

Διαχείριση χρονοσειρών σε εφαρμογές IoT

Τα τελευταία χρόνια με την εξάπλωση των αισθητήρων που είναι συνδεδεμένοι στο διαδίκτυο (Internet of Things - IoT) χρειάζεται να αποθηκεύονται μαζικά δεδομένα μετρήσεων σε μεγάλους ρυθμούς ή αλλιώς δεδομένα χρονοσειρών (TimeSeries Data). Στο περιβάλλον του IoT καλούμαστε εκτός από την δικτυακή συλλογή των μετρήσεων να κάνουμε και διαχείρισή τους εννοώντας την αποθήκευση, οπτικοποίηση (visualization) και μια τρόπο τινά «έτοιμη-επεξεργασία» εν είδη συναγερμών. Οι χρονοσειρές για IoT έχουν το χαρακτηριστικό ότι δεν είναι transactional, με τις παραδοσιακές βάσεις δεδομένων να μην είναι ιδιαίτερα αποδοτικές όσον αφορά τον χειρισμό τους. Σκοπός της παρούσας διπλωματικής είναι να αξιολογηθούν και να συγκριθούν δύο πλατφόρμες διαχείρισης χρονοσειρών η Prometheus [1] και η TimescaleDB [2].

Η Prometheus δεν ενδιαφέρεται για το εάν κάποιες τιμές δεν είναι διαθέσιμες κατά τη φάση εγγραφής στην υποδομή αποθήκευσης και επιτρέπει στον διαχειριστή να ορίζει πολιτικές λήθης (retention) ώστε να μην συσσωρεύονται «άχρηστες» πληροφορίες. Το δυνατό στοιχείο της είναι η γλώσσα περιγραφής PromQL [3,4], που χρησιμοποιείται συχνά στην παρακολούθηση δικτυακών υποδομών και είναι πιο εύχρηστη από την SQL. Η TimescaleDB είναι υλοποιημένη σαν επέκταση (extension) της PostgreSQL [6]. Παρέχει νέα χαρακτηριστικά όπως π.χ. υπερ-πίνακες (hyper-table) και συναρτήσεις χρόνου (π.χ. time_bucket), για τη χρονική διαχείριση των δεδομένων (datasource). Μπορεί να λειτουργήσει σαν υποδομή μακράς αποθήκευσης (Long Term Storage) για την Prometheus με χρήση κατάλληλων adapters [7] και παρέχει εργαλεία (π.χ. query builder) οπτικοποίησης των δεδομένων για το Grafana [8]. Τα προηγούμενα εργαλεία χρησιμοποιούνται σαν υπόβαθρο για να περιγράψουν συνθήκες (events) συναγερμών σε «υψηλότερα» εργαλεία π.χ. alert-manager [8] και hastic [9]. Το τελευταίο επιτρέπει να οριστούν «ενδιαφέροντα μοτίβα» (patterns) για να ενεργοποίηση συναγερμών.

Στην παρούσα διπλωματική θα αντληθούν δεδομένα από μια υφιστάμενη σχεσιακή (Relational - SQL) βάση (PostgreSQL 9) πραγματικών δεδομένων από υδρο-μετεωρολογικούς αισθητήρες. Θα στηθεί μια παράλληλη νέα «υποδομή» που θα εισάγει τις μετρήσεις σε βάση Prometheus και σε βάση TimescaleDB. Στη συνέχεια θα χρησιμοποιηθεί ο Query builder του grafana και θα γίνει αναπαράσταση με ρύθμιση συναγερμών με χρήση του alert-manager και του hastic.

[1] <https://prometheus.io/>

[2] <https://www.timescale.com/>

[3] PromQL <https://prometheus.io/docs/prometheus/latest/querying/basics/>

[4] PromQL and subqueries <https://prometheus.io/blog/2019/01/28/subquery-support/>

[6] <https://www.postgresql.org/>

[7] <https://docs.timescale.com/v1.2/tutorials/prometheus-adapter>

[8] Grafana <https://grafana.com/>

[9] Alert Manager <https://prometheus.io/docs/alerting/alertmanager/>

[9] hastic <https://hastic.io/>

Επικοινωνία: Ε. Δ. Συκάς (sykas@cn.ntua.gr), Δ. Καλογεράς (dkalo@noc.ntua.gr)