WLCG Experiments Test Framework Update

Marian Babik, CERN
IPv6 F2F, June 2021
News

- **Checkmk 2.0** was released
  - Adds Kubernetes, AWS, Azure, VMware monitoring, improves Docker monitoring and integrates with Prometheus, Grafana. Also introduces new network monitoring (ntop).
  - REST API is introduced in this release

- **ETF**
  - New job submission framework (jess) and new worker node testing now in production for all the experiments
  - CREAM-CE testing was retired

- **New ETF central instance**
  - Shows tests from both IPv4 and IPv6 nodes for ATLAS and CMS
  - There are many CEs in dual-stack, but blocking access over IPv6

- **CMS now uses native arc client and ARC’s AREX backend to submit jobs**
  - Motivated by high latency of the HTCondor/ARC GSIFTP backend
  - The plan is to migrate back to HTCondor once it supports submission via ARC REST

- **ETF -> MONIT/Sitemon interface**
  - Agreed on a plan for Sitemon to integrate IPv6 metrics
## Plugins/Tests

<table>
<thead>
<tr>
<th>Plugins</th>
<th>Users/Experiments</th>
<th>Maintained by</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Job Submission</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARC, HTCONDOR-CE</td>
<td>LHCb, ALICE, ATLAS, CMS, DUNE</td>
<td>ETF</td>
</tr>
<tr>
<td><strong>Worker Nodes</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATLAS (3), CMS (11), LHCb (7), DUNE(1)</td>
<td>ATLAS, CMS, LHCb, DUNE</td>
<td>ATLAS, CMS, LHCb, DUNE</td>
</tr>
<tr>
<td><strong>Storage</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GFAL2 (SRM, gsiftp, XRoot, HTTP)</td>
<td>ATLAS</td>
<td>ATLAS</td>
</tr>
<tr>
<td>GFAL2 (SRM)</td>
<td>CMS</td>
<td>CMS</td>
</tr>
<tr>
<td>XRoot**, new HTTP/webdav</td>
<td>CMS</td>
<td>CMS</td>
</tr>
<tr>
<td>HTTPs/WebDAV**</td>
<td>HTTP TF*</td>
<td>HTTP TF*</td>
</tr>
<tr>
<td><strong>Network</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>perfSONAR infrastructure**</td>
<td>WLCG Network Throughput WG</td>
<td>OSG, WLCG</td>
</tr>
</tbody>
</table>

**Uses new library for writing plugins ([python-nap](https://python-nap.org))  *Probe no longer supported by GFAL2 team**
Challenges and Plans

● Started working on support for tokens
  ○ IAM infrastructure on dual-stack will be needed
  ○ Plan is to integrate token support alongside x509

● Updates to Checkmk 1.6 and C8
  ○ Added new authentication backend, which can be extended to support OpenID/SAML
  ○ psetf.aglt2.org and etf.cern.ch already running Checkmk 1.6
  ○ Evaluating C8 middleware support

● WN IPv6-only testing
  ○ Requires a developing a smoke test to check and test IPv6 connectivity on the WN

● Python 3
  ○ ETF infrastructure is ready, but some of the probes will need to be migrated
  ○ Both CMS and LHCb plan to work on storage probes (but not much progress recently)

● IPv6 results in Sitemon/MONIT

● K8s prototyping ongoing (to be tested in QA later on)
  ○ Plan is to migrate to full Auto DevOps model (with Helm/Flux) and consolidate resources
Questions?

Docs: https://etf.cern.ch/docs/latest/
New central instance: https://etf-10.cern.ch/etf/check_mk/
Instances (access requires IGTF/x509 cert loaded in the browser):

- **CMS production**
  - CMS QA IPv6
  - CMS QA
  - Code: CMS gitlab

- **ATLAS production**
  - ATLAS QA IPv6
  - ATLAS QA
  - Code: ATLAS gitlab

- **LHCb production**
  - LHCb QA
  - Code: LHCb gitlab

- **ALICE production**
  - ALICE QA
  - Code: ALICE gitlab

- **pS production**
  - pS QA
  - Code: pS gitlab

ETF framework

ETF core containers  ETF Job Submission (Jess)
ETF nagios plugins lib. NAP  ETF rule-based configuration (ncgx)
ETF support channels: GGUS: Grid Monitoring or etf-support@cern.ch (SNOW)
Summary

- ETF is a container-based application combining open source software with a set of frameworks and APIs to provide flexible testing suite
- Easy to extend, re-locate and support new experiments and technologies
- Supported as part of the CERN IT Monitoring Stack
- Currently deployed at CERN for five experiments
  - Supporting IPv4-only and IPv6-only testing
  - Experiments contacts have access in case they need to debug and/or follow up on issues
  - Central instance provides a site-level view (one place to see results from all experiments)
- Additional deployment at OSG for perfSONAR infrastructure monitoring
  - Strong interest from other communities to have this available as a generic tool
- Feedback welcome via standard support channels or directly
  - Experiments priorities for different tasks and new use cases best communicated via tickets
  - Technical issues/merge requests can be added directly via gitlab
Jess

- **Simple job submission framework used to develop ETF JS plugins**
  - Independent (not tied to Nagios), rewrite from scratch

- **Pluggable** - easy to extend to support different submission systems
  - Direct submission to ARC, CREAM and HT-Condor-CE
  - Submissions via local HT-Condor (to ARC/CREAM/HTCondorCE and potentially other backends supported by HT-Condor)
  - **Submissions via remote HT-Condor pool**
  - ETF can also host a **local HT-Condor pool** (in a separate container) to which remote startds can connect (implements pull-based submission model - tested for CMS DODAS)

- **Generic job tracking and monitoring**
  - Currently tracking a single job per CE; can also be extended to track multiple jobs/CE
  - Manual re-scheduling of job submissions via web interface improved
  - Full log(s) of the running job (details depend on the backend)

- **Support for configuration/env on the worker nodes**
  - This can be configured by the experiments in the ETF core plugin (per host/service/site)

- **Drops length limit on the worker node results and job submission results**
Micro-framework to execute tests on the worker nodes
  - Moving away from a statically compiled nagios binary

Supports basic scheduling of tests
  - Can run tests in parallel (configurable), can timeout/kill runaway tests
  - Initial support for alternate schedule of tests (execute once every 3 runs)

Runs nagios standard compliant tests
  - Supports performance metrics readout (numerical values)

Configured directly from ETF frontend

WN environment setup
  - Using env/config passed from JESS (via env file) - easy to pass variables directly from frontend to the WN (like sitename, paths, originating CE/queue, etc.)

Support different backends for metric output
  - Directory queues, message queues, json, http upload, etc.

Can also run as a standalone component

Supports py 2.6, 2.7 and 3.4+; statically compiled version also avail.