Σύγχρονες Εφαρμογές Ασφάλειας Δικτύων

Άσκηση 1

Αναγνωστόπουλος Άγγελος Νικόλαος
5ο Έτος ΗΜΤΥ
up1066593

1.1 Απαντήσεις στις ερωτήσεις κατανόησης:

1. Για να είμαστε ακριβείς, δεν είναι απαραίτητο να συμβαίνει κάτι τέτοιο. Όταν όμως μας ενδιαφέρει η ταχύτητα τότε η ενσωμάτωση στο kernel είναι μονόδρομος. Θα μπορούσαμε να υλοποιούμε το φιλτράρισμα με διεργασίες οι οποίες δημιουργούνται από τον χρήστη, αυτό όμως θα μας κόστιζε πολύ σε χρόνο. Για αυτό το λόγο τα firewalls χτίζονται πάνω σε modules τα οποία βρίσκονται πάνω στο kernel.

2. Ένα παραδοσιακό packet filtering firewall είναι αυτό το οποίο τρέχει σε ένα host machine και διαχειρίζεται το φιλτράρισμα της κίνησης στο δίκτυο τόσο από, όσο και πρός την μηχανή. Τα proxies είναι μηχανές οι οποίες κάνουν την ίδια διαδικασία και χρησιμοποιούνται για να έχουμε ένα κεντρικό σημείο policy enforcement. Θα μπορούσαμε δηλαδή να έχουμε ένα proxy firewall “μπροστά” από ένα υποδίκτυο το οποίο προστατεύει όλους τους υπολογιστές μέσα σε αυτό, αφού δρα ως ο διαμεσολαβητής μεταξύ του υποδικτύου και του υπολοίπου κόσμου, εφαρμόζοντας τους κανόνες που έχουν οριστεί στο firewall του σε όλη τη σχετική κίνηση.

Προφανέστατα δεν απαγορεύεται να έχουμε χρήση και packet filtering firewall σε καθέναν από τους υπολογιστές που το proxy εξυπηρετεί (και αυτό είναι το πιο σύνηθες).

3. Οι πίνακες που υποστηρίζονται είναι οι: Filter, Mangle, Nat, Raw

4. Αυτά προστείθενται στην αντίστοιχη “αλυσίδα” με κανόνες με χρήση του flag **-A.** Το τμήμα του πακέτου το οποίο εξετάζεται είναι προφανέστατα ο header (αν και υπάρχει δυνατότητα για deep packet inspection σε κάποια NGFs αλλά δεν είναι της παρούσης).

5. Εάν το πακέτο δεν γίνει match με κανέναν από τους κανόνες που έχουμε ορίσει, τότε αυτό θα το αναλάβει το policy της αλυσίδας. Το policy είναι ο γενικός κανόνας που εφαρμόζεται σε όλα τα πακέτα εάν δεν γίνουν πρωτίστως match με κάποιον άλλο κανόνα. Μπορούμε να το ορίσουμε με το flag -P και τυπικά είναι DROP για τα εισερχόμενα πακέτα.

6. $sudo iptables -A INPUT -p tcp ! --syn -m state NEW -j DROP

Το -m μας λέει ότι θα γίνει χρήση του module state το οποίο ελέγχει την κατάσταση ενός πακέτου. Θα μπορούσαμε να γράψουμε τον κανόνα “με το χέρι” ελέγχοντας manually το εάν έχουμε ήδη εγκατεστημένη σύνδεση αλλά τα modules του iptables είναι σωτήρια σε μερικά κοινά administrative tasks.

7. Αρχικά όλες οι αλυσίδες αρχικοποιούνται να κάνουν ACCEPT τις συνδέσεις.
Για να κάνουμε flush τις αλυσίδες η εντολή είναι αρκετά προφανής:

$sudo iptables –flush

ή εναλλακτικά

$sudo iptables -t <name> -F

8. Σημαίνει “Οποιαδήποτε ICMP εντολή”. Δεν διακρίνει μεταξύ request, reply κ.α.

9. echo-request (ping)
echo-reply (pong)

10. Οι εντολές που υπάρχουν στον πίνακα raw έχουν προτεραιότητα πάνω σε όλους τους υπόλοιπους. Ό,τι ορίζεται σε αυτόν, θα είναι και το πρώτο πράγμα που θα ελεγχθεί για match.

11. Τα αιτήματα σύνδεσης ssh έρχονται by default στην πόρτα 22. Δεδομένου ότι δεν έχουμε κάνει κάποια αλλαγή στο configuration του sshd στον server μας, η εντολή θα πρέπει να κλείνει αυτή τη θύρα για εισερχόμενα tcp connections. Δηλαδη:

$sudo iptables -P INPUT DROP
$sudo iptables -A INPUT -p tcp –dport 22 -j DROP

ή εάν θέλουμε να στέλνουμε πίσω απάντηση ότι το αίτημα απορρίφθηκε:

$sudo iptables -A INPUT -p tcp –dport 22 -j REJECT

12. Ελέγχεται η κατάσταση του πακέτου με έλεγχο του header του από κάποιο connection tracking system (πχ. από το module **state**)

13. NEW, ESTABLISHED, RELATED, INVALID

14. Ο απλούστερος τρόπος αντί να αποθηκεύσουμε τις εντολές μας στα αρχεία που το λειτουργικό μας τρέχει στο startup από μόνοι μας, είναι να χρησιμοποιήσουμε το πολύ χρήσιμο εργαλείο iptables-persistent. Για την εγκατάσταση του γράφουμε:

$sudo apt install iptables-persistent (Εκδόσεις deb)
$sudo apt install iptables-services (Εκδόσεις rhl)

$sudo iptables-save > /*etc*/iptables/rules.vX
Όπου το X είναι 4 ή 6 ανάλογα με το αν οι κανόνες μας αφορούν ipv4 ή ipv6.

1.2 Εργασία στο iptables:

Παρόμοιες εντολές αλλά και πολλές ακόμα τρέχω στο RaspberryPi4 (headless) που χρησιμοποιώ σαν backup server. Θα μπορούσαμε να ορίζουμε σε μεταβλητές τις διευθύνσεις για ευαναγνωσιμότητα αλλά μίας και τα παραδείγματα ήταν απλά θεώρησα σωστό να τις βάλω κατευθείαν στις εντολές.

$sudo iptables -P OUTPUT ACCEPT

$sudo iptables -A INPUT -p tcp -s 150.140.139.194/27 –dport 22 -j ACCEPT

$sudo iptables -A INPUT -p tcp -s 192.168.1.1/16 –dport 22 -j ACCEPT

$sudo iptables -A INPUT -p tcp -s 192.168.1.10 –dport 80 -j ACCEPT
$sudo iptables -A INPUT -p tcp -s 192.168.1.10 –dport 443 -j ACCEPT

$sudo iptables -A INPUT -p tcp --dport 25 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
$sudo iptables -A OUTPUT -p tcp --sport 25 -m conntrack --ctstate ESTABLISHED -j ACCEPT

$sudo iptables -A INPUT -p tcp --dport 143 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
$sudo iptables -A OUTPUT -p tcp --sport 143 -m conntrack --ctstate ESTABLISHED -j ACCEPT

$sudo iptables -A INPUT -p icmp –icmp-type echo-request -j ACCEPT

$sudo iptables -A INPUT -p tcp -j REJECT –reject-with tcp-reset
$sudo iptables -A INPUT -p icmp -j REJECT