Aggregated Service Discovery

Problem Statement

Andrew Biggs
adb@cisco.com
IETF86, March 2013
Observations

• Providers offer a growing number of services.
  Email, directory, calendaring, IM/P, voice, streaming, collaboration, file sharing.

• Users use a variety of clients and devices.
  Smartphones, tablets, desktop apps, web browsers, phones, video endpoints.

• Users are mobile and access services from varied locations.
  Inside the firewall, outside the firewall, main campus, branch office.

• How a client connects can be a function of user identity.
  Organizationally distributed services, subscription services.

• Client configuration has become more complex.
  Growing need to leverage automated service discovery.
Service Discovery Options

• Service discovery is not a new problem...

• DHCP
  Good for discovering local network service instances, DNS, NTP, NIS, etc.  
  Limited to local network, app-level DHCP APIs not pervasive.

• DNS SRV
  Good for discovering multiple instances of services, XMPP, CalDAV, etc.  
  Limited to hostname/port, security issues, not for browser-based apps.

• DNS SD
  Great for advanced and custom configs, leverages existing DNS record types.  
  Not for user-specific or sensitive service access information.

• WebFinger
  Extensible, secure, user-oriented, easily deployed and accessed. 
  Public directory geared toward linking together a user’s resources.
Service Discovery Requirements

• Keep it simple for end users.
  Minimize what users must provide in order to bootstrap discovery.

• Enable adoption by client developers.
  Consider constrained clients, lower the barriers to adoption where possible.

• Make basic deployment simple, advanced deployment possible.
  Support multiple service instances, prioritization, user/location/device-specific configs.

• Consider user experience and mobile devices.
  Optimize for low latency, with minimal round-trips and aggregation.

• It should be extensible.
  Support registered schemas for standard protocol configs as well as custom extensions.

• It must be secure.
  Require secure channel, support authenticated access.