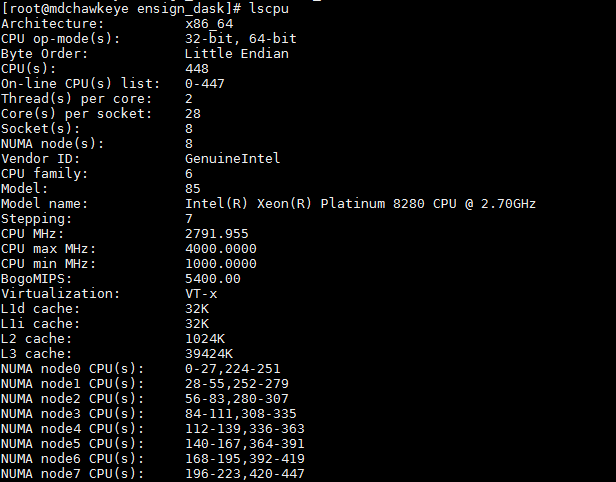
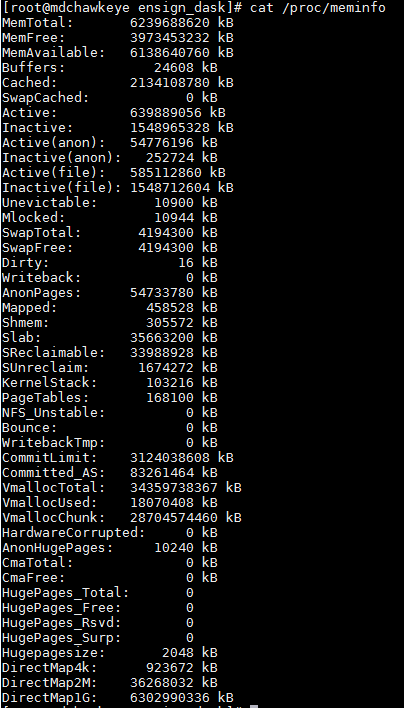
Implementing Multi-User Cybersecurity Graph Analytics with xGT & ENSIGN on SDFX (8S/6TB)

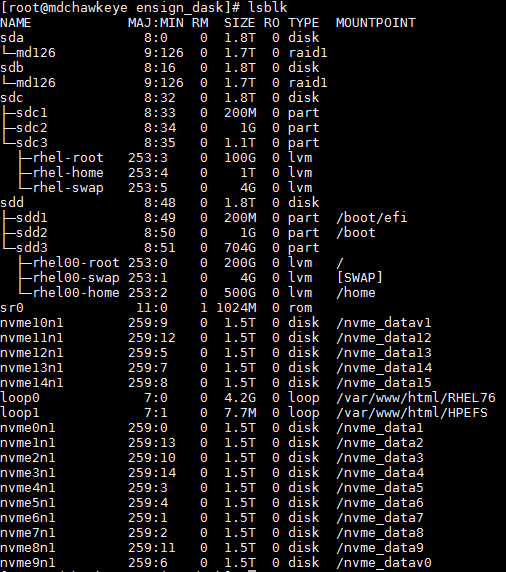
# HPE Lab Test environment Details

|  |  |
| --- | --- |
|  | **Configuration** |
| **No. Of Chassis** | 2 ( 1 nPar ) |
| **Processor Architecture** | Cascade Lake. Xeon-Platinum **8280**, (2.7 GHz/28-core/205 W) |
| **Total Sockets** | 4 Socket / Chassis , 4 \* 2 = 8 Socket |
| **Total Cores** | 28 Core / Socket , 8 \* 28 = 224 Cores |
| **Total Memory** | 24 \* 128 GB per chassis , 24 \* 128 \* 2 ~ 6 TB |
| **Total Storage** | 2 NVMe per socket , 1.6 TB \* 2 \* 8 ~ 25 TB |

**CPU configuration details**

**Memory Configuration**



**Storage Disk Configuration**

**Environment planning for Co-Execution Test Environment**

* 1. **Trovares**
     1. Dataset : LANL Dataset
     2. Total Raw Data Volume : 1V , 2V, Netflow

|  |  |
| --- | --- |
| **LANL (xGT)** | **Data Volume(GB)** |
| LANL 1V | 116 |
| LANL 2V | 356 |
| LANL Netflow | 946 |
| **Total** | **1418** |

* 1. **Reservoir Lab**
     1. Dataset : CIC Dataset
     2. Total RAW Data Volume :

|  |  |
| --- | --- |
| **CICDDoS2019 (ENSIGN)** | **Data Volume(GB)** |
| csv | 116 |
| pcap | 356 |
| **Total** | **472** |

* 1. **Resource Planning**
     1. **Master/Backup Copy of Dataset :** 3TB : 2NVMe Drives
     2. **Working Dataset :** 
        1. Trovares (xGT) : 6 \* 1.5 TB = 9 TB ( /nvme\_data1 - /nvme\_data6 )
        2. RL (ENSIGN) : 6 \* 1.5 TB = 9 TB ( /nvme\_data7 - /nvme\_data12 )
     3. **Memory Requirement for xGT & ENSIGN (Single User):**
        1. xGT : 3TB
        2. ENSIGN : 3 TB
     4. **Compute Allocation for xGT & ENSIGN (Single User):**
        1. xGT : 112 Cores ( CPUID 0-111 )
        2. ENSIGN : 112 Cores ( CPUID 112-223 )
  2. **Key Configuration Files**
     1. **xGT :** 
        1. **Install Location : “/opt/xgtd”**
        2. **Environment Variables :**

#setup xgt path

export PATH=$PATH:/opt/xgtd/bin

export PATH=$PATH:/opt/xgtd/lib

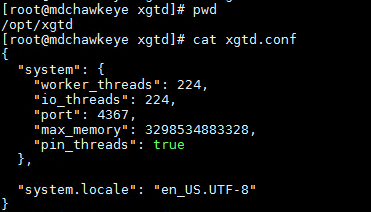
export PATH=$PATH:/opt/xgtd/util

export LSHOST="NO-NET"

export LSERVRC="/opt/xgtd/bin/lservrc"

export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/usr/local/lib:/opt/xgtd/lib

* + - 1. **Conf file : “/opt/xgtd/xgtd.conf”**



* + 1. **ENSIGN :** 
       1. **Install Location : “/root/ashish/reservoir\_lab/ENSIGN-42”**
       2. **Environment Variables :**

#add following environment variables for ENSIGN from RESERVOIR LAB

export ENSIGN\_BASE=/root/ashish/reservoir lab/ENSIGN-42

export ANACONDA\_HOME=/root/anaconda3/

export PATH=$ENSIGN\_BASE/ENSIGN\_4.2/bin:$PATH

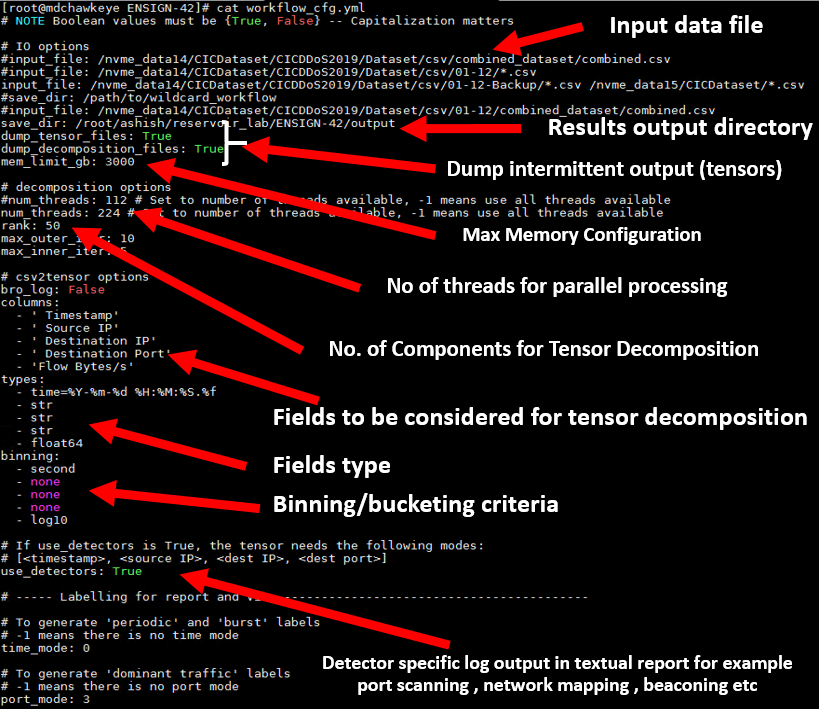
export PATH=$ENSIGN\_BASE/ENSIGN\_4.2/Ensign-Py3/bin:$PATH

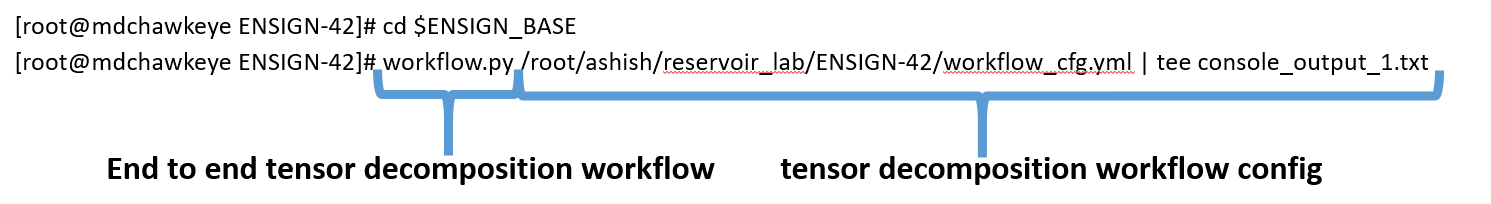
export PATH=$ANACONDA\_HOME/bin:$PATH

export PYTHONPATH=$ENSIGN\_BASE/ENSIGN\_4.2/Ensign-Py3

export LD\_LIBARY\_PATH=$LD\_LIBRARY\_PATH:$ENSIGN\_BASE/ENSIGN\_4.2/Ensign-CAPI/lib

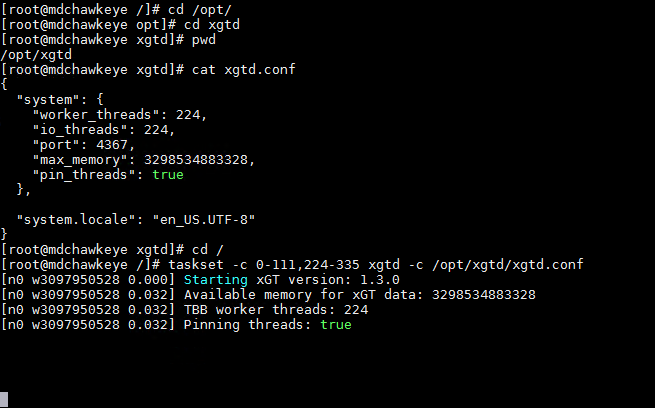
* + - 1. **Workflow Conf file : “/root/ashish/reservoir\_lab/ENSIGN-42/workflow\_cfg.yml”**





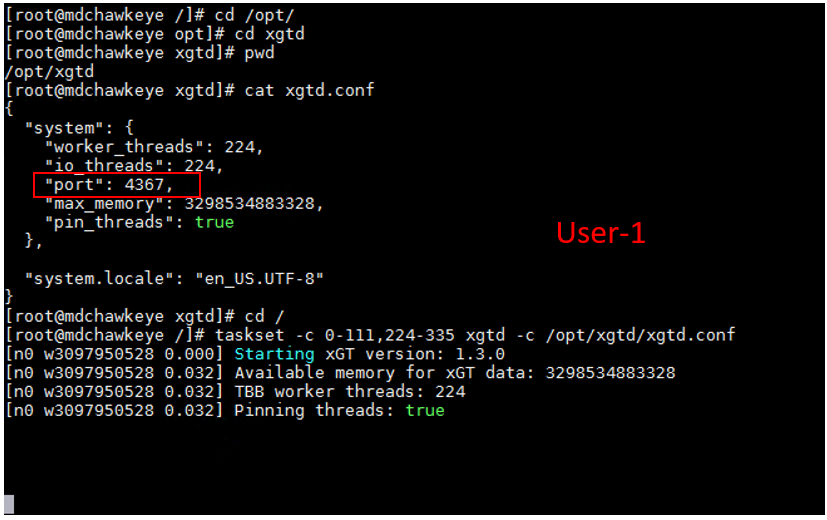
# Install and Configure xGT for Multi-User Graph Analytics for Known Threat Detection

* 1. Executing xGT Server with limited memory/cpu threads (**single-user**)

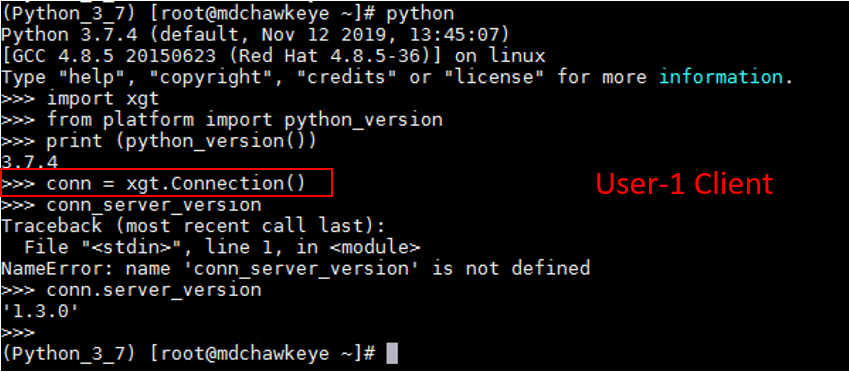


* 1. Executing xGT Server with limited memory/cpu threads (multi-user)

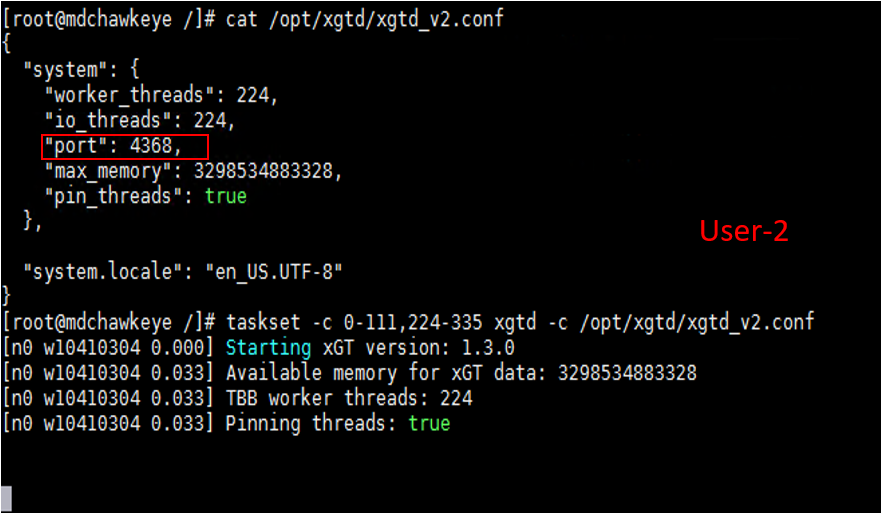
User-1 Server



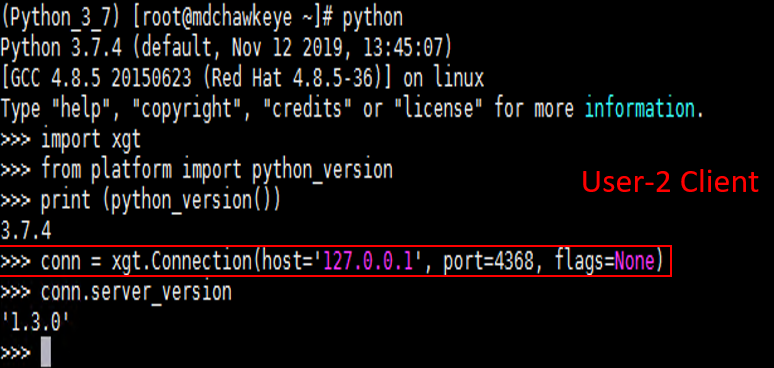
User-1 Client



User-2 Server



User-2 Client



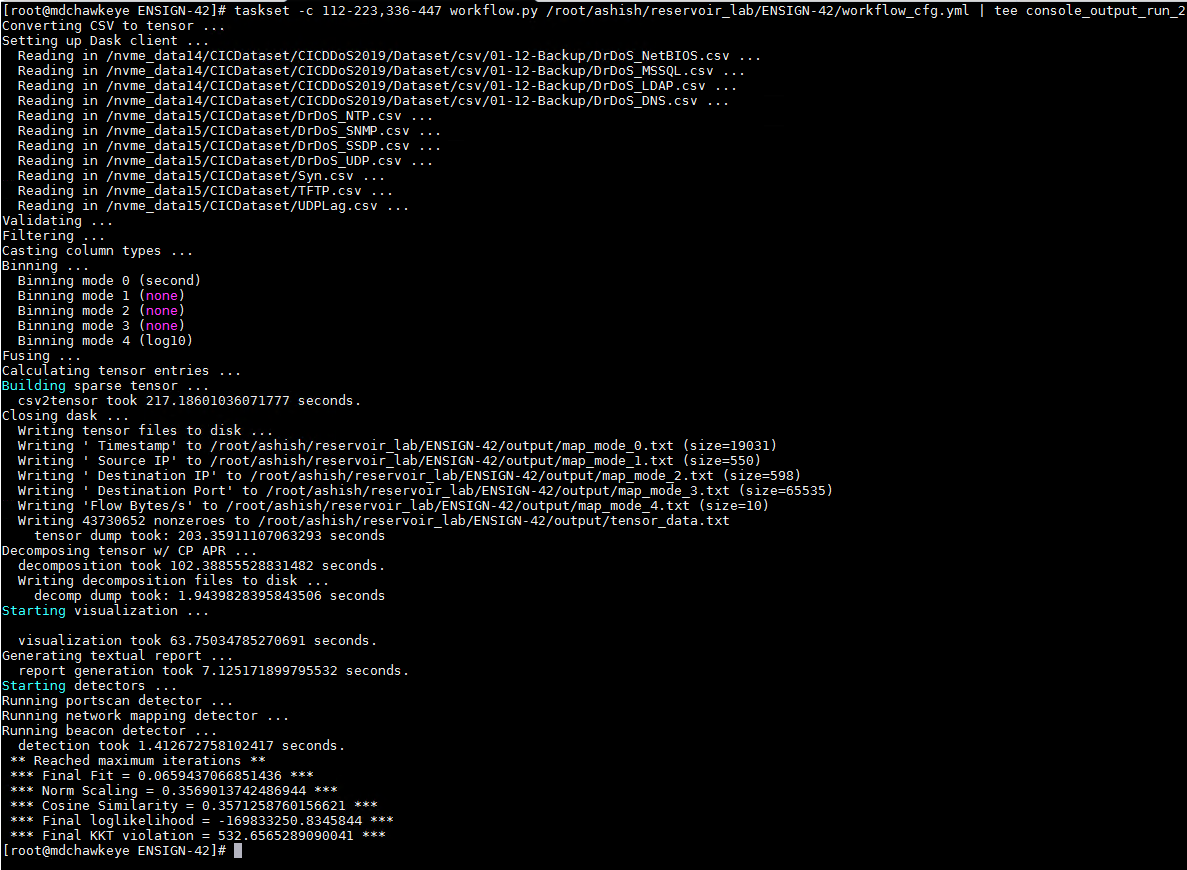
# Install and Configure ENSIGN for Multi-User Unsupervised Machine Learning for Un-Known Thread Detection

* 1. **Executing ENSIGN with memory/cpu threads limitations (single user)**

num\_threads = 112 ( no. of physical core allocated )

mem\_limit\_gb = 3000 ( memory allocated )

[root@mdchawkeye ENSIGN-42]# taskset -c 112-223,336-447 workflow.py /root/ashish/reservoir\_lab/ENSIGN-42/workflow\_cfg.yml | tee console\_output\_run\_2



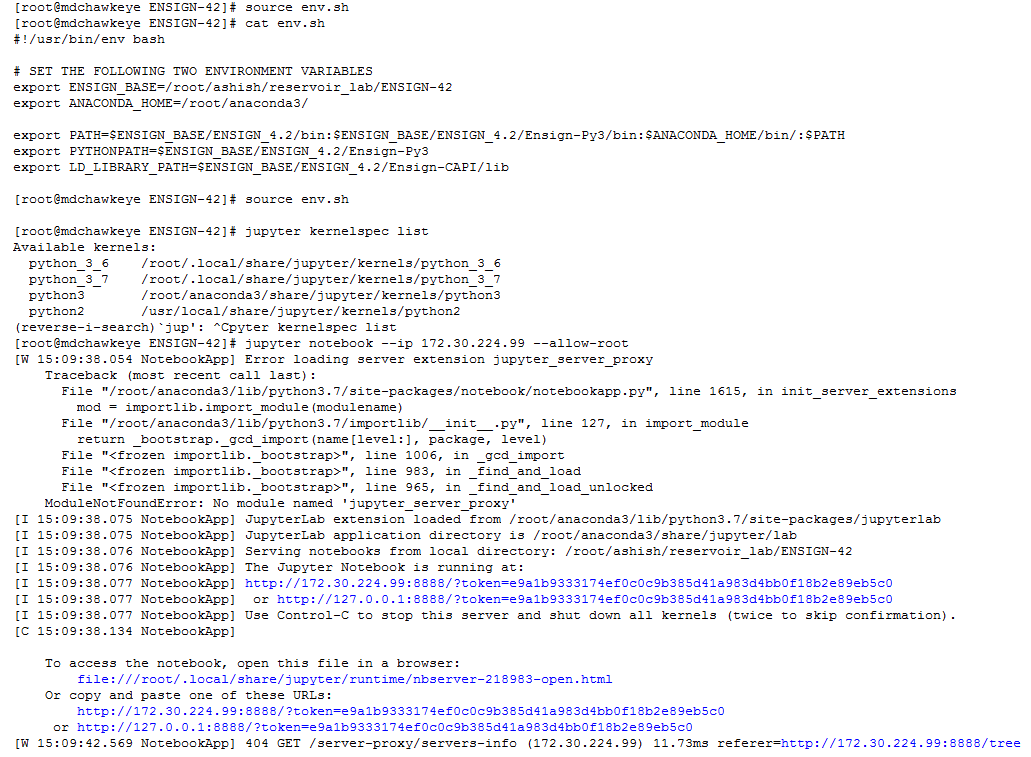
* 1. **Executing ENSIGN with memory/cpu threads limitations (multi user)**

1. Create (per user) workflow configuration file

2. Create (per user) input file path and output directory

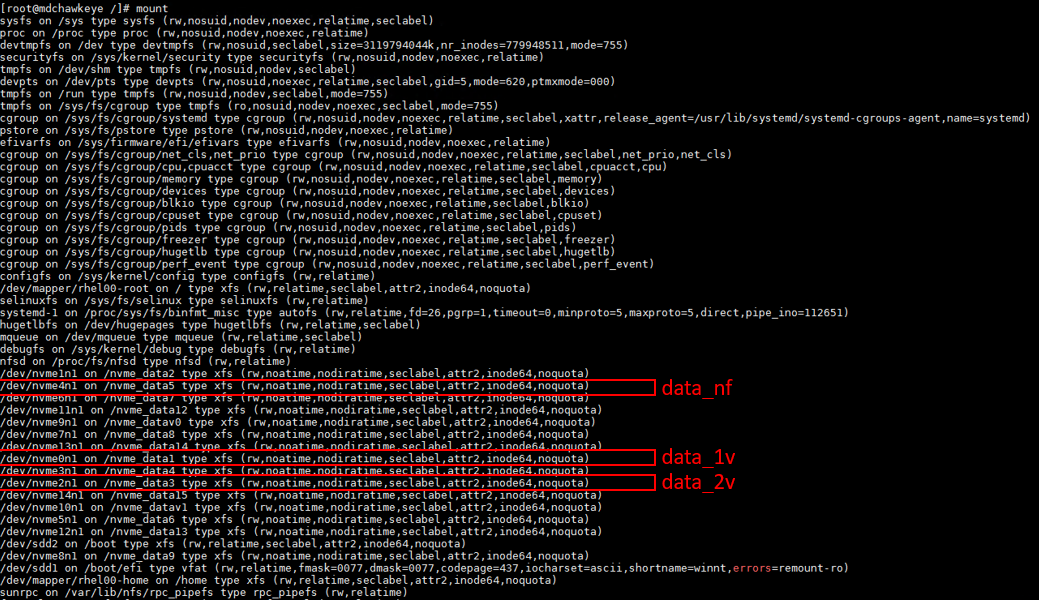
3. Execute (per user) workflow as indicated in step above

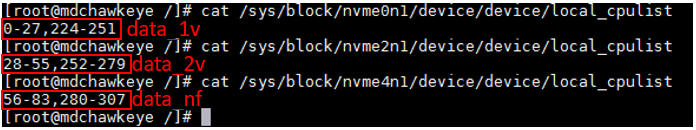
# Setup Jupyter for Multi-User Execution

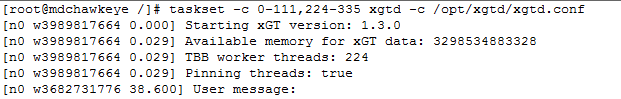


# Parallel Test Execution

**Executed xGT with limited core and memory**

1. Identify NVMe drives for storing input data & distribute data across multiple NVMe drives for parallel loading

2. Identify mapping CPUs to be allocated for xGT execution

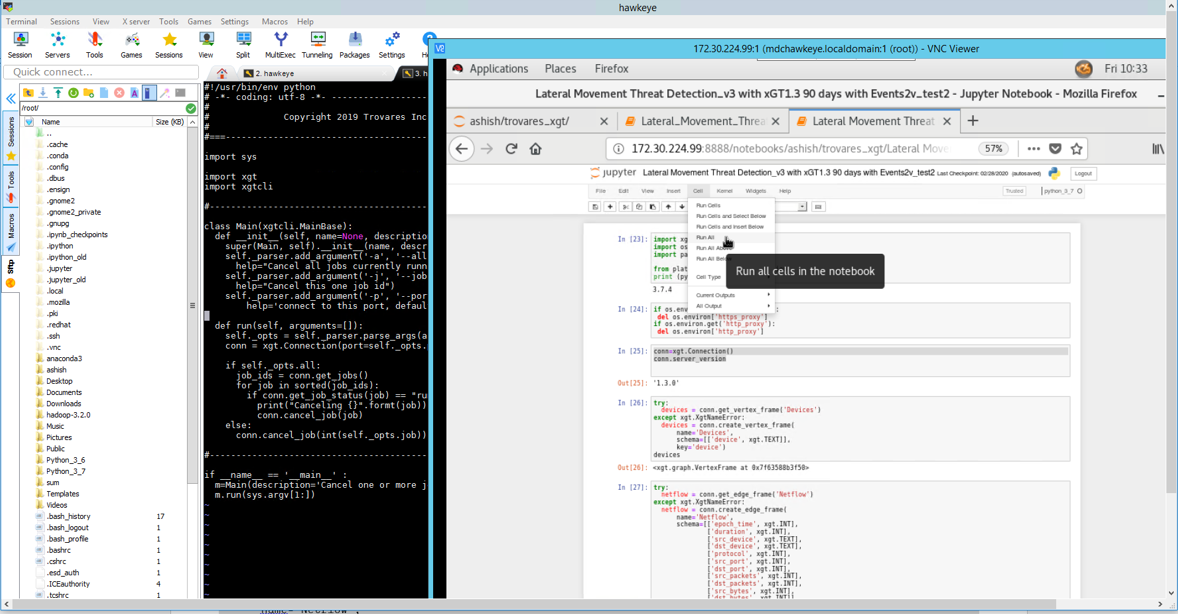
3. Start xGT server with pinned cores

4. Execute LANL detection Queries

a) Launch Jupyter Notebook Environment

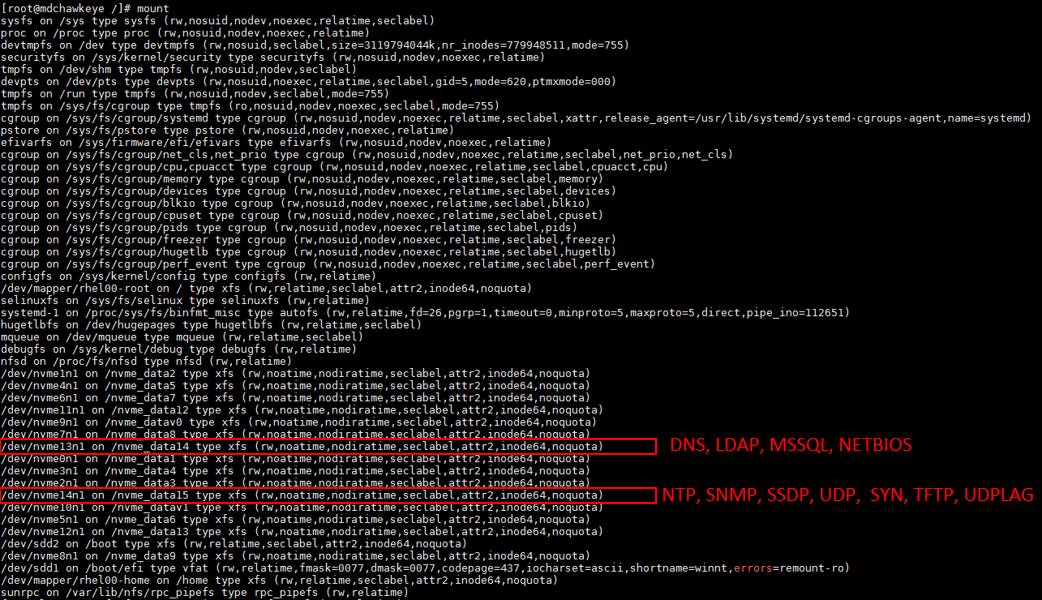
b) Load Lateral Movement Cyber-Threat detection notebook

c) Execute the notebook End-2-End with performance data collection enabled

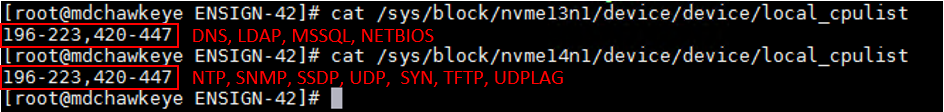


**Execute ENSIGN with limited core and memory**

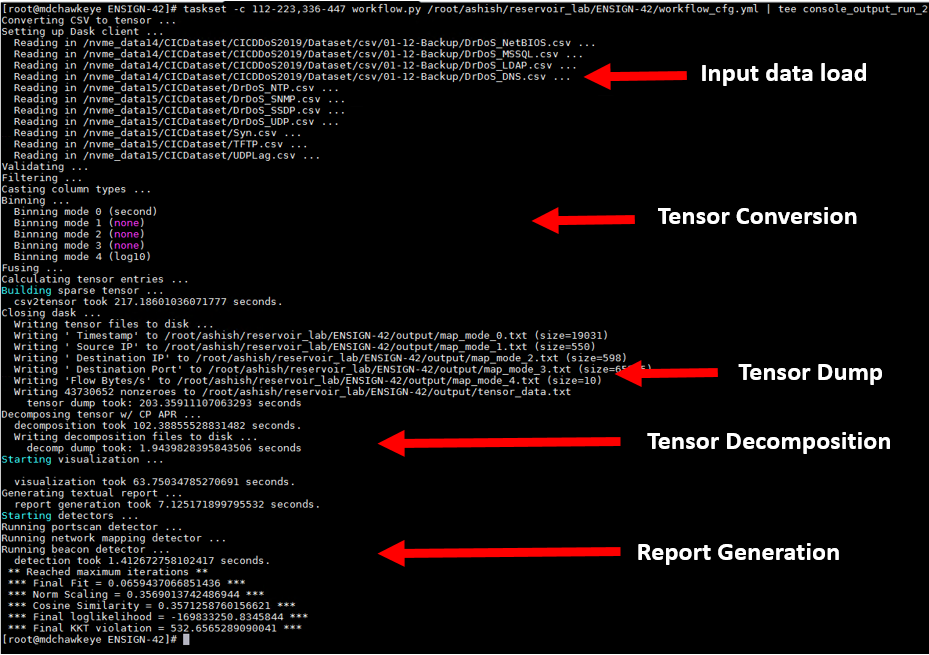
1. Identify NVMe drives for storing input data & distribute data across multiple NVMe drives for parallel loading



2. Identify mapping CPUs to be allocated for ENSIGN execution

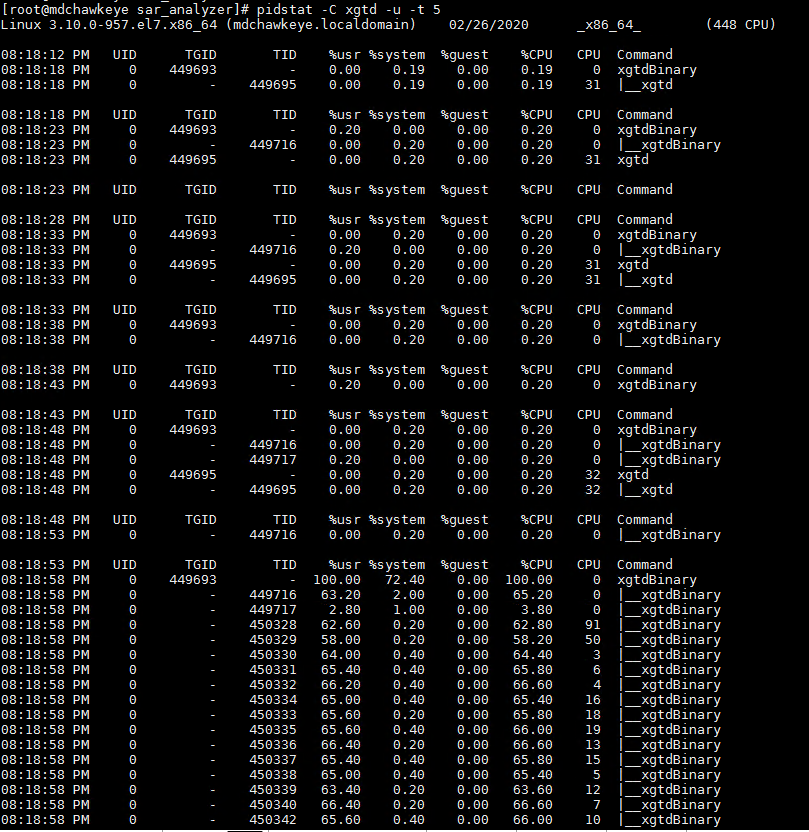


3. Start ENSIGN workflow with pinned cores

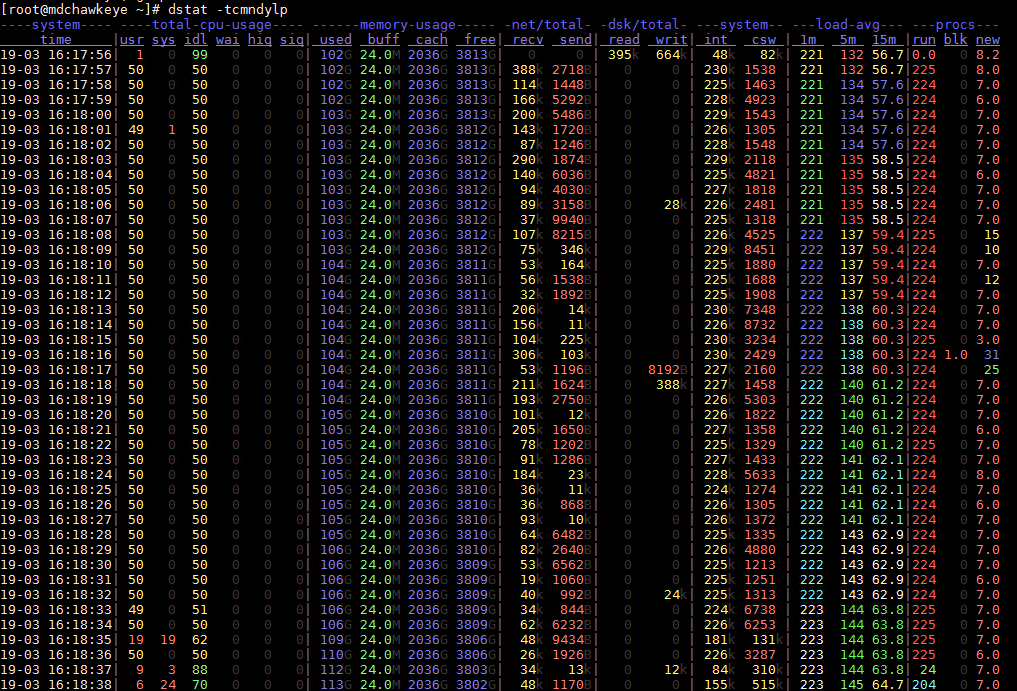


# Performance Metrics Collection

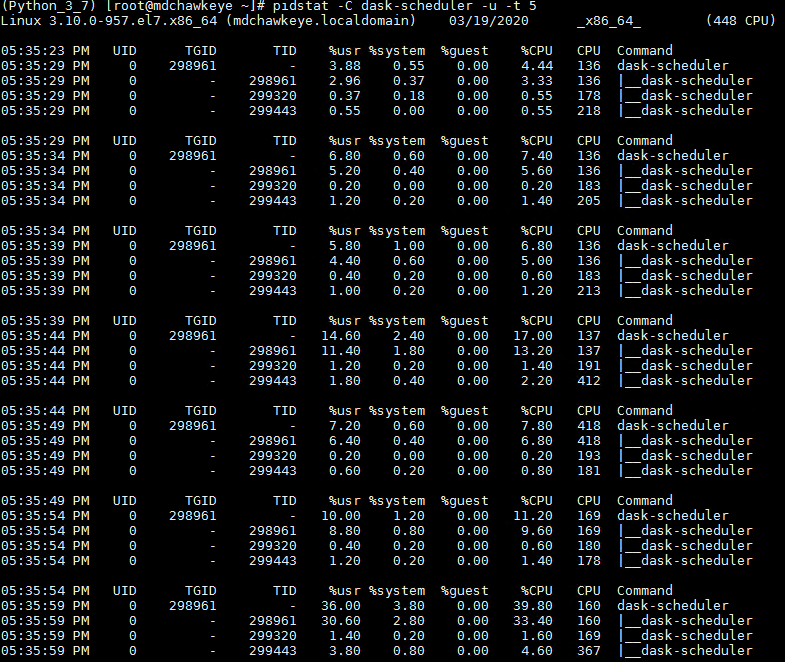
Executed xGT with limited core and memory – Individual Process Monitoring

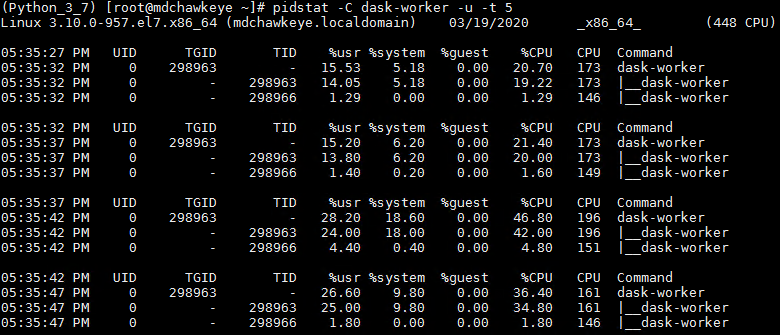
pidstat –C xgtd –u –t 5

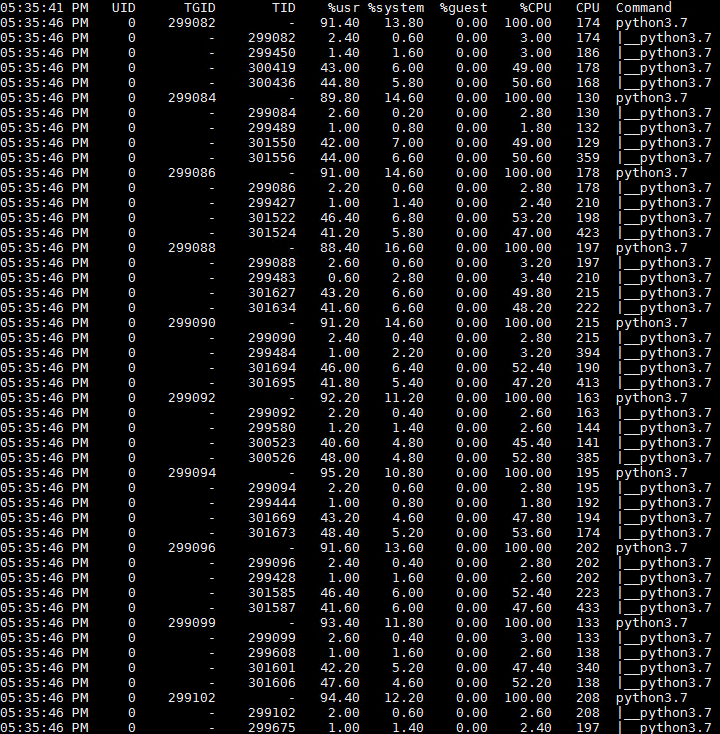
Overall System Performance

dstat -tcmndylp

Execute ENSIGN with limited core and memory

Monitor dask scheduler

Monitor dask worker

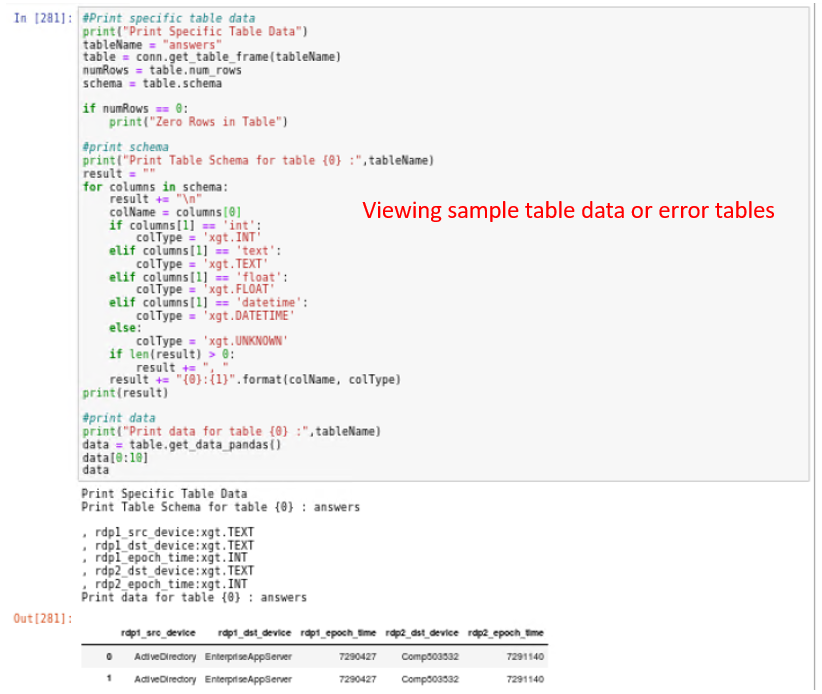
Monitor Python Child Processes: pidstat –C python3.7 –u –t 5

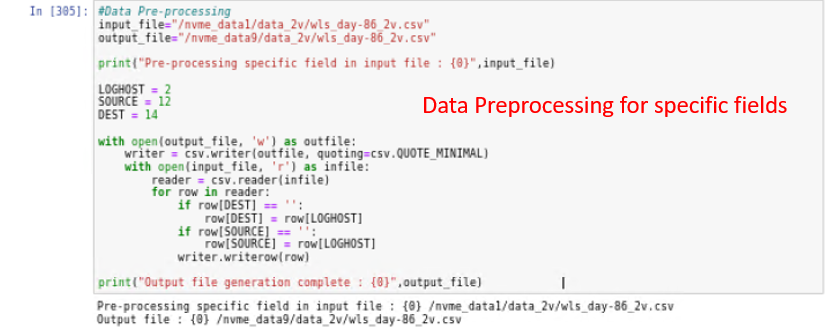
# Troubleshooting Tips

A. Notebook Script to monitor/manage/troubleshoot during xgt execution

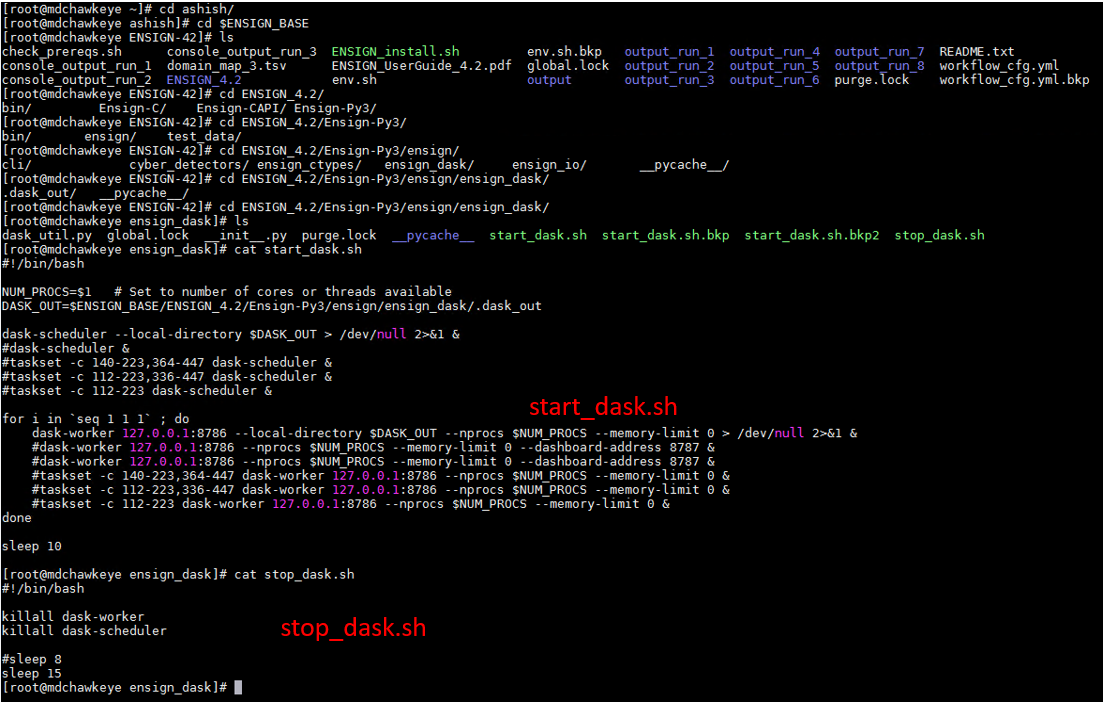








B. Script to monitor/manage/troubleshoot during ENSIGN execution



C. CICDDoS Dataset 2019

*[root@mdchawkeye nvme\_data14]# cat get\_data.sh*

*#GET CIC DDOS2019 Dataset in pcap format*

*wget "http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/PCAPs/03-11/PCAP-03-11.zip" -O “/nvme\_data14/CICDataset/CICDDoS2019/Dataset/PCAPs/PCAP-03-11.zip"*

*wget "http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/PCAPs/01-12/PCAP-01-12\_0-0249.zip" -O "/nvme\_data14/CICDataset/CICDDoS2019/Dataset/PCAPs/PCAP-01-12\_0-0249.zip"*

*wget "http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/PCAPs/01-12/PCAP-01-12\_0250-0499.zip" -O "/nvme\_data14/CICDataset/CICDDoS2019/Dataset/PCAPs/PCAP-01-12\_0250-0499.zip"*

*wget "http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/PCAPs/01-12/PCAP-01-12\_0500-0749.zip" -O "/nvme\_data14/CICDataset/CICDDoS2019/Dataset/PCAPs/PCAP-01-12\_0500-0749.zip"*

*wget "http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/PCAPs/01-12/PCAP-01-12\_0750-0818.zip" -O "/nvme\_data14/CICDataset/CICDDoS2019/Dataset/PCAPs/PCAP-01-12\_0750-0818.zip"*

*#GET CIC DDOS2019 Dataset in csv format*

*wget "http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/CSVs/CSV-01-12.zip" -O "/nvme\_data14/CICDataset/CICDDoS2019/Dataset/csv/CSV-01-12.zip"*

*wget "http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/CSVs/CSV-03-11.zip" -O "/nvme\_data14/CICDataset/CICDDoS2019/Dataset/csv/CSV-03-11.zip"*

*[root@mdchawkeye nvme\_data14]#*

D. LANL Dataset 2019

*[root@mdchawkeye nvme\_data1]# cat get\_data.sh*

*for i in {1..90}; do wget "https://datasets.trovares.com/LANL/xgt/wls\_day-"$i"\_2v.csv" -O "/nvme\_data1/data\_1v/wls\_day-"$i"\_2v.csv" ; done*

*for i in {1..90}; do wget "https://datasets.trovares.com/LANL/xgt/wls\_day-"$i"\_1v.csv" -O "/nvme\_data1/data\_2v/wls\_day-"$i"\_1v.csv" ; done*

*for i in {1..90}; do wget "https://datasets.trovares.com/LANL/xgt/nf\_day-"$i".csv" -O "/nvme\_data1/data\_nf/nf\_day-"$i".csv" ; done*

*[root@mdchawkeye nvme\_data1]#*