



# GlideinWMS and IAM

Marco Mambelli, Dennis Box - Fermilab

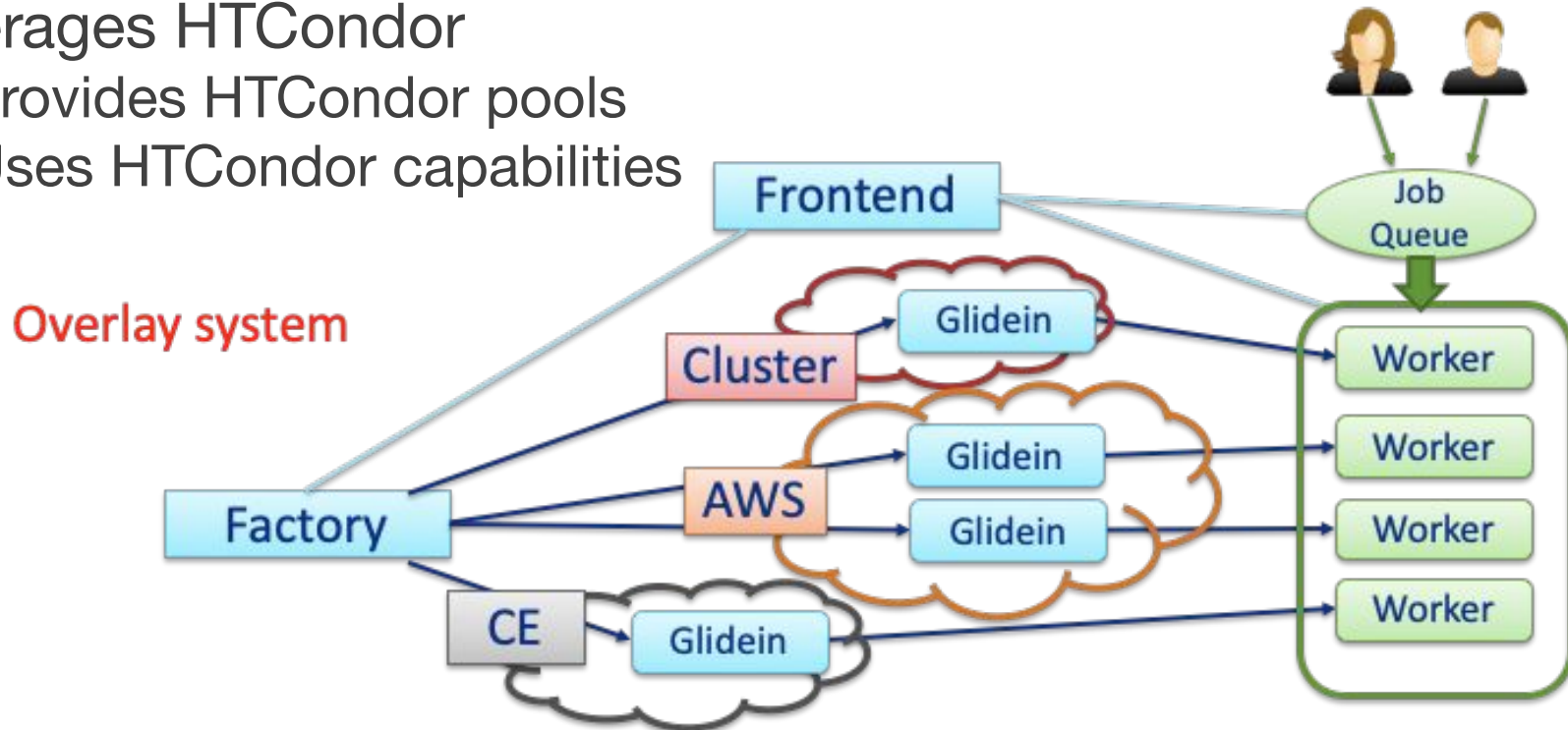
January 28, 2021

IAM Users Workshop

Science and Technology Facilities Council STFC (GB)

# GlideinWMS

- GlideinWMS is a pilot based resource provisioning tool for distributed High Throughput Computing
- Provides reliable and uniform virtual clusters
- Submits Glideins to unreliable heterogeneous resources
- Leverages HTCondor
  - Provides HTCondor pools
  - Uses HTCondor capabilities



# Frontend

- **Manages credentials and delegates them to the Factory**
  - **Managed by VO: stores and owns VO credentials**
  - **Long term credentials, forwarding short term ones**
  - **Manual input, interacts with IAMs**
- Monitors jobs to see how many Glideins are needed
- Compares what entries (sites) are available
- Requests Glideins from the Factory
- Requests Factory to kill Glideins if there are too many
- Pressure-based system
  - Works keeping a certain number of Glideins running or idle at the sites
  - Gradual Glideins requests to avoid spikes and overloads

# Factory

- **Keeps a cache of credentials used or forwarded to Glideins**
- A Glidein Factory knows how to submit to sites
  - Sites are described in a local configuration
  - Only trusted and tested sites are included
- Each site entry in the configuration contains
  - Contact info (hostname, resource type, queue name)
  - Site configuration (startup dir, OS type, ...)
  - VOs authorized/supported
  - Other attributes (Site name, core count, max memory, ...)
  - Glideins can also auto-detect resources
- Configuration can be auto-generated (e.g. from CRIC), admin curated, stored in VCS (e.g. GitHub)
- Condor does the heavy lifting of submissions.

# Glidein: node testing and customization

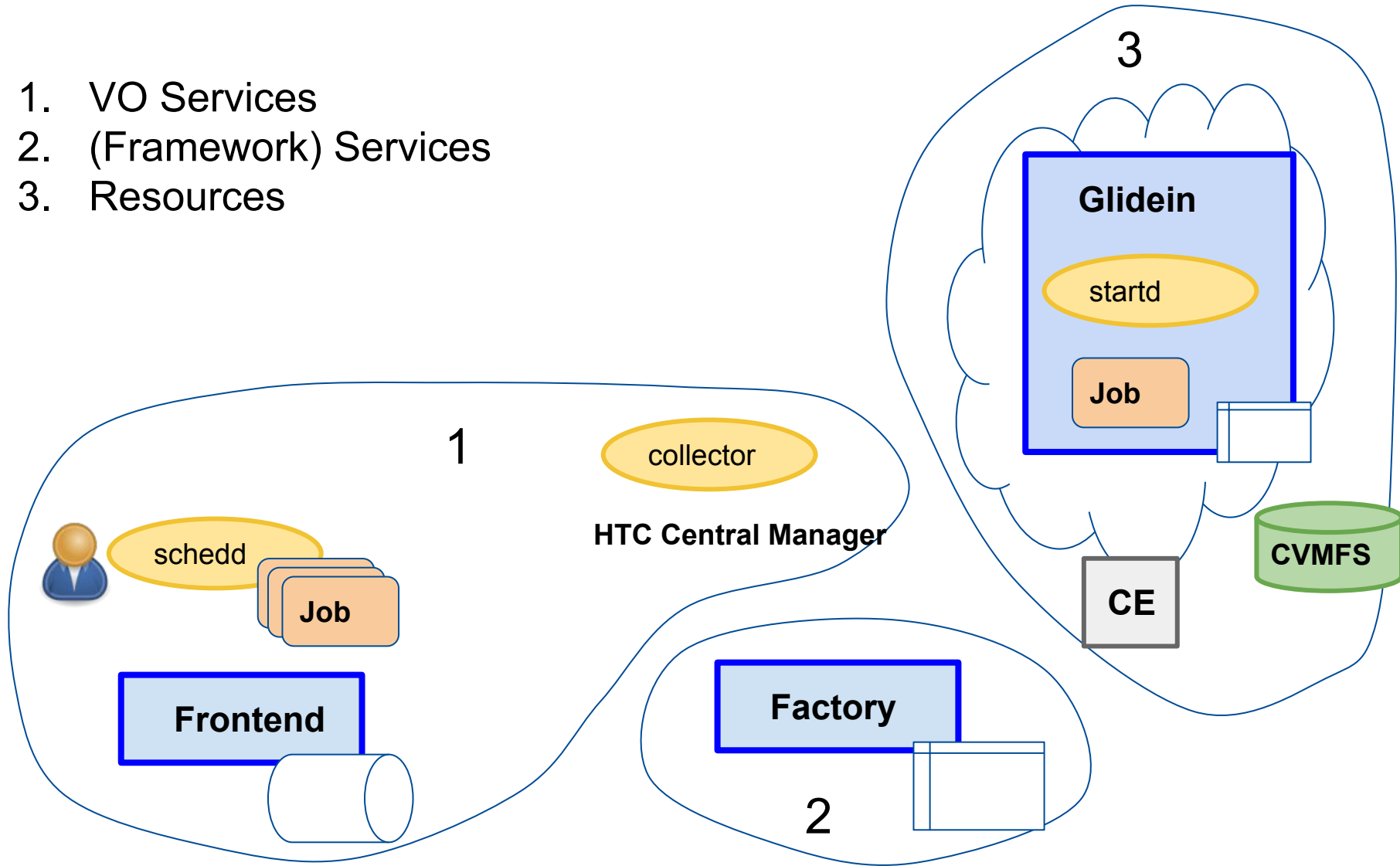
- Scouts for resources and validates the Worker node
  - Cores, memory, disk, GPU, ...
  - OS, software installed
  - CVMFS
  - VO specific tests
- Customizes the Worker node
  - Environment, GPU libraries, ...
  - Starting containers (Singularity, ...)
  - VO specific setup
- Provides a reliable and customized execute node to HTCondor
- Reports back to the Factory
- **Pilot (e.g. VO Group) credentials storage and use**
- **Safely receive and store Job credentials**

# GlideinWMS and IAM

- Carrier of different credentials, both identity or capability based
  - must support different resources
- Agnostic about the type
  - provider and service have to be compatible
- Internally using IDTOKENS (heavily reliant on HTCCondor)

# GlideinWMS system

1. VO Services
2. (Framework) Services
3. Resources

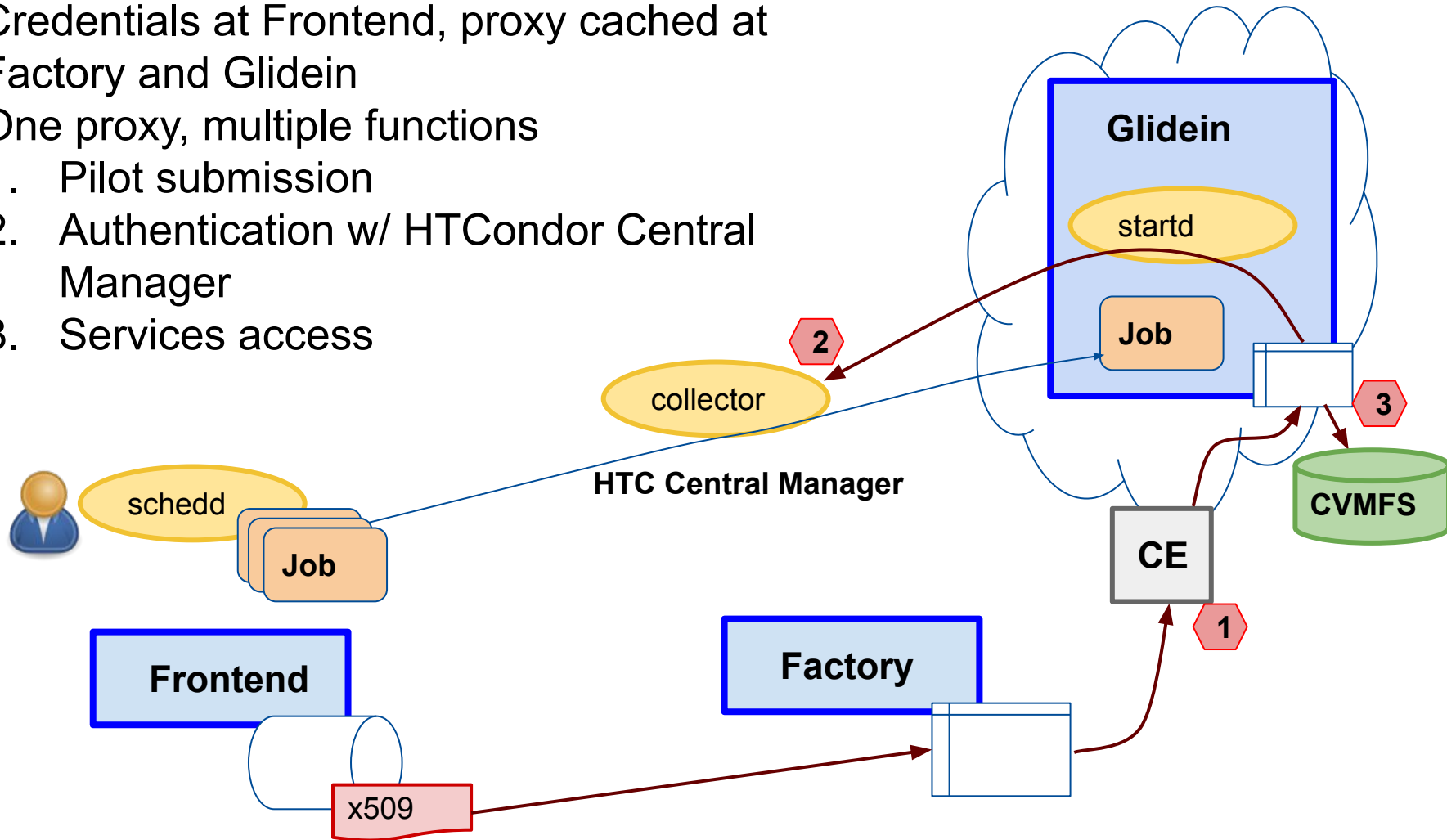


# Traditional x509 authentication - Pilot proxy

Credentials at Frontend, proxy cached at Factory and Glidein

One proxy, multiple functions

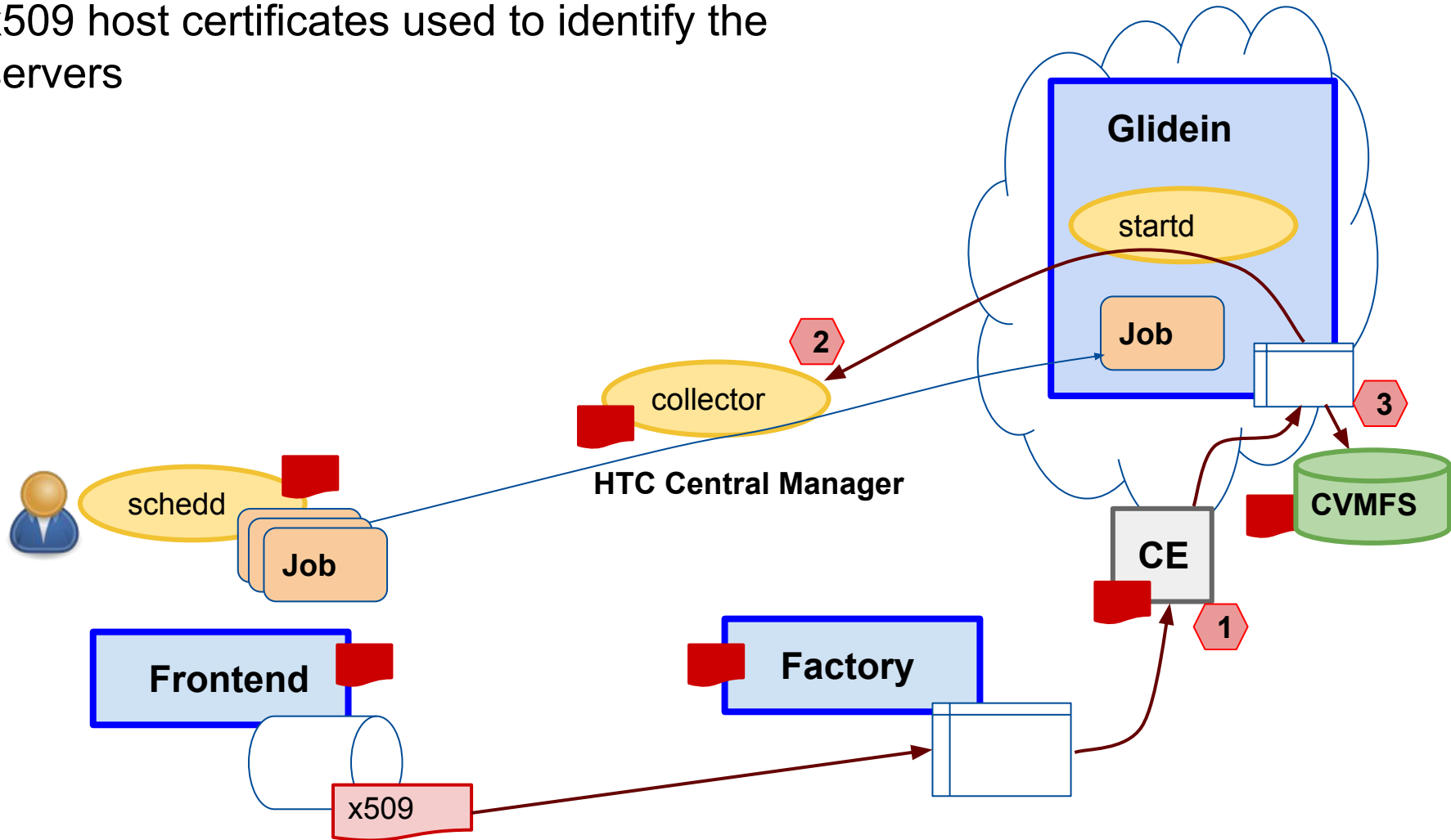
1. Pilot submission
2. Authentication w/ HTCondor Central Manager
3. Services access





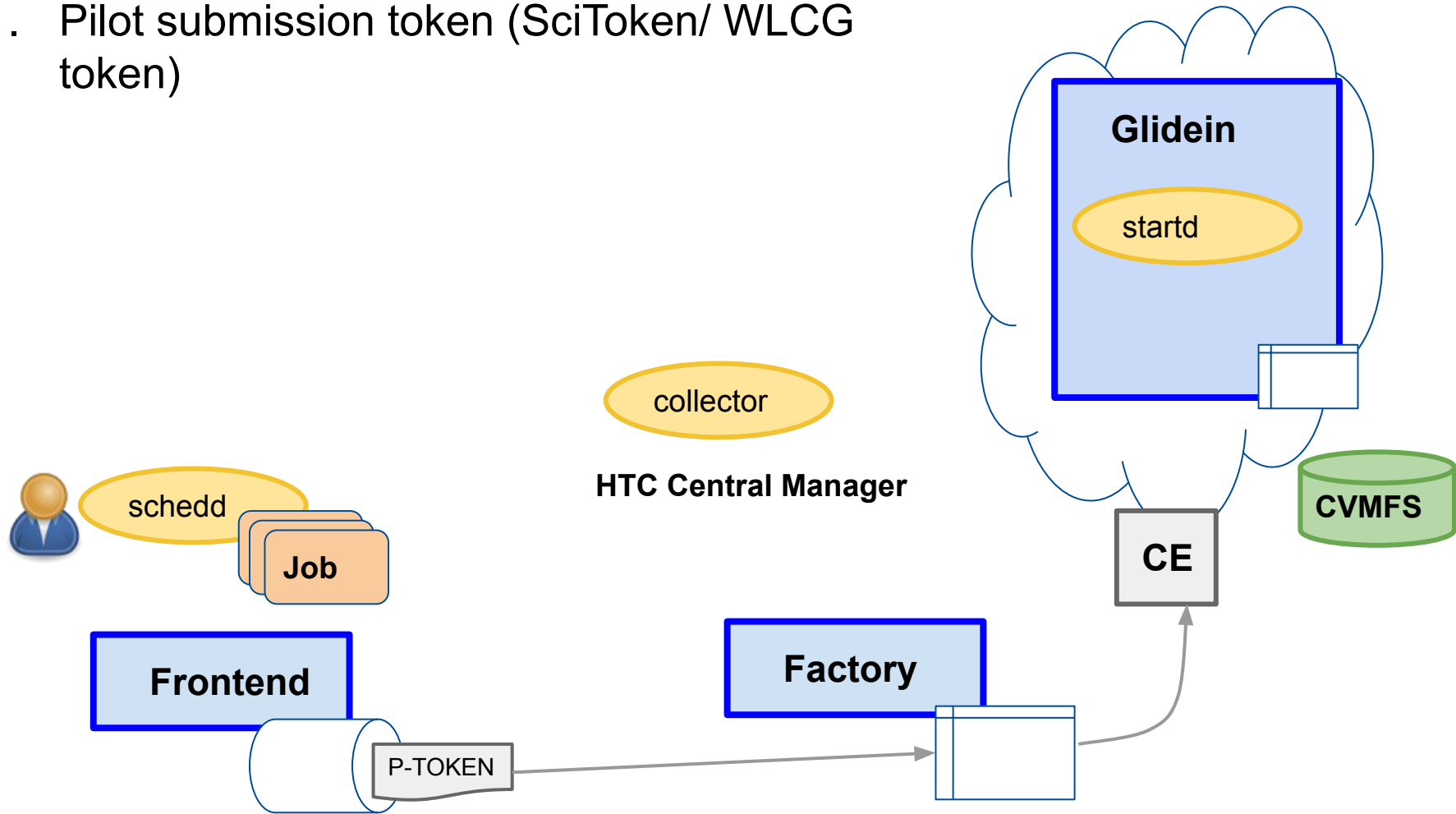
# Traditional x509 authentication - Host certificates

x509 host certificates used to identify the servers



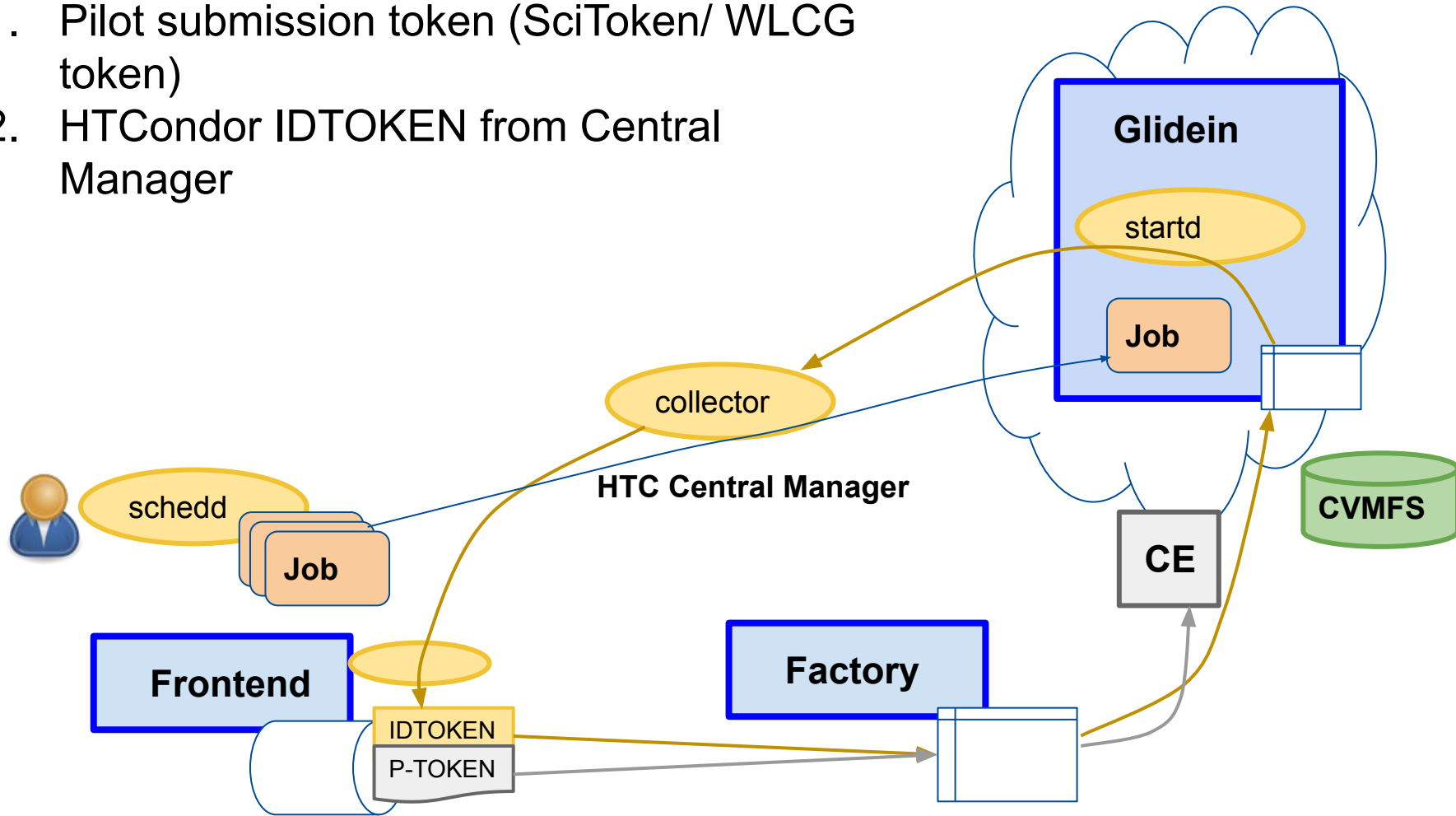
# Token authentications

1. Pilot submission token (SciToken/ WLCG token)



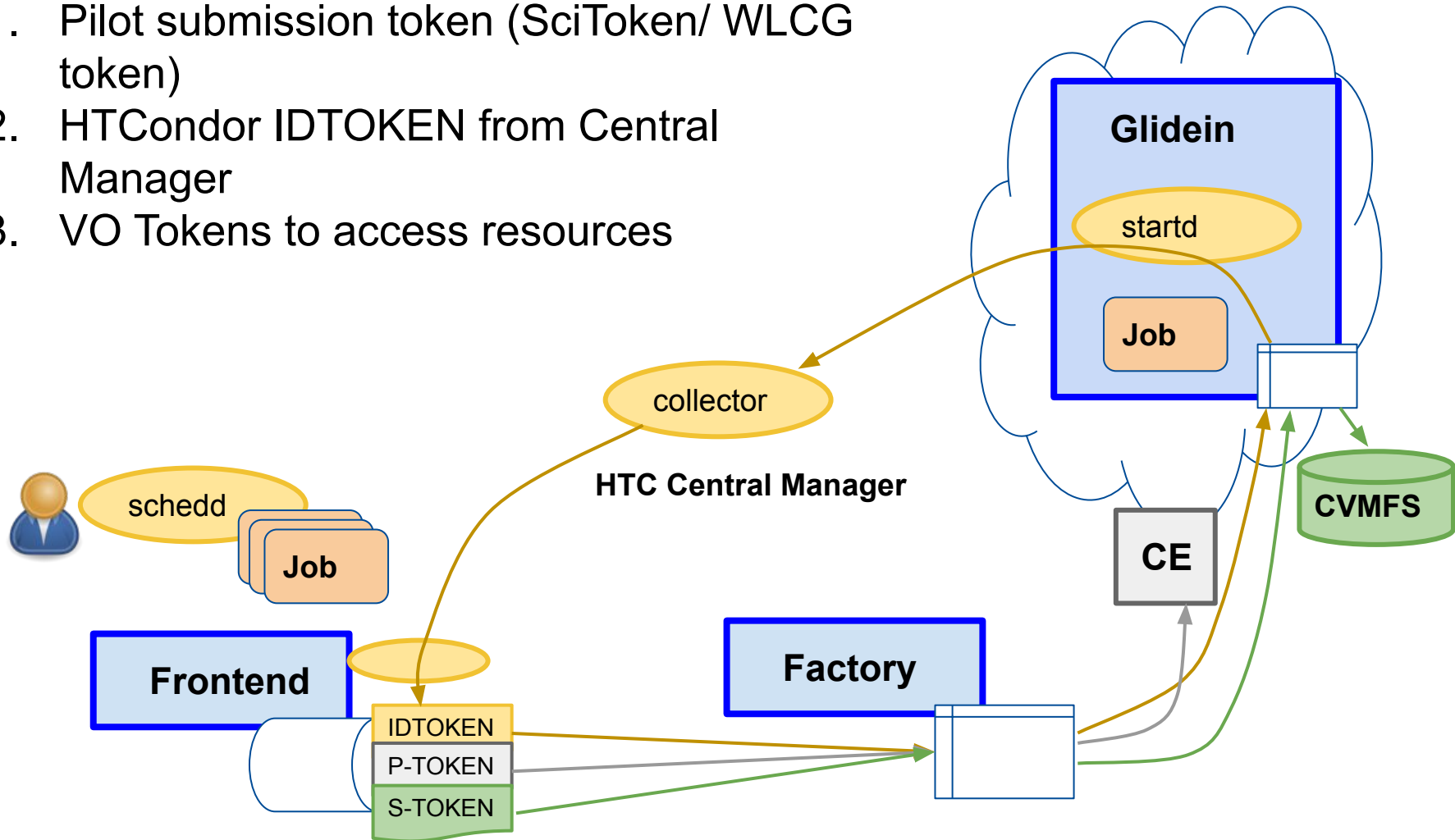
# Token authentications

1. Pilot submission token (SciToken/ WLCG token)
2. HTCondor IDTOKEN from Central Manager



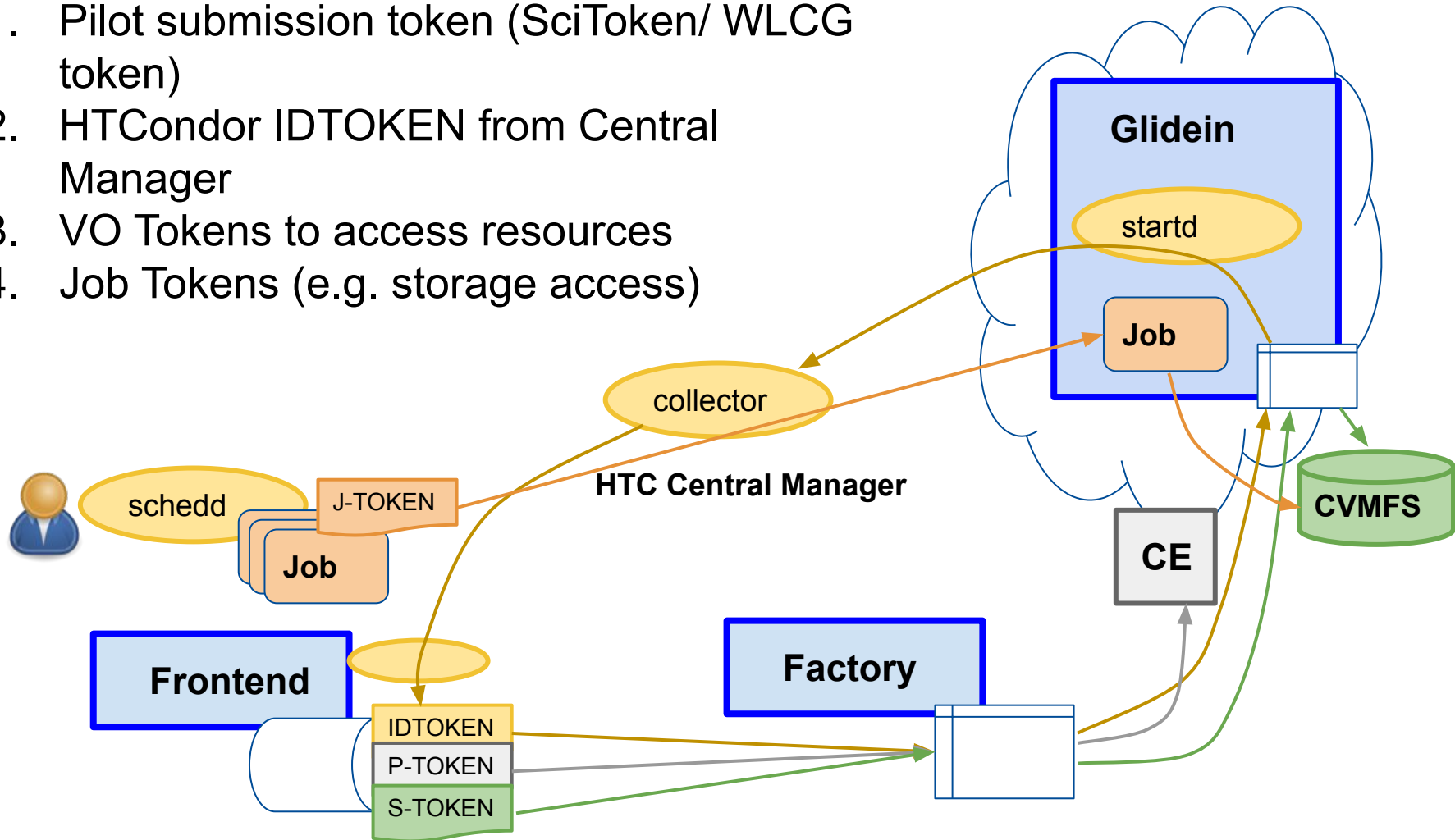
# Token authentications

1. Pilot submission token (SciToken/ WLCG token)
2. HTCondor IDTOKEN from Central Manager
3. VO Tokens to access resources



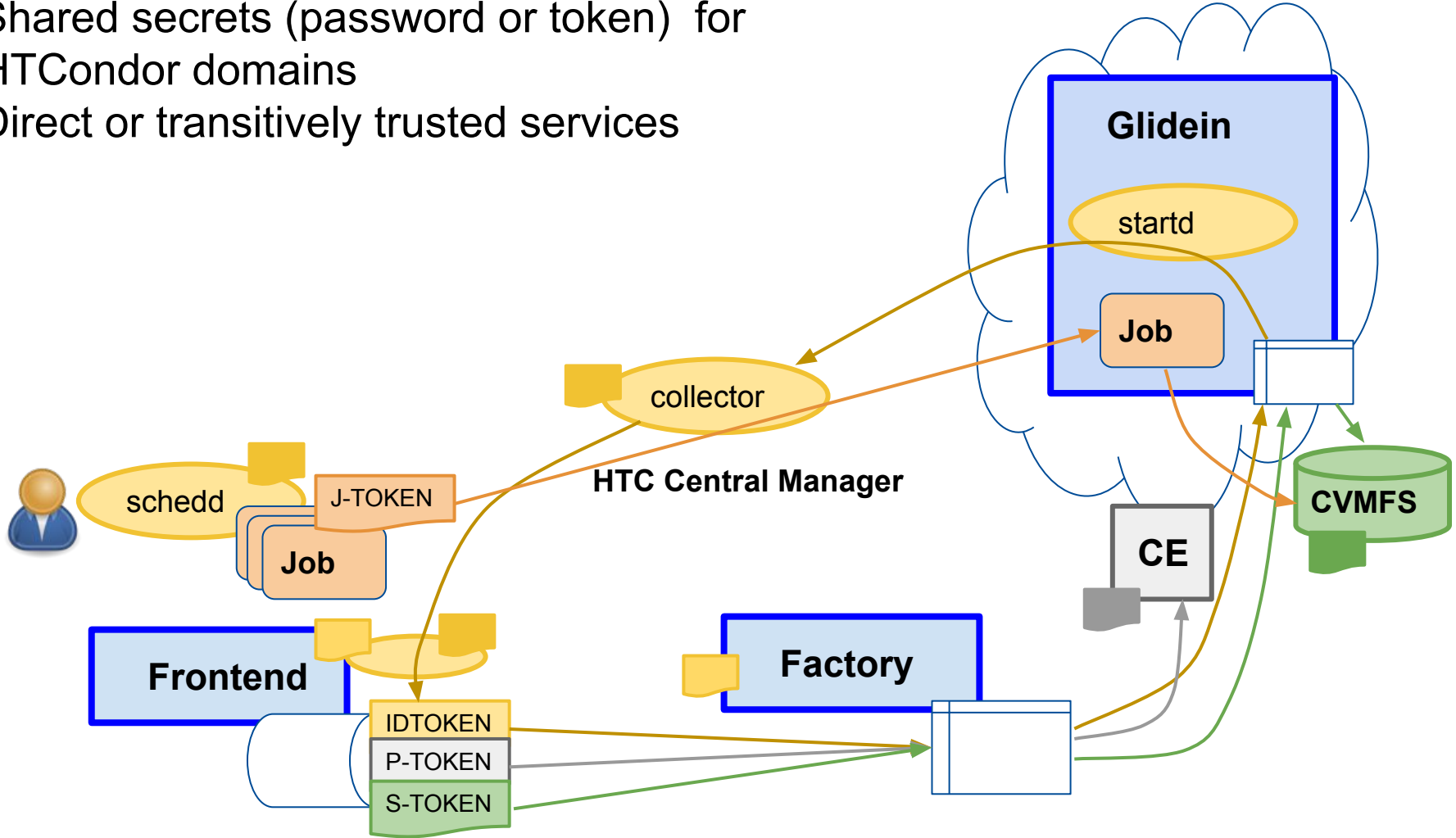
# Token authentications

1. Pilot submission token (SciToken/ WLCG token)
2. HTCondor IDTOKEN from Central Manager
3. VO Tokens to access resources
4. Job Tokens (e.g. storage access)



# Token authentications - services

Shared secrets (password or token) for HTCondor domains  
Direct or transitively trusted services



## Framework credentials use HTCondor (mostly)

- IDTOKEN created on the Factory (condor\_token\_create) copied to the Frontend
- Frontend owns a copy of the VO pool password to create per Site tokens for the Glideins
  - tokens can be invalidated changing the generating password
  - renewal mechanism will allow short lived tokens
- Monitoring servers issue JWT token for the Glidiens

# VO credentials

- Currently treated as files
- Created independently
  - Request from WLCG INDIGO IAM via oidc-agent
  - Create a self-signed SciToken via scitoken-admins-... commands
- Renewal works between the Frontend and the Factory
- Will add some tool to ease creation, handling and renewal



# Acknowledgements

This work was done under the GlideinWMS project

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

## References

<https://github.com/glideinWMS/glideinwms>