STATIC PARSER AND TESTS

1. Dockerfile

Chmod & Chown
Θεώρηση:
Αν και το Dockerfile αφορά την κατασκευή του image του container αποφάσισα να τα εισάγω στον parser, απλά κάνοντας τη θεώρηση ότι αφού επιτρέπονται στο Dockerfile – και προφανώς αφορούν το image και δεδομένα σε αυτό – να θεωρούμε ότι θα επιτρέπονται και στη συνέχεια εντός του image (για να έχουμε και μια μικρή πληροφορία από το Dockerfile).
Επιπλέον, δοκιμάζοντας δύο αντικρουόμενους κανόνες σε Dockerfile και profile – "chmod 777 /test_file" και "deny /testfile w" – είδα ότι υπερισχύει μεταξύ των 2 το Dockerfile. Συνεπώς, δεν έχει νόημα να απαγορεύσω κάτι που συμβαίνει στο Dockerfile.

Έτσι σε κάθε εντολή προσθέτουμε αντίστοιχο chmod/chown κανόνα:

- Chmod:
chmod rule → not supported!
Εδώ για να επιτρέπεται το chmod χρειάζομαι κανόνα "/path w" ώστε να επιτρέπεται στον χρήστη με permission write να αλλάζει τα permissions ενός αρχείου.
Επιπλέον στο chmod, θα προσθέσουμε κανόνα για file access, όπου στο path θα δώσουμε τα αντίστοιχα permissions τα οποία θα αναφέρονται στον owner (*εδώ αφήνω ερωτηματικό, για το που θα πρέπει να αναφέρονται τα permissions στο apparmor, owner/group/others* ***???*** *δεν εχω βρει κάτι για να καθορίσω ξεχωριστά τα δικαιώματα των group και others*) .
Προσθέτοντας το owner πριν τον file access rule, το file access αναφερεται στον owner.
Πώς όμως να καθορίσω τα δικαιώματα των group και others???
πχ. owner /path w (αν παραλείψω το owner πάλι ισχύουν και για owner, οπότε λογικά χωρίς το owner τα δικαιώματα είναι ίδια για όλους – owner, group, others)

- Chown:
Χρειαζόμαστε "chown capability" για να είναι επιτρεπτό το chown.
Για να αποτρέψουμε το chown ακόμα και αν έχουμε chown capability αρκεί να μην δώσουμε write permission σε συγκεκριμενο path: "deny /home/* w" (αυτό όμως δεν μπορούμε να το προβλέψουμε στον static parser προς το παρόν)
chown rule → not supported!

- Copy, Add, Workdir
Δεν έχει νόημα να συμπεριλάβουμε τελικά τα paths αυτών, γιατί αφενός δεν θα αναφερθούμε σε paths του host στο AppArmor profile μας οπότε το source δεν έχει νόημα για μας, αφετέρου τα destination paths (και το path του workdir) δεν έχουν κάποια ιδιαιτερότητα από τις εντολές αυτές, είναι σαν οποιοδήποτε άλλο

path του container, άρα δεν έχουμε να τους δώσουμε κάποιο συγκεκριμένο permission. Οπότε δεν παίρνουμε πληροφορία τελικά από αυτά.

Τι νόημα όμως είχαν για τον host τα chmod και chown εντός του container? Γιατί μας ενδιαφέρουν σχετικά με το isolation μεταξύ host – container? Γιατί μπορεί να αφορούν path το οποίο έχει mount στον host και συνεπώς να επηρεάζει και "δικά μας" αρχεία… Όσο λιγότερες δυνατότητες έχει γενικά το container – άρα και εντός των mount – τόσο πιο καλυμένοι είμαστε..

2. Docker Compose

Έχω μια υποψία πως σε αρκετά σημεία που δεν είχα την αναμενόμενη συμπεριφορά είναι γιατί στο .yml file εισάγω και τη γραμμή "security_opt: apparmor:static_profile" και ίσως να μην δουλεύει όπως δίνοντας το σαν flag στο docker run όπως έκανα όταν δεν είχα .yml file παρά μόνο Dockerfile. Αυτό το σημείο θέλω να το διερευνήσω παραπάνω, γιατί αν συμβαίνει όντως κάτι τέτοιο τότε έχουμε πρόβλημα με το docker compose και την εισαγωγή apparmor profile με αυτό και πρέπει να δοκιμάσω ξανά όλα τα παρακάτω..

- Capabilities add / drop: Δοκιμασμένο ήδη με το capability chown, δουλεύει. (άσχετα από το docker compose)

- Ports:
Για να χρησιμοποιήσω ports χρειάζομαι τον κανόνα network. Χωρίς αυτό δεν θα δουλέψει. (network bind x to y → not supported παρολο που αναφέρεται στο documentation). Το capability network_bind_service δεν φαίνεται να παίζει κάποιο ρόλο, αν λείπει από το profile και πάλι μπορώ να χρησιμοποιήσω τα ports → ίσως συμπεριλαμβάνεται στο network, ενώ σκέτο το capablity αυτό χωρίς το network δεν αρκεί για να χρησιμοποιήσω ports.

- Mount: Το apparmor δεν επιτρέπει να έχεις writable directory με mount στο container…
Ιδέες:
mount options=rw dir → dir : Δεν δούλεψε
--privileged (το container) : Δεν δούλεψε
/dir rw (μετά το mount) : Δεν δούλεψε
:z μετά τα dirs στα ορίσματα υποτίθεται ότι το κάνει writable αλλά στο docker compose δεν έχω βρει πώς (αν έχει υλοποιηθεί….) μπορώ να το ενσωματώσω

Χωρίς profile (apparmor:unconfined) το mount destination directory είναι writable…

Χωρίς τον κανόνα mount στο profile, δεν δίνεται η δυνατότητα για bind στο container.
Αυτό είναι το πιο αδύνατο σημείο του apparmor profile… Θα συνεχίσω να το ψάχνω μήπως υπάρχει κάποιος άλλος τρόπος…

- Devices: Τελικά η σχέση των devices είναι σαν τα mounts οπότε με εντολή mount θα κάνω bind μεταξύ 2 devices. Βρες 2 devices και test.

Εδώ υπήρξε ένα πρόβλημα γιατί δεν μου σήκωνε το container με διάφορα devices που δοκίμασα, οπότε θέλω να το ψάξω λίγο περισσότερο για να βρω ένα σωστό παράδειγμα να τεστάρω.

- Rlimit:
Περνάει σαν εντολή, δεν ξέρω αν ειναι supported ακόμα και προς το παρόν η δοκιμη για proc απέτυχε…
Εως το 2012 δεν το είχαν υλοποιήσει, σύμφωνα με το παρακάτω mail… Τώρα? Κάνω εγώ κάτι λάθος ή απλά δεν έχει υλοποιηθεί ακόμα?

https://lists.ubuntu.com/archives/apparmor/2012-February/002098.html