

HTCondor-CE: Configuration

ISGC 2019 - Taipei, Taiwan

Brian Lin

University of Wisconsin — Madison



HTCondor Configuration Refresher

- HTCondor-CE configuration lives in `/etc/condor-ce/`
 - Main configuration file in `/etc/condor-ce/condor_config`
 - Add local configuration to `/etc/condor-ce/config.d/` (files processed in lexicographic order)
- `condor_ce_config_val` to inspect config values
- `condor_ce_reconfig` to apply new configuration

Authentication and Authorization

- Authentication is configured via the HTCondor-CE unified mapfile `/etc/condor-ce/condor_mapfile`
 - One mapping per line with the following format:
`<AUTH METHOD> <AUTH NAME> <HTCONDOR NAME>`
 - Supports perl-compatible regular expressions
 - Selected mapping is determined by first-match
- HTCondor names (`<USERNAME>@<DOMAIN>`) determine authorization level
 - `<hostname>@daemon.htcondor.org`: authorized as a daemon
 - `.*@users.htcondor.org`: authorized to submit jobs
 - `GSS_ASSIST_GRIDMAP`: a special value telling HTCondor-CE to call out to another service for user mapping, e.g. LCMAPS, Argus
- <http://research.cs.wisc.edu/htcondor/manual/v8.8/Security.html#x36-2850003.8.4>

Authentication and Authorization

An example from our HTCondor-CE, lhcb-ce.chtc.wisc.edu:

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[-.A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claiamtobe
FS (.* ) \1
```

Authentication and Authorization

Authentication method (GSI is the default auth method for remote clients)

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[-.A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claiamtobe
FS (.* ) \1
```

Authentication and Authorization

The “authentication name”. In this case, this is the subject distinguished name (DN) of the host certificate:

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[- .A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claimtobe
FS (.* ) \1
```

Authentication and Authorization

The HTCondor name:

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[-.A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claiamtobe
FS (.* ) \1
```

Authentication and Authorization

Put it all together, this line allows all the daemons on the HTCondor-CE host to authenticate with each other.

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[-.A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claiamtobe
FS (.* ) \1
```


Authentication and Authorization

Mapping via VOMS FQANs are possible in the authenticated name

- <SUBJECT DN>, <VOMS FQAN 1>, . . . , <VOMS FQAN N>
- This line maps all X.509 credentials with an LHCb primary VOMS FQAN to the nu_lhcb user

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[-.A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claimtobe
FS (.* ) \1
```

Authentication and Authorization

Explicit mapping for a single user:

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[-.A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claiamtobe
FS (.* ) \1
```

Authentication and Authorization

Callout to external service configured via `/etc/grid-security/gsi-authz.conf`:

```
globus_mapping liblcas_lcmaps_gt4_mapping.so lcmaps_callout
```

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
```

```
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
```

```
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
```

```
GSI (.* ) GSS_ASSIST_GRIDMAP
```

```
GSI "(/CN=[- .A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
```

```
CLAIMTOBE .* anonymous@claimtobe
```

```
FS (.* ) \1
```

Authentication and Authorization

Unauthorized fallbacks

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[-.A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claiamtobe
FS (.* ) \1
```

Authentication and Authorization

Finally, map local accounts to themselves. The **UID_DOMAIN** (@users.htcondor.org) is automatically appended, i.e. they have submit privileges

```
GSI "/DC=org/DC=incommon/<snip>/CN=lhcb-ce.chtc.wisc.edu" lhcb-ce.chtc.wisc.edu@daemon.htcondor.org
GSI ".*,/lhcb/Role=pilot/Capability=.*" nu_lhcb@users.htcondor.org
GSI "/DC=org/DC=cilogon/<snip>/CN=Brian Lin A106521" blin@users.htcondor.org
GSI (.* ) GSS_ASSIST_GRIDMAP
GSI "(/CN=[- .A-Za-z0-9/= ]+)" \1@unmapped.htcondor.org
CLAIMTOBE .* anonymous@claiamtobe
FS (.* ) \1
```

Non-HTCondor Configuration

Configure the Batch GAHP (a.k.a. BLAHP) via

```
/usr/libexec/condor/glite/etc/batch_gahp.config
```

1. Disable Batch GAHP delegation of proxy certificates:

```
blah_disable_wn_proxy_renewal=yes
```

```
blah_delegate_renewed_proxies=no
```

```
blah_disable_limited_proxy=yes
```

2. If your batch system tools exist outside of `/usr/bin`, also edit `*_binpath` (e.g., `slurm_binpath=/opt/slurm/bin`). **NOTE** no spaces around the '='!

Log Levels

- Useful for temporary debugging
- Log level can be adjusted per daemon (e.g, `SCHEDD_DEBUG`) or across all daemons (`ALL_DEBUG`)
- Most common, helpful log levels for HTCondor-CE:
 - `D_CAT D_ALL :2` - shows the log level for each line (helpful for debugging HTCondor bugs!) and increases the log level of general messages
 - `D_SECURITY` - show authentication messages
 - `D_NETWORK` - show messages for TCP/UDP connections

Information Services

- An HTCondor-CE central collector requires no extra configuration, just start the `condor-ce-collector` service!
- To report to a central collector, specify the hostname and port. For example, `CONDOR_VIEW_HOST = collector.opensciencegrid.org:9619`
- Advertise Schedd ads to the central collector via TCP:
`CONDOR_VIEW_CLASSAD_TYPES = Scheduler`
`UPDATE_COLLECTOR_WITH_TCP = true`
- Add arbitrary attributes to the Schedd ad:
`FOO = "Bar"`
`SCHEDD_ATTRS = $(SCHEDD_ATTRS) FOO`

Configuring Job Routes

Job Router Configuration

- Declare your site policy
- Each route is described with ClassAds
- Job routes are constructed by combining each entry in `JOB_ROUTER_ENTRIES` with the `JOB_ROUTER_DEFAULTS`
- Each job is compared to each job route's requirements expression (`Requirements = True` by default) in order

```
$ condor_ce_job_router_info
-config
Route 1
Name      : "Local_Condor"
Universe  : 5
MaxJobs   : 10000
MaxIdleJobs : 2000
GridResource :
Requirements : true
ClassAd    :
    [
<snip>
```

Job Router Configuration

- For HTCondor batch systems, these configuration macros are required:
 - `JOB_ROUTER_SCHEDD2_NAME` - the hostname of the CE host
 - `JOB_ROUTER_SCHEDD2_POOL` - the <HOST>:<PORT> of the local HTCondor central manager
 - `JOB_ROUTER_SCHEDD2_SPOOL` - location of the local SPOOL directory (`condor_config_val SPOOL`)
- Configuration guide:
<https://opensciencegrid.org/docs/compute-element/job-router-recipes/>

Job Router ClassAds

Special job route functions are used to transform jobs, evaluated in the following order.

1. Copy an attribute from the original job ad to the routed job ad:

```
copy_foo = "original_foo";
```

2. Delete an attribute from the original job ad from the routed job ad:

```
delete_foo = True;
```

3. Set an attribute in the routed job ad to a value. If set to an expression, the expression is evaluated in the context of the routed job.

```
set_requirements = (OpSys == "LINUX");
```

4. Set an attribute in the routed job ad to value that is evaluated in the context of the original job ad.

```
eval_set_Experiment = strcat("cms.", Owner);
```

Job Router Classads

Use `set_*` or `eval_set*` for the following resource requests

- `default_xcount` to set the default number of cores
- `default_maxMemory` to set the default maximum memory (in MB)
- `default_maxWalltime` to set the default maximum walltime (in minutes)
- `default_queue` to set the default batch system queue (non-HTCondor only)

Job Router Defaults

HTCondor-CE automatically generates `JOB_ROUTER_DEFAULTS`. Modify existing attributes at your own risk, but feel free to add to it!

```
JOB_ROUTER_DEFAULTS @=jrd
$(JOB_ROUTER_DEFAULTS)
[
  # set the max walltime in minutes
  set_default_maxWallTime = 86400;
  # Route jobs to an HTCondor batch system
  TargetUniverse = 5;
  # Alternatively, route a job to a Slurm batch system:
  # GridResource = "batch slurm"
]
@jrd
```

Job Router Entries

- Use the multiline config syntax
- Each route is enclosed by []
- Each route requires a Name attr
- Since we're using ClassAds, we can use ClassAd functions!
- TARGET ensures that the job attribute is used to match the route

```
JOB_ROUTER_ENTRIES @=jre
[
  Name = "atlas_mcore";
  Requirements = regexp("^usatlas",
TARGET.Owner);
  set_default_xcount = 8;
]
[
  Name = "everything_else";
  set_default_xcount = 1;
]
@jre
```

Job Router Entries

- Use the multiline config syntax
- Each route is enclosed by []
- Each route requires a Name attr
- Since we're using ClassAds, we can use ClassAd functions!
- TARGET ensures that the job attribute is used to match the route

```
JOB_ROUTER_ENTRIES @=jre
[
  Name = "atlas_mcore";
  Requirements = regexp("^usatlas",
TARGET.Owner);
  set_default_xcount = 8;
]
[
  Name = "everything_else";
  set_default_xcount = 1;
]
@jre
```


Job Router Entries

- Use the multiline config syntax
- Each route is enclosed by []
- Each route requires a Name attr
- Since we're using ClassAds, we can use ClassAd functions!
- TARGET ensures that the job attribute is used to match the route

```
JOB_ROUTER_ENTRIES @=jre
[
  Name = "atlas_mcore";
  Requirements = regexp("^usatlas",
TARGET.Owner);
  set_default_xcount = 8;
]
[
  Name = "everything_else";
  set_default_xcount = 1;
]
@jre
```

Job Router Entries

- Use the multiline config syntax
- Each route is enclosed by []
- Each route requires a Name attr
- Since we're using ClassAds, we can use ClassAd functions!
- TARGET ensures that the job attribute is used to match the route

```
JOB_ROUTER_ENTRIES @=jre
[
  Name = "atlas_mcore";
  Requirements = regexp("^usatlas",
TARGET.Owner);
  set_default_xcount = 8;
]
[
  Name = "everything_else";
  set_default_xcount = 1;
]
@jre
```

Job Router Entries

- Use the multiline config syntax
- Each route is enclosed by []
- Each route requires a Name attr
- Since we're using ClassAds, we can use ClassAd functions!
- TARGET ensures that the job attribute is used to match the route

```
JOB_ROUTER_ENTRIES @=jre
[
  Name = "atlas_mcore";
  Requirements = regexp("^usatlas",
TARGET.Owner);
  set_default_xcount = 8;
]
[
  Name = "everything_else";
  set_default_xcount = 1;
]
@jre
```

HTCondor-Specific Route Configuration

A common use of `set_*` for HTCondor batch systems is to set periodic expressions on the routed job

```
JOB_ROUTER_ENTRIES @=jre
[
  name = "Setting periodic statements";
  # Puts the routed job on hold if the job's been idle and has been started at least once
  or if the job has tried to start more than once
  set_Periodic_Hold = ( NumJobStarts >= 1 && JobStatus == 1 ) || NumJobStarts > 1;
  # Remove routed jobs if their walltime is longer than 3 days and 5 minutes
  set_Periodic_Remove = ( RemoteWallClockTime > ( 3*24*60*60 + 5*60 ) );
  # Release routed jobs if the condor_starter couldn't start the executable and
  'VMGAHP_ERR_INTERNAL' is in the HoldReason
  set_Periodic_Release = HoldReasonCode == 6 && regexp("VMGAHP_ERR_INTERNAL", HoldReason);
]
@jre
```

Non-HTCondor-Specific Route Configuration

For batch system directives not covered, there's default_remote_cerequirements:

```
set_default_remote_cerequirements = strcat("Walltime == 3600 && AccountingGroup ==",  
x509UserProxyFirstFQAN, "\\");
```

Results in `$Walltime` and `$AccountingGroup` shell variables that can be used in the relevant `/usr/libexec/condor/glite/bin/*_local_submit_attributes.sh` for your batch system. An example PBS script:

```
#!/bin/bash  
  
echo "#PBS -l walltime=$Walltime"  
echo "#PBS -A $AccountingGroup"
```

Whose output is appended to the job submitted with `qsub`