναζήτηση μέσω ημερομηνίας μπορεί να αποδώσει πολλά ενδιαφέροντα στοιχεία καθώς γίνεται εισάγοντας ένα εύρος ημερομηνίας και

όχι μόνο μια συγκεκριμένη ημέρα.

ΚΕΦΑΛΑΙΟ 6

Έλεγχος λειτουργίας Προτάσεις - Συμπεράσματα για την υλοποίηση στο Τ.Ε.Ι Κρήτης Παράρτημα Χανίων

6.1 Στρατηγική ελέγχων

Ο έλεγχος της καλής λειτουργίας της υπηρεσίας, ουσιαστικά περιλαμβάνει την πιστοποίηση ότι οι radius servers τρέχουν κανονικά και ότι επιστρέφουν τα σωστά attributes για τους χρήστες. Εφόσον η υπηρεσία στηρίζεται για την πιστοποίηση των χρηστών και για την καταγραφή του accounting σε εξωτερικές υπηρεσίες (όπως η υπηρεσία ldap και οι διακομιστές mysql), είναι απαραίτητο αν εμφανίζονται προβλήματα να ερευνάται η πιθανότητα να υπάρχει πρόβλημα με κάποια από αυτές τις υπηρεσίες.

Σε περίπτωση που οι υπηρεσίες αυτές λειτουργούν κανονικά και συνεχίζει να παρουσιάζεται πρόβλημα στη λειτουργία της υπηρεσίας, μπορεί να χρησιμοποιηθεί η λειτουργία ‘Check Server’ που παρέχεται από το εργαλείο web διαχείρισης (daloradius). Η λειτουργία αυτή θα πρέπει να είναι επιτυχής και να επιστρέψει τις σωστές τιμές για τα διάφορα radius attributes.

Σε περίπτωση που είναι ανεπιτυχής θα πρέπει να ελεγχθεί ότι η διεργασία του radius server τρέχει κανονικά και αν απαιτηθεί να γίνει επανεκκίνηση της.

Σε περίπτωση που η λειτουργία επιστρέφει λάθος radius attributes θα πρέπει να ελεγχθεί είτε αν λειτουργεί κανονικά η υπηρεσία καταλόγου, είτε αν έχει γίνει κάποια αλλαγή στη διαμόρφωση της υπηρεσίας που δεν έχει

ανακοινωθεί. Πέραν των παραπάνω, καλό είναι να γίνεται έλεγχος της διαθέσιμης χωρητικότητας στους δίσκους στους οποίους κρατούνται τα logs της υπηρεσίας καθώς και στους δίσκους στους οποίους αποθηκεύονται οι πίνακες της mysql βάσης.

6.2 Αντιμετώπιση προβλημάτων χρηστών

Σε περίπτωση που ο χρήστης αναφέρει πρόβλημα με το password του, μέσα από την κεντρική σελίδα διαχείρισης του χρήστη στο περιβάλλον daloRadius, μπορούν να αντληθούν χρήσιμες πληροφορίες για το χρήστη. Θα πρέπει να δοθεί ιδιαίτερη προσοχή στο πεδίο ’Useful User Description’ το οποίο εάν ο κωδικός του χρήστη έχει ξεπεράσει τα όρια χρήσης ή είναι κλειδωμένος θα είναι κόκκινο και θα περιέχει ανάλογο κείμενο περιγραφής.

Σε περίπτωση που το πεδίο αυτό δεν περιέχει κάποια χρήσιμη πληροφορία, θα πρέπει να ελέγχεται η σελίδα που περιέχει το accounting του χρήστη. Εάν υπήρξε πρόβλημα στις τελευταίες συνδέσεις του χρήστη, τότε οι αντίστοιχες εγγραφές θα εμφανίζονται με κόκκινο χρώμα, ενώ ο λόγος της αποτυχίας θα εμφανίζεται στο πεδίο ‘terminate cause’. Συνήθεις περιπτώσεις προβλημάτων είναι:

- Αποτυχημένο password
- Υπέρβαση ορίου χρήσης (ημερήσιου, εβδομαδιαίου κτλ)
- Πολλαπλή πρόσβαση
- Κλήση έξω από επιτρεπόμενο διάστημα

Στην περίπτωση που το πρόβλημα έχει να κάνει με το password του χρήστη από την κεντρική σελίδα διαχείρισης, μπορεί να γίνει επαλήθευση του password, ενώ από τη σελίδα δοκιμών (‘Test User’) μπορεί να γίνει επαλήθευση της διαδικασίας login, όπως αυτή λαμβάνει χώρα στον radius server.

Το σύστημα κρατάει εγγραφές με τα bad logins του συστήματος στο αρχείο radius.log. Στο αρχείο αυτό αποθηκεύονται οι περιπτώσεις:

- Αποτυχημένης πιστοποίησης του χρήστη (login incorrect).
- Πολλαπλής πρόσβασης (multiple login).
- Υπέρβασης του ημερήσιου/εβδομαδιαίου διαθέσιμου χρόνου.
- Κλήσεις έξω από το επιτρεπτό χρονικό διάστημα.

Στις περιπτώσεις αυτές, είναι δυνατόν παράλληλα να γίνεται και αντίστοιχη εγγραφή στον πίνακα radacct. Αυτό μπορεί να επιτευχθεί με τη χρήση του utility log_badlogins που περιλαμβάνεται στο πακέτο daloRadius. Το utility αυτό είναι γραμμένο σε perl και απαιτεί το module Date::Manip. Επιπλέον στο ίδιο το script θα πρέπει να έχουν ενημερωθεί οι εξής μεταβλητές:

- *$domain*: Η μεταβλητή αυτή περιέχει το domain στο οποίο βρίσκονται οι access servers. Επειδή στο radius.log οι access servers καταγράφονται με το shortname είναι απαραίτητο να είναι γνωστό και το domain στο οποίο ανήκουν προκειμένου στη συνέχεια να μπορεί να προσδιοριστεί η IP τους. Με άλλα λόγια θα πρέπει για κάθε access server το $shortname . $domain να αντιστοιχεί σε μία IP.
- *$mysql*: Το path του mysql utility της MySQL.
- *$tmpfile*: Το path στο οποίο θα καταγράφεται το sql script με το οποίο γίνεται η ενημέρωση του MySQL server. Για κάθε MySQL server προστίθεται το hostname του στο τέλος του ονομάτος του αρχείου. Κατά αυτόν τον τρόπο ακόμα και αν ένας MySQL server δεν

αποκρίνεται το log_badlogins θα συνεχίσει να καταγράφει τα δεδομένα στους υπολοιπους servers, ενώ μόλις ο server επανέλθει θα του αποσταλούν όλα τα δεδομένα.

Πέραν των παραπάνω είναι απαραίτητο να έχουν οριστεί συγκεκριμένες τιμές σε διάφορες μεταβλητές στο αρχείο admin.conf του daloRadius, αλλά αυτό δεν περιλαμβάνεται στην παρούσα τεκμηρίωση. Η μορφή που θα έχουν τα αντίστοιχα accounting records είναι StartTime = StopTime, SessionTime=0 και terminateCause την αιτία του bad login. Οι εγγραφές αυτές είναι προσβάσιμες από την εφαρμογή web διαχείρισης daloRadius όπως περιγράφεται στην τεκμηρίωση της εφαρμογής.

6.3 Πολιτική Ασφάλειας

Το RADIUS πρωτόκολλο, εκ φύσεως προσφέρει ασφαλή τρόπο μεταφοράς των ευαίσθητων δεδομένων των χρηστών (passwords), καθώς γίνεται κρυπτογράφηση στα δεδομένα αυτά με χρήση του secret key επικοινωνίας radius server <-> access server (NAS). Επίσης, κατά την πιστοποίηση του χρήστη απο τον LDAP είναι δυνατόν να δημιουργείται ασφαλές κανάλι επικοινωνίας SSL μέσω του οποίου να γίνεται η αποστολή του username/password του χρήστη για την πιστοποίηση του.

Επιπλέον, είναι απαραίτητο να επιτρέπονται αιτήσεις μόνο από συγκεκριμένους πελάτες, οι οποίοι και θα είναι καταγραμμένοι στο αρχείο clients.conf, όπως προαναφέρθηκε. Οι πελάτες αυτοί θα πρέπει να είναι μόνο οι access servers οι οποίοι χρησιμοποιούν το RADIUS server για λειτουργίες AAA, το localhost, καθώς και όποιοι άλλοι πελάτες είναι δυνατόν να κάνουν ερωτήσεις στο RADIUS server, όπως για παράδειγμα ο υπολογιστής στον οποίο είναι εγκατεστημένη η εφαρμογή web based διαχείρισης daloRadius.

Καλό θα ήταν να αποκλειστεί μέσω ACIs στους αντίστοιχους δρομολογητές η δυνατότητα αποστολής RADIUS πακέτων στο διακομιστή

από πελάτες πέραν των ήδη γνωστών.

6.4 Αντίγραφα ασφαλείας

Η πολιτική αντιγράφων ασφαλείας θα πρέπει να είναι η εξής:

- Δημιουργία αντίγραφου ασφαλείας της αρχικής εγκατάστασης, αμέσως μετά το πέρας της εγκατάστασης.
- Διατήρηση αντιγράφων ασφαλείας του καταλόγου που περιέχει τα αρχεία διαμόρφωσης της υπηρεσίας (etc/raddb) κάθε εβδομάδα σε βάθος ενός μήνα.
- Καθημερινή δημιουργία αντιγράφων ασφαλείας των αρχείων dbm των ημερήσιων/εβδομαδιαίων μετρητών προκειμένου η απώλεια τους να μην επηρεάσει την υπηρεσία.

6.5 Ειδικά Θέματα

6.5.1 Multilink δυνατότητες των χρηστών

Δύο attributes στον LDAP ορίζουν τις Multilink δυνατότητες του κάθε χρήστη: Το Port-Limit και το Simultaneous-Use. Το πρώτο ορίζει πόσα κανάλια μπορεί να ανοίξει ο χρήστης. Το δεύτερο ορίζει τον αριθμό των logins που μπορεί να κάνει συγχρόνως ο κάθε χρήστης. Εάν ο χρήστης συνδέεται με multilink και αιτεί αριθμό καναλιών μικρότερο ή ίσο απο το Port-Limit που αντιστοιχεί στο χρήστη τότε η αίτηση του γίνεται αποδεκτή.

6.5.2 Δυνατότητα αντιμετώπισης RADIUS Server Fail Over

Η παρούσα υλοποίηση αξιοποιεί έναν Radius Server είναι δυνατόν όμως να εγκατασταθούν παραπάνω από ένας RADIUS servers προκειμένου να επιτευχθεί μεγαλύτερη διαθεσιμότητα από την υπηρεσία.

Πέραν των ίδιων των εξυπηρετητών RADIUS είναι απαραίτητο να υπάρχει υψηλή διαθεσιμότητα και στις χρησιμοποιούμενες βάσεις, δηλαδή στη βάση του RADIUS καί στη βάση της MySQL. Αναφορικά με τη βάση LDAP μπορεί να υλοποιηθεί replication στην υπηρεσία χρησιμοποιώντας τις δυνατοτήτες που προσφερουν τα αντίστοιχα πακέτα λογισμικού. Ως αναφορά την MySQL (δηλαδή το accounting) είναι δυνατόν να χρησιμοποιηθούν οι δυνατότητες που προσφέρει ο ίδιος ο freeradius για αυτή τη λειτουργία. Πιο συγκεκριμένα στο distribution του freeradius περιλαμβάνεται και το utility radrelay.

Η λειτουργία του παραπάνω προγράμματος. είναι να διαβάζει συνεχώς

ένα accounting detail αρχείο και να στέλνει τα δεδομένα που περιλαμβάνονται σε αυτό, σε έναν απομακρυσμένο RADIUS server. Αν ληφθεί επιβεβαίωση για τα δεδομένα, τότε αυτά διαγράφονται από το αντίστοιχο detail αρχείο. Κατά αυτόν τον τρόπο, εφόσον ο απομακρυσμένος εξυπηρετητής αποκρίνεται κανονικά, το detail αρχείο θα έχει μηδενικό μέγεθος. Αν σταματήσει να αποκρίνεται, τότε το πρόγραμμα απλώς προσπαθεί συνεχώς να στείλει τα αντίστοιχα δεδομένα, ενώ το detail αρχείο αυξάνει σε μέγεθος καθώς προστίθενται νέα accounting δεδομένα. Με άλλα λόγια, ακόμα και αν ένας απομακρυσμένος εξυπηρετητής δεν είναι διαθέσιμος, πάλι το accounting θα είναι συγχρονισμένο. Αν και στους δύο εξυπηρετητές λειτουργεί ένα αντίστοιχο πρόγραμμα, τότε όποιος εξυπηρετητής και αν εξυπηρετεί την υπηρεσία, το accounting θα παραμένει συγχρονισμένο μεταξύ τους. Το πρόγραμμα αυτό είναι αρκετά έξυπνο για να αποφεύγει ατέρμονα loops.

Ουσιαστικά λοιπόν, θα πρέπει σε κάθε RADIUS server να προστεθεί ένα instance του detail module ως εξής:

```
detail {
        detailfile = ${radacctdir}/detail
        locking = yes
        detailperm = 0600
}

accounting {
        .....
        detail
}
```

Επιπλέον θα πρέπει να εκτελεστεί το radrelay utility ως εξής:

/usr/local/radiusd/bin/radrelay -a /var/radiusd/log/radacct –S /usr/local/radiusd/etc/raddb/radrelay.secret -r <remote_radius_server> detail"

Το αρχείο radrelay.secret θα πρέπει να περιέχει το secret του RADIUS server.

6.5.4 Συστάσεις περί της MySQL

Συνιστάται ιδιαίτερα αν το accounting που θα διατηρείται στην MySQL, είναι μεγαλύτερο από μερικές χιλιάδες rows, να χρησιμοποιούνται InnoDB tables αντί για MyISAM tables, καθώς το table level locking των τελευταίων μπορεί να δημιουργήσει μεγάλα προβλήματα απόδοσης, ιδιαίτερα αν παράλληλα με την λειτουργία του RADIUS server εκτελούνται μεγάλα queries στη βάση μέσα από εργαλεία όπως το daloradius.

Επιπλέον, συνίσταται η δημιουργία ενός πρόσθετου multi column index που να περιλαμβάνει τα attributes UserName και AcctStopTime. Αυτό μπορεί να αυξήσει την απόδοση αν χρησιμοποιείται το sql module στο session section, καθώς και στα queries που εκτελούνται από το Daloradius.

6.5.5 Πρόσθετες απαιτήσεις από την υπηρεσία καταλόγου LDAP

Το λογισμικό δε θέτει ιδιαίτερες από την υπηρεσία καταλόγου. Βασική απαίτηση, είναι να έχει προστεθεί η κλάση radiusprofile με τα αντίστοιχα attributes στο LDAP schema. Επιπλέον, είναι προτεινόμενο το attribute uid να είναι indexed, προκειμένου να γίνονται γρήγορα οι αναζητήσεις για χρήστες, ενώ επιπλέον αν χρησιμοποιούνται και LDAP groups προτείνεται να είναι indexed το αντίστοιχο attribute που περιέχει το όνομα του group (συνήθως το cn).

Κατά την εκκίνηση του radius server, το ldap module δημιουργεί ένα connection pool από συνδέσεις προς τον ldap server (όπως αυτές έχουν οριστεί με το directive ldap_connections_num). Αυτές οι συνδέσεις χρησιμοποιούνται στη συνέχεια σε κάθε access-request. Για κάθε αίτηση πραγματοποιείται μία αναζήτηση στον ldap, προκειμένου με βάση το username του χρήστη να προσδιοριστεί το DN της εγγραφής του στον ldap και να εξαχθούν τα αντίστοιχα radius attributes που μπορεί να περιέχονται σε αυτή. Σε περίπτωση που έχουν ενεργοποιηθεί οι αντίστοιχες δυνατότητες, πραγματοποιούνται επιπλέον αναζητήσεις για τα *Default/User/Regular Profiles.* Κατά συνέπεια, κάθε access-request συνεπάγεται το λιγότερο μία αναζήτηση στον εξυπηρετητή ldap, ενώ μπορεί τελικά να πραγματοποιηθούν μέχρι και άλλες δύο επιπλέον αναζητήσεις. Το ldap module, δεν λαμβάνει μέρος στην επεξεργασία των accounting-requests και έτσι δεν δημιουργεί κανένα επιπλέον φόρτο στην επεξεργασία του accounting.

Τέλος, εφόσον έχει ενεργοποιηθεί το authentication μέσω ldap, το ldap module θα ανοίξει μία νέα ldap σύνδεση για κάθε αίτηση, προκειμένου να γίνει η ταυτοποίηση του password του χρήστη. Κάτι τέτοιο, προφανώς, συνεπάγεται αύξηση του χρόνου εξυπηρέτησης κάθε αίτησης, καθώς στο χρόνο εξυπηρέτησης προστίθεται το connection overhead. Εναλλακτικά, ο διαχειριστής μπορεί να ενεργοποιήσει την εξαγωγή των passwords του χρήστη από το ldap module και να χρησιμοποιήσει το PAP module για τη φάση του authentication.

Παράρτημα Ι

Πρότυπα

RFC 2881: Network Access Server

RFC 2058: RADIUS Packet Format

RFC 2865: Remote Authentication Dial In User Service (RADIUS)

RFC 2866: RADIUS Accounting

RFC 2869: RADIUS Extensions

RFC 4510: Lightweight Directory Access Protocol (LDAP)

Βιβλιογραφία

1) Implementing NAP and NAS Security Technologies Daniel V. Hoffman εκδόσεις Safari Books

2) RADIUS Securing Public Access to Private Resources Jonathan Hassell εκδόσεις O' Reilly

3) LDAP System Administration Gerald Carter εκδόσεις O' Reilly

4) NAS(Network Access Server):

   http://www.linktionary.com/n/nas_server.html

5) Λογισμικό Coova Chilli:

http://coova.org/wiki/index.php/CoovaChilli

6) Λογισμικό Freeradius:

   http://www.freeradius.org/

7) Daloradius Web περιβάλλον διαχείρισης:
   http://www.sourceforge.net/projects/daloradius/

8) Βάση δεδομένων MySQL:

   MySQL Database: http://www.mysql.com/

9) Υπηρεσία Καταλόγου ΕΔΕΤ:

   http://ds.grnet.gr/technical.php

Ορολογία

RADIUS	Remote Authentication Dial In User Service
LDAP	Lightweight Directory Access Protocol
EAP	Extensible Authentication Protocol
PPP	Point-to-Point Protocol
TLS	Transport Layer Security
ACI	Access Control Instruction
PAP	Password Authentication Protocol
CHAP	Challenge Handshake Authentication Protocol
MS-CHAP	Microsoft Challenge Handshake Authentication Protocol
SNMP	Simple Network Management Protocol
MTU	Maximum Transmission Unit
IP	Internet Protocol
NAS	Network Access Server

SHA1	Secure Hash Algorithm Ver1
MD5	Message Digest 5