



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
HAROKOPIO UNIVERSITY

Υπολογιστικές και Δικτυακές Υποδομές II (ΕΦΠ06)

Ροή απόκτησης δεδομένων

Μαρίνος Κουβαράς, ap23011

8 Απριλίου 2024

Περίληψη

Παραδοτέα αναφορά εργασίας στο μάθημα Υπολογιστικές και Δικτυακές Υποδομές II (ΕΦΠ06) στην ενότητα Υπηρεσίες Νέφους - Υπολογιστικών Νεφών 2024. Η εργασία αφορά το δεύτερο ερώτημα “Ροή Απόκτησης Δεδομένων” και ολοκληρώθηκε με τη χρήση του Node-RED και του Minio.

Πίνακας Περιεχομένων

Περίληψη	2
Πίνακας Περιεχομένων	3
Σενάριο	4
Εισαγωγή	4
Opensky	4
Αρχική Εγκατάσταση	8
Υλοποίηση	8
Node-Red	8
minIO	8
Σχεδίαση	9
Συμπεράσματα	18

Σενάριο

Ροή απόκτησης δεδομένων

- Εύρεση μιας διαθέσιμης πηγής ανοικτών δεδομένων (API, file κλπ)
- Δημιουργία μιας ροής στο Node-RED για τη λήψη αυτού του αρχείου
- Υπολογισμός κάποιων μεταδεδομένων πάνω στο αρχείο (min, max), πρόσθεση σαν μεταδεδομένα, και ανέβασμα του αρχείου στο minIO μέσω ροής

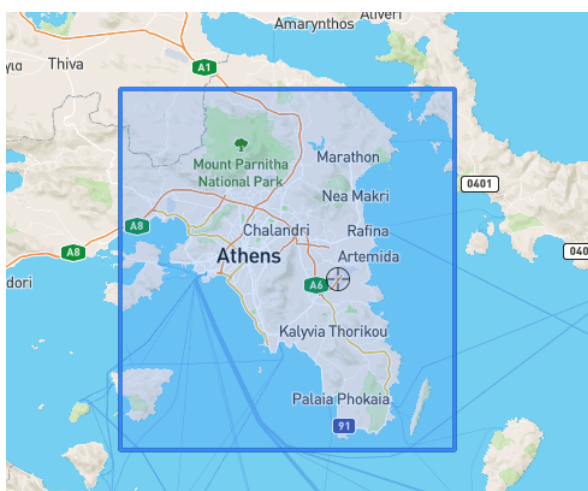
Εισαγωγή

Για την υλοποίηση του δοθέντος σεναρίου χρησιμοποιήθηκαν τα ανοιχτά δεδομένα που παρέχονται από την διεύθυνση <https://opensky-network.org/>. Αφού ανακτηθούν τα δεδομένα μέσω του Node-RED στη συνέχεια αποθηκεύονται τοπικά. Έπειτα εντοπίζονται η μέγιστη και η ελάχιστη ταχύτητα πλεύσης καθώς και ο συνολικός αριθμός των αεροσκαφών που περιλαμβάνει το μήνυμα. Στη συνέχεια τα υπολογιζόμενα στοιχεία εισάγονται σαν μεταδεδομένα και αποθηκεύονται μαζί με τα δεδομένα σε .json αρχείο στο minIO.

Opensky

Η διεύθυνση <https://opensky-network.org/> δίνει πρόσβαση σε δεδομένα πραγματικού χρόνου που αφορούν την εναέρια κυκλοφορία αεροσκαφών. Για την λήψη των δεδομένων γίνεται χρήση της υπηρεσίας [REST API](#) την οποία παρέχει. Ανατρέχοντας στην βιβλιογραφία εντοπίζεται η μέθοδος της κλήσης που θα πρέπει να πραγματοποιηθεί καθώς και οι παράμετροι που πρέπει να δοθούν. Για το σκοπό της εργασίας επιλέχθηκε μία σχετικά μικρή περιοχή του λεκανοπεδίου της Αττικής όπως φαίνεται σε σκούρο χρώμα στην εικόνα παρακάτω ενώ η κλήση που θα εκτελείται έχει την εξής μορφή:

<https://opensky-network.org/api/states/all?lamin=37.625109&lomin=23.442078&lamax=38.274845&lomax=24.205627>



Ανατρέχοντας ξανά στην βιβλιογραφία εντοπίζεται το επιστρεφόμενο μήνυμα, το οποίο είναι ένα json object με τα εξής πεδία:

Property	Type	Description
<i>time</i>	integer	The time which the state vectors in this response are associated with. All vectors represent the state of a vehicle with the interval [time-1,time]
<i>states</i>	array	The state vectors.

Το πεδίο *states* είναι ένα array δύο διαστάσεων που περιλαμβάνει με τη σειρά του τα εξής πεδία:

Index	Property	Type	Description
0	<i>icao24</i>	string	Unique ICAO 24-bit address of the transponder in hex string representation.
1	<i>callsign</i>	string	Callsign of the vehicle (8 chars). Can be null if no callsign has been received.
2	<i>origin_country</i>	string	Country name inferred from the ICAO 24-bit address.
3	<i>time_position</i>	int	Unix timestamp (seconds) for the last position

			update. Can be null if no position report was received by OpenSky within the past 15s.
4	<i>last_contact</i>	int	Unix timestamp (seconds) for the last update in general. This field is updated for any new, valid message received from the transponder.
5	<i>longitude</i>	float	WGS-84 longitude in decimal degrees. Can be null.
6	<i>latitude</i>	float	WGS-84 latitude in decimal degrees. Can be null.
7	<i>baro_altitude</i>	float	Barometric altitude in meters. Can be null.
8	<i>on_ground</i>	boolean	Boolean value which indicates if the position was retrieved from a surface position report.
9	<i>velocity</i>	float	Velocity over ground in m/s.
10	<i>true_track</i>	float	True track in decimal degrees clockwise from

			north (north=0°). Can be null.
11	<i>vertical_rate</i>	float	Vertical rate in m/s. A positive value indicates that the airplane is climbing, a negative value indicates that it descends. Can be null.
12	<i>sensors</i>	int[]	IDs of the receivers which contributed to this state vector. Is null if no filtering for sensor was used in the request.
13	<i>geo_altitude</i>	float	Geometric altitude in meters. Can be null.
14	<i>squawk</i>	string	The transponder code aka Squawk. Can be null.
15	<i>spi</i>	boolean	Whether flight status indicates special purpose indicator.
16	<i>position_source</i>	int	Origin of this state's position.
17	<i>category</i>	int	Aircraft category.

Το πεδίο ενδιαφέροντος είναι το εννιά (9) που μας δίνει την ταχύτητα εδάφους του σχετικού αεροσκάφους. Με βάση αυτό εκτελούνται οι υπολογισμοί των μεταδεδομένων.

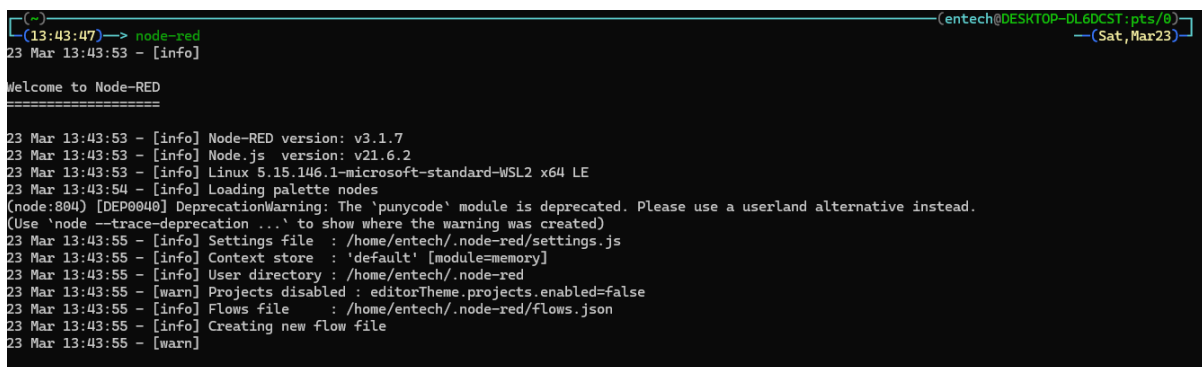
Αρχική Εγκατάσταση

Για την υλοποίηση της εργασίας χρησιμοποιήθηκε λειτουργικό σύστημα ubuntu. Τα βήματα που πραγματοποιήθηκαν είναι τα εξής:

- ☐ Αρχική εγκατάσταση του Node-Red τοπικά σε υπολογιστή ακολουθώντας τις οδηγίες που βρίσκονται [εδώ](#).
- ☐ Αρχική εγκατάσταση του minIO τοπικά σε υπολογιστή ακολουθώντας τις οδηγίες που βρίσκονται [εδώ](#).

Υλοποίηση

Node-Red



```
(entech@DESKTOP-DL6DCST:pts/0)
(13:43:47) -> node-red
23 Mar 13:43:53 - [info]
Welcome to Node-RED
=====
23 Mar 13:43:53 - [info] Node-RED version: v3.1.7
23 Mar 13:43:53 - [info] Node.js version: v21.6.2
23 Mar 13:43:53 - [info] Linux 5.15.146.1-microsoft-standard-WSL2 x64 LE
23 Mar 13:43:54 - [info] Loading palette nodes
(node:894) [DEP0049] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use 'node --trace-deprecation ...' to show where the warning was created)
23 Mar 13:43:55 - [info] Settings file : /home/entech/.node-red/settings.js
23 Mar 13:43:55 - [info] Context store : 'default' [module=memory]
23 Mar 13:43:55 - [info] User directory : /home/entech/.node-red
23 Mar 13:43:55 - [warn] Projects disabled : editorTheme.projects.enabled=false
23 Mar 13:43:55 - [info] Flows file : /home/entech/.node-red/flows.json
23 Mar 13:43:55 - [info] Creating new flow file
23 Mar 13:43:55 - [warn]
```

- ☐ Άνοιγμα ενός terminal και εκκίνηση του server του node-red.
- ☐ Η υπηρεσία παρέχεται στη διεύθυνση <http://127.0.0.1:1880/>
- ☐ Στο μονοπάτι που εκτελείται το node-red πρέπει να δημιουργηθεί ένας φάκελος με την ονομασία data όπου θα αποθηκεύονται τα δεδομένα.

minIO

- ☐ Στη συνέχεια σε νέο terminal εκκινείται η υπηρεσία του minIO.
- ☐ Η υπηρεσία API παρέχεται στη διεύθυνση <http://192.168.185.178:9000>
- ☐ Η υπηρεσία WebUI παρέχεται στη διεύθυνση <http://192.168.185.178:9001>


```
(entech@DESKTOP-DL6DCST:pts/2)
(15:02:46) -> ./minio server ./data --console-address ":9001"
MinIO Object Storage Server
Copyright: 2015-2024 MinIO, Inc.
License: GNU AGPLv3 <https://www.gnu.org/licenses/agpl-3.0.html>
Version: RELEASE.2024-03-15T01-07-19Z (go1.21.8 linux/amd64)

API: http://192.168.185.178:9000 http://127.0.0.1:9000
RootUser: minioadmin
RootPass: minioadmin

WebUI: http://192.168.185.178:9001 http://127.0.0.1:9001
RootUser: minioadmin
RootPass: minioadmin

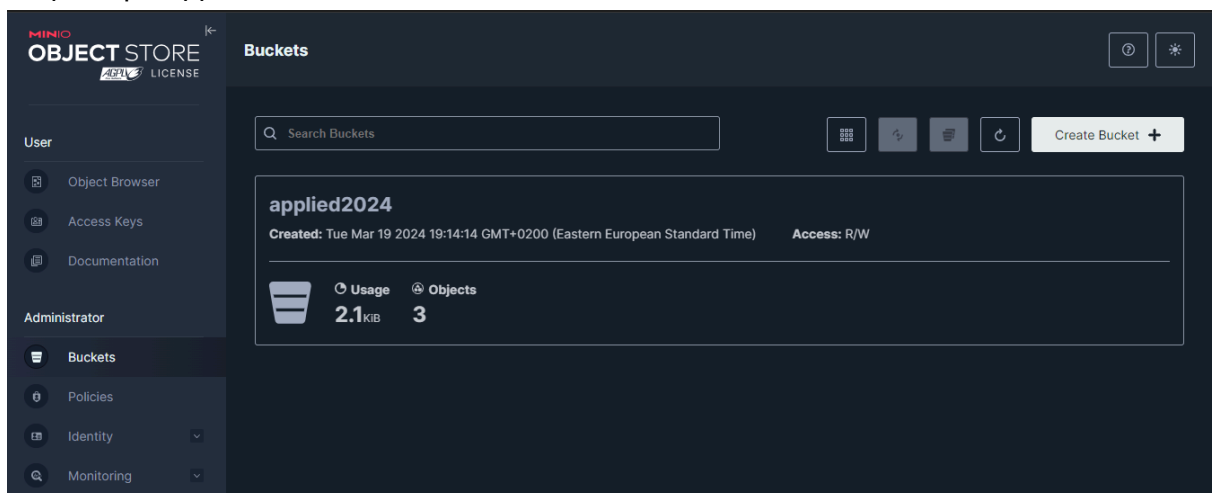
CLI: https://min.io/docs/minio/linux/reference/minio-mc.html#quickstart
$ mc alias set 'myminio' 'http://192.168.185.178:9000' 'minioadmin' 'minioadmin'

Docs: https://min.io/docs/minio/linux/index.html
Status: 1 Online, 0 Offline.

STARTUP WARNINGS:
- Detected default credentials 'minioadmin:minioadmin', we recommend that you change these values with 'MINIO_ROOT_USER' and 'MINIO_ROOT_PASSWORD' e
nvironment variables
- The standard parity is set to 0. This can lead to data loss.

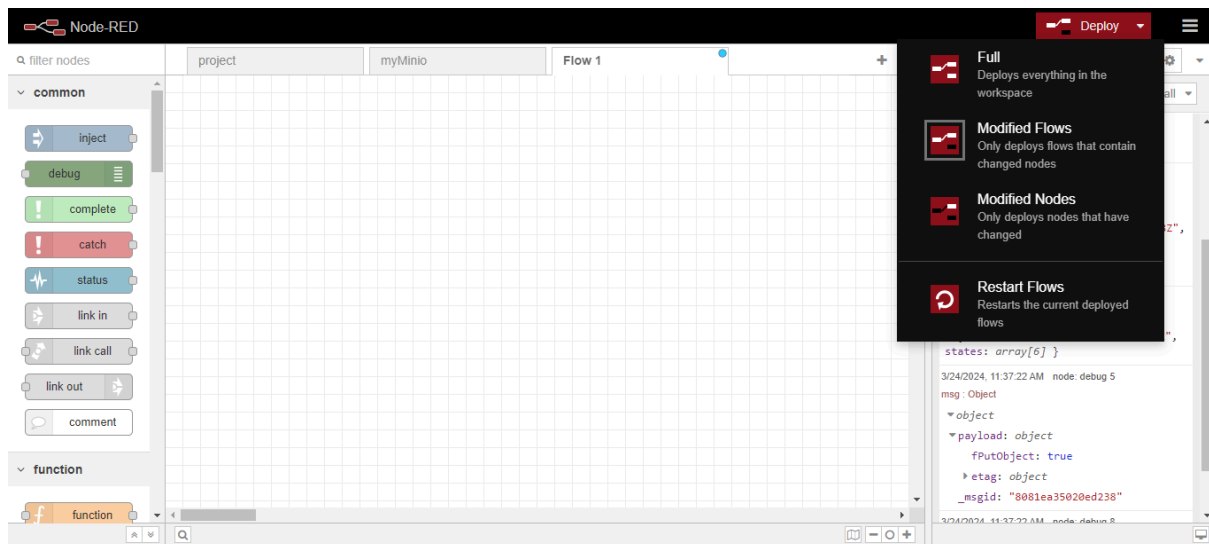
You are running an older version of MinIO released 6 days before the latest release
Update: Run 'mc admin update ALIAS'
```

- ☐ Αφού γίνει επίσκεψη στη διεύθυνση του WebUI, πραγματοποιείται είσοδος με τη χρήση των κωδικών minioadmin για username και password.
- ☐ Πρέπει να δημιουργηθεί ένα νέο bucket στο minIO όπου για τη συγκεκριμένη εργασία ονομάστηκε applied2024.

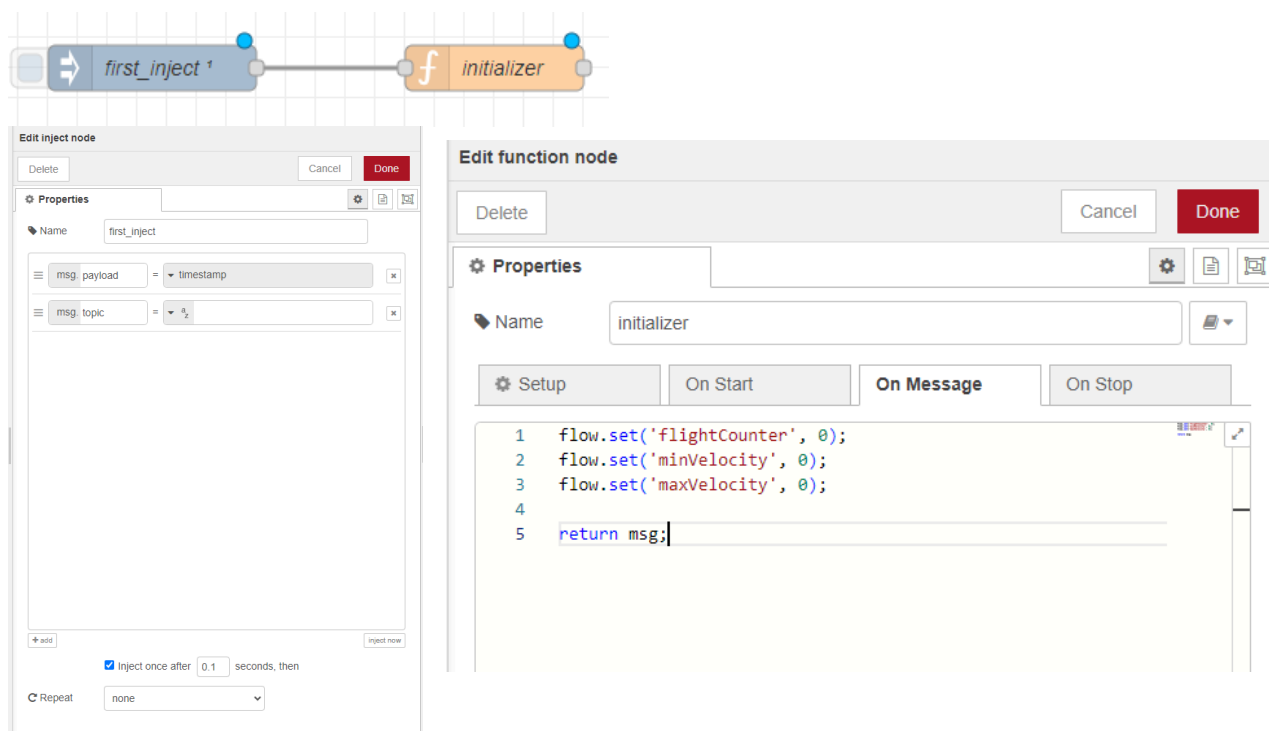


Σχεδίαση

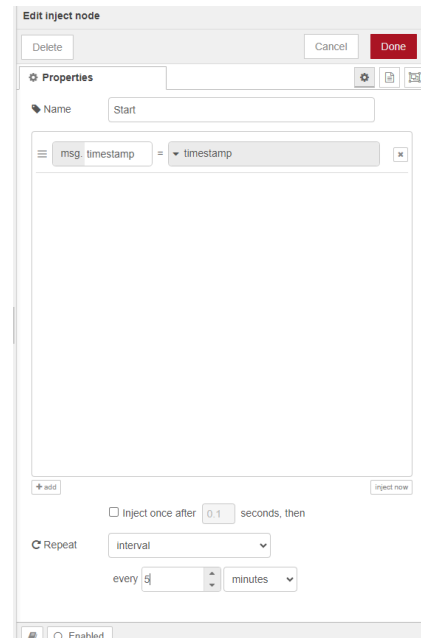
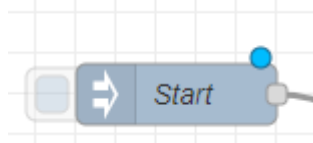
- ☐ Αρχικά γίνεται επίσκεψη με τον browser στη διεύθυνση του node-red και επιλέγεται από το μενού επιλογών το Modified Flows:



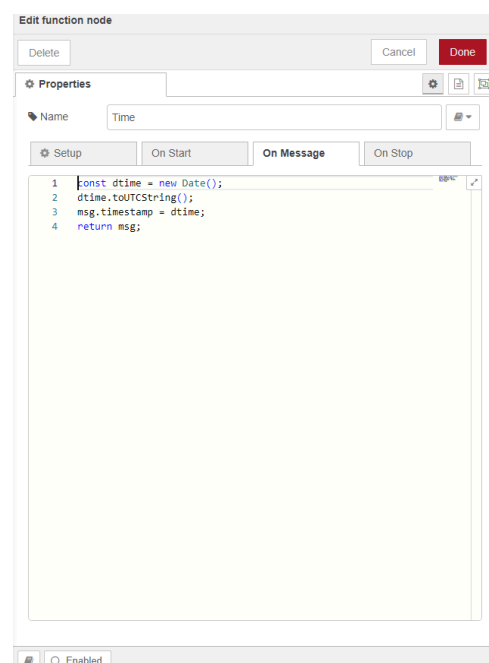
- ☐ Στη συνέχεια χρειάζεται η εγκατάσταση από το μενού palette των εξής βιβλιοθηκών:
 - ☐ node-red-contrib-minio-all.
- ☐ Έπειτα ακολουθεί η δημιουργία ενός νέου flow με ονομασία "Report".
- ☐ Αρχικά εισάγεται ένας κόμβος τύπου "inject" ο οποίος ονομάζεται first_inject, ρυθμίζεται να εκτελεί inject σε συγκεκριμένο χρονικό διάστημα και συνδέεται με μία συνάρτηση με την ονομασία initializer η οποία αρχικοποιεί τα μεταδεδομένα που επιθυμεί ο χρήστης να υπολογίσει (minVelocity, maxVelocity και flightCounter).



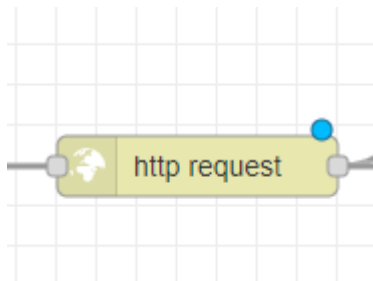
- Εισάγεται ένας κόμβος τύπου “inject” και ονομάζεται “Start” με επιλογή να επαναλαμβάνεται ανα πεντε (5) λεπτά, ο χρόνος αυτός καθορίστηκε για τη σχετική εργασία προκειμένου να αναδειχθούν τα επιθυμητά αποτελέσματα και μπορεί να τροποποιηθεί αναλόγως των αναγκών.



- Με μία συνάρτηση που ονομάζεται Time μορφοποιούνται τα δεδομένα του χρόνου που στέλνονται από τον κόμβο inject-Start προκειμένου να γίνεται εύκολη ανάγνωση.



- ☐ Τοποθετείται ένας κόμβος τύπου http request για την κλήση των δεδομένων και δίνονται τα κατάλληλα στοιχεία.



Edit http request node

Delete Cancel Done

Properties

Method GET

URL <https://opensky-network.org/api/states/all?lamin=3>

Payload Send as request body

☐ Enable secure (SSL/TLS) connection

☐ Use authentication

☐ Enable connection keep-alive

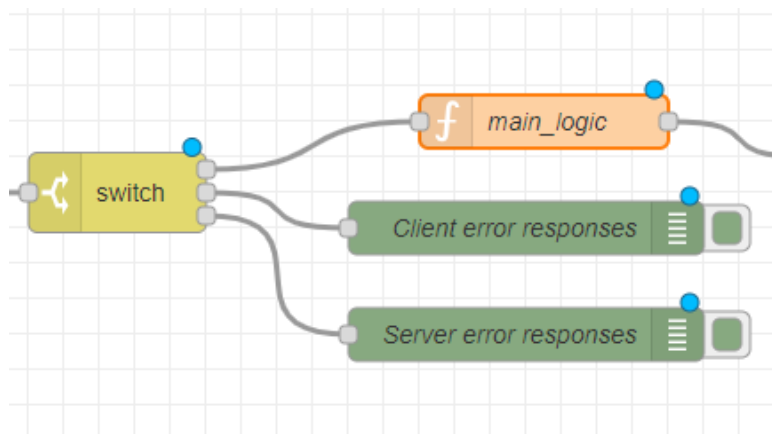
☐ Use proxy

☐ Only send non-2xx responses to Catch node

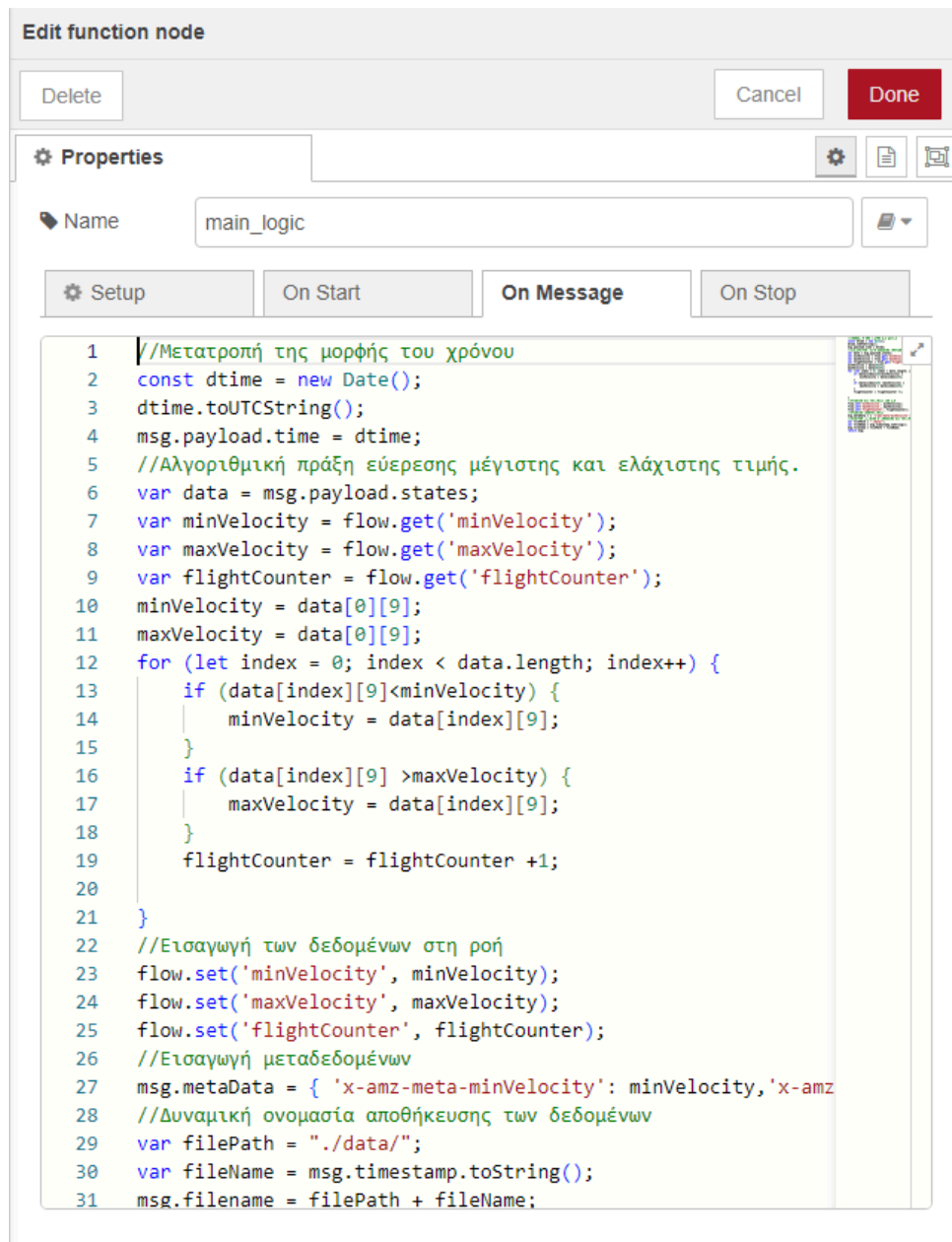
☐ Disable strict HTTP parsing

Return a parsed JSON object

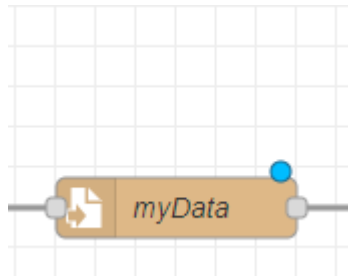
- ☐ Έπειτα τοποθετείται ένας κόμβος τύπου switch με σκοπό όταν καλείται το API να ενημερώνει για τον http response code και να συνεχίζει εφόσον η κλήση είναι επιτυχής. Για λόγους απλότητας σε περίπτωση μή επιτυχημένης κλήσης θα επιστρέφεται η απάντηση σε debug node.



- ☐ Με την επιτυχή εκτέλεση της κλήσης (response code:200) καλείται η συνάρτηση main_logic. Η συνάρτηση αυτή συνοπτικά:
 - ☐ Εκτελεί υπολογισμούς πάνω στο epoch time με σκοπό να το τροποποιήσει σε μορφή πιο φιλική προς τον άνθρωπο.
 - ☐ Με την εκτέλεση αλγοριθμικής πράξης εντοπίζει την ελάχιστη και τη μέγιστη τιμή ενώ υπολογίζει και το συνολικό πλήθος των αεροσκαφών του μηνύματος.
 - ☐ Αποθηκεύει τα δεδομένα αυτά σαν μεταδεδομένα.
 - ☐ Ενώ τέλος εκτελεί κάποια απαραίτητα βήματα ώστε τα δεδομένα να αποθηκεύονται στον φάκελο data που δημιουργήθηκε [προηγουμένως](#) με δυναμική ονομασία χρόνου και ημερομηνία απόκτησης.



- ☐ Με την εισαγωγή κόμβου τύπου “write file” που ονομάζεται “myData” αποθηκεύονται τα δεδομένα τοπικά.



Edit write file node

Delete Cancel Done

Properties

Filename

Action

☐ Add newline (\n) to each payload?

☒ Create directory if it doesn't exist?

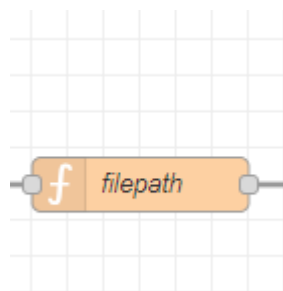
Encoding

Name

Tip: The filename should be an absolute path, otherwise it will be relative to the working directory of the Node-RED process.

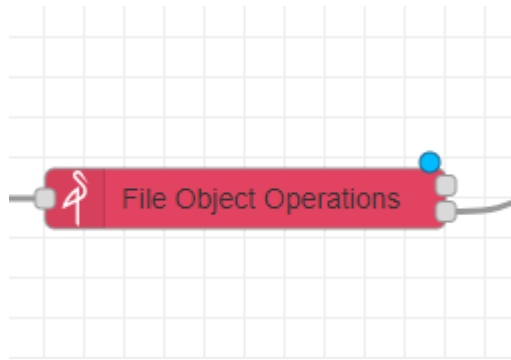
Enabled

- ☐ Η συνάρτηση “filepath” εξυπηρετεί στην εξαγωγή των επιθυμητών δεδομένων



προκειμένου να αποθηκεύονται τα δεδομένα στο minIO με την επιθυμητή ονομασία.

- ☐ Με την χρήση της βιβλιοθήκης minIO εισάγεται ένας κόμβος τύπου files με την ονομασία File Object Operations και προσδιορίζονται οι κατάλληλες παράμετροι.



Edit files node

Delete Cancel Done

Properties

Name: MinIO File Operations

MinIO Host: MinIO Instance

Operation: fPutObject

Bucket: applied2024

File: msg.objectName

Path: msg.filePath

MetaData: msg.metaData

Edit files node > Edit minio-config node

Delete Cancel Update

Properties

Name: MinIO Instance

MinIO Host: 192.168.185.178

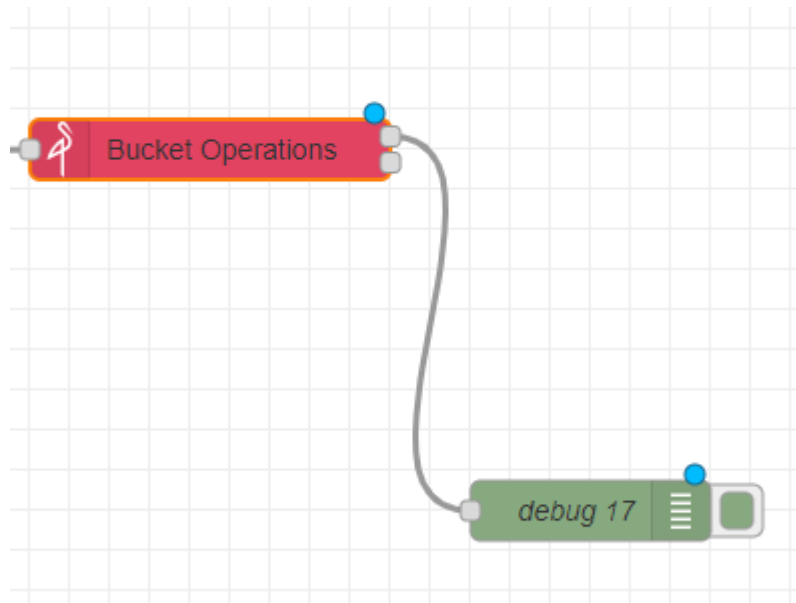
Port: 9000

Access Key: minioadmin

Secret Key:

Use SSL: ☐

- ☐ Τέλος εισάγεται ένας κόμβος τύπου “buckets” ο οποίος επιστρέφει τα μεταδεδομένα και εξυπηρετεί για λόγους επιβεβαίωσης της λειτουργίας.



Edit buckets node

Delete Cancel Done

Properties

Name MinIO Bucket Operations

MinIO Host MinIO Instance

Operation listObjectsV2WithMetadata

Bucket applied2024

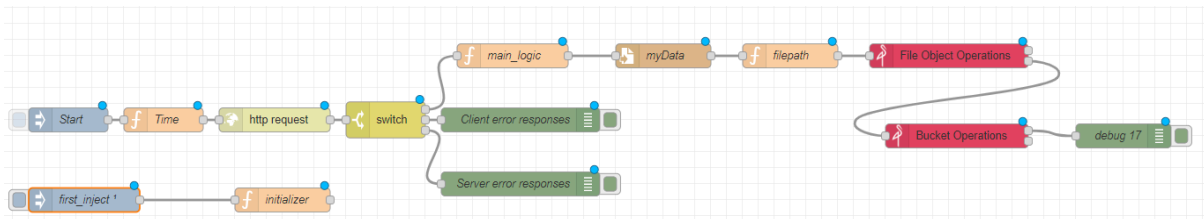
Prefix Prefix of objects to be listed (optional)

Recursive ☐

Start After List objects after this object name (optional)

Παρακάτω φαίνεται ολόκληρη η ροή

(Το json flow με τις αναλυτικές παραμέτρους επισυνάπτεται ενώ είναι διαθέσιμο και [εδώ](#))



Συμπεράσματα

Μετά την εκτέλεση της ροής παρατηρούμε τα εξής:














Τα αρχεία μας αποθηκεύονται τοπικά στον φάκελο data που δημιουργήσαμε

```
(~/node-red/data)
(21:13:53)→ ls -l
total 24
-rw-r--r-- 1 entech entech 1357 Mar 24 14:57 'Sun Mar 24 2024 14:57:28 GMT+0200 (Eastern European Standard Time)'
```



Επίσης τα αρχεία μας έχουν τα δεδομένα που μας παρέχει το API

```
GNU nano 6.2 Sun Mar 24 2024 15:12:00 GMT+0200 (Eastern European Standard Time)
{"time": "2024-03-24T13:12:01.196Z", "states": [{"46b825", "AEE2WD", "Greece", 1711285919, 1711285919, 24.0722, 38.0823, 929.64, false, 107.31, 216.81, -0.33, n
```

Στο bucket applied2024 έχουν προστεθεί τα αρχεία μας με βάση το χρονικό διαχωρισμό. Τα αρχεία μας είναι μορφής .json και αν δοκιμάσουμε να τα ανοίξουμε περιέχουν τα δεδομένα του payload.

	applied2024	Created on: Tue, Mar 19 2024 19:14:14 (GMT+2)	Access: PRIVATE	5.4 KiB			
		5 Objects					
		applied2024				Create new path 	
<input type="checkbox"/>	 Name	Last Modified		Size			
<input type="checkbox"/>	 Sun Mar 24 2024 14:57:28 GMT+0200 (Eastern Europea...	Today, 14:57		1.3 KiB			
<input type="checkbox"/>	 Sun Mar 24 2024 15:02:00 GMT+0200 (Eastern Europea...	Today, 15:02		825.0 B			
<input type="checkbox"/>	 Sun Mar 24 2024 15:07:00 GMT+0200 (Eastern Europea...	Today, 15:07		941.0 B			
<input type="checkbox"/>	 Sun Mar 24 2024 15:12:00 GMT+0200 (Eastern Europea...	Today, 15:12		1.2 KiB			
<input type="checkbox"/>	 Sun Mar 24 2024 15:17:00 GMT+0200 (Eastern Europea...	Today, 15:17		1.2 KiB			

Για σχετική επίδειξη του περιεχομένου απαιτείται τα αρχεία να κατέβουν τοπικά για το λόγο αυτό παρατίθεται το ανάλογο object info.

Object Info	
<hr/>	
Name:	Sun Mar 24 2024 15:07:00 GMT+0200 (Eastern European Standard Time).json
Size:	941.0 B
Last Modified:	3 days ago
ETAG:	102d30888ef064cc0a34b7b0b33e0efb
Tags:	N/A
Legal Hold:	Off
Retention Policy:	None
Metadata	
<hr/>	
Content-Type	application/octet-stream
X-Amz-Meta-Flightcounter	23
X-Amz-Meta-Maxvelocity	240.2
X-Amz-Meta-Minvelocity	52.52

Επιπλέον παρατηρούμε απο το debug console πως τα μεταδεδομένα μας επιστρέφονται σε κάθε νέα εισαγωγή π.χ. Τη στιγμή 3/24/2024, 3:12:01 PM το bucket μας περιέχει τέσσερα (4) αρχεία και ενδεικτικά απο το πρώτο αρχείο έχουμε τις καταγραφές δέκα (10) αεροσκαφών εκ των οποίων η μικρότερη ταχύτητα εδάφους είναι τα 55.05 m/s και η μεγαλύτερη 224.25 m/s.

3/24/2024, 3:12:01 PM node: debug 17

msg.payload : Object

▼ object

▼ listObjectsV2WithMetadata: array[4]

▼ 0: object

name: "Sun Mar 24 2024 14:57:28 GMT+0200 (Eastern European Standard Time).json"

lastModified: "2024-03-24T12:57:28.796Z"

etag: "b0d287eab727d0daa45fea96f410e8fa"

size: 1357

▼ metadata: object

content-type: "application/octet-stream"

X-Amz-Meta-Minvelocity: 55.05

X-Amz-Meta-Maxvelocity: 224.25

X-Amz-Meta-Flightcounter: 10

expires: "Mon, 01 Jan 0001 00:00:00 GMT"

▶ 1: object

▶ 2: object

▶ 3: object