CloudAuthZ
• Removes username and password databases
• Two-Factor Authentication
• No single point of failure
• Simplified and improved UX
• Pricing that scales for the cloud
What is CloudAuthZ?

Ontology / Syntax for describing:

• Distributed User Authorization
• Context Driven Entitlements
• Attribute / Provider Reliability Indexes
• Entitlements Catalog
Why is CloudAuthZ needed?

Authentication is the EASY PART!

How do you federate?

• Distributed User Authorization
• Context Driven Entitlements
• Attribute / Provider Reliability Indexes
• Entitlements Catalog
Authorization needs Authentication
Authentication needs Identity

So, taking a page out of the identity playbook........
The good thing about reinventing the wheel is that you can get a round one.

- Douglas Crockford
Identity
Distributed, Federated, or Local?

• Distributed
  – Don’t know ahead of time who all the players are
  – Internet collaboration

• Federated
  – Know who the players are, but not what exactly they’ll be doing
  – Cross-organizational collaboration

• Local
  – Control everything within a single stack
  – Within-app (username/pass)
  – Within-enterprise (LDAP)
Federated Identity Protocols Overview
OpenID 2.0

• Fully distributed, scales to internet size
  – Can work without prearranged IdP/RP relationship
• Designed around per-user trust decisions
  – Can use whitelists/blacklists for SSO
• Provides authentication
  – Extensible to include profiles and claims
  – Uses front channel for passing data
• Provides base-level assurance
  – RP has to really trust the IdP
  – Roughly equivalent to email-backed username/password pairs
SAML (Web SSO)

• Requires federation of central authorities
  – Preregistration of trust, pairwise
  – Can scale to small numbers of organizations

• Higher levels of assurance
  – Configured certificates for signing

• Can handle authentication, authorization, claims, and some profile
  – All passed as signed claims
  – Formats must be agreed upon between parties

• Wide deployment in enterprise environments
OAuth-style Directed ID

- Strong tie to the IdP
  - Preregistration
  - Implicit full trust of IdP
  - Profile bits packed in for free
  - Uses back channel for passing data
- Already used on the web
  - Twitter sign in
  - Facebook Connect
- Dumbfoundingly huge adoption
  - Millions of client sites on the public web
OpenID Connect

• Identity protocol built on OAuth 2.0
• Uses back-channel binding for data flow
• Can be higher-assurance
  – Preregistration, strong binding
  – Certificate-level verification between servers
• Can be ad-hoc
  – Dynamic registration
  – Arbitrary IdP for users is possible
The Venn of Federated Identity

From http://blogs.forrester.com/eve_maler/11-02-03-openid_successful_failures_and_new_federated_identity_options
The Identity Singularity*

• The Venn bubbles are collapsing
  – Open, distributed technology is moving toward supporting higher assurance
  – Enterprise technology is moving toward supporting more dynamic use

• Working systems are being defined now
  – It’s a real need across industries

*Term borrowed from Eve Maler http://www.xmlgrrl.com/
Distributed Authentication requires Distributed Authorization
Distributed Authorization

For authorization decisions that depend on:
• information belonging to other domains
• which cannot be directly accessed due to privacy issues
• authorization decision is delegated to the areas which could handle it
• results of such delegated decisions are combined to form an appropriate decision
CloudAuthZ
Policy Enforcement Points
The Future

- CloudAuthZ and Policy Enforcement Points
- A *system entity* that requests and subsequently enforces *authorization decisions*. 
The Result

• Just as authentication / identity verification can be offloaded to IdP, Context Driven Entitlements can be offloaded to PEP

End Result:

• Distributed systems that are resource efficient
• Entitlements don’t remain in silos
Thank You

Brian@certivox.com