

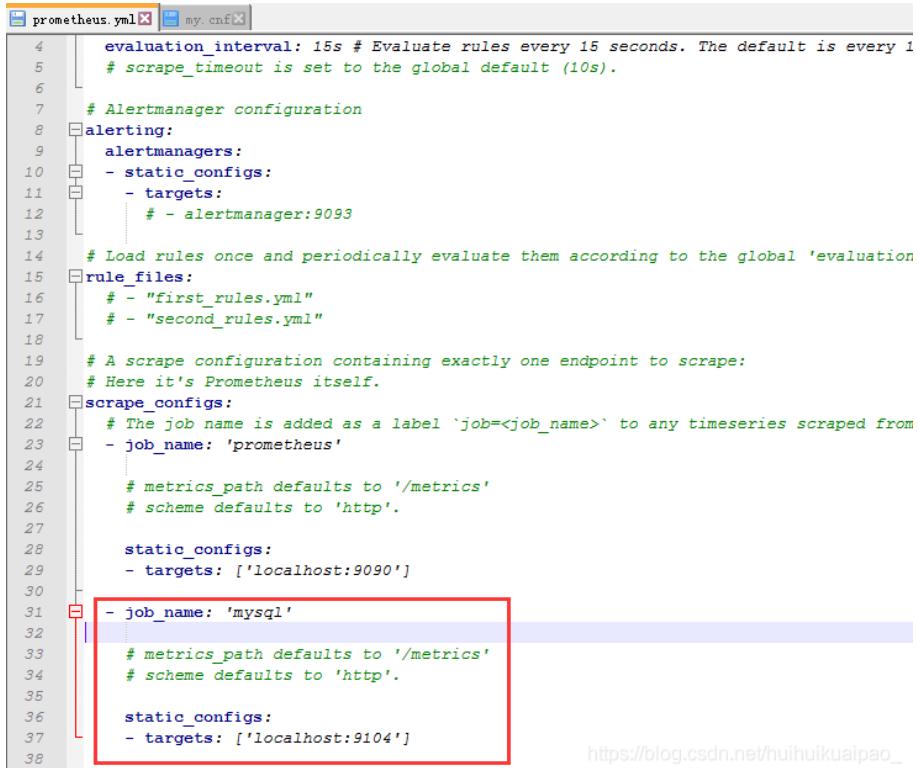
0519

事前工作: 安裝 MySQL-Exporter, 架起 MySQL 資料庫, 安裝 Grafana(demo 版本: 8.4.6)

MySQL-Exporter 操作步驟:

以下參照 [windows下prometheus+mysqld_exporter+grafana監控mysql](#)

1. prometheus.yml 內新增 mysql-exporter(default port: 9104)



```
4 evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1
5   # scrape_timeout is set to the global default (10s).
6
7   # Alertmanager configuration
8   alerting:
9     alertmanagers:
10       - static_configs:
11         - targets:
12           - alertmanager:9093
13
14   # Load rules once and periodically evaluate them according to the global 'evaluation
15   rule_files:
16     - "first_rules.yml"
17     - "second_rules.yml"
18
19   # A scrape configuration containing exactly one endpoint to scrape:
20   # Here it's Prometheus itself.
21   scrape_configs:
22     # The job name is added as a label 'job=<job_name>' to any timeseries scraped from
23     - job_name: 'prometheus'
24
25       # metrics_path defaults to '/metrics'
26       # scheme defaults to 'http'.
27
28       static_configs:
29         - targets: ['localhost:9090']
30
31     - job_name: 'mysql'
32
33       # metrics_path defaults to '/metrics'
34       # scheme defaults to 'http'.
35
36       static_configs:
37         - targets: ['localhost:9104']
38
```

https://blog.csdn.net/huihuikuaipao_

2. 執行 prometheus.exe
3. 開啟 Prometheus(default port: 9090)
4. mysql-exporter 下建立 my.cnf, 新增內容如下(對應數值需根據 MySQL 資料庫更改):

```
#```
[client]
host:127.0.0.1
```

port:3306

user=root

password=kebwImbhee

#````

(備註: 記得先在電腦啟動mySql 可以用mySql workbench)

5. 通過配置文件執行 mysqld_exporter.exe --config.my-cnf=my.cnf
6. 回到 Prometheus 下的 Targets 中確認 prometheus.yml 新增的 job 其 Status 是否正常(UP)

The screenshot shows the Prometheus Targets interface. At the top, there's a navigation bar with links for Prometheus, Alerts, Graph, Status, and Help. Below the navigation is a search bar labeled "Filter by endpoint or labels". A table lists a single target under the heading "mysql (1/1 up)". The columns are Endpoint, State, Labels, Last Scrape, Scrape Duration, and Error. The entry shows "http://localhost:9104/metrics" with "UP" status, "instance: \"localhost:9104\"", "job: \"mysql\"", "Last Scrape: 12.465s ago", "Scrape Duration: 46.856ms", and no errors.

若 Status 為 DOWN, MySQL 內下 query

```
CREATE USER 'exporter'@'localhost' IDENTIFIED BY 'XXXXXXXXXX' WITH
MAX_USER_CONNECTIONS 3;
GRANT PROCESS, REPLICATION CLIENT, SELECT ON *.* TO
'exporter'@'localhost';
```

回到 Graph 頁面, 搜尋 mysql, 應該會顯示出至少20筆的 promQL 可供查詢(少於10筆可能是配置時出現問題)

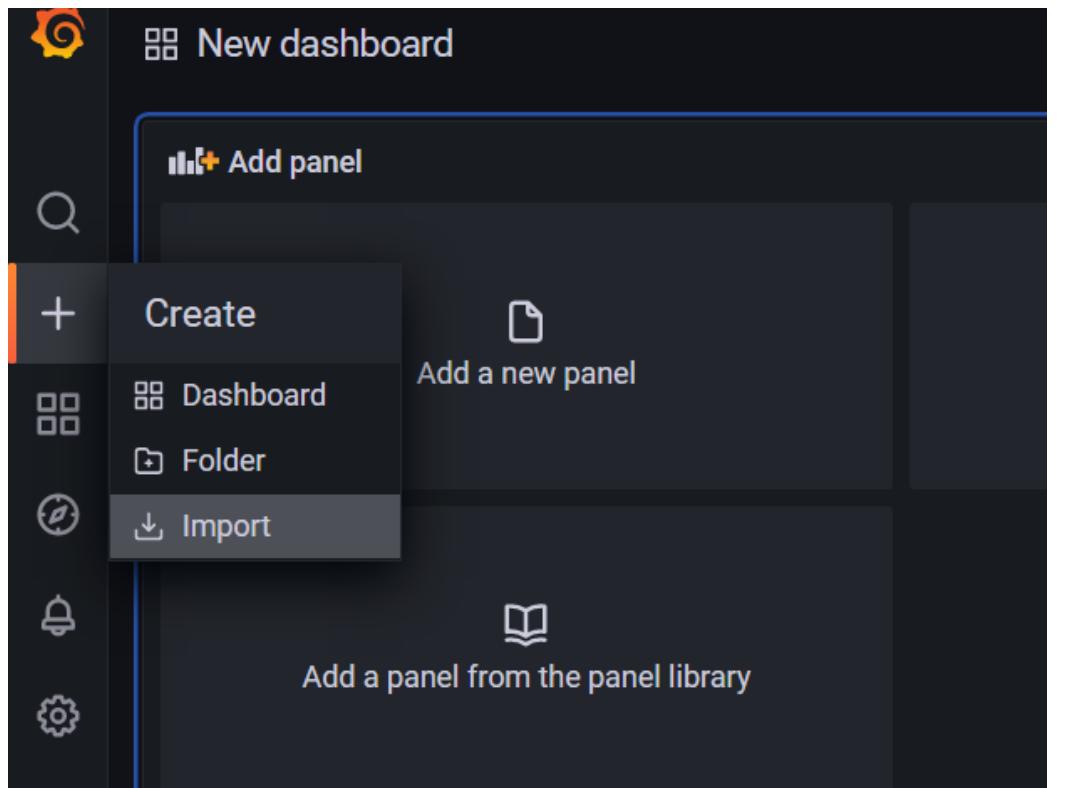
The screenshot shows the Prometheus Graph interface. At the top, there's a search bar with "mysql" and a table below it containing 20 rows of metrics. The table has columns for metric name, type, and description. Metrics listed include mysql_exporter_collector_duration_seconds (gauge), mysql_exporter_last_scrape_error (counter), mysql_exporter_scrapes_total (unknown), mysql_global_status_aborted_clients (unknown), mysql_global_status_aborted_connects (unknown), mysql_global_status_aborted_connects_preadauth (unknown), mysql_global_status_access_denied_errors (unknown), mysql_global_status_acl_column_grants (unknown), mysql_global_status_acl_database_grants (unknown), mysql_global_status_acl_function_grants (unknown), mysql_global_status_acl_package_body_grants (unknown), mysql_global_status_acl_package_spec_grants (unknown), mysql_global_status_acl_procedure_grants (unknown), mysql_global_status_acl_proxy_users (unknown), mysql_global_status_acl_role_grants (unknown), and mysql_global_status_acl_roles (unknown). A tooltip "Collector time duration." is visible over the first metric.

檢查 mysql-exporter (default port: 9104) 是否能正確查詢

The screenshot shows the Prometheus Metrics browser for the MySQL exporter. It displays a large amount of prometheus-generated Go code for various metrics. The code includes HELP and TYPE annotations for each metric, such as go_gc_cycles, go_gc_duration_seconds, and go_gc_heap_allocs_by_size_bytes_total.

7. 執行 grafana-server.exe (defalut position: C:\Program Files\GrafanaLabs\grafana\bin)
8. 開啟 Grafana(default port: 3000)

9. Create -> Import -> upload JSON file(下載 [MySQL Overview-1653661145398.json](#))
(0527更新)



10. Grafana -> Configuration -> Data Sources -> add Prometheus
11. Dashboards -> Browse -> check import dashboard -> double click to open

模版請下載以下檔案(已新增MySQL Status & Alert)之後進行 upload (參照 Step 9)
[MySQL Overview-1653661145398.json](#)

完成後的示意圖:

localhost_3000_d_7362_mysql-overview_orgId=1&refresh=5s.png

若使用cloud版本grafana可能無法正常使用data source query prometheus, 可以下載桌面版
windows: [windows版Grafana安装与配置- 简书](#)

其它 bug 解決方式: [Prometheus简单安装及问题记录 lee_fengyun的博客](#)

0526

Grafana 警報操作步驟:

以下參照: [Grafana 配置郵件告警](#)
[Configuration | Grafana documentation](#)

- 修改 C:\Program Files\GrafanaLabs\grafana\conf\defaults.ini(Win10) 下圖紅勾處
(默認套用 defaults.ini, 如果默認套用的配置文件有做更改, 可能需修改同路徑下的 custom.ini 或 sample.ini)
(Jay: 在我這邊測試是修改custom.ini才有效果)

google email SMTP 密碼設置參照下方連結, 經實測 port: 465 無效, port: 587 有效
[Google SMTP寄信\(兩段式驗證+應用程式密碼\)](#)

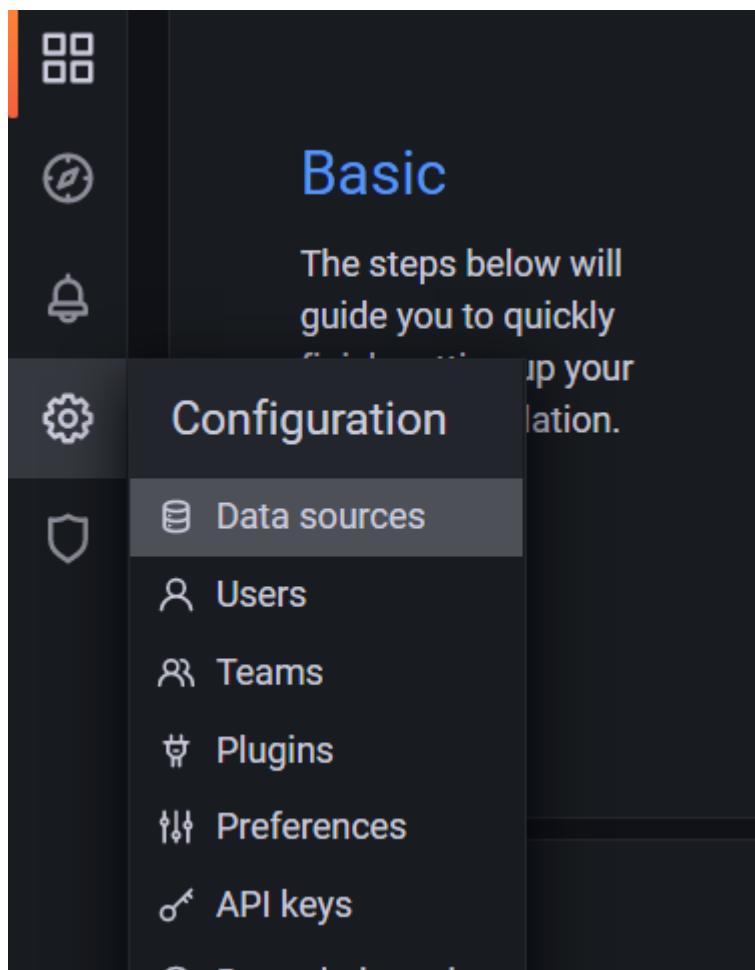
```
[smtp]
enabled = true
host = smtp.gmail.com:587
user = 108703017@g.nccu.edu.tw
# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""
password = [REDACTED]
cert_file =
key_file =
skip_verify = false
from_address = 108703017@g.nccu.edu.tw
from_name = Grafana
ehlo_identity =
startTLS_policy =
```

- 修改並儲存後開啟 Service -> Restart “Grafana” (用工作管理員操作無效)



4. 採用下圖配置(自訂 email)進行測試 (圖右) -> Save -> Test

6. Configuration -> Data Source -> Add data source -> Prometheus



7. double click Prometheus -> 輸入 URL

The screenshot shows the 'Data Sources / Prometheus' configuration page. At the top, there's a logo of a torch and the text 'Data Sources / Prometheus' followed by 'Type: Prometheus'. Below this, there are two tabs: 'Settings' (which is active) and 'Dashboards'. Under the 'Settings' tab, there's a table with four rows: 'Name' (set to 'Prometheus'), 'Default' (a toggle switch), 'HTTP' (with 'URL' set to 'http://127.0.0.1:9090'), and 'Timeout' (set to 'Timeout in seconds'). A red oval highlights the 'HTTP' section.

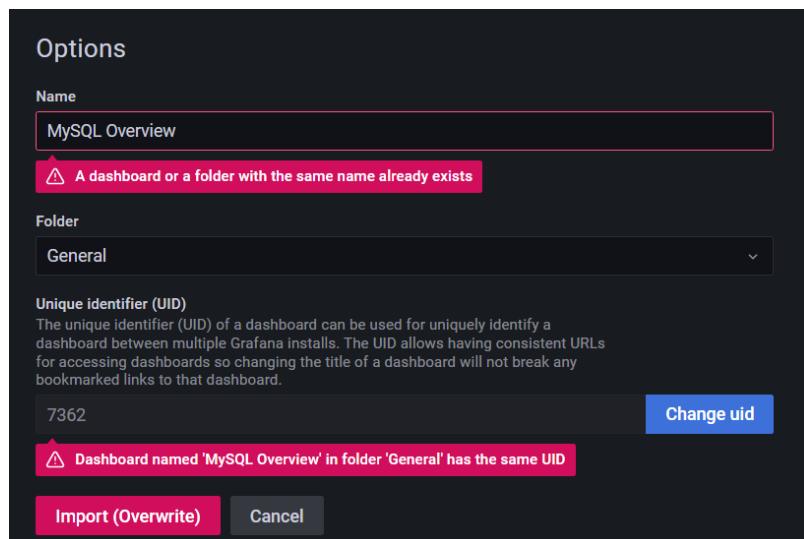
8. 以下配置參照:

[Alert and Monitoring with Grafana | by Hakan Eröztek | Trendyol Tech | Medium](#)

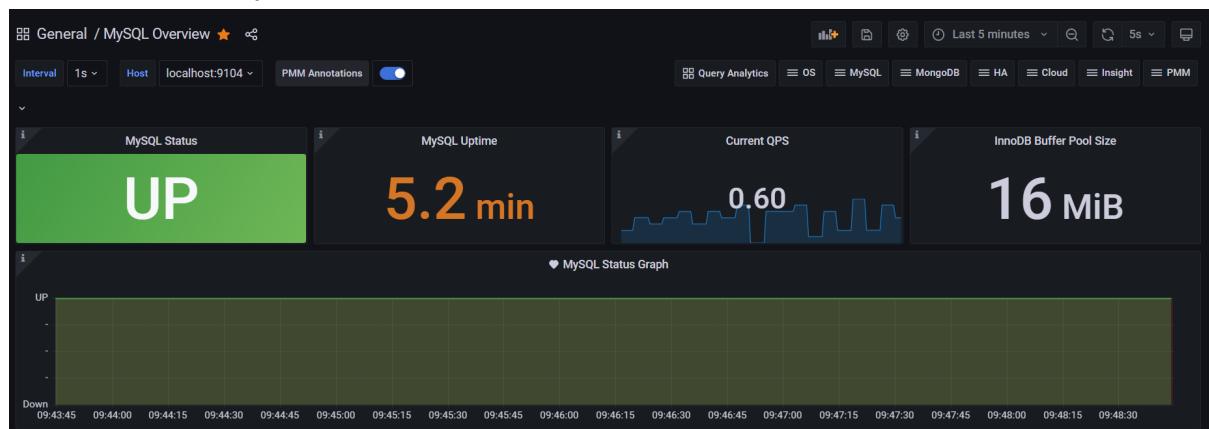
郵件測試完畢後，請重新確認 dashboard 是否如圖所示(0527 更新)

localhost_3000_d_7362_mysql-overview_orgId=1&refresh=5s.png

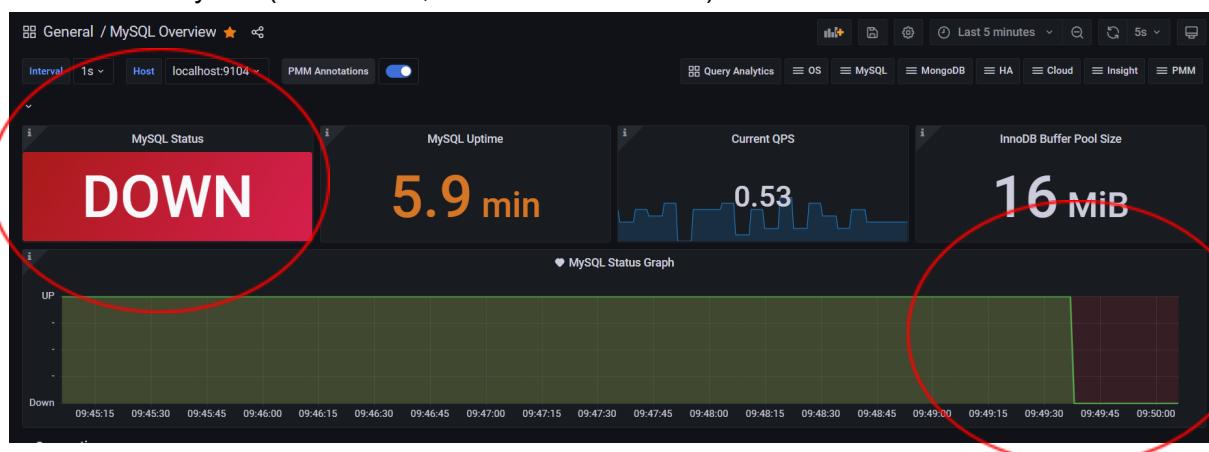
如果不是，請重新操作 0519 Step 9，選擇 Import (Overwrite)



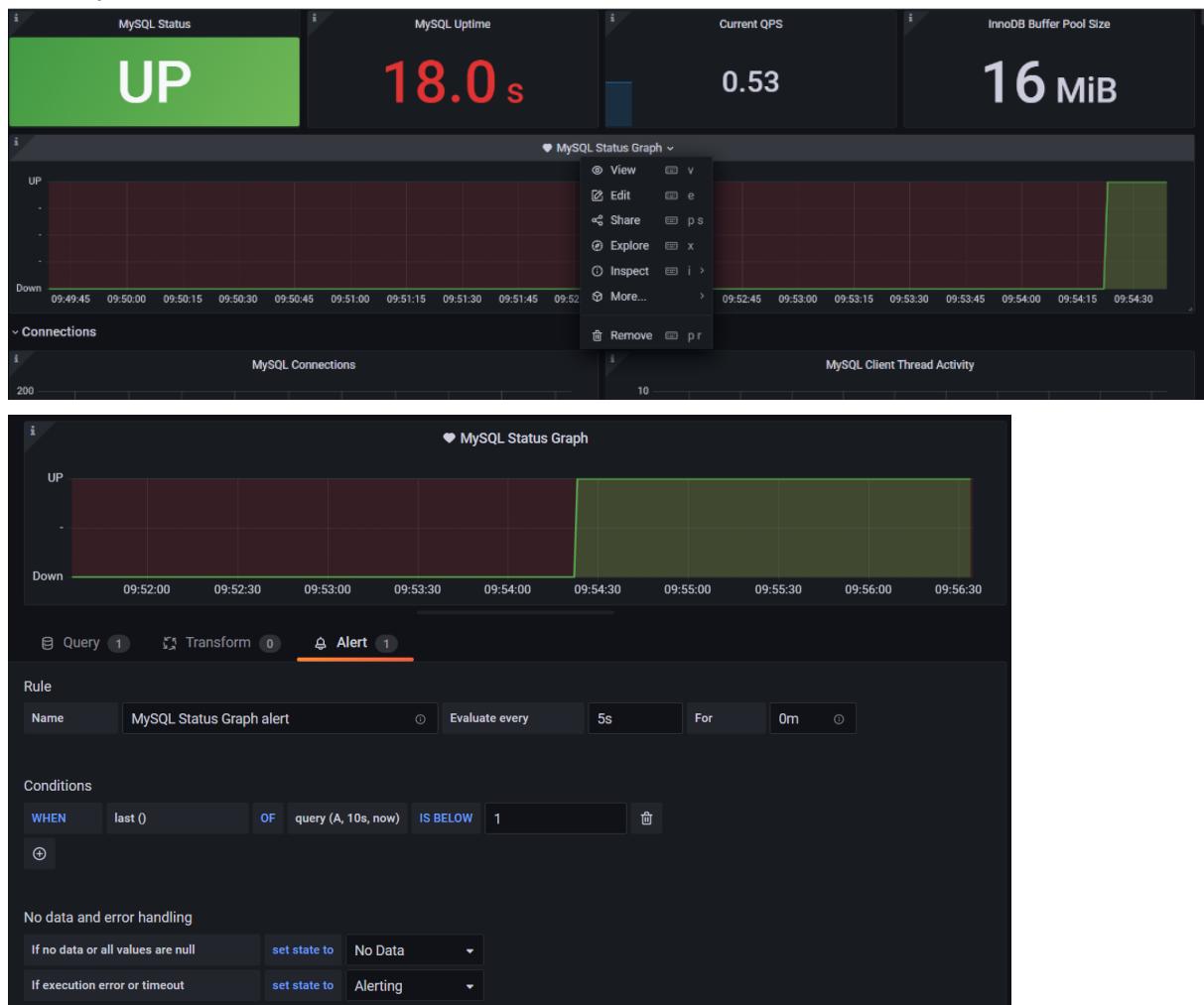
正常運行 MySQL:



關閉 MySQL(需等待 15s, dashboard 才會更新):

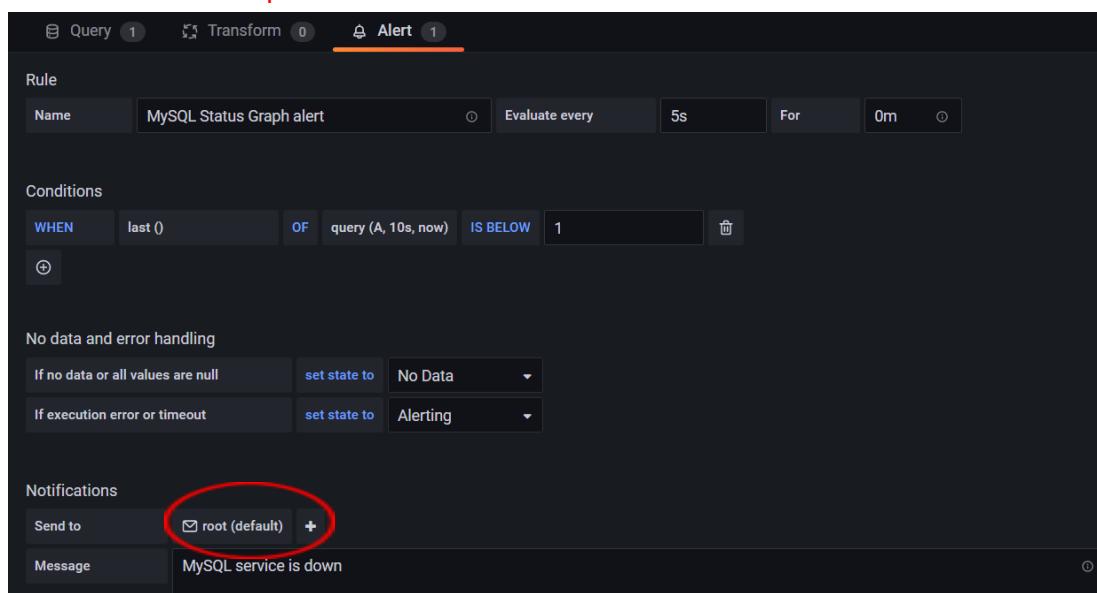


9. 點選 MySQL Status Graph -> Edit -> Alert

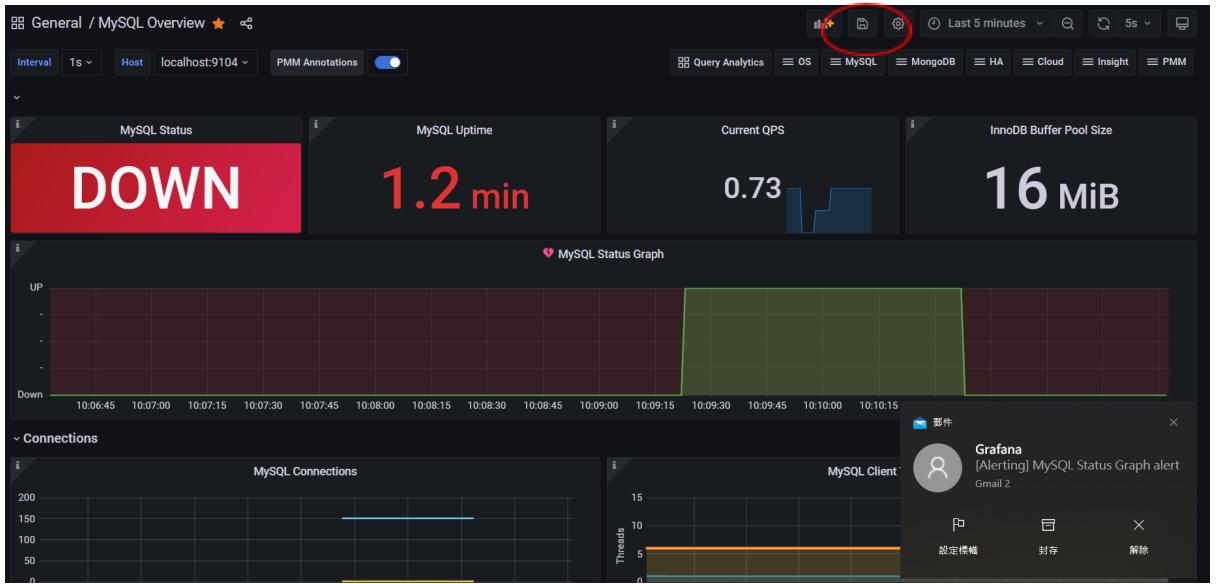


10. 如果有導入上方連結的 json file(參照 0519 Step 9), 可以看到已經有配置好一個 Alert (注意 Rule 下方的 Name 應已更名為 MySQL Status Alert)

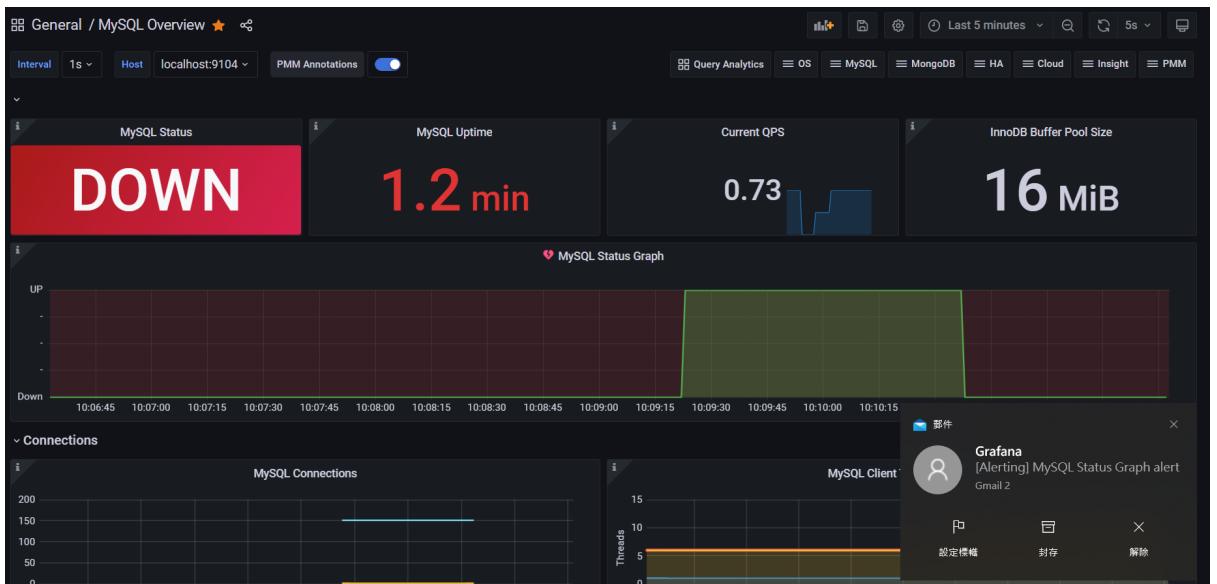
紅圈處自行選擇 Step 1~4 已經測試成功的配置



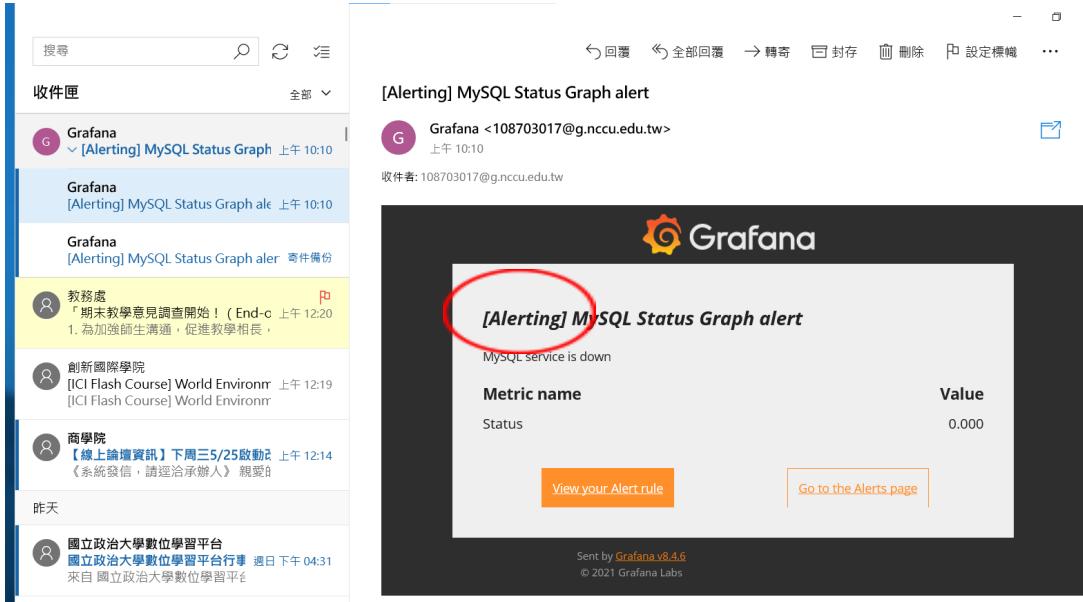
11. Save Dashboard



12. 關閉 MySQL 查看結果(若沒有出現郵件, 請跳至下方 Step 13 確認 Alert 是否開啟):



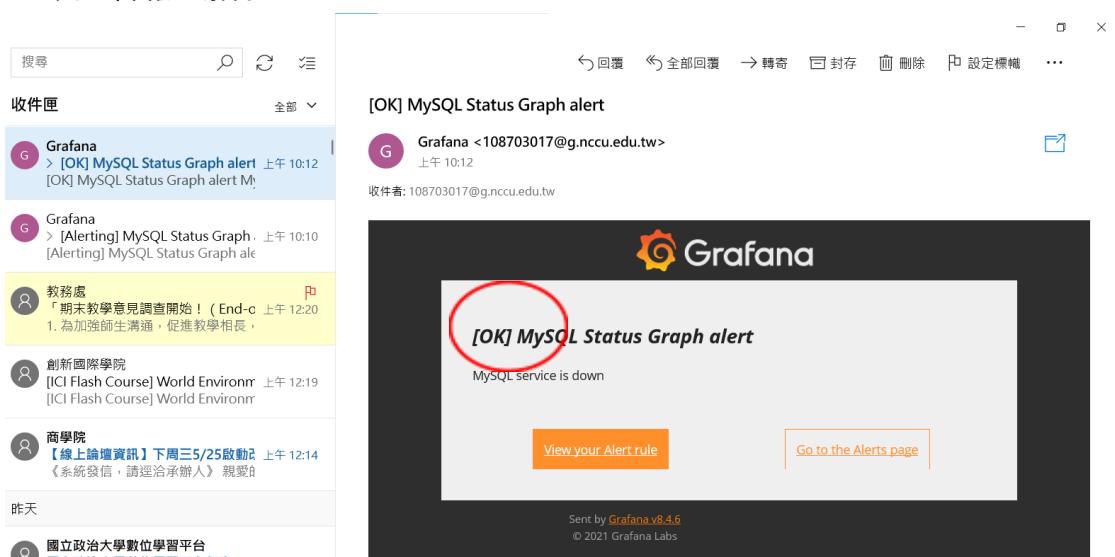
Alerting 表示警報偵測到條件觸發



啟動 MySQL 查看結果:



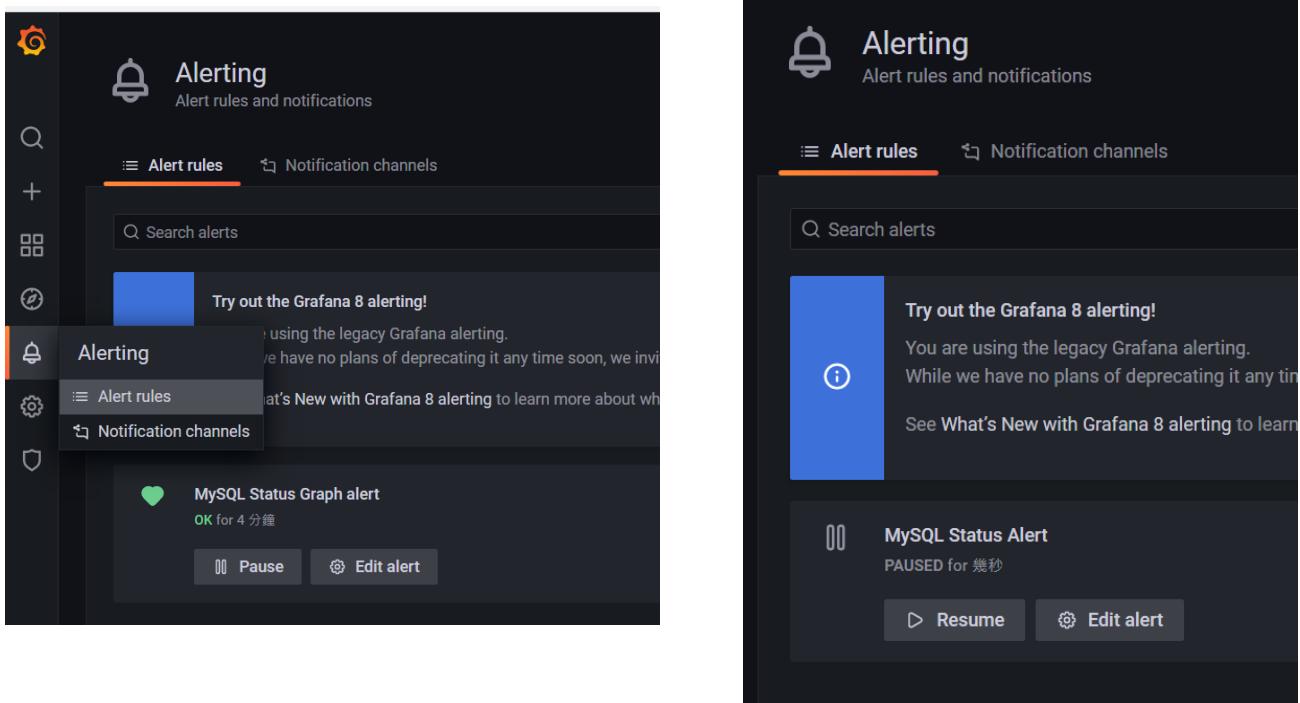
OK 表示警報已解除



13. 使用結束後務必先關閉 Alert 再關閉 MySQL & MySQL Exporter (不然你的信箱會被狂轟爛炸)

點擊 Alerting -> Alert rules -> MySQL Status Graph alert(已更名為 MySQL Status Alert) -> Pause (圖左)

下次使用時，先回到此頁面點擊 Resume (圖右)



===== Grafana 警報郵件 =====

MySQL 掛掉時 Prometheus 呈現 -> empty value when user query