Web Service Discovery

Phillip Hallam-Baker
Comodo Security Solutions
Requirements

• Discovery of Web Service endpoints for FOO service by
  • DNS Name “example.com”
  • Account identifier alice@example.com

• Convert to Web Service Endpoint
  • http://example.com/SOMETHING-UNIQUE-TO-FOO
  • https://example.com/SOMETHING-UNIQUE-TO-FOO

• If a service is implemented by more than one host
  • Go to the host that provides the service flavor we need.
Constraints

• Work within existing DNS infrastructure (only use widely supported records)

• Allow for (limited) service description
  • Information a service might like to know before it starts transport security
RFC 6763 Service discovery

• Constrains design space (great) does not specify single approach
  • Does not fully describe Web service interaction.

• Prefixed records to specify service and description:
  • SRV – specifies the service with support for fallover
  • TXT – allows for service description

• Does not:
  • Fallback strategy for when client cannot obtain full DNS access
  • Differentiate host vs service parameters (but implementations can)
Proposed approach

• Use SRV/TXT records as preferred mechanism
  • SRV records define the set of hosts providing service
  • TXT records prefixing service address describe service
  • TXT records prefixing host address describe specific host

• Fallback to <service>.<domain>
  • Use .well-known/wks/<service> to complete endpoint.

• Last chance <domain>
Example: alice@example.com

_mmm._tcp.example.com SRV host1.example.com 0 10 80 host1.example.com
_mmm._tcp.example.com SRV host2.example.com 0 40 80 host2.example.com
_mmm._tcp.example.com TXT "version=1.0-2.0"
mmm.example.com CNAME host3.example.com
host1.example.com A 10.0.1.1
host2.example.com A 10.0.1.2
_mmm._tcp.host2.example.com TXT "path=/service"
host3.example.com A 10.0.1.1
host3.example.com A 10.0.1.2
Ways forward

• Depends on extent of service description standardization

• If we are standardizing only PATH/VERSION/Encoding tags
  • Approach is 90% constrained by RFC 6763
  • Just make draft consistent & sensible

• If we want to standardize anything else... it gets complicated