# Pilot Log Anonymization in GlideinMonitor

Mirica Yancey, Valparaiso University – SIST Intern

#### Introduction

GlideinWMS is a pilot-based resource provisioning tool for distributed High **Throughput Computing** 

- Provides reliable and uniform virtual clusters to run scientific computations (analysis, simulations, reconstructions etc.)
  - Provisions computers from many sources (local clusters, Grid, Cloud, Supercomputers)

#### GlideinMonitor

- Web application to view GlideinWMS's log files
- User interface tool
- Useful for quick searches and decoding log content
- Contains an efficient, managed archive of log files

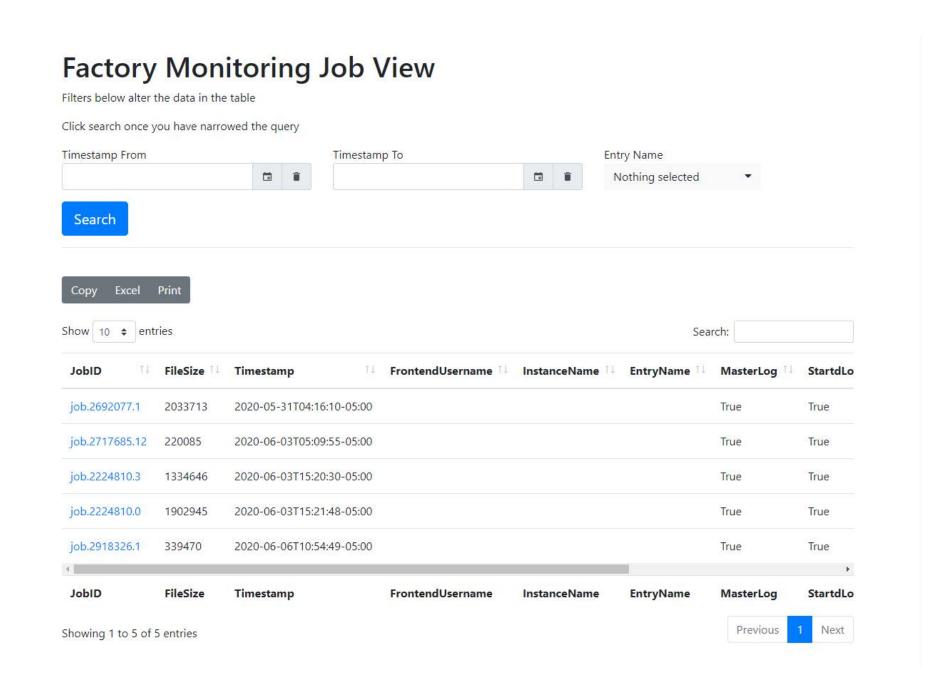


Figure 1 – GlideinMonitor Homescreen

#### **Problem**

 Log files contain sensitive information that makes the logs less secure and decreases the ease of distribution amongst developers.

### **Objective**

#### Create a script that:

- Locates the job submitter's IP Address
- Locates any personal identifiers of the job submitter
- Filters/Replaces those values with non-identifiable information

#### In order to:

- Protect user information.
- Allow easier distribution to developers

#### Methods

Identified what the personal information was and then cross referenced files to ensure there was a consistent reference for this information in order to search and retrieve it.

```
f findEmail(filename): #CONDOR SPECIFIC finds and returns user email (before @symbol)
 with open(filename, 'rb', 0) as file, \
     mmap.mmap(file.fileno(), 0, access=mmap.ACCESS_READ) as s:
     if s.find(b'x509UserProxyEmail') != -1:
        x = s.find(b'x509UserProxyEmail')
         end = s.find(b'@',x)
        lis = (((s[x: end].split(b'='))[1].decode("utf-8")).replace('"', '')).replace(' ', ''
```

Figure 2 – Email Identifier Script

```
ef findUserIds(filename): #CONDOR SPECIFIC finds and returns user email
  with open(filename, 'rb', 0) as file, \
       mmap.mmap(file.fileno(), 0, access=mmap.ACCESS READ) as s:
      if s.find(b'x509UserProxyFQAN = "') != -1
          x = s.find(b'x509UserProxyFQAN = "')
          end = s.find(b'Role=',x)
          lis = s[x: end].decode("utf-8")#
          lis = lis.split('CN=')
          lis = " ".join(lis).replace("/","").replace(",cms","").split(")
```

Figure 3 – User Information Identifier Script

 Created a method to replace this information and followed software engineering standards: added docstrings and unit tests to insure clarity and quality

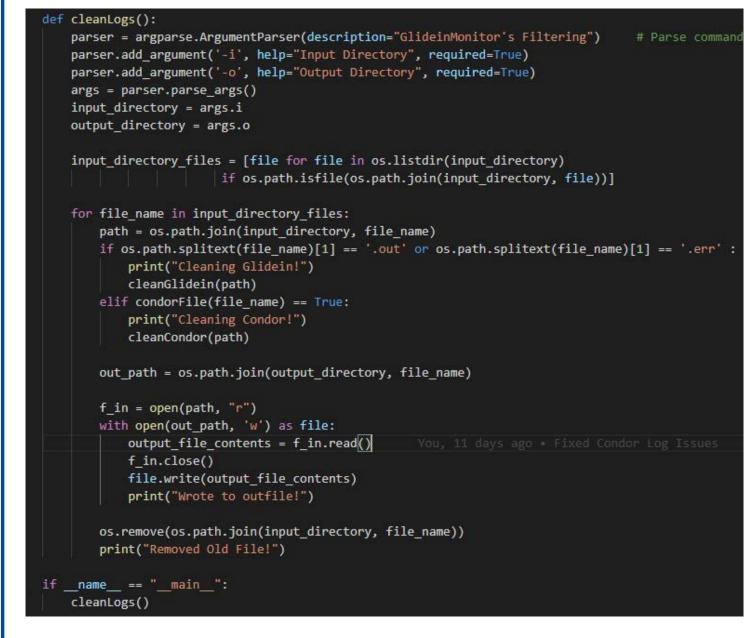


Figure 4 – Main Filtering Method

```
a few seconds ago | 1 author (You)
s TestFilter(unittest.TestCase):
 def test_glidein(self): #Glidein Logs
    file1 = "regular_logs/overall_test_in.txt"
    user_ids2 = findGlideinUserIDs(file1)
    cleanGlidein(file1)
    file = open(file1, "r", encoding="utf8")
    line = file.read()
    file.close()
    self.assertNotRegex(line, IP_REGEX[0])
    self.assertNotRegex(line, IP REGEX[1])
    self.assertNotRegex(line, rf"{user_ids2[0]}[\w \W]{user_ids2[1]}")
    for x in range(0, len(user ids2)):
        self.assertFalse(line.find(user_ids2[x]) != -1)
```

Figure 5 – Glidein Unit Test

#### Results

- Filtered logs contain no personal identifiers
- Script tested individually
- Script integrated into GlideinMonitor and included in RPM release

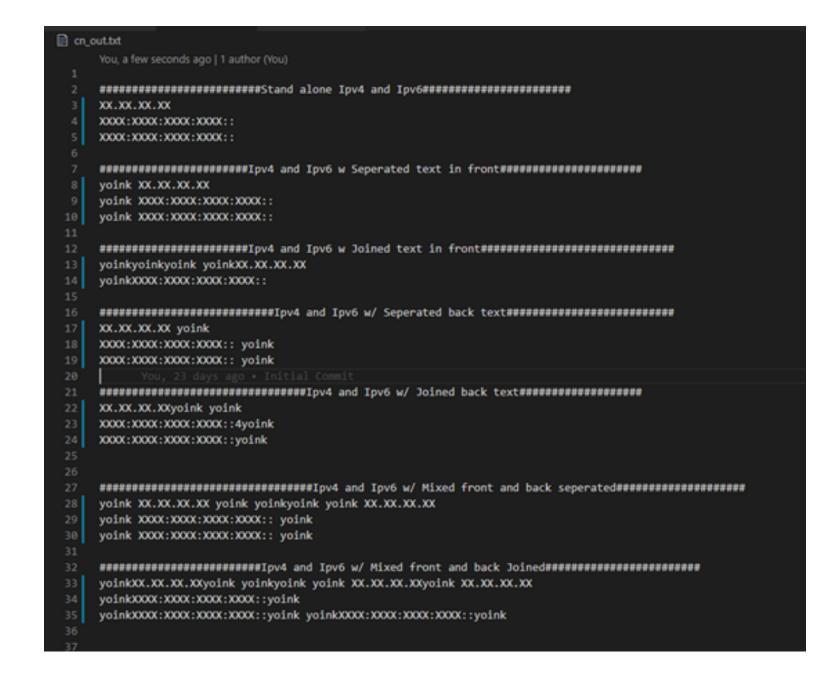


Figure 6 – Filtered Test File

```
MemoryUsage = ifthenelse(ProportionalSetSizeKb =?= undefined,(ResidentSetSize + 1023) / 1024,(ProportionalSetSizeKb + 1023) / 1024)
```

Figure 7 – Filtered IP Address in Log File

```
CRAB SubmitterIpAddr = "XXXX:XXXX:XXXX:XXXX::127"
CRAB TaskEndTime = 1593944968
CRAB TaskLifetimeDays = 30
CRAB TaskWorker = "crab-prod-tw02"
CRAB TFileOutputFiles = {
CRAB TransferOutputs = 1
CRAB_UserDN = "/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=USER/CN=USER/CN=USER USER"
CRAB_UserGroup = undefined
CRAB_UserHN = "USER"
```

Figure 8 – Filtered User Identifiers in Log File

## Acknowledgements

Thank you to my supervisor Marco Mambelli, the GlideinWMS team, and my SIST Mentoring team for consistently making me feel welcome and heard. Thank you for the belief in my skill set and the time you dedicated towards helping me grow that set. In addition to that, thank you Fermilab for the amazing opportunity you provided to me and the other interns this summer.



