

## TN3270E Service Location and Session Balancing

### Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

### Copyright Notice

Copyright (C) The Internet Society (2001). All Rights Reserved.

### Abstract

This document discusses the implementation of Service Location Protocol (SLP) and session balancing with a TN3270E emulator in a client server implementation with a TN3270E server.

Application program developer's can locate TN3270E services and load balance among those services (3270 host sessions), by using this SLP support.

### Table of Contents

1. Introduction and Terminology .....	2
1.1 Terminology .....	2
2. An Overview of <a href="#">RFC 2165</a> .....	3
2.1 SLP Agents .....	3
2.2 Service Agents .....	3
2.3 User Agents .....	4
3. TN3270E Server Environment and Load .....	4
3.1 TnN3270E Server Load .....	4
4. TN3270E Client Configuration .....	6
4.1 SLP Scope .....	6
4.2 DA-Discovery Time-Out .....	6
4.3 SA-Discovery Time-Out .....	7
5. TN3270E Client Implementation Information .....	7
5.1 Overview .....	7

5.2	How to Obtain List of TN3270E Servers Supporting SLP .....	8
5.3	TN3270E Sample Client Flow .....	9
5.3.1	Open the SLP connection .....	9
5.3.2	Query the list of TN3270E servers .....	9
5.3.3	Forward Looking Example using SLPv2 .....	10
5.3.4	Determine loading of each TN3270E server .....	10
5.4	Recommendations .....	11
6.	Sample Trace Flow of SLP and Session Balancing .....	11
7.	Service Templates and Service Registration .....	12
7.1	The TN3270E Service Type Template .....	12
7.2	The Server Service Template .....	16
7.3	Template Contact Information .....	17
7.4	Security Considerations .....	17
7.5	Sample TN3270 Service Registration Message .....	18
7.6	Sample Server Service Registration Message .....	19
8.	References .....	19
9.	Authors' Addresses .....	20
10.	Full Copyright Statement .....	21

## 1. Introduction and Terminology

This document will provide information on Service Location Protocol implementation to discover TN3270E servers in a network and session balance among those servers. This implementation follows the standards track [RFC 2165](#), Service Location Protocol [1] but also provides some examples when using Service Location Protocol version 2 to be forward looking. Service Location Protocol version 2 is documented in [RFC 2608](#) [4] and [RFC 2609](#) [2].

### 1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [1].

Session Balance - This refers to the ability of TN3270E client to use server load information to establish a TN3270E connection to the TN3270E server with the least load at that time. The purpose is to distribute the connection of TN3270E sessions among more than one TN3270E server, and one server will not be excessively loaded. The term "load balance" is a more general term, with respect to server load, and in this document we are focusing on the TN3270E session connections to least loaded servers.

SNA Gateway - A Systems Network Architecture (SNA) gateway allows multiple LAN-attached workstations to access SNA hosts through one or more physical connections to one or more hosts. A SNA gateway acts as a protocol converter between workstations attached to a LAN and a

WAN host line. It typically would support the SNA protocols LU 0, 1, 2, 3, and dependent LU 6.2 (APPC). SNA gateways typically include a TN3270E server capability.

LU Pool - The Logical Units (LUs) defined in the gateway can be dedicated to a particular workstation or pooled among multiple workstations. Pooling allows workstations to share common Logical Units (LUs), which increases the efficiency of the LUs and reduces the configuration and startup requirements at the host. When a client connects to the gateway, the gateway retrieves an LU from the pool to establish a session. The LU is returned to the pool for access by other workstations when the session is ended.

Commserver Service Type Template - Commserver service type is defined as an SNA Gateway server as previously defined above in this terminology section. A template describing the attributes for this service type is in [section 7.2](#).

## 2. An overview of [RFC 2165](#)

[RFC 2165](#), Service Location Protocol (SLP) [1], provides an automatic way for clients to discover services within an administrative domain.

These services have various attributes associated with them from which a client can base a service selection. The basic design involves the use of three agent types. These are: User Agents (UA's), Service Agents (SA's) and Directory Agents (DA's).

### 2.1 SLP Agents

User Agents are used to query Service Agents or Directory Agents. They acquire/request service information based upon the desired attributes and service needed for the user application.

Service Agents represent a specific service and advertise service information.

Directory Agents act as a central collection point for service registration information by Service Agents which is later requested by "user agents" in "intranets".

### 2.2 Service Agents

The service registers itself with the service agent so that the SA can start advertising this information over the network. The process of registration consists of the service giving the SA all relevant configuration information and attribute tag/value list pairs specific to this service. The Service template is an abstract schema that

applies to the service type. The service template for TN3270E is shown later, contains the URL which is the address of the server with the port, which should be used to connect to it. The URL also contains the service type which in this case is TN3270. The template also contains all the other attributes associated with this service.

### 2.3 User Agents

The User Agent working on the TN3270E client's behalf retrieves service information from the Service Agent(s) or a Directory Agent. Based on the gathered information and required attributes the TN3270E client or user can decide whether or not to connect with a particular server. Based on the service advertisements from various TN3270E servers, the client looks at the load attribute and can decide to connect to the least loaded server. If by the time it connects to that particular TN3270E server, the server becomes unavailable it can try connecting to the next server in its list (ie: the second least loaded server whose advertisement was retrieved by the client/user agent).

## 3. TN3270E Server Environment and Load

TN3270E Servers are pervasive in today's networked environment. SLP provides emulator clients with a way to discover TN3270E servers in the network and session balance among the servers. The TN3270E servers could be distributed across different SNA gateways with different connection methods to hosts. The use of LU pools provides an easy way for administrators to provide users access to hosts. Administrators can add users to LU pools that have pre-configured LU's with specific attributes, like LU types and model types.

These LU pools would typically have LUs from several different gateways assigned, and as members of the LU pool make TN3270E session connections, they would be making connections to different TN3270E servers, with different load factors, so that session balancing could be accomplished. The use of LU pools is not a requirement for SLP and session balancing. A TN3270E client could obtain a session by using SLP and session balancing to locate the least loaded server in the network. On a service request a wild card "\*" could be used when asking for LUPool if the emulator doesn't care which device types are supported in given pools or if it can assume given pools support only certain device types.

### 3.1 TN3270E Server Load

TN3270E servers providing load information, SHOULD include number of sessions available, not in current use, as part of the calculation in determining the total load for the server. There can be other

factors that might have an effect on server load. An example would be if a server is not dedicated to only SNA traffic, and is handling other processes, like file services and print services, etc. It is beyond the scope of this document to standardize the method of individual server load calculations. Different vendors server's may calculate load information based upon factors they consider important, and methods for calculating load may change over time.

If the TN3270E server coexists in a network with other TN server implementations using SLP for session balancing, TN3270E server load could be adjusted to compensate for differences in load calculations.

One way to allow TN3270E server administrators to compensate for differences in implementations of calculating server load measurement is to provide the ability to modify the load calculation on the TN3270E server. An element of control can be provided by allowing the administrator to modify the load measurement, by using an integral number between 0 and 100 (100 being the highest) to change the load. This load measurement acts as an additional factor on the server's actual load calculation, so that the administrator could bias up or down, the likelihood of that server being selected by a TN3270E client.

Load MUST be defined as one of the attributes for the TN3270E server. The Load attribute provided at the server will allow clients to determine which server to make a connection. If a UA provides only a Service Type, in an Attribute Request, then the reply includes all attributes and all values for that Service Type, and Load would be included. Attribute Requests MAY include a select clause, so you could be returned just load information. For more information on Attribute Requests refer to Service Location Protocol [1].

An application could issue a Service Request to locate a TN3270E server. Then an application designed to perform least-load location of a TN3270E service, could issue a series of Attribute Requests to obtain the load measurement of each server specified with a URL. It would specify a select clause similar to the one below to receive only load information.

URL = service:tn3270://9.37.51.254:23 Attribute filter = LOAD

The attribute LOAD would be returned along with its value. The application could then issue other Attribute Request calls for each URL.

The application would then select the least loaded server as a connection target. If it tries to connect to a server and that connection fails, it could then try to connect to the next least-loaded server.

#### 4. TN3270E Client Configuration

##### 4.1 SLP Scope

Scope is a parameter used to control and manage access by clients to servers in a network. It is the same as the Service Location Protocol scope defined in [RFC 2165](#) [1]. The control scope provides is necessary for two reasons:

As your network, the number of clients, and the number of servers grow, it becomes necessary to partition access to those servers by the growing number of clients in order to reduce overall traffic on the network. It allows administrators to organize users and servers into administrative groups.

The meaning of the values of scope is defined by the administrator of the network. These values can represent any entity. Commonly, they fall along either departmental, geographical, or organizational lines.

Each TN3270E server can be assigned to a single scope or multiple scopes. TN3270E clients using these servers can be configured for a single specific scope. If TN3270E clients are not configured with a scope they MUST use the scope "default".

SLP Service Agents and Directory Agents (DA) need to reside in the network that support the TN3270E server with configured scopes.

Attribute information for Service Types pertaining to a specific scope can be obtained from Directory Agents (DA). The DA will not return a result unless the requested scope matches.

For more information on SLP scope refer to Service Location Protocol [1].

##### 4.2 DA Discovery time-out

The DA Discovery time-out value, is used to control how long the SLP API must wait to discover Directory Agents (DAs) in the network. The discovery request is a multicast, and the amount of time required to gather all DA responses might vary depending on many factors. If there are no DAs in the network, this time-out value can be set to

zero to indicate that no DA discovery is to be done. The time-out is expressed in milliseconds. Time-out intervals and default values should be handled as described in [RFC 2165 \[1\]](#).

#### 4.3 SA Multicast time-out

The SA Multicast time-out value, is used to control how long the SLP API must wait to discover services, attributes, or service types in a network without at least one DA that supports the scope of the request. In this situation, these requests are multicast and the User Agent waits the time-out value to gather the multiple responses that are returned. The time-out is expressed in milliseconds. Time-out intervals and default values should be handled as described in [RFC 2165 \[1\]](#).

### 5. TN3270E Client Implementation Information

#### 5.1 Overview

A TN3270E client that implements TN3270E SLP session balancing does not need to configure an IP Host Address or TCP Port for the TN3270E server it desires to connect to. Instead, the IP Host Address and TCP Port of the least loaded TN3270E server is discovered by using the SLP session balancing described in this document.

The discovery of the least loaded TN3270E server is done entirely outside of and before the TN3270E telnet negotiation. Once the IP Host address and TCP Port of the least loaded TN3270E server is discovered, the TN3270E client can then start normal TN3270E telnet negotiation.

The TN3270E client MUST allow for configuration of the following parameters. These SLP specific configuration items are covered by configuration parameters in the SLP API [\[5\]](#).

##### Enable SLP Session Balancing

This configuration parameter indicates whether or not SLP session balancing is enabled. If it is enabled the following three configuration parameters MUST also be configurable. If this parameter is disabled, SLP session balancing is not supported and normal TN3270E telnet negotiation is performed.

**Scope Name** The scope name is a text string that specifies a group of TN3270E servers. The scope name can be used to identify groups of TN3270E servers in a departmental or geographic setting. For

example, if the scope name is Building-D, then the SLP session balancing would search all TN3270E servers in the Building-D scope to find the least loaded TN3270E server.

If the scope name is blank, then the scope name is not used. This is referred to as unscoped. It should be noted as in [section 4.1](#) above that any reference to unscoped services applies to Service Location Protocol version 1 only [1]. Service Location Protocol version 2 doesn't allow unscoped services but does allow the use of default scope [4]. In this case all TN3270E servers, with or without scope names, can be used to satisfy the request for least loaded TN3270E servers. In order to cut down on network overhead, it is recommended that either all servers be scoped or no servers be scoped. Refer back to [section 4.1](#) for more discussion of scope.

#### DA Discovery Time Out Value

This value is specified in milliseconds and is fully described in [section 4.2](#) of this document.

#### SA Multicast Time Out Value

This value is specified in milliseconds and is fully described in [section 4.3](#) of this document.

## 5.2 How to obtain the list of TN3270E servers supporting SLP

A TN3270E client that implements SLP session balancing uses API calls to obtain the list of TN3270E servers supporting SLP session balancing.

The following Service Location Version 2 API [5] calls, could be used with TN3270E SLP session balancing:

SLPOpen - returns an SLPHandle handle to be used

SLPFindSrvs - issues the query for services

SLPFindAttrs - returns service attributes matching the attribute ids for the indicated service URL or service type.

SLPClose - frees all resources associated with the handle.



### 5.3 TN3270E Sample Client Flow

#### 5.3.1 Open the SLP connection

The TN3270E client must first open a handle with the SLP User Agent. For Service Location Protocol version 2, SLPOpen API call [5] The SA multicast time out and DA discovery time out values would be passed as parameters to the SLPOpen API call.

#### 5.3.2 Query the list of TN3270E servers

The TN3270E client then queries for the list of TN3270E servers supporting SLP. This is done by using the Service Request API call.

The request string contains information that determines which type of TN3270E servers that this client desires to connect to. The request string can contain the scope name, pool name, session type and 3270 screen size.

The SLPv1 query string has the following format:

```
TN3270/<scope name>/LUPPOOL/ == <pool name><TAB><device type>
```

The <scope name> is the name of the scope that is configured for the TN3270E client. If the scope is blank or null (unscoped request), then the scope is not inserted into the request string.

The <pool name> is a 1 to 8 character upper case string that indicates the name of the pool to which the TN3270E client desires to connect. For SLP session balancing, the same pool name must be configured on different TN3270E servers.

The <TAB> is the '/'t' tab character which is hexadecimal 0x09. the <TAB> is a literal and is used as a separator.

The <device type> can be any of the following:

```
3270DSC for TN3270E device type IBM-3287-1
3270002 for TN3270E device types IBM-3278-2 and IBM-3278-2-E
3270003 for TN3270E device types IBM-3278-3 and IBM-3278-3-E
3270004 for TN3270E device types IBM-3278-4 and IBM-3278-4-E
3270005 for TN3270E device types IBM-3278-5 and IBM-3278-5-E
* for TN3270E device type IBM-DYNAMIC
```

Example:

For a TN3270E client searching for TN3270E servers in the ENGINEERING scope for a model 2 screen size and LUPPOOL name pool2, the following request SLPv1 string would be constructed:

```
"TN3270/ENGINEERING/LUPPOOL/ == POOL2<TAB>3270002"
```

Note: The " characters before and after the string are not part of the request string.

### 5.3.3 Forward Looking Example for SLPv2

For SLPv2 the scope and service type are no longer part of the query string. These are now separate fields in the message. The service type name is required to have the "service:" prepended. The service type field would look like "service:TN3270", and the scope field would be a comma separated list of scopes. A scope name is always required in SLPv2, if no other name is known, the scope name "DEFAULT" is used. The example below uses the same parameters as used in above [section 5.3.2](#).

Example: Service Type: service:TN3270 Scope string: ENGINEERING The query string would have the following format:

```
(LUPPOOL=<POOL2> <32700002>)
```

In SLPv2 queries, all whitespace is compressed to a single space character during matching, so the identity of the separator character does not matter. The tab character could be added for readability, but it will not affect the outcome of the query.

### 5.3.4 Determine loading of each TN3270E server

An attribute request for "service:tn3270e" specifying the attribute LOAD can be made and you will get back all the available loads. Say these are 35,88,78. You can then issue a service request for all tn3270E servers with "LOAD<40" for instance. Even if the load changes between the time you get the attribute reply and when you issue the request, you will still get the best the network has to offer.

The TN3270E client then uses the TN3270E server's IP Host address to start normal Telnet TN3270E negotiation.

#### 5.4 Recommendations

The TN3270E client SHOULD display the IP hostname and TCP Port that is being used for the TN3270E connection. This gives the user knowledge of which TN3270E server the session is connected to. For example, the IP host address could be displayed in the window system status bar.

The TN3270E client SHOULD display the resource name that is returned by the TN3270E server after connection and TN3270E negotiation is completed. This gives the user knowledge of which LU resource name in the LUPPOOL the session is connected to. For example, the resource name could be displayed in the Windows status bar, or even in the 3270 OIA line.

In the event that after the TN3270E client has determined the least loaded server and the connection to that server fails for some reason, the connection should be closed and an attempt made to connect to other TN3270E servers in the list of least loaded servers.

For example, a TN3270E server may reject a connection to a specific pool if the pool is full, or if the device type does not match what is available in the pool. If this occurs, then an attempt to other least loaded TN3270E servers SHOULD be performed.

#### 6. Sample Trace Flow of SLP and Session Balancing

This sample trace flow is provided for informational purposes only.

```
SLP API: Service Request: TN3270//LUPPOOL == POOL2 3270002/
```

```
SLP API: Service Reply: service:tn3270://206.109.45.139:23
```

```
SLP API: Service Reply: service:tn3270://206.109.45.140:23
```

```
Connecting to 206.109.45.139:23...
```

```
TerminalType=NVT
```

```
Connection established
```

```
Recv <- DO TN3270E
```

```
Send -> WILL TN3270E
```

```
TerminalType=TN3270E
```

```
Recv <- SEND DEVICE_TYPE
```

Send -> DEVICE\_TYPE REQUEST IBM-3278-2-E CONNECT POOL2

Recv <- DEVICE\_TYPE IS IBM-3278-2-E CONNECT TN8003

Send -> FUNCTIONS REQUEST BIND\_IMAGE SYSREQ

Recv <- FUNCTIONS IS BIND\_IMAGE SYSREQ

## 7. Service Templates and Service Registration

The Service Location Protocol uses the "service:" URL scheme name to define URLs called "service: URLs". These schemes provide a way for clients to obtain configuration information that is needed to establish a 3270 session through the TN3270E server. The Service Location Protocol provides for service: URLs to be registered and discovered.

**Service Registration** These service registrations contain a service: URL, and possible attributes associated with that service. The service registration information are shown below for the server.

**Service Templates** Service templates are documents defining in a formal way the attributes associated with that service that a client may want to use. For more information on service templates please refer to, Service Templates and service: Schemes. [2]. The server service template and TN3270 service templates are shown below.

### 7.1 The TN3270E Service Type Template

The 'service:tn3270:' template defined below conforms to the grammar described in "Service Templates and service: Schemes". Please refer to [2] for detailed explanation of the syntax.

Name of submitters: Jim Naugle <jnaugle@us.ibm.com>  
Gregg Ledford <gledford@zephyrcorp.com>  
K. Kasthurirangan <kasthuri@us.ibm.com>

Language of service template: en

Security Considerations:

Service Location Protocol can help clients discover security services supported by the TN3270E server. If security services are important or required, using SLP authentication, and protected scopes in Service Location Protocol version 1 is recommended [1]. Well known ciphersuite names are used in the template [3].

Template text:

-----template begins here -----

template-type=tn3270e

template-version=1.0

template-description=

The tn3270 service provides 3270 gateway access to an SNA network via the TN3270 protocol. The attributes reflect the types of 3270 devices, LU Pools, and load information available on the server.

template-url-syntax=

```
# service:tn3270://<hostname>:<port>
# <hostname>
# <port>
```

load=integer

```
# This is the load balancing quantity to use in determining the
# least loaded TN3270E server to attach to for the service. The
# range of valid values is an integral 0 to 100 with 0 indicating the
# lowest possible load and 100 the highest
```

LUPool=string X M L

```
# This attribute takes on one or more values as defined below.
# The <TAB> char. 0x09 is literal and will be used as a separator.
```

```
#
#
# <pool name> = <name> / <name> "<TAB>" <dev type>
# <name>      = 1*ALPHANUM
#              "3270DSC"
```

```
#
#
# Identifies the LU pool names of LU pools available for use on this
# service with the associated device types supported in each pool.
# Each value is a record where the first token is the pool name of
# the pool and the second token is a device type supported in that
# pool. A pool name without a device type indicates that LUs of
# unknown type are included in the pool. Records associated with a
# given pool name are repeated for each supported device type. A
# given pool is included in a registration request if any PU profile
# that contributes at least one LU to the pool is active on the
# server. The range of valid dev_types are:
```

```
#
# dev_type      Meaning
#
```

```
# 3270002      Lu Type 2 Model 2
#
# 3270003      Lu Type 2 Model 3
#
# 3270004      Lu Type 2 Model 4
#
# 3270005      Lu Type 2 Model 5
#
# 3270DSC      Printer LU
#
```

BIND=keyword

# The server supports the SNA bind image TN3270E function.

DATA=keyword

# The non-SNA 3270 data stream is supported by server.

RESPONSES=keyword

# The server supports SNA response mode.

SCS=keyword

# The server supports SNA 3270 SCS data stream.

SYSREQ=keyword

# The SYSREQ keyboard key is supported on server.

RFC1576=keyword

# RFC1576 options supported.

RFC1646=keyword

# RFC1646 options supported.

RFC2355=keyword

# RFC2355 options supported.

security=string M

# This is the security technique supported on the server.

# The defined values are:

NONE

SSLV3

Ciphersuites=string M

# Cipher specifications supported by this server.

# Additional values will be defined in future templates.

NULL\_NULL,

NULL\_MD5,

NULL\_SHA,

RC4\_MD5\_EXPORT,

```
RC4_MD5_US,  
RC4_SHA_US,  
RC2_MD5_EXPORT,  
DES_SHA_EXPORT,  
TRIPLE_DES_SHA_US
```

```
platform=string X
```

```
# This is the network operating system platform underlying the  
# advertising service. The defined values are:
```

```
#  
# IW          Server uses IntranetWare or NetWare operating system  
#  
# NT          Server uses the Microsoft NT operating system  
#  
# OS2         Server uses the OS2 operating system  
#  
# AIX         Server uses the AIX operating system  
#  
IW,NT,OS2,AIX
```

```
protocol=string X
```

```
# This is the protocol(s) supported by the server providing this  
# service. The defined values are:
```

```
#  
# IP          Server supports client connections over IP (TCP/IP or  
#             UDP/IP)  
#  
# IPX         Server supports client connections over IPX (SPX/IPX)  
#  
IP,IPX
```

```
server name=string
```

```
# This is the name of the server that was configured during  
# installation.
```

```
release=string X
```

```
# This is the version and release level of the server advertising  
# services. Its format is vv.rr.mm where "vv" is the major version  
# number, "rr" is the minor version number, and "mm" is the  
# modification level. All numbers are padded on the left with zeroes  
# to two characters.  
# Example: version 3, release 0, mod level 0 is "03.00.00"
```

```
-----template ends here -----
```

## 7.2 Server Service Type Template

The 'service:commserver:' template defined below conforms to the grammar described in "Service Templates and service: Schemes". Please refer to [2] for detailed explanation of the syntax.

Name of submitters: Jim Naugle <jnaugle@us.ibm.com>  
 Gregg Ledford <gledford@zephyrcorp.com>  
 K. Kasthurirangan <kasthuri@us.ibm.com>

Language of service template: en

Security Considerations:

Service Location Protocol can help clients discover security services supported by the TN3270E server. If security services are important or required, using SLP authentication, and protected scopes [1] is recommended.

Template text:

```
-----template begins below this line-----

template-type=commserver

template-version=1.0

template-description=
  The server service type is registered whenever the communications
  software is loaded on the server. It describes generic attributes of
  the server. These attributes are also repeated on the other service
  types provided.

template-url-syntax=
  # service:commserver://<hostname>:<port>
  # <hostname>
  # <port>

platform=string X
  # This is the network operating system platform underlying the
  # advertising service. The defined values are:
  #
  # IW          Server uses Novell IntranetWare or NetWare operating
  #             system
  # NT          Server uses the Microsoft NT operating system
  #
  # OS2         Server uses the OS2 operating system
  #
  # AIX         Server uses the AIX operating system
  #
  IW,NT,OS2,AIX
```



```
protocol=string X
#   This is the protocol(s) supported by the server providing this
#   service.  The defined values are:
#
# IP           Server supports client connections over IP (TCP/IP or
#               UDP/IP)
#
# IPX          Server supports client connections over IPX (SPX/IPX)
#
IP,IPX

server name=string
# This is the name of the server that was configured during
# installation.

release=string X
# This is the version and release level of the commserver
# advertising services.  Its format is vv.rr.mmm where "vv" is the
# major version number, "rr" is the minor version number, and "mmm"
# is the modification level.  All numbers are padded on the left with
# zeroes to two characters.
#
# Example: version 3, release 0, mod level 0 is "03.00.00"

-----template ends above this line-----
```

### 7.3 Template Contact Information

Jim Naugle <jnaugle@us.ibm.com>  
Kasthuri Kasthurirangan <kasthuri@us.ibm.com>  
Gregg Ledford <gledford@zephyrcorp.com>

### 7.4 Security Considerations

Service type templates provide information that is used to interpret information obtained by the Service Location Protocol. If these templates are modified or if false templates are distributed, services may not correctly register themselves, or clients might not be able to interpret service information.

The service: URLs themselves specify the service access point and protocol for a particular service type. These service: URLs could be distributed and indicate the location of a service other than that normally wanted to used. SLP [1] provides an authentication mechanism that allows service: URLs of registered services to be signed and for the signatures to be verified by clients.

Service Location Protocol can help clients discover security services supported by the TN3270E server. If security services are important or required, using SLP authentication, and protected scopes [1] is recommended.

#### 7.5 Sample TN3270 Service Registration Message

URL: service:tn3270://<addr-spec>:<port-number> Attributes:

```
[(SCOPE=<string>),]  
  
(RELEASE=03.00.00),  
  
(PLATFORM=IW),  
  
(PROTOCOL=IP),  
  
(SERVERNAME=<string>),  
  
(LOAD=<integer 0 to 100>),  
  
[(LUPPOOL=pool-name0/tANY,  
pool-name1/tdevice_type1,  
pool-name2/tdevice-type2, ...  
pool-namen/tdevice-typen)]  
  
BIND,  
  
DATA,  
  
RESPONSES,  
  
SCS,  
  
SYSREQ,  
  
(SECURITY=NONE),  
  
RFC1576,  
  
RFC1646,  
  
RFC2355
```

## 7.6 Sample Server Service Registration Message

URL:service:commserver://<addr-spec>:<port-number>

Attributes: [(SCOPE=<string>),]

(RELEASE=03.00.00),

(PLATFORM=IW),

(PROTOCOL=IP),

(SERVERNAME=<string>)

## 8. References

- [1] Veizades, J., Guttman, E., Perkins, C., and S. Kaplan, "Service Location Protocol", [RFC 2165](#), July 1997.
- [2] Guttman, E., Perkins, C. and J. Kempf, "Service Templates and service: Schemes", [RFC 2609](#), June 1999.
- [3] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", [RFC 2246](#), January 1999.
- [4] Guttman, E., Perkins, C., Veizades, J. and M. Day, "Service Location Protocol Version 2", [RFC 2608](#), June 1999.
- [5] Kempf, J. and E. Guttman, "An API for Service Location", [RFC 2614](#), June 1999.
- [6] Bradner, S., "Key Words for Use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

## 9. Authors' Addresses

Jim Naugle  
IBM  
P.O. Box 12195  
Research Triangle Park, N.C. 27709-2195  
USA

Phone: (919) 254-8789  
EMail: [jnaugle@us.ibm.com](mailto:jnaugle@us.ibm.com)

Kasthuri Kasthurirangan  
IBM  
P.O. Box 12195  
Research Triangle Park, N.C. 27709-2195  
USA

Phone: (919) 254-5721  
EMail: [kasthuri@us.ibm.com](mailto:kasthuri@us.ibm.com)

Gregg Ledford  
Zephyr Development Corporation  
8 Greenway Plaza Suite 1400  
Houston, Texas 77046  
USA

Phone: (713) 623-0089  
EMail: [gledford@zephyrcorp.com](mailto:gledford@zephyrcorp.com)

## 10. Full Copyright Statement

Copyright (C) The Internet Society (2001). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.