

第八組 Prometheus

110753163 資碩工 林昱辰

110971029 資碩專 鄭安翔

108703017 資科三 邱彥翔

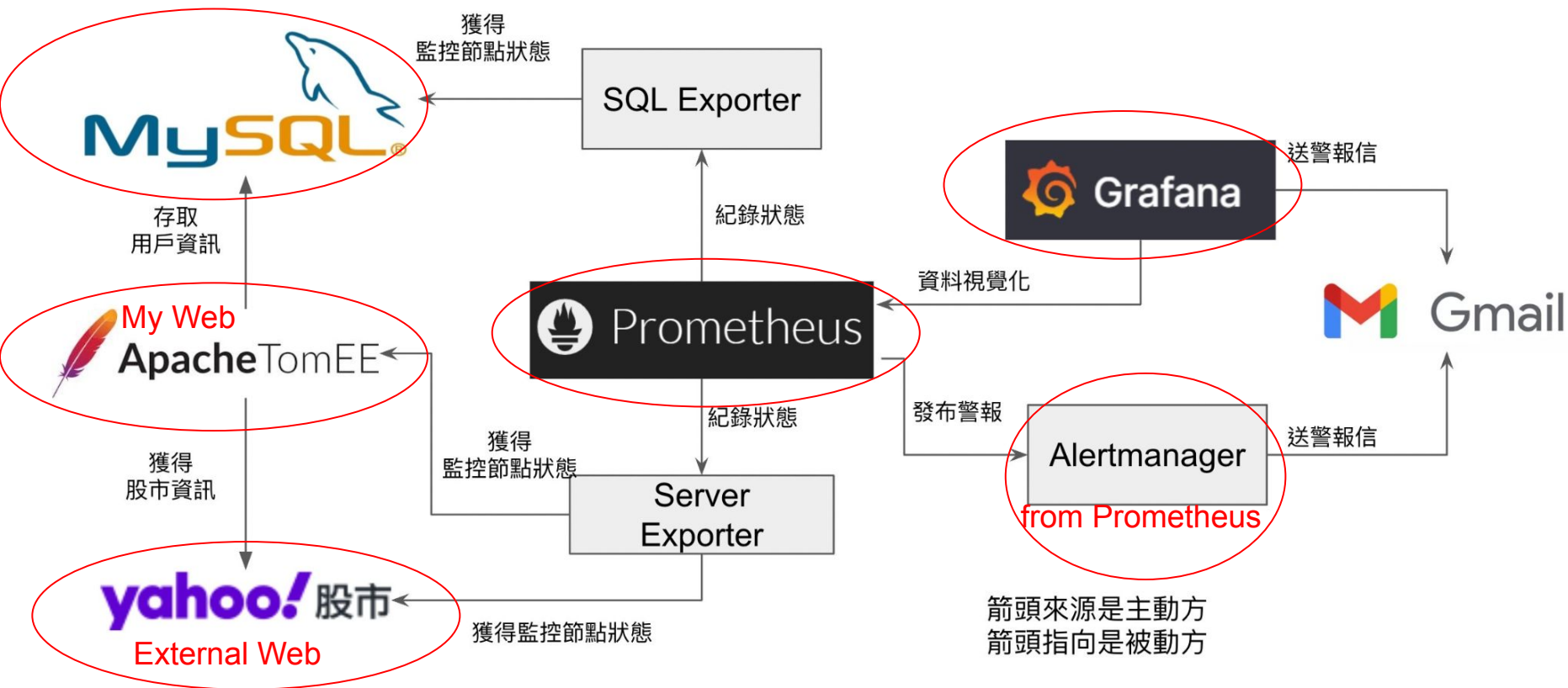
108703018 資科三 陳智詮

108703021 資科三 張誠宗

108703036 資科三 李佑晟

108703045 資科三 洪丞桀

架構圖



功能性 & 非功能性

功能性: 系統監控及郵件報警

非功能性: 在此系統中一分鐘最多約能獲取 3000 個 http_request 請求(如圖)



節點及功能

節點	功能
Prometheus	監控節點
MySQL	資料庫(作為監控節點)
Grafana	透過 Prometheus 處理 MySQL 視覺化及警報
Apache TomEE	My Web Server(作為監控節點)
Yahoo 股市	External Web Server(作為監控節點)
AlertManager	處理 Web Server 警報

分工

邱彥翔: 影片剪輯、MySQL-Exporter、Grafana、Github、
報告、投影片

鄭安翔: 測試網站、影片錄製、建立資料庫

陳智詮: server監控、alertmanager、Github

洪丞桀: MySQL-Exporter、Grafana、Github

李佑晟: server監控、alertmanager、Github

張誠宗: server監控、alertmanager

林昱辰: MySQL-Exporter、Grafana

影片: [1102 Distributed Systems 第八組 - Prometheus](#)

Github: https://github.com/kebwlmmbhee/1102_Dist_Sys



存取
用戶資訊



Apache TomEE

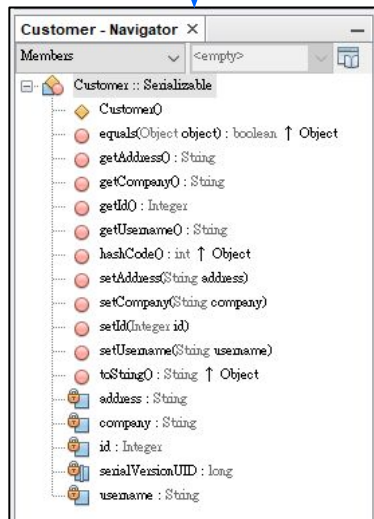
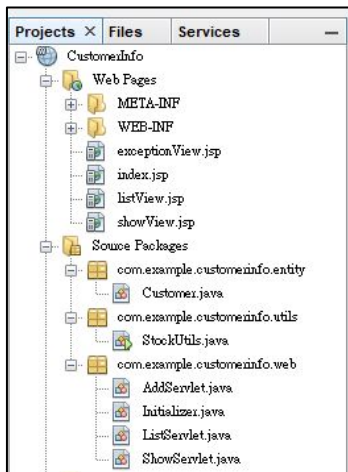
獲得
股市資訊

yahoo! 股市



#	id	username	address	company
1	1	王文淵	台北市士林區天母東路8號12樓	1301
2	2	王建民	台南市公園路30號	2891
3	3	陳金鋒	新北市永和路一段234號	2881
4	4	張忠謀	新竹市中正路100號	2330
5	5	郭台銘	台北市信義路五段200號	2317
6	6	彭政閔	台中市台灣大道三段168號	2891

#	CODE	NAME
1	1301	台塑
2	2317	鴻海
3	2330	台積電
4	2881	富邦金控
5	2891	中國信託



Customer List

客戶列表

姓名	地址	公司
王文淵	台北市士林區天母東路8號12樓	台塑
王建民	台南市公園路30號	中國信託
陳金鋒	新北市永和路一段234號	富邦金控
張忠謀	新竹市中正路100號	台積電
郭台銘	台北市信義路五段200號	鴻海
彭政閔	台中市台灣大道三段168號	中國信託



存取
用戶資訊



Apache TomEE

獲得
股市資訊

yahoo! 股市

```
9 public class StockUtils {
10     public String getStockPrice(String s) throws Exception {
11         String source="";
12         String urlString = "https://tw.stock.yahoo.com/quote/" + s;
13         URL url = new URL(urlString);
14         URLConnection conn = url.openConnection();
15         conn.setRequestProperty("User-agent", "IE/6.0");
16         InputStream is = conn.getInputStream();
17         Document doc = Jsoup.parse(is, "Big5", urlString);
18         String price = doc.select("span[class~=\"32px\"]").text().trim();
19         System.out.println(s + ":" + price);
20         is.close();
21         return price;
22     }
23 }
```



Database 無回應



存取
用戶資訊



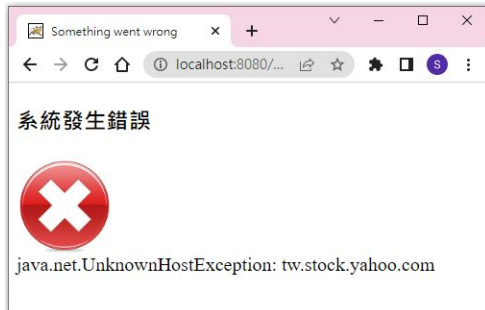
Apache TomEE

獲得
股市資訊

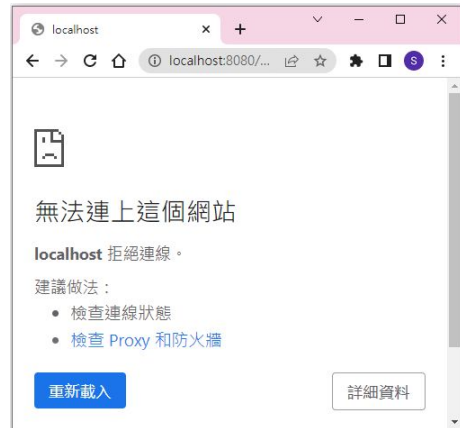
yahoo! 股市

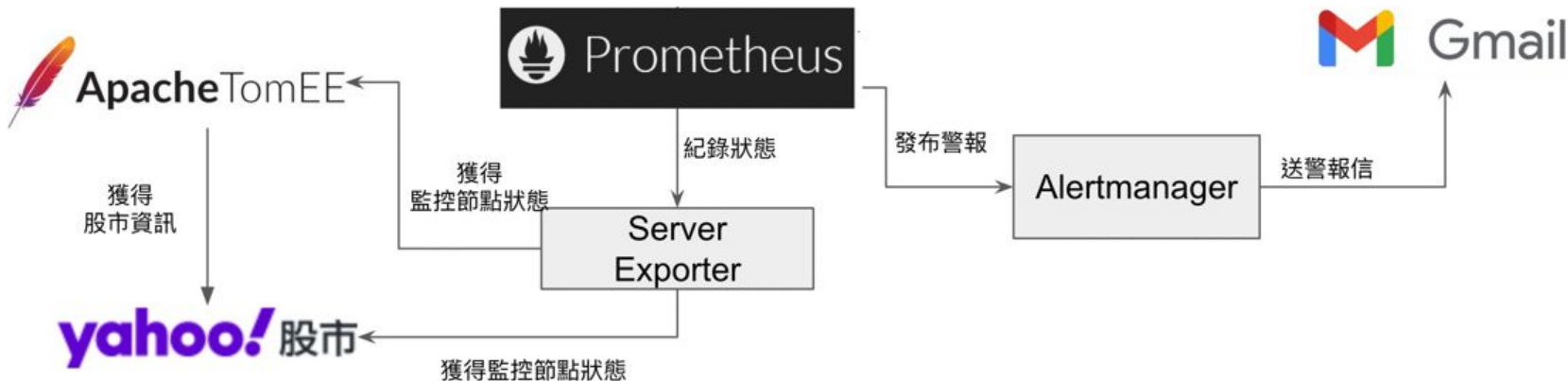


聯外網路中斷



Web Server 無回應





```
*server_exporter.py - E:\server_exporter.py (3.9.7)*
File Edit Format Run Options Window Help

1 from prometheus_client import start_http_server, Enum
2 import time
3 import platform
4 import socket
5 import subprocess
6
7 DEBUG = True
8 exporter_port = 8000 # exporter 使用的port
9 sleep_interval = 1 # seconds
10 my_web = ["127.0.0.1", 8080] # my web server的 ip and port
11 my_web_state = Enum("my_web_state", "Is my web alive", states=["success", "fail"])
12
13 external_web = "tw.stock.yahoo.com" # 目標server的domain name or ip
14 external_web_state = Enum("external_web_state", "Is the external web alive", states=["success", "fail"])
15 system = "windows" if platform.system().lower()=="windows" else "others"
16
17 # 透過socket建立與自己的網站的連線
18 def export_my_web_state(ip, port):
19     with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as sock:
20         sock.settimeout(0.5)
21         result = sock.connect_ex((ip, port))
22         if result == 0:
23             now = "success"
24         else:
25             now = "fail"
26
27     if DEBUG:
28         print("My web state: ", now)
29     my_web_state.state(now)
```

```
30
31 # 透過ping指令測試與目標外部網站的連線
32 def export_external_web_state(hostname):
33     cmd = ["ping", "-n" if system == "windows" else "-c", "1", hostname]
34     try:
35         response = subprocess.check_output(
36             cmd,
37             stderr=subprocess.STDOUT, # get all output
38             universal_newlines=True # return string not bytes
39         )
40     except subprocess.CalledProcessError:
41         response = None
42
43     if response != None:
44         now = "success"
45     else:
46         now = "fail"
47     if DEBUG:
48         print("External web state: ", now)
49     external_web_state.state(now)
50
51 if __name__ == '__main__':
52     # Start up the server to expose the metrics.
53     start_http_server(exporter_port)
54     # Generate some requests.
55     while True:
56         export_external_web_state(external_web)
57         export_my_web_state(*my_web)
58         time.sleep(sleep_interval)
59
60
```

```
(testEnv) E:\>python server_exporter.py
External web state: success
My web state: success
External web state: success
My web state: success
External web state: success
My web state: success
External web state: success
My web state: success
External web state: success
My web state: success
```



Apache TomEE

獲得
股市資訊

yahoo! 股市



Prometheus

紀錄狀態

發布警報



Gmail

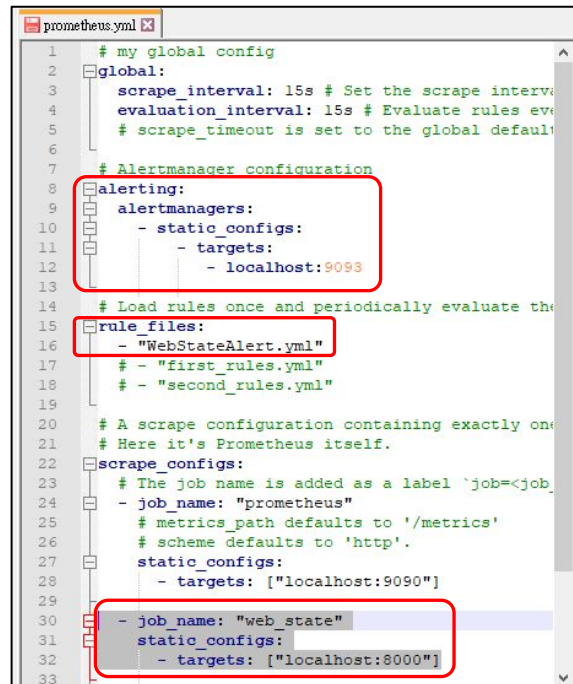
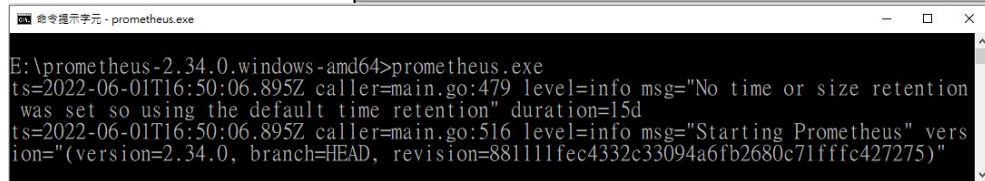
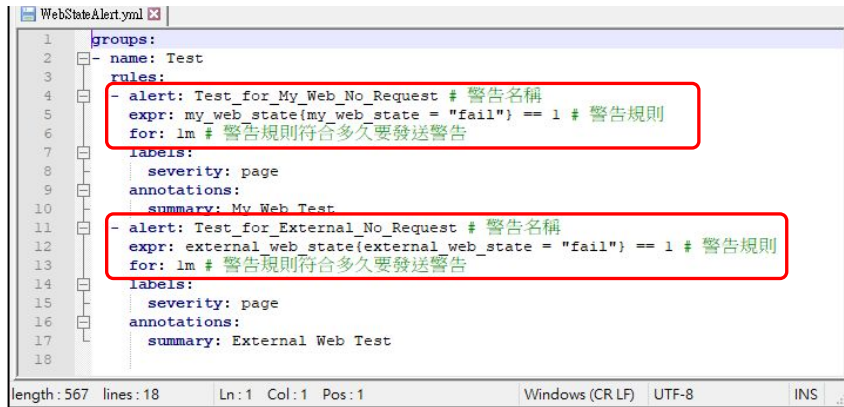
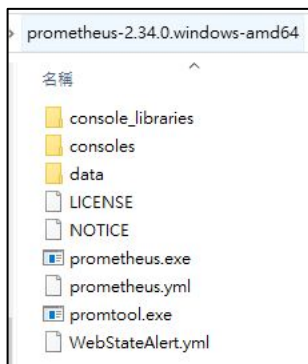
獲得
監控節點狀態

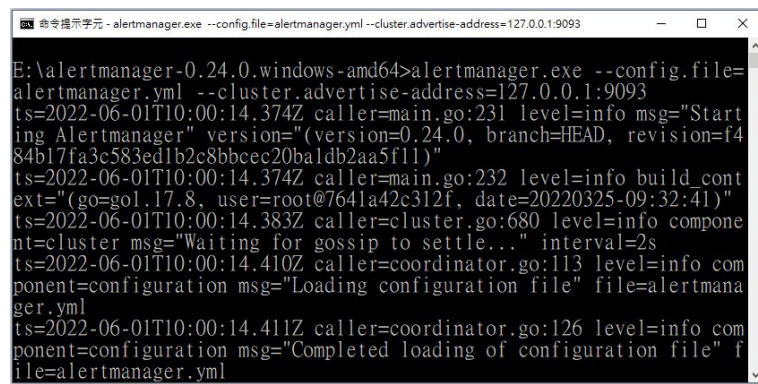
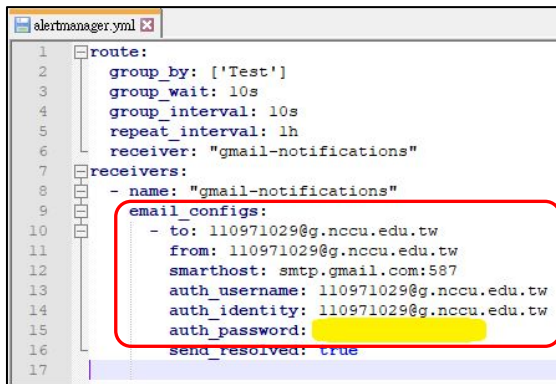
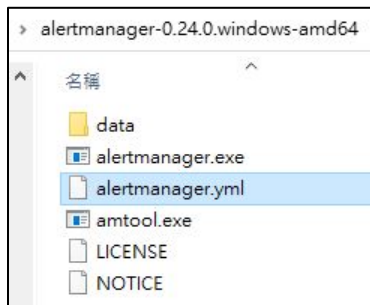
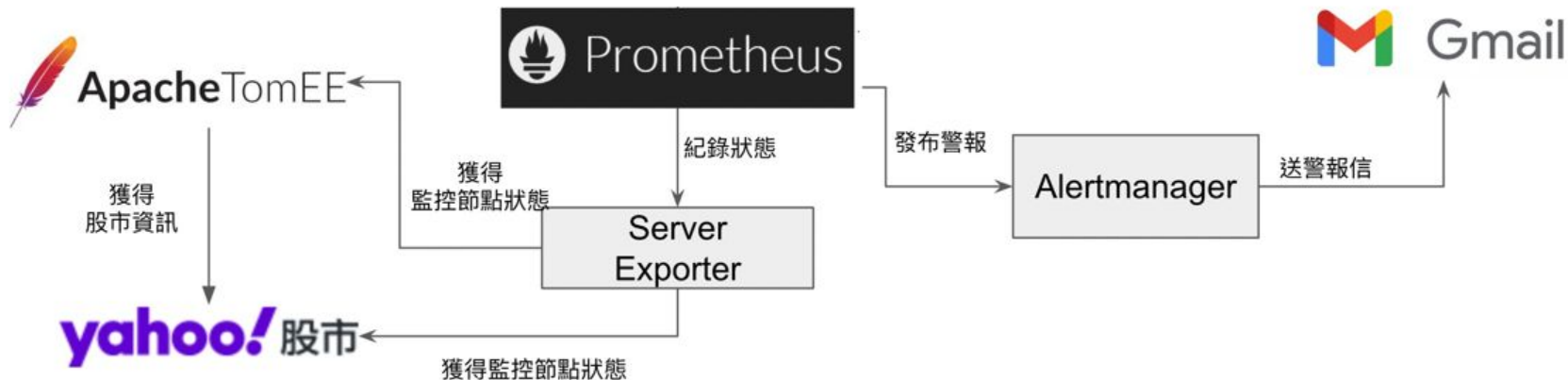
Server
Exporter

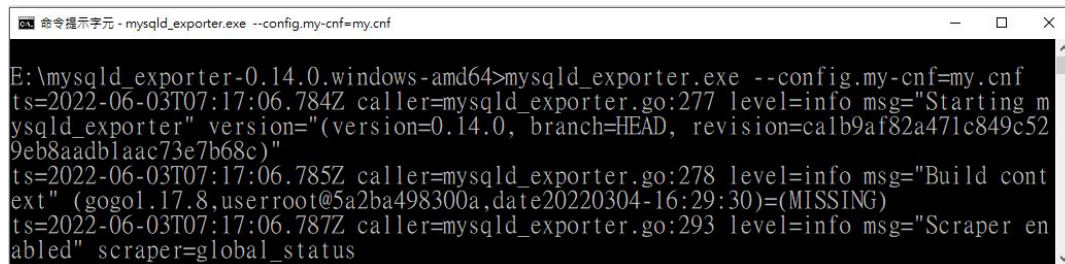
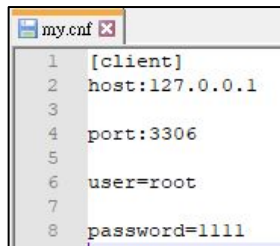
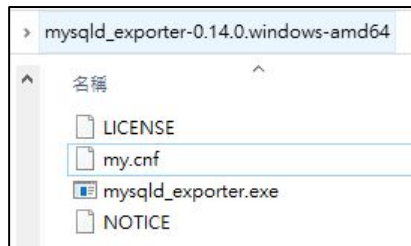
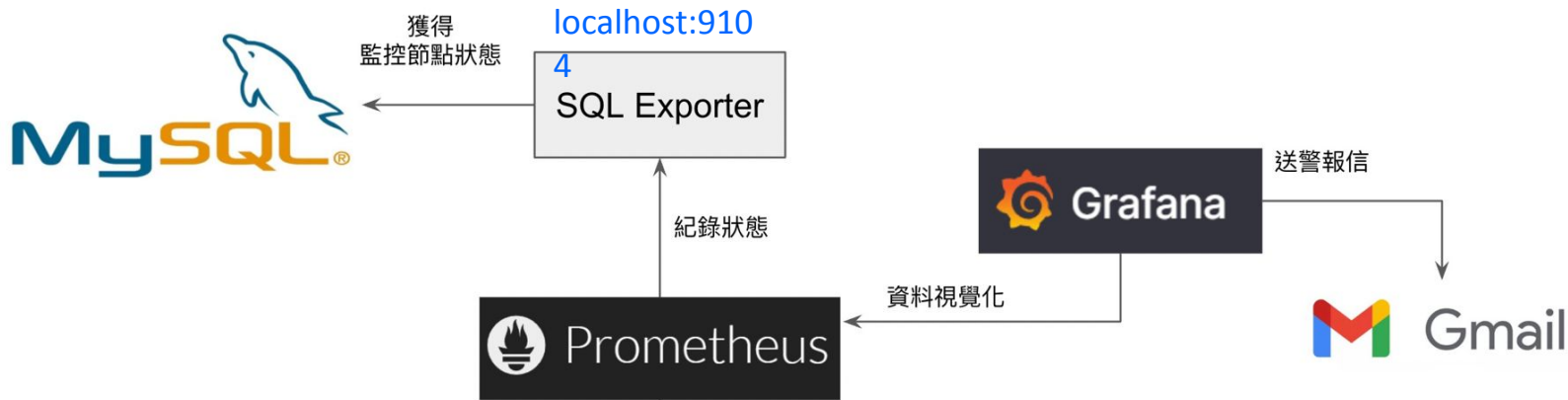
Alertmanager

送警報信

獲得監控節點狀態









獲得
監控節點狀態

SQL Exporter

紀錄狀態



Grafana

送警報信

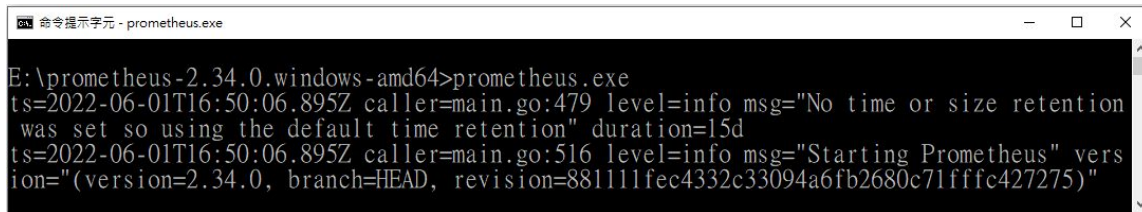
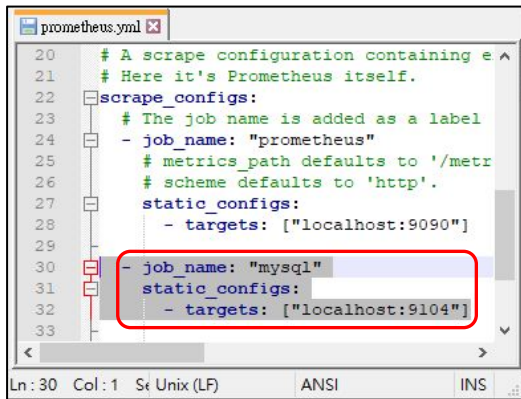
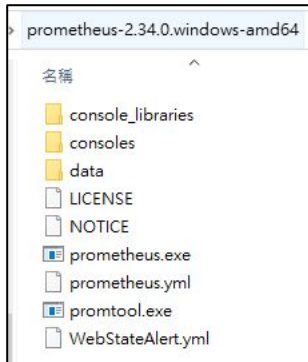


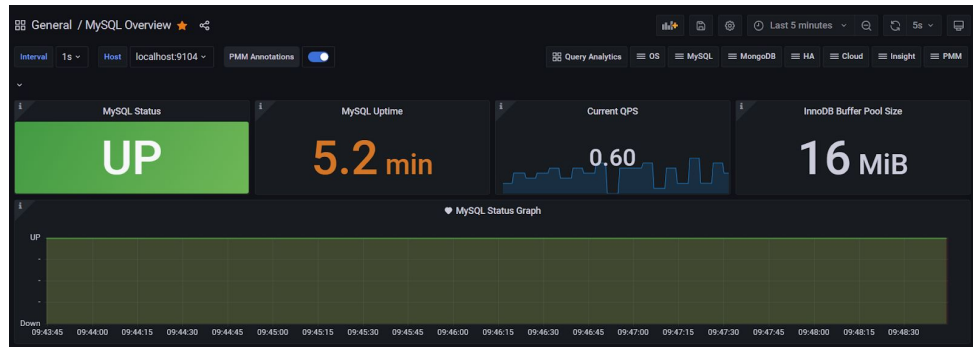
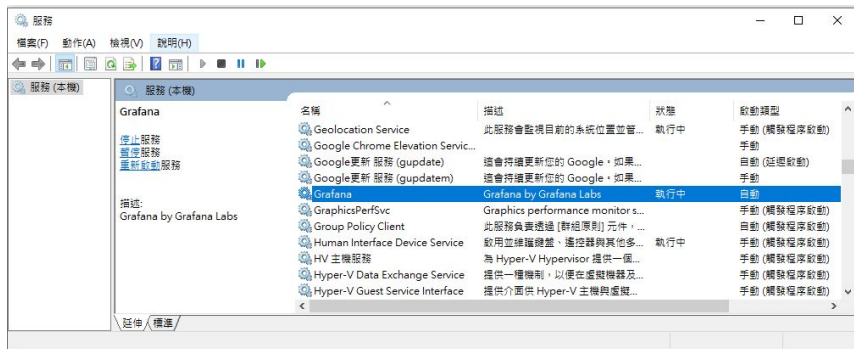
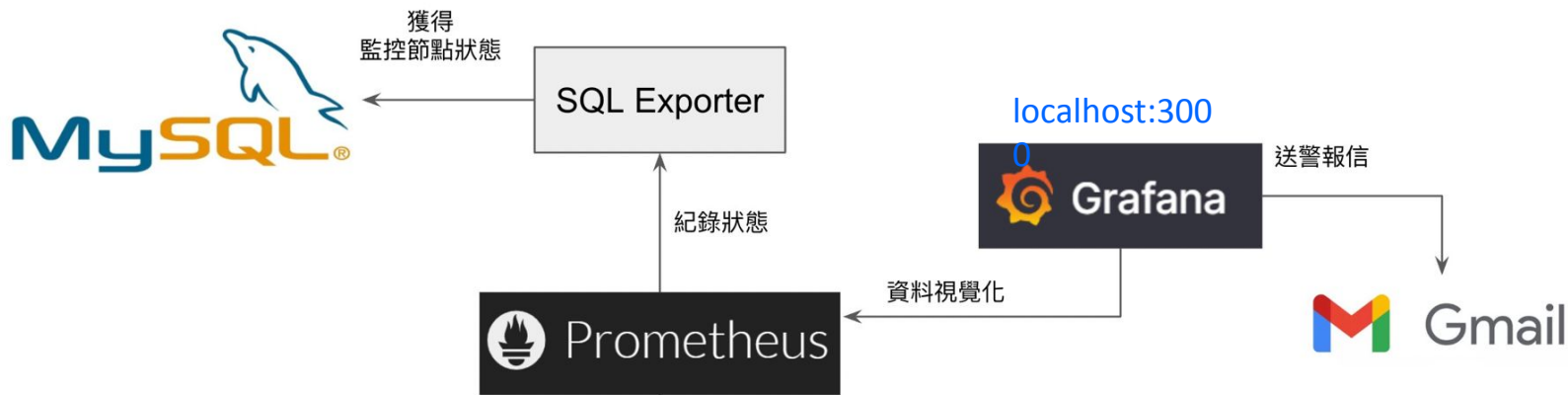
Gmail

資料視覺化



Prometheus



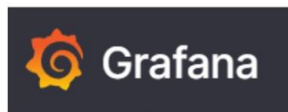




獲得
監控節點狀態

SQL Exporter

紀錄狀態



送警報信

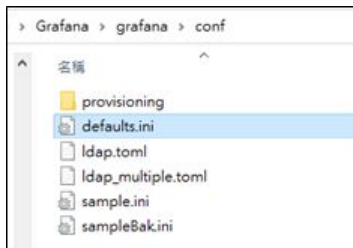


Gmail

資料視覺化

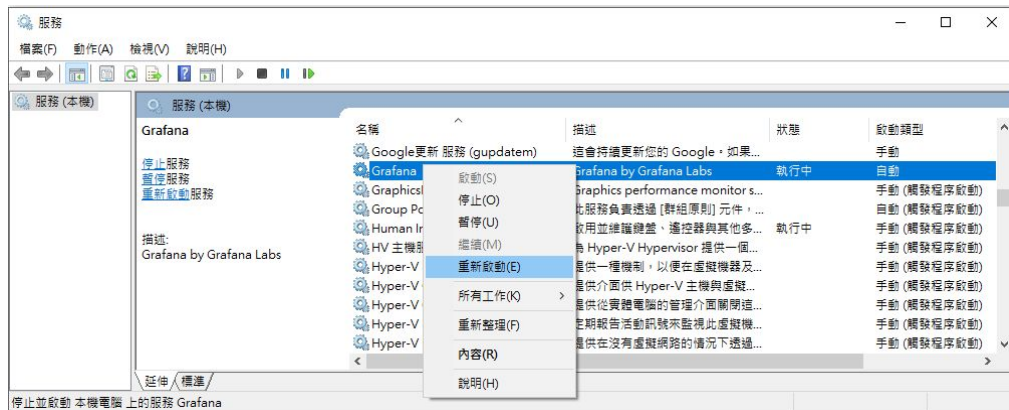


Prometheus



```
defaults.ini
##### SMTP / Emailing #####
[smtp]
enabled = true
host = smtp.gmail.com:587
user = 110971029@g.nccu.edu.tw
# If the password contains # or ; you have to wrap it
password = 
cert_file = 
key_file = 
skip_verify = false
from_address = 110971029@g.nccu.edu.tw
from_name = Grafana
ehlo_identity = 
startTLS_policy = 

[emails]
welcome_email_on_sign_up = false
templates_pattern = emails/*.html, emails/*.txt
content_types = text/html
```



影片: [1102 Distributed Systems 第八組 - Prometheus](#)

Github: https://github.com/kebwlmmbhee/1102_Dist_Sys