«Ασφάλεια Δικτύων και Διακομιστές Διαμεσολάβησης- Proxy Servers»

Φωτόπουλος Ιωάννης

Άρτα, 9 Φεβρουαρίου 2004

Πτυχιακή Εργασία, μέρος των απαιτήσεων
του τμήματος Τηλεπληροφορικής και Διοίκησης

Πίνακας περιεχομένων

Ακρωνύμια

CARP: Cache Array Routing Protocol

CERT: Computer Emergency Response Team

DMZ: Demilitarized Zone Firewall

DNS: Domain Name System

FTP: File Transfer Protocol

GUI: Graphic User Interface

HTML: HyperText Markup Language

ICMP: Internet Control Message Protocol

IP: Internet Protocol

IIS: Internet Information Service

IPX/SPX: Internet Packet Exchange/Sequenced Packet Exchange

IRC: Internet Relay Chat

ISAPI: Internet Service Application Programming Interface

ISDN: Integrated Services Digital Network

ISP: Internet Service Provider

LAN: Local Area Network

MAC: Media Access Control

NAT: Network Address Translation

NFS: Network File System

NIC: Network Interface Card

NNTP: Network News Transfer Protocol

NTP: Network Time Protocol

POP3: Post Office Protocol V.3

PPP: Point to Point Protocol

PPTP: Point to Point Tunneling Protocol

RAM: Random Access Memory

SHTTP: Secure HyperText Transport Protocol

SMTP: Simple Mail Transfer Protocol

SQL: Structured Query Language

SSL: Secure Sockets Layer

TCP/IP: Transmission Control Protocol/Internet Protocol

TTL: Time To Live

UDP: User Datagram Protocol

URL: Uniform Resource Locator

WAN: Wide Area Network

WWW: World Wide Web

ΔΗΛΩΣΗ ΠΕΡΙ ΛΟΓΟΚΛΟΠΗΣ

Όλες οι προτάσεις οι οποίες παρουσιάζονται σ' αυτό το κείμενο και οι οποίες ανήκουν σε άλλους αναγνωρίζονται από τα εισαγωγικά και υπάρχει η σαφής δήλωση του συγγραφέα. Τα υπόλοιπα γραφόμενα είναι επινόηση του γράφοντος ο οποίος φέρει και την καθολική ευθύνη γι' αυτό το κείμενο και δηλώνουμε υπεύθυνα ότι δεν υπάρχει λογοκλοπή γι' αυτό το κείμενο.

Ονοματεπώνυμο: Φωτόπουλος Ιωάννης

Υπογραφή……

Ημερομηνία:……

Εισαγωγή

Στα χρόνια πριν από την εξάπλωση της χρήσης των ηλεκτρονικών υπολογιστών ως εργαλεία επεξεργασίας της πληροφορίας, η διασφάλιση της μυστικότητας, ακεραιότητας και διαθεσιμότητας των σημαντικών πληροφοριών ενός οργανισμού γινόταν μέσω της φυσικής προστασίας των, καθώς και μέσω κάποιων διαδικασιών και κανονισμών ασφάλειας. Τα ευαίσθητα έγγραφα κλείνονταν μέσα σε ντουλάπες ή χρηματοκιβώτια στιβαρής κατασκευής τα οποία προστατεύονταν από κλειδαριές, ενώ μόνον εξουσιοδοτημένο προσωπικό είχε πρόσβαση σε αυτά. Τις τελευταίες δεκαετίες, δύο γεγονότα έχουν αλλάξει δραστικά τις ανάγκες των οργανισμών σε σχέση με την ασφάλεια των πληροφοριών.

Το πρώτο γεγονός είναι η εισαγωγή των υπολογιστών ως εργαλεία αποθήκευσης και επεξεργασίας της πληροφορίας. Η προστασία της πληροφορίας ανάγεται πλέον στην προστασία των αρχείων των υπολογιστών στα οποία είναι αποθηκευμένη η πληροφορία, στον έλεγχο της πρόσβασης στα αρχεία αυτά, καθώς και στην προστασία των προγραμμάτων εκείνων που μπορούν να απειλήσουν την ασφάλεια των αρχείων αυτών. Ο όρος που χρησιμοποιείται για να περιγράψει το σύνολο εργαλείων και διαδικασιών που έχουν σχεδιαστεί για την προστασία των ηλεκτρονικών δεδομένων είναι “ασφάλεια υπολογιστών”.

Το δεύτερο γεγονός το οποίο επηρέασε δραστικά τις ανάγκες σε ασφάλεια της πληροφορίας είναι η εισαγωγή των κατανεμημένων συστημάτων και η χρήση των δικτύων και τηλεπικοινωνιακών συστημάτων για την μεταφορά δεδομένων μεταξύ υπολογιστών. Ο όρος “ασφάλεια δικτύων” αναφέρεται στα μέτρα προστασίας των δεδομένων κατά την μεταφορά τους μέσω του δικτύου διασύνδεσης. Στα πλαίσια της διαχείρισης ενός δικτύου, η διαχείριση ασφάλειας αναφέρεται στην παροχή ασφάλειας σε όλα τα στοιχεία του δικτύου, δηλαδή σε ασφάλεια υπολογιστών και ασφάλεια δικτύου.

Μέρος Πρώτο

Ασφάλεια

1 Απειλές κατά της Ασφάλειας

Η ασφάλεια των υπολογιστών και δικτύων καλύπτει τις παρακάτω απαιτήσεις:

- μυστικότητα: απαιτείται η πληροφορία να είναι προσπελάσιμη για ανάγνωση μόνον από εξουσιοδοτημένους χρήστες. Αυτού του είδους η πρόσβαση περιλαμβάνει την εκτύπωση, την προβολή και άλλες φορές ακόμα και την αποκάλυψη ύπαρξης κάποιου είδους πληροφορίας.
- ακεραιότητα: απαιτείται οι πόροι του συστήματος (data, processes κλπ) να μπορούν να τροποποιηθούν μόνον από εξουσιοδοτημένους χρήστες. Η τροποποίηση περιλαμβάνει την εγγραφή, τροποποίηση, αλλαγή κατάστασης, διαγραφή και δημιουργία.
- Διαθεσιμότητα: απαιτείται οι πόροι του συστήματος να είναι διαθέσιμοι στους εξουσιοδοτημένους χρήστες.

1.1 Μορφές απειλών

Οι απειλές κατά των κινούμενων δεδομένων αφορούν στην ακεραιότητα, μυστικότητα και διαθεσιμότητα των δεδομένων και μπορούν να χωρισθούν σε δύο κατηγορίες:

A. **Απειλές παθητικής φύσης.** Απειλούν την μυστικότητα των δεδομένων (π.χ. μέσω ειδικών προγραμμάτων packet sniffers) με σκοπό την απόκτηση των πληροφοριών. Για παράδειγμα ο χρήστης ενός Η/Υ μπορεί να χρησιμοποιήσει ένα τέτοιο πρόγραμμα για να παρακολουθεί όλα τα πακέτα που εκπέμπονται στο τοπικό δίκτυο. Τέτοιου είδους ενέργειες είναι πολύ δύσκολο να αποκαλυφθούν διότι δεν προκαλούν αλλαγή στα δεδομένα και δεν επηρεάζουν την λειτουργία του δικτύου. Η παρακολούθηση των δεδομένων είναι δυνατή και μέσω παρακολούθησης των καλωδιώσεων χαλκού του δικτύου ή των τηλεφωνικών συνδέσεων πρόσβασης στο δίκτυο.

B. **Απειλές ενεργητικής φύσης.** Τέτοιου είδους απειλές έχουν σαν στόχο την τροποποίηση των κινούμενων δεδομένων. Είναι δυνατή μια περαιτέρω κατηγοριοποίηση τέτοιων απειλών ως εξής:

1. πρόκληση τροποποίησης της ροής των πακέτων δεδομένων (message-stream modification), όπου ένα τμήμα του μηνύματος τροποποιείται ή κάποια καθυστερούν, επαναλαμβάνονται ή τροποποιείται η διαδοχή τους για να προκληθεί κάποιο αποτέλεσμα
2. πρόκληση άρνησης παροχής υπηρεσιών (denial of service), κατά την οποία παρεμποδίζεται η κανονική χρήση των πόρων του δικτύου. Μία τέτοια μορφή επίθεσης είναι η υπερφόρτωση του δικτύου με πακέτα με αποτέλεσμα την επιβράδυνση ή και διακοπή της λειτουργίας του. Άλλο παράδειγμα είναι η εξάλειψη μηνυμάτων που απευθύνονται σε κάποιο συγκεκριμένο αποδέκτη, όπως για παράδειγμα σε ένα πρόγραμμα που εκτελεί την υπηρεσία ελέγχου ασφάλειας (security audit service).
3. μεταμφίεση (masquerade) κατά την οποία ο εισβολέας τροποποιεί τα δεδομένα με στόχο να ξεγελάσει τους μηχανισμούς ασφάλειας του δικτύου και να θεωρηθεί ως εξουσιοδοτημένος ή έμπιστος χρήστης. Τέτοια παραδείγματα είναι η αλλαγή της IP διεύθυνσης πακέτων του εξωτερικού εισβολέα έτσι ώστε το σύστημα firewall να νομίσει ότι τα πακέτα έρχονται από το εσωτερικό δίκτυο (IP Spoofing), ή η ηχογράφηση κάποιας συνομιλίας ελέγχου αυθεντικότητας (authentication) μεταξύ ενός εξουσιοδοτημένου χρήστη και του συστήματος και κατόπιν η χρήση της από τον εισβολέα.

1.2 Τρόποι προστασίας της δικτυακής μας τοποθεσίας

Οι άνθρωποι επιλέγουν ποικίλα πρότυπα ασφάλειας, ή προσεγγίσεις, που κυμαίνονται από καθόλου ασφάλεια , μέσω αυτού που καλείται «ασφάλεια

μέσω της ασημότητας- αφάνειας» και την ασφάλεια των host, στην ασφάλεια δικτύων.

❖ *Καμία Ασφάλεια*

Η απλούστερη πιθανή προσέγγιση είναι να μην τεθεί καμία προσπάθεια στην ασφάλεια, και να τρέξει με ο,τιδήποτε ελάχιστη ασφάλεια ο προμηθευτής σας σας παρέχει εξ ορισμού.

❖ *Ασφάλεια μέσω της ασημότητας- αφάνειας*

Ένα άλλο πιθανό πρότυπο ασφάλειας είναι αυτό καλούμενο συνήθως «η ασφάλεια μέσω της ασημότητας- αφάνειας.» Με αυτό το πρότυπο, ένα σύστημα θεωρείται ασφαλές απλά επειδή (υποθετικά) κανένας δεν ξέρει για την ύπαρξή του, το περιεχόμενο, τα μέτρα ασφάλειας, ή τίποτ' άλλο. Αυτή η προσέγγιση λειτουργεί σπάνια για πολύ γιατί υπάρχουν πάρα πολλοί τρόποι να βρεθεί ένας ελκυστικός στόχος.

Πολλοί άνθρωποι υποθέτουν ότι ακόμα κι αν οι επιτιθέμενοι μπορούν να τους βρουν, δεν θα ενοχλήσουν. Λογαριάζουν ότι μια μικρή επιχείρηση ή ένας ιδιωτικός υπολογιστής δεν πρόκειται να είναι ελκυστικός στόχος στους εισβολείς. Στην πραγματικότητα, πολλοί εισβολείς δεν στοχεύουν στους ιδιαίτερους στόχους, θέλουν ακριβώς να σπάσουν σε όσο το δυνατόν περισσότερους Η/Υ. Σε αυτούς, οι μικρές επιχειρήσεις και οι ιδιωτικοί υπολογιστές μοιάζουν απλά με εύκολους στόχους. Πιθανώς δεν θα μείνουν για πολύ, αλλά θα προσπαθήσουν να μπουν μέσα, και μπορούν να κάνουν ιδιαίτερη ζημιά προσπαθώντας να καλύψουν τα ίχνη τους.

Για να λειτουργήσει σε οποιοδήποτε δίκτυο, συμπεριλαμβανομένου το Διαδίκτυο, ένα site πρέπει να κάνει τουλάχιστον μια ελάχιστη εγγραφή και ένα μεγάλο μέρος αυτών των πληροφοριών εγγραφής είναι διαθέσιμο στον καθέναν. Κάθε φορά που ένα site χρησιμοποιεί τις υπηρεσίες του δικτύου, κάποιος – τουλάχιστον όποιος παρέχει την υπηρεσία – ξέρει ότι είναι εκεί. Οι εισβολείς

προσέχουν για τις νέες συνδέσεις, με την ελπίδα ότι στα site αυτά δεν θα έχουν ακόμα τα μέτρα ασφάλειας σε ισχύ.

Είναι εκπληκτικό το πόσοι τρόποι υπάρχουν για να ανακαλύψει κανείς τις ευαίσθητες από άποψη ασφάλειας πληροφορίες του site μας. Παραδείγματος χάριν, η γνώση του ποιου υλικού και λογισμικού έχετε και ποια έκδοση του λειτουργικού συστήματος χρησιμοποιείτε δίνει στους εισβολείς σημαντικές ενδείξεις για ποιες τρύπες ασφάλειας να προσπαθήσουν. Μπορούν συχνά να πάρουν αυτές τις πληροφορίες από την εγγραφή(registration) των host, ή με την προσπάθεια να συνδεθούν με τον υπολογιστή σας. Πολλοί υπολογιστές αποκαλύπτουν τον τύπο λειτουργικού συστήματός τους στο χαιρετισμό που παίρνετε κατά την διαδικασία του login, έτσι ένας εισβολέας δεν χρειάζεται την πρόσβαση για να αποκτήσει τέτοιου είδους πληροφορίες.

Οι εισβολείς έχουν πολύ χρόνο στη διάθεση τους και μπορούν συχνά να αποφύγουν να πρέπει να υπολογίσουν τα σκοτεινά γεγονότα με απλά να δοκιμάσουν όλες τις δυνατότητες. Μακροπρόθεσμα, δεν είναι μια έξυπνη επιλογή μεθόδου ασφάλειας η ασφάλεια μέσω της ασημότητας- αφάνειας.

❖ *Ασφάλεια των host*

Πιθανώς το πιο κοινό πρότυπο για την ασφάλεια υπολογιστών είναι η ασφάλεια των host. Με αυτό το πρότυπο, επιβάλλετε την ασφάλεια σε κάθε host χωριστά, και καταβάλλετε κάθε προσπάθεια να αποφύγετε όλα τα γνωστά προβλήματα ασφάλειας που έχουν επιπτώσεις σε εκείνο τον ιδιαίτερο host. Πού είναι το πρόβλημα με την ασφάλεια των host; Δεν είναι ότι δεν λειτουργεί στις μεμονωμένες μηχανές, αλλά ότι δεν βολεύει για μεγάλους αριθμούς μηχανών.

Το σημαντικότερο εμπόδιο στην αποτελεσματική ασφάλεια των host στα σύγχρονα υπολογιστικά περιβάλλοντα είναι η πολυπλοκότητα και η ποικιλομορφία εκείνων των περιβαλλόντων. Τα περισσότερα σύγχρονα περιβάλλοντα περιλαμβάνουν μηχανές από τους πολλούς και διάφορους προμηθευτές, κάθε ένας με το λειτουργικό σύστημά του, και κάθε ένας με το σύνολό του προβλημάτων ασφάλειας. Ακόμα κι αν το site έχει μηχανές από μόνο έναν προμηθευτή, οι διαφορετικές εκδόσεις του ίδιου λειτουργικού συστήματος έχουν συχνά σημαντικά διαφορετικά προβλήματα ασφάλειας. Ακόμα κι αν όλες αυτές οι μηχανές είναι από

έναν ενιαίο προμηθευτή και τρέχουν μια την ίδια έκδοση του λειτουργικού συστήματος, οι διαφορετικές διαμορφώσεις (διαφορετικές υπηρεσίες που χρησιμοποιεί ή τρέχει το καθένα, και τα λοιπά) μπορούν να φέρουν τα διαφορετικά υποσυστήματα στο παιχνίδι (και στη σύγκρουση) και οδηγούν στα διαφορετικά σύνολα προβλημάτων ασφάλειας. Και, ακόμα κι αν οι μηχανές είναι όλες απολύτως ίδιες, ο μεγάλος αριθμός τους επί μερικών site μπορεί να κάνει την ασφάλισή τους κάτι αρκετά δύσκολο. Απαιτείται ένα σημαντικό ποσό δύσκολης και συνεχούς εργασίας για να εφαρμοστεί αποτελεσματικά και να διατηρηθεί η ασφάλεια των host. Ακόμη και με όλη αυτήν την εργασία να γίνεται σωστά, η ασφάλεια των host συχνά αποτυγχάνει λόγω των bugs στο λογισμικό, ή λόγω έλλειψης αρκετά ασφαλούς λογισμικού για ορισμένες απαιτούμενες λειτουργίες.

Η ασφάλεια των host στηρίζεται επίσης στις καλές προθέσεις και την ικανότητα του καθενός που έχει προνόμια στην πρόσβαση σε οποιαδήποτε μηχανή. Δεδομένου ότι ο αριθμός μηχανών αυξάνεται, ο αριθμός προνομιούχων χρηστών αυξάνεται γενικά επίσης. Η ασφάλιση μιας μηχανής είναι δυσκολότερη από ότι να την συνδέσεις σε ένα δίκτυο. Η σύνδεση μιας μηχανής χωρίς τις κατάλληλες ασφαλίσεις μπορεί να επιφέρει απρόσμενες εκπλήξεις στο δίκτυο.

Ένα πρότυπο ασφάλειας των host μπορεί να είναι ιδιαίτερα κατάλληλο για μικρά site ή για site με ακραίες απαιτήσεις ασφάλειας. Πράγματι, όλες τα site πρέπει να συμπεριλάβουν κάποιο επίπεδο ασφάλειας host στα γενικά σχέδια ασφάλειάς τους. Ακόμα κι αν υιοθετείτε ένα πρότυπο ασφάλειας δικτύων, όπως περιγράφουμε στο επόμενο τμήμα, ορισμένα συστήματα στη διαμόρφωσή σας θα ωφεληθούν από την δυναμική της ασφάλειας των host. Παραδείγματος χάριν, ακόμα κι αν έχετε ένα firewall γύρω από το εσωτερικό δίκτυο και τα συστήματά σας, θα υπάρξουν ορισμένα συστήματα που θα είναι εκτεθειμένα στον εξωτερικό κόσμο τα οποία και απαιτούν ασφάλεια των host. Το πρόβλημα είναι ότι το πρότυπο ασφάλειας των host δεν είναι ακριβώς μια φτηνή λύση παρά μόνο για μικρά ή πολύ απλά site. Το να δουλέψει απαιτεί υπερβολικά πολλούς περιορισμούς και ανθρώπους .

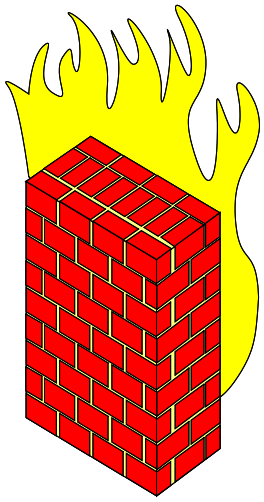
❖ *Ασφάλεια Δικτύων*

Καθώς τα υπολογιστικά περιβάλλοντα μεγαλώνουν και αλλάζουν και καθώς η δυσκολία της ασφάλειας σε μια host-προς-host βάση μεγαλώνει όλο και πιο πολλά

site στρέφονται σ' ένα μοντέλο ασφάλειας δικτύων. Με ένα πρότυπο ασφάλειας δικτύων, επικεντρώνεστε στο να ελέγχετε τη δικτυακή πρόσβαση στους διάφορους host σας και τις υπηρεσίες που αυτοί προσφέρουν, παρά στο να ασφαλιστούν ένας-ένας. Οι προσεγγίσεις ασφάλειας δικτύων περιλαμβάνουν το χτίσιμο αντιπυρικών ζωνών για να προστατεύσουν τα εσωτερικά συστήματα και δίκτυά σας, χρησιμοποιώντας προσεγγίσεις πιστοποίησης (όπως οι κωδικοί πρόσβασης "μιας χρήσεως"), και κρυπτογράφηση για την προστασία των ιδιαίτερα ευαίσθητων δεδομένων καθώς διέρχονται το δίκτυο.

Ένα site μπορεί να αποκτήσει τεράστια δύναμη και ευελιξία από τις προσπάθειες ασφάλειάς της με τη χρησιμοποίηση ενός προτύπου ασφάλειας δικτύων. Παραδείγματος χάριν, ένα firewall δικτύου μπορεί να προστατεύσει εκατοντάδες, χιλιάδες, ή ακόμα και δεκάδες χιλιάδες μηχανές από την επίθεση από τα δίκτυα πέρα από το firewall, ανεξάρτητα από το επίπεδο ασφάλειας των host των μεμονωμένων μηχανών.

Μέρος Δεύτερο



FIREWALLS

2 Firewalls

2.1 Τι είναι το firewall;

Το firewall είναι μια διάταξη εξειδικευμένων μηχανισμών ασφαλείας που ελέγχει την πρόσβαση και την μετακίνηση πληροφορίας μεταξύ ενός αξιόπιστου και ενός μη αξιόπιστου δικτύου. Δεν είναι απλώς ένα συστατικό λογισμικού ή υλικού αλλά μια ενιαία στρατηγική προφύλαξης πόρων.

Το firewall υλοποιεί και ενδυναμώνει μια πολιτική ασφαλείας. Χωρίς την ανάλογη πολιτική καθίσταται άσκοπο. Αφορά στο σύνολο του λογισμικού και των διαδικασιών που χρησιμοποιούνται για την υλοποίηση της πολιτικής ασφαλείας μέσω της διαχείρισης της εισερχόμενης και εξερχόμενης κίνησης από το εσωτερικό δίκτυο. Αποτελεί την πρώτη γραμμή άμυνας, αλλά οπωσδήποτε ποτέ την μόνη, έναντι οποιασδήποτε παράνομης κίνησης.

Η κύρια λειτουργία του είναι ο κεντρικός έλεγχος των σημείων πρόσβασης στο εσωτερικό μας δίκτυο. Το κρίσιμο θέμα είναι εάν μπορούν βέβαια να προσδιοριστούν όλα τα σημεία εισόδου και να προστατευθούν ανάλογα. Ακόμα και εάν έχει ληφθεί μέριμνα για τα παραπάνω, εφόσον εξωτερικοί χρήστες αποκτήσουν πρόσβαση στο εσωτερικό δίκτυο, χωρίς να περάσουν μέσω του firewall, η αποτελεσματικότητά του εκμηδενίζεται. Αυτό θα μπορούσε να συμβεί, για παράδειγμα, εάν ο υπάλληλος ενός οργανισμού επέλεγε να συνδεθεί με το internet μέσω ενός modem που βρίσκεται στο γραφείο του. Σε μια τέτοια περίπτωση δημιουργεί μια ανασφαλή σύνδεση, παρακάμπτοντας το firewall και εκθέτοντας το εσωτερικό δίκτυο στους επίδοξους εισβολείς.

Όπως και με κάθε μέτρο ασφαλείας, υπάρχουν συμβιβασμοί που πρέπει να γίνουν μεταξύ επιπέδων ασφάλειας και άνεσης. Το firewall θα πρέπει

να είναι διαφανές προς τους χρήστες, ενώ αντίθετα θα είναι ένα ορατό εμπόδιο για τους εξωτερικούς χρήστες.

2.2 Τα οφέλη και οι περιορισμοί των firewall

Τα firewall παρέχουν ορισμένους τύπους προστασίας:

- Μπορούν να μπλοκάρουν μη επιθυμητή κίνηση.
- Μπορούν να κατευθύνουν εσωτερική κίνηση σε πιο αξιόπιστα εσωτερικά συστήματα.
- Μπορούν να αποκρύψουν ευαίσθητα ή ευπρόσβλητα συστήματα, τα οποία δεν είναι εύκολο να αποκοπούν και να προστατευθούν από το Διαδίκτυο .
- Μπορούν να παρακολουθούν και να καταγράφουν την κίνηση από και προς το εσωτερικό δίκτυο.
- Μπορούν να αποκρύψουν ονόματα συστημάτων, τοπολογίες δικτύων, τύπους συσκευών δικτύων, τοπολογίες δικτύων, τύπους συσκευών δικτύων και ταυτότητες εσωτερικών χρηστών.
- Μπορούν να προσφέρουν καλύτερο και πιο αξιόπιστο έλεγχο ταυτότητας από ότι άλλες εφαρμογές.
- Δεν παρέχουν επαρκή προστασία όσον αφορά τους ιούς.

Το firewall είναι μια προσέγγιση στην ασφάλεια του εσωτερικού δικτύου. Συνεισφέρει στην υλοποίηση μιας πολιτικής ασφάλειας που ορίζει υπηρεσίες και επιτρεπόμενη πρόσβαση. Γενικά υλοποιούνται δύο κύριες σχεδιαστικές πολιτικές: η στάση ”default deny” και η στάση “default permit”. Η πρώτη απαγορεύει κάθε υπηρεσία εκτός και αν έχει επιτραπεί ρητά, ενώ η δεύτερη επιτρέπει κάθε υπηρεσία εκτός και αν έχει απαγορευτεί ρητά.

Η δεύτερη πολιτική διευκολύνει περισσότερο τους επίδοξους εισβολείς. Ο οργανισμός μπορεί να τοποθετήσει τον κεντρικό υπολογιστή που θα τρέχει έναν Web server έξω από το firewall , ενώ όταν ο web server θα πρέπει να επικοινωνήσει με βάσεις δεδομένων εντός του εσωτερικού δικτύου, η σύνδεση θα προστατεύεται από ένα firewall, υλοποιώντας έτσι μια αρχιτεκτονική ελεγχόμενων υποδικτύων (screened subnets).

Τα firewall που βασίζονται σε δρομολογητές δεν προσφέρουν έλεγχο ταυτότητας του χρήστη, ενώ αυτά που βασίζονται σε κεντρικό υπολογιστή υποστηρίζουν τα συνήθη password, password μιας χρήσης τα οποία αλλάζουν σε κάθε σύνδεση και ψηφιακά πιστοποιητικά. Η πολιτική θα πρέπει να ορίζει σαφώς εάν επιτρέπεται να κάνει και δρομολόγηση πακέτων ή απλώς θα τα προωθεί. Οι δρομολογητές που φιλτράρουν τα πακέτα (ενεργώντας ως firewall) κάνουν δρομολόγηση πακέτων. Αντίθετα οι *proxy server* δεν συνίσταται να κάνουν δρομολόγηση πακέτων, γιατί υπάρχει ο κίνδυνος να παρακαμφθούν οι έλεγχοι ασφάλειας. Επίσης, η δρομολόγηση πηγής (source routing) δεν πρέπει να επιτρέπεται και τα πακέτα να απορρίπτονται από το δρομολογητή. Εάν λειτουργεί και ως DNS Server, τότε οι εξωτερικοί υπολογιστές δεν γνωρίζουν τίποτα για το εσωτερικό δίκτυο. Μπορεί να χρησιμοποιηθεί για την προστασία υποτμημάτων ενός εσωτερικού δικτύου αλλά και για τη σύνδεση με ένα άλλο firewall, δημιουργώντας ένα ιδεατό δίκτυο (VPN). Τα περισσότερα προϊόντα πλέον υποστηρίζουν και αυτήν την δυνατότητα.

Προκειμένου ένας οργανισμός να υλοποιήσει ένα σύστημα firewall, συνίσταται η ασφαλής οδός: άρνηση κάθε υπηρεσίας εκτός αυτών που έχουν σαφώς οριστεί (default deny stance). Ο σχεδιαστής θα πρέπει να προσδιορίσει τα εξής:

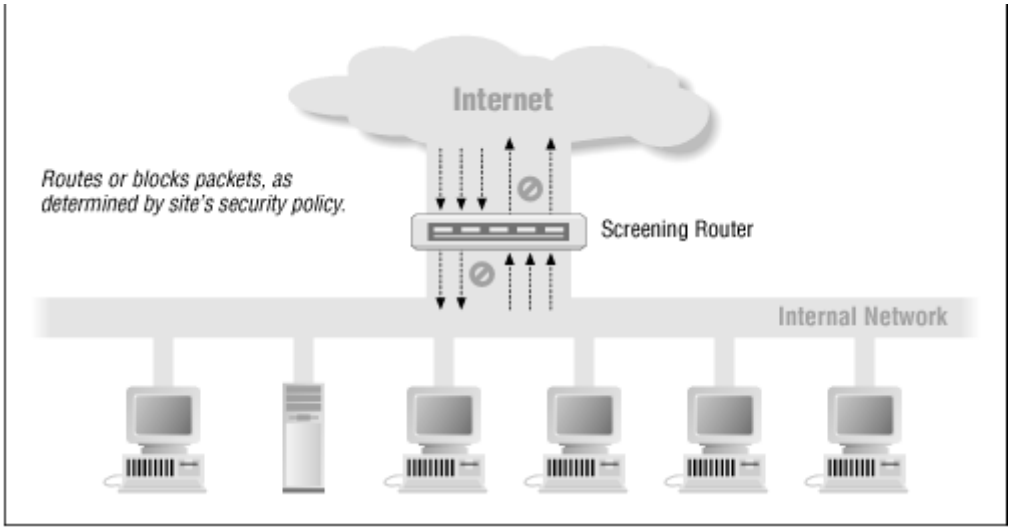
- Διαδικτυακές υπηρεσίες που χρειάζεται ο οργανισμός (telnet, http, smtp e-mail κλπ.)
- Τρόπους χρήσης των υπηρεσιών (τοπικά, από το σπίτι, από οποιοδήποτε σημείο του Internet κλπ.)
- Υποστήριξη πρόσθετων αναγκών όπως κρυπτογράφηση και dial-in.
- Κίνδυνοι που μπορούν να προέλθουν από την παροχή των συγκεκριμένων υπηρεσιών και επιπέδων πρόσβασης.
- Κόστος παροχής προστασίας σε επίπεδο ελέγχου και επίδρασης στους πόρους του δικτύου.
- Προτεραιότητα της ασφάλειας έναντι της χρήσης των πόρων και υπηρεσιών του δικτύου.

2.3 Σχεδιασμός των Firewall

Θα δώσουμε ορισμένους ορισμούς που αφορούν τα firewalls και την ασφάλεια δικτύων:

- *Firewall* – Μια συνιστώσα ή ένα σύνολο συνιστωσών που περιορίζει την πρόσβαση μεταξύ ενός προστατευμένου δικτύου και του Διαδικτύου ή μεταξύ κάποιων άλλων δικτύων.
- *Host* - Ένα υπολογιστικό σύστημα συνδεδεμένο σ' ένα δίκτυο.
- *Bastion host* - Ένα υπολογιστικό σύστημα που πρέπει να ασφαλιστεί ιδιαίτερα επειδή είναι τρωτό στις επιθέσεις, συνήθως επειδή εκτίθεται στο Internet και είναι ένα κύριο σημείο της επαφής για τους χρήστες των εσωτερικών δικτύων.
- *Dual homed host* – Ένα γενικής- χρήσεως υπολογιστικό σύστημα που έχει τουλάχιστον δύο κάρτες δικτύου.
- *Packet* – Η θεμελιώδης μονάδα επικοινωνίας στο Internet.
- *Packet Filtering* - Φιλτράρισμα πακέτων. Η δράση που αναλαμβάνει μια συσκευή για να ελέγξει επιλεκτικά τη ροή των δεδομένων από και προς ένα δίκτυο. Τα φίλτρα πακέτων επιτρέπουν ή αποτρέπουν τα πακέτα, συνήθως δρομολογώντας τα από το ένα δίκτυο στο άλλο (συχνότερα από το Διαδίκτυο σε ένα εσωτερικό δίκτυο, και αντίστροφα). Για να πραγματοποιήσετε το φιλτράρισμα πακέτων, οργανώνετε ένα σύνολο κανόνων που διευκρινίζουν ποιους τύπους πακέτων (π.χ., αυτά που προέρχονται από ή κατευθύνονται προς μια συγκεκριμένη διεύθυνση IP ή port) θα επιτρέψουμε και ποιους θα απορρίψουμε. Το φιλτράρισμα πακέτων μπορεί να εμφανιστεί σε έναν δρομολογητή, σε μια γέφυρα, ή σε έναν μεμονωμένο host. Είναι μερικές φορές γνωστό και ως *screening.*
- *Perimeter Network* – Περιμετρικό Δίκτυο. Ένα δίκτυο που προστίθεται μεταξύ ενός προστατευμένου δικτύου και ενός εξωτερικού δικτύου, ώστε να προστεθεί ένα επιπλέον στρώμα ασφάλειας. Πολλές φορές αναφέρεται και ως DMZ (από το "***D***e-***M***ilitarized ***Z***one)
- *Proxy Server* – Διακομιστής Διαμεσολάβησης. Ένα πρόγραμμα που διαπραγματεύεται με εξωτερικούς διακομιστές εκ μέρους εσωτερικών πελατών. Οι πελάτες-proxy μιλούν στον proxy server, ο οποίος αναμεταδίδει τις αιτήσεις των πελατών στους πραγματικούς διακομιστές, οι οποίοι με τη σειρά τους απαντούν στον proxy server. Τέλος, αυτός αναμεταδίδει τις απαντήσεις στον πελάτη του δικτύου.

2.4 Φιλτράρισμα πακέτων



Φιλτράρισμα Πακέτων 2.4.1

Τα συστήματα φιλτραρίσματος πακέτων δρομολογούν πακέτα μεταξύ εσωτερικών και εξωτερικών host, αλλά το κάνουν επιλεκτικά. Επιτρέπουν ή αποτρέπουν την προσπέλαση ορισμένων ειδών πακέτα με τρόπο που αντανακλά την πολιτική ασφάλειας του site μας (Εικόνα 2.4.1). Ο δρομολογητής ο οποίος χρησιμοποιείται από τα firewall φιλτραρίσματος πακέτων ονομάζεται *screening router.*

Κάθε πακέτο έχει ένα σύνολο από "κεφαλίδες" (headers), που περιέχουν συγκεκριμένες σημαντικές πληροφορίες, κάποιες από τις οποίες είναι: IP διεύθυνσης πηγής, IP διεύθυνσης προορισμού, το πρωτόκολλο (TCP,UDP ή ICMP), το TCP ή UDP port πηγής και προορισμού καθώς και τον τύπο του ICMP μηνύματος. Επιπρόσθετα ο δρομολογητής γνωρίζει κάποια πράγματα για τα πακέτα που εισέρχονται ή εξέρχονται από αυτόν τα οποία δεν αναφέρονται στις πληροφορίες των κεφαλίδων όπως για παράδειγμα το interface(η διεπαφή) απ' το οποίο μπήκε το πακέτο ή αυτό απ' το οποίο θα βγει.

Το γεγονός ότι χρησιμοποιούνται συγκεκριμένοι αριθμοί port στους server των υπηρεσιών Internet δίνει στον δρομολογητή τη δυνατότητα να επιτρέπει ή να αποτρέπει συγκεκριμένα είδη συνδέσεων απλά προσδιορίζοντας το κατάλληλο port (π.χ. TCP port 23 για συνδέσεις Telnet) στους κανόνες προσδιορισμού του φίλτρου πακέτων.

Εδώ παρουσιάζονται κάποια παραδείγματα με τον οποίο θα μπορούσαμε να προγραμματίσουμε έναν screening router ώστε να δρομολογεί τα πακέτα επιλεκτικά από ή προς το site μας:

- Μπλοκάρισμα όλων των εισερχόμενων συνδέσεων από συστήματα έξω από το δίκτυο μας, εκτός από τις εισερχόμενες SMTP συνδέσεις ώστε να λαμβάνουμε αλληλογραφία.
- Μπλοκάρισμα των συνδέσεων σε ή από μη-έμπιστα συστήματα.
- Αποδοχή υπηρεσιών αλληλογραφίας και FTP, αλλά μπλοκάρισμα επικίνδυνων υπηρεσιών όπως TFTP, του συστήματος X Window, των υπηρεσιών "*r*" (rlogin, rsh, rcp κτλ)

Για να γίνει πιο σαφής και κατανοητή η διαδικασία του φιλτραρίσματος πακέτων θα εξηγήσουμε τη διαφορά μεταξύ ενός συνηθισμένου router και ενός screening router.

Ένας συνηθισμένος δρομολογητής απλά ελέγχει τη διεύθυνση προορισμού του πακέτου και επιλέγει τον καλύτερο τρόπο που γνωρίζει ώστε να κατευθύνει το πακέτο προς τον προορισμό του. Η απόφαση που εκλαμβάνεται για τη μοίρα του πακέτου βασίζεται αποκλειστικά από τον προορισμό του. Υπάρχουν δύο εκδοχές που αφορούν τη μοίρα του πακέτου: είτε γνωρίζει ο δρομολογητής πώς να το στείλει προς τον προορισμό του και το πράττει, είτε δε γνωρίζει και το επιστρέφει από όπου ήρθε στέλνοντας και ένα ICMP μήνυμα "destination unreachable".

Από την άλλη ο screening router ρίχνει μια πιο προσεκτική ματιά στα πακέτα. Επιπρόσθετα, προσδιορίζοντας αν μπορεί ή όχι να δρομολογήσει το πακέτο προς τον προορισμό του, ένας screening router αποφαίνεται στο αν πρέπει ή όχι να το δρομολογήσει. Το αν πρέπει ή όχι προσδιορίζεται από την πολιτική ασφάλειας του site μας η οποία του έχει επιβληθεί.

Παρόλο που είναι δυνατό να βρίσκεται ένας screening router μεταξύ του Internet και του εσωτερικού μας δικτύου (Εικόνα 2.4.1) αυτό εναποθέτει τεράστια

ευθύνη σ' αυτόν. Όχι μόνο πρέπει να εκπληρώσει όλες τις διαδικασίες δρομολόγησης και λήψης αποφάσεων για τις δρομολογήσεις, αλλά είναι και το μόνο σύστημα ασφάλειας. Εάν η ασφάλειά του αποτύχει ή καταρρεύσει από μια επίθεση το εσωτερικό δίκτυο μένει εκτεθειμένο. Επιπρόσθετα, ένας γνήσιος screening router δεν μπορεί να τροποποιεί υπηρεσίες. Μπορεί να επιτρέψει ή όχι μια υπηρεσία , αλλά δεν μπορεί να προστατεύσει μεμονωμένες λειτουργίες μιας υπηρεσίας. Αν μια επιθυμητή υπηρεσία έχει κάποιες μη ασφαλής λειτουργίες ή αν η υπηρεσία συνήθως παρέχεται με έναν ανασφαλή server, το φιλτράρισμα πακέτων από μόνο του δε μπορεί να παρέχει την επιθυμητή ασφάλεια.

2.5 Υπηρεσίες Proxy

Οι υπηρεσίες *proxy* είναι εξειδικευμένες εφαρμογές ή προγράμματα διακομιστή τα οποία "τρέχουν" στο firewall το οποίο είναι είτε ένας *dual-homed host* με τη μία διεπαφή στο εσωτερικό δίκτυο και την άλλη στο εξωτερικό είτε ένας *bastion host* ο οποίος είναι προσπελάσιμος από τις εσωτερικές μηχανές του δικτύου και έχει πρόσβαση στο Διαδίκτυο. Αυτά τα προγράμματα υποκλέπτουν τις αιτήσεις των χρηστών για υπηρεσίες του Διαδίκτυου και τις προωθούν, καθώς αρμόζει σύμφωνα με την πολιτική ασφάλειας, προς τις πραγματικές υπηρεσίες. Τα proxy παρέχουν συνδέσεις αντικατάστασης και ενεργούν ως πύλες προς τις υπηρεσίες. Γι' αυτό το λόγο τα proxy είναι και γνωστά ως *πύλες επιπέδου εφαρμογής* (*application-level gateways).*

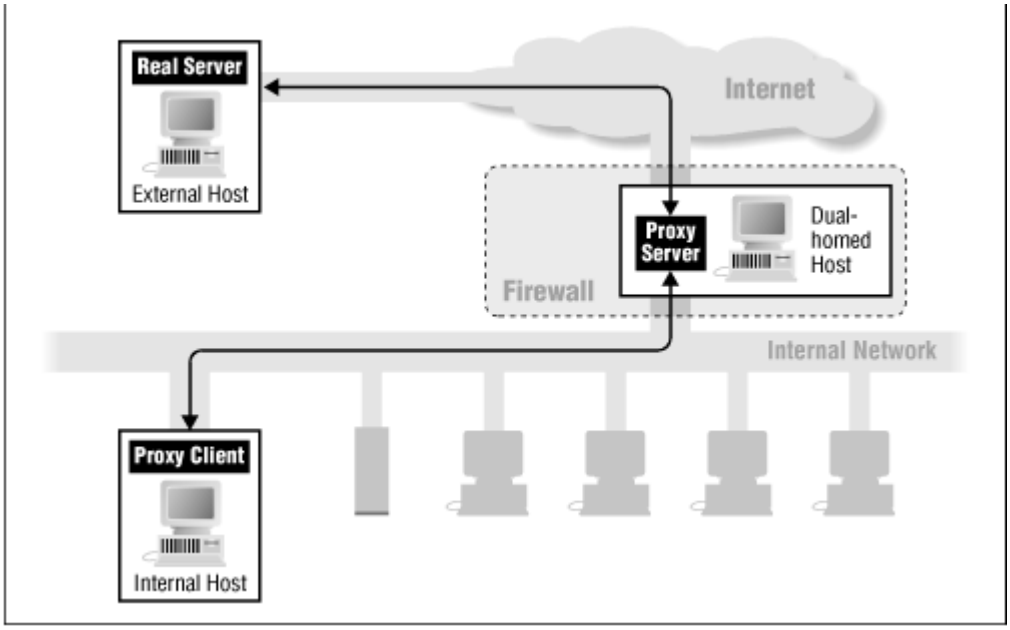
Οι υπηρεσίες proxy, άλλοτε αντιληπτές και άλλοτε όχι (transparent), βρίσκονται ανάμεσα στο χρήστη του εσωτερικού δικτύου και μιας υπηρεσίας έξω από αυτό (Διαδίκτυο). Αντί να μιλάνε κατ' ευθείαν ο ένας στον άλλον, μιλά ο καθένας σ 'έναν proxy. Οι proxy χειρίζονται όλες τις επικοινωνίες μεταξύ των εσωτερικών χρηστών και των υπηρεσιών του Διαδικτύου στο παρασκήνιο.

Η *διαφάνεια (transparency)* είναι το βασικό πλεονέκτημα των υπηρεσιών proxy. Στον πραγματικό server, ο proxy δίνει την ψευδαίσθηση ότι έχει να κάνει μ' έναν χρήστη απ' ευθείας στον proxy host. Στον πραγματικό χρήστη, ο proxy δίνει την ψευδαίσθηση ότι μιλά απ 'ευθείας με τον πραγματικό server.

Οι υπηρεσίες proxy μπορούμε να πούμε ότι είναι αποτελεσματικές όταν χρησιμοποιούνται σε συνάφεια με κάποιο μηχανισμό ο οποίος αποτρέπει την άμεση επικοινωνία μεταξύ των εσωτερικών και εξωτερικών host. Οι dual-homed hosts και τα φίλτρα πακέτων είναι τέτοιοι μηχανισμοί. Εάν υπάρχει επικοινωνία άμεση του εσωτερικού με το εξωτερικό περιβάλλον, εξουδετερώνεται η ανάγκη χρήσης του proxy, οπότε και δε θα χρησιμοποιούν. Μία τέτοια παρακαμπτήρια οδός πιθανώς δεν είναι σύμφωνη με την πολιτική ασφάλειας του site μας.

2.5.1 Υπηρεσίες Proxy σε ένα *Dual-homed Host.*

Δύο είναι οι συνιστώσες μιας υπηρεσίας proxy: ο proxy server και ο proxy client. Στην περίπτωσή μας ο server βρίσκεται στον Dual-homed host. Ο πελάτης είναι μια ειδική έκδοση ενός συνηθισμένου προγράμματος πελάτη (όπως είναι τα FTP,telnet κλπ.) που μιλά στον proxy server αντί για τον πραγματικό server. Επιπρόσθετα, αν στους χρήστες έχουν δοθεί συγκεκριμένες οδηγίες, τα συνηθισμένα προγράμματα πελάτη μπορούν να χρησιμοποιηθούν ως proxy πελάτες. Ο proxy server εκτιμά τις αιτήσεις των πελατών και αποφαίνεται στο αν θα τις δεχτεί ή αν θα τις απορρίψει. Αν αποδεχτεί κάποια αίτηση, τότε επικοινωνεί με τον πραγματικό server εκ μέρους του πελάτη και προχωρεί στην αναμετάδοση της αίτησης προς τον πραγματικό server από τον πελάτη και τις απαντήσεις από τον πραγματικό server προς τον πελάτη.



Υπηρεσίες proxy σε Dual-homed host 2.5.1

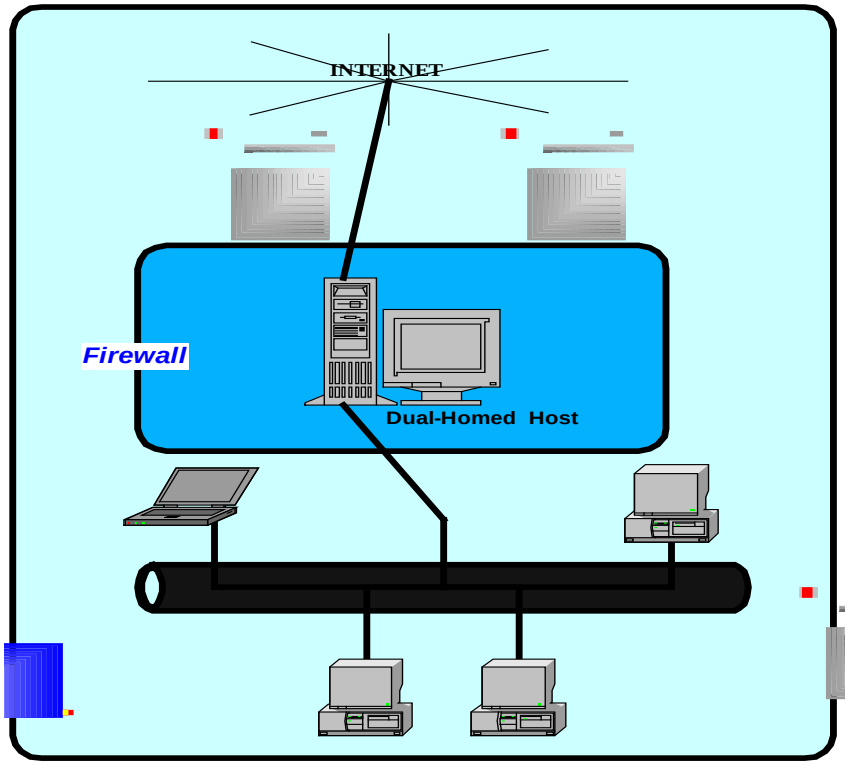
3 Αρχιτεκτονικές σχεδίασης των Firewalls

Σε αυτό το μέρος θα δούμε διάφορους τρόπους που μπορούμε να συνδυάσουμε στοιχεία των firewalls.

3.1 Η Αρχιτεκτονική Dual-Homed Host

Όπως καταλαβαίνουμε από τον τίτλο αυτή η αρχιτεκτονική είναι οικοδομημένη γύρω απ' το dual-homed host υπολογιστή, ένας υπολογιστής ο οποίος έχει τουλάχιστον δύο διεπαφές δικτύου. Ένας τέτοιος υπολογιστής θα μπορούσε να δουλεύει και σαν δρομολογητής μεταξύ των δικτύων που συνδέονται στις διεπαφές, έχει δηλαδή την ικανότητα να δρομολογεί πακέτα IP από το ένα δίκτυο στο άλλο. Όμως για να λειτουργήσει ένα firewall με αυτή την αρχιτεκτονική απενεργοποιούμε αυτή τη λειτουργία της δρομολόγησης. Έτσι, πακέτα από το ένα δίκτυο δεν δρομολογούνται απ' ευθείας στο άλλο. Συστήματα μέσα από το firewall μπορούν να επικοινωνήσουν με τον dual-homed host, εξωτερικά συστήματα μπορούν να επικοινωνήσουν με τον dual-homed host, αλλά δεν μπορούν να επικοινωνήσουν μεταξύ τους. Η IP κίνηση μεταξύ τους είναι εντελώς μπλοκαρισμένη.

Η αρχιτεκτονική του δικτύου με ένα dual-homed host firewall host είναι σχετικά απλή: ο dual-homed host κάθεται μεταξύ των δικτύων και είναι συνδεδεμένος σε αυτά (Εικόνα 3.1.1).



Αρχιτεκτονική του Dual-homed host 3.1.1

Οι Dual-homed hosts μπορούν να προσφέρουν υψηλό επίπεδο ελέγχου. Εάν δεν επιτρέπουμε την κίνηση μεταξύ εσωτερικού και εξωτερικού δικτύου και εντοπίσουμε πακέτο στο εσωτερικό δίκτυο το οποίο έχει εξωτερική πηγή σημαίνει ότι κάπου υπάρχει πρόβλημα ασφάλειας. Σε κάποιες περιπτώσεις ο Dual-homed host μπορεί να αποτρέψει συνδέσεις που ισχυρίζονται ότι είναι από μια συγκεκριμένη υπηρεσία αλλά δεν περιέχουν τη σωστή, για την υπηρεσία αυτή, δεδομένα, κάτι στο οποίο δεν τα καταφέρνει ένα φίλτρο πακέτων σε αυτό το επίπεδο ελέγχου.

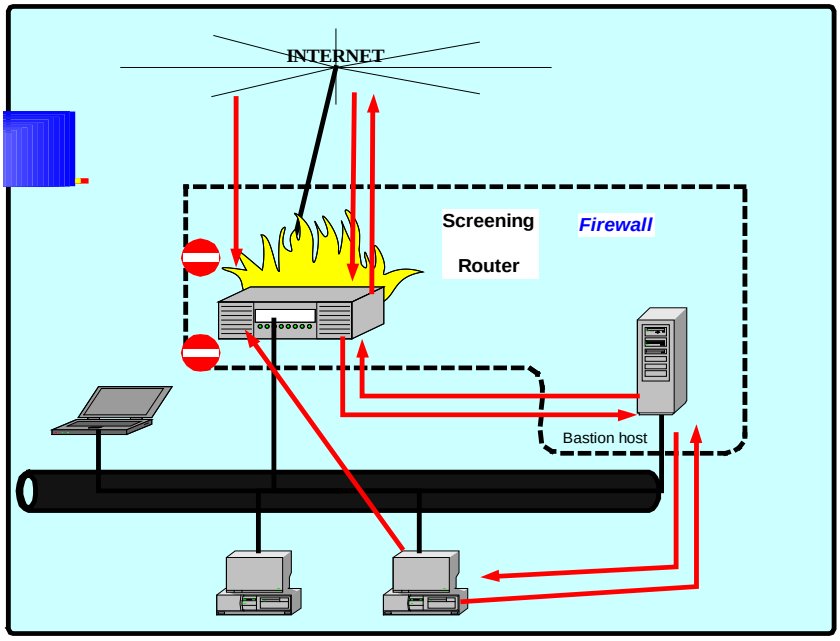
Ένας Dual-homed host μπορεί να προσφέρει υπηρεσίες μόνο μέσω της τεχνικής proxy ή βάζοντας τους χρήστες να συνδέονται απ’ ευθείας με τον Dual-homed host. Όμως, λόγω προβλημάτων στην ασφάλεια που κρύβουν οι λογαριασμοί χρηστών και ιδιαίτερα σ’ έναν Dual-homed host και λόγω του ότι

οι χρήστες δεν το βλέπουν σαν κάτι το εύκολο να συνδέονται σ ‘αυτόν, αυτή η μέθοδος δε χρησιμοποιείται ιδιαίτερα.

Η τεχνική του proxy παρουσιάζει πολύ λιγότερα προβλήματα αλλά μπορεί να μην προσφέρεται για όλες τις υπηρεσίες τις οποίες θέλουμε.

3.2 Αρχιτεκτονική του Screened Host

Η αρχιτεκτονική του screened host προσφέρει υπηρεσίες από ένα host ο οποίος είναι συνδεδεμένος μόνο στο εσωτερικό δίκτυο, χρησιμοποιώντας έναν ξεχωριστό router. Σε αυτή την αρχιτεκτονική, η βασική ασφάλεια παρέχεται από κάποιο φίλτρο πακέτων.(π.χ. τα φίλτρα πακέτων είναι αυτά που αποτρέπουν κάποιον από το να βγει έξω από το δίκτυο παρακάμπτοντας έναν proxy). Στην εικόνα 3.2.1 βλέπουμε μια απλή εκδοχή της αρχιτεκτονικής του screened host.



Αρχιτεκτονική του Screened Host 3.2.1

Ο bastion host βρίσκεται στο εσωτερικό δίκτυο. Το φίλτρο πακέτων στο screening router είναι έτσι ρυθμισμένος ώστε ο bastion host να είναι ο μόνος host του δικτύου που μπορεί να ανοίξει συνδέσεις προς το Διαδίκτυο (π.χ. για να παραδώσει την εισερχόμενη αλληλογραφία). Ακόμα και τότε δεν επιτρέπονται όλων των ειδών οι συνδέσεις. Οποιοδήποτε εξωτερικό σύστημα

αποπειραθεί να προσπελάσει εσωτερικά συστήματα ή υπηρεσίες θα συνδεθεί σε αυτόν τον host. Συνεπώς θα πρέπει να διατηρεί ένα υψηλό επίπεδο ασφάλειας host.

Τα φιλτράρισμα των πακέτων επιτρέπει επίσης στο bastion host να ανοίξει επιτρεπτές συνδέσεις προς τον έξω κόσμο.

3.3 Αρχιτεκτονική του Screened Subnet

Η αρχιτεκτονική του *screened subnet* προσθέτει ένα επιπλέον επίπεδο ασφάλειας στην αρχιτεκτονική του screened host προσθέτοντας ένα περιμετρικό δίκτυο που απομονώνει ακόμα παραπάνω το εσωτερικό δίκτυο από το Διαδίκτυο.

Λόγω της φύσης τους οι bastion hosts είναι τα πιο ευάλωτα μηχανήματα στο δίκτυο. Παρόλο τις προσπάθειες που γίνονται για την ασφάλισή του, είναι τα μηχανήματα που έχουν τις μεγαλύτερες πιθανότητες να δεχτούν επίθεση και αυτό επειδή αυτά είναι τα μηχανήματα *που μπορούν* να δεχτούν επίθεση.

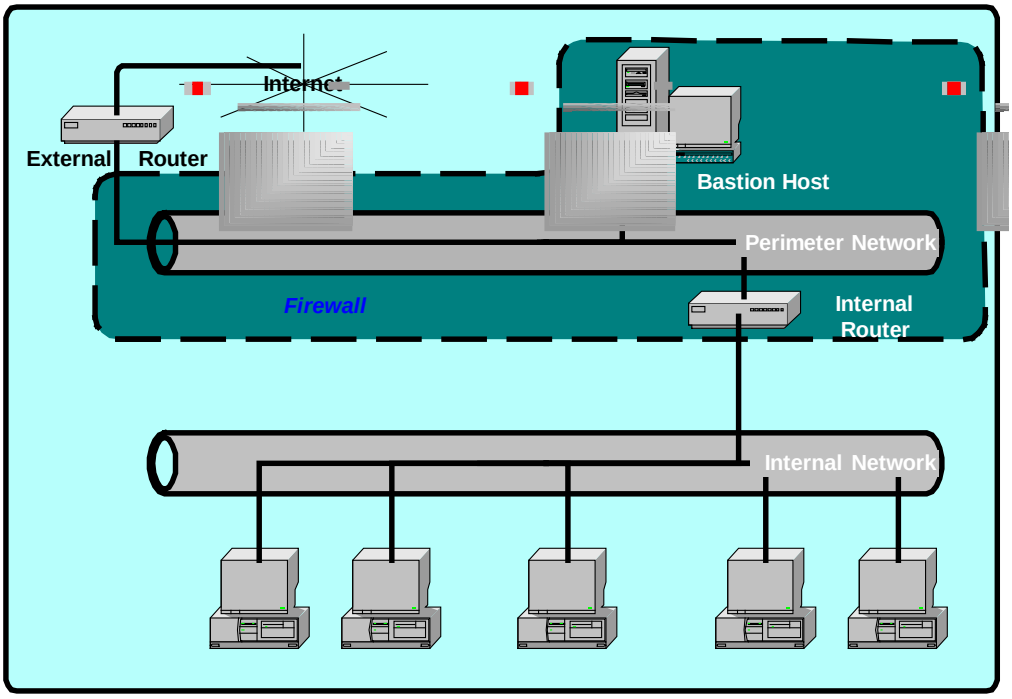
Απομονώνοντας τον bastion host σε ένα περιμετρικό δίκτυο μπορείς να μειώσεις την επίδραση μιας εισβολής σε αυτόν. Δίνει στον εισβολέα κάποια πρόσβαση αλλά όχι ολοκληρωτική.

Με την απλούστερη μορφή της αρχιτεκτονικής του screened subnet υπάρχουν δύο screening routers οι οποίοι συνδέονται και οι δύο στο περιμετρικό δίκτυο. Ο ένας βρίσκεται ανάμεσα στο περιμετρικό δίκτυο και στο εσωτερικό δίκτυο και ο άλλος βρίσκεται ανάμεσα στο περιμετρικό δίκτυο και στο εξωτερικό δίκτυο, συνήθως το Διαδίκτυο. Με αυτή την αρχιτεκτονική ένας εισβολέας θα πρέπει να περάσει και από τους δύο δρομολογητές για να φτάσει στο εσωτερικό δίκτυο. Ακόμα και αν ο επιτιθέμενος καταφέρει να εισβάλει στον bastion host θα πρέπει ακόμα να περάσει και τον εσωτερικό δρομολογητή. Έτσι εξαλείφεται η εκδοχή του ενός τρωτού σημείου που θα θέσει σε άμεσο κίνδυνο το εσωτερικό μας δίκτυο.

Ορισμένα site φτιάχνουν μια σειρά από επίπεδα περιμετρικών δικτύων μεταξύ του εξωτερικού και του εσωτερικού τους δικτύου. Οι λιγότερο έμπιστες υπηρεσίες και πιο ευάλωτες τοποθετούνται στα εξωτερικά περιμετρικά δίκτυα, όσο πιο μακριά από τ εσωτερικό δίκτυο. Η βασική ιδέα αυτής της δομής είναι να προσθέσουμε επίπεδα ασφάλειας στην πορεία προς τα περιμετρικά δίκτυα

ώστε να δυσκολέψουμε το έργο του εισβολέα, που έχει καταφέρει να παρακάμψει τα ανώτερα, και ασθενέστερα, περιμετρικά δίκτυα και θα προσπαθήσει να εισβάλει στο εσωτερικό μας δίκτυο. Αυτή η δομή βέβαια δεν έχει κανένα νόημα αν τα φίλτρα πακέτων επιτρέπουν στα ίδια πράγματα να περάσουν, δηλαδή να έχουν τους ίδιους κανόνες. Έτσι δεν έχουμε καμία αύξηση ασφάλειας.

Μια απεικόνιση της αρχιτεκτονικής των screened subnets φαίνεται στην εικόνα 3.3.1



Αρχιτεκτονική του Screened Subnet 3.3.1

3.3.1 Ορισμένες διαφοροποιήσεις των αρχιτεκτονικών

Ασφαλείς διαφοροποιήσεις:

- **Πολλαπλούς Bastion Host σε περιμετρικό δίκτυο**
- **Συγχώνευση εξωτερικού και εσωτερικού screening router ενός περιμετρικού δικτύου**
- **Συγχώνευση του bastion host με τον εξωτερικό screening router**
- **Χρήση πολλών εξωτερικών screening router**
- **Χρήση πολλών περιμετρικών δικτύων**
- **Χρήση dual-homed host και screened subnet**

Μη ασφαλείς διαφοροποιήσεις:

- **Συγχώνευση του εσωτερικού screening router με τον bastion host**
- **Χρήση πολλών εσωτερικών screening router**

3.4 Φιλτράρισμα Πακέτων (Packet Filtering)

Το φιλτράρισμα πακέτων είναι ένας μηχανισμός ασφάλειας δικτύων ο οποίος ελέγχει τα δεδομένα που ρέουν προς και από ένα δίκτυο. Επιτρέπει (allow, accept) ή αποτρέπει (deny, reject) τη μεταφορά δεδομένων βασιζόμενο:

- Στη διεύθυνση από όπου έρχονται τα δεδομένα
- Στη διεύθυνση στην οποία πηγαίνουν τα δεδομένα
- Τα πρωτόκολλα εφαρμογής και μεταφοράς που χρησιμοποιούνται για την μεταφορά των δεδομένων.

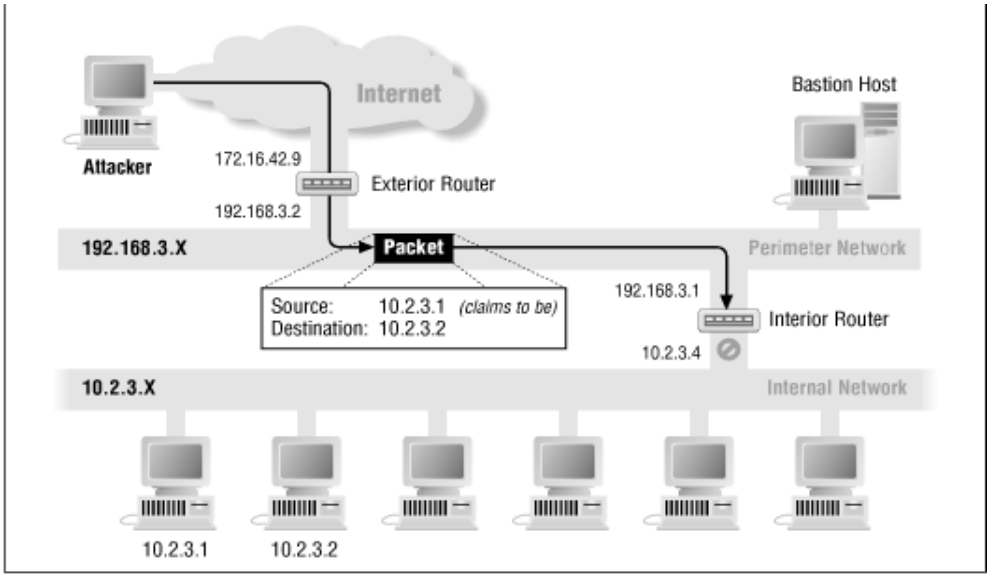
Τα περισσότερα φίλτρα πακέτων δεν παίρνουν αποφάσεις για την μοίρα των δεδομένων βασιζόμενα στα δεδομένα. Αποφάσεις που μπορεί να πάρει είναι του τύπου: Να μην επιτραπεί σε κανέναν από έξω να συνδεθεί προς τα μέσα με telnet, να επιτραπεί σε όλους να στείλουν αλληλογραφία με smtp, αυτός ο υπολογιστής μπορεί να στείλει ειδήσεις με nntp αλλά κανένας άλλος δεν μπορεί.

Δεν επιτρέπει εκφράσεις του τύπου: αυτός ο χρήστης μπορεί να χρησιμοποιήσει telnet για να συνδεθεί από έξω αλλά κανένας άλλος δεν μπορεί επειδή το "χρήστης" δεν είναι κάτι που μπορεί να αναγνωρίσει ένα φίλτρο πακέτων. Όπως επίσης η έκφραση "μετέφερε αυτό το αρχείο αλλά κανένα άλλο" δεν είναι αναγνωρίσιμη.

Το βασικό πλεονέκτημα των φίλτρων πακέτων είναι η ισχύς. Σου επιτρέπει την παροχή ασφάλειας σε ένα μοναδικό σημείο, για συγκεκριμένους τρόπους προστασίας που αφορούν όλο το δίκτυο. Αν, για λόγους ασφάλειας, έχουμε απενεργοποιήσει όλους τους telnet server στους υπολογιστές του δικτύου τότε τι γίνεται σε περίπτωση που κάποιος απ' τον οργανισμό φέρει μαζί του ένα νέο υπολογιστή ή εγκαταστήσει μόνος του έναν; Έχοντας ένα φίλτρο πακέτων όμως το οποίο αποτρέπει το telnet από έξω μας καθησυχάζει γιατί δεν έχει πια σημασία αν υπάρχει telnet server μέσα το δίκτυο.

Ορισμένα είδη προστασίας μπορούν να προσφερθούν *μόνο* από δρομολογητές φιλτραρίσματος και μπορούν να αναπτυχθούν σε συγκεκριμένες τοποθεσίες του

δικτύου μας. Για παράδειγμα μια καλή ιδέα είναι να απορρίπτουμε όλα τα πακέτα τα οποία έχουν εσωτερική διεύθυνση πηγής, δηλαδή πακέτα τα οποία ισχυρίζονται ότι έρχονται από το εσωτερικό δίκτυο αλλά στην πραγματικότητα έρχονται από το εξωτερικό δίκτυο γιατί τέτοια πακέτα είναι συνήθως μέρος μιας επίθεσης που ονομάζεται *address-spoofing.* Σε τέτοιες επιθέσεις ο επιτιθέμενος προσποιείται ότι έρχεται από το εσωτερικό μας δίκτυο. Αποφάσεις τέτοιου είδους μπορούν να παρθούν μόνο από ένα δρομολογητή φιλτραρίσματος ο οποίος βρίσκεται στην περίμετρο του δικτύου μας. Μόνο σε έναν δρομολογητή αυτού του είδους ,ο οποίος είναι η διαχωριστική γραμμή μεταξύ εσωτερικού και εξωτερικού δικτύου, έχουμε την δυνατότητα να αναγνωρίσουμε ένα τέτοιο πακέτο ελέγχοντας την διεύθυνση πηγής και το σύνδεση του δικτύου από την οποία προήλθε, από την εσωτερική ή την εξωτερική. Στην εικόνα 2.4.1 βλέπουμε ένα παράδειγμα της πλαστογράφησης της διεύθυνσης πηγής (source address forgery ή και IP spoofing).



IP Spoofing ή πλαστογράφηση διεύθυνσης πηγής 3.3.1

3.4.1 Πλεονεκτήματα του Φιλτραρίσματος Πακέτων

❖ Ένα από τα βασικότερα πλεονεκτήματα ενός φίλτρου πακέτων είναι ότι ένας μόνο, στρατηγικά τοποθετημένος δρομολογητής πακέτων μπορεί να προστατέψει ένα ολόκληρο δίκτυο.

❖ Το φιλτράρισμα πακέτων δεν απαιτεί γνώσεις ή συμμετοχή των χρηστών, όπως για έναν proxy. Δεν απαιτεί συγκεκριμένο λογισμικό ή ρυθμίσεις λογισμικού όπως για έναν proxy. Όταν περάσει ένα πακέτο το φίλτρο δεν υπάρχει καμία διαφοροποίηση από έναν απλό δρομολογητή. Οι χρήστες δεν θα καταλάβουν καν ότι υπάρχει εκτός και αν προσπαθήσουν κάτι το οποίο απαγορεύεται από την πολιτική ασφάλειάς μας. Αυτή η "διαφάνεια" σημαίνει ότι μπορούμε να κάνουμε φιλτράρισμα χωρίς την επίγνωση ή την συμμετοχή των χρηστών, κάτι το οποίο πολλές φορές διευκολύνει πολύ το έργο της ασφάλισης.

❖ Δυνατότητες φιλτραρίσματος πακέτων υπάρχουν στα περισσότερα προγράμματα δρομολόγησης (λογισμικό) και στους περισσότερους δρομολογητές (υλικό) που κυκλοφορούν στο Διαδίκτυο και στο εμπόριο.

3.4.2 Μειονεκτήματα του Φιλτραρίσματος Πακέτων

❖ Παρόλο την πληθώρα υλικών και λογισμικών φίλτρων πακέτων, τα φίλτρα δεν είναι ακόμα ένα τέλειο εργαλείο και συνεπώς έχουν και κάποια μειονεκτήματα. Ένα από αυτά είναι ότι οι κανόνες των φίλτρων είναι αρκετά δύσκολο να οριστούν. Όταν οριστούν, είναι δύσκολο να δοκιμαστούν. Οι δυνατότητες των φίλτρων πακέτων, σε ορισμένα προϊόντα, είναι ατελής με κάνοντας την εφαρμογή επιθυμητών δυνατοτήτων πολύ δύσκολη ή και απίθανη. Όπως κάθε είδος λογισμικού στους υπολογιστές, , μπορεί να υπάρχουν bugs. Στους δρομολογητές φιλτραρίσματος είναι πιο επικίνδυνα από άποψη ασφάλειας από ότι για έναν proxy, γιατί το φίλτρο πακέτων θα αφήσει απλά το πακέτο να περάσει, ενώ ο proxy απλά δεν θα το προωθήσει.

❖ Ορισμένα πρωτόκολλα δεν είναι κατάλληλα για φιλτράρισμα ακόμα και αν έχουμε κάνει μια πάρα πολύ καλή δουλειά στον ορισμό των κανόνων του φίλτρου. Τέτοια είναι τα NFS, NIS/YP (πρωτόκολλα που βασίζονται από το RPC) και οι εντολές "r" (rlogin, rdist, rcp κτλ.).

❖ Ορισμένες πολιτικές δεν μπορούν να εφαρμοσθούν σε ένα φίλτρο πακέτων. Για παράδειγμα, μπορείς να ορίσεις κανόνες με βάση τον host απ΄ τον οποίο έρχονται ή πηγαίνουν κάποια πακέτα αλλά δεν μπορείς να του ορίσεις κανόνες με βάση τον χρήστη. Ακόμα, στα φίλτρα μπορείς να ορίσεις κανόνες με

βάση το πρωτόκολλο μεταφοράς και το port ελπίζοντας ότι σε αυτό είναι ορισμένη η εφαρμογή για την οποία προορίζεται.

3.5 Τι κάνει ένας δρομολογητής τα πακέτα;

Όταν ένας δρομολογητής πακέτων τελειώσει με τον έλεγχο ενός πακέτου υπάρχουν δύο ενέργειες που μπορεί να κάνει: Να προωθήσει το πακέτο, να το δρομολογήσει σαν να ήταν ένας απλός δρομολογητής ή να το απορρίψει αν δεν συμφωνεί με τα κατάλληλα κριτήρια.

3.5.1 Ενέργειες Logging

Πέρα απ' το αν ένα πακέτο προωθείται ή απορρίπτεται, ίσως να θέλουμε να καταγράψουμε τις ενέργειες έκανε ο δρομολογητής μας για τα πακέτα (να κρατήσει δηλαδή ένα *log)*. Αυτό συνήθως συμβαίνει με τα πακέτα που απορρίπτονται για να γνωρίζουμε ποιες ενέργειες έγιναν οι ποίες δεν επιτρέπονταν.

Μπορεί επίσης να θέλουμε να καταγράψουμε τα TCP πακέτα τα οποία επιτρέπονται και ανοίγουν μια σύνδεση για να γνωρίζουμε το πλήθος των εισερχόμενων και εξερχόμενων συνδέσεων.

Διαφορετικές εφαρμογές φιλτραρίσματος πακέτων υποστηρίζουν διάφορες μορφές καταγραφής (logging). Κάποιες μπορεί να καταγράψουν συγκεκριμένες πληροφορίες ενός απορριφθέντος πακέτου και άλλες ολόκληρο το πακέτο. Κάποιες άλλες μπορεί να μην έχουν επιλογές για την καταγραφή πακέτων που επιτρέπονται. Καλό είναι πάντως να κρατάμε ένα αντίγραφο των logs και κάπου αλλού (π.χ. σύνδεση σε κάποιο host μέσω syslog).

3.5.2 Επιστρέφοντας Κωδικούς Σφαλμάτων ICMP

Εάν ένα πακέτο απορριφθεί, ο δρομολογητής μπορεί (ή και όχι) να επιστρέψει ένα μήνυμα κωδικού σφάλματος icmp που να ενημερώνει για το τι απέγινε το πακέτο. Αν αποφασίσουμε να επιστρέφουμε ένα icmp, η απόπειρα σύνδεσης θα αποτύχει αυτομάτως. Σε αντίθετη περίπτωση μπορεί να περάσουν κάποια λεπτά μέχρι το time out.

Υπάρχουν δύο ομάδες από μηνύματα icmp απ' τα οποία μπορούμε να διαλέξουμε :

- Τα γενικά “destination unreachable” (προορισμός απροσπέλαστος)- πιο συγκεκριμένα τα “host unreachable” και “network unreachable”.
- Οι κωδικοί "destination administratively unreachable" – συγκεκριμένα τα "host administratively unreachable" και "network administratively unreachable" codes.

Τα πρώτα φτιάχτηκαν για να δείχνουν ότι κάποια σύνδεση δεν υπάρχει ή δεν δουλεύει. Τα δεύτερα, φτιάχτηκαν συγκεκριμένα για να έχουν τα φίλτρα πακέτων κάτι να επιστρέφουν όταν ένα πακέτο απορρίπτεται.

3.6 Φιλτράροντας τη Διεύθυνση

Ο πιο απλός τρόπος για να φιλτράρουμε ένα πακέτο είναι το φιλτράρισμα της διεύθυνσης. Φιλτράροντας κατ’ αυτόν τον τρόπο περιορίζουμε τη ροή των πακέτων βασιζόμενοι στη διεύθυνση πηγής ή/ και προορισμού, δίχως να λάβουμε υπ’ όψιν το πρωτόκολλο που χρησιμοποιείται. Με τέτοιου είδους φιλτράρισμα μπορούμε να επιτρέψουμε σε ορισμένους εξωτερικούς host να επικοινωνούν με συγκεκριμένους εσωτερικούς ή να αποτρέψουμε μια επίθεση IP Spoofing. Ας πούμε για παράδειγμα ότι θέλουμε να αποτρέψουμε πακέτα με πλαστή διεύθυνση πηγής, θα δίναμε αυτόν τον κανόνα:

Κανόνας	**Κατεύθυνση**	**Διεύθυνση πηγής**	**Διεύθυνση προορισμού**	**Ενέργεια**
A	Μέσα	Εσωτερική	Οποιαδήποτε	Απόρριψη

Πίνακας 3.5.1.1

Δηλαδή, ένα πακέτο, με κατεύθυνση προς τα μέσα (προς το εσωτερικό δίκτυο) και διεύθυνση πηγής κάποια διεύθυνση από το δίκτυό μας και με κατεύθυνση προς οπουδήποτε, απορρίπτεται.

3.6.1 Κίνδυνοι που απορρέουν

Δεν είναι, σε γενικές γραμμές, ασφαλές να εμπιστευόμαστε διευθύνσεις πηγής, γιατί είναι δυνατόν να πλαστογραφηθούν. Εκτός και αν χρησιμοποιείται κάποια μέθοδος κρυπτογραφικής πιστοποίησης μεταξύ μας και του host με τον

οποίο μιλάμε ώστε να είμαστε σίγουροι ότι μιλάμε με τον συγκεκριμένο host. Υπάρχουν δύο είδη επιθέσεων οι οποίες βασίζονται σε πλαστογράφηση (forgery) της διεύθυνσης πηγής:

- ***Source address forgery.*** Σε αυτό το είδος επίθεσης, ο επιτιθέμενος στέλνει πακέτα τα οποία ισχυρίζονται ότι είναι από κάποιον τον οποίο εμπιστευόμαστε κατά κάποιον τρόπο, ελπίζοντας να υπάρξει κάποια αντίδραση από αυτήν την εμπιστοσύνη, χωρίς όμως απαραίτητα να περιμένει να του σταλθούν κάποια πακέτα (Εικόνα 3.3.1).
- ***Man in the Middle forgery.*** Αυτό το είδος επίθεσης βασίζεται στο να υπάρχει κανονική σύνδεση και επικοινωνία, ισχυριζόμενος ότι αυτός είναι ο έμπιστός μας host.

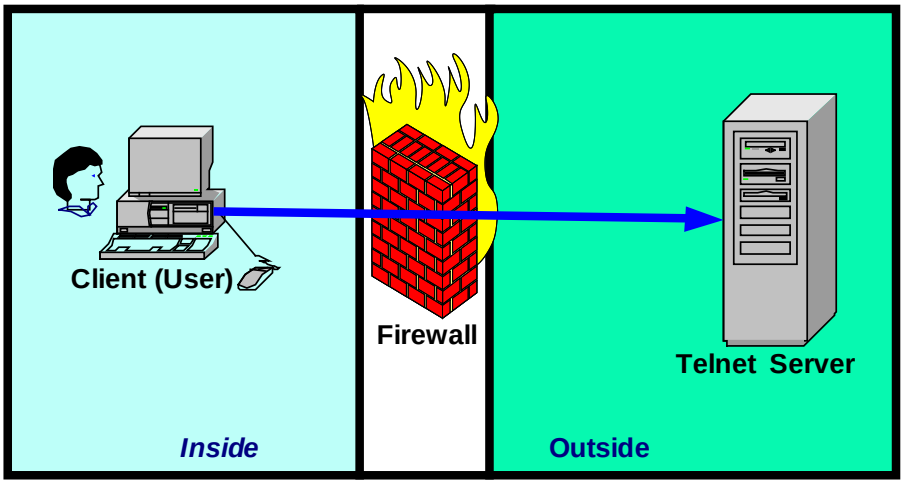
3.7 Φιλτράροντας την Υπηρεσία

Το μπλοκάρισμα των εισερχόμενων πακέτων με πλαστή διεύθυνση πηγής είναι η πιο κοινή χρήση του φιλτραρίσματος με τη διεύθυνση αποκλειστικά. Οι περισσότερες εφαρμογές του φιλτραρίσματος πακέτων αφορούν το φιλτράρισμα της υπηρεσίας.

Θα χρησιμοποιήσουμε ένα παράδειγμα με την υπηρεσία telnet. Θα δούμε την εξερχόμενη κίνηση telnet και την εισερχόμενη κίνηση telnet.

3.7.1 Εξερχόμενο Telnet.

Στην εξερχόμενη υπηρεσία telnet, κατά την οποία ένας τοπικός πελάτης (χρήστης) μιλά με έναν απομακρυσμένο διακομιστή, πρέπει να χειριστούμε και τα εισερχόμενα και τα εξερχόμενα πακέτα.



Εξερχόμενο Telnet 3.7.1.1

Τα εξερχόμενα πακέτα της εξερχόμενης υπηρεσίας περιέχουν την πληκτρολόγηση του χρήστη και αποτελούνται από τα εξής χαρακτηριστικά:

- Η IP διεύθυνση πηγής του εξερχόμενου πακέτου είναι η τοπική IP διεύθυνση της μηχανής του χρήστη.
- Η IP διεύθυνση προορισμού είναι η IP διεύθυνση του απομακρυσμένου διακομιστή.
- Το TCP port προορισμού είναι το 23, το γνωστό για telnet διακομιστές.
- Το TCP port πηγής είναι κάποιος τυχαίος αριθμός άνω του 1023. Θα το αποκαλούμε Y.
- Στο πρώτο εξερχόμενο πακέτο, το οποίο και εγκαθιστά τη σύνδεση, το ACK bit δεν θα είναι ανατεθειμένο. Σε όλα τα υπόλοιπα εξερχόμενα πακέτα θα είναι.

Τα εισερχόμενα πακέτα αυτής της εξερχόμενης υπηρεσίας περιέχουν τα δεδομένα που θα εμφανιστούν στην οθόνη του χρήστη και έχουν τα εξής χαρακτηριστικά:

- Η IP διεύθυνση πηγής του εισερχόμενου πακέτου είναι η IP διεύθυνση του απομακρυσμένου διακομιστή.
- Η IP διεύθυνση προορισμού του εισερχόμενου πακέτου είναι η IP διεύθυνση της μηχανής του τοπικού μας χρήστη.
- Ο τύπος του πακέτου IP είναι TCP.
- Το TCP port πηγής είναι 23, το port που χρησιμοποιεί ο διακομιστής δηλαδή.

- Το TCP port προορισμού είναι το ίδιο “Y” που χρησιμοποιήσαμε ως το port πηγής για τα εξερχόμενα πακέτα.
- Όλα τα πακέτα θα έχουν το ACK bit ανατεθειμένο γιατί έχει γίνει η σύνδεση από το πρώτο εξερχόμενο πακέτο.

3.7.2 Εισερχόμενο Telnet

Στην εισερχόμενη υπηρεσία telnet, ένας απομακρυσμένος πελάτης-χρήστης επικοινωνεί με έναν τοπικό διακομιστή telnet. Ας δούμε και εδώ τι γίνεται με τα εισερχόμενα και εξερχόμενα πακέτα.

Τα εισερχόμενα πακέτα για την εισερχόμενη υπηρεσία telnet περιλαμβάνουν την πληκτρολόγηση του χρήστη και περιέχουν τα ακόλουθα χαρακτηριστικά:

- Η IP διεύθυνση πηγής αυτών των πακέτων είναι η διεύθυνση της μηχανής του απομακρυσμένου χρήστη.
- Η IP διεύθυνση προορισμού είναι η διεύθυνση του τοπικού διακομιστή telnet.
- Ο τύπος του πρωτοκόλλου IP είναι TCP.
- Το TCP port πηγής είναι κάποιος τυχαίος αριθμός μεγαλύτερος του 1023 (Το βαφτίζουμε “Z”).
- Το TCP port προορισμού είναι το 23
- Το TCP ACK bit δεν θα ανατεθεί στο πρώτο εισερχόμενο πακέτο, που εγκαθιστά τη σύνδεση, αλλά θα ανατεθεί σε όλα τα υπόλοιπα εισερχόμενα πακέτα.

Τα εξερχόμενα πακέτα αυτής της εισερχόμενης telnet υπηρεσίας περιέχουν τις αποκρίσεις του διακομιστή μας, δηλαδή αυτά που θα εμφανιστούν στην οθόνη του απομακρυσμένου χρήστη και περιέχουν τα εξής χαρακτηριστικά:

- Η IP διεύθυνση πηγής είναι η διεύθυνση του τοπικού μας διακομιστή
- Η IP διεύθυνση προορισμού είναι αυτή του απομακρυσμένου χρήστη.
- Ο τύπος του πρωτοκόλλου IP είναι TCP.
- Το TCP port πηγής είναι το 23
- Το TCP port προορισμού είναι ο ίδιος τυχαίος αριθμός μεγαλύτερος του 1023 (“Z”).
- Το TCP ACK bit θα ανατεθεί σε όλα τα εξερχόμενα πακέτα.

3.7.3 Σύνοψη

Στον παρακάτω πίνακα βλέπουμε μια σύνοψη των εισερχόμενων και εξερχόμενων υπηρεσιών Telnet. (Πίνακας 3.7.3)

Κατεύθυνση υπηρεσίας	Κατεύθυνση πακέτου	Διεύθυνση πηγής	Διεύθυνση προορισμού	Τύπος πακέτου	PORT πηγής	PORT προορ.	ACK set
Εξερχόμενη	**Εξερχόμενη**	**Εσωτερική**	**Εξωτερική**	**TCP**	**Y**	**23**	**No**
Εξερχόμενη	**Εισερχόμενη**	**Εξωτερική**	**Εσωτερική**	**TCP**	**23**	**Y**	**Yes**
Εισερχόμενη	**Εισερχόμενη**	**Εξωτερική**	**Εσωτερική**	**TCP**	**Z**	**23**	**No**
Εισερχόμενη	**Εξερχόμενη**	**Εσωτερική**	**Εξωτερική**	**TCP**	**23**	**Z**	**Yes**

Συνοπτικός πίνακας για υπηρεσία TELNET 3.7.3.1

Αν θέλαμε να επιτρέψουμε εξερχόμενο telnet, αλλά τίποτα άλλο θα δίναμε τους κανόνες:

Rule	**Direc-tion**	**Source Address**	**Dest. Address**	**Pro-tocol**	**Source Port**	**Dest. Port**	**ACK Set**	**Action**
A	Out	Internal	Any	TCP	>1023	23	Either	Permit
B	In	Any	Internal	TCP	23	>1023	Yes	Permit
C	Either	Any	Any	Any	Any	Any	Either	Deny

Εξερχόμενο Telnet 3.7.3.2

- Ο κανόνας A επιτρέπει εξερχόμενα προς τους απομακρυσμένους διακομιστές telnet.
- Ο κανόνας B επιτρέπει τα πακέτα που επιστρέφουν από αυτήν τη σύνδεση να εισέλθουν. Ελέγχει αν το ACK bit είναι ανατεθειμένο έτσι ώστε να μην είναι δυνατόν να ανοιχτεί μια εισερχόμενη TCP σύνδεση από το port 23 της άλλης πλευράς σε ένα port μεγαλύτερο του 1023 στη δική μας πλευρά από κάποιον που μας επιτίθεται.
- Ο κανόνας C είναι ο προεπιλεγμένος κανόνας. Αν κανένας από τους προηγούμενους κανόνες δεν ταιριάξει, το πακέτο απορρίπτεται.

3.7.4 Κίνδυνοι που απορρέουν από το φιλτράρισμα της Υπηρεσίας

Υπάρχει ένα θεμελιώδες πρόβλημα με αυτόν τον τρόπο φιλτραρίσματος: μπορούμε να εμπιστευτούμε το port πηγής μόνο τόσο όσο εμπιστευόμαστε τον host τον οποίο χαρακτηρίζει.

Αν κατά λάθος μείνει ανοιχτό ένα port το οποίο δεν εξυπηρετείται από κάποια υπηρεσία, ένας απομακρυσμένος χρήστης θα μπορούσε να τρέξει έναν διακομιστή ή ένα πρόγραμμα πελάτη σε αυτό το port.

Όπως είπαμε και νωρίτερα, δεν μπορούμε να εμπιστευόμαστε μια διεύθυνση πηγής ποτέ. Αυτό που μπορούμε να κάνουμε είναι να περιορίσουμε τα ανοιχτά port όσο πιο πολύ μπορούμε. Πρέπει να μένουν ανοιχτά μόνο τα port στα οποία αντιστοιχούν κάποιες έμπιστες υπηρεσίες-διακομιστές, τους οποίους πρέπει να ασφαλίζουμε και αυτούς όσο μπορούμε περισσότερο.

Επειδή πολλές υπηρεσίες χρησιμοποιούν τυχαίους αριθμούς port άνω του 1023 για τους πελάτες και επειδή ορισμένες υπηρεσίες κάνουν το ίδιο για κάποιους διακομιστές πρέπει συχνά να αποδεχόμαστε εισερχόμενα πακέτα για port τα οποία μπορεί να έχουν μη-έμπιστους διακομιστές. Με το TCP μπορούμε να αποδεχόμαστε εισερχόμενα πακέτα χωρίς να αποδεχόμαστε εισερχόμενες συνδέσεις με το να απαιτούμε το ACK bit να είναι ορισμένο. Με το UDP δεν έχουμε αυτή την επιλογή γιατί δεν υπάρχει αντίστοιχο ACK bit.

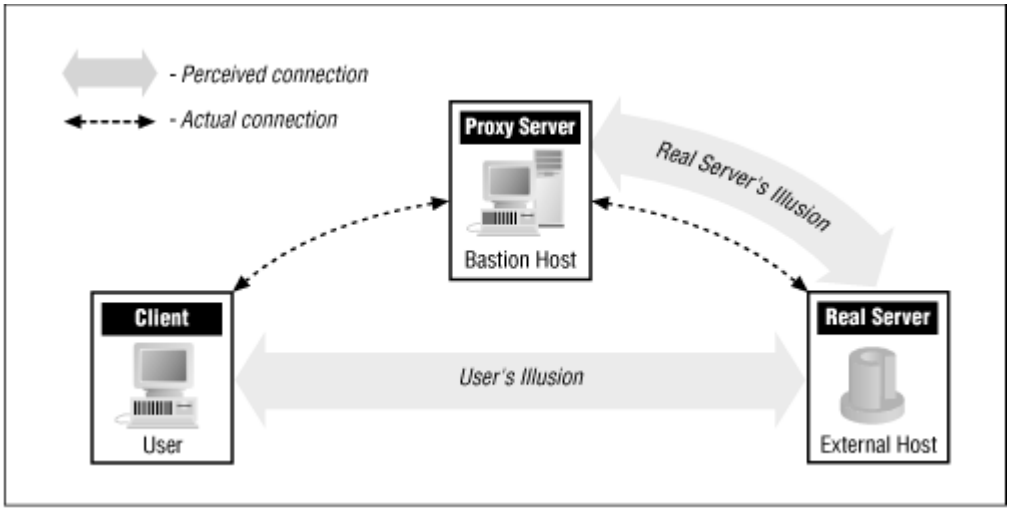
Μέρος Τρίτο

Proxy Servers

4 Διακομιστές διαμεσολάβησης- Proxy Servers

Ένας proxy είναι ένας μεσάζων σε μια διαδικτυακή συναλλαγή. Είναι μια εφαρμογή βρίσκεται μεταξύ ένα πελάτη και έναν πραγματικό διακομιστή. Χρησιμοποιούνται πάρα πολύ συχνά και για firewalls για την παροχή ασφάλειας. Επιτρέπουν και μπορούν και να καταγράψουν αιτήσεις από το εσωτερικό μας δίκτυο προς το εξωτερικό δίκτυο.

Ένας proxy συμπεριφέρεται και σαν πελάτης, για τους εξωτερικούς διακομιστές, και σαν διακομιστής για τους εσωτερικούς πελάτες. Ο proxy δέχεται και επεξεργάζεται αιτήσεις από τους εσωτερικούς πελάτες και μετά τις προωθεί, σαν δικές του αιτήσεις προς τους εξωτερικούς διακομιστές. Όταν απαντήσει ένας εξωτερικός διακομιστής στον proxy, αυτός προωθεί τις απαντήσεις στον κατάλληλο πελάτη του δικτύου. Πολλές φορές οι proxy αναφέρονται και σαν “application layer gateways” (πύλες επιπέδου εφαρμογής). Αυτό το όνομα αντικατοπτρίζει το γεγονός ότι ο proxy βρίσκεται στο επίπεδο εφαρμογής του μοντέλου OSI , όπως και οι πελάτες –διακομιστές.



Διακομιστές Διαμεσολάβησης-Proxy Servers 4.1

Οι proxy βρίσκουν πολλές εφαρμογές στον κόσμο των δικτύων. Μερικές από αυτές είναι:

- logging. Η καταγραφή συμβάντων
- Access controls.

- Φιλτράρισμα.
- Μετάφραση-μεταγλώττιση
- Έλεγχος για ιούς.
- Caching.
- Reverse proxy
- Reverse hosting
- Server proxying

Τα βασικότερα πλεονεκτήματα ενός proxy server είναι

- ***Η ασφάλεια:*** η δυνατότητα να επιτρέπεις ή να αποτρέπεις την πρόσβαση σε εξωτερικούς διακομιστές με την χρήση κάποιων "access list".
- ***Καταγραφή συμβάντων:*** Η καταγραφή των κινήσεων των πελατών με πρόσβαση στο εξωτερικό δίκτυο. Αναφορές και στατιστικά μπορούν να γεννηθούν από τα *logs.*
- ***Caching:*** ιστοσελίδες που ζητούνται πολύ συχνά από πελάτες του δικτύου, αποθηκεύονται τοπικά σε κάποιο κοινόχρηστο πόρο και είναι προσβάσιμες για όλους τους τοπικούς πελάτες.. Αυτό εξυπηρετεί διότι κάνουμε εξοικονόμηση του bandwidth (εύρους ζώνης) της σύνδεσης του Internet.

Υπάρχουν δύο τύποι proxy. Ο πρώτος είναι ο application-level proxy (επιπέδου εφαρμογής), ο οποίος καταλαβαίνει την υπηρεσία του επιπέδου εφαρμογής για την οποία κάνει την διαμεσολάβηση. Καταλαβαίνει και μπορεί και διερμηνεύει τις εντολές του πρωτοκόλλου του επιπέδου εφαρμογής που χρησιμοποιείται.

4.1 Πώς λειτουργούν οι Proxy Servers και οι Transparent Proxy Servers

Ας υποθέσουμε ότι έχουμε την εταιρία comp.com. Έχουμε ένα εσωτερικό δίκτυο και μια σύνδεση για το Internet, την ppp στον proxy (proxy.comp.com με εξωτερική διεύθυνση την 1.2.3.4 και εσωτερική την 192.168.0.1). έχουμε και έναν host που θα τον λέμε "εγώ" και έχει διεύθυνση 192.168.0.100.

4.1.1 Παραδοσιακό proxy

Σε αυτό το σενάριο, τα πακέτα από το ιδιωτικό δίκτυο προς το Internet ποτέ δεν θα το διασχίσουν και το αντίθετο. Οι διευθύνσεις του δικτύου πρέπει να είναι εσωτερικές (οι 192.168.*.*, 10.*.*.*, 172.16.*.* - 172.31.*.* δεν είναι πραγματικές διευθύνσεις του Internet). Ο μόνος τρόπος που βγαίνει κάποιο πακέτο προς το Internet είναι από τον proxy-firewall. Έχουμε εγκαταστήσει τον proxy μας στο port 8080. Ο "εγώ" έχει ρυθμίσει τον φυλλομετρητή του να χρησιμοποιεί τον proxy στο port 8080. Στο ιδιωτικό δίκτυο δεν χρειάζεται να ορίσουμε gateway.

Δίνουμε στο φυλλομετρητή του host την διεύθυνση http://www.teiep.gr. Ο φυλλομετρητής μας πηγαίνει στον proxy port 8080 χρησιμοποιώντας για εαυτό του το port 1100. Του ζητά την σελίδα. Εάν την έχει στην cache του την επιστρέφει. Αν όχι, τότε ψάχνει το www.teiep.gr και βρίσκει την διεύθυνση Α.Β.Γ.Δ Ανοίγει τότε μια σύνδεση προς αυτόν από το port 1123 στο port 80 του διακομιστή και ζητά την ιστοσελίδα. Καθώς την παραλαμβάνει, την κρατά στην cache και την προωθεί στη σύνδεση προς το φυλλομετρητή του "εγώ".

Από την πλευρά του teiep.gr, η σύνδεση γίνεται από το 1.2.3.4 της ppp σύνδεσης του proxy με port 1123 προς το Α.Β.Γ.Δ στο port 80 του διακομιστή του. Από την πλευρά του "εγώ", η σύνδεση γίνεται από το 192.168.0.100 με port 1100 προς το 192.168.0.1, την εσωτερική διασύνδεση του proxy, στο port 8080.

4.1.2 Διαφανής Proxy (Transparent proxy)

Και σε αυτό το σενάριο, τα πακέτα από το ιδιωτικό δίκτυο προς το Internet ποτέ δεν θα το διασχίσουν και το αντίθετο. Οι διευθύνσεις του δικτύου μας είναι και εδώ ιδιωτικές. Ο μόνος τρόπος που βγαίνει κάποιο πακέτο προς το Internet είναι από τον proxy-firewall ο οποίος συνδέεται και στα δύο δίκτυα. Τρέχουμε ένα πρόγραμμα για transparent proxying (διαφανής διαμεσολάβηση) και τα πακέτα που εξέρχονται από αυτό το μηχάνημα, αλλάζουν προορισμό και πηγαίνουν προς τον transparent proxy. Διαφανής διαμεσολάβηση (transparent proxy) σημαίνει ότι οι πελάτες δεν χρειάζεται ότι ανακατεύεται ένας proxy.

Θα χρησιμοποιήσουμε ένα παρόμοιο παράδειγμα με το προηγούμενο. Οι διευθύνσεις του δικτύου είναι ίδιες με το παραπάνω παράδειγμα, καθώς και του "εγώ", του transparent proxy-firewall και του www.teiep.gr. Ο proxy είναι

εγκατεστημένος στο port 8080. Ο πυρήνας κάνει ανακατεύθυνση του port πηγής 80 προς το port 8080 του proxy. Ο φυλλομετρητής μας είναι ρυθμισμένος να συνδέεται απ' ευθείας. Το gateway στο ιδιωτικό μας δίκτυο πρέπει να δείχνει στον proxy – firewall (gw:192.168.0.1).

Δίνουμε στο φυλλομετρητή μας την διεύθυνση www.teiep.gr. Τότε ανοίγει μια σύνδεση προς αυτή τη διεύθυνση από το τοπικό port 1050 και ζητά από το διακομιστή την ιστοσελίδα (port 80). Καθώς τα πακέτα από το Εγώ(1050) παιρνούν στο μηχάνημα με τον proxy (80), ανακατευθύνονται στον αναμένοντα proxy στο port 8080. Ο proxy ανοίγει μια σύνδεση από το port 1100 προς το Α.Β.Γ.Δ port 80 όπου πήγαιναν τα πρωτότυπα πακέτα. Καθώς ο proxy παίρνει τα δεδομένα από αυτή τη σύνδεση τα ανακατευθύνει προς τη σύνδεση με τον φυλλομετρητή ο οποίος και ανακατασκευάζει και προβάλλει την ιστοσελίδα.

Από την πλευρά του teiep.gr η σύνδεση γίνεται από το 1.2.3.4 (port 1100) προς το Α.Β.Γ.Δ στο port 80. Από την πλευρά του εγώ, η σύνδεση γίνεται από το 192.168.0.100 port 1050 προς το Α.Β.Γ.Δ στο port 80, αλλά στην πραγματικότητα μιλά με τον transparent proxy μας.

4.2 Χαρακτηριστικά ενός Caching Proxy Server

Το βασικότερο χαρακτηριστικό ενός caching proxy είναι η δυνατότητα να αποθηκεύει απαντήσεις άλλων διακομιστών για μετέπειτα χρήση, κάτι το οποίο μας γλιτώνει χρόνο και εύρος ζώνης. Συνήθως έχουν και πολλά άλλα πολύ χρήσιμα χαρακτηριστικά που μπορούν να φανούν πολύτιμα. Τα περισσότερα από αυτά είναι λίγο άσχετα με το caching αλλά μπορείς να τα κάνεις μόνο με έναν caching proxy. Για παράδειγμα αν θέλεις να αυθεντικοποιείς τους χρήστες σου αλλά δεν ενδιαφέρεσαι να κάνεις caching, είναι πολύ πιθανό να χρησιμοποιήσεις έναν caching proxy για αυτό το σκοπό.

4.2.1 Authentication (Αυθεντικοποίηση)

Ο proxy μπορεί να ζητά από τους χρήστες του να αυθεντικοποιούνται πριν εξυπηρετήσει οποιεσδήποτε αιτήσεις τους. Αυτό είναι πολύ χρήσιμο για proxy-firewall. Όταν ο κάθε χρήστης έχει το δικό του όνομα χρήστη και κωδικό πρόσβασης, μόνο εξουσιοδοτημένοι χρήστες μπορούν, π.χ. να δουν ιστοσελίδες από το WWW από το δίκτυό μας. Ακόμα, προσφέρει έναν ποιοτικότερο τρόπο παρακολούθησης πιθανών προβλημάτων.

4.2.2 Φιλτράρισμα αιτήσεων

Οι caching proxy συχνά χρησιμοποιούνται για να φιλτράρουν αιτήσεις (Request Filtering) των χρηστών. Οι οργανισμοί συνήθως έχουν κάποιες πολιτικές οι οποίες απαγορεύουν στο προσωπικό την πρόσβαση πορνογραφικού υλικού τις ώρες εργασίας. Για την ενίσχυση της εφαρμογής αυτής της πολιτικής μπορεί να ρυθμιστεί ο proxy να απορρίπτει αιτήσεις προς γνωστές πορνογραφικές ιστοσελίδες. Αυτού του είδους το φιλτράρισμα είναι πολλές φορές αμφισβητήσιμο. Πολλοί το εξισώνουν με λογοκρισία και διευκρινίζουν, σωστά συχνά, ότι το φιλτράρισμα αιτήσεων δεν είναι και τέλειο.

4.2.3 Φιλτράρισμα απαντήσεων

Οι proxy μπορούν να φιλτράρουν απαντήσεις (response filtering). Αυτό συνήθως αναφέρεται στον έλεγχο των περιεχομένων ενός αντικειμένου που κατεβάζουμε. Ένα φίλτρο που ελέγχει για ιούς σε λογισμικό είναι ένα καλό παράδειγμα.

4.2.4 Prefetching

Prefetching είναι η διαδικασία ορισμένων δεδομένων προτού ζητηθεί. Συστήματα δίσκων και μνημών συνήθως χρησιμοποιούν τη μέθοδο “Prefetching” επίσης γνωστό και ως “read ahead” (προ διάβασμα). Για τον ιστό συνήθως χρησιμοποιείται για να ανακτήσει υπέρ-συνδέσμους και εικόνες από ένα αρχείο HTML.

Είναι μια ανταλλαγή μεταξύ του χρόνου απόκρισης (latency) και του εύρους ζώνης. Ένας proxy επιλέγει αντικείμενα για το prefetch υποθέτοντας ότι κάποιος πελάτης θα τα ζητήσει. Σωστές προβλέψεις έχουν ως αποτέλεσμα μείωση του χρόνου απόκρισης. Λανθασμένες προβλέψεις ωστόσο χρησιμοποιούν άδικα το εύρος ζώνης.

4.2.5 Μετάφραση και Μετατροπή κώδικα

Η μετάφραση και η μετατροπή του κώδικα αναφέρονται στην επεξεργασία του περιεχομένου χωρίς τη σημαντική μετατροπή του νοήματος ή της εμφάνισης. Σαν παράδειγμα μπορούμε να φανταστούμε μια εφαρμογή η οποία μεταφράζει μια ιστοσελίδα από αγγλικά σε ελληνικά καθώς την κατεβάζει.

Η μετατροπή του κώδικα συνήθως αναφέρεται σε χαμηλού επιπέδου αλλαγές σε ψηφιακά δεδομένα παρά σε υψηλού επιπέδου ανθρώπινη γλώσσα. Η αλλαγή του format μιας εικόνας από giff σε jpeg είναι ένα καλό παράδειγμα. Το νόημα αυτής της διαδικασίας είναι ότι μια εικόνα σε jpeg είναι μικρότερη σε μέγεθος, άρα και μπορούμε να μειώσουμε το χρόνο μεταφοράς. Ένα ζεύγος proxy που συνεργάζονται, μπορούν να συμπιέσουν όλες τις μεταφορές μεταξύ τους και να τις αποσυμπιέσουν πριν φτάσουν στον πελάτη.

4.2.6 Σχηματισμός Κίνησης (Traffic Shaping)

Ένας σημαντικός αριθμός οργανισμών χρησιμοποιούν proxy επιπέδου εφαρμογής για να ελέγχουν τη χρησιμοποίηση του εύρους ζώνης. Κατά μιαν έννοια, αυτή η διαδικασία γίνεται στο επίπεδο δικτύου όπου είναι δυνατόν ο έλεγχος της ροής των πακέτων ωστόσο, το επίπεδο εφαρμογής παρέχει πολύ χρήσιμες, επιπλέον, πληροφορίες που οι διαχειριστές.

Πριν συνεχίσουμε, πρέπει να εξηγήσουμε τι είναι cache hits και cache misses. Τα “cache-hits” είναι οι αιτήσεις σελίδων ή αντικειμένων που βρέθηκαν στην cache και δεν χρειάσθηκε να γίνει καν σύνδεση με τον πραγματικό διακομιστή. Αυτό συνήθως γίνεται επειδή κάποιος χρήστης έχει νωρίτερα ζητήσει την σελίδα και αυτή έχει κρατηθεί στην cache. Αυτή είναι η πλέον επιθυμητή κατάσταση μιας cache-proxy. Τα “cache-misses” είναι οι αιτήσεις οι οποίες δεν ικανοποιήθηκαν από την cache μας και χρειάστηκε η επικοινωνία της cache με τον πραγματικό διακομιστή ώστε να πάρει απάντηση ο πελάτης. Τα ποσοστά αυτών των επιτυχιών και αποτυχιών, καθώς και άλλα στατιστικά στοιχεία μπορούμε να τα καταγράφουμε (logging).

4.3 Transparent Caching

Το transparent caching ή διαφανής caching, λέγεται έτσι γιατί αναχαιτίζει την δικτυακή κίνηση στον φυλλομετρητή. Σε αυτή την κατάσταση, η cache “βραχυκυκλώνει” την διαδικασία ανάκτησης εάν το επιθυμητό αρχείο βρίσκεται στην cache. Τα transparent caches είναι εξαιρετικά χρήσιμα για τις εταιρίες παροχής υπηρεσιών Internet οι φυλλομετρητές δεν χρειάζονται ρύθμιση. Αλλά είναι και ο πιο απλός τρόπος να χρησιμοποιούμε μία cache σ’

ένα ιδιωτικό δίκτυο και αυτό επειδή δεν απαιτούν κάποιο σαφή συντονισμό με άλλες cache.

Ο όρος διαφανής- transparent είναι υπερφορτωμένος, έχοντας διαφορετικές έννοιες κατά περίσταση. Κάποιες φορές εννοείται μια ρύθμιση η οποία παρεμβάλλεται στην κίνηση του port 80 προς εξωτερικούς διακομιστές από κάποιο χρήστη και την παρακάμπτει και κάποιες άλλες εννοείται ένας σημασιολογικά διαφανής proxy που δεν αλλάζει την σημασία ή το περιεχόμενο των αιτήσεων και απαντήσεων του πελάτη. Στην πραγματικότητα δεν υπάρχει πραγματική διαφάνεια, μόνο ημιδιαφάνεια και σίγουρα δεν υφίσταται διαφανής cache.

4.3.1 Πλεονεκτήματα της διαφανούς Caching

Τα πλεονεκτήματα της διαφανούς caching είναι περίπου τα αντίθετα από αυτά του proxy caching. Τα βασικότερα είναι:

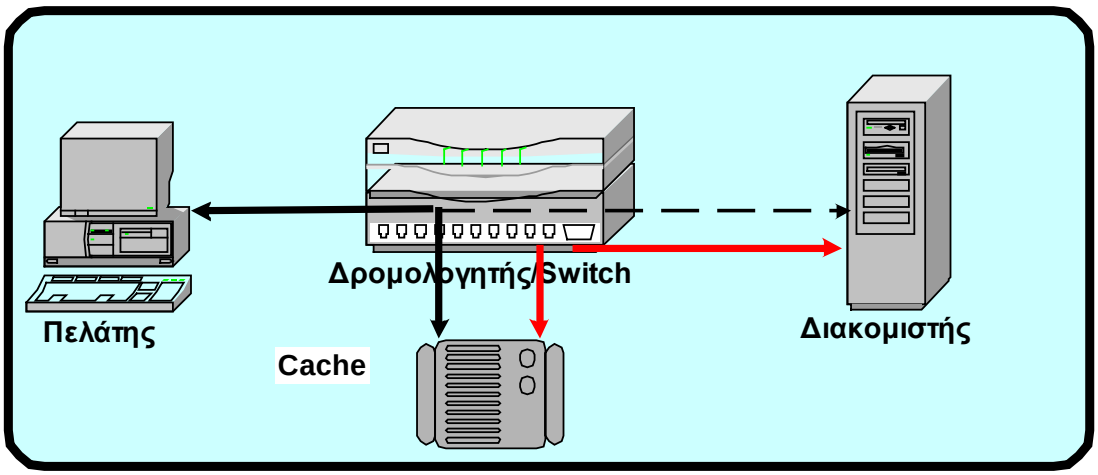
- Εύκολη Διαχείριση- Ο φυλλομετρητής μας δεν χρειάζεται να είναι κατάλληλα ρυθμισμένος για να επικοινωνεί με την cache.
- Κεντρικός Έλεγχος- Ο χρήστης δεν μπορεί να αλλάξει τις ρυθμίσεις του φυλλομετρητή του ώστε να παρακάμψει τον proxy.

4.3.2 Μειονεκτήματα της διαφανούς caching

Κάποια από τα μειονεκτήματα που έχει η διαφανής caching είναι :

- Έλλειψη σταθερότητας- Λόγω του ότι βασίζεται στη σταθερή διαδρομή δρομολόγησης μεταξύ του πελάτη και του πραγματικού διακομιστή, η οποία τυγχάνει να περνά μέσα από μια cached διαδρομή, είναι ευάλωτη σε αλλαγές δρομολόγησης στο Διαδίκτυο. Δηλαδή, αν μια γίνει σύνδεση ενός πελάτη με μια cache και συμβεί μια αλλαγή δρομολόγησης η οποία αναγκάζει τον πελάτη να πάρει μια διαδρομή η οποία δεν περνά από την συσκευή που έκανε την εκτροπή, η συνεδρία θα διακοπεί και ο χρήστης θα πρέπει να ξαναζητήσει την σελίδα. Αν, διαδρομές στο Διαδίκτυο αλλάζουν συνεχώς τότε τα αποτελέσματα θα είναι ακόμα πιο απρόβλεπτα.

- Έλεγχος χρηστών- Η διαφανής caching παίρνει τον έλεγχο από το χρήστη. Πολλοί χρήστες έχουν σοβαρές προκαταλήψεις σε ότι αφορά το caching και θα άλλαζαν παροχέα για να την αποφύγουν ή να την αποκτήσουν.
- Προαπαίτηση φυλλομετρητών- Πολλές διαφανής cache έχουν συγκεκριμένες απαιτήσεις από τους φυλλομετρητές των πελατών, δηλαδή στην κεφαλίδα του πακέτου να αναγράφεται το όνομα του host για τον οποίο προορίζεται και αυτό διότι οι cache αυτές δεν μπορούν να προσπελάσουν την IP διεύθυνση προορισμού από την IP διεύθυνση του πακέτου. Δηλαδή, σε περίπτωση που δεν υπάρχει η ζητούμενη στην cache, δεν μπορούν να καταλάβουν ποιος είναι ο πραγματικός διακομιστής για να ζητήσουν από αυτόν τη σελίδα. Αν και σήμερα, πάνω του 90% των φυλλομετρητών παρέχουν αυτό το χαρακτηριστικό.



Διαφανής (Transparent) Caching 4.3.1

4.3.3 Η Δρομολόγηση

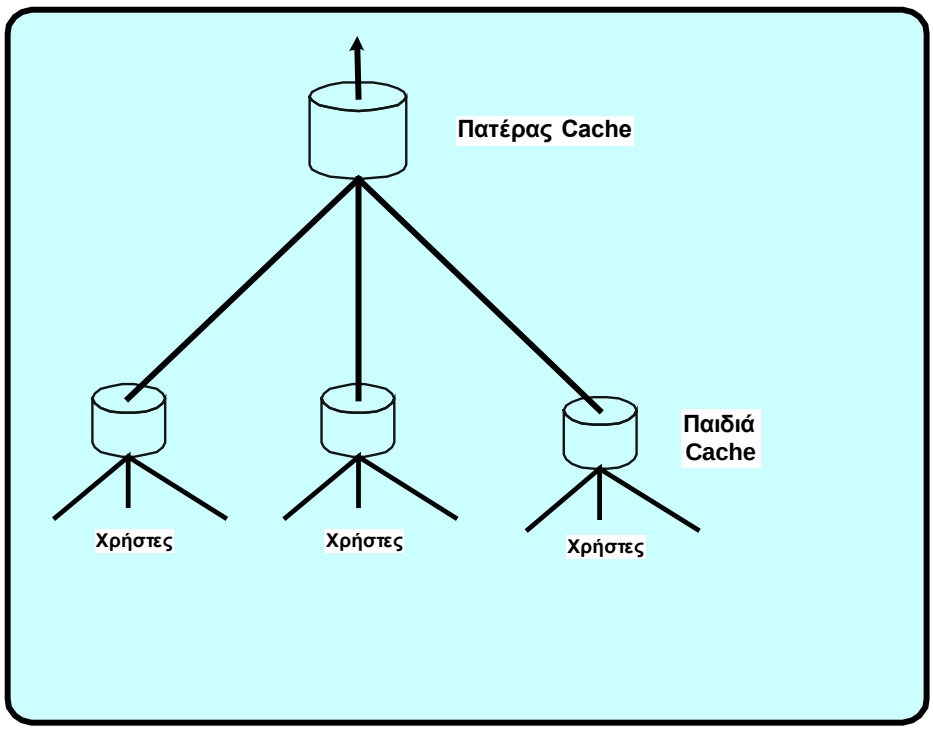
Η διαδικασία της "απαγωγής" των πακέτων ξεκινά στο επίπεδο δικτύου (IP), όπου όλα τα IP πακέτα δρομολογούνται μεταξύ κόμβων. Σε αυτό το επίπεδο ένας δρομολογητής ή ένα switch αναγνωρίζει πακέτα HTTP και τα εκτρέπει προς μια cache αντί να τα προωθήσει στον αρχικό προορισμό. Υπάρχουν αρκετοί τρόποι για να πετύχουμε την απαγωγή:

- ***Inline*** **–** Ένα Inline cache είναι μια συσκευή η οποία συνδυάζει δρομολόγηση (ή και γεφύρωμα δικτύων-bridge) και caching ιστού σε ένα κομμάτι υλικού. Ένα παράδειγμα μπορεί να είναι ένας υπολογιστής με δύο ή περισσότερες κάρτες δικτύου το οποίο έχει για λειτουργικό σύστημα Linux ή Unix και σε αυτό τρέχει ο Squid cache-proxy.
- ***Επιπέδου 4 Switch*** **–** Το switch δουλεύει συνήθως στο επίπεδο 2 (επίπεδο σύνδεσης δεδομένων). Ένα switch επιπέδου 4 μπορεί να παίρνει και αποφάσεις προώθησης βασιζόμενο στα χαρακτηριστικά του επιπέδου 4 (επίπεδο μεταφοράς), όπως είναι οι διευθύνσεις IP και αριθμοί port του TCP. Χρησιμοποιούνται επίσης και για εξισορρόπηση του φόρτου του διακομιστή.
- ***Web Cache Coordination Protocol*** **–** Το **WCCP** είναι ένα πρωτόκολλο της Cisco Systems που απαιτεί την υλοποίησή του σ' ένα δρομολογητή (ή ακόμα και ένα switch) και στην cache. Αποτελείται από δύο συστατικά μέρη. Το πρώτο είναι το πρωτόκολλο ελέγχου και το δεύτερο είναι ο μηχανισμός ανακατεύθυνσης της κίνησης.
- ***Cisco Policy Routing*** **–** Η πολιτική δρομολόγησης (policy routing) αναφέρεται στην ικανότητα ενός δρομολογητή να παίρνει αποφάσεις για την προώθηση βασιζόμενο όχι μόνο στην διεύθυνση προορισμού του πακέτου. Μπορούμε να το χρησιμοποιήσουμε αυτό για να ανακατευθύνουμε πακέτα βασιζόμενοι στους αριθμούς port προορισμού.

4.4 Ιεραρχίες Cache

Μια ιεραρχία από cache είναι μια διευθέτηση από cache που συνεργάζονται μεταξύ τους. Σε μια τέτοια ιεραρχία, οι cache των κατώτερων επιπέδων προωθούν αποτυχίες (cache misses) της cache προς τα ανώτερα επίπεδα ώσπου να αποδεχτεί την αίτηση κάποια άλλη cache ή να προωθηθεί στον πραγματικό διακομιστή. Οι ιεραρχίες είναι ελκυστικές γιατί μπορούν να προσφέρουν βελτιώσεις της αποδοτικότητας. Κάποιες αποτυχίες της cache μας θα είναι επιτυχίες σε κάποιες από τις ανώτερες cache με αποτέλεσμα την μείωση του

εύρους ζώνης του δικτύου και βελτιώνει την ταχύτητα του κατεβάσματος (downloads).



Ιεραρχία Cache 4.4.1

Το παιδί-cache προωθεί τις αποτυχίες του (cache misses) σ' έναν πατέρα-cache. Τότε ο πατέρας του παρέχει μια απάντηση από τη δική του cache, τον πραγματικό διακομιστή ή μια άλλη cache. Ένας πατέρας μπορεί να χρησιμοποιεί bandwidth προς τους πραγματικούς διακομιστές ώστε να ικανοποιήσει την αίτηση της cache του παιδιού.

Οι σχέσεις των αδερφών-cache είναι σχεδιασμένες ώστε να αποτρέπουν μια cache να επιβαρύνει με κάποιο τίμημα μια άλλη cache. Όλες οι αιτήσεις που στέλνονται προς έναν αδερφό-cache θα πρέπει να είναι επιτυχίες (cache hits). Ένας αδερφός δε θα πρέπει ποτέ να δώσει ένα αντικείμενο που έχει ζητηθεί από έναν πατέρα-cache και δεν υπάρχει στην cache του. Αν δεν υπάρχει επιστρέφει ένα μήνυμα ότι αρνείται να προωθήσει την αίτηση. Μια cache επικοινωνεί με τα παιδιά-cache χρησιμοποιώντας ένα από τα πρωτόκολλα *"Intercache"* . Αυτά τα πρωτόκολλα επιτρέπουν στις cache να μαθαίνουν εάν μια γειτονική cache έχει κάποιο συγκεκριμένο αντικείμενο στην cache τους. Μια αίτηση πρέπει να σταλθεί μόνο σ' έναν αδερφό εάν το πρωτόκολλο intercache προβλέπει ότι θα είναι επιτυχία cache.

Αυτές οι σχέσεις δεν είναι δεδομένες. Μια cache μπορεί να είναι πατέρας για κάποιες cache και αδερφός για άλλες. Αυτή είναι η πιο χρησιμοποιημένη μορφή ιεραρχίας από τους παροχείς Internet. Μια απεικόνιση αυτής της ιεραρχίας φαίνεται στην εικόνα 4.4.1.

4.4.1 Πλεονεκτήματα της ένταξης σε ιεραρχία

- Αποδοτικότητα. Κλειδιά στο αν η ένταξη σε μια ιεραρχία cache θα είναι αποδοτική, είναι η επιτυχία σε μια γειτονική cache σε περίπτωση που είναι αποτυχία στη δική μας cache, η γειτονική επιτυχία cache να φτάνει σε εμάς πιο γρήγορα από ότι θα έφτανε αν ήταν αποτυχία και ερχόταν από τον πραγματικό διακομιστή. Ακόμα, οι αποτυχίες των ανωτέρων cache δεν θα πρέπει να είναι αισθητά πιο αργές από ότι θα ήταν η απάντηση από τον πραγματικό διακομιστή.
- Μη προεπιλεγμένη δρομολόγηση. Οι πατέρες-cache είναι χρήσιμοι όταν πρέπει να επιβάλλεις την ροή της κίνησης μέσω μιας συγκεκριμένης διαδρομής στο δίκτυο. Ένα διάσημο παράδειγμα είναι όταν έχεις ένα firewall. Θέλεις να περνά όλη η κίνηση του δικτύου μέσω αυτού. Είναι πολύ συνηθισμένο πλέον πολλοί οργανισμοί και πολλές εταιρίες να χρησιμοποιούν ένα διαφανή (transparent) proxy. Οι HTTP συνδέσεις των πελατών ανακατευθύνονται προς έναν caching proxy σε αυτές τις περιπτώσεις και δεν απορρίπτονται. Αν σε μια τέτοια περίπτωση πελάτης είναι ο caching proxy, το firewall proxy είναι ένας πατέρας-cache και ας μην το καταλαβαίνει το παιδί. Ακόμα, μπορούν να χρησιμοποιηθούν πολλαπλοί πατέρες-cache για πολλαπλές συνδέσεις προς τα εξωτερικά δίκτυα όπου το φόρτο μοιράζεται αναλόγως με τις απαιτήσεις.

4.4.2 Μειονεκτήματα της ένταξης σε ιεραρχία

Πριν αποφασίσουμε την ένταξη σε μια ιεραρχία θα πρέπει να γνωρίζουμε και κάποια από τα μειονεκτήματά της.

- Εμπιστοσύνη. Όταν ενταχθούμε σε μια ιεραρχία είναι σαν να λέμε ότι εμπιστευόμαστε όλα τα μέλη της ιεραρχίας απόλυτα. Πιστεύουμε ότι

τα δεδομένα που μας δίνει είναι έγκυρα, δεν έχουν τροποποιηθεί με κάποιο τρόπο. Εμπιστευόμαστε σε όλα τα μέλη την ιδιωτικότητα των αιτήσεών μας.

- Χαμηλά ποσοστά επιτυχίας (Low hit Ratio). Τα ποσοστά επιτυχιών από πατέρες και αδερφούς-cache είναι συνήθως πολύ χαμηλά σε σύγκριση με μια cache η οποία εξυπηρετεί απ' ευθείας τελικούς χρήστες.
- Επιπτώσεις στις δρομολογήσεις. Όσο η απόσταση μεταξύ των δρομολογητών αυξάνεται (hops), αυξάνονται και οι επιπτώσεις στις διαφοροποιήσεις της δρομολόγησης. Αν για παράδειγμα ένας πατέρας έχει πολλές συνδέσεις για το Internet και κάποια από αυτές διακοπεί, ενώ το παιδί-cache έχει και αυτό μια σύνδεση Internet, αν μια απ' τις συνδέσεις του πατέρα κοπεί δεν θα μπορεί να επικοινωνήσει με κάποιους πραγματικούς διακομιστές και θα στέλνει στα παιδιά-cache μηνύματα σφαλμάτων. Όμως, εφόσον υπάρχει και η άλλη σύνδεση του παιδιού, ίσως αυτό να θελήσει να χρησιμοποιήσει αυτήν.
- Η διατήρηση της συνέπειας και της εγκυρότητας, από χρονικής άποψης, μιας σελίδας μεταξύ των μελών της ιεραρχίας είναι δύσκολη διαδικασία. Σε μια περίπτωση όπου ένα παιδί έχει δύο πατέρες-cache και έχουν και οι δύο μια απάντηση πως θα μπορούμε να ξέρουμε ποια απάντηση είναι πιο έγκυρη; Το καλύτερο που μπορούμε να κάνουμε είναι να χρησιμοποιήσουμε μια διαδικασία ακύρωσης αντικειμένων (object invalidation process). Ορισμένα από τα πρωτόκολλα cache έχουν τέτοια χαρακτηριστικά.
- Μεγάλες οικογένειες. Τα πολλά επίπεδα στην ιεραρχία πολλές φορές προκαλούν προβλήματα, ειδικά στα ανώτερα επίπεδά της και αυτό λόγω των πολλών αιτήσεων που δέχονται από τους εκατοντάδες ή και χιλιάδες πελάτες που βρίσκονται από κάτω.
- Όποτε ένας proxy προωθεί μια αίτηση προς έναν πραγματικό διακομιστή καταγράφει τη σύνδεση από την IP του proxy. Όταν ένας παροχέας υπηρεσιών ή ο ιδιοκτήτης της σελίδας πιστεύει ότι γίνεται κάποια κατάχρηση ή κακομεταχείριση επικοινωνεί με τον υπεύθυνο της IP από όπου έρχονται οι αιτήσεις (proxy) για τα παράπονα.

- Πολλές φορές δεν επιστρέφουν σωστά ή έγκυρα μηνύματα σφαλμάτων. Ένα παράδειγμα είναι ότι ένας proxy δεν μπορεί να καταλάβει τη διαφορά μεταξύ ενός ονόματος DNS που πραγματικά δεν υπάρχει ή απλά δεν δουλεύει η υπηρεσία προσωρινά.
- Μεταξύ μιας σχέσης αδερφών-cache υπάρχει ένας μηχανισμός πρόβλεψης επιτυχίας (cache hit) η οποία γίνεται από τα πρωτόκολλα cache. Στην περίπτωση που η πρόβλεψη δεν είναι σωστή, δηλαδή όταν μια αίτηση προβλέπεται να είναι επιτυχία στην cache του αδερφού αλλά τελικά δεν είναι, το αποκαλούμε *λανθασμένη επιτυχία (false hit).*
- Βρόχος προώθησης. Ένας βρόχος προώθησης παρουσιάζεται όταν μια αίτηση στέλνεται πάνω – κάτω μεταξύ δύο ή παραπάνω κόμβων της ιεραρχίας. Αυτό μπορεί να συμβεί μεταξύ δυο cache όταν στον καθένα είναι δηλωμένος ο άλλος ως πατέρας-cache.
- Βλάβες και άρνηση υπηρεσίας (Service Denial). Υπάρχουν κάποια προβλήματα, που είναι πιο δύσκολα να εντοπιστούν, από ότι μια ολοκληρωτική, μηχανική βλάβη σε έναν πατέρα-cache όπως είναι η υπερφόρτωση με κίνηση του πατέρα , το οποίο έχει σαν αποτέλεσμα την αύξηση του χρόνου των αποκρίσεων. Μια πιθανή αιτία άρνησης υπηρεσίας είναι κάποιο πρόβλημα με τον διακομιστή DNS του πατέρα-cache. Αυτές οι ενδείξεις βέβαια δεν είναι σίγουρο ότι οφείλονται σε κάποιο σφάλμα.

4.5 InterCache Protocols

Αυτά τα πρωτόκολλα χρησιμοποιούνται μεταξύ συνεργαζόμενους cache-proxy για πολλούς λόγους, ο βασικότερος εκ των οποίων είναι να βοηθούν σε αποφάσεις προώθησης, δηλαδή δεδομένου κάποιων στοιχείων, προς ποια κατεύθυνση να στείλει την αίτηση;

4.5.1 ICP

Είναι το αυθεντικό πρωτόκολλο intercache. Πρωταρχικός του σκοπός είναι να μαθαίνει εάν κάποια γειτονική cache έχει πιο φρέσκο αντίγραφο ενός συγκεκριμένου αντικειμένου. Οι γείτονες-cache απαντούν είτε με ένα ναι (HIT),

είτε με ένα όχι (MISS). Η cache συλλέγει ένα συγκεκριμένο αριθμό από απαντήσεις ICP και παίρνει μια απόφαση προώθησης. Ακόμα και αν όλοι οι γείτονες απαντήσουν με MISS, το ICP μπορεί να παρέχει πρόσθετες υποδείξεις που βοηθούν στο να διαλέξει τον καλύτερο πατέρα-cache.

Το ICP είναι πέραν του τέλειου, όμως χρησιμοποιείται ακόμα ευρέως.

4.5.2 CARP (Cache Array Routing Protocol)

Το CARP δεν είναι ένα πρωτόκολλο αυτό καθεαυτό. Σχεδιάστηκε για να επιλύσει ένα πού συγκεκριμένο πρόβλημα. Το πώς να επιτύχουμε αποδοτικά και κλιμακωτά την εξισορρόπηση του φόρτου ενώ αυξάνουμε τα ποσοστά επιτυχιών (hit ratios) και μειώνουμε το χρόνο αδράνειας. Είναι πολύ χρήσιμο σε περιπτώσεις που η cache μας αποτελείται από πολλά μηχανήματα (cache *Cluster*).

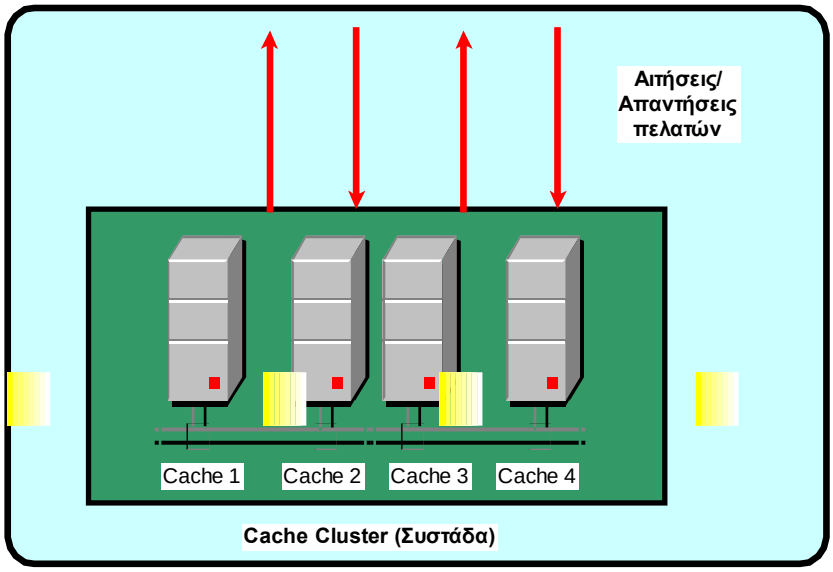
Για κάθε αίτηση, το CARP υπολογίζει ένα βαθμό για κάθε proxy cache. Η αίτηση προωθείται στον proxy με το μεγαλύτερο βαθμό. Εάν αυτό αποτύχει, τότε δοκιμάζεται η cache με το δεύτερο μεγαλύτερο βαθμό. Ο βαθμός είναι ένας υπολογισμός βασιζόμενος σ' ένα hash του URL, ένα hash του ονόματος της cache, και τα ειδικό βάρος που έχει ανατεθεί στην κάθε cache. Το σημαντικότερο χαρακτηριστικό αυτής της διαδικασίας είναι ότι εάν προστεθεί άλλη μια μηχανή για την cache δεν αλλάζει την ιεραρχία των βαθμών των υπολοίπων cache, αλλά δημιουργεί νέες βαθμολογίες. Στατιστικά, οι νέοι βαθμοί θα είναι μεγαλύτεροι από τους προηγούμενους για το μέρος των URL που είναι ανάλογο στο βάρος της cache στο cluster.

Το CARP καθορίζει και έναν τύπο αρχείων για ένα *Proxy Array Membership Table.* Έναν πίνακα δηλαδή, ο οποίος επιτρέπει σε πελάτες να διαπιστώσουν ποιες cache ανήκουν σε μια ομάδα, σε ένα *group.* Ο αλγόριθμος του CARP μπορεί να χρησιμοποιηθεί σε οποιοδήποτε πελάτη ιστού, όπως είναι ένας φυλλομετρητής ή έναν proxy cache. Το CARP δουλεύει μόνο για σχέσεις γονέων- παιδιών γιατί προβλέπει cache hits.

4.6 Cache Cluster (Συστάδα Cache)

Ένα *Cache cluster* είναι μια ομάδα από ξεχωριστούς proxy cache που είναι ρυθμισμένοι να ενεργούν σαν να ήταν ένας διακομιστής. Δηλαδή, οι χρήστες και οι πελάτες τους αντιλαμβάνονται ως μια μονάδα.

Μια συστάδα διαφέρει από την ιεραρχία σε κάποιες λεπτομέρειες. Αρχικά, τα μέλη της συστάδας βρίσκονται το ένα κοντά στο άλλο, φυσικά και τοπολογικά, δηλαδή βρίσκονται στο ίδιο δωμάτιο και ανήκουν στο ίδιο υποδίκτυο. Πολλοί οργανισμοί χρησιμοποιούν cache clusters για να εξυπηρετούν ή να παρέχουν περισσότερες σελίδες και περίσσιες υπηρεσίες. Αν σε έναν οργανισμό υπάρχει ένας proxy, αλλά η κίνηση αυξάνεται και επιβαρύνονται, με αποτέλεσμα να γίνονται αργές, οι υπηρεσίες τότε μια καλή λύση, χωρίς να χαθεί το υπάρχον προϊόν cache και τα περιεχόμενα της, είναι να πάρει άλλη μια μηχανή και να φτιάξει ένα μικρό cluster.



Cache Cluster 4.6.1

Βέβαια, υπάρχουν και άλλοι λόγοι για να χρησιμοποιήσει ένας οργανισμός συστάδες. Αναφέρω τρεις:

4.6.1 Η “Ρεζέρβα”

Ένας τρόπος να παρέχουμε πλεονάζουσες υπηρεσίες είναι να έχουμε σε αναμονή μια δεύτερη cache. Σε κανονική λειτουργία, όλες οι αιτήσεις πηγαίνουν στην πρωταρχική cache. Αν αυτή αποτύχει αναλαμβάνει η δεύτερη.

Αυτή η διαρρύθμιση δεν είναι ακριβώς μια συστάδα, μιας και μόνο μία εκ των δύο cache λειτουργεί σε μια χρονική στιγμή. Οι τεχνικές όμως είναι παρόμοιες.

Κάποια γνωστά προϊόντα (Switch Επιπέδου 4 και 7) μπορούν να ρυθμιστούν να δουλεύουν ως transparent proxy ή με μια εικονική διεύθυνση διακομιστή (virtual server). Μπορούμε να πούμε στο switch τις πραγματικές IP διευθύνσεις για την πρωταρχική cache και την cache – ρεζέρβα. Κανονικά προωθεί όλες τις συνδέσεις στην πρωταρχική. Αν το switch διαπιστώσει ότι η πρωταρχική έχει πέσει, τότε χρησιμοποιεί τη ρεζέρβα. Εφόσον οι χρήστες μιλούν στον εικονικό διακομιστή, δεν υπάρχουν προβλήματα με DNS και ARP timeouts.

4.6.2 Ταχύτητα διεκπεραίωσης και Κατανομή Φόρτου

Ένα cache cluster με κατανομή φόρτου (load sharing) μπορεί να βελτιώσει την ταχύτητα διεκπεραίωσης και την αξιοπιστία. Η ταχύτητα διεκπεραίωσης αυξάνεται διότι πολλές cache μπορούν να χειριστούν περισσότερη κίνηση από ότι μία. Η αξιοπιστία αυξάνεται γιατί όταν παρουσιαστεί κάποιο πρόβλημα σε μία cache, οι άλλες απορροφούν τον αυξημένο φόρτο.

Ο πιο φτηνός τρόπος για να πετύχουμε κατανομή του φόρτου είναι με το DNS Server, να δηλώσουμε το ίδιο όνομα host σε όλα τα μέλη της συστάδας, δηλαδή η μέθοδος *round-robin,* κατά την οποία ο DNS server θα ανακυκλώνει τις IP των μελών και θα δίνει άλλη IP σε κάθε lookup.

Μια πιο ρωμαλέα προσέγγιση, αν και πιο ακριβή, είναι η χρήση ενός switch επιπέδου 4 ή κάποια γνωστά προϊόντα για εξισορρόπηση φόρτου. Με αυτή την προσέγγιση η κατανομή του φόρτου γίνεται αρκετά πιο ισορροπημένα από ότι στην προσέγγιση round-robin και δεν έχουμε μεγάλες καθυστερήσεις.

Σε πολλές περιπτώσεις όπου υπάρχει cache cluster και στον πραγματικό διακομιστή (ιστού) με ένα switch επιπέδου 4 μπροστά από αυτό, το οποίο διανέμει τις αιτήσεις με βάση τη διεύθυνση IP. Όταν η επικοινωνία όμως περιέχει πληροφορίες συνεδρίας (session information, π.χ. χρήση “cookies”), και κάποια ακόλουθα πακέτα πάνε σε κάποιον άλλον διακομιστή του cluster, απορρίπτονται. Αυτό λύνεται με τη χρήση των switch επιπέδου 7 τα οποία καταλαβαίνουν πληροφορίες όπως τα cookies.

4.6.3 Εύρος Ζώνης (Bandwidth)

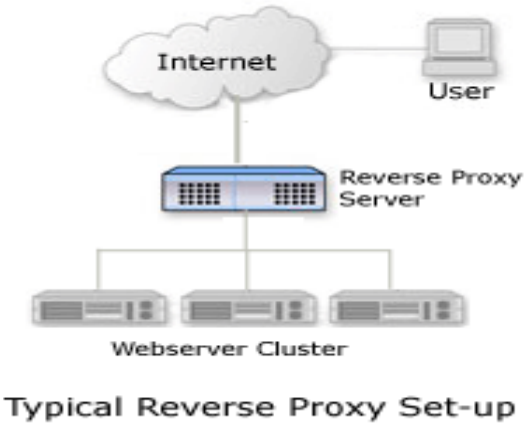
Μια απλή διαμόρφωση για κατανομή φόρτου αποτελεί χαμένο χώρο στο αποθηκευτικό μέσο και σπατάλη του εύρους ζώνης. Λέμε χαμένος χώρος γιατί η ίδια απάντηση μπορεί να αποθηκευτεί σε πολλές cache και λέμε σπατάλη του εύρους ζώνης γιατί δεν χρειάζεται πάντα να προωθούμε ένα “cache miss” προς τον πραγματικό διακομιστή εάν γνωρίζουμε ότι ένα άλλο μέλος της συστάδας έχει ήδη την απάντηση αποθηκευμένη.

Υπάρχουν δύο τρόποι να βελτιώσουμε την χρήση των δίσκων και του εύρους ζώνης. Ο ένας είναι να διανεμηθούν οι αιτήσεις πριν μπουν στη συστάδα, δηλαδή κάποια συσκευή ή κάποιος αλγόριθμος φροντίζει ώστε η ίδια αίτηση να πηγαίνει προς το ίδιο μέλος της συστάδας. Ο δεύτερος τρόπος είναι να δημιουργήσουμε “αδερφικές σχέσεις” μεταξύ των cache-μελών της συστάδας και να χρησιμοποιήσουμε ένα πρωτόκολλο intercache να εντοπίζουμε απαντήσεις που βρίσκονται ήδη στην cache μας. Σε αυτή την περίπτωση δεν μας ενδιαφέρει ποια cache παρέλαβε την αρχική αίτηση.

Υπάρχουν αρκετές τεχνικές και προϊόντα που διανέμουν αιτήσεις και τις αναθέτουν σε συγκεκριμένα μέλη της συστάδας. Κάποια από αυτά είναι το WCCP, τα switch επιπέδου 4 και επιπέδου 7 και το πρωτόκολλο CARP.

4.7 Reverse Proxy

Το *reverse proxy cache*, γνωστό και ως *Web Server Acceleration* (επιτάχυνση διακομιστή ιστού), είναι ένας τρόπος να μειώνουμε το φόρτο ενός αρκετά φορτωμένου web server, βάζοντας ανάμεσα σε αυτόν και το Διαδίκτυο έναν proxy cache, το οποίο προσθέτει και επιπλέον ασφάλεια. Με σωστή χρήση του reverse proxy διευκολύνεται πολύ η δουλειά ενός web server ο οποίος παράγει στατικά και δυναμικά αντικείμενα. Τα στατικά μπορούν να αποθηκευτούν στην cache του reverse proxy, ενώ ο web server θα είναι πιο ελεύθερος να παράγει το δυναμικό περιεχόμενο.



Reverse proxying ή web acceleration 4.7.1

Εφαρμόζοντας έναν reverse proxy παράλληλα με κάποιους web servers, το site μας μπορεί:

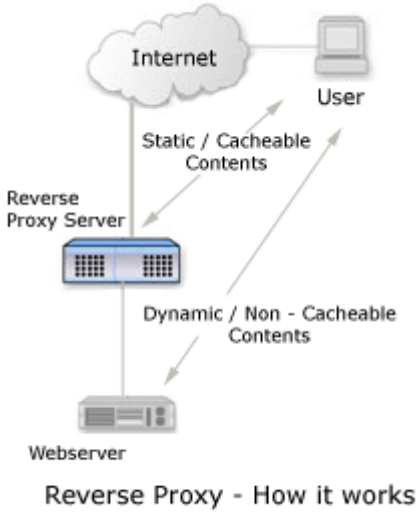
- Να αποφύγει περιττά έξοδα για την αγορά πρόσθετων web server, αυξάνοντας τις ικανότητες του υπάρχοντος.
- Θα εξυπηρετούν περισσότερες αιτήσεις για στατικό υλικό από τον web server.
- Θα εξυπηρετούν περισσότερες αιτήσεις για δυναμικό υλικό από τον web server
- Να αυξήσει το κέρδος της επιχείρησης, μειώνοντας τα λειτουργικά έξοδα συμπεριλαμβανομένου και τα έξοδα που απαιτούνται για το εύρος ζώνης που χρειάζεται.
- Επιτάχυνση του χρόνου απόκρισης των σελίδων και επιτάχυνση των download για τους εξωτερικούς χρήστες, μεταφέροντάς τους μια γρηγορότερη και καλύτερη εμπειρία της σελίδας και των υπηρεσιών μας.

Εάν η ιστοσελίδα μας δεν έχει γραφτεί με τρόπο να δουλεύει με κάποιο proxy, δε θα μπορεί να εκμεταλλευτεί όλες τις δυνατότητες ενός reverse proxy.

Σε κατάσταση reverse proxy, ο διακομιστής proxy συμπεριφέρεται κατά κύριο λόγο, σαν διακομιστής ιστού. Ενώ οι εσωτερικοί χρήστες χρειάζονται κάποιες ρυθμίσεις για να μπορούν να επικοινωνούν με τον proxy, οι εξωτερικοί δεν χρειάζονται καμία απολύτως. Το URL του site μας δρομολογεί τον πελάτη

στον proxy σα να ήταν αυτός ο web server. Το αντιγραμμένο περιεχόμενο παραδίδεται από τον proxy cache στον εξωτερικό πελάτη χωρίς να εκτίθεται ο πραγματικός διακομιστής ή το ιδιωτικό μας δίκτυο που βρίσκονται πίσω από το firewall. Πολλαπλοί reverse proxy μπορούν να χρησιμοποιηθούν για την εξισορρόπηση του φόρτου (cache cluster).

Ένας reverse proxy cache διαφέρει από ένα συνηθισμένο ή έναν διαφανή proxy στο ότι μειώνει το φόρτο στον web server αντί να μειώνει το, προς τα έξω, εύρος ζώνης από την πλευρά των πελατών. Απαλλάσσουν τον web server από αιτήσεις πελατών για στατικό περιεχόμενο, αποτρέποντας έτσι την υπερφόρτωση του πραγματικού διακομιστή από απρόβλεπτες, απότομες αυξήσεις κίνησης. Ο proxy βρίσκεται ανάμεσα στο Διαδίκτυο και το site μας και χειρίζεται την κίνηση πριν φτάσει στον διακομιστή ιστού και αναχαιτίζει τις αιτήσεις προς τον διακομιστή ιστού και απαντά αντί γι' αυτόν από την εναποθήκευση που έχει κάνει στην cache του με προηγούμενες απαντήσεις. Αυτή η μέθοδος βελτιώνει την απόδοση μειώνοντας το ποσό των ιστοσελίδων που αναπαράγονται από τον web server.



Εικόνα 4.7.2

Όταν ένας πελάτης-φυλλομετρητής δημιουργεί μια αίτηση HTTP, ο DNS θα δρομολογήσει την αίτηση προς τον proxy (εικόνα 4.7.2). ο proxy ελέγχει την cache του να δει αν περιέχει το ζητούμενο αντικείμενο. Εάν δεν το έχει, συνδέεται με τον web server και το κατεβάζει στην cache του. Ένας reverse proxy μπορεί να ικανοποιεί αιτήσεις για URL's τα οποία μπορεί να αποθηκεύσει στην cache του, όπως είναι οι σελίδες html και εικόνες.

Δυναμικό περιεχόμενο, όπως είναι τα cgi scripts, ASP, PHP δεν μπορούν να αποθηκευτούν στην cache. Ο proxy μπορεί να αποθηκεύσει στατικές σελίδες βασιζόμενος στις ετικέτες των κεφαλίδων HTTP (header tags) που επιστρέφει η ιστοσελίδα. Οι τέσσερις πιο σημαντικές ετικέτες είναι οι:

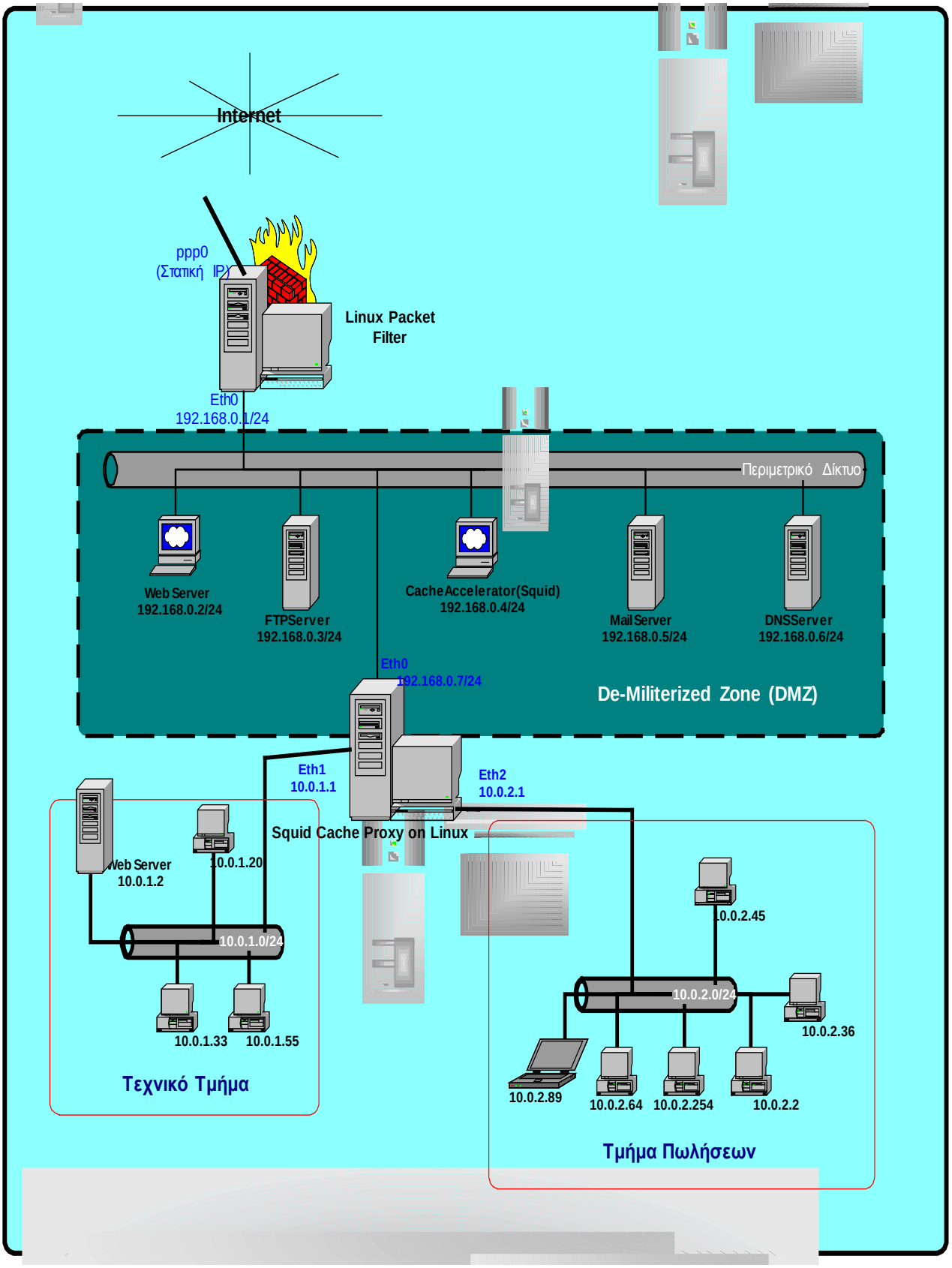
- Last-Modified. Πληροφορεί τον proxy για την τελευταία τροποποίηση της σελίδας.
- Expires. Πληροφορεί τον proxy για τον χρόνο που θα πρέπει να σβήσει την σελίδα από την cache του.
- Cache-Control. Πληροφορεί για το εάν πρέπει να αποθηκευτεί στην cache ή όχι.
- Pragma. Παρόμοιο με το Cache-Control.

5 Case Study

Στο case study θα αναφέρουμε τη δομή της ασφάλειας ενός δικτύου μιας επιχείρησης η οποία αποτελείται από δύο βασικά τμήματα. Το τμήμα πωλήσεων, όπου ασχολείται με την προώθηση και την αγορά προϊόντων για την επιχείρηση και το τεχνικό τμήμα, το οποίο ασχολείται με τεχνικά θάματα της ίδιας της επιχείρησης και τεχνικά θέματα που αφορούν πελάτες της.

Θα χρησιμοποιηθεί η αρχιτεκτονική της *αποστρατικοποιημένης ζώνης* (De-Militarized Zone, DMZ). Η επιχείρηση έχει στη διάθεσή της ένα C class δίκτυο. Θα υποθέσουμε ότι οι διευθύνσεις 192.168.0.0/ 255.255.255.0 είναι πραγματικές διευθύνσεις δημοσιευμένες στο Διαδίκτυο και όχι ιδιωτικού δικτύου. Για ιδιωτικού δικτύου διευθύνσεις θα χρησιμοποιήσουμε τις διευθύνσεις 10.0.1.0/24 και 10.0.2.0/24. Χρησιμοποιούμε για στάση αποτυχίας την στάση default deny.

Για εξωτερικό δρομολογητή χρησιμοποιούμε έναν Η/Υ με Linux (kernel 2.2.19) για λειτουργικό σύστημα ο οποίος έχει μια κάρτα δικτύου (eth0) και μια μόνιμη σύνδεση point-to-point. Αυτός είναι και ο δρομολογητής μας. Μέσα στο περιμετρικό δίκτυο, το οποίο χρησιμοποιεί πραγματικές διευθύνσεις, έχουμε:



Case Study 5.1

- Έναν mail server για την επιχείρησή μας με διεύθυνση 192.168.0.5
- Έναν δημόσιο FTP Server με διεύθυνση 192.168.0.3
- Έναν δημόσιο DNS Server με διεύθυνση 192.168.0.6
- Έναν δημόσιο Web Server με διεύθυνση 192.168.0.2, ο οποίος δημοσιεύεται μέσω ενός cache-proxy accelerator.
- Ένας cache proxy accelerator με διεύθυνση 192.168.0.4 ο οποίος εξυπηρετεί τον Web Server μας.
- Τέλος, ένας ιδιωτικός proxy-cache server με διεύθυνση 192.168.0.7, σε Η/Υ με Linux (kernel 2.2.19) και Squid 2.3 STABLE 5 ο οποίος εξυπηρετεί τα ιδιωτικά μας δίκτυα.

Στον εξωτερικό μας firewall χρησιμοποιούμε ipchains. Κάνουμε τον έλεγχο των πακέτων στην αλυσίδα “input”. Τα ipchains έχουν τρεις αλυσίδες που δεν μπορούν να σβηστούν, την αλυσίδα input, την αλυσίδα output, την αλυσίδα forward. Πάνω σε αυτές βασίζονται οι αλυσίδες των χρηστών. Όλες οι αλυσίδες έχουν και μια πολιτική ασφαλείας. Εξ’ ορισμού οι τρεις προαναφερθέντες έχουν default ACCEPT. Εδώ θα χρησιμοποιήσουμε και αλυσίδες χρηστών.

1. ipchains –N exo-mesa

*2.*ipchains –N mesa-exo

*3.*ipchains –A input –i !lo –s 127.0.0.0/8 –l –j DENY

*4.*ipcains –A input –i lo –j ACCEPT

*5.*ipchains –A input –i ppp0 –j exo-mesa

*6.*ipchains –A input –i eth0 –j mesa-exo

*7.*ipchains –A exo-mesa –s 192.168.0.0/24 –l –j DENY

*8.*ipchains –A exo-mesa –p icmp –d 192.168.0.7 --icmp-type echo-request –j DENY

*9.*ipchains –A exo-mesa –p icmp –d 192.168.0.2 --icmp-type echo-request –j DENY

10. ipchains –A exo-mesa –p tcp –d 192.168.0.5 smtp –j ACCEPT

11. ipchains –A exo-mesa –p udp –d 192.168.0.6 domain –j ACCEPT

12. ipchains –A exo-mesa –p tcp –d 192.168.0.6 domain –j ACCEPT

13. ipchains –A exo-mesa –p tcp –d 192.168.0.4 www –j ACCEPT

14. ipchains –A exo-mesa –p tcp –d 192.168.0.3 ftp –j ACCEPT

15. ipchains –A exo-mesa –p tcp –d 192.168.0.3 ftp-data –j ACCEPT

16. ipchains –A exo-mesa –p tcp ! –y –d 192.168.0.7 1024:65535 –j ACCEPT

17. ipchains –A exo-mesa –p icmp –j ACCEPT

18. ipchains –A exo-mesa -l –j DENY

CHAIN MESA-EXO

1. ipchains –A mesa-exo –s ! 192.168.0.0/24 –l –j DENY

*2.*ipchains –A mesa-exo –p tcp –s 192.168.0.5 smtp –j ACCEPT

*3.*ipchains –A mesa-exo –p tcp –s 192.168.0.6 domain –j ACCEPT

*4.*ipchains –A mesa-exo –p udp –s 192.168.0.6 domain –j ACCEPT

*5.*ipchains –A mesa-exo –p tcp ! –y –s 192.168.0.4 www –j ACCEPT

*6.*ipchains –A mesa-exo –p tcp ! –y –s 192.168.0.3 ftp –j ACCEPT

*7.*ipchains –A mesa-exo –p tcp ! –y –s 192.168.0.3 ftp-data –j ACCEPT

*8.*ipchains –A mesa-exo –p tcp -s 192.168.0.7 1024:65535 –j ACCEPT

*9.*ipchains –A mesa-exo –p icmp –j ACCEPT

10. ipchains –A mesa-exo –l –j REJECT

Στους κανόνες 1 και 2 δημιουργούμε τις αλυσίδες mesa-exo και exo-mesa.

Στους κανόνες 3 και 4 γίνεται έλεγχος για IP spoofing της loopback στην αλυσίδα input. Αν διαπιστωθεί τέτοια περίπτωση, τότε το πακέτο απορρίπτεται. Αν όχι, επιτρέπεται.

Κανόνες 5,6. Αν εισέλθει πακέτο από το interface της εξωτερικής σύνδεσης, τότε να ελεγχθεί στην αλυσίδα exo-mesa που δημιουργήσαμε. Αντίστοιχα, αν εισέλθει πακέτο από το interface της κάρτας δικτύου (eth0) να ελεγχθεί στην αλυσίδα mesa-exo που δημιουργήσαμε.

Από τον κανόνα 7 ξεκινούν οι κανόνες για την αλυσίδα χρήστη exo-mesa.

Κανόνας 7. Έλεγχος για IP Spoofing. Αν το πακέτο που εισήλθε από το interface ppp0 έχει διεύθυνση πηγής κα΄ποια διεύθυνση του δικτύου μας, τότε απορρίπτεται.

Κανόνας 8 και 9. Έλεγχος αν το πακέτο είναι τύπου icmp echo-request, δηλαδή αν κάποιος από έξω κάνει ping στον cache server (192.168.0.7) και στην πραγματική διεύθυνση του web server μας (192.168.0.2) και αν ναι, τότε απορρίπτεται (και κατ' επέκταση δεν δίνεται απάντηση από αυτούς)

Στους κανόνες 10,12,13,14 και 15 ελέγχεται το πακέτο με βάση το πρωτόκολλο (εδώ TCP), τη διεύθυνση προορισμού και το port προορισμού. Εάν

κάποιο αντιστοιχεί στον web server, ftp server, dns server, mail server και στο αντίστοιχο port, τότε το πακέτο περνά.

Στον κανόνα 11 γίνεται η ίδια διαδικασία για τον dns server μας, μόνο που εδώ το πρωτόκολλο που ελέγχεται είναι το udp. Εάν ταιριάξει ο κανόνας, τότε και πάλι το πακέτο περνά.

Στον κανόνα 16 ελέγχεται το πακέτο, αν είναι tcp το πρωτόκολλο, αν δεν είναι πακέτο που ζητά να ανοίξει σύνδεση tcp και πηγαίνει στον cache-proxy στην διεύθυνση 192.168.0.7 σε port μεγαλύτερα του 1023. αυτό σημαίνει ότι το πακέτο θα είναι κάποια απάντηση σε κάποια αίτηση του cache-proxy μας. Αν ταιριάξει, τότε περνά.

Στον κανόνα 17 ελέγχεται το πρωτόκολλο του πακέτου εάν είναι icmp και αν είναι περνά.

Στον κανόνα 18 απορρίπτονται και καταγράφονται στα log files όλα τα υπόλοιπα πακέτα.

Εδώ τελειώνουν οι κανόνες της αλυσίδας exo-mesa και αρχίζουν οι κανόνες της άλλης αλυσίδας χρήστη, η mesa-exo.

Στον κανόνα 1 γίνεται έλεγχος για IP Spoofing στα πακέτα που εισέρχονται από την κάρτα δικτύου μας. Αν το πακέτο δεν έχει διεύθυνση πηγής κάποια διεύθυνση του εσωτερικού μας δικτύου, τότε καταγράφεται και απορρίπτεται.

Στον κανόνα 2, 3 γίνεται έλεγχος για το αν το πρωτόκολλο είναι tcp και αν το πακέτο κατευθύνεται προς τον Mail server ή τον DNS server στα αντίστοιχα port του καθενός και αν ταιριάξει, τότε το πακέτο περνά.

Στον κανόνα 4 γίνεται ότι και στον κανόνα 2, μόνο που ελέγχεται αν το πρωτόκολλο είναι udp.

Στους κανόνες 5,6,7 γίνεται έλεγχος για το αν το πρωτόκολλο είναι tcp, αν είναι ήδη ανοιχτεί tcp σύνδεση (δεν είναι πακέτο που ζητά να γίνει σύνδεση), και προορίζεται για τον cache accelerator, ή τον ftp server στα αντίστοιχα port και αν ταιριάξει τον κανόνα το πακέτο περνά.

Στον κανόνα 8 γίνεται έλεγχος του πακέτου με βάση το πρωτόκολλο (tcp), και η πηγή του πακέτου, αν είναι ο cache-proxy μας και αν το port είναι μεγαλύτερο του 1023.

Ο κανόνας 9 επιτρέπει τη διέλευση του πακέτου αν το πρωτόκολλο είναι icmp.

Στον τελευταίο κανόνα απορρίπτονται και καταγράφονται όλα τα άλλα πακέτα. Η απόρριψη REJECT στέλνει απάντηση icmp ότι για κάποιο λόγο δεν έφτασε το πακέτο ώστε να μην συνεχιστεί η προσπάθεια μέχρι το timeout.

Cache Proxy. Ο cache-proxy είναι εγκατεστημένος σε Η/Υ με λειτουργικό Linux (kernel 2.2.19) ο οποίος δεν προωθεί τα πακέτα παρά σε μια περίπτωση, όπως θα δούμε και πιο κάτω. Έχει τρεις κάρτες δικτύου, τις eth0, eth1, eth2. Η eth0 αντιστοιχεί στην πραγματική διεύθυνση 192.168.0.7 του περιμετρικού δικτύου μας. Οι άλλες δύο αντιστοιχούν στα δύο εσωτερικά, ιδιωτικά δίκτυα του τεχνικού τμήματος (με IP 10.0.1.1/24) και του τμήματος πωλήσεων (με IP 10.0.2.1/24). Έχουμε εγκαταστήσει και τον Squid Cache Proxy 2.3 STABLE 5.

Χρησιμοποιούμε ipchains για την προστασία των ιδιωτικών δικτύων από το περιμετρικό και γενικά τα εξωτερικά. Επειδή ο proxy μας θα εξυπηρετεί τις αιτήσεις από μέσα προς διακομιστές www, ftp θα πρέπει να χρησιμοποιήσουμε NAT ή Masquerading για τις υπόλοιπες υπηρεσίες που θέλουμε να παρέχουμε, συγκεκριμένα, ηλεκτρονική αλληλογραφία. Οπότε οι κανόνες ορίζονται ως εξής:

1) ipchains –A input –i eth0 –s 10.0.0.0/16 –j DENY –l
2) ipchains –A input –i eth1 –s ! 10.0.1.0/24 –j DENY –l
3) ipchains –A input –i eth2 –s ! 10.0.2.0/24 –j DENY –l

Σε αυτούς τους τρεις κανόνες γίνεται η αντιμετώπιση του IP Spoofing στα τρία interface.

4) ipchains –A input –i eth0 –p tcp ! –y -s 192.168.0.6 domain –j ACCEPT
5) ipchains –A input –i eth0 –p udp -s 192.168.0.6 domain –j ACCEPT
6) ipchains –A input –i eth0 –p tcp ! –y --dport 1024:65535 –j ACCEPT
7) ipchains –A input –i eth0 –p icmp –s 192.168.0.0/24 –j ACCEPT
8) ipchains –A input –i eth0 –j DENY –l

Εδώ γίνονται έλεγχος του eth0 στην αλυσίδα εισόδου (input). Ο κανόνας 5 είναι λίγο ρίσκο γιατί δεν μπορούμε να ελέγξουμε αν η σύνδεση έχει γίνει λόγω του πρωτοκόλλου udp όμως πολλές φορές απαιτείται για τα DNS lookups. Οι κανόνες 4 και 6 ελέγχουν την σύνδεση, αν έχει γίνει δηλαδή, για τον κανόνα 4 επιτρέπει το πακέτο αν έρχεται από τον DNS server μας. Ο κανόνας 6 επιτρέπει το πακέτο αν το port προορισμού είναι μεγαλύτερο του 1024 και έχει το γίνει σύνδεση. Ο καν.7

επιτρέπει όλα τα ICMP από το DMZ. Ο καν. 8 απορρίπτει όλα τα άλλα πακέτα που εισέρχονται από το eth0.

9) ipchains –A input –i eth1 –p icmp –j ACCEPT
10) ipchains –A input –i eth1 –p tcp –d 192.168.0.5 smtp –j ACCEPT
11) ipchains –A input –i eth1 –p tcp –d 192.168.0.5 pop3 –j ACCEPT
12) ipchains –A input –i eth1 –p tcp --dport 3129 -j ACCEPT
13) ipchains -A input -i eth1 -j REJECT -l

Εδώ γίνεται έλεγχος του eth1 στην αλυσίδα input- εισόδου. Επιτρέπουμε τα icmp πακέτα και αιτήσεις για αποστολή και λήψη αλληλογραφίας και αιτήσεις tcp προς το port του proxy. Όποιο άλλο πακέτο εισέλθει στο eth1 απορρίπτεται και σε στέλνεται ενημερωτικό icmp μήνυμα, καθώς και καταγράφεται.

14) ipchains –A input –i eth2 –p icmp –j ACCEPT
15) ipchains –A input –i eth2 –p tcp –d 192.168.0.5 smtp –j ACCEPT
16) ipchains –A input –i eth2 –p tcp –d 192.168.0.5 pop3 –j ACCEPT
17) ipchains –A input –i eth2 –p tcp --dport 3129 -j ACCEPT
18) ipchains -A input -i eth2 -j REJECT -l

Εδώ γίνεται ότι ακριβώς έγινε για το eth1 στους κανόνες 9-13, αλλά γίνεται για το eth2 τώρα.

Επειδή τα πακέτα που περνούν από τα εσωτερικά δίκτυα προς το εξωτερικό-περιμετρικό μας δίκτυο δεν έχουν πραγματικές διευθύνσεις πρέπει να υποστούν “Masquerading” . Αυτό γίνεται στην αλυσίδα forward ως εξής:

19) ipchains -A forward -p tcp -d 192.168.0.5 -j MASQ
20) ipchains -A forward -p tcp -s 192.168.0.5 -j MASQ

Δηλαδή οποιοδήποτε πακέτο φτάσει σε αυτή την αλυσίδα και έχει πηγή ή προορισμό τον mail server μας περνά και υφίσταται Masquerading.

21) ipchains -A output -p tcp -d 192.168.0.5 smtp -j ACCEPT
22) ipchains -A output -p tcp -d 192.168.0.5 pop3 -j ACCEPT
23) ipchains -A output -p tcp -s 192.168.0.5 smtp -j ACCEPT
24) ipchains -A output -p tcp -s 192.168.0.5 pop3 -j ACCEPT
25) ipchains -A output -i eth0 -p tcp -s 192.168.0.6 domain -j ACCEPT
26) ipchains -A output -i eth0 -p tcp --sport 1024:65535 -j ACCEPT
27) ipchains -A output -i eth0 -p icmp -j ACCEPT
28) ipchains -A output -i eth0 -l -j REJECT

Στους κανόνες 21-24 επιτρέπουμε την έξοδο από οποιοδήποτε interface πακέτων που προορίζονται ή προέρχονται από τον Mail server μας. Στον κανόνα 25 επιτρέπουμε να εξέλθουν από το eth0 πακέτα προς τον DNS Server μας. Στον κανόνα 26 επιτρέπουμε όλα τα πακέτα από το eth0 να βγουν αν έχουν port πηγής μεγαλύτερο του 1024. Στον κανόνα 27 αφήνουμε να εξέλθουν από το eth0 όλα τα πακέτα icmp και τέλος δεν επιτρέπουμε και καταγράφουμε οποιοδήποτε άλλο πακέτο αποπειραθεί να βγει από το eth0.

Squid Cache Proxy. Ο proxy μας τρέχει στο port 3129. Πρόσβαση στην cache έχουν χρήστες από τα δύο ιδιωτικά μας δίκτυα καθώς και ο localhost, ο ίδιος ο Η/Υ όπου τρέχει ο squid cache proxy.

Στον Squid φτιάχνουμε τα Access Control Lists (ACL's) και τα ενεργοποιούμε με τα acl operator, δηλαδή τα : http_access για http αιτήσεις και τα icp_access τα οποία είναι για επικοινωνία της cache μας με άλλες cache, κάτι το οποίο δεν γίνεται εδώ.

1) acl texniko src 10.0.1.0/255.255.255.0
2) acl poliseis src 10.0.2.0/24
3) acl private_nets src 10.0.0.0/16

   # Δήλωση των ιδιωτικών μας δικτύων. Πρώτα δηλώσαμε το τεχνικό τμήμα, #έπειτα το τμήμα πωλήσεων και, τέλος, όλα τα πιθανά υποδίκτυα με Subnet #Mask 255.255.0.0.

4) acl bad_domains dstdomain hack.com intruder.com #δήλωση των domain που θεωρούμε μη έμπιστα. Αυτά είναι το hack.com και το intruder.com.
5) acl weekends time SA # δήλωση των ημερών Κυριακή και Σάββατο
6) acl badsites url_regex -i sex xxx porn # δήλωση των url που περιέχουν τις #λέξεις xxx, sex και porn. Case insensitive (-i).
7) acl downloads url_regex -i \.avi$ \.mpeg$ \.mpg$ #δήλωση των url τα οποία #τελειώνουν με .mpeg , .mpg , .avi ώστε να επιτραπούν ή να αποτραπούν τα download τέτοιων αρχείων.
8) acl all src 0.0.0.0/0.0.0.0 #δήλωση όλων των διευθύνσεων IP
9) acl manager proto cache_object #δήλωση αντικειμένων cache
10) acl localhost src 127.0.0.1/255.255.255.255 #δήλωση του localhost
11) acl SSL_ports port 443 563 #δήλωση των γνωστών port για το SSL

12) acl Safe_ports 80 21 443 563 70 210 1024-65535 #δήλωση των port που #φαίνονται.
13) acl Connect method CONNECT #δήλωση της μεθόδου CONNECT που #χρησιμοποιείται σε συνδέσεις SSL αντί για το GET.

Εφαρμογή των δηλώσεων- μετατροπή τους σε κανόνες

Οι κανόνες πρέπει να μπουν με μια σειρά. Ελέγχονται ο ένας μετά τον άλλον. Αν ταιριάξει κάποιος γίνεται η αποδεκτή η αίτηση ή απορρίπτεται. Αυτό που θέλουμε να κάνουμε είναι 1) να επιτρέψουμε το πρωτόκολλο cache_object μόνο από τον Η/Υ που τρέχει τον proxy. 2) Να επιτρέψουμε την πρόσβαση στον localhost, 3) να απορρίψουμε αιτήσεις CONNECT σε μη ασφαλή SSL port, 4) να απορρίψουμε την πρόσβαση στους τεχνικούς τα σαββατοκύριακα, 5) να απορρίψουμε την πρόσβαση στους πωλητές στα site πορνογραφικού περιεχομένου καθώς και το κατέβασμα αρχείων βίντεο τύπου avi, mpeg και mpg, 6) να επιτρέψουμε την πρόσβαση στα υποδίκτυά μας και σε άλλα υποδίκτυα που πιθανώς να υπάρξουν σε κάθε άλλη περίπτωση και 7) να αποτρέψουμε την πρόσβαση σε οποιονδήποτε άλλον.

Ακόμα, θέλουμε οι αιτήσεις του υποδικτύου των τεχνικών προς των web server που βρίσκεται σε αυτό το υποδίκτυο να μην περνούν από την cache, να πηγαίνουν απ' ευθείας.

Έτσι, συντάσσουμε τους εξής κανόνες με τη σειρά που είπαμε:

1) http_access allow manager localhost
2) http_access allow localhost
3) http_access deny manager
4) http_access deny ! Safe_ports
5) http_access deny Connect ! SSL_ports
6) http_access deny texniko weekends
7) http_access deny poliseis downloads
8) http_access deny poliseis badsites
9) http_access allow private_nets
10) http_access deny all

και για απ’ ευθείας πρόσβαση στον web server του texniko από τους τεχνικούς

always_direct allow texniko

never_direct allow all.

Επίλογος

Στην εργασία αυτή δείξαμε την αναγκαιότητα της ασφάλειας των δικτύων με τα firewall και τη σημαντική συμβολή σε αυτά, των διακομιστών διαμεσολάβησης (proxy servers). Δείξαμε και εξηγήσαμε κάποια είδη και κάποιες αρχιτεκτονικές για την ασφάλεια των δικτύων καθώς και εφαρμογές τους. Αναλύσαμε τα είδη των διακομιστών διαμεσολάβησης και εξηγήσαμε ένα πολύ διαδεδομένο και χρήσιμο είδος των proxy, τους cache proxy servers και αρκετά δικά του χαρακτηριστικά.

Πάνω από όλα όμως δείξαμε την αναγκαιότητα της ασφάλειας σε ατομικό και συλλογικό επίπεδο, στην υπεράσπιση δικαιωμάτων του απλού πολίτη και την απόκρυψη υπερπολύτιμων δεδομένων επιχειρήσεων και οργανισμών. Εκεί έγκειται η δύναμη της ασφάλειας, στις ανάγκες που εξυπηρετεί και τις λειτουργίες που πραγματώνει και όχι στους αλγορίθμους που διαρκώς βελτιώνονται για να την ισχυροποιήσουν.

Το μέλλον στον τομέα απόκρυψης δεδομένων είναι ευοίωνο και οι τεχνολογικές εξελίξεις του πρόσφατου παρελθόντος αποτελούν τα εχέγγυα για ισχυρότερη ασφάλεια, για ισχυρότερη προστασία της ελευθερίας

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. “Building Internet Firewalls”
D.Brent Chapman & Elizabeth D. Zwicky
O’Reilly, First Edition, November 1995

2. “X RAM, Firewalls, τα αντιπυρικά τείχη”
Θ. Μότσιος
΄Τεύχος 6 Μάιος 2001 Ειδική έκδοση του RAM, Εκδόσεις Λαμπράκης “

3. “Web Caching”
Duane Wessels
O’ Reilly & Associates, Inc – 2001

4. “Squid. A user’s Guide”
Oskar Pearson
Copyright © 2000 by Oskar Pearson

5. “Linux Firewall and Proxy Server HOWTO, http://www.tldp.org
Mark Grennan

6. “Microsoft® Proxy Server 2.0 MCSE Study System”
Simmons Curt
IDG Books Worldwide, Inc - 1999

7. “Hacker Proof”
Lars Klander
Jamsa Press 1997

8. Squid a user’s guide : http://www.sdconsult.no/linux/Squid/user_guide/book1.htm

9.PureSight for Squid : http://www.icognito.com/PDF/PureSight_Squid.pdf

10.ACLs & Delay Pools Under Squid :
http://staff.pisoftware.com/bmarshal/publications/aclsquid.html

11. Basic Installing and Configuration of SQUID proxy on Caldera OpenLinux 3_1_1
:http://www.caldera-benelux.com/cookbooks/squid.cookbook.html

12. A Brief Introduction to Squid :
http://www.linuxgazette.com/issue78/adam.html

13.Adv-Routing-HOWTO :
http://sunsite.ui.ac.id/pub/linux/docs/HOWTO/Adv-Routing-HOWTO.html

14. Config-HOWTO : http://www.linuxselfhelp.com/HOWTO/Config-HOWTO/

15. Daemon News Setting Up Squid on FreeBSD :
http://ezine.daemonnews.org/200209/squid.html

16.Help Net Security:
http://www.net-security.org/article.php?id=109

17. Linux Headquarters ipchains Configuration :
http://www.linuxheadquarters.com/howto/networking/ipchains.shtml

18. IPCHAINS-HOWTO
Rusty Russell
v.1.0.8 4/7/2000

19. Linux Internet Server Security and Configuration Tutorial :
http://yolinux.com/TUTORIALS/LinuxTutorialInternetSecurity.html

20. Linux VPN Masquerade HOWTO :

21. squid reverse proxy :
http://squid.visolve.com/white_papers/reverseproxy.htm#top

22. Transparent Proxy with Linux and Squid mini-HOWTO :
http://www.ibiblio.org/pub/Linux/docs/HOWTO/mini/other-formats/html_single/TransparentProxy.html

23. Networking-HOWTO :
http://www.isg.inf.ethz.ch/docu/linux/redhat-7.0/doc/HOWTOS/Net-HOWTO

24. Security-HOWTO :
http://www.linux.org.tw/CLDP/HOWTO/Security-HOWTO.html

25. Implementing Transparent Caching using Squid
http://squid.visolve.com/white_paper/