Finding Services on IP Networks with SLP

James Kempf
Senior Staff Engineer
Sun Microsystems Laboratories
What is SLP?

• **Service Location Protocol** is a new IETF standards track protocol for discovering services on IP intranets.

• A *service* is any network accessible hardware device or software server.
  
  • Examples: printers, file servers, video cameras, HTTP servers, etc.

• **Discovery** is the process by which potential clients of the service obtain enough information to access the service.

• Replaces or supplements proprietary service discovery protocols such as:
  
  • Appletalk NDS
  • NetWare SAP
  • Microsoft CIFS Browsing Protocol.
**Characteristics of SLP**

- Clients find services by type and desired attributes rather than by name.

- Services can be organized into administrative, logical, or physical groupings called scopes.
  - Scopes provide upward scalability.

- No preconfiguration needed for bootstrapping.
  - SLP has been designated a bootstrapping protocol by the IETF.

- Operates in the absence of a directory server.

- Object-based security using public key cryptography assures that discovered information came from a secure server.
SLP Entities

• *Service Agents* (SAs) advertise services on behalf of a client offering them.

• *User Agents* (UAs) look for services on behalf of a client requiring them.

• *Directory Agents* (DAs) act as a cache of service advertisements, reducing multicast network traffic in large installations.

• UAs communicate requests to SAs via IP multicast.

• UAs communicate requests to DAs via TCP or UDP.

• SAs communicate requests to DAs via TCP or UDP.
SLP Architecture
**Service Advertisements**

- SAs register service advertisements with DAs using SrvReg or manage the advertisements themselves if no DA.

- A service advertisement in SLP consists of the following components:
  - A service URL, including the service type, and an advertisement lifetime.
  - A collection of attributes.

- The advertisement is characterized by *service type*.

- Service type template documents define service types.
  - Templates include definitions of attributes and the service URL syntax.

- Templates can be registered with IANA for maximum interoperability.
Service Types

• Three kinds of service types:
  • Protocol types - name corresponds to networking protocol.
    • Example: service:ldap
  • Abstract types - name corresponds to a common function shared by several protocols (SLPv2 only).
    • Examples: service:printing:lpr, service:printing:ipp
  • URL scheme types - name corresponds to a standard URL scheme (SLPv2 only).
    • Example: http

• If the service type is not registered with IANA, an optional naming authority can be included in the type name (protocol and abstract types only).
  • Examples: service:video.sun:mpeg4, service:file-printer.demo
Service URL

• Service URLs bundle the service type with the service access point and other information needed to access the service.

• Examples:
  
nfs://slag.eng.sun.com/src/slp
  service:ldap://www.research.sun.com
  service:sap.novell/ipx/0ffab724:badbaddade44:4242

• A SrvRqst from a UA is answered with a SrvRply containing the service URLs of services whose attributes match the client query.

• Service URLs are registered with a 16 bit positive integer lifetime, which indicates how long the information is valid.
Attributes

• Attributes consist of an LDAPv3 compatible tag and zero or more values.

• Attributes having no values are called keyword attributes.

• Attribute values are transmitted as UTF8 strings but types are determined when the strings are decoded.

• Four attribute types:
  • Integer - ranges over signed 32 bit integers. Any number outside that range is a string.
  • Boolean - value is "true" or "false" (case insensitive). Single valued only.
  • String - all valid UTF8 characters. Some characters important to protocol must be escaped.
  • Opaques - arbitrary sequence of bytes. Transmitted as UTF8 strings in escaped format.
Service Requests

• Service requests consist of:
  • A service type
  • The scopes in which the service should appear.
  • A query consisting of boolean expressions comparing attributes and values.

• Query syntax is base LDAPv3
  • Extensible queries are not supported.

• Example:
  Service Type - service:printing:lpr
  Scopes: default, td, mpk15
  Query: (&(location-description=TD Fax/Printer Room)(duplex-mode=duplex))
Optional Protocol Messages

- **SrvReg/fresh** - updates an advertisement with new attribute information or a new service URL lifetime.

- **SrvDereg** - delete an advertisement.

- **SrvDereg/attr** - delete some attributes from an advertisement.

- **AttrRqst/URL** - return attribute values for a particular registered URL matching a set of tags.

- **AttrRqst/type** - return all attribute values matching a set of tags for a particular service type.

- **SrvTypeRqst** - return all service types in a particular naming authority or all without regard to naming authority.
How Agents Discover a DA

- Static configuration (discouraged).
- DHCP options 78 and 79.
- Active discovery.
- Passive discovery.

If no DAs are discovered, then UAs and SAs use multicast to communicate.
Active DA Discovery

- When a UA or SA comes up, it multicasts an active DA discovery message.

- An active discovery message is a SrvRqst with service type "directory-agent" and an optional list of scopes required by the client agent.

- Reply from DA is a DAAdvert message with URL for directory agent.
  - Example: service:directory-agent://142.142.42.42

- If no scopes are included in the request, all DAs reply.

- If scopes are included in the request, only DAs supporting those scopes reply.
Passive DA Discovery

• When a DA comes up and periodically after, the DA multicasts a DAAdvert message.
  • Recommended multicast period is 3 hours to avoid excessive network traffic.

• SAs must and UAs should listen for unsolicited DAAdvert messages on the SLP port (427) and multicast address.

• When an SA discovers a new DA, it must register all its service advertisements with the DA if the DA supports the same scopes as the SA.

• UAs may use the new DA if they require services from the scopes that the new DA supports.
Scope Configuration

• Initial configuration of scopes requires human intervention.

• Two configurations:
  • User selectable scopes
  • Administrative scopes

• User selectable scopes
  • DAs and also possibly SAs get their scopes from DHCP or static configuration file.
  • UAs and also possibly SAs get their scopes from passive or active DA discovery.
  • Users select scopes in UAs when they want to find something.

• Administrative scopes
  • DAs, UAs, and SAs get their scopes from DHCP or static configuration file.
  • Scope of UA requests is predetermined by administrative fiat.
Security

- SLP *Security Parameters Index* (SPI) is a string transmitted with registrations indicating keying material and cryptoalgorithm parameters.

- SAs sign registrations using private key.

- DAs check signature using public key to ensure authenticity.

- DAs pass signature along to UAs when answering queries, and UAs check signature to be sure DAs haven’t tampered with it.

- To sign DAAdverts, DAs manage their own private keys, UAs and SAs have DA public key.
**SLP and LDAP**

- Discovery of LDAP server not possible within LDAP protocol.
  - DHCP and DNS naming convention alternatives are not dynamic.

- Every system vendor is implementing a different way for clients to find their LDAP server.
  - Microsoft uses DNS (ldap.microsoft.com) or DHCP, Novell uses SLP or DHCP, Linux is considering using SLP.

- Building LDAP into devices problematic because:
  - Server discovery will require a separate, system dependent protocol (DHCP, naming convention, SLP) hindering autoconfiguration.
  - DHCP and naming convention solutions don’t allow autorollover if the primary server goes down.

- Solution: use SLP for service discovery and insert SLP service advertisements into the LDAP directory server.
Integrating SLP and LDAP

SLP
James Kempf
Progress in LDAP and SLP Integration

• Prototype of SLP DA with LDAP backend developed as proof of concept.

• SrvLoc Working Group draft on translating SLP service templates into LDAP schema.
  • http://www.ietf.org/internet-drafts/draft-ietf-svrloc-template-conversion-03.txt

• Contact with Common Information Model (CIM) about integrating SLP template definitions into LDAP schema via CIM.

• Translation draft needs work and involvement of LDAP schema standardization groups.
SLP and JINI

• Service discovery is a small part of what JINI provides.

• Service discovery in JINI returns a service object.
  • Service object interface available for object-oriented client access.

• JINI’s object query semantics, transactions, and event notification are superior to LDAP or SLP for describing complex distributed object operations.

• JINI uses RMI, restricting it to JVM-based clients only.

• For small embedded systems and enterprise C clients, a JVM may be unavailable.

• An SLP/JINI bridge allows JINI networks to access industry standard SLP-enabled devices.
Integrating SLP and JINI

Note: Unicast UA/SA contact also possible
**SLP and JNDI**

- Java Naming and Directory Interface (JNDI) is a standard Java extension (javax package):
  - For obtaining objects from directory services.
  - JNDI API is independent of directory service provider.
  - Naming is dependent on the directory service provider.

- An SLP service provider for JNDI uses SLP to deliver service objects for network services to clients:
  - SLPv1 service provider is available at:
  - Service object delivered by JNDI can be the same as for JINI.

- Widespread deployment of JNDI:
  - Seven service providers available from website, others as commercial products.
  - Many enterprise client products under development use JNDI, several already delivered.
**SLP/JNDI Architecture**

Service Client

JNDI

SLP Service Provider

SLP API

SLP

HTTP

LPR

Text File

Service Object Jar File

Service Object

service URL

JNDI driver URL

SrvRqst (multicast or to DA)

SrvRply (to multicast)

SrvReg (to DA)
Industry Momentum for SLP

• Axis has a network printing product with SLP built in.

• HP’s latest network enabled printer line and WebJet administration product have SLP built in.

• Novell Netware 5.0 has integrated LDAP directory services with SLP as a replacement for their proprietary SAP protocol.

• Salutation Consortium, MNCRS, and the Intel Wired for Management initiative have adopted SLP.

• Apple is implementing SLP as a replacement for Appletalk NBP.

• IBM has a load sharing terminal client and server using SLP.
Summary

• SLP provides a way for network services to autoconfigure without much human intervention.

• SLP is attractive to device manufacturers because it is a bootstrapping protocol.

• SLP complements LDAP directory services.

• SLP provides a window into the world of standard, IETF protocols and nonJava clients for JINI.

• SLP with JNDI provides service objects for enterprise clients.

• Please talk to your customers and system vendors about SLP!

• For more information: http://www.srvloc.org.