Hewlett Packard
Enterprise

Secure Workflows for HPC — a PoC

Harvey Richardson, HPE HPC/AI EMEA Research Lab

September 5, 2022

HPC/AI EMEA Research Lab

Focused on collaborations, technical relationships and new technologies with a view to creating reusable PoCs and IP in Europe relevant to future products

Research Interests

- HPC, Cloud, AI, Quantum
- Data movement, analysis, and workflows
- Heterogeneous computing and novel accelerators
- Programming languages and models
- Compilers and mathematical optimisation
- Performance portability, security, and containerisation
- Energy efficiency and sustainability

Engagement Models

- Centres of Excellence
  - GW4
  - ARCHER2
  - LUMI (Secure Workloads in HPC)
- Advanced Collaboration Centres
- Value-add projects
- Joint-funded research projects
- Nationally/internationally funded research projects
- Ph.D. and Placements

Problem Statement

Requirements

"Users should be able to

1. *run an application and*
2. *access protected data on the HPC system that*
3. *can only be executed on (a designated subset of) the compute nodes*
4. *(only once), and*
5. *can not be inspected in transport or in storage*
6. *with audit trail."*

Interpretation

(First attempt)

1. containerized application
2. encrypted data
3. dedicated set of attested nodes with per-allocation key for application
4. Per-job key
5. Decryption of payload on attested (=trusted) compute nodes
6. auditable key management, system attestation

Supporting Technologies

- Application packaging, signing and encryption
- Secrets Management Infrastructure
- Secure Identity Plane for platform and workload attestation
- Per-user/job encrypted file system
- Secure Execution Enclave

Supporting Technologies

- Application packaging, signing and encryption
  - Container technologies (docker/Singularity)
- Secrets Management Infrastructure
  - HashiCorp Vault
- Secure Identity Plane for platform and workload attestation
  - SPIFFE/SPIRE
- Per user/job encrypted file system
- Secure Execution Enclave
  - Secure Boot
  - BIOS/ Processor and Memory security features
  - Network separation: VLANs, VNI, per-job encrypted segment?
  - Encrypted FS access – ceph? Lustre?

PoC with CSC and AMD

Content

- Run Speech defect recognition code
  - Data decryption only on compute nodes
    - inside tmpfs, cleansed after run
  - SPIRE-attestation of resources
    - Providing trusted enclave
  - Per-job encryption key (one-time) of payload
    - Provides replay-protection
  - Per-user encryption key (personalized) for results

Status: First demo

```
seb@u176i026:/data/seb/secure_workloads_poc/hellotrunk> /opt/slurm/bin/srun -N1 ./3_run -s trunkfished_my_app_sec.sif -c container_key.pem

***trunkfish entry policy engine starts

***trunkfish entry policy engine ends

***secure computation finished and result stored into /trunkfish/output/my_secret_out***

***trunkfish exit policy engine starts


encrypting output... filelist
----------------------------
my_secret_out
----
encrypting output tarball...
[note]: use the following command later to decrypt
age -i data_key.pem output.tar.enc
...

***trunkfish exit policy engine ends

copied '/dev/shm/out-107/output.tar.enc' -> '/data/seb/output-107.tar.enc'
removed '/dev/shm/out-107/output.tar.enc'
seb@u176i026:/data/seb/secure_workloads_poc/hellotrunk>
seb@u176i026:/data/seb/secure_workloads_poc/hellotrunk>
seb@u176i026:/data/seb/secure_workloads_poc/hellotrunk>
seb@u176i026:/data/seb/secure_workloads_poc/hellotrunk>
```

PoC Workflow

- Start with application container (in our case containing the Kaldi application)

- Generate encryption keys for container, input and output data
- Ship keys to vault on HPC system, set as not reusable and only accessible from one compute node
- Framework builds encrypted container: developer supplies entry point for execution
- Specific paths are used for use in container and for input and output data outside container
- Container entry point set which implements policies (in/out) and then executes developer entry point
- Framework builds container (OCI to encrypted SIF)
- Container shipped to HPC system
- Input data encrypted and sent to HPC system

(On HPC System)

- Application run by Slurm on the one node allowed, keys only available on that node.
- Output data encrypted (container teardown)

Next Steps / Question

Next steps

- Further develop Secure Enclave part
  - more secure hardware platform
- Attempt a distributed PoC with external party
  - Needs federation of user identity
  - Can user get credentials for access to remote data to your HPE resource (3 parties)?

Question

- Is anyone interested in reaching out from their TRE to an external HPC resource

Acknowledgements

Groups involved:

- HPE HPC/AI EMEA Research Lab
- LUMI CoE
- CSC
- AMD

Thank you