

Operational Intelligence @ Glasgow



Outline

- Glasgow Tier2 site
- Cluster Monitoring & Logging
- Visualization & Querying
- Alerts & Automation
- Latest Addition: auto-update



ScotGrid Glasgow:

Emanuele Simili, Gordon Stewart, Samuel Skipsey, David Britton

ScotGrid Glasgow Site

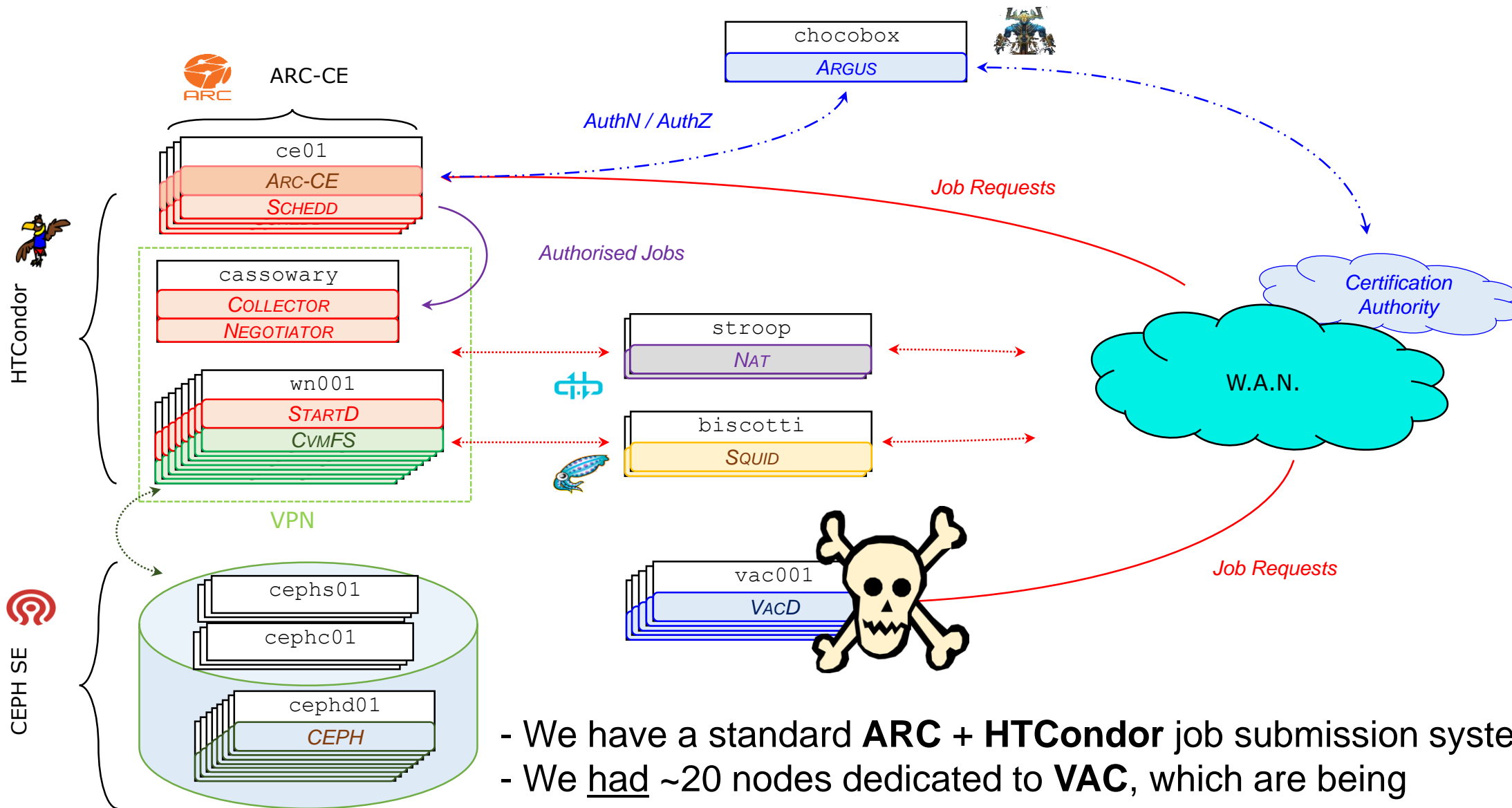


At present *, ScotGrid Glasgow consists of:

- ~13'000 CPU cores
- ~12 PB physical storage (CEPH + DPM)
- 40 Gb/s internal network bandwidth

For a total under-estimated at about 61 Khs06 #

Cluster Map



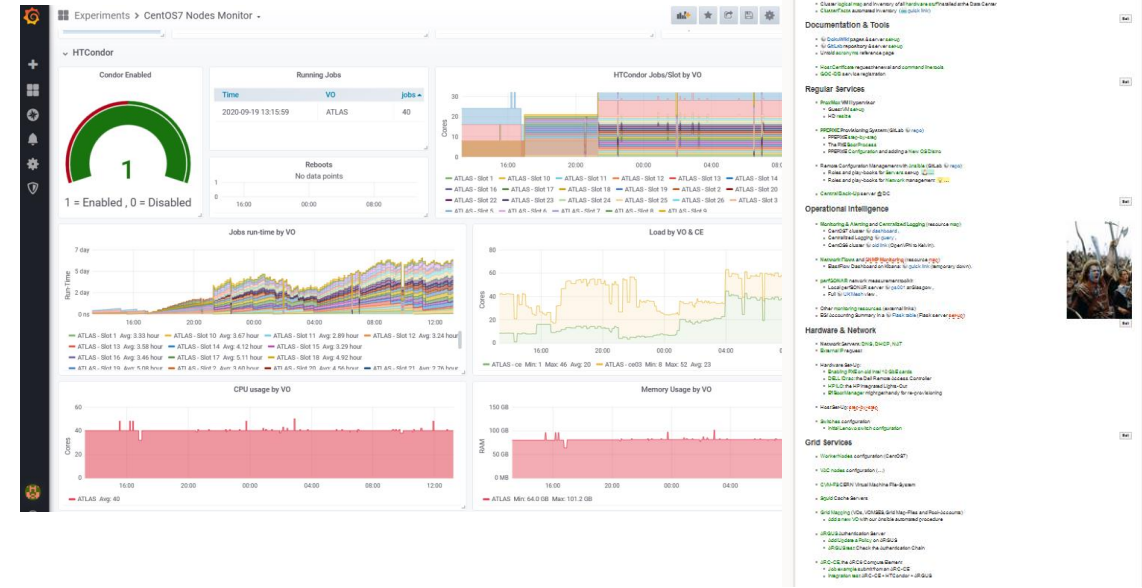
- We have a standard **ARC + HTCondor** job submission system,
- We had ~20 nodes dedicated to **VAC**, which are being decommissioned (as of Sep. 2021).

Automation Tools

As a site of significant size, we use many sys-admin empowering tools:

- A network provisioning system (**PPE PiXiE**)
- A remote management tool (**Ansible**)
- A central monitoring & logging system (**PLG**) ... *which I will talk about*

The image shows two side-by-side screenshots of web interfaces. The left screenshot is the GitLab project page for 'ppepixie', showing details like 6 commits, 1 branch, and 0 tags. The right screenshot is the Ansible project page for 'ScottGrid', showing details like 636 commits, 2 branches, and 0 tags. Both pages show a list of files and their last commit dates.



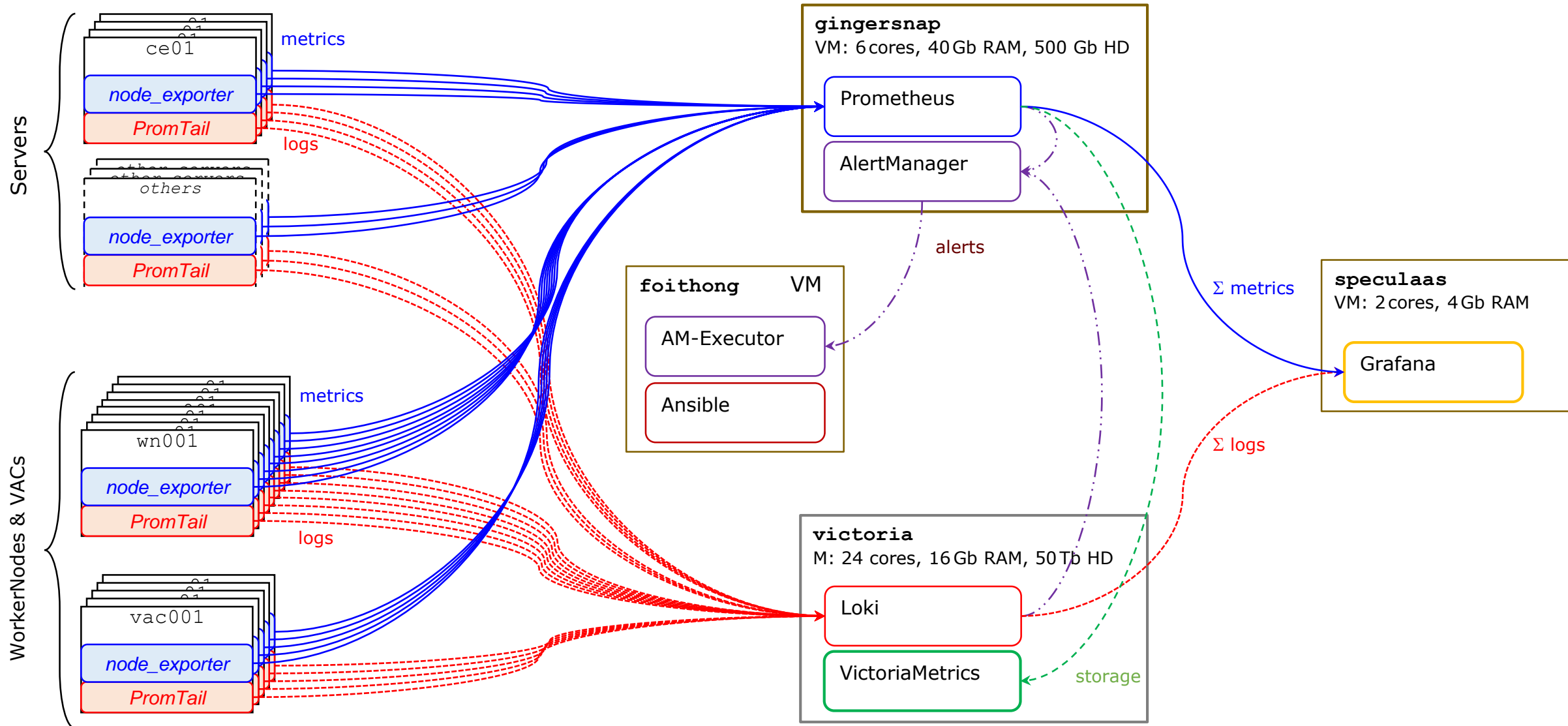
We have invested quite some time in preparing such environments because, in the long run, it has saved us more time than we spent making it.

Monitoring & Logging (1)

Our system is built on **Prometheus**, **Loki** and **Grafana (PLG)**

- Metrics are exported by **node_exporter** (installed on nodes) and collected by a central **Prometheus** instance
- Logs are exported by **PromTail** (nodes) and collected by **Loki** (central)
- Data visualization and querying is done by **Grafana**, which pulls data from both **Prometheus** and **Loki**
- Alerts are sent out by **AlertManager**, installed on the Prometheus server, and received by **AM-Executor**, installed on the configuration management server (which also runs **Ansible**)
- Long term storage of metrics is done by **VictoriaMetrics**, installed on the same server as **Loki** to take full advantage of the large storage

Monitoring & Logging (2)



Exported Metrics

node_exporter produces by default a large set of metrics (e.g., CPU load, network activity, disk I/O, system info) and it can be further customised with ad-hoc metrics:

Metric	Purpose	Tool
Temperature HD	check for overheat	S.M.A.R.T. tools
Temperature CPU	check for overheat	IPMI tools
HTCondor jobs stats	check the status and ownership of running jobs	condor_status
VAC jobs stats	check the number and status of virtual machines	check-vacd
Running processes	runtime and resource usage	ps + grep
Number of Reboots	static counter to keep track of unexpected restarts	cron job
Static info	check for unexpected changes in hardware (e.g., memory bank fail)	lscpu, vmstat , lsblk

Metrics produced with any script must be written in a properly formatted text file and placed in a folder where **node_exporter** can scrape them:

`node_exporter_dir:` `"/var/lib/node_exporter/textfile_collector"`

Exported Logs

PromTail tails the content of local logs and sends it to a central **Loki**.

We have chosen this tool (instead than the widely spread **ElasticStearch**) because:

- it well integrates with Grafana
- it is very easy to set-up ...

Logs to scrapes are specified in the configuration (we scrape default system logs* and specific service logs)

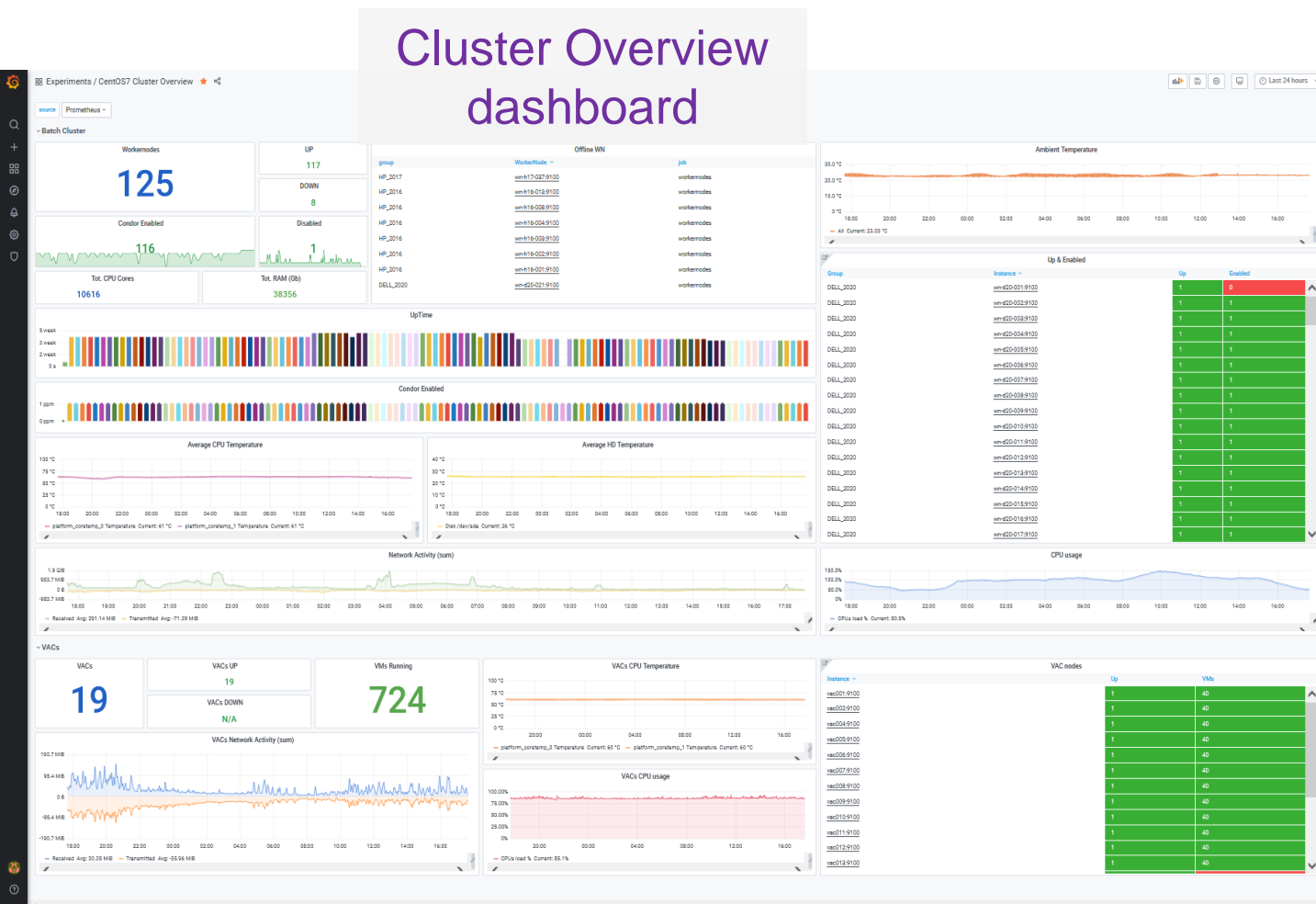
** we do not export the journal because it clogs the system*

Service	Exported Logs
all	/var/log/messages /var/log/secure /var/log/cron /var/log/yum.log
HTCondor	/var/log/condor/
ARC-CE	/var/log/arc/ /var/log/arc/bdii/
ARGUS	/var/log/argus/pap/ /var/log/argus/pdp/ /var/log/argus/pepd/
Squid	/var/log/squid/
Apache	/var/log/httpd/
Ansible	/var/log/ansible/ /var/log/alerts/

Visualization (1)

With a few custom dashboards and over 100+ graphs, it is easy to check the correct functioning of all machines from remote and to quickly identify potential issues.

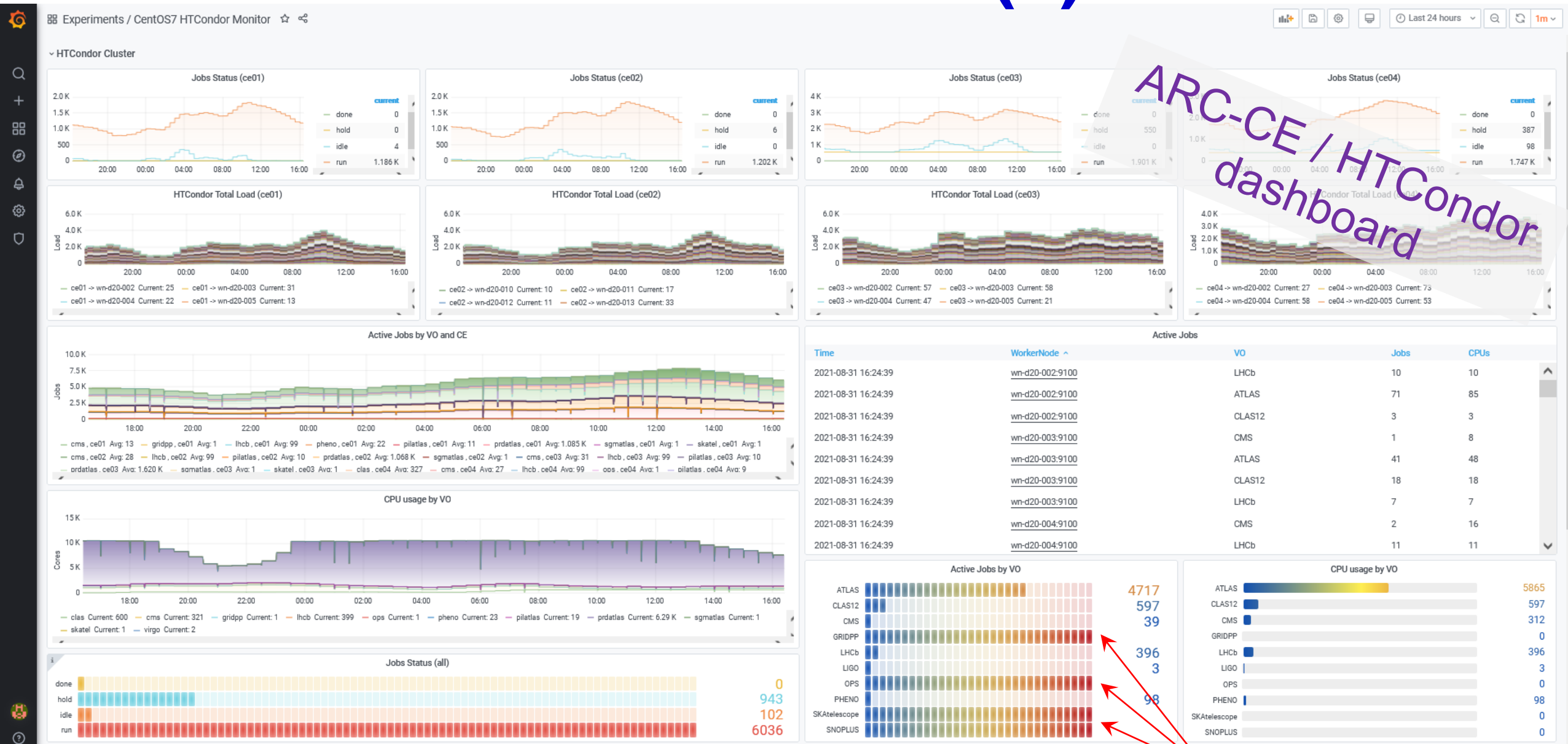
Cluster Overview dashboard



WorkerNode dashboard



Visualization (2)



Grafana issue #18785

Logs DB

Explore

Loki_Victoria

Split

↻

Last 1 hour

🔍

🗑️

Clear All

Run Query

Live

Log labels

rate({job="condor"}) |~ "[Ee]rror"[1h]) > 0.02

Line limit

auto

12.3s

👁️

—

+ Add query

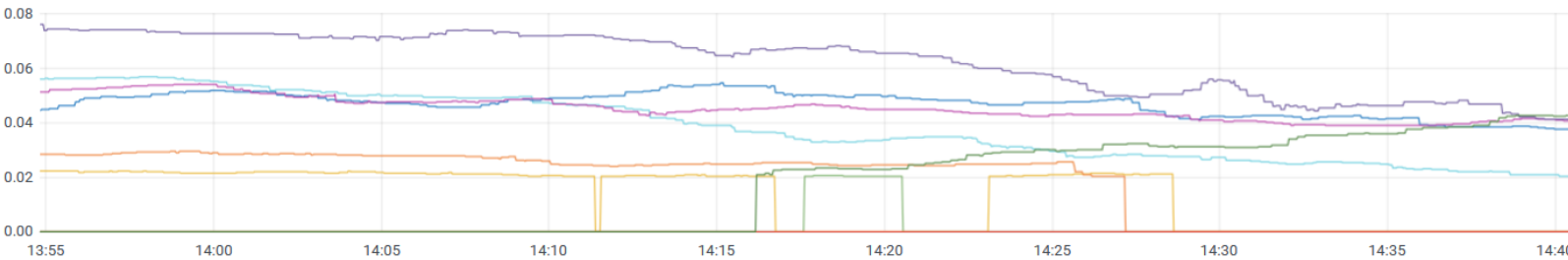
🕒 Query history

🔍 Query inspector

Example query:

rate of HTCondor error messages from all workernodes

Graph



— {filename="/var/log/condor/ProcLog",host="wn-d20-002",job="condor"} — {filename="/var/log/condor/ProcLog",host="wn-d20-003",job="condor"} — {filename="/var/log/condor/ProcLog",host="wn-d20-004",job="condor"}
— {filename="/var/log/condor/ProcLog",host="wn-d20-006",job="condor"} — {filename="/var/log/condor/ProcLog",host="wn-d20-031",job="condor"} — {filename="/var/log/condor/ProcLog",host="wn-d20-032",job="condor"}
— {filename="/var/log/condor/ProcLog",host="wn-d20-038",job="condor"} — {filename="/var/log/condor/ProcLog",host="wn-d20-039",job="condor"} — {filename="/var/log/condor/ProcLog",host="wn-d20-040",job="condor"}

Table

Time	filename	host	job
2021-04-29 14:54:50	/var/log/condor/ProcLog	wn-d20-031	condor
2021-04-29 14:54:50	/var/log/condor/ProcLog	wn-d20-032	condor
2021-04-29 14:54:50	/var/log/condor/ProcLog	wn-d20-038	condor
2021-04-29 14:54:50	/var/log/condor/ProcLog	wn-d20-039	condor
2021-04-29 14:54:50	/var/log/condor/ProcLog	wn-d20-040	condor

Explore

Loki_Victoria

Split

↻

🕒

🔍

🗑️

Clear All

Run Query

Live

Log labels

{job="condor"} |~ "[Ee]rror"

Line limit

auto

4.2s

👁️

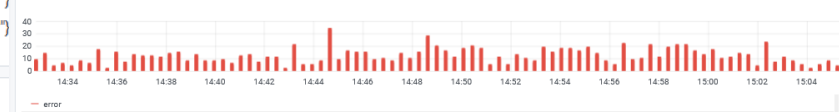
—

+ Add query

🕒 Query history

🔍 Query inspector

Logs



Time

Unique labels

Wrap lines

Dedup

None

Exact

Numbers

Signature

Flip results order

Common labels: condor Limit: 1000 (1000 returned) Total bytes processed: 105 MB

> 2021-04-29 15:05:18 wn-d20-031 04/29/21 15:05:17 error: problem finding resource for 4M4 (DEACTIVATE_CLAIM_FORCEBLY)
> 2021-04-29 15:05:18 wn-d20-031 04/29/21 15:05:17 error: can't find resource with Claimed (c10.1.68.31-96180c2f@socio2845_8387_3#1618511305298058...) -- perhaps this claim was already removed?
> 2021-04-29 15:05:18 wn-d20-031 04/29/21 15:05:16 : GroupTracker (pid = 65126): fopen error: Failed to open file '/proc/65126/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:05:09 wn-d20-031 04/29/21 15:05:09 : GroupTracker (pid = 34544): fopen error: Failed to open file '/proc/34544/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:05:08 wn-d20-038 04/29/21 15:05:08 : GroupTracker (pid = 49590): fopen error: Failed to open file '/proc/49590/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:05:05 wn-d20-040 04/29/21 15:05:05 : GroupTracker (pid = 87069): fopen error: Failed to open file '/proc/87069/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:05:04 wn-d20-039 04/29/21 15:05:04 : GroupTracker (pid = 38917): fopen error: Failed to open file '/proc/38917/cgroup'. Error: No such file or directory (2)

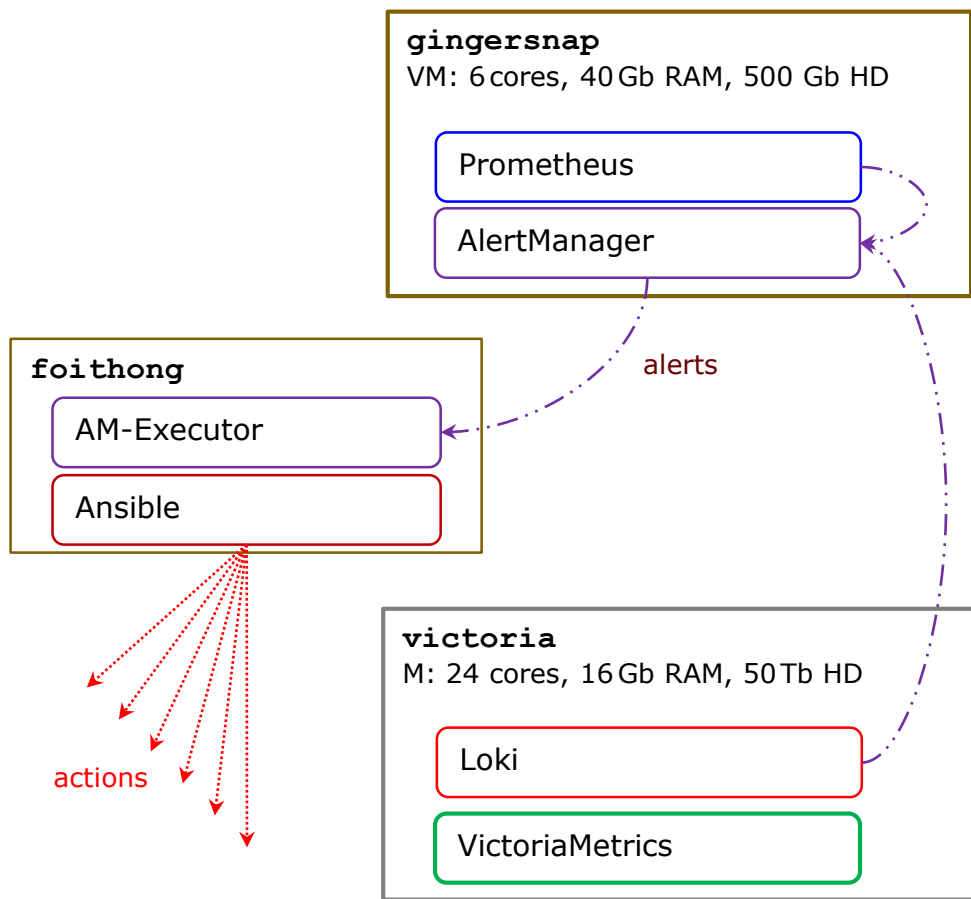
Log Labels:

🔍 🕒 🗑️ filename /var/log/condor/ProcLog
🔍 🕒 🗑️ host wn-d20-031
🔍 🕒 🗑️ job condor
Parsed Fields:
🔍 🕒 🗑️ ts 2021-04-29T15:05:04.124Z
🔍 🕒 🗑️ tsNs 1618795184234549114

> 2021-04-29 15:05:02 wn-d20-039 04/29/21 15:05:02 : GroupTracker (pid = 38020): fopen error: Failed to open file '/proc/38020/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:05:02 wn-d20-039 04/29/21 15:05:02 : GroupTracker (pid = 38021): fopen error: Failed to open file '/proc/38021/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:04:49 wn-d20-038 04/29/21 15:04:49 : GroupTracker (pid = 4241): fopen error: Failed to open file '/proc/4241/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:04:49 wn-d20-038 04/29/21 15:04:49 : GroupTracker (pid = 38959): fopen error: Failed to open file '/proc/38959/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:04:41 wn-d20-031 04/29/21 15:04:41 : GroupTracker (pid = 65866): fopen error: Failed to open file '/proc/65866/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:04:41 wn-d20-038 04/29/21 15:04:41 : GroupTracker (pid = 45663): fopen error: Failed to open file '/proc/45663/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:04:37 wn-d20-038 04/29/21 15:04:36 : GroupTracker (pid = 43673): fopen error: Failed to open file '/proc/43673/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:04:36 wn-d20-038 04/29/21 15:04:36 : GroupTracker (pid = 43673): fopen error: Failed to open file '/proc/43673/cgroup'. Error: No such file or directory (2)
> 2021-04-29 15:04:36 wn-d20-038 04/29/21 15:04:36 : GroupTracker (pid = 43673): fopen error: Failed to open file '/proc/43673/cgroup'. Error: No such file or directory (2)

Alerts & Automation (1)

Alerts are generated according to the outcome of conditional tests involving metrics or queries against certain criteria. Both **Prometheus** and **Loki*** can be configured to trigger alerts when a custom set of conditions is met.

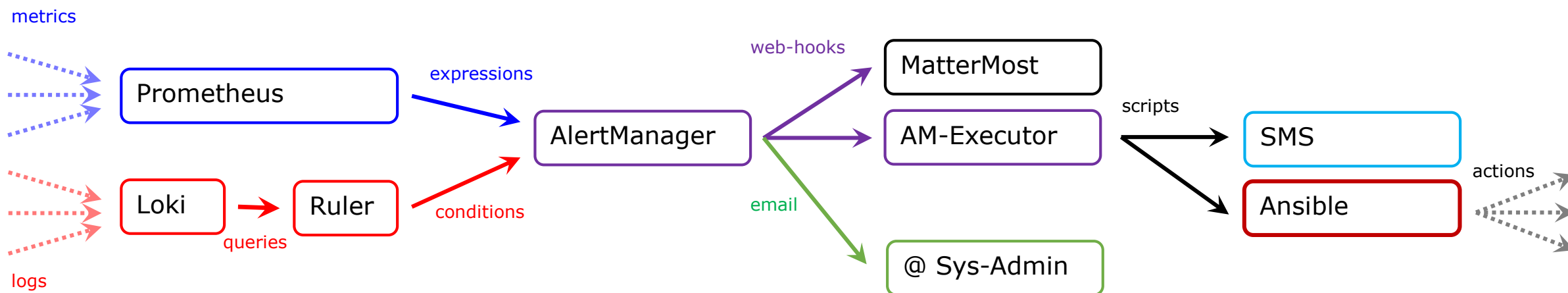


Alerts are handled and sent out by the **AlertManager** ...

... and are received by the **AM-Executor**, which performs some pre-encoded actions.

* Loki's alert system is new in v2.0 and uses a specific plug-in (`ruler`), which in turns calls `AlertManager` ...

Alerts & Automation (2)



In summary, these are the stages of an alert:

- 0) **Prometheus** or **Loki** verify a test condition and triggers **AlertManager**
- 1) An alert messages is generated by **AlertManager** and sent as a web-hook
- 2) **AM-Executor** is an HTTP server that listens for alerts, parses their content, and triggers custom commands matching the alert type and source
- 3) By running **AM-Executor** on the same management server that runs **Ansible**, we can harness the full power of Ansible remote management, with a variety of play-books prepared beforehand

AM-Executor

AM-Executor is configured with a custom script that defines the action to be taken based on the alert:

- Actions implemented in a case-switch
- Multiple alerts handled in a for loop
- The script can call an Ansible play-book or trigger another script (SMS)
- Privileges for Ansible remote management are set by a password-less SSH key *

* This is not the best practice, but so far it has been an ok solution

New addition: jobs draining, (wait), custom update and reboot

```
# case-switch to assign an action
case $AM_ALERTNAME in
    InstanceFull) ansible-playbook -l $AM_NODE /etc/ansible/playbooks/do_cvmfsclean.yml -b
        echo "> ansible-playbook -l $AM_NODE do_cvmfsclean.yml -b" >> $AM_LOGFILE
        ;;

    CpuHot)      ansible-playbook -l $AM_NODE /etc/ansible/playbooks/do_shutdown.yml -b
        echo "> ansible-playbook -l $AM_NODE do_shutdown.yml -b" >> $AM_LOGFILE
        ;;

    HdHot)       ansible-playbook -l $AM_NODE /etc/ansible/playbooks/do_shutdown.yml -b
        echo "> ansible-playbook -l $AM_NODE do_shutdown.yml -b" >> $AM_LOGFILE
        ;;

    NodeLazy)    ansible-playbook -l $AM_NODE /etc/ansible/playbooks/do_clean-enable.yml -b
        echo "> ansible-playbook -l $AM_NODE do_clean-enable.yml -b" >> $AM_LOGFILE
        ;;

    NodeDrain)   ansible-playbook -l $AM_NODE /etc/ansible/playbooks/do_update.yml -b
        echo "> ansible-playbook -l $AM_NODE do_update.yml -b" >> $AM_LOGFILE
        ;;

    WayTooHot)   ansible-playbook /etc/ansible/playbooks/do_shutdown.yml -b
        echo "> ansible-playbook do_shutdown.yml -b" >> $AM_LOGFILE
        ;;

    TooHot)      . /etc/am-executor/send_sms_alert.sh
        echo "> . /etc/am-executor/send_sms_alert.sh" >> $AM_LOGFILE
        ;;

    *)           echo "Alert not recognised: $AM_ALERTNAME !" >> $AM_LOGFILE
        ;;
esac
```

new

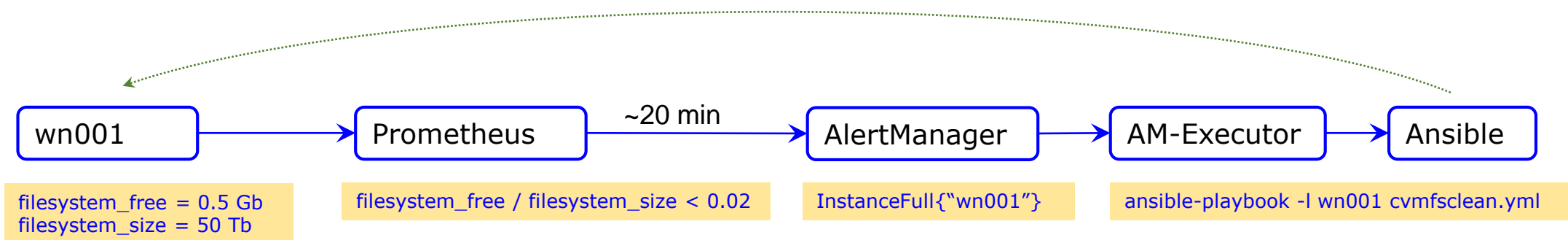
Alert Conditions

So far, only a limited number of alert-cases have been implemented:

Alert condition	Alert Name	Action	Mail Chat	SMS
$\text{avg}(\text{node_sensors_ambient}\{all\}) > 25$	TooHot	/	✓	✓
$\text{avg}(\text{node_sensors_ambient}\{all\}) \geq 30$	WayTooHot	shutdown{all}	✓	
$\text{node_hwmon_temp}\{node\} > 80$ $\vee \text{node_sensors_hddtemp}\{node\} > 35$	CpuHot HdHot	shutdown{node}	✓	
$\frac{\text{node_filesystem_free}\{node\}}{\text{node_filesystem_size}\{node\}} < 0.02$	InstanceFull	clean{node}	✓	
$\text{avg}(\text{node_condor_cpu}\{node\})[1h] \leq 1$ $\wedge \text{avg}(\text{node_condor_cpu})[1h] > 10$	NodeLazy	reboot{node} + enable{node}	✓	

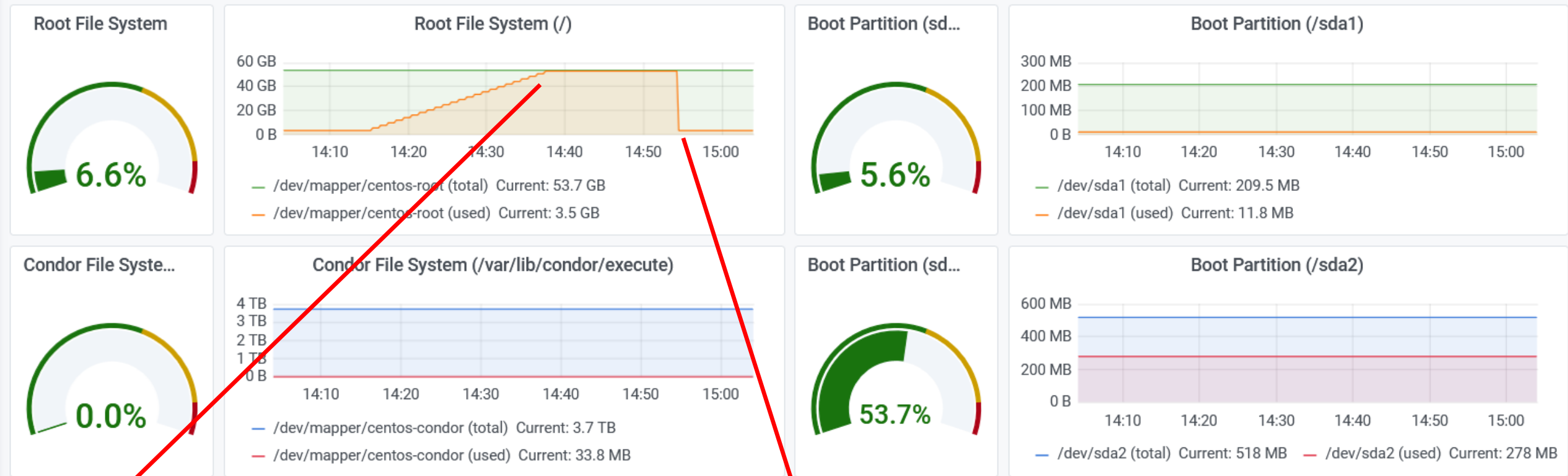
+ NodeDrain ...

Example (**InstanceFull** alert):



Example Alert & Solution

See the InstanceFull Alert being generated and automatically resolved:



MatterMost messages (ClusterComms channel):

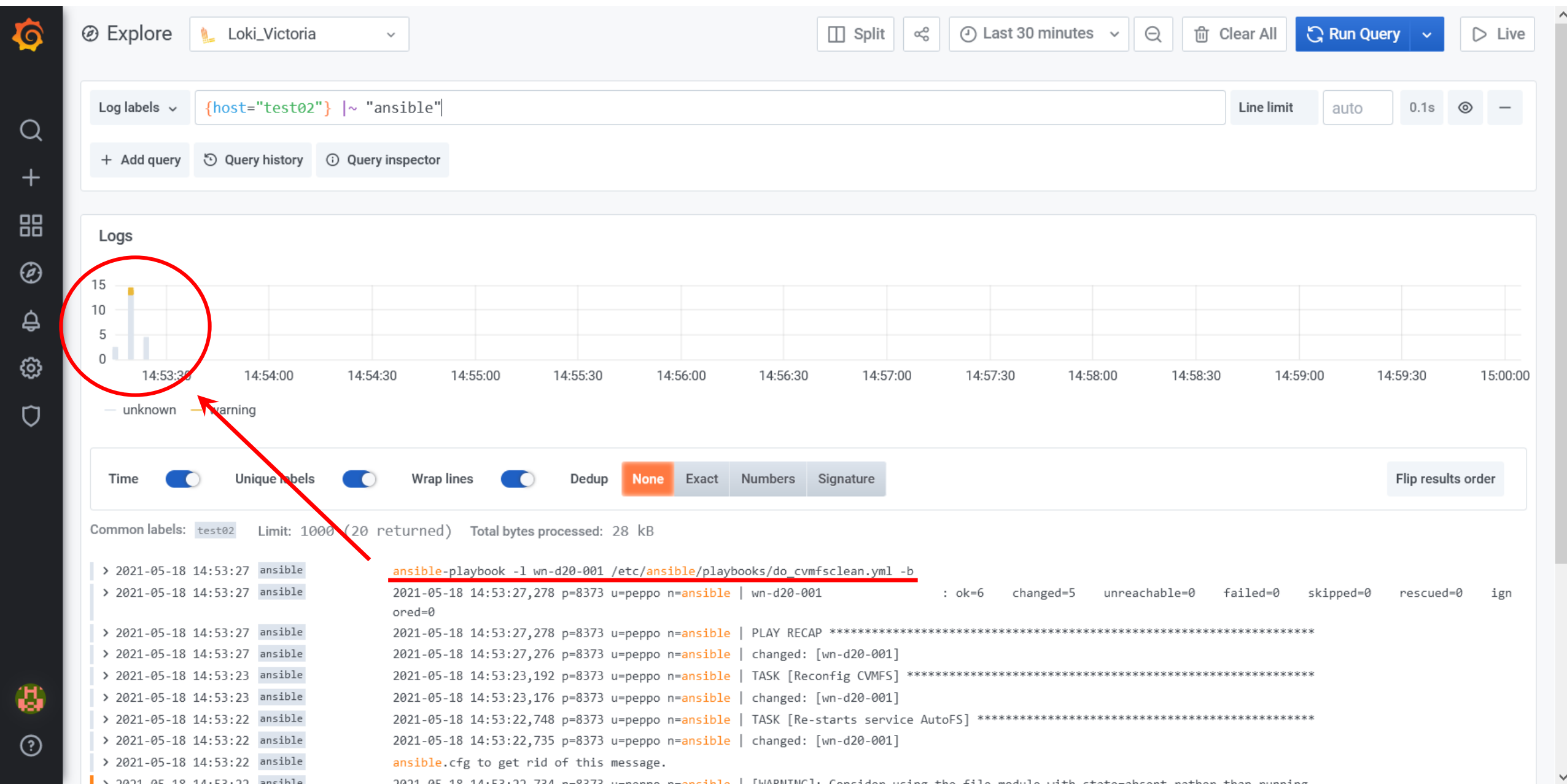
AlertManager BOT 2:53 PM

[FIRING:1] InstanceFull local nodes (/dev/mapper/centos-root xfs new_dell wn-d20-001:9100 workernodes /)

AlertManager BOT 3:04 PM

[RESOLVED] InstanceFull local nodes (/dev/mapper/centos-root xfs new_dell wn-d20-001:9100 workernodes /)

Ansible Logs



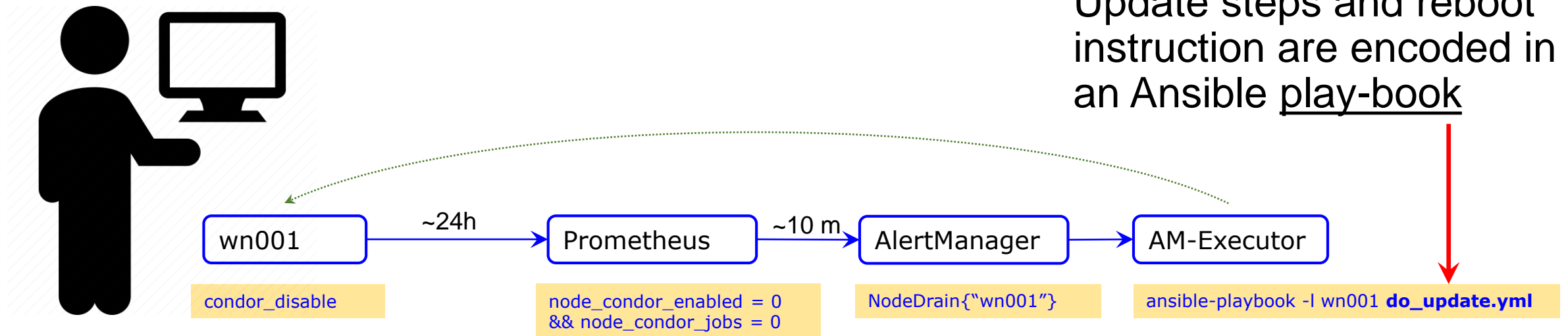
Patching & Updating

Some maintenance tasks, such as HTCondor updates, Kernel updates or other CVE patches, do require the node to be rebooted.

In order not to loose running job, this is normally done in two steps:

- first the node is drained (HTCondor stops accepting new jobs)
- when all running jobs are finished (~24h), the node can be updated, rebooted and then put back online.

Leveraging the existing Alert system, this can be easily automated:



Conclusions & Outlook

The system is being evaluated since Jan. 2021 and it seems well-behaved *:

- ✓ The alerting system can identify issues and send Alerts to site admins via dashboard/emails/SMS/chat.
- ✓ In a few, well known cases, it can trigger simple recovery actions that run automatically (clean HD, reboot, ...)
- ✓ The system can be exploited to cover maintenance tasks, such as installing patches and updates that require the node to be drained and rebooted
- ✓ It was presented at vCHEP and shared with CERN Operational Intelligence #
- Can be extended to identify complex alerts based on combinations of metrics and multi-step recovery actions (e.g., re-provisioning)
- ? Next idea: use Prometheus metrics for Anomaly Detection ...
- ? To be added: Networking metrics (sFlow) ...

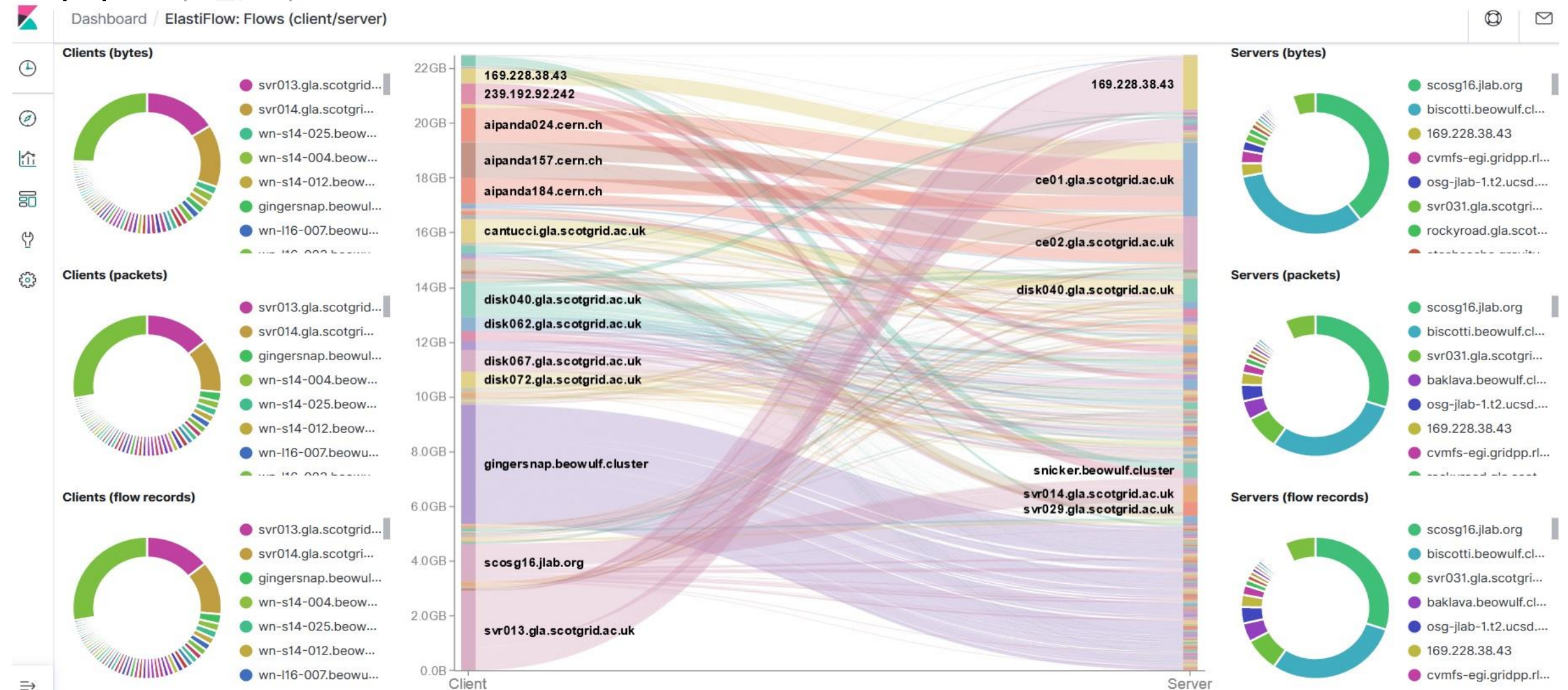
* It is well known that giving full control to the machines without proper evaluation will eventually lead to



Thanks.

ElasticFlow

We did some experimenting with **ElasticSearch** modules to analyze network flows from



... but this is another story.

Example Procedures

Create a new workernodes (assuming the node is cabled and plugged in):

- Fill in MAC, IP and HostName (in Ansible), then run the play-book to update the network
vi roles/dhcp/files/hosts.conf roles/dnsmasq/files/10.0.0.0_8
ansible-playbook update_networking.yml
- Create a PXE system profile, then boot the machine to install OS
vi systems/nodes/wn00x.yml
systemctl restart pxecli
- Finally run ansible to install the software and configure the workernode
ansible-playbook -l wn00x workernodes.yml

Add a new VO to the standard job submission system:

- Fill in the VO coordinates in a proper Ansible role, then run it
cp -r roles/vo-template roles/vo-xxx
vi roles/vo-xxx/files/xxx.grid-mapfile
ansible-playbook add_vo.yml -extra-vars "vo=xxx"

Clean CVMFS cache and re-enable node **00x**:

- Just run the ansible play-book
ansible-playbook -l wn00x cleanCVMFS.yml

Occasional Kernel or HTCondor update (needing a reboot):

- Drain jobs (24h), custom play-book , reboot, re-enable ...

Example Logging

As an example, these are the bits in the **PromTail** configuration (promtail.yml) that export general and HTCondor logs:

```
1  # PromTail Configuration defines the rules for scraping local logs
2
3  server:
4    http_listen_port: 9080
5    grpc_listen_port: 0
6
7  # Positions
8  positions:
9    filename: /tmp/positions.yaml
10
11 # Loki Server URL
12 clients:
13   - url: http://{{ central_log_server }}:3100/loki/api/v1/push
14
15 scrape_configs:
16   ## Common Logs
17   - job_name: messages      # log messages (/var/log/messages)
18     static_configs:
19       - targets:
20         - "{{ inventory_hostname }}:9080"
21         labels:
22           job: messages
23           host: "{{ inventory_hostname }}"
24           __path__: /var/log/messages
25   # ...
```

General Logs (system messages)

```
27
28 ## Service Logs
29 - job_name: condor
30   static_configs:
31     - targets:
32       - "{{ inventory_hostname }}:9080"
33       labels:
34         job: condor
35         host: "{{ inventory_hostname }}"
36         __path__: /var/log/condor/*og
```

HTCondor logs:

/var/log/condor/

Alert Pending

Prometheus Alerts Graph Status ▾ Help Classic UI

✓ Inactive (6)

✓ Pending (2)

✓ Firing (1)

Show annotations

/etc/prometheus/alertrules.yml > toohot

> TooHot_DC (0 active)

/etc/prometheus/alertrules.yml > vcchep

> WayTooHot (0 active)

> NoJobs (0 active)

> NoVACjobs (0 active)

> CpuHot (1 active)

> HdHot (0 active)

> InstanceFull (1 active)

> NodeLazy (1 active)

> VacLazy (0 active)

▼ InstanceFull (1 active)

name: InstanceFull

expr: node_filesystem_free_bytes{device=~"/dev/mapper/.*",job="workernodes"} / node_filesystem_size_bytes{device=~"/dev/mapper/.*",job="workernodes"} < 0.02

for: 10m

labels:

severity: local

type: nodes

annotations:

description: {{ \$labels.instance }} has the {{ \$labels.device }} partition full (mountpoint = {{ \$labels.mountpoint }}). Clean it up!

summary: Instance {{ \$labels.instance }} is full

Labels

	State	Active Since	Value
alertname=InstanceFull device=/dev/mapper/centos-root fstype=xfs group=new_dell instance=wn-d20-001:9100 job=workernodes mountpoint=/ severity=local type=nodes	PENDING	2021-05-18T13:38:16.967886192Z	0.013029280654616513

Annotations

description

wn-d20-001:9100 has the /dev/mapper/centos-root partition full (mountpoint = /). Clean it up!

summary

Instance wn-d20-001:9100 is full

Alert Firing

Prometheus Alerts Graph Status ▾ Help Classic UI

✓ Inactive (6)

✓ Pending (1)

✓ Firing (2)

Show annotations

/etc/prometheus/alertrules.yml > toohot

> TooHot_DC (0 active)

/etc/prometheus/alertrules.yml > vchep

> WayTooHot (0 active)

> NoJobs (0 active)

> NoVACjobs (0 active)

> CpuHot (1 active)

> HdHot (0 active)

> InstanceFull (1 active)

> NodeLazy (1 active)

> VacLazy (0 active)

▼ InstanceFull (1 active)

name: InstanceFull

expr: node_filesystem_free_bytes{device=~"/dev/mapper/.*",job="workernodes"} / node_filesystem_size_bytes{device=~"/dev/mapper/.*",job="workernodes"} < 0.02

for: 10m

labels:

severity: local

type: nodes

annotations:

description: {{ \$labels.instance }} has the {{ \$labels.device }} partition full (mountpoint = {{ \$labels.mountpoint }}). Clean it up!

summary: Instance {{ \$labels.instance }} is full

Labels	State	Active Since	Value
<div>alertname=InstanceFull</div> <div>device=/dev/mapper/centos-root</div> <div>fstype=xfs</div> <div>group=new_dell</div> <div>instance=wn-d20-001:9100</div> <div>job=workernodes</div> <div>mountpoint=/</div> <div>severity=local</div> <div>type=nodes</div>	FIRING	2021-05-18T13:38:16.967886192Z	0.013029662310698584

Annotations

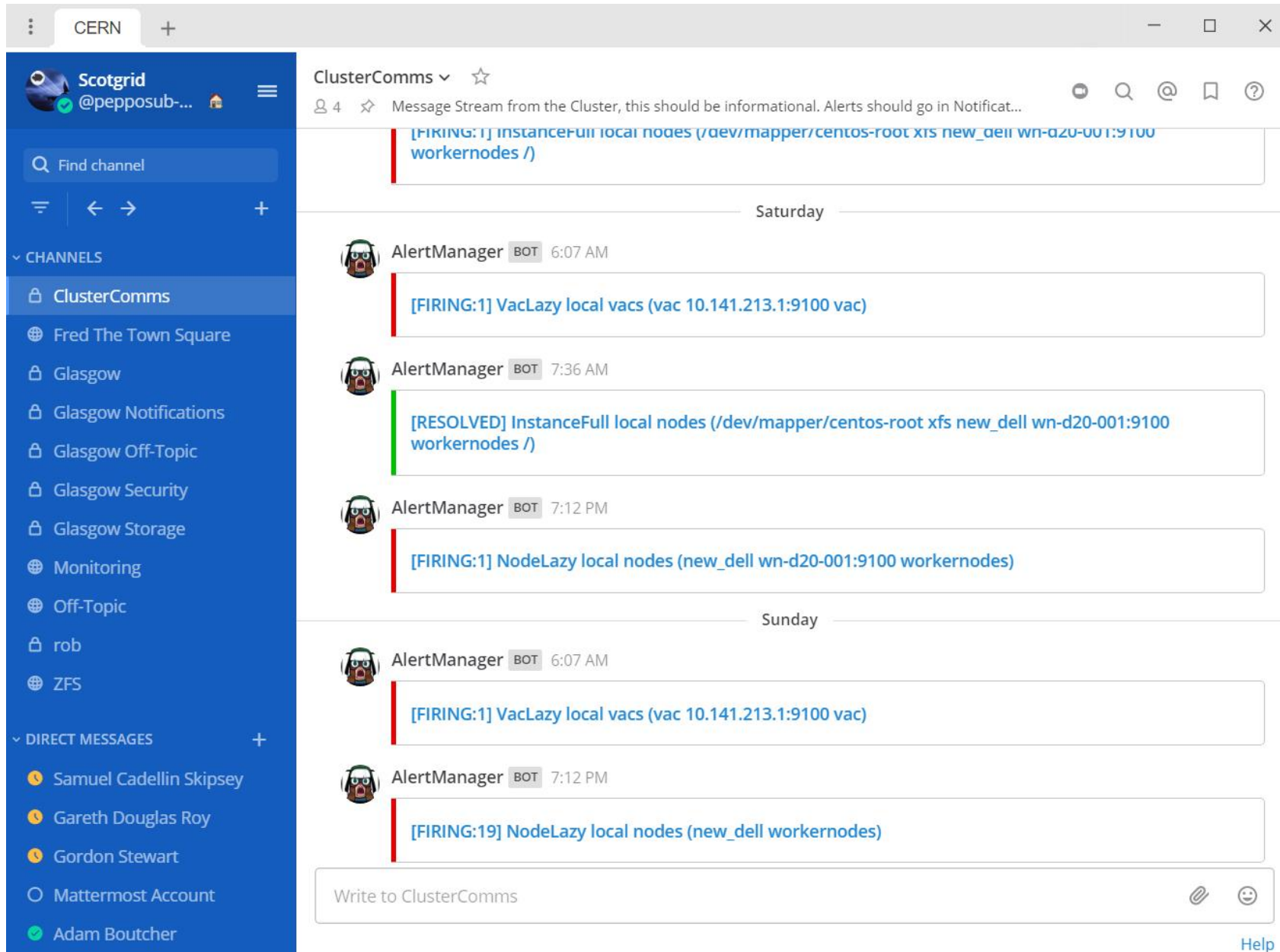
description

wn-d20-001:9100 has the /dev/mapper/centos-root partition full (mountpoint = /). Clean it up!

summary

Instance wn-d20-001:9100 is full

MatterMost Alerts



The screenshot displays a Mattermost web interface. On the left is a blue sidebar with a user profile for 'Scotgrid' and a search bar. Below the search bar is a 'CHANNELS' section with a list of channels: 'ClusterComms' (highlighted with a red arrow), 'Fred The Town Square', 'Glasgow', 'Glasgow Notifications', 'Glasgow Off-Topic', 'Glasgow Security', 'Glasgow Storage', 'Monitoring', 'Off-Topic', 'rob', and 'ZFS'. Below this is a 'DIRECT MESSAGES' section with a list of contacts: 'Samuel Cadellin Skipsey', 'Gareth Douglas Roy', 'Gordon Stewart', 'Mattermost Account', and 'Adam Boutcher'. The main area shows the 'ClusterComms' channel with a message stream. The stream includes a header for 'Saturday' and three messages from 'AlertManager BOT'. The first message is '[FIRING:1] instanceFull local nodes (/dev/mapper/centos-root xfs new_dell wn-d20-001:9100 workernodes /)'. The second message is '[RESOLVED] InstanceFull local nodes (/dev/mapper/centos-root xfs new_dell wn-d20-001:9100 workernodes /)'. The third message is '[FIRING:1] NodeLazy local nodes (new_dell wn-d20-001:9100 workernodes)'. Below these is a header for 'Sunday' and two more messages from 'AlertManager BOT'. The first message is '[FIRING:1] VacLazy local vacs (vac 10.141.213.1:9100 vac)'. The second message is '[FIRING:19] NodeLazy local nodes (new_dell workernodes)'. At the bottom is a text input field labeled 'Write to ClusterComms' with a 'Help' link.

CERN

Scotgrid
@pepposub-...

Find channel

CHANNELS

ClusterComms

Fred The Town Square

Glasgow

Glasgow Notifications

Glasgow Off-Topic

Glasgow Security

Glasgow Storage

Monitoring

Off-Topic

rob

ZFS

DIRECT MESSAGES

Samuel Cadellin Skipsey

Gareth Douglas Roy

Gordon Stewart

Mattermost Account

Adam Boutcher

ClusterComms

Message Stream from the Cluster, this should be informational. Alerts should go in Notificat...

[FIRING:1] instanceFull local nodes (/dev/mapper/centos-root xfs new_dell wn-d20-001:9100 workernodes /)

Saturday

AlertManager BOT 6:07 AM

[FIRING:1] VacLazy local vacs (vac 10.141.213.1:9100 vac)

AlertManager BOT 7:36 AM

[RESOLVED] InstanceFull local nodes (/dev/mapper/centos-root xfs new_dell wn-d20-001:9100 workernodes /)

AlertManager BOT 7:12 PM

[FIRING:1] NodeLazy local nodes (new_dell wn-d20-001:9100 workernodes)

Sunday

AlertManager BOT 6:07 AM

[FIRING:1] VacLazy local vacs (vac 10.141.213.1:9100 vac)

AlertManager BOT 7:12 PM

[FIRING:19] NodeLazy local nodes (new_dell workernodes)

Write to ClusterComms

Help

Git & Wiki

Ansible repository on our internal Git:
~70 Ansible roles
~50 Ansible play-books (incl. maintenance)

ScotGrid > Ansible > Details



Ansible
Project ID: 1

🔔 Star 1 🍴 Fork 0

636 Commits 2 Branches 0 Tags 296.2 MB Files 296.2 MB Storage

Ansible roles to build up the whole Glasgow site :)

master ansible / + History Find file Web IDE Clone



VAC certificate role added
Emanuele Simili authored 6 days ago

4d860844

Add README Add LICENSE Add CHANGELOG Add CONTRIBUTING Enable Auto DevOps
Add Kubernetes cluster Set up CI/CD

Name	Last commit	Last update
cve	fixed loki installer	1 month ago
docs	updated pw file: mailing lists; updated perfSONAR installer: boot script for dual in...	1 week ago
inventory	further development of alerting rules etc. (post vCHEP2021)	1 month ago
playbooks	VAC certificate role added	6 days ago
roles	VAC certificate role added	6 days ago
vars	further development of alerting rules etc. (post vCHEP2021)	1 month ago
.gitignore	.gitignore fix	1 year ago
ansible.cfg	logs	1 month ago

Internal wiki with
detailed procedures
and code snippets
for cluster built:
~50 pages

