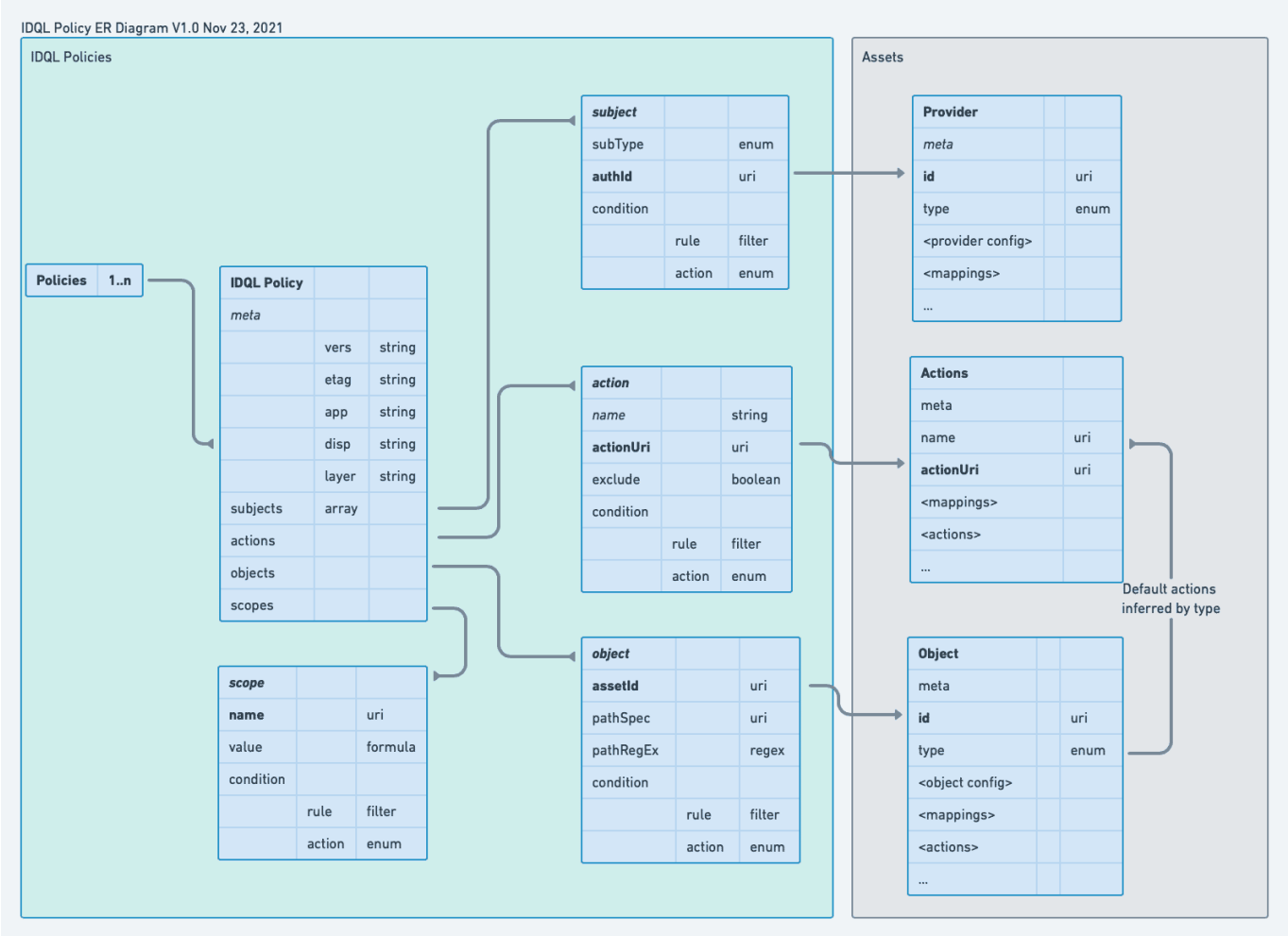
IDQL : le langage qui se voulait standard de l'IAM multicloud

Comment exploiter mon *bucket* S3 depuis mon cluster Azure ? Derrière cette question, un défi : la gestion des identités et des accès en multicloud. Pour le résoudre, divers leviers. En particulier, des mécanismes de fédération… plus ou moins aboutis cependant, et c'est compter la variété des nomenclatures entre plates-formes.

Face à cette diversité, un projet s'est constitué l'an dernier. Sa promesse : pouvoir configurer des politiques de contrôle d'accès « transversales », exprimées en YAML ou en JSON. Le socle : un format unique, nommé IDQL (IDentity Query Language).

En guise d'implémentation de référence, les porteurs du projet – en tête desquels l'entreprise américaine Strata Security – ont associé à IDQL un orchestrateur, sous la forme d'une *web app*. Son nom : Hexa. Son rôle principal : distribuer des règles IDQL vers les différents environnements, avec un *mapping* ou une conversion automatique au format natif.



L'idée est à la fois d'unifier la gestion des accès entre les couches des applications et entre les plates-formes d'infrastructure. Chez ces dernières, AWS, Azure et GCP sont les premières cibles. Snowflake, Versa SASE et F5 NGINX figurent aussi sur la feuille de route publique.

Le duo IDQL/Hexa est candidat à la CNCF (Cloud Native Computing Foundation), au premier niveau d'incubation (*sandbox*). Il joue la complémentarité avec un autre projet passé sous l'aile de cette fondation, et qui a quant à lui atteint le plus haut seuil de maturité (*graduated*) : Open Policy Agent. Ce dernier, axé sur Kubernetes, a aussi son langage pour la définition de règles : Rego. Il constituerait le point de connexion avec IDQL.

Illustration principale © everything possible – Shutterstock