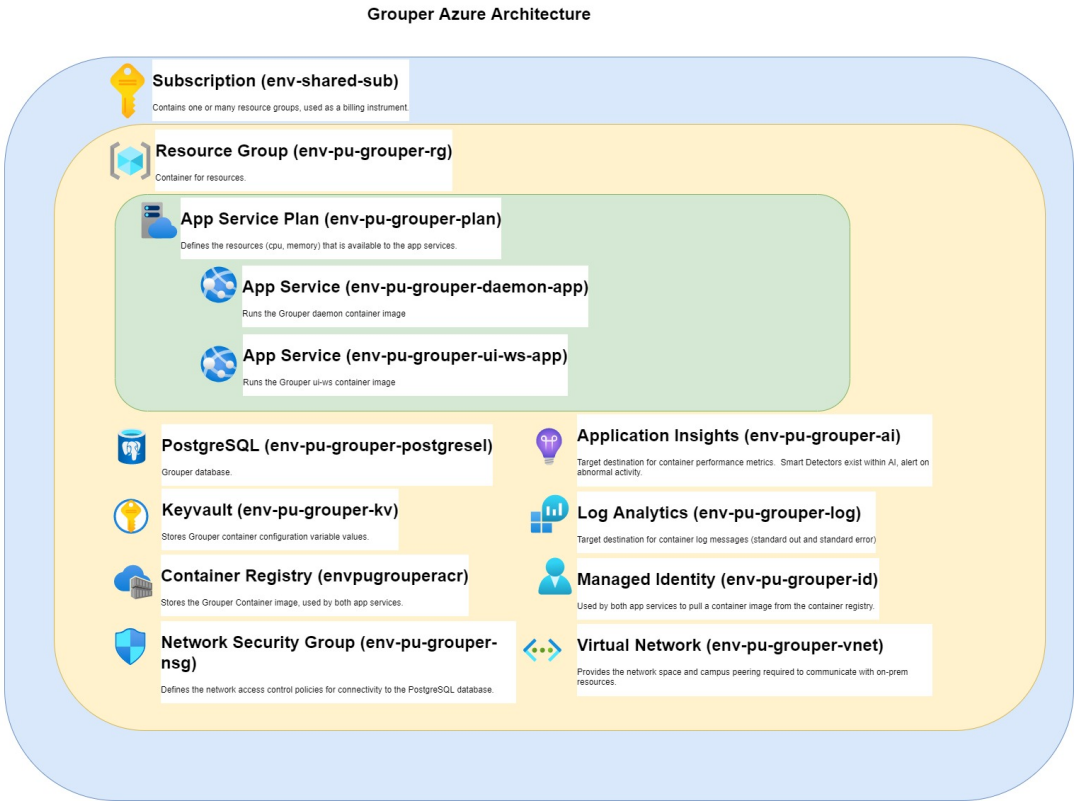
Princeton University Grouper Page

Wiki Home	Grouper Release Announcements	Grouper Guides	Grouper Deployment Guide	Community Contributions	Internal Developer Resources

The purpose of this contribution page will be to share our approach to deploy, update container images, and monitor Grouper. We will also share our architecture diagrams and a nifty solution to create a New Employees Grouper group.

Grouper Architecture

We opted to deploy Grouper into Azure. We have an Azure ExpressRoute back to our Source of Record (Microsoft Identity Manager) and our primary provisioning target (Active Directory).



The Grouper containers run inside an Azure App Service Plan, which basically defines the resources (CPU/ memory) that are available to the App Services (which run the actual containers). Both containers send log messages to our Log Analytics workspace and performance metrics are sent to Application Insights. We have enabled Smart Detectors within Application Insights to alert us of abnormal activity via an action group; we have received a few alerts and have adjusted settings within Grouper accordingly.

Notes regarding our Azure Resources:

- Container registry is where our container images are stored. We had an issue, 30 days post initial deployment, where the App Services stopped running and could not connect to the ACR. To resolve this, we created a Managed Identity that has permission to connect to the ACR and pull container images.
- Grouper is a noisy application from a logs perspective, and we are seeing lots of log messages to sent to our log analytics workspace. We are considering implement data caps to reduce the cost associated with the log storage.
- KeyVault is used to store sensitive information (DB password) and configuration variable values that are passed to the container during startup

See also:

- Application performance monitoring

- Azure Release Pipeline
- Container update process
- New employees group