Εργαστήριο 2

Ασκήσεις: Αλληλεπίδραση με Δεδομένα HTTP (Web) Server και Διαχείριση Δικτύου (nmap, iptables)

Πρωτόκολλο HTTP και Apache HTTP server

Το πρωτόκολλο HTTP ακολουθεί το μοντέλο πελάτη-εξυπηρετητή (client-server). Ο εξυπηρετητής ονομάζεται HTTP server και ο πελάτης HTTP client ή πιο απλά φυλλομετρητής (browser). Για το σκοπό της άσκησης θα χρησιμοποιήσουμε τον Apache HTTP server που είναι ήδη εγκατεστημένος στο VM σας. Πιο συγκεκριμένα, το εγκατεστημένο πρόγραμμα ονομάζεται apache2[1] και είναι ο Apache HTTP server, ο οποίος είναι σχεδιασμένος να τρέχει σαν αυτόνομη διεργασία (standalone daemon process). Από προεπιλογή, ο HTTP server είναι ρυθμισμένος στο να «ακούει» για αιτήσεις (http requests) στην θύρα (port) 80 μιας μηχανής αλλά αυτό μπορεί να τροποποιηθεί (δείτε πιο κάτω). Για κάθε εισερχόμενη αίτηση, ο HTTP server δημιουργεί είτε μια διεργασία (process) ή ένα νήμα (thread) για να χειριστεί την αίτηση. Οι διεργασίες και τα νήματα περιλαμβάνονται στην ύλη του μαθήματος, σε μεταγενέστερο στάδιο, και δεν θα μας απασχολήσουν στο εργαστήριο αυτό. Μπορείτε να διαχειριστείτε τον Apache HTTP server (όπως και κάθε υπηρεσία στο linux) μέσω της γραμμής εντολών.

Ο Apache HTTP server μπορεί να «σερβίρει» προς τον έξω κόσμο τα αρχεία και τους καταλόγους που βρίσκονται στον μονοπάτι /var/www/html της μηχανής σας. Το μονοπάτι αυτό ορίζεται από τη μεταβλητή DocumentRoot που βρίσκεται στο configuration file του apache (βλέπε πιο κάτω).

Εργαλείο nmap (network mapper)

Ένας τρόπος για να δείτε αν το πρόγραμμα apache2 τρέχει στη μηχανή σας, είναι να χρησιμοποιήσετε το εργαλείο nmap[2] που είναι ένα ανοικτού κώδικα εργαλείο για την εξερεύνηση του δικτύου και τον έλεγχο της ασφάλειας που είδαμε και στο προηγούμενο εργαστήριο. Το nmap είναι ένα από τα πολυτιμότερα και πιο γνωστά εργαλεία για τους διαχειριστές συστημάτων. Λειτουργεί ως σαρωτής ασφαλείας και χρησιμοποιείται για να ανακαλύψει κεντρικούς υπολογιστές και τις υπηρεσίες σε ένα δίκτυο υπολογιστών, δημιουργώντας έτσι ένα "χάρτη" του δικτύου. Η λειτουργία του παρέχει στο χρήστη μια αναλυτική εικόνα, του προς ελέγχου δικτύου φανερώνοντας πιθανά προβλήματα και ελλείψεις ασφάλειας. Στο εργαστήριο αυτό θα το χρησιμοποιήσετε για να σαρώσετε τη μηχανή σας και να ελέγξετε για το αν υπάρχουν ανοικτές θύρες (π.χ. η θύρα 80 στην περίπτωση σας) και ποιες είναι αυτές. Αν μια θύρα είναι ανοικτή, αυτό προφανώς σημαίνει ότι τρέχει κάποια υπηρεσία που «ακούει» στη θύρα αυτή. Γενικά, δεν υπάρχει λόγος να τρέχουν διάφορες επιπλέον υπηρεσίες, πέραν από αυτές που μας χρειάζονται σε μια μηχανή, γιατί μέσω των θυρών τους δύναται να εισέλθουν κακόβουλοι χρήστες.

Εργαλείο netstat (network statistics)

Ένας άλλος τρόπος για να δείτε αν το πρόγραμμα apache2 τρέχει στη μηχανή σας, είναι να χρησιμοποιήσετε το εργαλείο netstat[3]. Το netstat είναι ένα χρήσιμο εργαλείο ελέγχου σε μια μηχανή:

- για την προβολή της δικτυακής δραστηριότητας. Μέσω της εντολής netstat -a μπορείτε να δείτε τις εισερχόμενες/εξερχόμενες τοπικές/διαδικτυακές συνδέσεις. Πιο συγκεκριμένα μπορείτε να δείτε

1 Αν το πρόγραμμα apache δεν είναι εγκατεστημένο, να το εγκαταστήσετε μέσω της εντολής sudo apt install apache2.

2 Αν το εργαλείο nmap δεν είναι εγκατεστημένο, να το εγκαταστήσετε μέσω της εντολής sudo snap install nmap.

3 Το εργαλείο netstat είναι μέρος του πακέτου net-tools και μπορεί να εγκατασταθεί μέσω της εντολής sudo apt-get -y install net-tools.

όλα τα sockets[4] σε κατάσταση σύνδεσης ή σε κατάσταση αναμονής για σύνδεση). Μέσω της εντολής netstat -l μπορείτε να δείτε τα sockets που είναι σε κατάσταση αναμονής για σύνδεση, listening sockets.

- για την προβολή των πινάκων δρομολόγησης (routing tables) μέσω netstat -r,
- για την προβολή στατιστικών σχετικά με τις διεπαφές (interfaces) μέσω netstat -i
- και άλλα ενδιαφέροντα στοιχεία για τις δικτυακές διασυνδέσεις.

Η εντολή αυτή είναι σημαντική για κάθε χρήστη, καθώς μπορεί να δει εάν κάποιο trojan ή spyware, πραγματοποιεί συνδέσεις στη μηχανή του χωρίς να το γνωρίζει. Στην εργασία αυτή μπορείτε να χρησιμοποιήσετε το εργαλείο για να δείτε στοιχεία για τις διεπαφές της μηχανής σας.

Χρήσιμες εντολές

Με τη βοήθεια των εντολών nmap, netstat μέσω του terminal μπορείτε να δείτε ποιες υπηρεσίες τρέχουν στη μηχανή σας (localhost – συζητούμε γι' αυτό πιο κάτω) και σε ποιες θύρες ακούνε. Οι πιο κάτω εντολές παρουσιάζουν πως μπορείτε να δείτε τις τρέχουσες υπηρεσίες της μηχανής σας και να διαχειριστείτε τον HTTP server:

- `nmap localhost`

Το εργαλείο nmap είναι ένα ανοικτού κώδικα εργαλείο για την εξερεύνηση του δικτύου και τον έλεγχο της ασφάλειας. Το nmap είναι ένα από τα πολυτιμότερα και πιο γνωστά εργαλεία για τους διαχειριστές συστημάτων. Λειτουργεί ως σαρωτής ασφαλείας και χρησιμοποιείται για να ανακαλύψει κεντρικούς υπολογιστές και τις υπηρεσίες σε ένα δίκτυο υπολογιστών, δημιουργώντας έτσι ένα "χάρτη" του δικτύου. Η λειτουργία του παρέχει στο χρήστη μια αναλυτική εικόνα, του προς ελέγχου δικτύου φανερώνοντας πιθανά προβλήματα και ελλείψεις ασφάλειας. Στο ερώτημα αυτό θα το χρησιμοποιήσετε για να σαρώσετε τη μηχανή σας και να ελέγξετε για το αν υπάρχουν ανοικτές θύρες και ποιες είναι αυτές. Αν μια θύρα είναι ανοικτή, αυτό προφανώς σημαίνει ότι τρέχει κάποια υπηρεσία που «ακούει» στη θύρα αυτή.

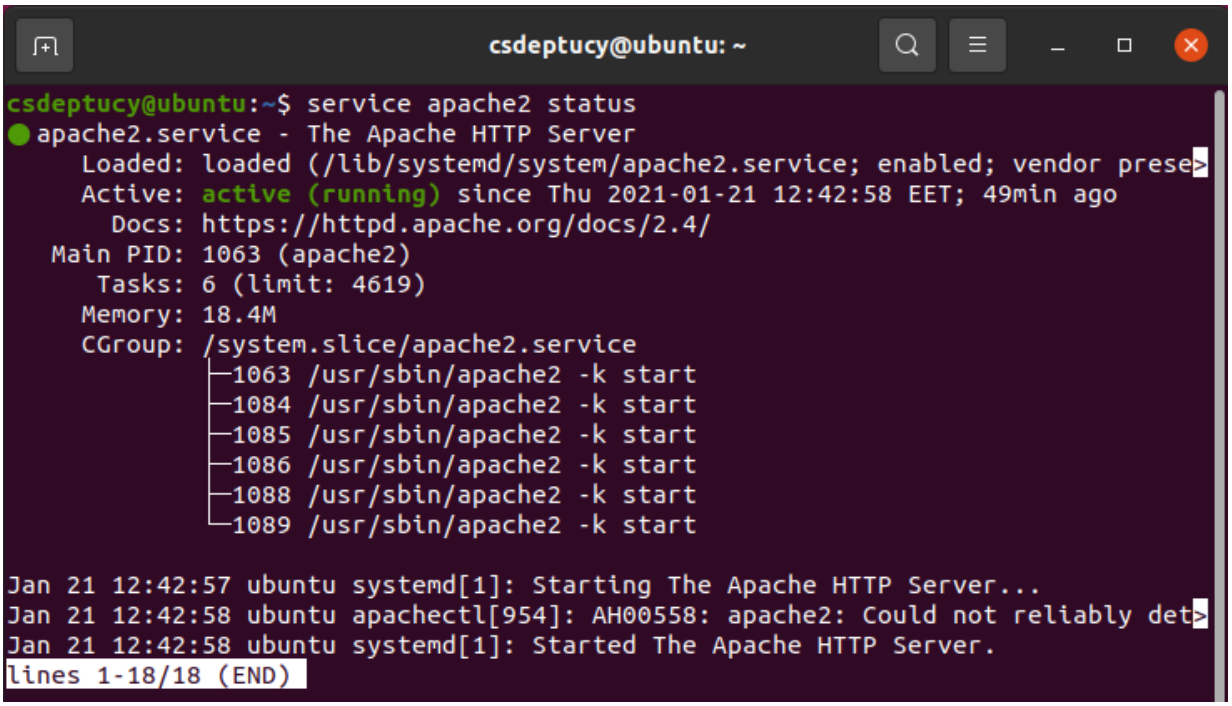
```
Starting Nmap 7.91 ( http://nmap.org ) at 2021-01-21 13:11 EET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000050s latency).
Not shown: 997 closed ports
PORT         STATE SERVICE
22/tcp       open  ssh
80/tcp       open  http
631/tcp      open  ipp
3306/tcp     open  mysql

Nmap done: 1 IP address (1 host up) scanned in 1.04 seconds
```

Από όσο βλέπετε, οι μόνες υπηρεσίες που τρέχουν στην πιο πάνω μηχανή είναι η ssh (ασφαλής απομακρυσμένη πρόσβαση στη TCP θύρα 22), η http (http server στη TCP θύρα 80), η ipp (internet printing protocol – CUPS service) και η mysql (mysql server στη θύρα 3306).

- `service apache2 status`

Εντολή που δείχνει την κατάσταση της υπηρεσίας HTTP server. Προς το παρόν, λόγω του ότι είναι ενεργή θα εκτυπωθεί το μήνυμα:

4 Socket (υποδοχή) ονομάζεται το τερματικό σημείο (endpoint) ενός αμφίδρομου διαύλου επικοινωνίας (two-way communication link) μεταξύ 2 διεργασιών που επικοινωνούν μέσω του δικτύου (είτε βρίσκονται πάνω στην ίδια μηχανή ή βρίσκονται σε ξεχωριστές μηχανές).



- `service apache2 start`

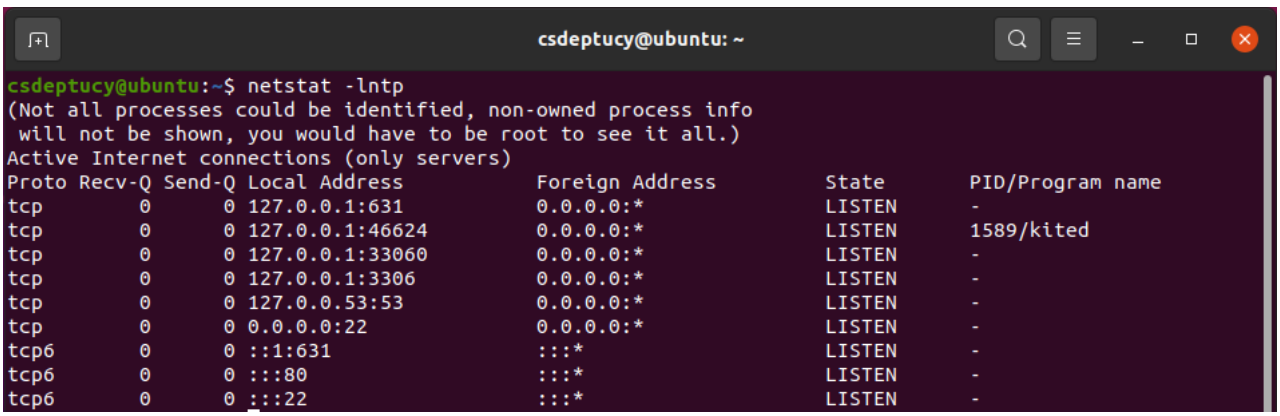
  Εντολή που ενεργοποιεί τον HTTP server. Αν ο HTTP server είναι απενεργοποιημένος πότε αν δώσουμε την εντολή:
  "`service httpd status`" θα λάβουμε την ίδια απάντηση που λάβαμε και πιο πάνω.

- `nestat`

  Η εντολή αυτή ανάλογα με τις παραμέτρους που δέχεται σαν όρισμα μπορεί να εκτυπώσει τις δικτυακές συνδέσεις της τοπικής μηχανής, τους πίνακες δρομολόγησης, στατιστικά για τις διεπαφές (interfaces) της τοπικής μηχανής κ.α. Αν δεν δοθεί κανένα όρισμα εκτυπώνει τις ενεργές συνδέσεις.

- `netstat -lntp`

  Η εντολή αυτή παρουσιάζει τα sockets που είναι σε κατάσταση αναμονής (listening sockets) για σύνδεση (-l), παρουσιάζοντας τις υπηρεσίες με τον αριθμό της θύρας στην οποία ακούνε (-n) (httpd=80, sshd=22, smtp=25), τις ταυτότητες (PID) των διεργασιών (-p) και έχουν σχέση με το πρωτόκολλο tcp (-t).



Στη στήλη Local Address, η διεύθυνση 127.0.0.1 (localhost ή loopback address) είναι η διεύθυνση της τοπικής μηχανής και χρησιμοποιείται συνήθως για τη διάγνωση και αντιμετώπιση προβλημάτων αλλά και για τοπική σύνδεση σε εξυπηρετητές (servers) που τρέχουν πάνω στην ίδια τη μηχανή. Για παράδειγμα, όταν εγκαταστήσουμε ένα server στην τοπική μηχανή για να επικοινωνήσουμε μαζί του χρειαζόμαστε το IP του και αυτό είναι το 127.0.0.1. Στην πιο πάνω εικόνα, ο MySQL server ακούει (LISTEN) στο 127.0.0.1:3306 που συνεπάγεται ότι δέχεται συνδέσεις (αιτήσεις) στη θύρα 3306 μόνο αν προέρχονται από την ίδια μηχανή (127.0.0.1). Δεν μπορεί να απαντήσεις σε αιτήσεις που προέρχονται από άλλες μηχανές (δηλ. με IP εκτός από το 127.0.0.1).
Εκτός από την διεύθυνση 127.0.0.1, μια μηχανή μπορεί να έχει και άλλες IP διευθύνσεις που δίνονται από τα δίκτυα στα οποία συνδέεται η μηχανή. Αν η μηχανή έχει 2 διεπαφές (interfaces ή κάρτες δικτύου) και μέσω αυτών συνδέεται σε 2 διαφορετικά δίκτυα τότε κατά συνέπεια θα έχει επιπλέον 2 διευθύνσεις IP π.χ. 192.168.1.10 και 10.0.2.7 (που εκχωρούνται συνήθως από τους routers κάθε δικτύου).
Η διεύθυνση 0.0.0.0 που φαίνεται στην πιο πάνω εικόνα αντιστοιχεί σε όλες οι διευθύνσεις IPv4. Με απλά λόγια, ο SSH server που ακούει στο 0.0.0.0:22 δέχεται συνδέσεις (αιτήσεις) στη θύρα 22 από οποιαδήποτε IPv4 διεύθυνση.
Η διεύθυνση ::: που φαίνεται στην πιο πάνω εικόνα αντιστοιχεί σε όλες οι διευθύνσεις IPv6. Με απλά λόγια, οι HTTP & SSH servers που ακούνε στο :::80 και :::22 αντίστοιχα δέχονται συνδέσεις (αιτήσεις) στη θύρα 80 και 22 αντίστοιχα από οποιαδήποτε IPv6 διεύθυνση.

- `service apache2 stop`
  Η εντολή αυτή απενεργοποιεί τον HTTP server.

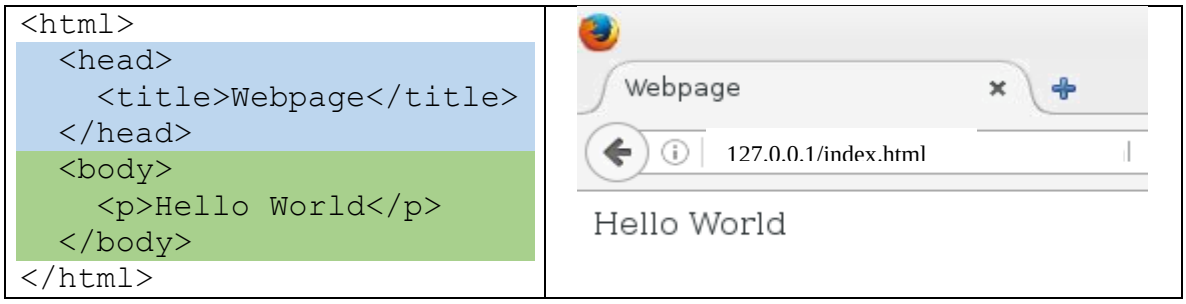
- `vim /etc/apache2/apache2.conf`
  Η εντολή αυτή ανοίγει το configuration file του HTTP server. Αν κάνουμε οποιαδήποτε αλλαγή στο αρχείο αυτό, για να ενεργοποιηθεί θα πρέπει να επανεκκινήσουμε τον HTTP server με την εντολή "`service apache2 restart`".

- `vim /etc/apache2/sites-available/000-default.conf`
  Η εντολή αυτή ανοίγει ένα δεύτερο configuration file του HTTP server σχετικά με τα virtual hosts (websites που μπορεί να «σερβίρει» ο HTTP server).

  Μέσα σε αυτό το αρχείο υπάρχουν διάφορες πληροφορίες όπως η γραμμή:

  ```
  DocumentRoot "/var/www/html"
  ```

  που δηλώνει ποιος κατάλογος (μονοπάτι) της μηχανής θα είναι η ρίζα των αρχείων και καταλόγων (για τον κάθε virtual host) τα οποία θα είναι ορατά στον έξω κόσμο μέσω του HTTP server. Για παράδειγμα, στον κατάλογο `/var/www/html` βάλετε το αρχείο `index.html` με τον κώδικα html που φαίνεται πιο κάτω αριστερά, και ανοίξετε τον browser (είτε στο VM ή το host machine) και πληκτρολογήστε τη διεύθυνση VM_IP_address/index.html. Αν ο HTTP server είναι ενεργοποιημένος, θα δείτε στον browser αυτό που φαίνεται στην επόμενη εικόνα στα δεξιά.



Εικόνα 1: Στα αριστερά φαίνεται αρχείο index.html με το <head> και το <body> sections. Στα δεξιά φαίνεται η απεικόνιση του αρχείου από τον φυλλομετρητή Mozilla Firefox μέσα από το VM.

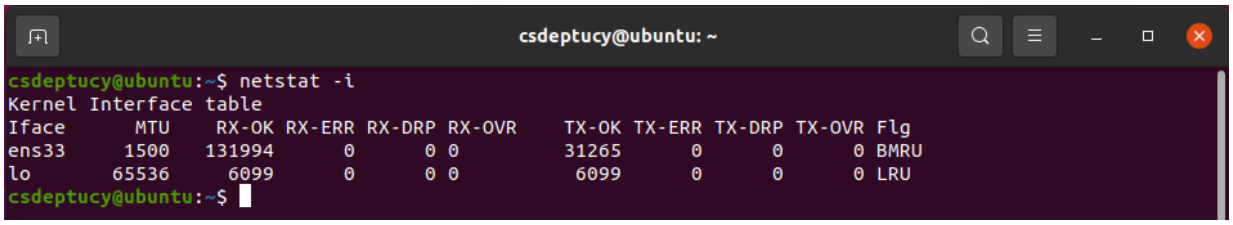
- `vim /etc/apache2/ports.conf`

  Η εντολή αυτή ανοίγει ένα ακόμα configuration file του HTTP server που αναφέρεται στις θύρες (ports) με τις οποίες σχετίζεται ο HTTP server
  Μια σημαντική γραμμή στο εν λόγω αρχείο είναι η γραμμή που αναφέρει τη θύρα (port) στην οποία ακούει ο HTTP server. Συγκεκριμένα η γραμμή:

```
Listen 80
```

  υποδηλοί ότι ο HTTP server δέχεται αιτήσεις στη θύρα 80 που προέρχονται από οποιαδήποτε διεπαφή (interface) της μηχανής. Για να δείτε ποιες διεπαφές έχει η μηχανή σας εκτελέστε την εντολή:

```
netstat –i
```



  Η πρώτη διαπαφή (lo), loopback interface, που αναφέραμε ήδη πιο πάνω, είναι μια εικονική διεπαφή δικτύου την οποία μια μηχανή χρησιμοποιεί για να επικοινωνήσει με τον εαυτό της. Η διεπαφή ens33 είναι επίσης εικονική διεπαφή δικτύου για τη διασύνδεση της μηχανής με τον έξω κόσμο. Είναι εικονική διεπαφή μιας και η μηχανή σας δεν είναι πραγματική αλλά εικονική. Σε πραγματικές μηχανές, θα δείτε συνήθως τις διεπαφές eth0 (κάρτα Ethernet) και wlan0 (κάρτα ασύρματου δικτύου π.χ. wifi). Αν υπάρχουν 2 κάρτες Ethernet, η πρώτη θα έχει το όνομα eth0 και η άλλη το eth1 κ.ο.κ.

  Αν αλλάξουμε την πιο πάνω γραμμή του configuration file σε:

```
Listen 80
Listen 8080
```

  τότε ο HTTP server δέχεται αιτήσεις και στη θύρα 80 και στη θύρα 8080, από οποιαδήποτε διεπαφή. Αν αλλάξουμε την πιο πάνω γραμμή του configuration file σε:

```
Listen 127.0.0.1:80
Listen 8080
```

τότε ο HTTP server δέχεται αιτήσεις στη θύρα 80 μόνο ΑΝ προέρχονται από την ίδια τη μηχανή (όχι από τον έξω κόσμο) και στη θύρα 8080 από οποιαδήποτε διεπαφή (τοπική – localhost) ή από τον έξω κόσμο.
Δεν είναι ανάγκη να κάνετε τις 2 αυτές αλλαγές.

Χρήσιμα αρχεία

Τα αρχεία καταγραφής συμβάντων (log files) περιέχουν εξαιρετικά χρήσιμες πληροφορίες για τη εύρυθμη λειτουργία του HTTP server. Στα λειτουργικά συστήματα τύπου Red Hat όπως είναι το CentOS τα log files του HTTP server βρίσκονται στον κατάλογο /var/log/apache2. Μέσα στο φάκελο αυτό μπορείτε να διαφόρων ειδών log files όπως για παράδειγμα error logs και access logs.
Μέσα στα error logs ο Apache HTTP server βάζει διάφορες διαγνωστικές πληροφορίες για τη λειτουργία του και καταγράφει σφάλματα που παρουσιάζονται κατά την διάρκεια των αιτήσεων που δέχεται. Είναι η πρώτη πηγή πληροφοριών που θα κοιτάξει ο διαχειριστής όταν συμβεί ένα πρόβλημα κατά την εκκίνηση του server ή με τη λειτουργία του server γενικότερα διότι συχνά περιέχει λεπτομέρειες σχετικά με το τι πήγε λάθος και πώς μπορεί να διορθωθεί. Μια τυπική γραμμή ενός τέτοιου αρχείου είναι:

```
[Fri Jan 08 10:42:29.902022 2021] [core:error] [pid 35708:tid 4328636416]
[client 72.15.99.187] File does not exist:
/usr/local/apache2/htdocs/favicon.ico
```

Μέσα στα access logs ο Apache server καταγράφει όλες τις αιτήσεις που δέχεται. Περιέχει τον αποστολέα και το περιεχόμενο της κάθε αιτήσεως. Από τα περιεχόμενα των αρχείων αυτών μπορεί μετά από περαιτέρω ανάλυση να παραχθούν χρήσιμα στατιστικά.

Ερωτήσεις

Με βάση τις πιο πάνω πληροφορίες απαντήστε τις ερωτήσεις:

1) Εκτυπώστε όλες τις πληροφορίες για τις ενεργές συνδέσεις της μηχανής σας που χρησιμοποιούν το πρωτόκολλο TCP

2) Εκτυπώστε μόνο τα ονόματα (domain name) των μηχανών που είναι ενεργά συνδεδεμένες με τη μηχανή σας.

3) Εκτυπώστε μόνο τα ονόματα (domain name) των μηχανών που είναι ενεργά συνδεδεμένες με τη μηχανή σας έτσι ώστε το κάθε όνομα να εμφανίζεται μόνο μια φορά αλλά να φαίνεται και ο αριθμός των συνδέσεων της κάθε μηχανής.

4) Εκτυπώστε το όνομα της διεπαφής (interface) που έχει λάβει τα πιο πολλά πακέτα καθώς και τον αριθμό των εισερχόμενων πακέτων.

5) Εκτυπώστε τα μόνο τα ονόματα των log files του HTTP server που δημιουργήθηκαν μέσα στο μήνα Ιανουάριο.

6) Επειδή τα log files στη μηχανή σας δεν έχουν πολλές πληροφορίες προς επεξεργασία, θα κατεβάσουμε δεδομένα από μια ιστοσελίδα που περιέχει traces από access logs. Πιο συγκεκριμένα θα κατεβάσουμε ένα access log file από ένα FTP server που βρίσκεται στο Research Triangle Park, NC στην Αμερική και περιέχει πληροφορίες αιτήσεων προς τον server για ένα 24 ωρο. Διαβάστε περισσότερα στοιχεία για το συγκεκριμένο access log file εδώ. Κατεβάστε το στη μηχανή σας και αποσυμπιέστε:

```
wget ftp://ita.ee.lbl.gov/traces/epa-http.txt.Z

uncompress epa-http.txt.Z
```

α) Τυπώστε τα ονόματα (domain names) των 10 μηχανών που έκαναν τις πιο πολλές αιτήσεις GET στον HTTP server (μπορείτε να τυπώσετε και τον αριθμό των αιτήσεων).

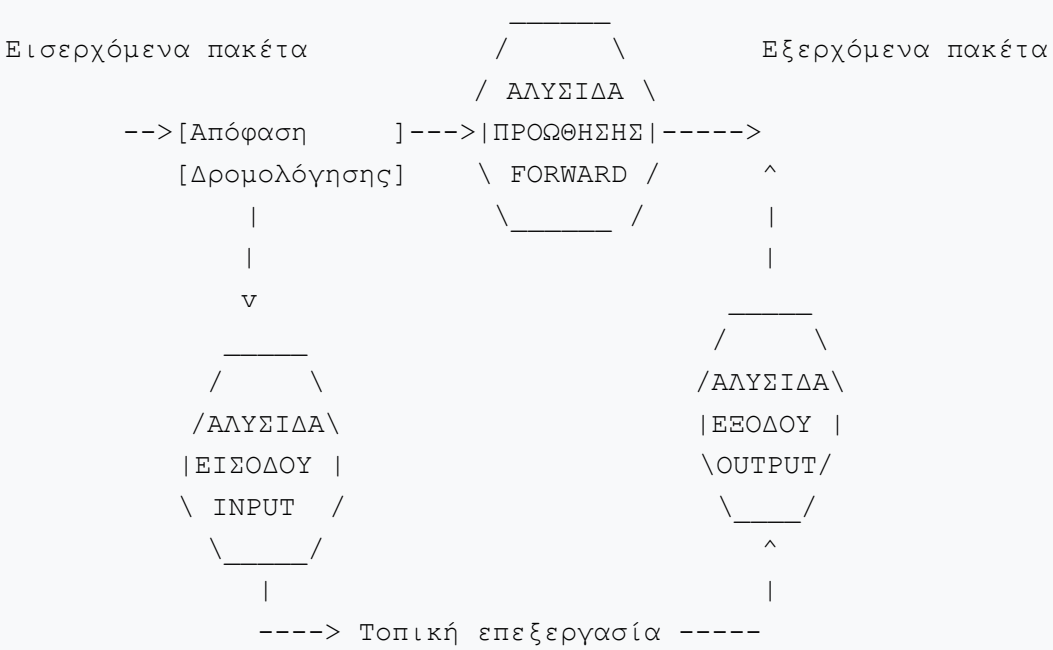
β) Τυπώστε τα ονόματα (domain names) των 10 μηχανών που έκαναν τις πιο πολλές αιτήσεις POST στον HTTP server (μπορείτε να τυπώσετε και τον αριθμό των αιτήσεων).

7) Χρησιμοποιώντας το εργαλείο nmap μπορείτε να δείτε τις ανοιχτές εισερχόμενες θύρες (ports) της μηχανής σας:

```
nmap localhost | grep open
```

Όπως θα δείτε από την πιο πάνω εντολή, η μηχανή σας πιθανόν να έχει ανοικτές τις θύρες 22 (ssh), 80 (http) και 3306 (mysql). Γενικά, σε μια μηχανή, δεν υπάρχει λόγος να τρέχουν διάφορες υπηρεσίες πέραν από αυτές που μας χρειάζονται, γιατί μέσω των θυρών τους δύναται να εισέλθουν κακόβουλοι χρήστες. Στο ερώτημα αυτό θα χρησιμοποιήσουμε το εργαλείο iptables που μπορεί να θέσει κανόνες στα πακέτα που φτάνουν σε μια μηχανή.

Το παρακάτω διάγραμμα μας δείχνει τη βασική δομή λειτουργίας του φιλτραρίσματος πακέτων στο linux. Οι κύκλοι αντιπροσωπεύουν αλυσίδες. Μια αλυσίδα είναι ένα σύνολο από κανόνες.



Όταν ένα πακέτο φτάσει σε έναν κύκλο του διαγράμματος, τότε η αλυσίδα εξετάζεται για να αποφασιστεί η μοίρα του πακέτου. Αν η αλυσίδα πει DROP (απόρριψη), τότε το πακέτο σκοτώνεται, αλλιώς αν η αλυσίδα πει ACCEPT (αποδοχή) , τότε συνεχίζει την πορεία του στο διάγραμμα.
Μία αλυσίδα είναι μια λίστα ελέγχου από κανόνες. Κάθε κανόνας λέει 'Αν η επικεφαλίδα του πακέτου είναι ως εξής τότε κάνε αυτό με το πακέτο'. Αν ο κανόνας δεν ικανοποιείται από το πακέτο τότε εξετάζεται ο επόμενος κανόνας στην αλυσίδα. Τέλος αν δεν υπάρχουν άλλοι κανόνες προς έλεγχο ο πυρήνας συμβουλεύεται την **πολιτική** της αλυσίδας για να αποφασίσει τι θα πράξει. Σε ένα συνειδητό προς την ασφάλεια σύστημα συνήθως η πολιτική είναι να απορρίπτεται το πακέτο.
Όταν ένα **πακέτο** φτάνει στον υπολογιστή μας ο πυρήνας του linux κοιτάει στην επικεφαλίδα του πακέτου το πεδίο **προορισμός** και παίρνει μια απόφαση ανάλογα με τον προορισμό. Αυτό το ονομάζουμε **δρομολόγηση**.

- Αν το πακέτο προορίζεται για τον υπολογιστή μας τότε κατευθύνεται στην αλυσίδα εισόδου (INPUT). Εκεί εξετάζεται με τη σειρά από τους κανόνες της αλυσίδας και τέλος από την πολιτική. Αν περάσει επιτυχώς τότε κατευθύνεται προς διεργασίες που θα το χρησιμοποιήσουν ειδάλλως απορρίπτεται.
- Ειδάλλως αν δεν έχουμε ενεργοποιήσει την προώθηση το πακέτο θα απορριφθεί. Όμως αν έχουμε ενεργοποιήσει την προώθηση το πακέτο θα μεταφερθεί στην αλυσίδα προώθηση και εκεί αν ικανοποιήσει τους κανόνες της θα μεταβιβαστεί προς κάποια κάρτα δικτύου του υπολογιστή για να συνεχίσει τη διαδρομή της αλλιώς θα απορριφθεί (DROP)
- Τέλος ένα πρόγραμμα μπορεί να στείλει πακέτα στο διαδίκτυο. Αυτά πάνε στην αλυσίδα εξόδου (OUTPUT) και εφόσον ικανοποιήσουν τους κανόνες της (ACCEPT) συνεχίζουν προς τη διασύνδεση που προορίζονται

Με την εντολή iptables μπορούμε να επεξεργαζόμαστε τις αλυσίδες κάνοντας μια σειρά από ενέργειες. Οι τρεις προκαθορισμένες αλυσίδες INPUT, OUTPUT FORWARD δεν μπορούν να διαγραφούν. Ενέργειες διαχείρισης ολόκληρης αλυσίδας:

- Δημιουργία νέας αλυσίδας (-N).
- Διαγραφή άδειας αλυσίδας (-X).
- Αλλαγή της πολιτικής μιας προκαθορισμένης αλυσίδας (-P).
- Εμφάνιση λίστας των κανόνων μιας αλυσίδας (-L).
- Καθάρισε όλους τους κανόνες από μια αλυσίδα (-F).
- Μηδένισε το πακέτο και τους μετρητές byte σε όλους τους κανόνες μιας αλυσίδας. (-Z).

Υπάρχουν πολλοί τρόποι να χειριστείς κανόνες μέσα σε μια αλυσίδα:

- Προσθήκη νέου κανόνα σε μια αλυσίδα (-A).
- Εισαγωγή νέου κανόνα σε κάποια θέση στην αλυσίδα (-I).
- Αντικατέστησε έναν κανόνα σε κάποια θέση σε μια αλυσίδα (-R).
- Διέγραψε έναν κανόνα σε κάποια θέση σε μια αλυσίδα, ή τον πρώτο που ικανοποιεί ένα δοθέν κριτήριο (-D).

```
Αρχικά όλες οι αλυσίδες είναι κενές:
 # sudo iptables -L
 Chain INPUT (policy ACCEPT)
 target     prot opt source               destination

 Chain FORWARD (policy ACCEPT)
 target     prot opt source               destination

 Chain OUTPUT (policy ACCEPT)
 target     prot opt source               destination
```

Η πιο κάτω εντολή προσθέτει (-A) ένα κανόνα στην αλυσίδα INPUT για πακέτα του πρωτοκόλλου (-p) TCP που κατευθύνονται στη θύρα με αριθμό PORT-NUMBER. Τα πακέτα αυτά θα μεταπηδήσουν (jump, -j) στην ενέργεια απόρριψη (DROP).

```
sudo iptables -A INPUT -p tcp --destination-port {PORT-NUMBER} -j DROP
```

Για να δούμε τους κανόνες που έχουν τεθεί στην αλυσίδα INPUT, εκτελούμε την εντολή:

```
sudo iptables –L INPUT
```

Δώστε την εντολή που προσθέτει ένα κανόνα στην αλυσίδα INPUT για την απόρριψη πακέτων του πρωτοκόλλου TCP που κατευθύνονται στη θύρα 631.

Επιβεβαιώστε με την εντολή nmap ότι όντως η θύρα 631 δεν είναι πλέον ανοικτή.