

IBM Communications Server for Linux



管理指南

版本 6.2.2

IBM Communications Server for Linux



管理指南

版本 6.2.2

注意:

在使用本资料及其支持的产品之前, 请一定要阅读第 159 页的附录 C, 『声明』中的一般信息。

第三版 (2006 年 7 月)

本版本适用于 IBM Communications Server for Linux V6.2.2 及所有后续发行版和修订版, 直到在新版本或技术简报中另有声明为止。

通过您当地的 IBM 代表或 IBM 分部可订购出版物。以下地址不备有出版物。

IBM 欢迎您提出宝贵意见。读者意见表格在此出版物的背面提供。如果已除去该表格, 则您可以将意见寄往以下地址:

IBM 中国公司上海分公司, 汉化部
中国上海市淮海中路 333 瑞安广场 10 楼
邮政编码: 200021

如果您喜欢通过电子形式发送意见, 则可使用下列其中一种方法:

- IBMLink: [ibmcn\(ctsrcrf\)](mailto:ibmcn(ctsrcrf)@cn.ibm.com)
- 互联网: ctsrcrf@cn.ibm.com
- 中国: 021-63857881
- 其他国家或地区: (86-21)63857881

当您发送信息给 IBM 后, 即授予 IBM 非专有权, IBM 可以它认为合适的任何方式使用或分发此信息, 而无须对您承担任何责任。

© Copyright International Business Machines Corporation 1998, 2006. All rights reserved.

目录

| | |
|---|-----------|
| 表 | vii |
| 图 | ix |
| 关于本书 | xi |
| 谁应使用本书 | xi |
| 如何使用本书 | xi |
| 本书的结构 | xii |
| 印刷约定 | xii |
| 图形约定 | xiii |
| 本发行版的新增内容 | xiii |
| 新功能 | xiv |
| 已引退的功能 | xiv |
| 相关出版物 | xiv |
| 第 1 章 SNA 术语与概念 | 1 |
| 系统网络体系结构 | 1 |
| 基本 SNA 概念 | 1 |
| 网络类型 | 2 |
| SNA 节点 | 2 |
| 连接 | 4 |
| 事务程序 | 4 |
| 应用程序编程接口 | 5 |
| 网络可访问单元 | 5 |
| 会话 | 7 |
| 对话 | 9 |
| 方式 | 10 |
| 路由选择 | 10 |
| 服务等级 | 11 |
| 基本 APPN 概念 | 11 |
| APPN 节点类型 | 12 |
| APPN 控制点 | 13 |
| 找到资源 | 14 |
| 会话路由 | 17 |
| 分支扩展程序 | 23 |
| 访问 APPN 网络的子区域网络 | 24 |
| 第 2 章 管理 Communications Server for Linux | 27 |
| Communications Server for Linux 管理概述 | 27 |
| 管理责任 | 27 |
| 管理工具 | 28 |
| 管理许可权 | 32 |
| 规划 Communications Server for Linux 配置 | 33 |
| 规划工作表 | 33 |
| 任务表 | 33 |
| 启用与禁用本地系统上的 Communications Server for Linux | 33 |
| 指定 Communications Server for Linux 程序的路径 | 34 |
| 启用 Communications Server for Linux 服务器 | 34 |
| 禁用 Communications Server for Linux 服务器 | 35 |
| 使用 Motif 管理程序 | 36 |

| | |
|-------------------------|-----------|
| 调用 Motif 管理程序 | 36 |
| “资源”窗口 | 37 |
| 资源对话框 | 42 |
| 状态对话框 | 44 |
| 帮助窗口 | 44 |
| 使用命令行管理程序 | 45 |
| 第 3 章 基本配置任务 | 47 |
| 配置客户机 / 服务器功能 | 47 |
| 配置节点 | 48 |
| 节点配置参数 | 48 |
| 其他配置 | 49 |
| 配置记录 | 49 |
| 第 4 章 定义连接组件 | 51 |
| 定义 DLC、端口和连接网络 | 52 |
| DLC、连接网络以及端口配置参数 | 53 |
| 其他配置 | 55 |
| 定义链路站 | 55 |
| 链路站配置参数 | 56 |
| 其他配置 | 61 |
| 定义 DLUR PU | 61 |
| DLUR PU 配置参数 | 62 |
| 下游节点的传递 DLUR 的参数 | 63 |
| 其他配置 | 63 |
| 第 5 章 配置从属 LU | 65 |
| 定义 0-3 型 LU | 65 |
| 0-3 型 LU 配置参数 | 65 |
| 其他配置 | 66 |
| 定义 LU 池 | 67 |
| LU 池配置参数 | 67 |
| 第 6 章 配置 APPC 通信 | 69 |
| 定义本地 LU | 69 |
| 本地 LU 配置参数 | 70 |
| 其他配置 | 71 |
| 定义远程节点 | 71 |
| 远程节点配置参数 | 72 |
| 其他配置 | 72 |
| 定义伙伴 LU | 72 |
| 伙伴 LU 配置参数 | 73 |
| 定义伙伴 LU 的链路站路由 | 74 |
| 其他配置 | 75 |
| 定义 TP | 75 |
| 服务器上的 TP 调用参数 | 77 |
| TP 定义参数 | 79 |
| 定义方式和服务等级 | 80 |
| 方式配置参数 | 82 |
| 其他配置 | 84 |
| 定义 CPI-C 辅助信息 | 84 |
| CPI-C 配置参数 | 84 |
| 其他配置 | 86 |
| 配置 APPC 安全性 | 86 |
| 配置会话安全性 | 86 |
| 配置对话安全性 | 87 |

| | |
|--|------------|
| 配置安全性访问列表 | 87 |
| 第 7 章 配置用户应用程序 | 89 |
| 第 8 章 配置传递服务 | 91 |
| 配置 TN 服务器 | 91 |
| 配置 TN 服务器访问记录 | 92 |
| 配置 TN 服务器关联记录 | 94 |
| 配置 TN 重定向器 | 95 |
| 配置 TN 重定向器访问记录 | 95 |
| 配置 SNA 网关 | 98 |
| 下游 LU 配置参数 | 99 |
| 其他配置 | 99 |
| 配置 DLUR | 99 |
| 第 9 章 从 NetView 中管理 Communications Server for Linux | 101 |
| 使用主机 NetView 程序 | 101 |
| NetView 屏幕显示 | 102 |
| 更改命令输入区的大小 | 102 |
| RCF 命令语法概述 | 102 |
| 大写字符和转义字符 | 103 |
| 使用 SPCF | 103 |
| 对用于 SPCF 的管理命令的限制 | 104 |
| SPCF 命令的示例 | 104 |
| 使用 UCF | 105 |
| UCF 命令语法 | 105 |
| 允许的命令 | 106 |
| UCF 命令的示例 | 106 |
| Linux 系统命令的输出 | 106 |
| 取消命令 | 107 |
| UCF 安全性 | 107 |
| 第 10 章 管理 Communications Server for Linux 客户机 / 服务器系统 | 109 |
| 更改客户机 / 服务器配置 | 110 |
| 将客户机移到另一个域中 | 110 |
| IP 联网需求 | 111 |
| 设置 IP 端口号 | 111 |
| LAN 访问超时 | 112 |
| 对 Remote API Client 的 HTTPS 访问 | 112 |
| 在 Windows 上管理 Remote API Client | 113 |
| 在 Windows 上启用 Remote API Client | 114 |
| 在 Windows 上查看 Remote API Client 的状态 | 114 |
| 在 Windows 上禁用 Remote API Client | 114 |
| Windows 上的 Remote API Client 配置 | 115 |
| 在 AIX 或 Linux 上管理 Remote API Client | 126 |
| 在 AIX 或 Linux 上启用和禁用 Remote API Client | 126 |
| 客户机网络数据文件 (sna_clnt.net) | 126 |
| 定义客户机 TP | 129 |
| 附录 A. 配置计划工作表 | 131 |
| 节点工作表 | 131 |
| APPN 网络节点 | 131 |
| APPN 端节点 | 132 |
| APPN 分支网络节点 | 132 |
| 低入口联网节点 | 133 |
| 连接工作表 | 133 |

| | |
|--|------------|
| SDLC | 134 |
| 令牌环. | 136 |
| 以太网. | 137 |
| QLLC (X.25) | 139 |
| 多路径通道 | 140 |
| 企业扩展程序 (HPR/IP) | 141 |
| 传递服务工作表. | 142 |
| 本地节点上的 DLUR | 142 |
| 下游节点的传递 DLUR | 143 |
| SNA 网关 | 143 |
| TN 服务器 | 144 |
| TN 重定向器 | 144 |
| 用户应用程序支持工作表. | 145 |
| APPC | 146 |
| CPI-C | 148 |
| 5250 | 149 |
| 3270 | 149 |
| LUA | 150 |
| 附录 B. 从命令行配置可调用 TP. | 151 |
| 可调用 TP 定义的文件格式. | 152 |
| 附录 C. 声明. | 159 |
| 注册商标 | 161 |
| 书目提要. | 163 |
| Communications Server for Linux V6.2.2 出版物 | 163 |
| 系统网络体系结构 (SNA) 出版物. | 164 |
| 主机配置出版物. | 165 |
| z/OS Communications Server 出版物 | 165 |
| TCP/IP 出版物 | 165 |
| X.25 出版物 | 165 |
| APPC 出版物 | 165 |
| 编程出版物 | 165 |
| 其他 IBM Networking 出版物 | 166 |
| 索引 | 167 |

表

| | |
|------------------------------|-----|
| 1. 印刷约定 | xii |
| 2. 标准方式和 COS 名. | 81 |
| 3. 在 RCF 命令中使用转义字符 | 103 |



| | |
|---|-----|
| 1. SNA 子区域网络 | 3 |
| 2. 多个会话和并行会话 | 9 |
| 3. 事务程序与逻辑单元之间的通信 | 10 |
| 4. 样本 APPN 网络的一部分 | 12 |
| 5. 低入口联网节点目录 | 15 |
| 6. 端节点目录 | 16 |
| 7. 网络节点目录 | 16 |
| 8. 网络节点中的网络拓扑数据库 | 19 |
| 9. 使用共享访问传输设施的 APPN 网络 | 21 |
| 10. 从节点 EN1 至 APPN 网络中的每个节点的直接链路所需要的定义 | 22 |
| 11. 使用虚拟节点的直接链路所需要的定义 | 23 |
| 12. 分支扩展程序 | 24 |
| 13. Communications Server for Linux 域窗口 | 38 |
| 14. 节点窗口 | 39 |
| 15. Communications Server for Linux 工具栏 | 41 |
| 16. 样本对话框 | 43 |
| 17. 样本状态对话框 | 44 |
| 18. 样本帮助窗口 | 45 |
| 19. NetView 屏幕的示例 | 102 |

关于本书

本书指导您启用、配置和管理 IBM® Communications Server for Linux®, 它是一种使运行 Linux 的计算机能够与 SNA (系统网络体系结构) 网络中其他节点交换信息的 IBM 软件产品。

视运行 IBM Communications Server for Linux 的硬件而定, 该产品有两种不同的安装变体:

Communications Server for Linux

Communications Server for Linux (程序产品号为 5724-i33) 可在下列平台上运行:

- 运行 Linux (i686) 的 32 位 Intel® 工作站
- 运行 Linux (x86_64) 的 64 位 AMD64/Intel EM64T 工作站
- 运行 Linux (ppc64) 的 IBM pSeries® 计算机

Communications Server for Linux on System z

Communications Server for Linux on System z (程序产品号为 5724-i34) 可在运行 Linux for System z (s390 或 s390x) 的 System z 大型机上运行。

在本书中, 除非明确地描述了差别, 否则使用名称 Communications Server for Linux 来指示这两个变体中的任一变体, 并使用术语 “Communications Server for Linux 计算机” 来指示运行 Communications Server for Linux 的任何类型的计算机。

本书适用于版本 6.2.2 的 Communications Server for Linux。

谁应使用本书

本书供使用 Communications Server for Linux 的系统管理员和应用程序员使用。

系统管理员

系统管理员安装 Communications Server for Linux、配置系统进行网络连接以及维护系统。它们应该熟悉运行 Communications Server for Linux 的硬件以及 Linux 操作系统。他们还必须了解系统所连接的网络并大体上理解 SNA 概念。

应用程序员

应用程序员设计并编写事务程序与应用程序, 这些程序使用 Communications Server for Linux 编程接口来通过 SNA 网络发送和接收数据。他们应该完全熟悉 SNA、与事务程序或应用程序进行通信的远程程序以及 Linux 操作系统编程和操作环境。

手册提供了每个 API 的与编写应用程序相关的更详细信息。

如何使用本书

本指南说明如何启用、配置和管理 Communications Server for Linux。

本书的结构

本书结构如下：

- 第 1 页的第 1 章，『SNA 术语与概念』提供 SNA 和 APPN（Advanced Peer-to-Peer Networking[®]（高级对等联网））概念的概述。
- 第 27 页的第 2 章，『管理 Communications Server for Linux』描述 Communications Server for Linux 管理工具并说明如何准备 Communications Server for Linux 配置、如何在服务器上启用和禁用 Communications Server for Linux 软件以及如何使用 Motif 和命令行管理程序。
- 第 47 页的第 3 章，『基本配置任务』说明如何执行 Communications Server for Linux 服务器的基本配置任务，包括配置客户机 / 服务器操作、配置 SNA 节点和配置 Communications Server for Linux 的消息记录。
- 第 51 页的第 4 章，『定义连接组件』说明如何配置 Communications Server for Linux 节点的连接。
- 第 65 页的第 5 章，『配置从属 LU』说明如何配置 0-3 型 LU（逻辑单元）和 LU 池的从属 LU。
- 第 69 页的第 6 章，『配置 APPC 通信』说明如何配置 APPC（高级程序间通信）。
- 第 89 页的第 7 章，『配置用户应用程序』说明如何配置用户应用程序。
- 第 91 页的第 8 章，『配置传递服务』说明如何配置传递服务，传递服务支持主机系统与未直接连接的本地系统之间的通信。
- 第 101 页的第 9 章，『从 NetView 中管理 Communications Server for Linux』说明如何使用 Communications Server for Linux 远程命令设施（RCF）来管理 Communications Server for Linux 并从运行 NetView[®] 的主机在 Communications Server for Linux 节点上运行命令。
- 第 109 页的第 10 章，『管理 Communications Server for Linux 客户机 / 服务器系统』说明如何配置和管理 IBM Remote API Client。
- 第 131 页的附录 A，『配置计划工作表』提供 Communications Server for Linux 的配置工作表。
- 第 151 页的附录 B，『从命令行配置可调用 TP』提供关于命令行实用程序的信息，该实用程序使用户或 TP 安装程序的编写者能够定义可调用 TP。

印刷约定

表 1 显示了本文档使用的印刷样式。

表 1. 印刷约定

| 特殊元素 | 印刷样本 |
|----------------------|--|
| 强调的词 | 删除之前备份文件 |
| 文档标题 | 《Communications Server for Linux 管理指南》 |
| 文件或路径名 | /usr/spool/uucp/myfile.bkp |
| 程序或应用程序 | snaadmin |
| 参数或 Motif 字段 | <i>opcode; LU 名</i> |
| 用户可以输入的字面值或选项（包括缺省值） | 255; 在节点启动时 |
| Motif 按钮 | 状态 |
| Motif 菜单 | 服务 |
| Motif 菜单项 | 配置节点参数 |

表 1. 印刷约定 (续)

| 特殊元素 | 印刷样本 |
|-----------------|--|
| 用户输入 | Op1 |
| 计算机输出 | CLOSE |
| 命令或 Linux 实用程序 | define_node; cd |
| 对特定类型的所有命令的一般引用 | query_* (指示所有用于查询资源详细信息的 管理命令) |
| 选项或标志 | -i |
| 表示提供的值的变量 | <i>filename; LU_name; user_ID</i> |
| 返回值 | 0; -1 |
| 3270 键 | “执行” 键 |
| 键盘键 | Ctrl+D; Enter |
| 十六进制值 | 0x20 |
| 环境变量 | PATH |
| 函数、调用或入口点 | ioctl |
| 编程动词 | GET_LU_STATUS |

图形约定

AIX, LINUX

此符号用来指示仅适用于 AIX® 或 Linux 操作系统的文本部分的开始。它适用于 Linux 服务器和在 AIX、Linux、Linux for pSeries 或 Linux for System z 上运行的 IBM Remote API Client。

WINDOWS

此符号用来指示适用于 Windows® 上的 IBM Remote API Client 的文本部分的开始。

■

此符号指示特定于操作系统的文本部分的结束。此符号后面的信息与操作系统无关。

本发行版的新增内容

Communications Server for Linux V6.2.2 替换 Communications Server for Linux V6.2 和 Communications Server for Linux V6.2.1。

仍支持此产品的以下发行版:

- Communications Server for Linux V6.2
- Communications Server for Linux V6.2.1

此产品的以下发行版不再受支持:

- Communications Server for Linux V6.0.1 (V6.0.1), 它过去作为 PRPQ 5799-RQA 或 5799-RXL 提供。

新功能

以下功能已添加至本发行版的 Communications Server for Linux:

- 客户机 / 服务器支持现在包括大量服务器和客户机。
 - 服务器可以是下列任何一个。
 - 运行 Linux (i686) 的 32 位 Intel 工作站
 - 运行 Linux (x86_64) 的 64 位 AMD64/Intel EM64T 工作站
 - 运行 Linux (ppc64) 的 IBM pSeries 计算机
 - 运行 Linux for System z (s390 或 s390x) 的 System z 大型机
 - 客户机可以为服务器支持的任意 Linux 或 Linux for System z 平台、AIX 工作站或运行 32 位版本的 Microsoft® Windows (Microsoft Windows 2000、2003 或 XP) 或 x64 版本的 Microsoft Windows (Microsoft Windows Server 2003 x64 Edition 或 Microsoft Windows XP Professional x64 Edition) 的 PC。这些客户机可使用 TCP/IP 与 Communications Server for Linux 服务器 (或 CS/AIX 服务器) 进行通信, 或使用 HTTPS 通过 WebSphere® 服务器来与这些服务器进行通信。
 - Linux、Linux for System z 和 AIX 客户机支持 Linux 或 Linux for System z 服务器上提供的所有 API 库。(早期 CS/AIX v4.2 产品中提供的 API 在此版本中不受支持。)
 - Windows 客户机支持 APPC、CPI-C、CSV、LUA (包括 LUA RUI 和 SLI) 和 NOF API。
 - 对 NOF API 的客户机支持 (在所有客户机类型上) 仅提供查询功能。不提供对定义、删除、启动或停止资源的调用。
 - Windows 客户机可在 Windows Terminal Server (WTS) 环境中运行, 在此环境中, 两个或更多用户可共享同一客户机, 但其应用程序独立运行而不会互相干扰。
- 现在支持 SLES10 Linux 分发版 (适用于服务器和客户机)。
- Communications Server for Linux 现在支持在 Linux 服务器上使用 WAN 适配器的 SDLC 和 X.25 (QLLC) 连接。这需从适配器供应商处获取设备驱动程序和 DLC 代码; IBM 不提供这些适配器或设备驱动程序。有关合适的 WAN 适配器供应商的详细信息, 请参阅 Communications Server for Linux Support Web 页面。
- TN 服务器和 TN 重定向器现在提供指定 TN 传递服务将要侦听客户机连接所在的特定本地地址。此功能是可选的; 您可以将该服务配置为支持 TN3270 客户机在任何本地地址上进行连接, 或将它限制为指定的地址。

已引退的功能

客户机和服务器代码在 RedHat Advanced Server 2.1 上不再受支持。

相关出版物

有关 SNA、APPN 或 LU 6.2 体系结构的信息, 请参阅下列 IBM 文档:

- *IBM System/390 Principles of Operation* (SA22-7201)
- *IBM z/Architect Principles of Operation* (SA22-7832)
- *IBM Systems Network Architecture:*
 - *LU 6.2 Reference—Peer Protocols* (SC31-6808)

- *APPN Architecture Reference* (SC30-3422)
- *Management Services* (SC30-3346)
- *Formats* (GA27-3136)
- *Technical Overview* (GC30-3073)

相关出版物

第 1 章 SNA 术语与概念

本章定义系统网络体系结构 (SNA) 的术语和概念, 它们对于理解和使用 Communications Server for Linux 非常重要。有关 Communications Server for Linux、它的功能以及它实现所描述的不同 SNA 概念的方式的信息, 请参阅《*Communications Server for Linux 快速入门*》。如果您已熟悉 SNA 和 Communications Server for Linux, 则可以从第 27 页的第 2 章, 『管理 Communications Server for Linux』开始。

本章分为以下部分:

- 『系统网络体系结构』提供 SNA 的定义。
- 『基本 SNA 概念』说明适用于任何 SNA 网络的术语和概念。
- 第 11 页的『基本 APPN 概念』说明仅适用于支持 Advanced Peer-to-Peer Networking (高级对等联网) (APPN) 的 SNA 网络的术语和概念。
- 第 24 页的『访问 APPN 网络的子区域网络』介绍适用于组合了 SNA 和 APPN 的网络的术语和概念。

注: 本章没有打算提供 SNA 概念的完整参考信息。关于 SNA 的详细信息可以在第 xiv 页的『相关出版物』中列示的 SNA 出版物中找到。

系统网络体系结构

系统网络体系结构 (SNA) 是 IBM 的一种数据通信体系结构, 它指定了用于在各种各样的硬件和软件数据通信产品之间进行通信的公共约定。此体系结构由两类定义组成: 格式 (用于定义由网络组件交换的消息的布局) 和协议 (用于定义网络组件在响应消息时执行的操作)。

SNA 网络是链接在一起并使用 SNA 进行通信的计算机的集合。

最初, SNA 旨在能够与主机通信。每个网络或子网由主机控制; 其他计算机直接与主机通信, 但相互之间不通信。这种由主机控制的旧式网络通常被称为子区域 SNA。SNA 目前已扩展为支持在网络中的计算机之间进行直接的对等通信, 而不需要主机。这种新的同级联网就是 APPN。

许多 SNA 网络既具有子区域元素也具有对等联网元素。当网络从子区域 SNA 迁移至 APPN 时, 具有 APPN 能力的主机可以起到控制旧系统的作用, 同时还充当新系统的同级系统。同样, 单一计算机可以访问 (APPN 网络中的) 对等计算机和旧主机; 它与主机的通信由主机控制, 但它与其他计算机的通信是对等的, 因而不涉及主机。

基本 SNA 概念

SNA 定义了设备 (从大型机至终端) 使用的标准、协议和功能, 以使它们能够在 SNA 网络中相互通信。

SNA 功能分成具有层次结构的独立层，每个层都执行特定的一组功能。将网络功能分层的方法使网络设备能够共享信息并处理资源，而不需要有关网络中的每个设备的详细信息。工作站的用户可以与另一用户通信，而不需要了解任何关于网络中的物理设备或那些设备之间的连接的信息。

网络类型

SNA 支持下列类型的网络：

- 子区域网络是由子区域节点和外围节点组成的分层组织网络。子区域节点（如主机和通信控制器）处理一般网络路由。外围节点（如终端）连接至网络，而不需要了解一般网络路由情况。
- 同级网络是一种组织得很协调的网络，它由所有参与一般网络路由的同级节点组成。
- 混合网络是既支持主机控制的通信又支持对等通信的网络。

注：运行 Communications Server for Linux 的 Linux 系统可以在子区域网络中充当外围节点、在同级网络中充当同级节点或同时充当这两者。

SNA 节点

在 SNA 网络中，节点是具有关联软件组件的 Linux 系统或其他设备，节点实现了 SNA 协议，并且至少有一条至网络中的另一节点的通信路径。每个节点都管理网络通信路径的自己一端，并使用 SNA 协议来与每条路径另一端的节点进行通信。

因为子区域网络 and 同级网络以不同的方式来定义节点之间的关系，所以它们对节点类型也使用不同的术语（用来描述节点在网络中充当的角色）。

子区域网络中的节点类型

SNA 子区域网络支持下列节点类型：

- 子区域节点控制所有相连接的节点的通信和网络资源。SNA 按照子区域节点的能力和它们对其他节点拥有的控制程度来对子区域节点分类：
 - 5 型节点提供一些 SNA 功能，这些功能控制网络资源、支持事务程序、支持网络操作员并提供最终用户服务。因为这些功能通常是由主机处理器提供的，所以 5 型节点也被称为主机节点。由 5 型子区域节点控制的设备和资源组成了该节点的域。
 - 4 型节点提供一些 SNA 功能，这些功能在一部分网络中路由数据流并控制数据流。因为这些功能通常是由通信控制器提供的，所以 4 型节点也被称为通信控制器节点。
- 外围节点在子区域网络中充当下级角色。例如，外围节点可以支持 3270 仿真或从属 LU 6.2 通信。外围节点是分布式处理器、群集控制器或工作站之类的设备；它们还分为 2.0 型和 2.1 型节点：
 - 2.0 型节点总是由 4 型或 5 型节点控制。没有 4 型或 5 型节点的参与，它们无法与其他节点建立通信。2.0 型节点被称为从属节点。
 - 2.1 型节点可以充当从属节点，但它们还可以直接与其他 2.1 型节点通信。

注：运行 Communications Server for Linux 的 Linux 计算机可以作为 2.1 型或 2.0 型节点工作。

与外围节点相连接的 4 型或 5 型子区域节点充当边界节点。它通过在子区域节点使用的网络地址和外围节点使用的本地地址之间进行转换来执行边界功能。

简单的子区域网络包含下列组件:

主机 主机是与最初的 IBM System/370™ 兼容的大型机。主机通常是 5 型节点。但 Communications Server for Linux on System z 作为 2.1 型或 2.0 型节点在主机上运行。

通信控制器

通信控制器也被称为前端处理机 (FEP)，它是连接至主机的独立处理器。它管理主机与其他计算机的通信。

通信链路

通信链路将主机站点与最终用户站点连接在一起。用户与主机通常位于不同的站点上，因此两个站点需要通信链路来连接。

终端控制器

通信链路的远程端是终端控制器，也被称为群集控制器。它负责控制链路的使用，并将数据路由至终端。最著名的 IBM 终端控制器是 3174 和 3274。

终端 用户从终端运行主机应用程序或将工作提交给主机。最著名的 IBM 终端是 3270。可以通过终端控制器来连接终端，也可以将终端直接连接至通信控制器。

打印机 打印机 (如 IBM 3287) 也可以连接至终端控制器。它们可以接收主机的输出。

如图 1 所示，子区域网络的示意图看起来像是一棵倒置的树。

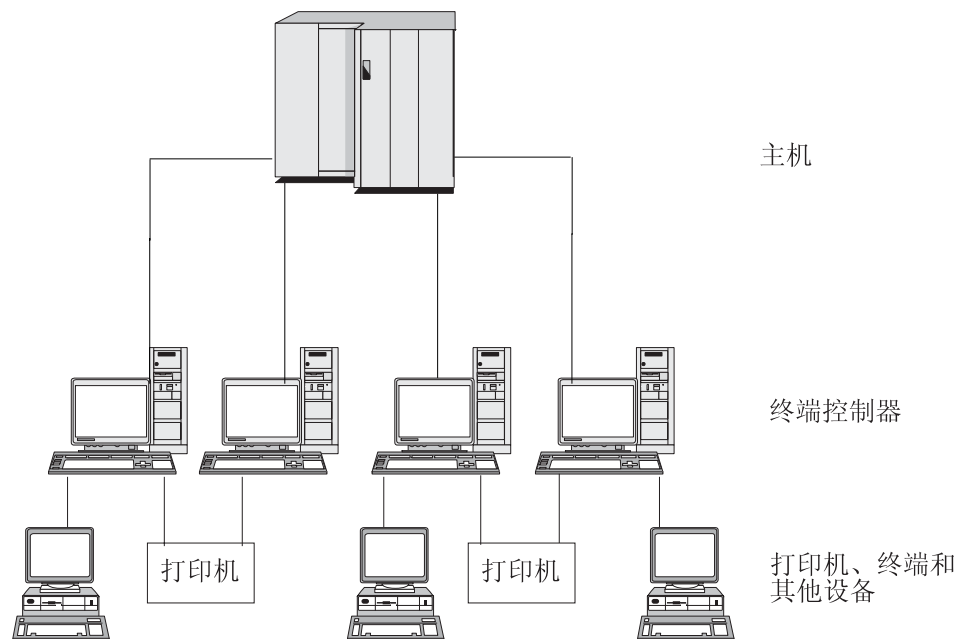


图 1. SNA 子区域网络

树根 (位于示意图的顶部) 是负责控制网络的计算机。树枝是从主机到网络中的其他计算机 (终端控制器) 的通信链路; 树叶 (位于示意图底部) 是连接至这些计算机的终端或打印机, 它们由用户访问。

此处描述的传统子区域 SNA 设置使用户能够使用单个主机系统的资源。这些终端只对终端控制器提供或从中接受简单的数据输入和显示功能; 终端控制器负责处理终端与主机之间的 SNA 通信。

终端控制器及其终端可以由使用诸如 Communications Server for Linux 之类的产品的 SNA 节点替换。从主机的观点来看，节点是作为终端控制器出现的。但它还向用户提供其他功能，如访问多个主机系统的能力以及用于定制屏幕显示的设施。另外，Communications Server for Linux 能够在也可用于执行与 SNA 不相关的其他任务的 Linux 计算机上运行（这与终端控制器不同，终端控制器仅用于与主机通信）。

同级网络中的节点类型

与子区域网络不同，同级网络不按层次将节点分类。与其他节点进行的交换不受主机或其他集中处理器控制。相反，任何节点都可以与任何其他节点建立通信。

同级网络由 2.1 型节点组成。同级网络中的节点可以充当下列角色：

- APPN 网络节点 (NN) 标识网络资源的位置、确定这些资源间的会话的路由、路由会话并为与网络节点直接连接的端节点 (EN) 和低入口联网 (LEN) 节点提供服务。APPN 网络节点的域由其本身和它为之提供网络服务的任何端节点组成。
- APPN 端节点可以访问远程资源，而不要求在该端节点上配置那些资源。端节点可以独立地与相邻节点通信，但需要网络节点服务器的服务才能访问非相邻节点。APPN 端节点的域仅包含其本身。
- APPN 分支网络节点允许将 APPN 网络分成多个分支，以简化其拓扑并减少网络管理开销。分支网络节点为从 APPN 主网络中分离出来的分支网络中的端节点提供网络节点功能，而同时又在主网络中充当端节点。有关更多信息，请参阅第 23 页的『分支扩展程序』。
- 低入口联网节点是不支持 APPN 功能的 2.1 型节点。它们可以与 APPN 网络中的相邻节点通信，但不参与 APPN 网络。在低入口联网节点中，必须明确地预定义与远程 LU 的所有潜在会话，或通过一个缺省条目来进行预定义，该条目指示所有远程 LU 都驻留在相邻网络节点上（可使用特定链路来访问该网络节点）。低入口联网节点的域仅包含其本身。

有关面向同级的节点类型的更多信息，请参阅第 12 页的『APPN 节点类型』。

连接

对于两个要进行通信的节点，每个节点都必须具有支持节点间的数据流的硬件和软件组合。硬件组件由每个节点上的适配器和连接两个适配器的传输介质组成。软件组件提供对硬件以及通过该硬件交换的数据的控制。

每个连接至网络的节点都具有一个或多个链路站，这些链路站是节点中的硬件和软件，它们控制着至特定相邻节点的数据流。要在两个相邻节点间建立通信，其中一个链路站必须首先激活节点间的链路。

事务程序

通过 SNA 网络交换信息的程序被称为事务程序 (TP)。

以下是可以包括 SNA TP 的应用程序的示例：

- 仿真程序
- 文件传送
- 数据库事务处理
- 网络管理
- 集中数据服务

TP 通过逻辑单元 (LU) 访问网络, 逻辑单元建立并维护与另一节点上的伙伴 LU 的会话。有关逻辑单元的更多信息, 请参阅『逻辑单元』。

注: Communications Server for Linux 提供了大多数受支持的 API 的样本 TP。有关样本 TP 的更多信息, 请参阅 API 的程序员指南。您也可以作为其他产品的一部分购买 SNA TP 或创建自己的 TP (请参阅『应用程序编程接口』)。

应用程序编程接口

SNA TP 是使用应用程序编程接口 (API) 编写的。API 提供了使 SNA TP 能够访问 SNA 功能的特定子例程, 如那些用于交换数据和执行控制功能的子例程。这些子例程使 SNA TP 能够与远程节点上的另一个 SNA TP 通信。

在所有平台上, Communications Server for Linux 都提供下列 API:

- APPC - 仅 6.2 型 LU
- CPI-C (用于通信的公共编程接口) - 仅 6.2 型 LU
- CSV (公共服务动词) API
- LUA API

另外, Communications Server for Linux 还包括以下专用编程接口:

- MS (管理服务) API (仅用于 AIX 或 Linux 系统)
- NOF (节点操作员设施) API

网络可访问单元

TP 和 SNA 网络之间的通信是通过网络可访问单元 (即 NAU, 以前被称为“网络可寻址单元”) 进行的, NAU 是可以由其他网络资源访问 (通过唯一的本地地址进行访问) 的唯一网络资源。

SNA 提供了下列类型的 NAU:

- 物理部件 (请参阅『物理部件』)
- 逻辑单元 (请参阅『逻辑单元』)
- 控制点 (请参阅第 7 页的『控制点』)

注: 由于 TP 被认为是网络用户, 而不是组件, 因此未将它们归类为 NAU。

物理部件

每个 SNA 节点都包含物理部件 (PU)。PU 管理资源 (如链路资源) 并支持与主机进行通信。

注: 在 2.1 型节点 (它们可以是 APPN 节点) 中, 控制点除了提供其他服务外还提供 PU 服务 (请参阅第 7 页的『控制点』)。两个 2.1 型节点 (如 Communications Server for Linux 节点) 可以直接通信, 而不需要主机的服务来建立通信。

逻辑单元

每个 SNA 节点都包含一个或多个逻辑单元 (LU)。LU 提供了一组功能, TP 和最终用户使用这些功能来提供对网络的访问。LU 直接与本地 TP 和设备通信。

SNA 定义了几种类型的 LU，每种类型的 LU 都针对特定应用程序类作了优化。不同类型的 LU 不能相互通信，但同类型的 LU 即使驻留在不同类型的系统上也可以相互通信。

例如，只要两个 TP 使用相同的 LU 类型，运行 Linux 系统的 TP 就可以与 AS/400® 计算机上的 TP 通信，就像与另一个 Linux 系统上的 TP 进行通信那样容易。

Communications Server for Linux 支持下列 LU 类型：

LU 6.2 (对于 APPC、5250、APPC 应用程序套件和 CPI-C)

LU 6.2 支持在分布式数据处理环境中进行程序间通信。LU 6.2 数据流或者是 SNA 通用数据流 (GDS，这种一种结构化字段数据流)，或者是用户定义的数据流。LU 6.2 可以用于两个 5 型节点、一个 5 型节点和一个 2.0 型或 2.1 节点或者两个 2.1 型节点之间的通信。(2.1 型节点可以作为 APPN 节点工作。)

与任何其他 LU 类型相比，此类 LU 能够提供更多的功能和更大的灵活性。除非受到现有硬件或软件的约束，否则，在开发新的应用程序时，LU 6.2 是合理的选择。

注：只有 LU 6.2 可以提供独立 LU 功能。

LU 3 (对于 3270 打印)

LU 3 支持使用 SNA 3270 数据流的应用程序和打印机。

例如，LU 3 可以支持在客户信息控制系统 (CICS®) 中运行的应用程序并将数据发送至与“IBM 3174 建立控制器”相连的 IBM 3262 打印机。

LU 2 (对于 3270 显示器)

LU 2 支持在交互环境中使用 SNA 3270 数据流进行通信的应用程序和显示工作站。2 型 LU 还使用 SNA 3270 数据流来进行文件传送。

例如，LU 2 协议可以支持 3270 仿真程序，后者使工作站能够执行 IBM 3270 系列终端的功能。另外，其他程序使用 LU 2 与主机应用程序通信，这些主机应用程序通常向 3270 显示设备提供输出。此类 TP 使工作站能够与主机实现某种形式的协作处理。

LU 1 (对于 SCS 打印和 RJE)

LU 1 支持应用程序和单设备或多设备数据处理工作站在交互式、批处理数据传输或分布式数据处理环境中进行通信。1 型 LU 所使用的数据流符合 SNA 字符串或文档内容体系结构 (DCA)。

例如，1 型 LU 可以支持在信息管理系统 / 虚拟存储 (IMS/VS) 之下运行并与 IBM 8100 信息系统通信的应用程序。这使操作员能够更正应用程序维护的数据库。

使用 LU 1 的应用程序通常被描述成远程作业输入 (RJE) 应用程序。

LU 0 (对于 LUA)

LU 0 是早期的 LU 定义，它支持基本的程序间通信。某些主机数据库系统 (如 IMS/VS (信息管理系统 / 虚拟存储)) 以及零售和银行业的某些销售点系统 (如 IBM 4680 商店系统操作系统) 使用 LU 0。这些产品的当前发行版还支持 LU 6.2 通信，后者是新应用程序的首选协议。

注：有关 SNA 逻辑单元使用的数据流的信息，请参阅 *Systems Network Architecture Technical Reference*。

控制点

控制点 (CP) 是一个 NAU, 它管理其域中的网络资源, 并控制资源激活、释放和状态监视。CP 管理诸如链路之类的物理资源以及诸如网络地址之类的逻辑信息。

SNA 定义了下列类型的网络控制点:

系统服务控制点

在 5 型节点上, CP 被称为系统服务控制点 (SSCP)。它管理并控制子区域网络中的网络资源。例如, SSCP 可以使用网络资源的目录来找到在其控制之下的特定 LU, 并可以在它的域中的两个 LU 之间建立通信。SSCP 还可以与其他 SSCP 合作, 以便在不同子区域域中的 LU 之间建立连接。

SSCP 还为主机系统的网络操作员提供了一个界面, 使操作员能够检查并控制网络中的资源。

物理部件控制点

在子区域网络中的 4 型节点和 2.0 型节点上, 控制点被称为物理部件控制点 (PUCP)。

控制点 在 2.1 型节点上, 控制点提供 PU 功能和 LU 功能, 如激活本地链路站、与本地操作员进行交互作用以及管理本地资源。它还可以提供网络服务, 如伙伴 LU 定位和本地 LU 路由选择。

在子区域网络中, Communications Server for Linux 节点上的 CP 充当 2.0 型 PU。它与主机上的 SSCP 通信, 并且不与子区域网络中的其他 CP 通信。

当 CP 参与 APPN 网络时, 它与相邻节点中的 CP 交换网络控制信息。CP 还可以作为 6.2 型独立 LU 工作。CP 充当本地节点上的 TP 的缺省 LU。有关 APPN 控制点的更多信息, 请参阅第 13 页的『APPN 控制点』。

会话

NAU 通过被称为会话的临时逻辑通信信道与其他节点中的 NAU 通信。在两个 TP 可以通信之前, 它们的 LU 必须建立会话。在本地节点上管理会话的 LU 被称为本地 LU; 在远程节点上管理会话的 LU 被称为伙伴 LU。

会话类型

Communications Server for Linux 主要涉及下列类型的会话:

LU-LU 会话

要使两个 TP 能够通信, 支持这两个 TP 的 LU 必须首先建立 LU-LU 会话。通常, 当一个 SNA 节点中的 TP 尝试与另一节点中的 TP 通信, 但在两个节点上的 LU 之间没有现有的会话可用时, 就会建立会话。

SSCP-LU 会话

从属 LU (请参阅第 8 页的『从属 LU 和独立 LU』) 在可以与子区域网络中的 LU 建立会话之前, 必须与 5 型节点上的 SSCP 建立活动 SSCP-LU 会话。一旦 SSCP-LU 会话处于活动状态, 从属 LU 就可以请求 LU-LU 会话。

SSCP-PU 会话

在可以建立 SSCP-LU 会话之前, 控制 LU 的 PU 必须与 5 型节点上的 SSCP 建立活动 SSCP-PU 会话。SSCP-PU 会话用来在 PU 与 SSCP 之间传递控制数据和网络管理数据。

CP-CP 会话

在 APPN 网络中，相邻节点建立 CP-CP 会话。这些会话用来在 APPN 网络中搜索资源以及用来维护拓扑信息（请参阅第 13 页的『APPN 控制点』）。

会话的逻辑单元属性

逻辑单元具有一些确定它们在 LU-LU 会话期间如何进行交互作用的属性。这些属性由 SNA 体系结构确定。LU 可以是主 LU 或辅助 LU，并可以是附属 LU 或独立 LU。

主 LU 和辅助 LU: 要建立会话，一个 LU 通过将 BIND 请求发送至另一个 LU 来请求激活会话：

- 主 LU 是为给定 LU-LU 会话发送 BIND 请求的 LU。
- 辅助 LU 是接收 BIND 请求的 LU。

同级网络不使用固定节点层次结构，并且没有预先确定的主 LU 或辅助 LU。

注: 在同级网络中，参与多个会话（请参阅『多个会话与并行会话』）的独立 LU 可以在一个会话中作为主 LU 并在另一个会话中作为辅助 LU。

附属 LU 和独立 LU: 所有 0、1、2 和 3 型 LU 都是附属 LU。可以将 6.2 型 LU 配置为附属 LU 或独立 LU。

- 附属 LU（也被称为从属于 SSCP 的 LU）需要 SSCP 的服务才能与另一个 LU 建立会话。必须先建立 SSCP-LU 会话，然后才能建立附属 LU-LU 会话。

附属 LU 只能与 SNA 主机上的 LU 建立会话。由于此限制，附属 LU 通常使用子区域网络（也被称为主机为中介的网络）。然而，附属 LU 请求者（DLUR）功能使来自附属 LU 的会话通信能够通过 APPN 网络流动。有关 DLUR 的更多信息，请参阅第 24 页的『访问 APPN 网络的子区域网络』。

外围节点上的附属 LU 总是辅助 LU。

- 独立 LU 可以与其他独立 LU 建立会话，而不需要借助于 SNA 主机。LU 6.2 是唯一可以作为独立 LU 的 LU 类型。

在建立会话时，独立 LU 可以作为主 LU 或辅助 LU。

多个会话与并行会话

独立 LU 可以同时参与多个远程 LU 的会话（多个会话）。

独立 LU 还可以参与并行会话或与同一远程 LU 的多个并发会话。

附属 LU（包括附属 LU 6.2）不能具有多个会话。

第 9 页的图 2 显示了具有多个会话和并行会话的 LU。LUA 和 LUB 具有并行会话。LUA 还具有多个会话：两个与 LUB 的会话和一个与 LUD 的会话。LUD 具有与 LUA 和 LUC 的多个会话。

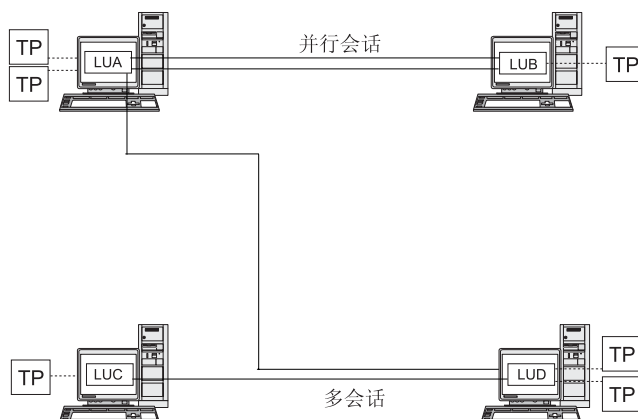


图 2. 多个会话和并行会话

对话

本节仅适用于 LU 6.2。

一旦在两个 LU 间建立了会话，LU-LU 会话就支持在这两个 TP 之间进行信息交换，这两个 TP 对会话进行独占使用以执行事务。这种信息交换被称为对话。每次只能有一个对话可以使用特定的会话，但会话是串行可再用的（许多对话可以一个接一个地使用同一个会话）。

要启动对话，源 TP 将请求发送至其 LU，请求该 LU 分配与远程 TP 的对话。调用 TP（或源 TP）启动对话，就像电话对话的呼叫方一样。可调用 TP 或目标 TP（远程 TP）是对话中的伙伴，就像接收电话呼叫的那一方。

如第 10 页的图 3 所示，在 TP 和 LU 之间交换信息，以允许一个节点与另一个节点通信。虽然 TP 似乎是直接进行通信，但每个节点上的 LU 是每次信息交换的中介。

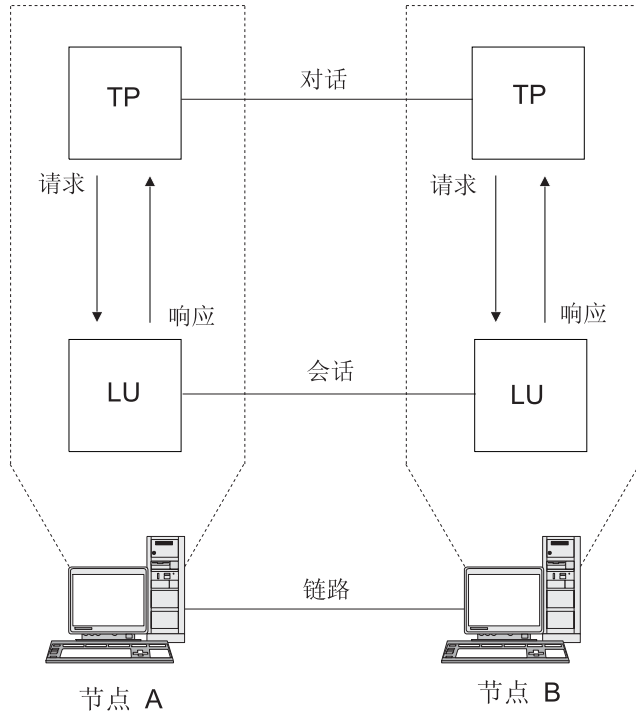


图 3. 事务程序与逻辑单元之间的通信

SNA 定义了两种类型的对话：基本对话和映射对话。这两种对话使用不同的方法来指示要在 Communications Server for Linux 与 TP 之间传递的发送数据包或接收数据包的长度。

- 在基本对话中，在将数据提供给 SEND 函数之前，TP 必须将该数据格式化为逻辑记录。

逻辑记录由两字节或四字节头以及跟随其后的最多 32,765 个字节的数据组成，逻辑记录的头以长度为两个字节的字段（通常表示为“LL”）开始。可以将逻辑记录组合到一起并成块地发送，即通过调用一次 SEND 函数来发送多个逻辑记录。

- 在映射对话中，将信息作为指向单个未格式化的数据块的指针传递给 SEND 函数；块的长度作为另一个参数传递。不能将块作为一个或多个逻辑记录接收；接收 TP 必须执行进行记录级格式化所需要的任何操作。

方式

每个 LU-LU 会话都具有相关联的方式，方式定义了一组会话特征。这些会话特征包括同步参数、会话限制（如两个 LU 之间的最大会话数）、消息大小以及路由参数。

每个方式由唯一方式名标识。方式名在使用该方式的所有 SNA 节点上必须相同。

路由选择

要建立 LU-LU 会话，必须计算两个 LU 所驻留的节点之间的路由。路由是链路和节点的有序序列，它表示两个节点之间的路径。

SNA 网络支持下列路由选择方法：

- 对于子区域网络，必须预定义子区域节点间的所有路由。

- 对于不支持 APPN 的同级网络，2.1 型节点只能支持与相邻节点的会话；不能通过中间节点来路由它们的会话。
- 对于 APPN 网络，SNA 可以使用对会话使用的方式指定的服务等级来在启动会话时动态地计算路由（请参阅『服务等级』）。

APPN 的高性能路由（HPR）功能部件提供了下列功能：

- 快速传输协议（RTP）使路由网络层分组通过会话路由上的中间节点的周期和存储需求最小。
- 自动网络路由（ANR）使 APPN 网络在最初计算得到的路由的某部分发生故障时能够自动重新路由会话。

服务等级

服务等级（COS）是传输网络（数据链路控制和路径控制）特征（如路由安全性、传输优先级和带宽）的定义，本地节点可以使用该定义来建立特定的会话。COS 定义对诸如可接受安全级别、每个字节的成本、每个连接时间的成本、传播延迟以及有效容量等因素指定一些相对值。

在子区域网络中，COS 是从与会话相关联的方式派生的，在主机系统中定义。

APPN 网络节点使用 COS 来计算独立 LU 之间的会话路由。有关 APPN 网络中会话路由的更多信息，请参阅第 17 页的『会话路由』。

基本 APPN 概念

高级对等联网（APPN）是一种支持分布式网络控制的网络体系结构。它使网络易于配置和使用，它提供了集中的网络管理，并支持灵活的连接。

APPN 网络由 2.1 型节点组成。网络中的每个节点由链路连接至 APPN 网络中的至少一个其他节点。CP-CP 会话是通过这些至相邻节点（同一网络中可以建立直接链路而不必通过第三个节点的节点）的链路中的每一条链路建立的。APPN 网络中的所有节点共享一个公共网络名。

APPN 节点可以包含各种大小的处理器，如 Application System/400[®]（AS/400）、运行 CS/NT 的 PC、使用虚拟终端访问方法（VTAM[®]）的系统以及运行 Communications Server for Linux 的 Linux 服务器。

APPN 提供了下列功能：

- 对 APPN 网络节点和端节点以及非 APPN 同级节点的支持（请参阅第 12 页的『APPN 节点类型』）
- APPN 控制点功能（请参阅第 13 页的『APPN 控制点』）
- 支持查找特定逻辑单元的目录服务（请参阅第 14 页的『找到资源』）
- 拓扑和路由服务，它支持使用中间会话路由（ISR）、自动网络路由（ANR）或连接网络（CN）来建立会话（请参阅第 17 页的『会话路由』和第 22 页的『APPN 连接网络』）

注：APPN 节点还可以连接至子区域网络，既作为同级网络中的 APPN 节点又作为子区域网络中的外围节点。

APPN 节点类型

下列类型的节点可以是 APPN 网络的一部分:

- 网络节点 (请参阅『APPN 网络节点』)
- 端节点 (请参阅第 13 页的『APPN 端节点』)

另外, 低入口联网 (LEN) 节点也可以连接至 APPN 网络, 但它们不使用 APPN 功能 (请参阅第 13 页的『低入口联网节点』)。

图 4 显示了一个样本 APPN 网络。

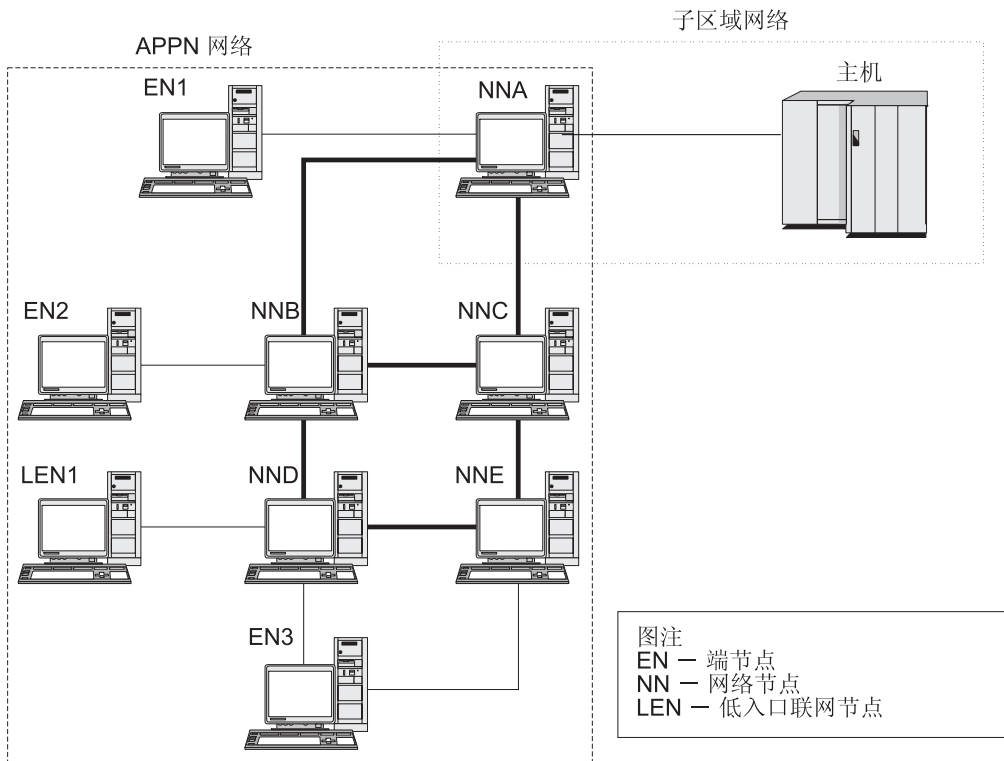


图 4. 样本 APPN 网络的一部分

此示例显示了一个 APPN 网络, 它包含 5 个网络节点 (NN)、3 个端节点 (EN) 和 1 个低入口联网节点。这些网络节点形成 APPN 网络的主干; 端节点通过网络节点来访问网络。任何节点上的 LU 6.2 TP 都可以与网络中的任何其他 LU 6.2 TP 通信。

其中一个 APPN 网络节点 (NNA) 还参与子区域网络, 从而通过通信控制器连接至主机。此节点在与 APPN 网络中的节点通信时作为 APPN 节点工作, 而在与子区域网络中的节点通信时作为外围节点工作。通过此网络节点, APPN 网络中其他节点上的 6.2 型 LU 可以与主机上的 6.2 型 LU 建立 LU-LU 会话。

APPN 网络节点

APPN 网络节点是 2.1 型节点, 它为其域中的所有 LU 提供分布式目录服务和路由服务。这些 LU 本身可以位于网络节点上, 也可以位于网络节点为之提供服务的 APPN 端节点或低入口联网节点上。因为 APPN 网络节点作为它的域中的端节点和低入口联网节点的网络入口点, 所以还将网络节点称为那些节点的网络节点服务器。

网络节点提供下列服务:

- 为它的本地 LU 提供 LU-LU 会话服务
- 为它的域中的所有 LU 提供目录搜索和路由选择服务
- 中间会话路由 (请参阅第 20 页的『中间路由』)
- 接受服务的端节点与 MS 焦点之间的管理服务 (MS) 数据 (如警报) 的路由

APPN 端节点

APPN 端节点是在 APPN 网络中作为端点工作的 2.1 型节点。它只为本地资源维护目录信息。APPN 端节点可以在本地 LU 与相邻节点上的 LU 之间独立地建立会话。对于与未直接连接至端节点的节点上的 LU 的会话, 端节点使用 CP-CP 会话来从它的网络节点服务器请求路由和目录信息。

APPN 端节点可以向其网络节点服务器注册其本地 LU。此能力意味着网络节点服务器的网络操作员不必预定义网络节点为之提供服务的已连接端节点上所有 LU 的名称。

APPN 端节点可以连接至多个网络节点 (请参阅第 12 页的图 4 中的 EN3), 但它每次只能与一个网络节点 (它的网络节点服务器) 具有活动 CP-CP 会话。其他网络节点只能用来为该端节点提供中间路由, 或在主网络节点服务器变为不可用时作为替代网络节点服务器。

APPN 端节点还可以具有至另一 APPN 端节点或低入口联网节点的直接链路, 但从不会在两个端节点间建立 CP-CP 会话。

低入口联网节点

低入口联网节点是使用独立 LU 6.2 协议的 2.1 型节点, 但它不支持 CP-CP 会话。它可以连接至 APPN 网络节点或端节点, 但它不支持 APPN 功能。

APPN 网络节点可以为已连接的低入口联网节点提供路由服务, 从而使低入口联网节点能够参与 APPN 网络, 而不要求在低入口联网节点与 APPN 网络中的所有节点之间定义链路站。

必须对低入口联网节点定义低入口联网节点可能要与之建立会话的 APPN 网络中的 LU, 就好像它们驻留在低入口联网节点的网络节点服务器上一样。低入口联网节点与其网络节点服务器上的 LU 建立会话。网络节点将会话通过 APPN 网络路由至网络中该 LU 实际驻留的节点。必须对为低入口联网节点提供服务的网络节点预定义低入口联网节点上的 LU。低入口联网节点上的 LU 资源 (与端节点上的 LU 资源不同) 不能在网络节点服务器上注册。

APPN 端节点不能提供中间路由。当低入口联网节点的唯一链路是至 APPN 端节点的时候, 该低入口联网节点只能通过两个节点间的直接链路与端节点上的 LU 通信。

APPN 控制点

APPN 控制点是一组功能, 它管理节点资源并支持 2.1 型节点上的物理部件和逻辑单元功能。APPN CP 控制本地节点功能 (如激活与释放适配器和链路)、提供目录和拓扑信息并辅助 LU 启动与终止会话。

APPN 网络中的相邻节点使用一对并行 CP-CP 会话来交换网络信息并提供目录和路由选择服务。给定并行 CP-CP 会话对的两个会话都必须是活动的, 以便伙伴 CP 开始并保持它们的交互作用。不同节点类型以不同方式使用这些会话, 如下所述:

- 在 APPN 网络节点和每个相邻网络节点之间建立两个并行 CP-CP 会话。这些 CP-CP 会话用来交换目录、拓扑和管理服务数据。
- 在 APPN 端节点和作为该端节点的服务器的相邻网络节点之间建立两个并行 CP-CP 会话。这些 CP-CP 会话用来交换目录、拓扑和管理服务数据。
- 低入口联网节点不支持 CP-CP 会话。

CP-CP 会话提供的功能根据所涉及的节点类型的不同而有所变化，如下所述：

- 所有 CP-CP 会话都控制目录搜索。
- 端节点与网络节点之间的 CP-CP 会话提供了下列功能：
 - 注册资源。
 - 在端节点与焦点之间路由管理服务数据（如警报）。
 - 将拓扑数据从每个端节点路由至其网络节点服务器。网络节点服务器可使用此信息来计算不流经该网络节点服务器的路由。
- 相邻网络节点间的 CP-CP 会话交换拓扑信息。作为此交换操作的结果，每个网络节点都创建内部网络拓扑数据库。

设置节点时，必须定义 CP 名称。CP 还是可支持用户会话的 LU，并且它可以是节点中定义的唯一 LU（如果您这样选择的话）。

找到资源

为了支持 TP 之间的通信，Communications Server for Linux 首先在控制那些 TP 的逻辑单元之间建立会话。APPN 使节点上的 CP 能够找到整个 APPN 网络中的 LU，而不要求该节点具有远程 LU 的任何配置信息。在网络中动态找到 LU 的 APPN 功能被称为目录服务。一旦找到了资源，就通过 APPN 网络计算会话的路由。

资源名称

每个节点都具有唯一的名称，该名称由网络名和控制点名称这两部分组成。它们共同组成标准 CP 名称。此名称标识每个节点，从而使其与网络中所有其他节点区别开。同样，每个逻辑单元也由一个标准 LU 名称标识，该标准 LU 名称由网络名和 LU 名组成。

注：有关网络命名约定的更多信息，请参阅《*Communications Server for Linux 快速入门*》。

目录服务

每个 APPN 节点都维护一个网络资源目录。目录服务是节点 CP 的组件，该组件管理局部目录数据库并在网络节点中搜索整个 APPN 网络以找到网络资源。

初始化节点后，节点包含下列信息：

- 节点类型（APPN 网络节点、APPN 端节点或低入口联网节点）
- 节点的网络标识
- 节点的 CP 名称

每个节点目录都维护资源（LU 和 CP）的条目，包括每个资源的标准名称、类型和注册状态。存储在每个本地目录中的特定资源取决于节点类型：

- 低入口联网节点维护包含它自己的 LU 的目录。还必须用低入口联网节点的所有可能伙伴 LU 的目录条目来配置低入口联网节点。必须对低入口联网节点定义低入口联网

节点可能要与之建立会话的 APPN 网络中的 LU，就好像它们驻留在低入口联网节点的网络节点服务器上一样。低入口联网节点与其网络节点服务器上的 LU 建立会话。网络节点将该会话通过 APPN 网络路由至网络中的适当节点。

低入口联网节点也可以在目录条目中使用通配符来指定可以在特定链路上访问的多个伙伴 LU。

- APPN 端节点维护包含它自己的 LU 的目录。还可以将其配置为存储相邻节点中的伙伴 LU 的目录条目。这使本地 LU 能够建立与那些 LU 的对等会话，而不必使用 APPN 功能。

如果未在本地对端节点定义资源，或者端节点当前无法访问该资源，则端节点发送一个请求给它的网络节点服务器，要求它在 APPN 网络中搜索该资源。

- APPN 网络节点维护一个目录，该目录包含 APPN 网络节点自己的 LU 以及它的域中的端节点和低入口联网节点 LU。端节点可以向其网络节点服务器动态注册其 LU。（低入口联网节点不能向网络节点服务器注册 LU，因此必须在其网络节点服务器中配置其 LU。）网络节点目录还可以包含不在网络节点域中但已通过先前的搜索确定了其位置的 LU 的高速缓存条目。

网络节点以两种方式为其他节点提供目录服务：

- 搜索远程资源以响应来自端节点或低入口联网节点的会话请求
- 当在本地目录中找到指定的资源时，以肯定方式响应来自其他网络节点的目录搜索请求

低入口联网节点目录： 图 5 中显示了低入口联网节点目录的示例。因为低入口联网节点不支持 CP-CP 会话，所以节点 LEN1 的目录必须包含与它通信的所有 LU。节点 LEN1 目录将其网络节点服务器（NNA）标识为不在相邻同级端节点上的任何 LU 的位置。因为节点 LEN1 只能通过节点 NNA 访问 LU，所以它将网络节点上的 CP 定义为所有 LU（包括位于端节点上的 LU）的“拥有 CP”。

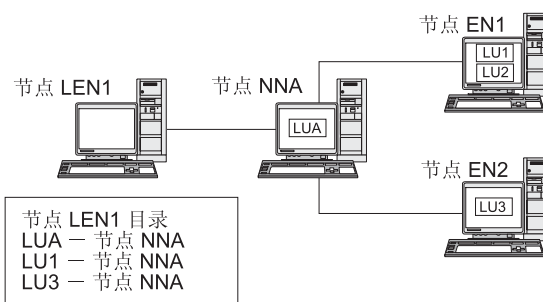


图 5. 低入口联网节点目录

要建立与未直接连接的节点上的 LU 的会话，节点 LEN1 发送一个 LU-LU 会话激活（BIND）请求至其网络节点服务器（节点 NNA）。服务器自动找到目标 LU 并转发该 BIND。

注： 在此示例中，节点 LEN1 可以通过其网络节点服务器 NNA 建立与节点 EN1 上 LU1 的会话。然而，节点 EN1 上的 LU2 未在节点 LEN1 的目录中定义，因此节点 LEN1 不能建立与该 LU 的会话。

端节点目录: 当 LU 未在端节点目录中出现时, 端节点启动 LOCATE 搜索来查找期望的 LU。要激活对远程 LU 的搜索, 端节点调用其网络节点服务器的服务。图 6 显示了端节点目录的示例。

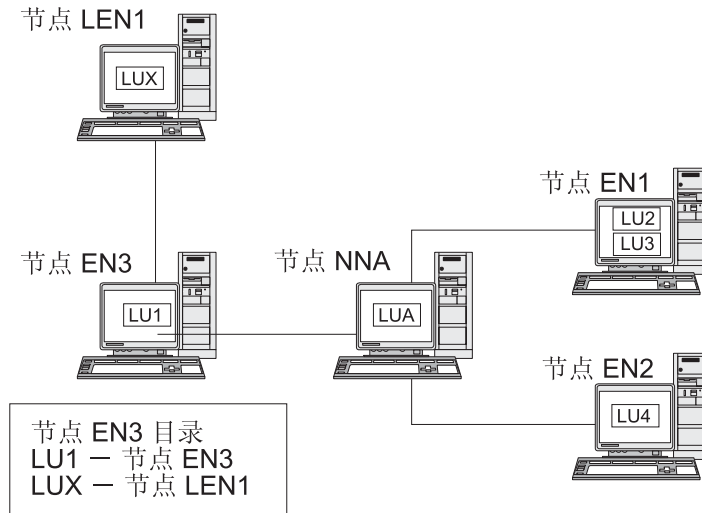


图 6. 端节点目录

不需要对端节点定义 APPN 网络中的潜在伙伴 LU。然而, 为了使节点 EN3 建立与节点 LEN1 上 LUX 的会话, 必须将在低入口联网节点上的 LU 配置为节点 EN3 上的伙伴 LU。

网络节点目录: 网络节点为其所服务的端节点提供分布式目录服务。

图 7 中显示了网络节点目录的示例。

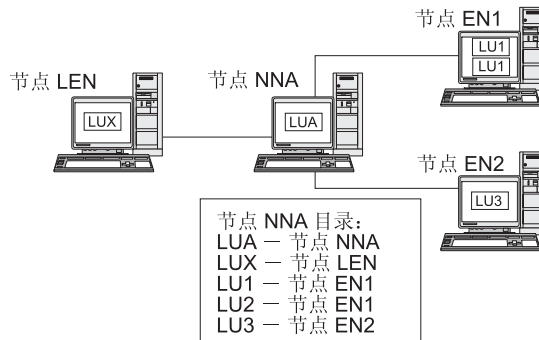


图 7. 网络节点目录

网络节点按如下所述找到远程 LU:

1. 网络节点接收要求找到 LU 的请求。请求可以是下列任何一项:
 - 由端节点或低入口联网节点发送至其网络节点服务器的目标 LU 的名称
 - 来自端节点的 LOCATE 搜索请求中指定的 LU 名
 - 来自低入口联网节点的 BIND 请求中指定的 LU 名
 - 由网络节点上的 TP 指定的 LU 名

2. 如果目标 LU 不在网络节点上（而是出现在它的目录中），则网络节点发送定向搜索请求至目标网络节点服务器以验证该 LU 的位置。

如果 LU 不在网络节点目录中，则节点通过发送一个广播搜索至每个相邻网络节点来启动对网络的搜索。

3. 每个节点依次传播广播并返回指示成功或失败的应答。

为了将来的需要，网络节点高速缓存从成功的广播搜索中获得的信息。

APPN 端节点还可以接收（或响应）来自其网络节点服务器的 LOCATE 搜索请求以搜索端节点中的特定 LU，或确认其依旧存在。

每个 APPN 端节点通过向网络节点发送注册消息来向其网络节点服务器注册其 LU。网络节点以这种方式维护其域中端节点的当前目录信息。低入口联网节点不能向其网络节点服务器注册 LU。因此，必须通过配置对网络节点服务器预定义低入口联网节点上的所有 LU。

会话路由

APPN 支持下列动态路由选择过程：

- 对于与相邻节点的会话，为直接会话路由。
- 对于通过一个或多个中间节点的会话，为下列其中一项：
 - 中间会话路由（ISR），它提供在会话过程中不会更改的路由。
 - 高性能路由（HPR），包括快速传输协议（RTP）和自动网络路由（ANR）设施。RTP 使通过会话路由上的中间节点路由网络层分组的周期和存储需求最小，ANR 则允许您在路由故障或拥挤处重新路由会话通信。

提供动态路由选择的 APPN 功能称为拓扑和路由服务（TRS）。

拓扑和路由服务

每个 APPN 节点都包括一个拓扑数据库，该数据库存储关于其他 APPN 节点和关于传输组的信息，传输组是一对特定节点间的链路组。特定节点数据库的内容取决于节点类型：

- 所有网络节点共享网络拓扑数据库的一个副本。此共享数据库包含关于所有其他网络节点的信息（包括网络标识、CP 名称和其他节点特征）以及关于每对网络节点之间的传输组的信息。此数据库提供网络主干拓扑（可用于网络中任何节点对之间路由会话的节点和传输组）的完整视图。

另外，每个网络节点上的拓扑数据库包含关于从该网络节点至相邻端节点或低入口联网节点的传输组的本地信息。

网络节点使用拓扑数据库来计算其域中 LU 与远程 LU 之间会话的路由，或为其他网络节点提供信息以使它们能计算会话路由。

- 每个端节点具有一个局部拓扑数据库，该数据库具有关于从该端节点至相邻节点的传输组的信息。

端节点向其网络节点服务器提供此信息作为请求的一部分，以找到 LU 并计算至该 LU 的会话路由。网络节点服务器在计算端节点的会话路由时使用端节点的拓扑信息。端节点在建立与相邻节点上预定义 LU 的会话时使用此信息。端节点拓扑数据库只支持与相邻节点的通信。

注:

1. APPN 网络节点和端节点还维护关于至连接网络的链路的拓扑信息（请参阅第 22 页的『APPN 连接网络』）。
2. 低入口联网节点维护本地拓扑信息。它们不将此信息转发至网络节点服务器。

如第 19 页的图 8 中所示，网络拓扑信息在所有网络节点中复制，本地拓扑信息存储在网络节点和端节点中。

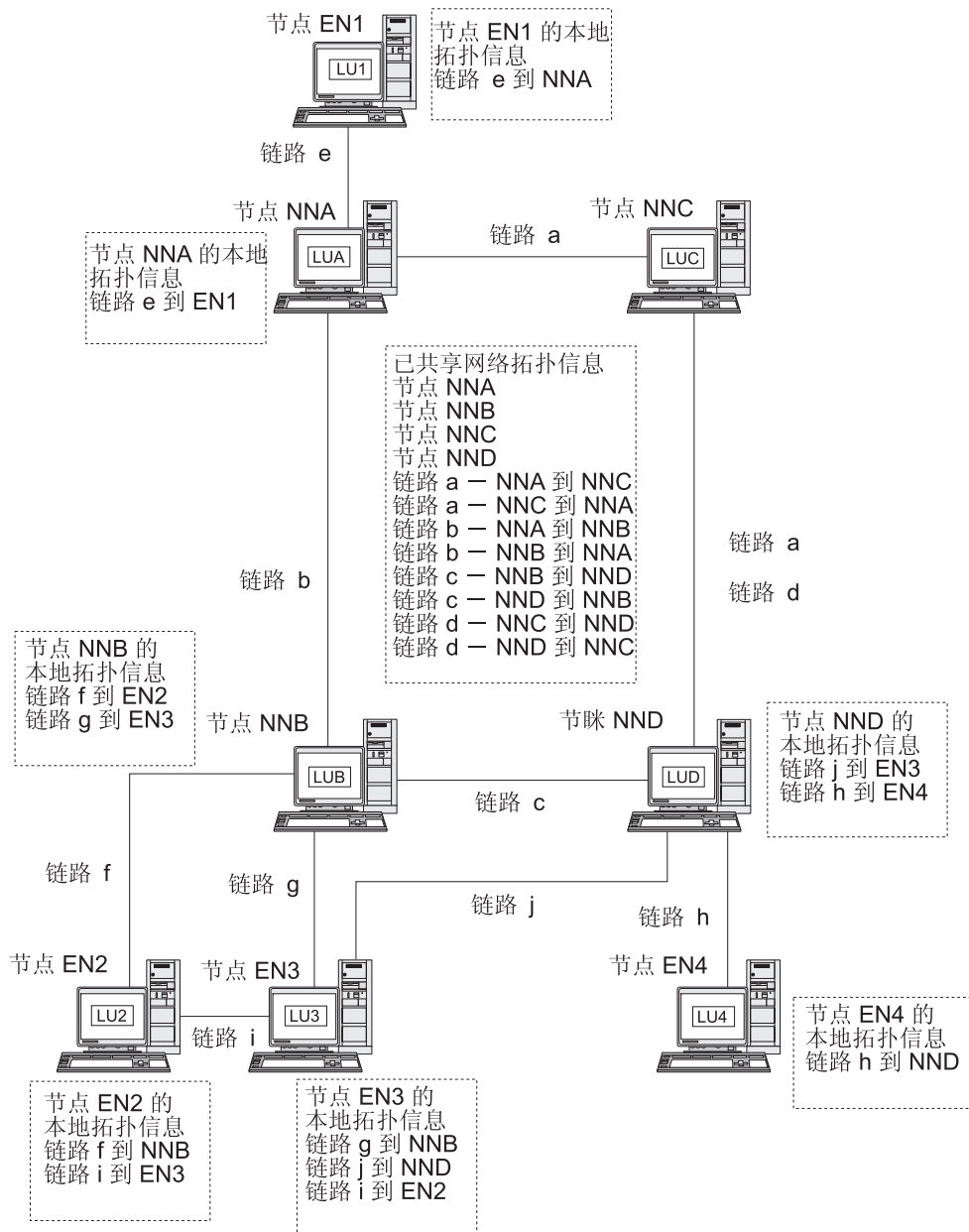


图 8. 网络节点中的网络拓扑数据库

节点 NNA、NNB、NNC 和 NND 中复制了共享网络拓扑数据库。另外，这几个节点都包括了本地拓扑信息（节点 NNC 除外，因为它不具有任何至端节点的链路，所以没有任何本地拓扑信息）。例如，节点 NNB 包括关于至节点 EN2 的链路 f 以及至节点 EN3 的链路 g 的信息，但它不包括关于链路 i 的信息，链路 i 将节点 EN2 和 EN3 连接在一起。

端节点只包括至相邻节点的链路的信息。例如，节点 EN2 包括关于至节点 NNB 的链路 f 和至节点 EN3 的链路 i 的信息。

拓扑数据库更新: APPN 网络节点使用 CP-CP 会话来在资源（如节点或两个网络节点间的链路）被激活或释放时或在现有资源的特征发生更改时交换网络拓扑信息。当这样的更改发生时，网络节点生成拓扑数据库更新（TDU），它包含节点标识、节点

和链路特征以及标识要更新的资源和资源更改的更新序号。每个 TDU 被发送至所有活动网络节点以确保网络拓扑数据库在整个网络中保持最新。

APPN 网络中的路由选择: APPN 目录服务找到特定的会话伙伴; 在网络中找到会话伙伴后拓扑和路由服务计算最佳会话路由。每个网络节点为它自己的 LU 以及它所服务的端节点或低入口联网节点上的 LU 所发起的会话提供路由选择服务。网络节点使用它自己的本地拓扑信息以及来自共享网络拓扑数据库的信息来动态计算节点间的路由。

一旦找到了会话伙伴, 网络节点就执行下列步骤来选择路由:

1. 获取会话路由所需要的特征。

请求会话的 LU 指定标识会话特征的方式名。关联的方式标识服务等级, 该服务等级指定用来路由会话通信的链路的需求。

2. 获取可能路由的所有传输组和网络节点:

- 如果会话请求来自端节点, 则端节点提供关于它所具有的至其网络节点服务器的链路和至连接网络的链路 (如果存在的话) 的信息。
- 如果会话伙伴不在相邻节点上, 则请求会话的 LU 的网络节点服务器使用网络拓扑数据库来标识至会话伙伴的路由中的网络节点和中间传输组。
- 如果会话伙伴在端节点上, 则该端节点 (或其网络节点服务器) 提供关于网络节点服务器与端节点之间链路 (或端节点与连接网络之间的链路) 的信息。

3. 排除不满足指定的会话路由特征的所有网络节点和传输组。

4. 计算会话的最佳路由。

根据指定的服务等级, 路由计算算法计算每个节点和逻辑链路的权重值, 然后总计每条路由的权重。为了选择最佳路径, 网络节点计算从包含源 LU 的节点到包含目标 LU 的节点的当前最小加权路径。

中间路由

中间路由允许 APPN 网络节点接收并路由送往另一节点的数据。数据的原点和目标可以是端节点、另一网络节点或 LEN。

中间路由支持不在相邻节点上的 LU 间的会话。在已为会话选择了路由之后, 路由中的 APPN 网络节点使用中间路由将会话数据转发至路由中的下一个节点。

由拓扑数据库维护的资源特征可以包括拥挤状态。如果某个网络节点变得过于拥挤, 则该网络节点可以将此信息传递至网络中的其他网络节点, 使得拥挤的网络节点不太可能包括在为新会话计算的会话路由中。

APPN 提供两种类型的中间路由:

- 在所有网络节点中可用的中间会话路由 (ISR) 中, 网络节点保持对每个中间会话的跟踪。每个中间节点调整会话数据的调步以控制数据在相邻节点间流动的速率。每个中间节点还可以对已分段数据执行分段和组装。在 ISR 中, 一旦建立会话路由, 该会话中的所有数据就使用该路由。如果部分路由失效, 则该会话结束。
- 在支持 APPN 高性能路由 (HPR) 功能的网络节点中可用的自动网络路由 (ANR) 中, 如果部分路由失效, 则中间网络节点可以动态重新路由会话通信。ANR 未提供中间会话调步或分段和组装。

ANR 允许中间节点路由会话通信的速度比传统 APPN ISR 可能的速度快得多。然而，ANR 在 RTP（快速传输协议）端点处需要更多的开销。在中间节点不多的路由中，由于端点的处理时间，ANR 路由实际上可能比 ISR 路由慢。对于包含大量中间节点（中继段）的路由，ANR 路由一般会更快。不分胜负点的确切位置取决于 RTP 节点的效率。

直接连接

直接连接允许会话通信直接在两个节点间传播，而不需要 APPN 网络节点来路由会话。一般而言，直接连接的节点间的会话可以比通过网络节点路由数据的会话更快地交换数据。对于共享访问传输设施（SATF）上的节点（例如，图 9 中显示的令牌环上的节点），通过在网络中每对节点间定义链路，效率将会得到提高。然而，这可能是一个困难的任務 - 链路站的数目是 $n \times (n-1)$ ，其中 n 是网络中的节点数。

令牌环上的 APPN 网络显示在图 9 中。

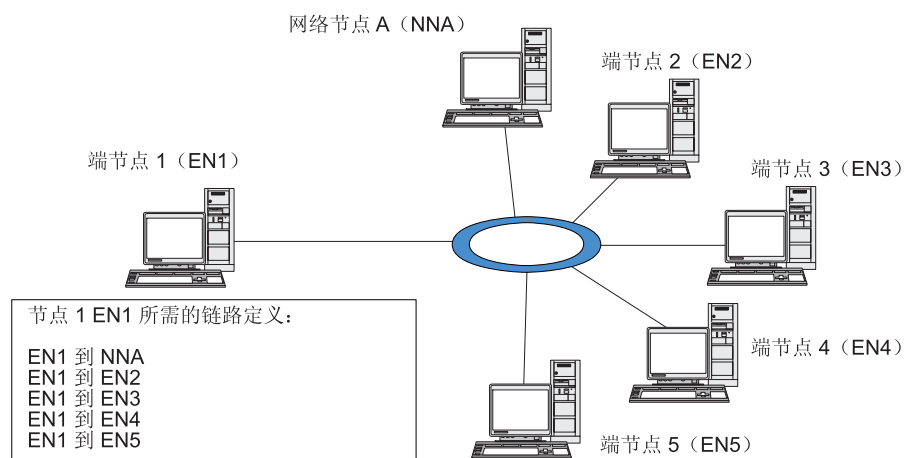


图 9. 使用共享访问传输设施的 APPN 网络

如果节点 EN1 对于网络中的每条链路都有链路定义，则它可以建立至任何节点的直接链路。支持节点 EN1 与 APPN 网络中每个其他节点之间的直接链路所需要的链路定义显示在第 22 页的图 10 中。对于一个包含 5 个其他节点的网络，节点 EN1 需要 5 个链路定义：

- EN1 至 NNA
- EN1 至 EN2
- EN1 至 EN3
- EN1 至 EN4
- EN1 至 EN5

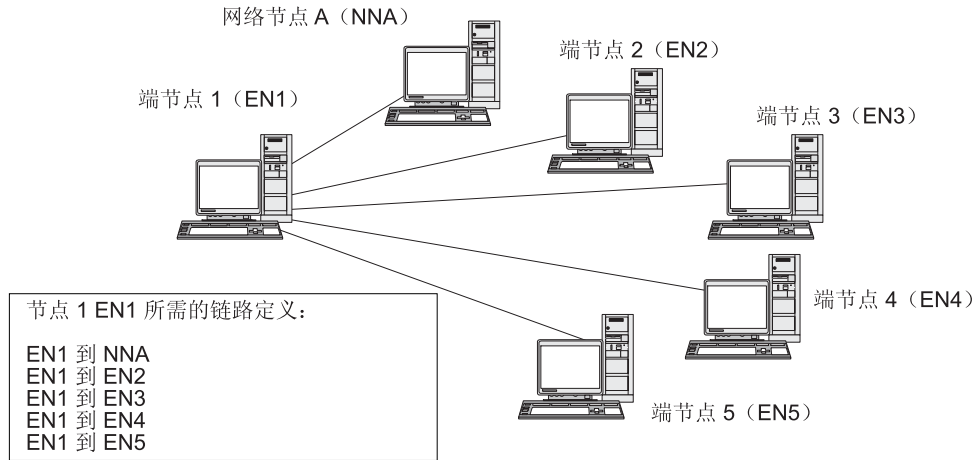


图 10. 从节点 EN1 至 APPN 网络中的每个节点的直接链路所需要的定义

如果网络中的所有节点都要支持至每个其他节点的直接链路，则在此示例中，6 个节点上总共需要 30 个链路定义。一般而言，链路定义的数量可以这样计算： $n \times (n-1)$ ，其中 n 是网络中的节点数。在一个大型网络中，链路定义的数目很快变得庞大。增加网络节点间链路定义的数目还会增加流经网络的 TDU 的数目，这会降低网络的性能。

APPN 连接网络提供了对此问题的解决方案。

APPN 连接网络

对于连接至共享访问传输设施 (SATF) 的 APPN 网络，APPN 连接网络极大地减少了支持网络中节点间直接连接所需要的链路定义的数目。在连接网络中，APPN 端节点只需要配置至相邻网络节点服务器的单个链路以及至连接网络的链路，而不需要配置至每个节点的每条可能的链路。

要使用连接网络功能部件，APPN 网络必须满足下列条件：

- APPN 网络中的节点必须使用诸如令牌环或以太网之类的交换介质来链接。
- APPN 连接网络中的所有链路必须使用相同的介质。
- 必须充分连接包含连接网络的 APPN 网络。在一个充分连接的网络中，每个节点至少具有一条支持 CP-CP 会话的至相邻节点的链路。

在一个连接网络中，SATF 充当直接连接至连接网络中每个节点的虚路由节点 (VRN)。连接网络的名称用作 VRN 的控制点的名称。VRN 支持连接网络中任何两个节点间会话数据的直接路由，但不与其他节点建立 CP-CP 会话，且不会生成 TDU。连接网络中的每个节点只需要一条至其网络节点服务器的链路。

第 23 页的图 11 中显示了使用连接网络时所需要的链路定义。通过使用虚拟节点，连接网络支持“节点 EN1”和 APPN 网络中每个其他节点间的直接链路，而仅需要两个链路定义。

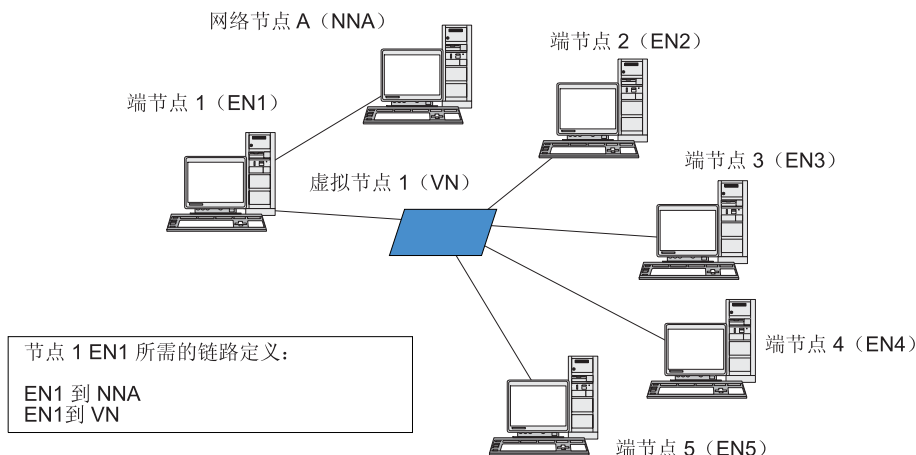


图 11. 使用虚拟节点的直接链路所需要的定义

要支持 APPN 网络中任何两个端节点间的直接链路，总共需要 10 个链路定义。（每个端节点需要两个链路定义：一个至网络节点服务器，一个至虚拟节点。）与不使用连接网络的 APPN 网络的直接连接的需求相比（请参阅第 22 页的图 10），您可以有少得多的链路定义数量（此示例中是 10 而不是 30）。在一个大型网络中，定义需求的差别变得更加显著。

连接网络中两个节点上的 LU 之间的会话按如下方式建立：

1. 每个端节点首先与其网络节点服务器建立 CP-CP 会话。（如果两个端节点具有不同的网络节点服务器，则那些网络节点必须有支持 CP-CP 会话的链路。）
2. 端节点还将其 VRN 链路和本地地址信息报告给网络节点服务器。本地地址信息可以是服务访问点（SAP）地址和介质访问控制（MAC）地址。
3. 服务器通常选择两个端节点间的直接链路作为 LU-LU 会话的最佳路由。它为具有主 LU 的节点提供了该节点建立至具有伙伴 LU 的节点的动态链路所需要的信息。
4. 端节点然后可以建立 LU-LU 会话而不需要中间会话路由。

分支扩展程序

如前面章节所述，APPN 网络中的网络节点需要维护拓扑信息（关于网络中其他节点的位置以及这些节点之间的通信链路的信息）并在拓扑发生更改时将此信息转发至网络。当网络的规模扩大时，存储的信息量和有关拓扑的网络流量可能变得非常大而难于管理。

可以通过将该网络分成多个子网来避免这些问题，以便每个节点只需要维护关于它自己的子网中的节点的拓扑信息。然而，当尝试查找其他子网中的资源时，这会导致增加网络流量。

APPN 的分支扩展程序功能部件（在第 24 页的图 12 中进行了说明）提供了这些问题的解决方案。

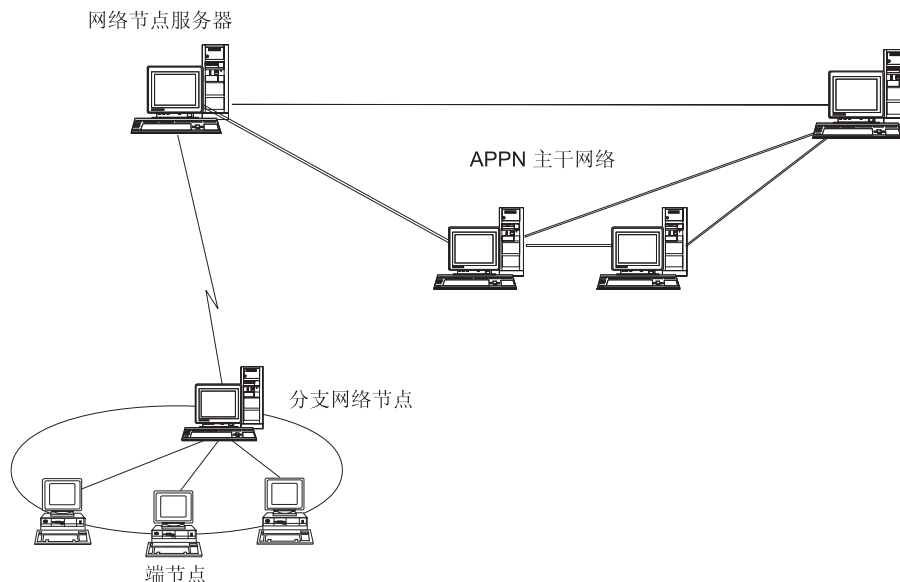


图 12. 分支扩展程序

顾名思义，分支扩展程序是为这样的网络设计的，这些网络可以分成不同区域，如大型组织的独立分支。它的功能是将分支从 APPN 主干网络（例如，组织总部所在地的网络）分离出来。

每个分支包含一个新型节点，称为“分支网络节点”（BrNN），该节点被连接到 APPN 主干网络中的“网络节点”。BrNN 组合了 APPN 网络节点和 APPN 端节点的功能。

- 对于主干网络，BrNN 表现为主干网络中连接到其“网络节点服务器”（NNS）的“端节点”：
 - 主干网络中的节点不知道分支中的节点，从而减少了必须存储的拓扑信息量。
 - 因为 BrNN 表现为“端节点”，所以它不从主干网络中接收拓扑信息（拓扑信息只在“网络节点”之间传送）。
 - BrNN 向其 NNS 注册分支中的所有资源，好像这些资源位于 BrNN 本身。这意味着主干网络中的节点可以找到分支中的资源，而不必知道分支中的各个节点。
- 对于分支网络，BrNN 表现为“网络节点”，充当分支中“端节点”的 NNS。分支中的每个节点将网络中其余节点看成是通过其 NNS 进行连接的，其方式与用于标准 NNS 的方式相同。

访问 APPN 网络的子区域网络

尽管 APPN 网络不需要主机来控制网络中的资源，但主机经常参与 APPN 网络。APPN 已在许多主机平台中实现，它允许主机充当 APPN 网络中的网络节点，同时仍提供 SSCP 来控制任何旧的子区域 SNA 功能。

许多 SNA 网络既包含子区域 SNA 的元素又包含 APPN 的元素。网络的主干是从必须桥接从属 LU 和主机上设施之间的间隔的网络节点构建的。要达到此目的需要另外两个服务：

- 主机上的从属 LU 服务器（DLUS）提供对旧的 SSCP 功能的访问权和与 APPN 网络的接口。

- 网络节点或端节点上的从属 LU 请求者 (DLUR) 提供了一种将会话通信通过 APPN 网络自从属 LU 传输至主机的方法。此功能允许从属 LU 会话利用由 APPN 提供的更通用的路由功能。

DLUR 与 DLUS 的这种组合 (通常简称为 DLUR) 允许在 APPN 主干上传输从属 LU 数据传输。可以不加修改地保留使用从属 LU 的现有 SNA 应用程序, 同时还利用 APPN 的网络管理、动态资源定位以及路由选择能力。DLUR 以此方式提供从子区域 SNA 至 APPN 的有用迁移路径。

从属 LU 不需要驻留在提供 DLUR 功能的节点上。如果 DLUR 功能由网络节点提供, 则从属 LU 可以在相邻网络节点、端节点或低入口联网节点上。如果 DLUR 功能由端节点提供, 则从属 LU 必须在该端节点本身上。

第 2 章 管理 Communications Server for Linux

要了解有关 Communications Server for Linux 管理以及提供的不同管理工具的概述，请参阅『Communications Server for Linux 管理概述』。

管理 Communications Server for Linux 的第一个步骤是配置节点及其资源。首先按第 33 页的『规划 Communications Server for Linux 配置』所述进行配置规划。

在可以配置 Communications Server for Linux 之前，必须按第 33 页的『启用与禁用本地系统上的 Communications Server for Linux』所述启用 Communications Server for Linux 软件。

当启用 Communications Server for Linux 时，可以运行 Motif 管理程序（请参阅第 36 页的『使用 Motif 管理程序』）。Motif 管理程序指导您完成支持使用 Communications Server for Linux 进行的 SNA 通信所需要的配置。Motif 管理程序是推荐的管理工具，因为它使您需要提供的配置信息最少，并指导您完成支持不同类型的通信（如 3270 或 APPC 通信）所必须执行的每个步骤。

另外，可以按第 45 页的『使用命令行管理程序』所述使用命令行管理程序。

对于每个管理任务，本指南提供可用于 Motif 或命令行管理的信息。其他配置方法在第 28 页的『管理工具』中讨论。

Communications Server for Linux 管理概述

作为 Communications Server for Linux 管理员，您负责安装 Communications Server for Linux 软件并负责管理其资源。在开始管理 Communications Server for Linux 之前，您必须了解 Communications Server for Linux 产品的主要功能部件（请参阅《Communications Server for Linux 快速入门》）。本节描述您必须执行的管理任务以及您可以用来执行这些任务的工具。

管理责任

要管理 Communications Server for Linux 系统，需要执行下列任务：

1. 根据将要运行的用户程序的需要，定义 Communications Server for Linux 系统的资源。与 Communications Server for Linux 通信的主机或对等计算机的管理员一起工作，以确保 Communications Server for Linux 配置与远程系统的配置匹配。
2. 初始化 Communications Server for Linux 软件。
3. 在您的需求更改时任意地动态修改配置 - 通过添加或除去资源或通过激活或释放已定义的资源。
4. 监视活动资源的状态并收集诊断信息以诊断发生的任何问题。
5. 任意地创建应用程序或 shell 脚本以使标准管理操作自动化。

这些任务通常由安装了 Communications Server for Linux 系统的站点的系统管理员执行。然而，Communications Server for Linux 还提供了服务点命令设施（SPCF），该设施允许操作员使用 NetView 程序来通过在 NetView 控制台上发出管理命令以远程方式执行

步骤 2 和 3。有关 SPCF 的更多信息，请参阅第 101 页的第 9 章，『从 NetView 中管理 Communications Server for Linux』。

管理工具

Communications Server for Linux 提供了一系列用于管理系统的工具。根据您的需求的不同，可能并不需要使用所有工具。本节概述了这些工具中的每个工具所提供的功能。

注:

1. 本文档提供关于 Communications Server for Linux 管理的一般信息，您可以使用本节描述的任何工具来执行此管理工作。对于大多数场合，建议使用 Motif 管理程序，这是因为它为节点配置与管理提供了上下文相关指导。
2. 有关控制哪些用户可以使用 Communications Server for Linux 管理工具以及他们可以使用的管理功能范围的信息，请参阅第 32 页的『管理许可权』。

Communications Server for Linux 包含下列管理工具:

- Motif 管理程序（请参阅『Motif 管理程序』）。
- 命令行管理程序（请参阅第 29 页的『命令行管理程序』，或者参阅 *Communications Server for Linux Administration Command Reference*）。
- 服务点命令设施（请参阅第 30 页的『远程命令设施』）。
- 配置文件（请参阅第 30 页的『配置文件』）。
- 诊断工具（请参阅第 31 页的『诊断工具』）。

所有 Communications Server for Linux 管理工具都使用 NOF API。您也可以使用该 API 来编写自己的管理工具。有关更多信息，请参阅第 31 页的『NOF 应用程序』。

Motif 管理程序

定义和修改 Communications Server for Linux 配置的最容易的方法是使用 Motif 管理程序 (**xsnaadmin**)。此程序提供了一个图形用户界面，可通过该界面查看和管理 Communications Server for Linux 资源。

可以执行下列管理操作:

- 定义 Communications Server for Linux 资源
- 启动与停止节点及其连接资源
- 更改已定义的资源配置
- 查询已定义的资源配置及其当前状态（如果这些资源处于活动状态的话）
- 删除资源

Motif 管理程序可用来管理节点资源（适用于 LAN 中的任何服务器，只要在该服务器上运行 Communications Server for Linux 软件）和域资源。对于每种类型的通信（如 3270 或 APPC），该程序将指导您设置必需资源的配置。

注: Motif 管理程序中的窗口和对话框可能与本指南中显示的有所不同，这取决于您在特定对话框上所作的选择。

Motif 管理程序包括一些帮助屏幕，这些屏幕提供关于 SNA 和 Communications Server for Linux 的概述信息、关于 Communications Server for Linux 对话框的参考信息以及关于执行特定任务的指导。

在启动 Motif 管理程序之前，确保已启用 Communications Server for Linux 软件（有关更多信息，请参阅第 27 页的第 2 章，『管理 Communications Server for Linux』）。如同任何 X/Motif 应用程序一样，您可能需要设置 DISPLAY 环境变量来指示合适的 X 服务器。

要在后台启动 Motif 管理程序，发出下列命令：

```
xsnaadmin &
```

所有已启动的 Communications Server for Linux 服务器都显示在主屏幕中。对于那些已配置的服务器，程序允许选择节点，接着显示所选节点的配置。否则，程序将提示您选择节点并指导您完成定义该节点所需的步骤。

有关如何使用 Motif 管理程序来定义和管理 Communications Server for Linux 资源的更多信息，请参阅第 36 页的『调用 Motif 管理程序』或者参阅程序提供的帮助屏幕。

注： Motif 管理程序使您能够为标准 Communications Server for Linux 配置设置所有必需的参数。对于高级参数，Motif 管理程序提供缺省值。您只需要提供必需的配置信息，这允许您快速方便地设置 SNA 通信。

其他 Communications Server for Linux 管理工具（包括命令行配置和 NOF 应用程序）提供了对大量配置参数和选项的访问，那些配置参数和选项比 Motif 管理程序中显示的配置参数和选项的范围更广。但是，由于 Motif 管理程序显示配置所需要的关键字段，而隐藏大多数用户应该不需要修改的字段，所以，在大多数情况下，可以从 Motif 管理程序中执行所需的所有配置。因为 Motif 程序可以根据正在执行的配置任务的上下文来更明智地选择值，所以命令行配置提供的缺省值可能与 Motif 管理程序提供的缺省值不同。

如果需要使用其他的这些功能，仍可以使用 Motif 管理程序来设置基本的配置，并使用其他管理工具来指定其他功能。当以后使用 Motif 管理程序来管理经过修改的配置时，尽管您配置的其他功能未显示在 Motif 程序中，但该程序将保留您使用其他工具所作的更改。

命令行管理程序

命令行管理程序 **snaadmin** 使您能够通过发出命令来管理各个 Communications Server for Linux 资源。您可以直接从 Linux 命令提示符使用 **snaadmin**，也可以从 shell 脚本中使用它。

可以对特定 Communications Server for Linux 节点发出命令来管理该节点的资源，可以对 SNA 网络数据文件发出命令来管理主控服务器和备份服务器，也可以对域配置文件发出命令来管理域资源。

一些管理命令（Communications Server for Linux Administration Command Reference 对那些管理命令作了定义）显式地指出它们可以从 IBM Remote API Client 中发出。对于所有其他命令，如果指定带有服务器名的 **-n**，则只能从客户机发出命令行管理命令。从客户机使用 **-n** 发出的命令与从指定的服务器发出该命令效果相同。

可以使用下列任何命令来获取有关命令行管理的帮助:

- **snaadmin -h** 提供有关命令行管理的基本帮助和关于命令行帮助的用法信息。
- **snaadmin -h -d** 提供可以供给 **snaadmin** 程序的命令的列表。
- **snaadmin -h command** 提供对指定的 *command* 的帮助。
- **snaadmin -h -d command** 提供对指定的 *command* 的详细帮助, 包括可随该命令指定的配置参数的列表。

有关更多信息, 请参阅 *Communications Server for Linux Administration Command Reference*。

远程命令设施

远程命令设施 (RCF) 提供了下列设施来支持从主机上的 NetView 控制台对 Communications Server for Linux 进行管理:

- 服务点命令设施 (SPCF) 允许主机 NetView 控制台上的操作员通过发出 Communications Server for Linux 管理命令来从 NetView 中管理 Communications Server for Linux。
- UNIX® 命令设施 (UCF) 使 NetView 操作员能够在 Communications Server for Linux 计算机上发出标准 Linux 命令。

有关 RCF 的更多信息, 请参阅第 101 页的第 9 章, 『从 NetView 中管理 Communications Server for Linux』。

配置文件

Communications Server for Linux 系统的配置信息保存在下列文本文件中:

节点配置文件

/etc/opt/ibm/sna/sna_node.cfg 文件包含关于特定节点的 Communications Server for Linux 节点资源的信息。此文件驻留在运行节点的计算机上。它包含关于节点资源的信息, 并指定在节点上启动 Communications Server for Linux 时哪些资源处于活动状态。

此文件提供可用资源的初始定义; 接着, 可以使用其他管理工具来在您的需求更改时修改正在运行的节点的资源。您所作的任何修改都被自动保存至该文件, 因此, 在停止并重新启动节点后, 可以再次使用经过修改的配置。

域配置文件

/etc/opt/ibm/sna/sna_domn.cfg 文件包含关于 Communications Server for Linux 域资源 (不与特定本地节点相关联的资源) 的信息。此文件的主副本驻留在主控服务器上。

可调用 TP 数据文件

/etc/opt/ibm/sna/sna_tps 文件包含 Communications Server for Linux 启动可调用 (目标) TP 所需要的信息, 并且还可提供其他信息 (如访问该 TP 所需的安全级别)。此文件驻留在运行 TP 的计算机上。

有关此文件的更多信息, 请参阅第 75 页的『定义 TP』。

可以使用 Motif 管理程序、命令行管理程序或 NOF API 来修改配置。所有这些工具都以适当的方式对节点配置文件或域配置文件进行必需的更改。因为配置信息是作为纯文本存储的, 所以您还可以使用标准 ASCII 文本编辑器 (如 **vi**) 或通过 shell 脚本使用 **awk** 或 **sed** 之类的 Linux 实用程序来直接修改该文件。使用文本编辑器对配置文

件所作的任何更改都必须在启动 Communications Server for Linux 之前进行。有关 Communications Server for Linux 配置文件格式的更多信息，请参阅 *Communications Server for Linux Administration Command Reference*。

注：Communications Server for Linux 配置是一个动态过程；在启动 Communications Server for Linux 软件之前不必定义整个配置。配置文件提供了可用资源的初始定义，但您可以在运行 Communications Server for Linux 软件时根据需要添加、删除或修改资源。Communications Server for Linux 存储当前定义，因此在需要重新启动系统时可以再次使用该定义。

下列文件包含关于 Communications Server for Linux 客户机 / 服务器网络的信息：

SNA 网络数据文件

/etc/opt/ibm/sna/sna.net 文件包含关于哪个服务器是主控服务器以及哪些服务器可以充当备份服务器的信息。这个二进制文件驻留在主控服务器上。可以使用管理程序或 NOF API 来修改此文件的内容。

有关此文件以及如何修改它的更多信息，请参阅第 47 页的『配置客户机 / 服务器功能』。

客户机网络数据文件

sna_clnt.net 文件包含关于如何访问 Communications Server for Linux 服务器的信息，这些信息是 IBM Remote API Client 所必需的。此文本文件驻留在客户机上。可以使用标准 ASCII 文本编辑器来修改此文件的内容。

有关此文件以及如何修改它的更多信息，请参阅第 126 页的『客户机网络数据文件 (sna_clnt.net)』。有关在 Windows 客户机上配置等效信息的信息，请参阅第 109 页的第 10 章，『管理 Communications Server for Linux 客户机 / 服务器系统』。

NOF 应用程序

Communications Server for Linux NOF API 提供了与命令行管理程序相同的管理功能，从而使您能够定义和管理 Communications Server for Linux 资源。这意味着您可以编写自己的应用程序来管理 Communications Server for Linux。

有关更多信息，请参阅 *Communications Server for Linux NOF Programmer's Guide*。

诊断工具

Communications Server for Linux 提供了几个诊断工具来帮助您诊断和更正在 Communications Server for Linux 操作期间遇到的问题：

- 任何检测到问题或异常（可能指示问题原因的异常条件）的组件都将一个条目写入错误日志文件。另外，可将所有重要的系统事件都记录在审计日志文件中。您可以确定要记录哪些类型的事件（问题、异常或审计）。在客户机 / 服务器网络配置中，可以对要在所有服务器上记录的事件类型指定全局设置，然后在各个服务器上覆盖这些设置（如果有必要的话）。
- Communications Server for Linux 还维护一个使用情况日志文件，该文件用来记录关于 Communications Server for Linux 资源的当前使用情况和峰值使用情况的信息。
- 您可以指定用来保存每一类日志信息的文件的名称和目录；如果愿意的话，可以将错误和审计日志信息发送到同一个文件中。在客户机 / 服务器系统上，可以将消息从所有服务器发送到一台服务器上的中央日志文件中（中央记录），也可以将日志消息发送到每台服务器上的独立文件中。

- 日志文件是作为文本文件生成的，可以使用标准 ASCII 文本编辑器（如 **vi**）来查看日志文件。
- 可以选择完全记录（对于每条消息，这将把日志中的原因的详细信息以及所需的任何操作包括在日志文件中）或简明记录（仅包括日志和消息文本的来源的摘要）。当使用简明记录时，如果需要进一步的信息，则可以使用 **snahelp** 命令行实用程序来获取特定消息号的全部原因和操作文本。
- 对于某些错误状态，Communications Server for Linux 除了将问题消息写入错误日志文件外，还将一条消息发送至 Linux 控制台以警告操作员。
- 许多组件可以生成用于记录该组件的活动的跟踪文件。跟踪会降低 Communications Server for Linux 组件的性能，因此通常禁用跟踪。
- 通过使用命令行实用程序，可以对跟踪文件进行过滤以抽取特定的信息，然后对跟踪信息进行格式化以解释其内容或生成消息流的摘要。可以使用标准 ASCII 文本编辑器（如 **vi**）来查看已格式化的输出文件。
- Communications Server for Linux 可以生成警报并将它们发送至主机上的 NetView 程序。这些警报可以是下列任何一项：
 - 来自连接组件的链路警报，用于提供关于连接问题的信息
 - 由使用 MS API 的应用程序提供的警报

有关 Communications Server for Linux 日志消息、使用 Communications Server for Linux 跟踪设施以及解释跟踪文件的信息，请参阅 *Communications Server for Linux Diagnostics Guide*。

有关使用 MS API 的信息，请参阅 *Communications Server for Linux MS Programmer's Guide*。

管理许可权

Communications Server for Linux 管理工具供具有管理 SNA 资源的许可权的“SNA 管理员”受限组使用。要实现这一点，可执行文件归组所有权为 **sna** 的系统管理员登录 **root** 用户拥有。只有作为 **sna** 组的成员的用户才可以修改、启动或停止 Communications Server for Linux 资源；任何将要获得 SNA 管理员许可权的用户都必须是此组的成员。

在标准 Communications Server for Linux 安装中，不是 **sna** 组的成员的用户根本不能运行 Communications Server for Linux 管理工具。适当的话，可以允许这些用户以只读方式运行这些工具，以便他们可以查看配置和状态信息，但不能修改、启动或停止资源。为此，请使用 **chmod** 来将一个或多个适当可执行文件的读取和执行许可权授予任何用户：

| 管理工具 | 可执行文件 |
|------------|---------------------------------------|
| Motif 管理程序 | /opt/ibm/sna/bin/X11/xsnaadmin |
| 命令行管理程序 | /opt/ibm/sna/bin/snaadmin |

然后，任何用户都可以运行适当的管理工具并查看信息，但 Communications Server for Linux 仍阻止不在 **sna** 组中的用户修改、启动或停止资源。

注：如果按上述过程修改文件许可权，则在安装 Communications Server for Linux PTF 或新发行版之后将需要重复此过程。

规划 Communications Server for Linux 配置

在做任何配置更改之前，全面地进行规划是很重要的。您所作的更改不仅对本地节点的用户有影响，而且还可能对网络上的所有用户有影响。

您可能会发现，绘制一张您打算对网络拓扑进行任何更改的图表是有用的。若您打算添加或删除与其他节点的连接，则绘制一张显示您的节点和其他节点的图表。可以使用 Motif 管理程序来收集关于所有现有连接的配置信息并将该信息添加至图表。

当将新的资源添加至图表时，很容易就可以看出它们是否与现有的资源重复或是否有任何冲突名称。同样，图表可以帮助您决定需要除去哪些资源并帮助您避免删除必要的资源。

如果您要给客户机 / 服务器 Communications Server for Linux 系统配置多个节点，则确保在图表中包括所有 Communications Server for Linux 节点及其连接资源。

一旦确定需要执行的更改，就可以收集需要的配置信息。可以使用 Motif 管理程序的联机帮助文件中的任务表或『规划工作表』中描述的规划工作表来指导您收集特定 Communications Server for Linux 功能的配置信息。

规划工作表

在开始配置 Communications Server for Linux 的资源之前，应该收集新资源的所有配置数据。要记录您需要支持的特定功能或应用程序的所有信息，使用第 131 页的附录 A，『配置计划工作表』中的规划工作表。

您可能将需要从几个信息源（如网络管理员、主机管理员、应用程序员以及最终用户等）处收集配置信息。

若您正在尝试与另一节点连接，则那个节点的管理员是一个关键联系人。节点的管理员可以告诉您其节点上所有资源的名称、地址和特性。通常，您将需要确保在本地节点和远程节点上输入了匹配的配置参数。

任务表

Motif 管理程序的联机帮助屏幕包含一些任务表，这些任务表提供对特定配置任务的指导。这些任务表包含一些指针，这些指针指向您将用来输入配置信息的对话框的所有帮助屏幕。您可以使用这些指针来浏览帮助并准确地查看您必须收集的数据。

这些任务表还引用了您必须用来输入配置信息的各个窗口和对话框的更详细的帮助。那些帮助屏幕解释了您必须填写或选择的每个字段。

启用与禁用本地系统上的 Communications Server for Linux

本节说明如何在 Linux 服务器上启用和禁用 Communications Server for Linux 软件。

在可以使用任何 Communications Server for Linux 工具（包括 Motif 管理程序）之前，必须启用 Communications Server for Linux 软件。通常在安装 Communications Server for Linux 后，该软件会自动启用，但是，如果有必要的话，可以手工启用它。

指定 **Communications Server for Linux** 程序的路径

Communications Server for Linux 可执行程序存储在特定于 Communications Server for Linux 的目录中；当运行这些程序时，需要指定此目录的路径。您可以在第一次运行这些程序之前通过将目录添加至 PATH 环境变量中来指定路径，或通过每次运行这些程序时包括目录名来指定路径。

Motif 管理程序存储在目录 `/opt/ibm/sna/bin/X11` 中，其他程序则存储在目录 `/opt/ibm/sna/bin` 中。如果将这些目录添加至 `.login` 或 `.profile` 文件的 PATH 环境变量的定义中，则 Communications Server for Linux 能够自动找到这些程序。或者，您可以在运行程序时指定目录名，如下列示例中所示：

```
/opt/ibm/sna/bin/sna start
```

```
/opt/ibm/sna/bin/snaadmin query_node
```

```
/opt/ibm/sna/bin/X11/xsnaadmin
```

本手册中显示的样本命令行假定您已将这些目录添加至 PATH 环境变量，因此这些命令行不包括目录名。

启用 **Communications Server for Linux** 服务器

本节描述如何在作为服务器安装的（即安装了 SNA 节点组件的）计算机上启用 Communications Server for Linux。如果要在客户机上启用 Communications Server for Linux，请参阅第 126 页的『在 AIX 或 Linux 上启用和禁用 Remote API Client』。

在可以配置或管理本地节点（在本地或从远程 Communications Server for Linux 节点进行操作）之前，必须在本地系统上启用 Communications Server for Linux。

要启用 Communications Server for Linux 软件，在 Linux 命令提示符处输入下列命令：

```
sna start [ -s ] [ -m kernel_memory_limit ] [ -t ]
```

当您安装 Communications Server for Linux 时，安装实用程序自动更新启动文件 `/etc/rc.d/init.d/snastart`，以便包含 `sna start` 命令。这可确保 Communications Server for Linux 在系统启动时自动启动。如果您不想自动启动 Communications Server for Linux，则可除去或注释掉此行，然后遵循本节中的指示信息以便以手工方式启用 Communications Server for Linux 软件。

`sna start` 命令的参数和选项如下所示：

-s 指定 Communications Server for Linux 不应该将消息写至系统控制台。如果未使用此选项，则 Communications Server for Linux 在结束时会将消息写至控制台，并将某些错误日志消息文本写至控制台以及日志文件。

-m kernel_memory_limit

指定 Communications Server for Linux 在任何时候应该使用的内核内存的最大容量（以千字节计）。（内核内存用于内部数据结构。）如果 Communications Server for Linux 组件尝试分配的内核将导致当前分配给 Communications Server for Linux 组件的内存总量超过此限制，则该分配尝试失败。

如果您未使用此选项，则不限制内核内存的使用。

- t 在内核组件之间的所有接口上激活跟踪，并激活客户机 / 服务器跟踪。（此选项不打开 DLC 跟踪。）跟踪允许您诊断启动期间发生的问题。如果未使用此选项，则跟踪在所有接口上都是不活动的；然后在需要时可以使用命令行管理程序 **snaadmin** 来激活对特定接口的跟踪。

对所有接口进行跟踪会降低 **Communications Server for Linux** 组件的性能。在启用软件后，可以使用命令行管理程序 **snaadmin** 来停止对任何不需要跟踪的接口的跟踪。有关跟踪的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。

Communications Server for Linux 将消息写至标准错误（通常是终端的屏幕）以表示它正在初始化，并表示初始化是否成功完成。

如果初始化失败，则这些消息将包含关于错误原因的信息，并且包含其他信息（在适当的时候），如 **Linux** 操作系统错误消息。写至标准错误的文本还可能包括指示您可以在错误日志文件中查找进一步信息的信息。然后 **sna start** 命令以指示错误性质的非零退出代码结束。

有关退出代码值的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。

用于 **sna start** 命令的高级选项

在某些情况下，尤其是当您在彻底检验新的 **Communications Server for Linux** 配置时，您可能想要使用已保存至临时文件的配置（而非文件 **/etc/opt/ibm/sna/sna_node.cfg** 和 **/etc/opt/ibm/sna/sna_domn.cfg** 中的标准配置）来启动 **Communications Server for Linux**。为此，您可以对 **sna start** 命令使用下列其他选项：

- n *node_config_file*
- d *domain_config_file*

node_config_file 是您将节点配置保存至的文件的完整路径名（而非 **/etc/opt/ibm/sna/sna_node.cfg**）；*domain_config_file* 是您将域配置保存至的文件的完整路径名（而非 **/etc/opt/ibm/sna/sna_domn.cfg**）。

注：通常不应使用这些选项。除非您有特定需求，否则请勿使用它们。

在您使用这些选项来运行 **Communications Server for Linux** 时，**snagetpd** 命令将不能正常工作，这是因为它总是从标准配置文件中收集信息。使用 **snagetpd** 之前，请在不采用这些选项的情况下启动 **Communications Server for Linux**，以确保运行时使用标准配置文件。

禁用 **Communications Server for Linux** 服务器

在服务器上禁用 **Communications Server for Linux** 软件会自动停止 **Communications Server for Linux** 节点及其关联的连接组件。禁用 **Communications Server for Linux** 还会阻止任何其他进程（如 3270 仿真程序）使用此服务器上的 **Communications Server for Linux** 资源。

一般来说，您应该在用户完成使用各个服务时停止这些服务，并仅当没有任何 **Communications Server for Linux** 活动时才禁用系统。在客户机上禁用 **Communications Server for Linux** 软件会阻止该客户机上运行的任何程序访问 **Communications Server for Linux** 设施。

若您在用户活动时需要禁用 **Communications Server for Linux**，则警告用户 **Communications Server for Linux** 将停止，在禁用该软件之前给他们时间结束其活动。使用 **Motif** 管理程序或命令行管理程序来查看活动用户的详细信息。

当禁用 **Communications Server for Linux** 软件时，如果某个 3270 仿真程序正在使用节点上的 LU，则所有使用这些 LU 的 3270 仿真会话都将结束。该程序继续运行，但用户在重新启用该软件之前不能使用这些会话。使用 **APPC**、**CSV**、**LUA**、**NOF** 或 **MS API** 的应用程序由 **COMM_SUBSYSTEM_ABENDED** 返回码通知，**CPI-C** 应用程序由 **CM_PRODUCT_SPECIFIC_ERROR** 返回码通知。

要禁用 **Communications Server for Linux** 软件，在 Linux 命令提示符处输入下列命令：

```
sna stop
```

若禁用 **Communications Server for Linux** 成功，则 **sna stop** 返回退出代码 0。任何其他退出代码指示发生错误且未禁用 **Communications Server for Linux** 软件。有关退出代码值的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。

使用 **Motif** 管理程序

Motif 管理程序提供了一个用户友好的界面来配置 **Communications Server for Linux**。此程序是管理 **Communications Server for Linux** 的建议工具，它指导您完成配置过程，并且最大程度地减少了创建可工作的配置所需提供的信息。

您还可以使用 **Motif** 管理程序来管理处于活动状态的 **Communications Server for Linux** 系统。该管理程序使您能够在 **Communications Server for Linux** 处于活动状态时对配置进行更改并应用这些更改。您可以添加、修改和除去资源（在大多数情况下，甚至在节点和其资源活动时），并可立即使用修改的配置继续操作。

Motif 管理程序通过用于执行配置的同一界面来显示最新状态信息，这使得易于访问域和节点资源的状态信息。

另外，可以使用 **Communications Server for Linux** 命令来进行配置和系统管理。第 45 页的『使用命令行管理程序』中提供了配置和管理命令摘要。

调用 **Motif** 管理程序

要将 **Motif** 管理程序用于 **Communications Server for Linux**，首先确保按第 34 页的『启用 **Communications Server for Linux** 服务器』中所述启用 **Communications Server for Linux**。（如同任何 X/**Motif** 应用程序一样，您可能还需要设置 **DISPLAY** 环境变量来指示合适的 X 服务器。）

要启动在后台运行的 **Motif** 管理程序，发出下列命令：

```
xsnaadmin &
```

在客户机 / 服务器环境中，**Communications Server for Linux** 显示“域”窗口。

对于单机系统，Communications Server for Linux 通常显示“节点”窗口。然而，如果尚未配置本地节点，则它会显示一个帮助屏幕，帮助您首次配置节点。

注：本指南使用术语“窗口”来描述显示有关 Communications Server for Linux 资源的信息的 Motif 窗口。一个窗口可以包含一个或多个部分或窗格。对话框是可供输入信息的 Motif 窗口。

“资源”窗口

“域”窗口和“节点”窗口显示您需要的大部分信息，并且它们使您能够方便地访问其他信息。从那些窗口中，可以方便地显示关于本地网络中的资源的信息。

“域”窗口显示所有已定义的节点，并允许添加、删除、启动和停止节点。双击任何节点将对该节点打开“节点”窗口。

“节点”窗口显示特定节点的所有关键资源。

“域”和“节点”窗口中的菜单提供了下列功能：

选择 此菜单中的功能与当前在“域”窗口中选择的节点或者当前在“节点”窗口中选择的项相关。从此菜单中，可以启动或停止节点或者对其进行缩放以显示其“节点”窗口。当在“节点”窗口中选择一个项时，可以使用此菜单中的控件来控制、修改或删除该项，也可以在当前选择的窗格中添加新项。

服务 此菜单使访问为公共任务配置节点所需要的所有对话框很容易。使用此菜单，您可以添加或修改资源，或者获取对配置和管理任务的帮助。

诊断 可以通过此菜单中的菜单项来对记录和跟踪进行控制。

窗口 可从此菜单很容易地访问其他窗口。这些窗口包括：

- LU 池窗口
- CPI-C 目标名称窗口

根据您选择的资源和选项，管理程序可以提供其他资源窗口、配置对话框或状态记录。您还将看到允许您选择配置特定资源的上下文对话框、要求您确认选择的确认对话框以及提供反馈或错误信息的弹出式消息。每个窗口和对话框还包括一个帮助选项。

“域”窗口

“域”窗口显示正在使用的系统的 Communications Server for Linux 域中的每个活动 SNA 节点。（如果节点上没有运行 Communications Server for Linux，则该节点不会出现在“域”窗口中。）每个节点都是使用系统的名称标识的。“域”窗口还显示域中每个节点的当前状态。

注：如果某台服务器意外地未出现在“域”窗口中的节点列表中，请验证该服务器是否已打开并且是否正在该服务器上运行 Communications Server for Linux 软件。如有必要，使用 **sna start** 命令在该节点上启动 Communications Server for Linux 软件（请参阅第 34 页的『启用 Communications Server for Linux 服务器』）。

在一个域中，总有一个节点被标识为域的配置服务器。“域”窗口在该节点旁显示“主控”一词。主控配置服务器始终包含域资源的配置信息。在此窗口中，备份配置服务器由“备份”一词标识。备份配置服务器包含域资源的配置信息的副本。

图 13 中显示了“域”窗口的示例。

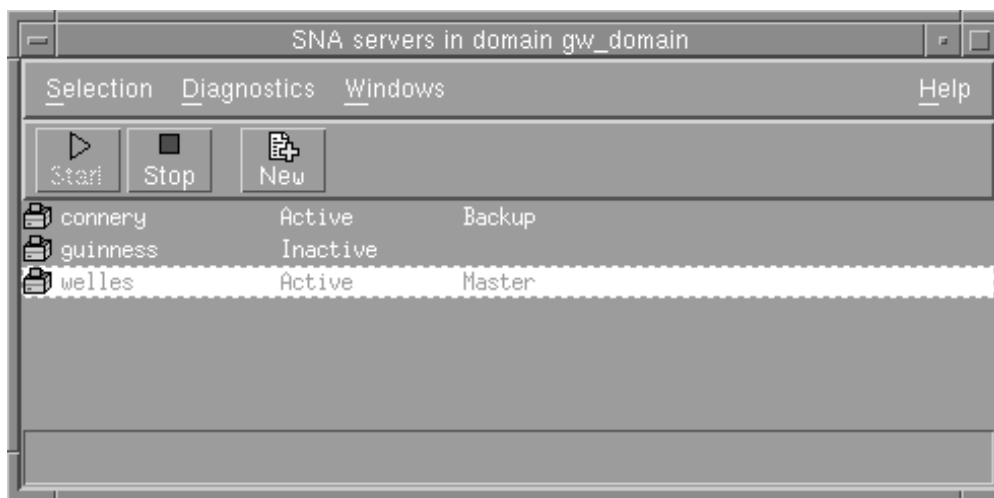


图 13. Communications Server for Linux 域窗口

如果域中的任何活动节点（运行 Communications Server for Linux 的节点）未经配置，则 Communications Server for Linux 将提示您配置该节点。

注：“域”窗口不列示 IBM Remote API Client。客户机使用 Communications Server for Linux 服务器（SNA 节点）的资源来访问 SNA 资源。

可以从“域”窗口中执行下列任何管理任务：

启动或停止域中的任何节点

选择节点所在的行并单击此窗口中的**启动**或**停止**按钮。（另外，可以单击节点所在的行并接着从**选择**菜单中选择**启动节点**或**停止节点**。）

管理特定节点

在“域”窗口中双击该节点所在的行。（另外，可以单击节点所在的行并接着从**选择**菜单中选择**属性**。）还可以从**窗口**菜单中选择节点的窗口。

当您选择要管理的节点时，Communications Server for Linux 将显示“节点”窗口，如第 39 页的图 14 中所示。（对于单机系统，由于域只有一个节点，所以 Communications Server for Linux 不显示“域”窗口。相反，当您启动管理程序时，Communications Server for Linux 将立即显示“节点”窗口。）

将节点添加至域的服务器列表

单击节点所在的行并从**选择**菜单中选择**成为配置服务器**。

从域的服务器列表中除去节点

单击节点所在的行并从**选择**菜单中选择**除去配置服务器**。

对域中的所有节点配置记录

从**诊断**菜单中选择**记录**。

对特定节点打开或关闭跟踪

单击节点所在的行并从**诊断**菜单中选择**跟踪所选节点**。

获取关于域资源的信息

在**窗口**菜单中选择任何选项。除了共享域资源之外，**窗口**菜单还列示域中的每个“节点”窗口。

节点窗口

图 14 显示了一个样本“节点”窗口。标题栏显示系统的名称。

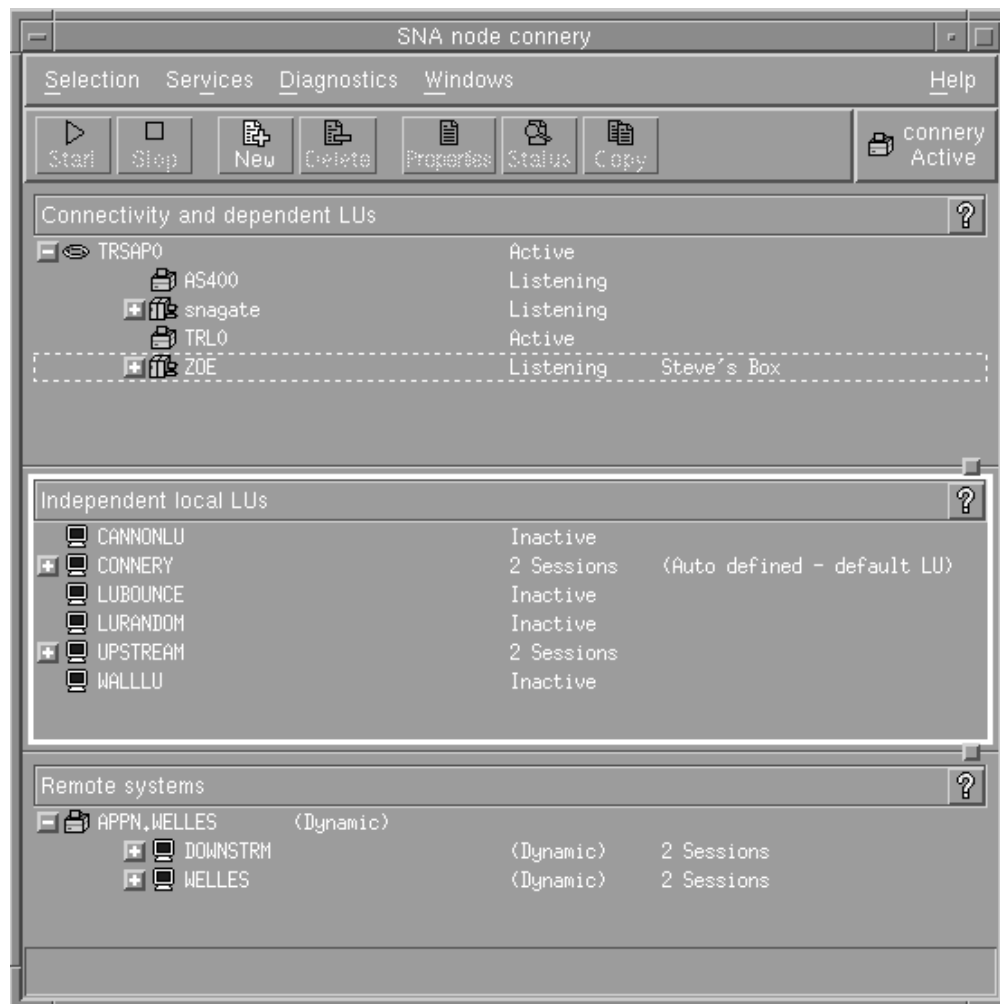


图 14. 节点窗口

从“节点”窗口，可以添加、删除、修改和管理 Communications Server for Linux 节点的所有资源和组件。窗口中资源的布局显示了资源之间的相互关系并允许您控制显示哪些资源。

“节点”窗口右上角的“节点”框指示节点是活动的还是不活动的。

总是显示节点上定义的任何端口、本地 LU 和远程节点。“节点”窗口将每个链路站显示在其父端口之下，每个从属 LU 显示在其父链路站之下。它还在本地 LU 和远程节点之下显示了伙伴 LU。

对于不同类型的节点资源，“节点”窗口的主体分成下列窗格：

“连接”窗格

“节点”窗口的顶部窗格列示节点的连接资源，包括端口、每个端口上的链路站或 LU、特定链路站或 PU 上的从属 LU。对于每个资源，此窗口显示当前状态信息。

“独立本地 LU” 窗格

中间窗格显示节点的独立 LU。对于每个 LU，此窗口还显示关于使用该 LU 的会话的信息。

“远程系统” 窗格

底部窗格显示关于远程节点和伙伴 LU 的信息。它还显示关于每个远程节点或伙伴 LU 的会话信息。

要更改窗格的相对大小，单击并拖动窗格间的边界。

您可以在窗格内单击来选择该窗格。还可以通过单击资源所在的行来选择窗格中的特定资源。要查看或修改某项的配置，可以双击该项。（您还可以使用此窗口上的按钮和菜单来访问特定资源的配置信息。）

对于列示的每个项，属于该项的资源都嵌套在项的信息中。例如，链路站被分组在它们所属的端口之下。如果某个项的资源当前未显示，则可以单击该项旁边的**展开按钮**



来显示它们，也可以单击**收缩按钮**



来隐藏某个项的资源。

可以从“节点”窗口执行下列管理任务：

启动或停止资源

选择资源并单击**启动**或**停止**按钮。（或者，可以从**选择**菜单选择**启动项**或**停止项**。）

为某个项添加新资源

选择项并单击**新建**按钮（或从**选择**菜单选择**新建**）。例如，要为端口添加链路站，选择该端口并单击**新建**按钮。

删除资源

选择资源并单击**删除**按钮（或从**选择**菜单选择**删除**）。

查看或修改任何资源的配置

选择资源并单击**属性**按钮（或从**选择**菜单选择**属性**）。

获取关于任何资源的状态信息

选择资源并单击**状态**按钮（或从**选择**菜单选择**状态**）。

复制任何资源的配置

选择资源并单击**复制**按钮（或从**选择**菜单选择**复制**）。

另外，可以从**服务**菜单中选择节点的特定配置任务，从**诊断**菜单中对记录（对于域）和跟踪（对于节点）进行控制，以及通过选择**窗口**菜单中的其中一个项来查看或修改域资源。

资源项

窗口中资源的布局显示了资源之间的关系。

如果某个项具有一个或多个与它关联的子项，则该项旁边有一个**展开按钮**或**收缩按钮**。**展开按钮**指示关联的子项是隐藏的。可以单击**展开按钮**来显示这些子项。**收缩按钮**指示子项已显示。可以单击**收缩按钮**来隐藏它们。如果某个项旁边这两个按钮中的一个也没有，则表示该项没有相关联的子资源。

例如，某个链路站与某个特定的端口关联。在“节点”窗口的“连接”窗格中，该链路站以及与那个端口关联的所有其他链路站一起显示在其父端口之下。始终显示端口，但您可以选择是显示还是隐藏关联链路站的列表。同样，可以展开或收缩具有关联 LU 列表的链路站来显示或隐藏这些 LU。

父资源必须总是在其子资源之前配置，删除父资源会导致删除其所有子资源。

工具栏按钮

“资源”窗口包括工具栏按钮，以便执行常用功能。Communications Server for Linux 的工具栏显示在图 15 中。



图 15. Communications Server for Linux 工具栏

并非所有的按钮都出现在每个资源窗口的工具栏中。如果按钮的操作对于当前选择的项无效（或某个操作需要选择项，但尚未进行选择），则该按钮的外形显示为灰色，不能选择该功能（不能按该按钮）。下列按钮出现在资源窗口上：



启动选择的项。



停止选择的项。



添加新的资源项。（在“节点”窗口中，您将资源添加至选择的窗格中。）



删除选择的项。



打开所选项的对话框以查看或修改该项的配置。



复制选择的项。按此按钮会打开一个其字段复制所选项配置的对话框。完成该对话框的字段（填写新项的名称）以添加新资源。



显示所选项的当前状态。

许多资源（如端口和链路站）在活动时不能修改。然而，通过选择资源并单击**属性**按钮以打开其对话框，可以查看活动资源的参数，或单击**状态**按钮以查看该资源的详细状态信息。

资源对话框

资源对话框显示该资源的当前配置信息。第 43 页的图 16 中显示了 0-3 型 LU 的样本对话框。

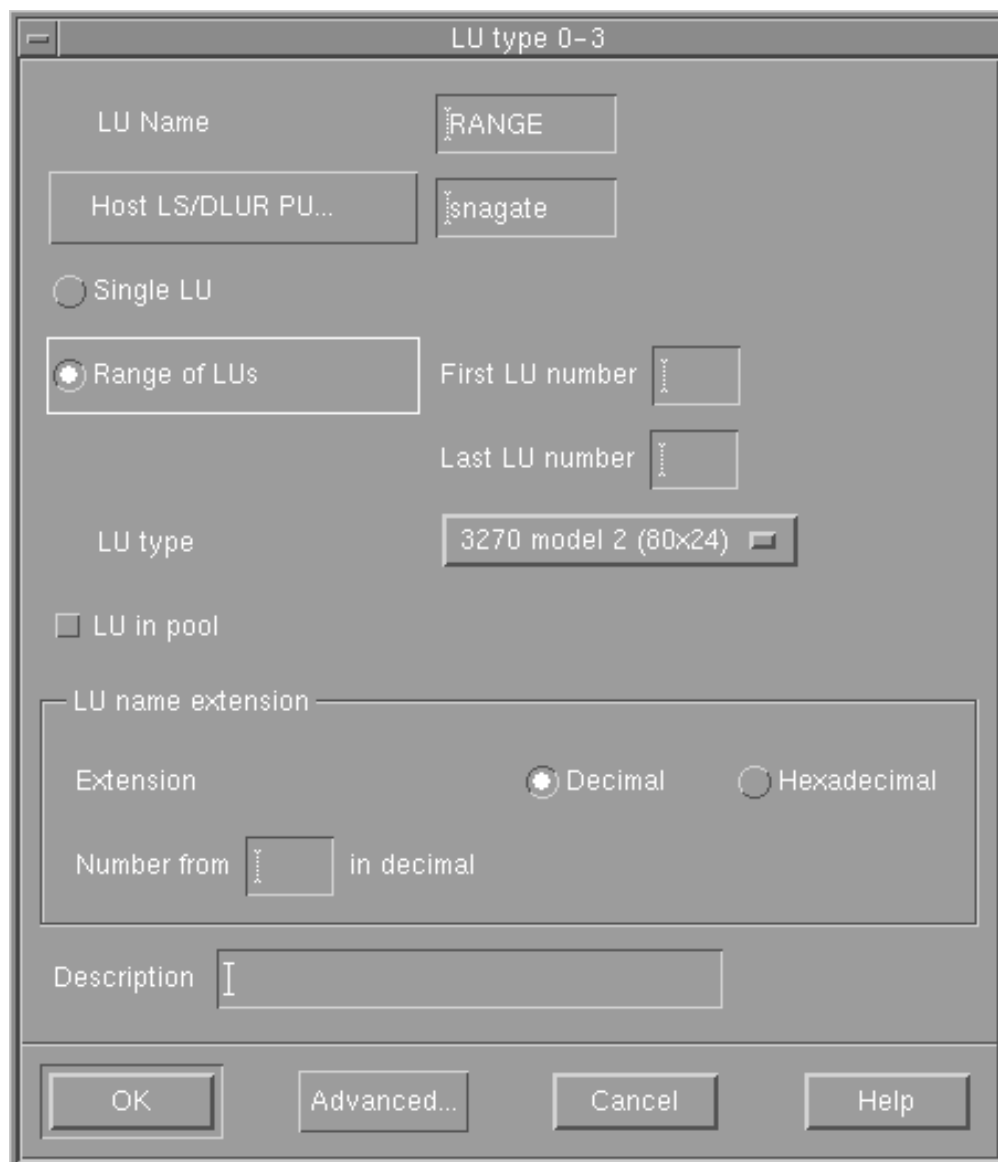


图 16. 样本对话框

资源对话框指导您完成配置过程并只要有可能就提供缺省值。例如，当您添加从属 LU 时，Motif 管理程序用您指定的链路站上的可用 LU 号自动填写 LU 号字段。如果未提供必需值，则程序将显示一条弹出消息，该消息指示了需要提供的信息。

大多数对话框提供了描述字段；您在那里输入的信息在显示该资源的窗口中显示。

如果允许您在资源对话框中更改信息（当您正在添加新项或修改现有的项时），该对话框包含确定和取消按钮。完成时按确定按钮，或按取消按钮退出而不更改该资源的配置。

如果不能在资源对话框中更改信息（例如，如果资源的配置不能在它活动时被修改），则该对话框包括一个关闭按钮而不是确定按钮。当您查看完对话框中的信息时单击此按钮。

要获取对话框的上下文相关的帮助，单击帮助按钮。

注：基本 Motif 对话框只显示关键配置字段；Communications Server for Linux 为高级字段提供缺省值。要访问高级配置参数，单击**高级**按钮。如果您决定调整高级参数，在打开高级对话框之前完成基本对话框，因为高级对话框可以根据您为基本参数输入的值而更改。有关高级配置字段的信息，请参阅 Motif 管理程序的联机帮助。

状态对话框

当您选择某个资源并单击**状态**按钮时，Motif 管理程序将显示该资源的详细状态信息，如图 17 中所示。

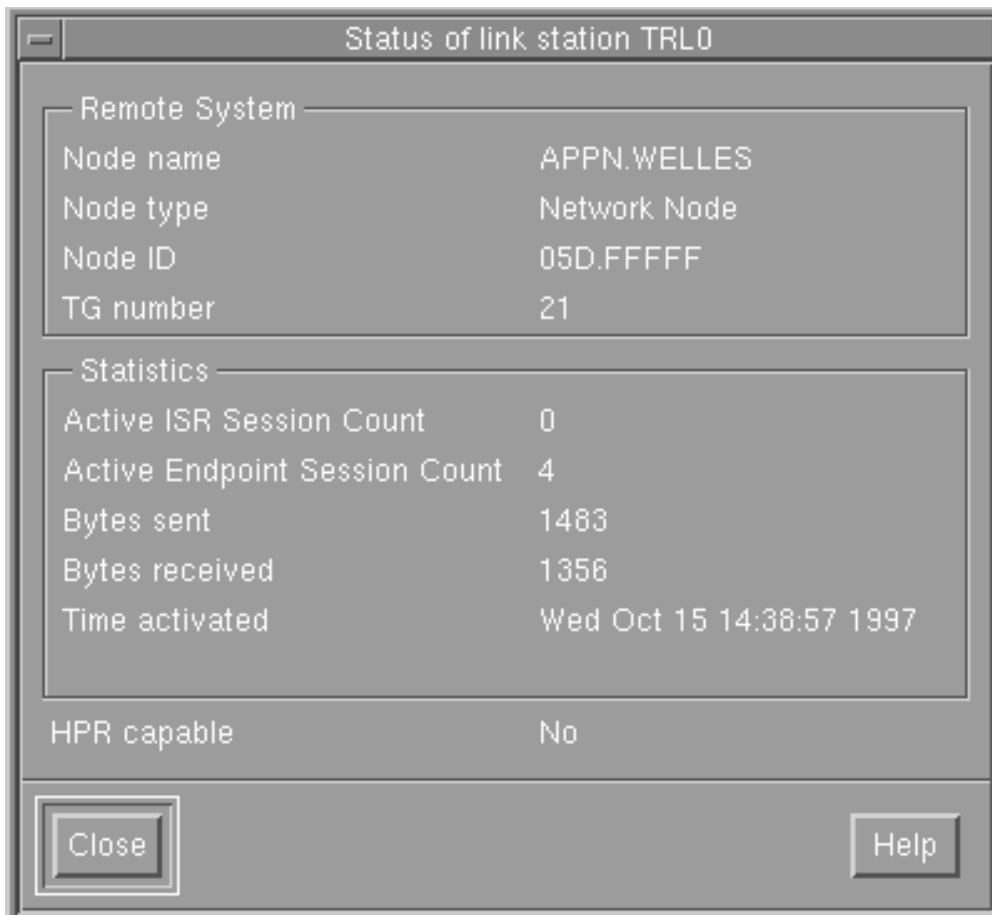


图 17. 样本状态对话框

状态对话框显示关于该资源当前状态的信息。信息在您查看它时动态更新。

帮助窗口

Motif 管理程序的联机帮助对您需要执行的每个配置任务提供详细指导。特别，任务表可以指导您完成在配置特定资源时需要执行的每个步骤。第 45 页的图 18 中显示了配置节点参数的任务表（始终是配置 Communications Server for Linux 的第一个步骤）。

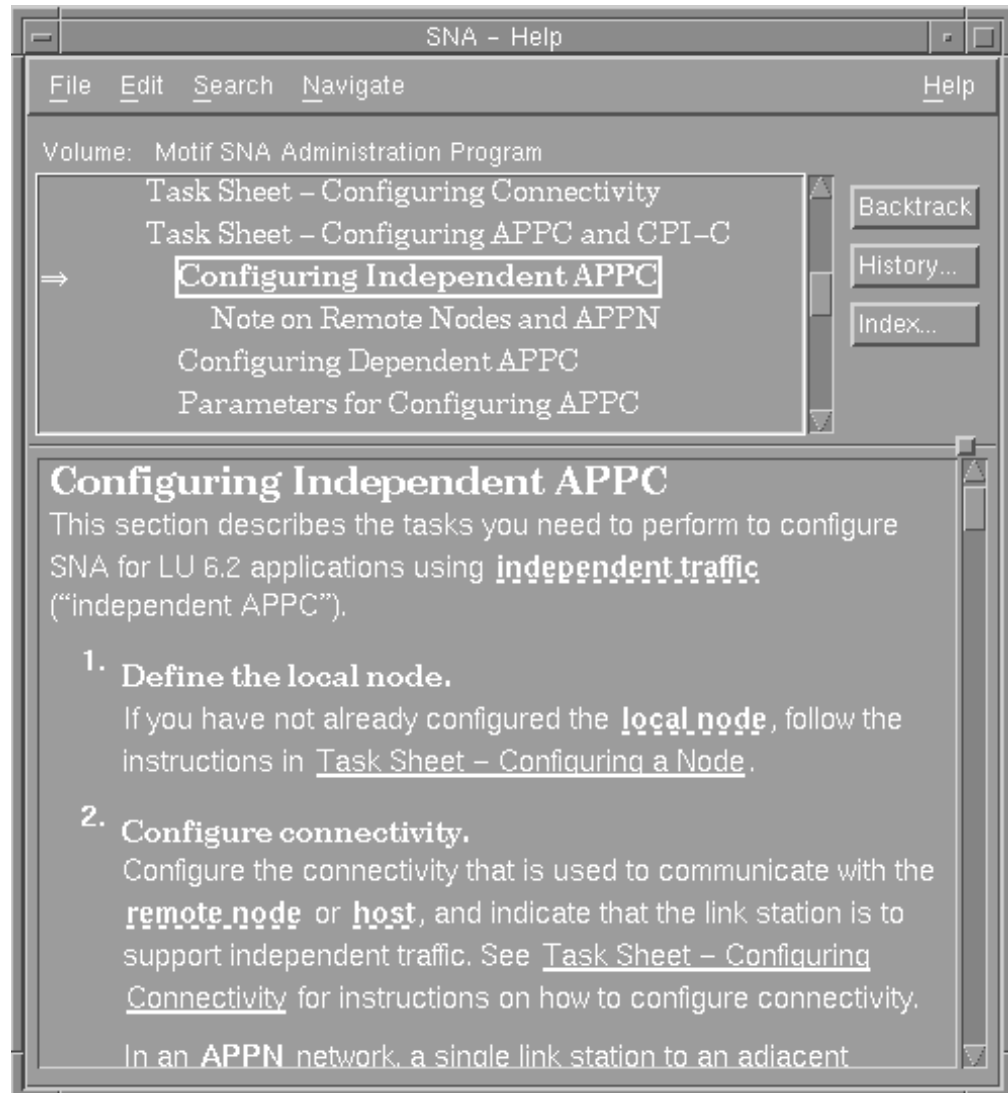


图 18. 样本帮助窗口

每个窗口和对话框、错误消息以及 SNA 概念都包括其他的帮助窗口。

使用命令行管理程序

命令行配置使您能够更改所有 Communications Server for Linux 配置参数。您可以使用它来配置整个 Motif 管理程序中可用的任何资源，可以设置或更改未在 Motif 程序中显示的配置参数。然而，此管理方法与 Motif 管理相比，一般需要提供更多的信息。另外，还必须确保您提供的信息有效并与现有的资源定义一致。（建议使用 Motif 管理程序，因为它确保您输入的数据一致。另外，它还可以根据菜单或对话框选项推导出许多配置值，并根据可用的定义填写各个值。）

大多数管理命令通过 **snaadmin** 命令行管理程序使用。可以按以下格式发出 **snaadmin** 命令：

```
snaadmin command, parameter1=value1, parameter2=value2, ....  
           {subrecord_name1}, sub_param1=sub_value1,  
           sub_param2=sub_value2...
```

可以使用下列任何命令来获取有关 **snaadmin** 命令行管理的帮助:

- **snaadmin -h** 提供有关命令行管理的基本帮助和关于命令行帮助的用法信息。
- **snaadmin -h -d** 提供可以提供给 **snaadmin** 程序的命令的列表。
- **snaadmin -h *command*** 提供对指定的 *command* 的帮助。
- **snaadmin -h -d *command*** 提供对指定的 *command* 的详细帮助, 包括可随该命令指定的配置参数的列表。

如果命令包含 **-n** 选项来指定服务器名称, 则某些命令可以从 IBM Remote API Client 中发出。这样的命令与从指定的服务器上发出该命令效果相同。

本节的余下部分概述了不同类型资源的管理命令。某些命令类型列示如下:

status_*
提供资源类型的摘要信息。

define_*
在配置文件中创建新的 **define_*** 记录, 或用新定义替换相同资源的现有记录。

delete_*
从文件中除去相应的 **define_*** 记录。

query_*
从配置文件中返回关于适当组件的信息, 但不修改该文件。

有关命令行配置的完整信息, 请参阅 *Communications Server for Linux Administration Command Reference*。

第 3 章 基本配置任务

本章概述了配置任务并说明如何配置 Communications Server for Linux 节点。它还说明当在客户机 / 服务器环境中使用 Communications Server for Linux 时如何配置主控服务器和备份服务器。

配置客户机 / 服务器功能

仅当您安装了 Communications Server for Linux 来在客户机 / 服务器环境（在同一网络中具有多个 Communications Server for Linux 节点）中运行时，本节内容才和您相关。

许多资源（如端口和 LU）是在各个节点上配置的。这些资源被称为“节点资源”。

其他资源是所有节点的公共资源；在整个域中，只维护该资源的一个定义。此类资源被称为“域资源”。域资源定义只存储在域的主控服务器上，并且可以从域中的所有节点访问它们。

注：独立的 Communications Server for Linux 系统只有一台服务器；该服务器始终充当主控服务器。

在客户机 / 服务器环境中，服务器可标记为配置服务器；Communications Server for Linux 保持这些配置服务器的列表。列示的第一个服务器是主控服务器，列示的其他任何服务器是备份服务器。服务器是按顺序列示的，因此，如果主控服务器不可用，则由列示的第二个服务器（即第一个备份服务器）取代，如果主控服务器和第一个备份服务器都不可用，则由列示的第三个服务器（即第二个备份服务器）取代，依此类推。

当域中的任何节点都是活动的时候，域中第一个可用的配置服务器（可联系到且正在运行 Communications Server for Linux 软件的第一个服务器）成为主控服务器。如果当前主控服务器不可用（因为可能由于网络故障而联系不到它，或因为它正在运行的 SNA 软件已停止），则列表中下一个可用的配置服务器成为新的主控服务器。

Communications Server for Linux 可在没有主控服务器的情况下运行。如果联系不到配置服务器列表中的任何服务器，则会出现这种情况。如果出现这种情况，则您可仅查看和配置可联系的服务器上的节点资源。

注：您不能直接指示哪个节点充当主控服务器；将基于将节点添加至配置服务器列表的顺序来选择主控服务器。如果想要将某个服务器移至列表顶部，则从列表中除去所有其他节点，然后再次添加它们。

还可以使用下列管理命令来查询、添加和删除配置服务器：

query_sna_net

在文件中列示服务器。

add_backup

将新服务器添加至列表末尾。

delete_backup

从列表中除去服务器。可以使用 **delete_backup** 命令来删除主控服务器（以使列示的第二台服务器作为主控服务器接管控制权）或备份服务器（以使其不再作为主控服务器）。

注：如果某个服务器是列示的仅有的一个正在运行 Communications Server for Linux 软件的服务器，则您不能删除它，因为在这种情况下，没有任何其他服务器可取代主控服务器。在客户机 / 服务器配置中，至少需要一个启用的主控服务器。

第 109 页的第 10 章，『管理 Communications Server for Linux 客户机 / 服务器系统』提供有关高级客户机 / 服务器配置的信息，包括如何将客户机和服务器移至不同 Communications Server for Linux 域和如何配置客户机操作的详细信息。

配置节点

在系统上配置 Communications Server for Linux 的第一个步骤是配置本地节点。节点配置提供节点参与 APPN 网络所需要的基本信息。必须先配置节点，然后才能定义节点的连接或其他资源。

如果节点已配置，则在更改节点配置之前必须停止该节点。

要配置该节点，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单选择配置节点参数。

命令行管理程序

发出 **define_node** 命令。

节点配置的高级参数提供了对与未定义伙伴 LU 的会话、安全性故障的报告以及有限资源超时的控制。

节点配置参数

进行节点配置需要下列信息：

APPN 支持

节点的 APPN 支持的级别：

- 如果网络不是 APPN 网络，则将节点配置为低入口联网节点。
- 要参与另一节点提供会话路由服务的 APPN 网络，或只在本地节点上使用 DLUR，则将节点配置为端节点。
- 要在 APPN 网络中提供中间路由服务，或要为下游节点提供传递（Pass-Through）DLUR 服务，则将节点配置为网络节点。
- 要对分支网络中不是 APPN 主干网络的一部分的其他节点提供会话路由服务，将该节点配置为分支网络节点。

控制点名称

本地节点的标准控制点名称。因为可能需要在网络中的其他节点上配置此名称，所以咨询 SNA 网络设计者以确定该名称。

当您定义控制点时，Communications Server for Linux 自动定义具有相同名称的本地 LU。该 LU 可充当节点的缺省本地 LU。

控制点别名

缺省本地 LU 的本地别名。如果缺省本地 LU 由独立 LU 6.2 LU 使用，则提供此值。

节点标识

本地节点上的 PU 的标识。仅当节点将用于使用缺省（控制点）LU 的从属通信时才提供值。

其他配置

在配置节点后，继续执行下列配置任务：

- 按第 51 页的第 4 章，『定义连接组件』所述配置连接。
- 按第 69 页的第 6 章，『配置 APPC 通信』或第 89 页的第 7 章，『配置用户应用程序』所述配置节点资源（LU）。
- 按第 89 页的第 7 章，『配置用户应用程序』所述配置应用程序。

配置记录

Communications Server for Linux 将描述异常事件（以及正常事件（可选））的日志消息写入日志文件。当您尝试诊断问题时，首先要查看的就是日志文件，因为记录消息提供了关于问题的原因以及您应该执行的操作的信息。

Communications Server for Linux 记录下列类别的事件的消息：

问题 以用户可察觉的方式降低系统性能的异常事件（如会话的异常终止）。

异常 降低系统性能但用户不会立即察觉的异常事件（如资源缺乏），或不降低系统性能但可能指示后面的异常事件或问题的原因的事件（如从远程系统接收意外的消息）。

审计 正常事件（如启动会话）。

Communications Server for Linux 还维护一个使用情况日志文件，该文件用来记录关于 Communications Server for Linux 资源的当前使用情况和峰值使用情况的信息。

为了区分与正常状态和错误状态相关的日志，将不同类别的消息记录至不同的文件。将问题和异常消息记录至错误日志文件；将审计消息记录至审计日志文件。

Communications Server for Linux 提供了备份机制以防止日志文件变得太大而消耗磁盘资源。当日志文件达到所允许的最大大小时，Communications Server for Linux 将其当前内容复制到备份文件，然后清除该日志文件。

缺省情况下，Communications Server for Linux 使用下列日志文件：

错误日志文件

/var/opt/ibm/sna/sna.err

/var/opt/ibm/sna/bak.err（备份）

审计日志文件

/var/opt/ibm/sna/sna.aud

/var/opt/ibm/sna/bak.aud（备份）

使用情况日志文件

/var/opt/ibm/sna/sna.usage

/var/opt/ibm/sna/bak.usage (备份)

可以使用文本编辑器或其他 Linux 系统实用程序来查看日志文件:

- vi** 在文本编辑器中查看文件。此编辑器允许您在文件中前后移动并允许搜索特定条目。
- pg** 每次查看文件的一页。此实用程序简单且易于使用，但仅当日志文件较小时才有用。
- tail** 查看文件尾部。文件末尾是最新的记录消息所在的地方。带 **-f** 选项使用此实用程序以在系统运行时监视日志文件。

如果已选择简明记录而不是详细记录，则可以使用 **snahelp** 命令来确定特定消息号的原因和操作信息。

对于大多数场合，缺省记录设置就足够了，但您可以进行下列类型的更改:

- 指示要记录哪些类别的消息。
 - 总是记录问题消息，不能禁用对问题消息的记录。通常对其他两类消息禁用记录，但可以根据需要启用对它们的记录。
- 指定记录消息时所用的详细信息级别。
- 指定中央记录（对于域）或本地记录（对于每个节点）
- 更改日志文件的名称和大小。

要配置记录，使用下列其中一个方法:

Motif 管理程序

从“节点”窗口或“域”窗口上的“诊断”菜单中选择“记录”。

命令行管理程序

发出下列其中一个命令:

- **set_central_logging**
- **set_global_log_type**
- **set_log_type**
- **set_log_file**

Motif 管理程序中的“记录”对话框影响整个域中的记录设置。通过使用命令行，可以通过在特定机器上配置本地日志设置来覆盖域设置。

除了提供对记录的控制外，Motif 管理程序还提供对跟踪的节点级控制。命令行界面提供对记录和跟踪功能的更高级别的控制。有关记录和跟踪的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。

第 4 章 定义连接组件

为了使 Communications Server for Linux 节点能够与其他节点通信，必须配置与至少一个相邻节点的连接。可配置连接链路来传送从属数据传输和 / 或独立数据传输。

可以将用于一个或多个链路协议的适配卡安装在计算机中。配置连接所需要输入的大部分信息取决于您正在使用的链路协议。远程节点还必须有与您选择的类型相同的适配卡，或者，本地节点和远程节点之间必须有网桥或路由器。有关 Communications Server for Linux 所支持的链路协议的列表，请参阅第 52 页的『定义 DLC、端口和连接网络』。

要配置链路，需要定义端口，如第 52 页的『定义 DLC、端口和连接网络』所述。另外（在大多数情况下），还必须配置链路站，如第 55 页的『定义链路站』所述。如果本地节点上的 LU 要使用 DLUR 来与主机通信，则还必须在本节点上定义 DLUR PU，如第 61 页的『定义 DLUR PU』所述。

使用 Motif 管理程序时，数据链路控制（DLC）自动配置为端口配置的一部分。另外，还具有将端口定义为连接网络一部分的选项。使用命令行配置时，此配置独立于端口配置。

链路配置所需要的信息取决于链路协议，您的网络是否为 APPN 网络以及链路是用于从属通信、独立通信还是用于这两者。另外，您需要配置的链路还取决于您需要支持哪种通信：

LUA 如果您打算使用 LUA，则需要配置与主机的链路。必须为从属通信配置该链路，且必须在主机及 Communications Server for Linux 节点上配置该链路，因此应咨询 SNA 网络设计者。

使用 CPI-C 或 APPC

如果您打算使用 CPI-C 或 APPC，且您的网络不是 APPN 网络，则需要配置至您想要访问的所有相邻节点的链路。必须为独立通信配置这些链路，且必须在相邻节点以及 Communications Server for Linux 节点上配置这些链路，因此可能需要咨询 SNA 网络设计者。

起 APPN 节点作用

如果 Communications Server for Linux 节点是 APPN 网络中的端节点或网络节点，则可以大大减少需要配置的链路的数目。可以配置至一个或多个相邻网络节点的链路并使用这些链路访问 APPN 网络中的所有节点。如果想要直接访问其他相邻节点，则也可以配置至它们的链路 - 这通常不是必要的，但它可提供更好的性能。如果相邻节点是通过令牌环连接到一起的，则可以动态地设置直接链路，因此不需要配置它们 - 只需确保在定义端口时将网络配置为连接网络。

独立 APPC 总是可以从 APPN 联网中受益，但除非您使用 DLUR，否则 LUA 无法从 APPN 联网中受益。（DLUR 支持在主机与本地节点上的或 APPN 网络中的下游节点上的从属 LU 之间进行通信。）仅当主机支持 DLUS 时才可以使 DLUR，因此，如果对使用 DLUR 感兴趣，则应该咨询 SNA 网络设计者。

定义 DLC、端口和连接网络

端口将通信链路的本地端表示为网络中的唯一访问点。每个端口与一个特定的链路协议关联，链路协议可以是下列任何一种协议：

- SDLC
- 令牌环
- 以太网
- X.25 或 QLLC（限定逻辑链路控制）
- 多路径通道（MPC）（仅限于 Communications Server for Linux on System z）
- 企业扩展程序（HPR/IP）

可以配置多个使用特定链路协议的端口。通常，一个端口对应于单个物理访问点（如适配卡），但某些链路协议（如令牌环）允许为单个适配器定义多个端口。不同的端口是通过地址（如 SAP 号）区分的。

当使用 Motif 管理程序来定义特定链路协议的端口时，Communications Server for Linux 自动为该端口定义一个 DLC（如果尚未定义该类型的 DLC 的话）。对于命令行配置，必须使用不同的命令来定义端口和 DLC。

在使用令牌环链路协议的 APPN 网络中，还可以使用“SAP 配置”对话框来指示端口是连接网络的一部分。

如果您在使用 SNA 网关，则可以定义一个用来生成隐式链路站（未显式配置的链路站）定义的模板。隐式链路站可以支持下游 LU。如果在端口活动时修改隐式 PU 字段，则这些更改会影响更改之后生成的任何隐式链路站实例。

要配置端口、连接网络和 DLC，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择**连接和新建端口**。

命令行管理程序

要配置 DLC：

```
define_ type_dlc
```

要配置端口：

```
define_type_port
```

在这些命令中，*type* 指示链路协议类型（sdlc、tr、ethernet、qllc、mpc 或 ip）。

要配置连接网络：

```
define_cn
```

高级端口配置参数提供对 BTU 大小、允许的活动链路数、隐式下游 LU 的生成以及动态链路站设置等的控制。

DLC、连接网络以及端口配置参数

下列参数是端口配置所必需的。（当使用 Motif 管理程序时，端口配置还提供关于 DLC 的信息，并使您能够对连接网络指定端口。）

SNA 端口名

本地识别的端口名。

适配卡号

此字段不用于企业扩展程序端口。

标识要使用的适配卡的编号（如果在此计算机中有多个相同类型的卡）。

端口号

此字段不用于企业扩展程序端口。

要使用的端口的编号（如果适配卡可以支持多个端口的话）。有效端口号的范围从 0 至该适配卡所支持的端口数减一。对该适配卡上的第一个端口，输入 0。

仅当该适配卡可支持多个端口时，此字段才适用。

最初处于活动状态

是否在启动节点时自动激活端口。此设置允许激活使用该端口的链路站以响应来自相邻节点的请求，或由本地节点按要求激活该链路站。（激活端口并不激活任何链路站；链路站是单独激活的。）

以下各节描述了特定于链路类型的其他端口参数。QLLC 不需要其他端口参数。

SDLC 的其他端口参数

线路详细信息

下列参数描述 SDLC 连接的类型：

类型 选择下列其中一个值：

租用线路

对此计算机和远程系统之间的 SDLC 链路使用专用线路。

入局交换

对入局呼叫使用标准的电话机网络。

对于非主端口（由链路角色字段指定），您还需要配置轮询地址（对于出局呼叫，该地址在链路站上配置）。该轮询地址是单字节地址（缺省情况下为 C1），它需要与在远程链路站配置的轮询地址相匹配。该端口处于活动状态时，将响应使用此轮询地址发送的帧。

对于主端口，您不需要配置轮询地址；该端口对入局呼叫使用由远程链路站指定的轮询地址。对于其他类型的端口，可在每个链路站上配置该轮询地址。

出局交换

对出局呼叫使用标准的电话机网络。

链路角色

选择一个描述在此端口上定义的链路站的本地节点角色的值。在 SDLC 通信中，一端用于管理链路，被称为主链路站。另一端是辅助链路站。

对此字段使用下列其中一个值:

- 辅助** 链路的另一端将作为控制器，并且远程系统被配置为主系统。如果您要配置与主机系统的链路，则几乎总是这种情况。
- 主** 此端口将充当链路的 SDLC 控制器，并且远程系统被配置为辅助系统。
- 可协商** 为了获得最大灵活性，此设置使两端能够协商由哪一端履行主角色。如果您不知道为远程系统配置了哪个角色，则选择此值。
- 可以对同级链路使用此设置，但要注意：当激活链路时，协商角色会导致出现短暂延迟。

主多分支

该链路被租借，并且此端口将充当至若干辅助节点的多分支链路的控制器。

当想要配置从本地节点至不同远程节点的若干链路站（例如，下游节点的链路）时，请使用此设置。其他每个节点均必须配置为辅助节点，并且您必须使用租用线路。

辅助多 PU

此本地端口为由远程系统上的端口控制的多分支链路上的其中一个辅站。

如果不知道如何配置这些参数，则咨询 SNA 网络设计者。

令牌环和以太网的其他端口参数

本地 SAP 号

SAP 的地址，对于 Intel 和 OSA2 适配器，通常为 04。仅当需要在卡上使用多个 SAP 时才使用不同的值。对于 OSA-Express 适配器，本地 SAP 号必须与在 OSA/SF 中与此 Linux 映像上的 ethX 接口对应的 I/O 设备地址定义的 SAP 号匹配。

SAP 号必须是 4 的倍数。

如果您不知道要为此字段输入什么值，请与 SNA 网络设计者联系。

定义连接网络

SAP 是否要将 LAN 作为连接网络访问。定义连接网络使连接网络上节点间的链路能够动态启动，而不必预先配置。

仅当本地节点不是低入口联网节点时此字段才适用，因为低入口联网节点不能使用连接网络。

CN 名

连接网络的名称。除非指定了定义连接网络选项来定义连接网络上的 SAP，否则不必输入 CN 名。CN 名用作虚路由节点的名称以便建立连接网络上节点间的链路。

在连接网络的所有节点上指定相同的 CN 名。

以太网类型

此字段只适用于以太网链路。

网络是标准以太网网络还是 IEEE 802.3 网络。

企业扩展程序（HPR/IP）的其他端口参数

本地 IP 接口

用于 IP 链路的本地网络适配卡的标识。如果正在端口所在的服务器上运行 Motif 管理程序，则可以从列表中选择此标识，而不是输入此标识。

定义连接网络

适配卡是否要将 LAN 作为连接网络访问。定义连接网络使连接网络上节点间的链路能够动态启动，而不必预先配置。

仅当本地节点不是低入口联网节点时此字段才适用，因为低入口联网节点不能使用连接网络。

CN 名

连接网络的名称。除非指定了定义连接网络选项来定义连接网络上的端口，否则不需要输入 CN 名。CN 名用作虚路由节点的名称以便建立连接网络上节点间的链路。

在连接网络的所有节点上指定相同的 CN 名。

隐式链路的其他端口参数

最大活动模板实例数

指定要从模板生成的最大链路站实例数。

为隐式 PU 访问配置下游 LU

是否要配置使用此 PU 的下游 LU（请参阅第 98 页的『配置 SNA 网关』）。

HPR 在隐式链路上受支持

是否支持隐式链路上的“高性能路由”。

隐式链路上的链路级别错误恢复

是否要使用链路级别错误恢复在隐式链路上发送 HPR 通信。

其他配置

在完成端口配置之后，继续执行下列配置任务：

- 要在已配置的端口上定义链路站，请参阅『定义链路站』。
- 定义 DLUR PU，请参阅第 61 页的『定义 DLUR PU』。
- 要支持 APPC 通信，请参阅第 69 页的第 6 章，『配置 APPC 通信』。

定义链路站

要与 SNA 网络中的其他节点通信，必须配置至 SNA 网络中的相邻节点的链路站（LS）的特征。在可以定义链路站之前，必须为正在使用的适配器（和链路协议）定义一个端口。无论使用什么协议，配置链路站所需要的大多数信息都是相同的。

链路站表示 Communications Server for Linux 本地节点与远程计算机之间通过 SNA 网络的逻辑路径。远程计算机可以是下列任何一个：

- 主机，Communications Server for Linux 在主机上使用 3270 或 LUA 通信来访问主机程序（或使用 APPC 或 CPI-C 来进行程序间通信）
- 对等计算机，作为对等伙伴与 Communications Server for Linux 和远程计算机通信（APPN 网络中的典型布局）

- 下游计算机，它使用 Communications Server for Linux SNA 网关功能部件或 DLUR 功能部件来访问主机

链路站与特定端口关联；可在每个端口上定义一个或多个链路站。

支持从属通信的每个链路站都有一个关联的 PU（物理部件）。因为 PU 与链路站相关联，所以 Communications Server for Linux 不将它们视为独立资源；它们被配置为链路站配置的一部分，并随着链路站的启动与停止而启动与停止。链路站显示在“节点”窗口的连接部分中；PU 则不在任何窗口中显示。

注：大多数情况下，需要将链路站添加至端口。然而，如果只想将动态创建的链路站用于下游 SNA 网关或 APPC 通信，则对于链路总是由远程节点激活的情况，不必显式配置链路站。

如果远程节点尝试与本地节点连接，但未定义与入局呼叫所指定的地址相匹配的链路站，则当本地节点上已定义了合适的端口时，Communications Server for Linux 可以隐式地定义链路站。在连接的持续时间中，这个动态创建的链路站出现在“节点”窗口的连接部分中。

要配置链路站，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的**服务**菜单中选择**连接**和**新建链路站**。

命令行管理程序

发出以下命令：

```
define_type_ls
```

在此命令中，*type* 指示链路协议类型（sdlc、tr、ethernet、qllc、mpc 或 ip）。

链路站的高级参数提供了对传输特征、XID 交换、可选的链路设施、使用该链路的 LU 0-3 会话的压缩以及重新激活过程的其他控制。

链路站配置参数

在 Motif 中，“链路站配置”对话框包含下列各个部分，每个部分包含不同类别的配置参数：

链路站

使用此对话框区域来提供所有链路站都需要的信息，而不管它们是支持从属 LU 数据传输、独立 LU 数据传输还是这两者都支持。要获取此部分中参数的说明，请参阅第 57 页的『公共链路站参数』。

独立 LU 数据传输

仅当正在使用独立通信的链路站时才提供此信息。要获取此部分中参数的说明，请参阅第 59 页的『独立 LU 数据传输的参数』。

从属 LU 数据传输

仅当正在使用从属通信的链路站时才提供此信息。要获取此部分中参数的说明，请参阅第 60 页的『从属 LU 数据传输的参数』。

公共链路站参数

下列参数是所有链路站所必需的，而不管它们是支持从属通信、独立通信还是这两者都支持。

有关此对话框上的参数的更多信息，请参阅联机帮助或 *Communications Server for Linux Administration Command Reference*。

名称 在本地标识链路站的名称。

SNA 端口名

用来访问相邻节点的端口。

激活 用来激活链路站的方法。指定下列其中一个方法：

由管理员

仅当本地系统管理员请求时才激活链路站。

在节点启动时

在节点启动时链路站自动启动。

按要求 链路站在需要为应用程序提供连接时自动启动。

链路站是从端口单独激活的，因此，即使端口已活动，仍必须激活链路站。激活端口本身并不会激活任何链路站，并且，将端口配置为最初处于活动状态也并非表示它的任何链路站会在节点启动时自动激活。然而，激活端口确实使激活链路站成为可能。除非本地节点和相邻节点上的端口都是活动的，否则不能激活链路站。

如果链路是要收取使用费的，则应避免不必要地激活链路，以降低成本。

如果不知道如何设置此字段，则咨询 SNA 网络设计者。

LU 数据传输

流经链路的 LU 数据传输的类型。此选项确定链路定义所需要的其他参数。

此参数不用于企业扩展程序（HPR/IP）链路，这是因为此链路类型只支持独立通信。

任何 链路站可用于独立和从属 LU 数据传输。对于此选项，除了本节所述的那些字段以外，还必须为第 59 页的『独立 LU 数据传输的参数』和第 60 页的『从属 LU 数据传输的参数』中描述的字段提供值。

仅独立 链路站仅可以用于独立 LU 数据传输。对于此选项，除了本节中所述的那些字段以外，还必须为第 59 页的『独立 LU 数据传输的参数』中描述的字段提供值。

仅从属 链路站仅可以用于从属 LU 数据传输。对于此选项，除了本节中所述的那些字段以外，还必须为第 60 页的『从属 LU 数据传输的参数』中描述的字段提供值。

您还需要提供关于与相邻节点联系的寻址信息。需要的寻址信息的类型取决于端口的 DLC 类型。如果您未提供远程节点的地址，则链路站充当非选择性侦听链路站，接受来自任何远程节点的入局呼叫。

SDLC 的其他链路站参数：

轮询地址

远程站的轮询地址。将该地址指定为两位数（一个字节）十六进制值，通常以

C1 开头。主链路站使用此值来轮询远程站。辅助链路站使用此值来响应轮询。视链路角色而定，将以不同方式输入轮询地址：

- 如果链路是点到点链路（而不是多分支链路），则通常使用地址 C1。
- 如果此链路的父端口是入局交换，则在该端口上配置轮询地址，并且不能为每个链路站独立配置该轮询地址。
- 如果您要配置主出局交换链路站，并且不知道想要与之通信的远程辅助链路站的轮询地址，则可以对主链路站指定轮询地址 0xFF。此值使节点能够接受来自辅助链路站的响应，而不管它所配置的轮询地址是什么。对于非主链路或不是出局交换的链路，0xFF 不是有效的地址。
- 如果您要使用多分支配置，则与同一主链路站通信的所有辅助链路站都必须具有不同的轮询地址。

链路两端的轮询地址必须匹配。如果不知道在远程系统上配置的地址，请与 SNA 网络设计者联系。

在 VTAM 主机上，轮询地址在 VTAM PU 定义中被配置为 ADDR= 参数。

在 AS/400 系统上，轮询地址是线路描述的 STNADR 参数。

令牌环和以太网的其他链路站参数:

MAC 地址

作为一系列十六进制位输入的远程站 MAC 地址。MAC 地址唯一地标识远程系统上的适配卡。

如果不知道要使用什么值，则咨询 SNA 网络设计者。

如果此链路的远程端是 VTAM 主机，则可以在“VTAM 端口”定义的 MACADDR= 参数中找到远程端的 MAC 地址。

如果正在配置至 AS/400 系统的链路，则 MAC 地址是线路描述中的 ADPTADR 参数。

SAP 号

远程计算机上端口的 SAP 号。SAP 号将使用同一适配卡的不同链路区别开。这是一个十六进制数，通常为 04。它必须是 4 的倍数。

如果不知道要使用什么值，则咨询 SNA 网络设计者。

如果此链路的远程端是 VTAM 主机，则 SAP 号是 VTAM PU 定义的 SAPADDR= 参数。

如果正在配置至 AS/400 系统的链路，则 SAP 号是线路描述中的 SSAP 参数。

X.25 (QLLC) 的其他链路站参数:

远程 X.25 地址

如果该链路为交换虚电路，则将远程 DTE 的 DTE 地址作为一系列十六进制数字输入。

如果该链路为永久虚电路，则通过 **snaadmin** 使用参数 `vc_type=PVC` 并将 `pvc_id` 设置为通道标识来配置它。

MPC 的其他链路站参数:

MPC 组

MPC 驱动程序配置中指定的用于标识特定通道的 MPC（多路径通道）组名称。

企业扩展程序 (HPR/IP) 的其他链路站参数:**远程 IP 主机名**

此链路的目标节点的远程主机名。可以将此名称指定为点分十进制 IP 地址（如 193.1.11.100）、名称（如 newbox.this.co.uk）或别名（如 newbox）。如果您指定名称或别名，则 Linux 系统必须能够（通过使用本地 TCP/IP 配置或使用域名服务器）将它解析为标准名称。

独立 LU 数据传输的参数

需要下列信息来配置此链路站以供独立 LU（供 APPC、5250 或 CPI-C 应用程序使用的 6.2 型 LU）使用:

远程节点名

远程节点的标准 CP 名称。

如果远程系统是 VTAM 主机，则可以在 VTAM 启动列表的 *NETID* 参数中找到网络名（标准名称的前八个字符）。后八个字符位于 VTAM 启动列表的 *SSCPNAME* 参数中。

注: 如果输入新远程节点的名称，则可以添加远程节点的定义以使您能够在新远程节点上定义伙伴 LU。如果本地节点为低入口联网节点，则不必指定远程节点类型，远程节点类型字段不适用。）

要以此方式定义新的远程节点，将远程节点类型指定为除“发现”之外的值，并指定远程节点名。

另外，可以指定“发现”并保留远程节点名为空白。这意味着任何相邻节点都可以使用该链路站。如果本地节点为低入口联网节点，则“发现”选项不可用。

远程节点类型

通过此链路站访问的远程节点上的 APPN 支持的级别（仅当本地节点为端节点或网络节点时才适用）。

如果不知道远程节点是低入口联网节点还是端节点，或者是否为网络节点，则可以选择“发现”。发现远程节点上 APPN 支持的级别会稍微延迟链路激活，所以如果知道类型，最好指定它。这也有助于确保网络配置的一致性。

如果链路站是按要求激活的，则不能选择“发现”。

如果本地节点为低入口联网节点，则此字段不适用。

分支链路类型

至通过此链路站访问的远程节点的链路的类型（仅当本地节点为分支网络节点时才适用）。

如果远程节点是 APPN 主干网络中的网络节点，则选择上行链路（至主干网络）。如果远程节点是分支网络中的端节点，则选择下行链路（在分支网络中）。

如果远程节点要配置为网络节点，则分支链路类型自动设置为上行链路（至主干网络），并且您不能更改它。

从属 LU 数据传输的参数

由于企业扩展程序 (HPR/IP) 链路只支持独立通信, 所以这些参数不适用于该链路类型。

为从属 LU 数据传输配置链路站会自动以与该链路站相同的名称创建适当的 PU。

需要下列信息来配置链路站以供从属 LU (用于 3270 或 LUA 应用程序的 0-3 型 LU) 使用:

本地节点标识

标识 SNA 网络中本地节点的值。

通常可以将同一节点标识 (缺省值) 用于同一节点上的所有链路。然而, 如果您需要 255 个以上的从属 LU 来访问特定的主机, 则必须配置多个至该主机的链路站, 每个链路站最多具有 255 个从属 LU, 且每个链路站具有不同的本地节点标识。

要确保将远程节点配置为能够识别本地节点标识, 请与 SNA 网络设计者联系。

在 VTAM 配置中, 前三位数字应该与 PU 定义中的 *IDBLK* 参数相匹配, 后五位应该与 *IDNUM* 参数相匹配。

在 AS/400 系统上, 节点标识是在 *EXCHID* 参数中配置的。

远程节点标识

远程链路站的节点标识 (可选; 仅当需要限制对此链路站的访问时才适用)。如果指定远程节点标识, 则仅当远程节点的节点标识与此定义中指定的值匹配时才激活链路。当在交换端口上配置了若干个链路站时这一点非常有用, 因为这使得远程节点在激活这些链路站时能够将它们区分开。链路站也可以通过远程节点的 CP 名称来区分, 但对于在激活链路时不发送其 CP 名称的远程节点, 则必须使用远程节点标识来代替。

如果不指定远程节点标识, 则在激活链路时不检查远程节点的节点标识。

远程节点角色

远程 (相邻) 节点的角色:

主机 链路站支持用于与主机的会话的从属 LU (如 3270 LU) (最普通的情况)。如果链路所至的节点使用 SNA 网关或 DLUR 来提供主机连接, 则仍应该将相邻节点的角色设置为“主机”, 即使该链路不是直接至主机的。

下游 (SNA 网关)

链路站连接至的节点是将使用本地节点的 SNA 网关能力与主机通信的下游节点 (对主机来说, 下游节点上的 LU 似乎是驻留在本地节点上)。

下游 (DLUR)

链路站连接至的节点是将使用本地节点的 DLUR 能力与主机通信的下游节点。(对主机来说, 下游节点上的 LU 似乎是驻留在本地节点上)。

仅当本地节点是 APPN 网络节点时才能使用这样的链路。

下游 PU 名

与下游节点相关联的 PU 名。此值必须与主机上为下游节点配置的 PU 名匹配。如果不知道用于此名称的值, 则咨询 SNA 网络设计者。

仅当指定了此链路站是连接至将使用本地节点的 DLUR 能力与主机通信的下游 PU 时，此字段才适用。可以通过为远程节点角色字段指定下游 (DLUR) 来指示此字段。

有关更多信息，请参阅『定义 DLUR PU』。

上游 DLUS 名

支持 DLUS 的主机 LU 的标准 LU 名称 (下游 PU 要访问的 LU 服务器)。如果不知道用于此名称的值，则咨询 SNA 网络设计者。

仅当指定了此链路站是连接至将使用本地节点的 DLUR 能力与主机通信的下游 PU 时，此字段才适用。可以通过为远程节点角色字段指定下游 (DLUR) 来指示此字段。

其他配置

在执行链路站配置之后，继续执行下列配置任务：

- 定义 DLUR PU，请参阅『定义 DLUR PU』。
- 要配置传递服务，请参阅第 91 页的第 8 章，『配置传递服务』。
- 要支持特定用户应用程序，请参阅第 89 页的第 7 章，『配置用户应用程序』。
- 要支持 APPC 通信，请参阅第 69 页的第 6 章，『配置 APPC 通信』。

定义 DLUR PU

通常，从属 LU 会话需要一条至主机的直接通信链路。若 APPN 网络中许多节点 (包括主机节点) 连接在一起，则其中有些节点可能不直接连接至主机，而是只通过另一个节点间接地连接至主机。从这些间接连接的节点中的 LU 建立至主机的从属 LU 会话是不可能的。

从属 LU 请求者 (DLUR) 是为克服此限制而设计的 APPN 功能部件。

本节说明如何配置提供与主机的连接的 DLUR PU。配置 DLUR PU 使本地节点能够提供 DLUR 服务。

APPN 节点 (如运行 Communications Server for Linux 的节点) 上的 DLUR 与主机上的从属 LU 服务器 (DLUS) 配合工作，以将会话从 DLUR 节点上的从属 LU 通过 APPN 网络路由至 DLUS 主机。至主机的路由可以跨多个节点，并且可以利用 APPN 的网络管理、动态资源定位以及路由计算等设施。DLUR 在这些 LU 所在的节点上必须是可用的，并且 DLUS 在主机节点上必须是可用的，但会话路由中的任何中间节点上不需要 DLUR。

如果 Communications Server for Linux DLUR 节点是网络节点，则它还可以为与 Communications Server for Linux 节点连接的下游计算机上的从属 LU 提供传递 DLUR 设施。这些 LU 可以使用 Communications Server for Linux 节点上的 DLUR 以节点内部的 LU 所用的同一方式在网络上访问主机。

要对下游节点提供传递 DLUR 服务，必须首先配置 (在本地节点上) 与下游节点关联的 PU 名。此值必须与主机上为下游节点配置的 PU 名匹配。

要配置 DLUR PU，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择连接和新建 DLUR PU。

命令行管理程序

发出以下命令：

```
define_internal_pu
```

DLUR PU 配置参数

下列参数是 DLUR PU 配置所必需的：

PU 名

为本地节点上的每个 DLUR PU 各指定一个 PU 名。此名称应该与主机上配置的 PU 名相匹配。（Communications Server for Linux 将 PU 名和 PU 标识都发送至主机以标识该 PU。通常，主机通过 PU 的 PU 名来标识该 PU，如果找不到匹配的 PU 名，就通过 PU 标识来标识该 PU。）

DLUS 名

支持 DLUS 的主机 LU 的标准 LU 名称。

为了使用 DLUR，Communications Server for Linux 的 DLUR 组件必须建立与主机上的 DLUS 的 LU-LU 会话。

与 SNA 网络设计者联系以确定主机 LU 的名称。

备份 DLUS 名

此参数是可选的。当 DLUS 名指定的名称不可用时，可以使用该备份主机 LU 的标准 LU 名称。

与 SNA 网络设计者联系以确定主机 LU 的名称。

PU 标识

支持与主机连接的本地节点上的 PU 的 PU 标识。PU 标识由两个十六进制字符串组成，一个三位（称为块号），另一个五位。

每个从属 LU 都与一个 PU 关联。PU 和 LU 都在主机上配置。对于每个 PU，您需要在 Communications Server for Linux 节点上定义 DLUR PU。该 PU 标识必须与主机上为此 PU 配置的 PU 标识匹配。

在许多情况下，PU 标识与节点标识相同，所以节点标识是缺省值。然而，如果您需要 255 个以上的从属 LU 来访问特定的主机，则需要配置多个 DLUR PU，每个 DLUR PU 最多具有 255 个从属 LU，并且每个 DLUR PU 具有不同的 PU 标识。

如果不知道如何设置此字段，则咨询 SNA 网络设计者。

在 VTAM 配置中，前三位应该与 PU 定义中的 *IDBLK* 参数匹配，后五位应该与 *IDNUM* 设置匹配。

最初处于活动状态

是否在启动节点时自动激活 DLUR PU。如果不设置此选项，则必须人工启动 DLUR PU。

支持压缩

对于使用此 PU 的 LU 0-3 会话，是否支持数据压缩。如果设置此选项，则将在主机请求压缩时使用压缩。如果未设置此选项，则不使用压缩。

无限地重试联系 DLUS

在 Communications Server for Linux 第一次尝试联系 DLUS 失败后是否进行重试。如果您设置此选项，则 Communications Server for Linux 在第一次尝试失败后将无限地进行重试。如果未设置此选项，则它将只重试一次。

下游节点的传递 DLUR 的参数

您需要下列信息才能将 Communications Server for Linux 配置为使用传递 DLUR 来在下游节点上的从属 LU 和主机之间传输通信：

下游 PU 名

与下游节点相关联的 PU 名。PU 名必须与在主机上配置的 PU 名匹配。

一个下游节点可以支持多个 PU。在这种情况下，每个下游 PU 都与不同的链路相关联，所以需要在 Communications Server for Linux DLUR 节点和下游节点之间配置多个链路，并且需要知道每个链路的下游 PU 名称。

请咨询 SNA 网络设计者以了解与下游节点相关联的 PU 名。

DLUS 名

支持 DLUS 的主机 LU 的标准 LU 名称。为了使用 DLUR，Communications Server for Linux 的 DLUR 组件必须建立与主机上的 DLUS 的 LU-LU 会话。

请咨询 SNA 网络设计者以了解主机 DLUS 服务器的 LU 名。

其他配置

在配置 DLUR 之后，继续执行下列配置任务：

- 要为 DLUR 配置 LU，请参阅第 99 页的『配置 DLUR』。
- 要配置其他传递服务，请参阅第 91 页的第 8 章，『配置传递服务』。
- 要支持特定用户应用程序，请参阅第 89 页的第 7 章，『配置用户应用程序』。
- 要支持 APPC 通信，请参阅第 69 页的第 6 章，『配置 APPC 通信』。

定义 DLUR PU

第 5 章 配置从属 LU

本章提供有关配置 LU 和 LU 池以支持使用 3270、TN3270 和 LUA 通信的用户应用程序的指示信息。要使用这些通信，必须配置从属 LU。

在可以配置本章中描述的资源之前，必须执行下列配置：

- 按第 48 页的『配置节点』所述配置节点。
- 按第 51 页的第 4 章，『定义连接组件』所述配置连接。对于 3270、TN3270 以及 LUA，必须将链路配置为支持从属 LU 数据传输。

如果正在使用上游 SNA 网关或 DLUR，则不需要配置至主机的直接链路。有关更多信息，请参阅第 98 页的『配置 SNA 网关』和第 61 页的『定义 DLUR PU』。

定义 0-3 型 LU

必须配置 0-3 型的从属 LU 以支持与主机系统进行通信。可以使用本节中的信息来定义支持 3270 或 LUA 的 LU。还可以定义 LU 的范围，以在单个操作中配置多个相同类型的 LU。

要配置 0-3 型 LU，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的**服务**菜单中选择下列其中一项。

- **3270 和新建 3270 显示器 LU 或新建 3270 打印机 LU**
- **LUA 和新建 LUA LU**
- **TN 服务器和新建主机 LU**

命令行管理程序

发出下列其中一个命令：

```
define_lu_0_to_3
```

```
define_lu_0_to_3_range
```

可以使用高级对话框来限制对特定 SSCP 的访问或指定不活动超时。

0-3 型 LU 配置参数

下列参数是 0-3 型 LU 配置所必需的：

LU 名 1-8 个字符的 LU 名（表示单个 LU）或 1-5 个字符的基本名称（表示一系列的 LU，将前缀添加至基本名称来形成已定义的 LU 的所有名称）。

该 LU 名仅在本地使用；它不必与主机上定义的名称相对应。

主机 LS/DLUR PU

提供至主机的链路的链路站。该 LU 定义属于您选择的链路站。（如果从属 LU 驻留在支持 DLUR 的节点上，则此字段标识提供与主机的连接的 DLUR PU。）

定义 0-3 型 LU

LU 号

LU 号或 LU 号的范围。LU 号可以在 1-255 之间。

这些 LU 号必须对应于主机 VTAM 配置中的那些 LU 号。如果不知道在主机上配置了什么号，则咨询 SNA 网络设计者。

LU 类型

下列其中一种 LU 类型（取决于您正在配置的 LU 的类型）：

- 如果要将此 LU 与主机上的 TN 服务器和 DDDL U 配合使用，则不能在主机上配置该 LU。这种情况下，此处指定的 LU 类型用来动态定义主机上的 LU。如果想要将 LU 模型类型定义为与下游 TN3270 客户机请求的类型相匹配，则指定未限制（对于命令行配置，指定未知）。
- 对于 3270 显示器 LU，根据屏幕大小指定适当的型号：
 - 3270 型号 2 (80x24)
 - 3270 型号 3 (80x32)
 - 3270 型号 4 (80x43)
 - 3270 型号 5 (132x27)
- 对于打印机 LU，指定下列其中一项：
 - 3270 打印机
 - SCS 打印机
- 如果不了解 LU 类型、如果 LU 用来支持从本地节点到主机（上游 LU）的 SNA 网关或者如果 LU 是用于 LUA 应用程序的，则指定未限制（对于命令行配置，指定未知）。

LU 类型应与主机上的 LU 的配置匹配。如有必要，优先选用主机上配置的 LU 类型。

池中的 LU

是否将 LU 指定给 LU 池。

注：如果将此 LU 指定给 LU 池，并将用户的会话指定给此 LU，则用户的会话将使用此 LU（如果它可用的话）；否则，它将使用池中的任何空闲 LU，就像将其指定给 LU 池而不是特定的 LU 一样。如果您想要用户仅使用指定的 LU，以便该 LU 已在使用中时不能建立用户的会话，则确保该 LU 不在池中。

池名 LU 池的名称。

其他配置

在执行 0-3 型 LU 配置之后，继续执行下列配置任务：

- 要对 3270 显示器、TN3270 或 LUA 使用从属 LU 池，请按第 67 页的『定义 LU 池』所述定义 LU 池。
- 对于 TN3270，请按第 91 页的『配置 TN 服务器』所述定义 TN3270 客户机访问记录。

定义 LU 池

对于 3270、TN3270 和 LUA，可以定义 LU 池来简化用户配置并在建立主机会话时提供更大的灵活性。

注：可将用户的会话指定给特定 LU 或 LU 池。

- 如果您将用户的会话指定给池中的特定 LU，则在此 LU 可用时会话使用它；否则会话使用池中的任何可用 LU，就好像您已将会话指定给 LU 池而不是特定 LU。
- 如果您想要用户仅使用指定的 LU，以便该 LU 已在使用中时不能建立用户的会话，则确保该 LU 不在池中。

LU 池甚至可以跨多台 Communications Server for Linux 服务器 - 只需在不同服务器上定义同名的 LU 池。如果某台服务器发生故障或停止提供服务，则使用 LU 池的客户机可以使用另一服务器。使用 LU 池还简化了客户机配置，并且，通过添加另一服务器或通过现有服务器上添加 LU，还可以很方便地提高容量。

可使用“LU 池”窗口来查看 Communications Server for Linux 域的所有 LU 池。此窗口列示系统中配置的 LU 池，并使您能够选择要添加至 LU 池的 LU。LU 池中的各个 LU 列示在 LU 池之下。

标识 LU 如下：

- 3270 显示器 LU
- 未限制的 LU
- SCS 打印机
- 3270 打印机

不要在同一池中混合不同类型的 LU（例如，不要把显示器和打印机 LU 置于同一池中）。除非您正在支持 TN3270E 客户机，否则您未必需要打印机 LU 的池。

要配置 LU 池，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的窗口菜单中选择 **LU 池**，然后选择**新建**以添加池。

命令行管理程序

发出以下命令：

```
define_lu_pool
```

LU 池配置参数

下列参数是 LU 池配置所必需的：

名称 标识 LU 池的名称。仅当您正在添加一个新的 LU 池时，此字段才适用。不能更改现有池的名称。

指定的 LU

要指定给池的 LU。一个 LU 只能是一个池的成员。

定义 LU 池

第 6 章 配置 APPC 通信

APPC 应用程序、5250 仿真程序以及 CPI-C 应用程序都需要您首先配置 APPC。APPC 应用程序使用节点的 6.2 型 LU 资源来与主机或对等计算机上的另一个 APPC 或 CPI-C 应用程序通信（使用指定的方式）。

如果应用程序使用 CPI-C，则在配置 APPC 之后您可能需要执行其他的 CPI-C 配置。CPI-C 应用程序使用节点的 6.2 型 LU 和方式资源与主机或对等计算机上的另一个 APPC 或 CPI-C 应用程序通信。为 CPI-C 应用程序定义与 APPC 应用程序相同的资源。另外，如果 Communications Server for Linux 计算机上的 TP 是调用 TP（启动对话的 TP），则您可能需要为其定义一个或多个辅助信息条目，如第 84 页的『定义 CPI-C 辅助信息』中所述。这些项的每一个都提供了关于伙伴 TP 和用来访问该伙伴 TP 的 LU 和方式资源的信息以及任何必需的安全性信息。

APPC 的配置步骤取决于 LU 6.2 通信是从属的还是独立的。除非远程节点是主机，否则必须使用独立通信。如果远程节点是主机，则可以使用从属通信或独立通信。

必须先执行下列配置，然后才能配置 APPC 通信：

- 按第 48 页的『配置节点』所述配置节点。
- 按第 51 页的第 4 章，『定义连接组件』所述配置连接。

注：在 APPN 网络中，至相邻网络节点的单个链路站可用来与该网络中的任何远程节点通信，所以不需要配置至每个远程节点的独立链路站。

在许多情况下，APPC 应用程序可使用本地节点和远程节点上的控制点 LU 和标准方式。在此情况下，您的配置就为 APPC 准备好了，不需要进行任何进一步配置。

下列步骤可用来配置本地节点上的 APPC 通信。根据本地节点和远程节点的类型以及您的应用程序，您可能不需要执行这些步骤。

1. 定义本地 LU，如『定义本地 LU』所述。
2. 定义远程节点，如第 71 页的『定义远程节点』所述。
3. 定义伙伴 LU，如第 72 页的『定义伙伴 LU』所述。
4. 定义可调用 TP，如第 75 页的『定义 TP』所述。
5. 定义方式，如第 80 页的『定义方式和服务等级』所述。
6. 定义 CPI-C 辅助信息，如第 84 页的『定义 CPI-C 辅助信息』所述。
7. 定义 APPC 安全性，如第 86 页的『配置 APPC 安全性』所述。
8. 要配置 5250 通信，请参阅第 89 页的第 7 章，『配置用户应用程序』。

定义本地 LU

在很多情况下，应用程序可以使用本地节点的控制点 LU，该 LU 是在您配置节点时自动定义的。这是缺省 LU - 若应用程序未指定特定的 LU，则它可以使用此 LU。若应用程序使用缺省 LU，则您不需要定义本地 LU。请检查 APPC 应用程序的文档，或与应用程序员联系。

若正在配置供 APPC 或 CPI-C 应用程序使用的 6.2 型从属 LU，则可能想要将它们定义为缺省池的成员。从定义为缺省 LU 的 LU 池中为未指定特定本地 LU 的应用程序指定一个未使用的 LU。

可以将从属 LU 6.2 定义为缺省 LU（并可以在多个节点上定义缺省 LU）。可以将请求缺省 LU 的应用程序指定给任何这些 LU（如果它们可用的话）；LU 不必与应用程序位于同一计算机上。但是，如果正在为应用程序定义伙伴 LU，则必须在所有定义了缺省 LU 的节点上定义该伙伴 LU，因此，应用程序可以使用任何节点上定义的任何缺省本地 LU 来与正确的伙伴 LU 联系。

独立 APPC 和 5250 使用独立 LU。每个 LU-LU 会话涉及一个本地 LU 和一个伙伴 LU。对于本地 LU，可以使用预定义的与节点控制点关联的缺省 LU，或者可以配置新的本地 LU。如果 Communications Server for Linux 节点在 APPN 网络中是端节点或网络节点，则根本不需要配置伙伴 LU，因为 APPN 可动态地找到伙伴 LU。然而，如果您的网络不是 APPN 网络或者如果节点是低入口联网节点，则必须配置伙伴 LU。在这种情况下，必须配置伙伴 LU 所驻留的远程节点，然后在远程节点上定义该伙伴 LU。（如果伙伴 LU 是远程节点上的缺省 LU，则不需要显式地定义它，这是因为定义远程节点时会自动添加该 LU。）

要配置 APPC 本地 LU，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC 和新建独立本地 LU 或新建从属本地 LU**。

命令行管理程序

发出以下命令：

```
define_local_lu
```

可以使用高级对话框来指定同步点支持、连接路由特征、是否应该使用密码替换、对 SSCP 访问的限制以及安全性。

本地 LU 配置参数

下列参数是本地 LU 配置所必需的：

LU 名 本地 LU 的 LU 名。

如果不知道要使用什么名称，则咨询 SNA 网络设计者。

此 LU 名是本地 LU 的标准 LU 名称的第二部分。标准 LU 名称的第一部分（网络名）总是与本地节点的 CP 名称的第一部分相同。

LU 别名

LU 的 LU 别名。如果未输入别名，则将 LU 名用作别名。

主机 LS/DLUR PU

LU 所属的主机链路站或 DLUR PU 的名称。（仅当 LU 为从属 LU 时此字段才适用。）

LU 号

从属 LU 的 LU 号。（仅当 LU 为从属 LU 时此字段才适用。）

缺省池的成员

是否要使 LU 成为缺省从属 APPC LU 池的成员。对于未指定要使用的特定本地 LU 的应用程序，将为那些应用程序指定缺省池中的可用 LU。

仅当 LU 为从属 LU 时此字段才适用。

其他配置

在执行本地 LU 配置之后，继续执行下列配置任务：

- 要定义远程节点，请参阅『定义远程节点』。
- 要定义伙伴 LU，请参阅第 72 页的『定义伙伴 LU』。
- 要定义可调用 TP，请参阅第 75 页的『定义 TP』。
- 要定义方式，请参阅第 80 页的『定义方式和服务等级』。
- 要定义 CPI-C 辅助信息，请参阅第 84 页的『定义 CPI-C 辅助信息』。
- 要定义 APPC 安全性，请参阅第 86 页的『配置 APPC 安全性』。
- 要配置 5250 通信，请参阅第 89 页的第 7 章，『配置用户应用程序』。

定义远程节点

在下列情况下必须定义远程节点（以及该节点上的伙伴 LU）：

- 若本地节点为低入口联网节点，则必须定义所有远程节点以及使用 APPC 与该低入口联网节点通信的远程节点上的任何伙伴 LU。低入口联网节点不能动态找到伙伴 LU；而远程节点定义则使低入口联网节点能够动态找到伙伴 LU。
- 若远程节点是低入口联网节点而本地节点是充当低入口联网节点的网络节点服务器的网络节点，则必须将该低入口联网节点（及其伙伴 LU）定义为网络节点服务器上的远程节点。此定义使 APPN 网络其余部分的节点能够找到低入口联网节点上的 LU。
- 若远程节点在一个不同的 APPN 网络中，则必须定义该远程节点，因为它不能被动态找到。

如果需要定义远程节点但在定义链路站时未定义远程节点，则在可以通过链路使用 APPC 通信之前必须定义远程节点。

当添加远程节点定义时，自动添加与远程节点有相同名称的伙伴 LU；这是远程节点的控制点 LU。若您的应用程序使用此伙伴 LU，则不需要添加另一伙伴 LU，虽然您可能想要为该伙伴 LU 添加 LU 别名。要添加别名，双击该伙伴 LU 并在“伙伴 LU 配置”对话框中输入别名。

如果本地节点和远程节点都是端节点或网络节点并且是 APPN 网络的一部分，则根据需动态找到伙伴 LU。在此情况下，不要定义 LU 所在的远程节点，这是因为定义该节点可能会导致 APPN 中用于动态找到 LU 的协议发生故障。

为了防止发生此故障，Communications Server for Linux 不允许定义与它有活动 CP-CP 会话（或过去已经与它有 CP-CP 会话）的远程节点。另外，如果先前已定义了远程节点，并且 Communications Server for Linux 建立了与该远程节点的 CP-CP 会话，则该条目被临时转换为动态条目。应该在节点不活动时通过删除远程节点定义来排除故障。

要配置远程节点，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC** 和新建远程节点。

命令行管理程序

要定义远程节点，发出以下命令：

```
define_directory_entry
```

要定义伙伴 LU，发出以下命令：

```
define_partner_lu
```

远程节点配置参数

下列参数是远程节点配置所必需的：

节点的 *SNA* 网络名

远程节点的标准 CP 名称。在此对话框上输入的值必须与该远程节点上配置的 CP 名称匹配。

其他配置

在执行远程节点配置之后，继续执行下列配置任务：

- 要定义伙伴 LU，请参阅『定义伙伴 LU』。
- 要定义可调用 TP，请参阅第 75 页的『定义 TP』。
- 要定义方式，请参阅第 80 页的『定义方式和服务等级』。
- 要定义 CPI-C 辅助信息，请参阅第 84 页的『定义 CPI-C 辅助信息』。
- 要定义 APPC 安全性，请参阅第 86 页的『配置 APPC 安全性』。
- 要配置 5250 通信，请参阅第 89 页的第 7 章，『配置用户应用程序』。

定义伙伴 LU

如果本地节点和远程节点都是网络节点，或者一个是网络节点而另一个是端节点，并且应用程序使用 LU 名来引用伙伴 LU，则不需要定义伙伴 LU，这是因为应用程序可以使用 APPN 来动态地找到伙伴 LU。然而，如果应用程序使用 LU 别名来引用它的伙伴 LU，则应该添加伙伴 LU 别名定义。

若本地节点或远程节点是低入口联网节点，则必须将该伙伴 LU 定义为远程节点的子代，因为低入口联网节点不能参与对 LU 的动态定位。若您的应用程序使用远程节点的控制点 LU 作为其伙伴 LU，则该控制点 LU 是在您定义远程节点时自动定义的。

可以使用通配符来配置全都位于同一远程节点上并且名称以相同字符开头的多个伙伴 LU。使用通配符意味着不需要分别配置每个伙伴 LU。

要配置伙伴 LU，使用下列其中一个方法：

Motif 管理程序

可以使用 Motif 管理程序来添加伙伴 LU 别名、添加特定远程节点上的伙伴 LU 的定义或者使用通配符来定义多个伙伴 LU。从“节点”窗口上的服务菜单中选择 **APPC**、**新建伙伴 LU** 以及下列其中一项。

- 伙伴 LU 别名

- 远程节点上的伙伴 LU
- 远程节点上的通配符伙伴 LU

命令行管理程序

要定义伙伴 LU，发出以下命令：

```
define_partner_lu
```

要将低入口联网节点定义为伙伴 LU，发出以下命令：

```
define_adjacent_len_node
```

```
define_directory_entry
```

伙伴 LU 配置参数

下列参数是伙伴 LU 配置所必需的：

伙伴 LU 名

伙伴 LU 的标准 LU 名称。此名称必须与远程节点上为此 LU 配置的名称匹配。如果不知道那个名称，则咨询 SNA 网络设计者。

在定义特定远程节点上的伙伴 LU 或定义伙伴 LU 别名时，此字段适用。

通配符伙伴 LU 名

与多个伙伴 LU 的标准 LU 名称匹配的名称。（仅当使用通配符来定义伙伴 LU 时此字段才适用。）通配符伙伴 LU 名由两个字符串组成，每个的长度都是 1-8 个字符：

- 第一个字符串可以是与标准伙伴 LU 名称的第一部分完全匹配的完整 SNA 网络名，或者是与这些伙伴 LU 的网络名的开头部分匹配的通配符前缀。如果提供通配符前缀来作为第一个字符串的值，则保留第二个字符串为空白。例如，通配符条目 **A** 将与 SNA 网络中名为 A、ANT 或 APPN（而不是 BUFFALO 或 ZEBRA）的所有 LU 匹配。
- 若为第一个字符串提供完整的 SNA 网络名，则也可以为第二个字符串输入一个值。（如果没有为第一个字符串提供有效的 SNA 网络名，则不能指定第二个字符串。）第二个字符串被当作通配符前缀对待，它必须匹配标准伙伴 LU 名称的第二部分的开头部分。例如，通配符条目 **A.F** 与伙伴 LU 名 A.FRED 或 A.FREDDY（而不是 APPN.FRED 或 A.B）匹配。

如果将两个字符串都保留为空白，则通配符伙伴 LU 定义与任何伙伴 LU 名匹配。

别名 可在本地显示的伙伴 LU 别名。如果没有本地应用程序使用 LU 别名来引用伙伴 LU，则不必指定 LU 别名。

在定义特定远程节点上的伙伴 LU 或定义伙伴 LU 别名时，此字段适用。

未解释名称

当从属本地 LU 请求主机在伙伴 LU 与本地 LU 之间启动 LU-LU 会话时使用的未解释名称。此名称使本地配置的（并由应用程序使用的）伙伴 LU 名能够与主机上配置的伙伴 LU 名不同。

缺省未解释名称是伙伴 LU 名的第二部分。这在大多数情况下是正确的。如果有疑问，请咨询 SNA 网络设计者。

在定义特定远程节点上的伙伴 LU 或定义伙伴 LU 别名时，此字段适用。

支持并行会话

伙伴 LU 是否可以同时支持多个会话。大多数情况下，伙伴 LU 同时支持许多会话，但某些低入口联网节点不支持并行会话。

在定义特定远程节点上的伙伴 LU 或定义伙伴 LU 别名时，此字段适用。

位置 伙伴 LU 所驻留的节点的标准 CP 名称，或者是可以提供对伙伴 LU 的访问的节点的标准 CP 名称。如果您提供尚未定义的远程节点的名称，则在无法动态发现该节点时，需要定义该节点。

仅当定义特定远程节点上的伙伴 LU 时此字段才适用。

定义伙伴 LU 的链路站路由

可以使用链路站路由来通过用来访问伙伴 LU 的链路站来标识伙伴 LU 的位置。

注:

1. 在可以动态找到资源的 APPN 网络中，链路站路由不是必需的。不建议在 APPN 网络中使用链路站路由，这是因为该网络会绕过正常的 APPN 路由机制。
2. 不能将链路站路由与企业扩展程序 (HPR/IP) 链路站配合使用。这是因为此链路类型上的所有通信都必须流经 RTP 连接，该连接对于特定链路站不是固定的，它可以切换到另一路径。

要配置伙伴 LU 的链路站路由，使用下列其中一个方法:

Motif 管理程序

从“节点”窗口上的**服务**菜单中选择 **APPC**、**新建伙伴 LU** 以及**链路站上的伙伴 LU**。

命令行管理程序

发出以下命令:

```
define_ls_routing
```

链路站路由参数

下列参数是链路站路由配置所必需的:

LU 名 控制链路站的本地 LU 的名称 (如果要通过特定链路站来找到伙伴 LU 的话)。

LS 名 链路站的名称。

伙伴 LU 名

伙伴 LU 的标准 LU 名称或通配符名:

- 标准 LU 名称由两个字符串组成，每个的长度都是 1-8 个字符。

此名称必须与远程节点上为此 LU 配置的名称匹配。如果不知道那个名称，则咨询 SNA 网络设计者。

- 通配符伙伴 LU 名与多个伙伴 LU 的标准 LU 名称匹配。通配符伙伴 LU 名由两个字符串组成，每个的长度都是 1-8 个字符:
 - 第一个字符串可以是与标准伙伴 LU 名称的第一部分完全匹配的完整 SNA 网络名，或者是与这些伙伴 LU 的网络名的开头部分匹配的通配符前缀。如果提供通配符前缀来作为第一个字符串的值，则保留第二个字符串为空白。例如，通配符条目 **A** 将与 SNA 网络中名为 A、ANT 或 APPN (而不是 BUFFALO 或 ZEBRA) 的所有 LU 匹配。

- 若为第一个字符串提供完整的 SNA 网络名，则也可以为第二个字符串输入一个值。（如果没有为第一个字符串提供有效的 SNA 网络名，则不能指定第二个字符串。）第二个字符串被当作通配符前缀对待，它必须匹配标准伙伴 LU 名称的第二部分的开头部分。例如，通配符条目 **A.F** 与伙伴 LU 名 A.FRED 或 A.FREDDY（而不是 APPN.FRED 或 A.B）匹配。

如果将两个字符串都保留为空白，则通配符伙伴 LU 定义与任何伙伴 LU 名匹配。

将伙伴 LU 名用作通配符

是否要将伙伴 LU 名用作通配符，而不是用作文字标准 LU 名称。

其他配置

在执行伙伴 LU 配置之后，继续执行下列配置任务：

- 要定义可调用 TP，请参阅『定义 TP』。
- 要定义方式，请参阅第 80 页的『定义方式和服务等级』。
- 要定义 CPI-C 辅助信息，请参阅第 84 页的『定义 CPI-C 辅助信息』。
- 要定义 APPC 安全性，请参阅第 86 页的『配置 APPC 安全性』。
- 要配置 5250 通信，请参阅第 89 页的第 7 章，『配置用户应用程序』。

定义 TP

本节说明如何定义 APPC TP。

在大多数情况下，不需要定义在 Communications Server for Linux 系统上运行的 TP；但在下列情况下需要配置 TP 定义：

APPC 特征

如果 Communications Server for Linux 计算机上的 TP 是调用 TP（即源 TP - 启动 APPC 对话的 TP），并且不需要限制对该 TP 的访问，则不需要定义该 TP。然而，可以定义 APPC TP 以指定下列特征，如第 79 页的『TP 定义参数』所述：

- 为 TP 定义对话安全性。
- 指示 TP 是使用基本对话还是使用映射对话。
- 指定同步点处理。
- 指定 PIP 数据处理。

可调用 TP

要使 TP 能够自动启动以响应入局分配请求，需将其定义为可调用 TP，如第 77 页的『服务器上的 TP 调用参数』所述。

可调用 TP（即目标 TP）是在响应入局分配请求时启动的 TP。必须为可调用 TP 创建 TP 定义。可调用 TP 可以是发出 RECEIVE_ALLOCATE 的 APPC TP，也可以是发出 Accept_Conversation 或 Accept_Incoming 的 CPI-C 应用程序。

注：在本手册中，短语“Receive_Allocate”用来指示这三个 API 调用中的任一 API 调用。

还可以定义可调用 TP 来将入局分配请求路由至正在运行的 TP。

定义 TP

对于可调用 TP，还可以指定超时值以限制对分配请求的等待。（仅可以使用命令行管理来配置此选项。）

Communications Server for Linux 将可调用 TP 定义用于下列用途：

- 当 TP 发出 Receive_Allocate 时，Communications Server for Linux 用适当的 TP 名搜索可调用 TP 定义。如果该定义存在，并包含 Receive_Allocate 超时值，则 Communications Server for Linux 在处理 Receive_Allocate 时使用此值；否则，它使用缺省值（无超时，这导致 TP 无限期地等待）。
- 当入局 Allocate 请求到达目标系统，并且请求的 TP 已不在和未完成的 Receive_Allocate 一起运行时，Communications Server for Linux 用入局 Allocate 中指定的 TP 名搜索 TP 定义。如果该定义存在，则 Communications Server for Linux 使用此定义中的信息来启动该 TP（如果允许多个实例或该 TP 尚未运行的话），或确定它应该将入局 Allocate 排队（如果 TP 已在运行并且不允许多个实例）。

必要的话，可以为同一 TP 配置两种类型的定义（例如，为可调用 TP 定义对话安全性）。

要配置 TP 定义，使用下列其中一个方法：

要定义 APPC 特征：

使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC** 和事务程序。当 Communications Server for Linux 显示 TP 窗口后，选择底部窗格并单击 **新建按钮**，或者选择现有的 TP 定义并单击 **属性按钮**。

命令行管理程序

发出 **snaadmin define_tp** 命令。

要定义可调用 TP：

服务器和客户机的配置方法是不同的：

- 在服务器上，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC** 和事务程序。当 Communications Server for Linux 显示 TP 窗口后，选择顶部窗格并单击 **新建按钮**，或者选择现有的可调用 TP 定义并单击 **属性按钮**。

命令行管理

发出 **snatpinstall** 命令。

•

AIX, LINUX

在 AIX 或 Linux 上的 IBM Remote API Client 上，发出 **snatpinstall** 命令。

•

WINDOWS

在 Windows 客户机上，发出 **tpinst32** 命令。（此命令适用于 32 位和 x64 版本的 Windows。）

有关使用 **snatpinstall** 或 **tpinst32** 命令的信息，请参阅第 151 页的附录 B，『从命令行配置可调用 TP』。

服务器上的 TP 调用参数

本节描述在服务器上配置可调用 TP 时 Motif 管理程序或命令行管理程序所需的参数。有关在客户机上配置可调用 TP 的信息，请参阅第 151 页的附录 B，『从命令行配置可调用 TP』。

下列参数是可在本地节点上调用的 TP 所必需的：

TP 名

具有下列其中一种格式的 TP 名：

应用程序 TP

如果 TP 是用户应用程序，则以正常字符提供该名称（最大长度为 64 个字符）。

服务 TP

如果 TP 是 SNA 服务事务程序，则输入十六进制的名称（最多 8 个十六进制位，表示 4 个字节）。

可以定义具有相同 TP 名的多个 APPC 可调用 TP，假定每个 TP 定义指定一个不同的 LU 别名。不能对 CPI-C 可调用 TP 这样做，因为您不能指定要使用的特定 LU 别名；每个 CPI-C 可调用 TP 必须具有不同的名称。

参数用于在任何 LU / 在特定 LU 上的调用

如果 TP 是 APPC TP，则此参数指定是要在任何 LU 上还是仅在特定 LU 上使 TP 成为可调用的。缺省情况下，可以在任何 LU 上调用 TP。

注：如果 TP 是 CPI-C 应用程序，则必须将此字段设置为使 TP 在任何 LU 上成为可调用的。CPI-C 不支持从特定本地 LU 接受入局 Attach；尝试为 CPI-C 应用程序指定此选项会在将入局 Attach 路由到 TP 时导致错误。

LU 别名

如果 TP 是 CPI-C 应用程序，则不得使用此字段。如果 TP 是 APPC 应用程序，则仅当指定了此 TP 定义的参数是用于在任何 LU 上的调用时此字段才适用。

TP 将接受入局 Attach 的本地 LU 别名。此名称必须与 Communications Server for Linux 节点上本地 APPC LU 的名称匹配。如果未指定 LU 别名，则 TP 接受来自任何本地 LU 的入局 Attach。

如果指定了非空 LU 别名，则该 TP 必须使用 RECEIVE_ALLOCATE 动词的扩展格式并指定此 LU 别名作为该动词的一个参数。这使 Communications Server for Linux 能够将入局 Attach 路由至正确的 TP。有关不同格式的 RECEIVE_ALLOCATE 的更多信息，请参阅 *Communications Server for Linux APPC Programmer's Guide*。如果需要允许 TP 在运行时确定正确的 LU 别名，而不是将该别名构建到应用程序中，则可以通过设置一个包含适当 LU 别名的

环境变量（使用 *Environment* 参数），并设计应用程序读取此环境变量以确定如何发出 `RECEIVE_ALLOCATE` 来执行此操作。

可以定义有相同 TP 名的多个 TP，提供的每个 TP 定义指定一个不同的 LU 别名。

支持多个实例

如果未选择此选项，则 TP 是排队 TP。当 TP 正在运行时到达的任何入局 Allocate 请求都会排队，直到该 TP 发出另一个 `Receive_Allocate` 或直到它运行完成并可以重新启动为止。仅当配置成能将入局 Allocate 请求路由至此计算机的 LU 接收入局 Allocate 请求，或者此计算机上没有配置路由信息的 LU 接收它时，才将该入局 Allocate 请求路由至此 TP。

如果选择此选项，则 TP 是不排队 TP。每当 Communications Server for Linux 的入局 Allocate 请求到达时，Communications Server for Linux 就启动该 TP 的新副本。不排队 TP 不能由操作员启动；它总是由 Communications Server for Linux 自动启动。对于不排队 TP，Communications Server for Linux 允许同时运行 TP 的多个副本。所有副本都以相同的用户和组标识以及同一工作目录运行，这由用户标识和组标识参数定义。如果 TP 写入本地系统上的文件，则需要确保 TP 的不同副本不覆盖彼此的文件。

在不排队 TP 已结束一个对话之后，该 TP 可以终止或发出另一个 `RECEIVE_ALLOCATE`。对于频繁使用的程序，这提供了一种方法来避免为每个对话启动新程序实例时的性能开销。每当自动启动的不排队 TP 接收到 Attach 时，Communications Server for Linux 检查此 TP 的实例中是否已经有未完成的 `RECEIVE_ALLOCATE`。如果已经有的话，则将此 TP 用于入局对话；否则 Communications Server for Linux 启动程序的新实例。

将入局 Allocate 路由至正在运行的 TP

仅当不支持多个实例时此选项才适用。

如果 TP 是广播排队 TP，则选择此选项。当 TP 正在运行时到达的任何入局 Allocate 请求都会排队，直到该 TP 发出另一个 `Receive_Allocate` 或直到它运行完成并可以重新启动为止。当 TP 启动时，将关于 TP 的信息广播至 LAN 中的所有服务器；如果另一台计算机上的 LU 接收到入局 Allocate 请求并且未配置路由信息，则它可以动态地找到 TP 并将 Allocate 请求路由至该 TP。

使用此选项时，不必在 LU 上配置显式的路由信息，并且通过在不同计算机上运行同一个 TP 的多个副本启用了负载均衡。但是，如果要避免广播信息以减少 LAN 数据传输，或者需要确保始终将抵达特定 LU 的入局 Allocate 请求路由至 TP 的同一个副本，则不要选择此选项。

TP 可执行文件的全路径

此 TP 的可执行文件的全路径和文件名。

该文件必须具有由用户标识参数指定的用户的执行许可权。另外，如果要在用户标识设置为 root 的情况下运行可执行文件，则该文件必须归 root 用户所有，并且必须具有 `setuid` 和 `setgid` 许可权，这样才能被 Communications Server for Linux 自动启动。

自变量 将被传递至 TP 的任何命令行自变量，由空格分隔。这些自变量按它们在此处显示的相同顺序传送至 TP。

此值是可选的。如果未包括此行，则在不指定任何命令行自变量的情况下调用 TP。

用户标识

Communications Server for Linux 启动 TP 时所用的用户标识。此行是必需的，必须指定此行。此标识必须是 Communications Server for Linux 计算机上的有效 Linux 登录标识。

TP 是在与此用户标识相关联的主目录中启动的。此主目录也是跟踪文件和由 TP 访问的任何其他文件的缺省路径（除非应用程序通过指定全路径来覆盖此缺省路径）。如果应用程序指定不带路径的文件名，则 Communications Server for Linux 在此主目录中搜索该文件；如果应用程序指定带有相对路径的文件名，则 Communications Server for Linux 在以此主目录为基准的指定目录中搜索该文件。

由 TP 可执行文件的全路径参数指定的 TP 的可执行文件必须有指定用户的执行许可权。另外，如果用户标识设置为 root，则该文件必须归 root 用户所有，并且必须具有 setuid 和 setgid 许可权，这样才能被 Communications Server for Linux 自动启动。

组标识 Communications Server for Linux 启动 TP 时所用的组标识。这必须是 Communications Server for Linux 计算机上的有效 Linux 组标识。

此参数是可选的。如果未包括它，则缺省值为 sna。

标准输入

指定由 TP 使用的标准输入文件或设备的全路径名。

此参数是可选的。如果未包括它，则缺省值为 /dev/null。

标准输出

指定由 TP 使用的标准输出文件或设备的全路径名。

此参数是可选的。如果未包括它，则缺省值为 /dev/null。

标准错误

指定由 TP 使用的标准错误文件或设备的全路径名。

此参数是可选的。如果未包括它，则缺省值为 /dev/null。

环境

指定 TP 需要的任何环境变量。

每个变量都以 *environment_variable=value* 的格式指定，最大长度为 255 个字符。字符串 *environment_variable=value* 在 = 字符之前或之后一定不能包含空格或制表符字符。

在 Motif 管理程序中，如果需要指定多个环境变量（最多 64 个），则使用 | 字符分隔它们。按这些变量在此处显示的相同顺序设置它们。

如果 TP 是 CPI-C 应用程序，则注意不能使用此字段设置环境变量 APPCLLU。不能为自动装入的 CPI-C 应用程序指定本地 LU。

此字段是可选的。如果未包括它，则不使用环境变量。

TP 定义参数

可以配置 APPC TP 以指定对话安全性、对话类型、同步级别以及 PIP 数据的处理。下列参数是定义 APPC 通信的 TP 所必需的：

TP 名

具有下列其中一种格式的 TP 名：

应用程序 TP

如果 TP 是用户应用程序，则以正常字符提供该名称（最大长度为 64 个字符）。

服务 TP

如果 TP 是 SNA 服务事务程序，则提供十六进制的名称（最多 8 个十六进制位，表示 4 个字节）。

需要对话级安全性

如果分配请求必须包括有效用户名和密码（或已验证密码的指示符），则选择此选项。如果未选择此选项，则无需验证。

限制访问权

如果必须将用户名包括在安全性访问列表上，则选择此选项。仅当选择需要对话级安全性选项时此字段才适用。

安全性访问列表

安全性访问列表的名称，该列表包含允许访问此 TP 的用户标识。如果选择了限制访问选项，则必须提供此值。

对话类型

指定 TP 是只接受基本对话、映射对话还是这两种类型对话都接受。

同步级别

指定 TP 接受的确认同步级别。有关确认同步的更多信息，请参阅 *Communications Server for Linux APPC Programmer's Guide*。选择下列其中一个值：

- 无
- 确认
- 同步点
- 无或确认
- 无、确认或同步点

允许 PIP

如果 TP 接受 PIP 数据（程序初始化参数）则选择此选项。

定义方式和服务等级

方式指定本地 LU（6.2 型 LU）用来与其伙伴 LU 通信的一组特征。这些特征包括关于在两个 LU 之间传送数据的方式的信息（如最大 RU 长度和调步窗口大小），以及关于这些 LU 是否可以建立并行会话的信息。

另外，您可能需要指定 LU 之间通信路径的需求，如强制某一级别的网络安全性、最小化传输时间或避免使用昂贵的通信链路。可以使用服务等级（COS）定义这些需求，COS 指定特征（如传输时间、传输成本和网络安全性）的最小和最大可接受值。COS 还指定与这些值的不同范围相关联的权重。这使得节点能够在两个或多个至同一远程 LU 的路由可用时计算网络上的最佳路由。

如果 Communications Server for Linux 节点是网络节点，则每个方式的定义都包含该方式所必需的 COS 的名称。如果 Communications Server for Linux 节点是低入口联网节点或端节点，则您不需要使 COS 与方式相关联；COS 名是动态确定的。

SNA 定义了涵盖大多数系统的需求的标准方式和关联的 COS，一般不需要定义其他方式和 COS。仅当必需的方式不是其中一个预定义的标准方式时（这可以在“方式”窗口中查看），您才需要定义方式。

如果入局对话中的方式名是不可识别的，则使用缺省方式。如果未指定缺省方式，则缺省方式为空白方式名。

标准方式名及其关联的 COS 名显示在表 2 中。有关与这些标准名称相关联的参数的更多信息，请参阅 IBM SNA 手册 *LU 6.2 Reference—Peer Protocols*（对于方式）和 *APPN Architecture Reference*（对于 COS）。

表 2. 标准方式和 COS 名

| 方式名 | 关联的 COS 名 | 用途 |
|----------|-----------|------------------------------|
| (空白) | #CONNECT | 未指定方式名的会话（基本缺省 COS 参数） |
| #BATCH | #BATCH | 由批处理应用程序使用的会话 |
| #BATCHSC | #BATCHSC | 由批处理应用程序使用的、具有最小级别路由安全性的会话 |
| #BATCHC | #BATCH | 在批处理应用程序中使用压缩的会话 |
| #BATCHCS | #BATCH | 在批处理应用程序中使用压缩并具有最小级别路由安全性的会话 |
| #INTER | #INTER | 由交互式应用程序使用的会话 |
| #INTERSC | #INTERSC | 由交互式应用程序使用的、具有最小级别路由安全性的会话 |
| #INTERC | #INTER | 在交互式应用程序中使用压缩的会话 |
| #INTERCS | #INTER | 在交互式应用程序中使用压缩并具有最小级别路由安全性的会话 |
| SNASVCMG | SNASVCMG | CNOS（更改会话数）和管理服务会话 |
| CPSVCMG | CPSVCMG | 节点间的 CP-CP 会话 |
| CPSVRMGR | CPSVRMGR | 用于从属 LU 请求者（DLUR）的 CP-CP 会话 |
| QPCSUPP | #CONNECT | 用于 5250 仿真的会话 |

一旦配置了方式，任何 APPC 或 CPI-C 应用程序就可使用该方式来激活本地 LU 和伙伴 LU 之间的会话。APPC 应用程序必须指定要使用的方式，但 CPI-C 应用程序可以使用 CPI-C 辅助信息（它包括方式名）。有关配置 CPI-C 辅助信息的更多信息，请参阅第 84 页的『定义 CPI-C 辅助信息』。

要配置方式或服务等级，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC** 和方式，然后选择“方式”窗口上的新建。

命令行管理程序

要定义方式，发出以下命令：

```
define_mode
```

要更改缺省方式，发出以下命令：

```
define_defaults
```

要定义服务等级，发出以下命令：

define_cos

方式配置参数

下列参数是方式配置所必需的:

名称 您正在定义的方式的名称。方式名是 1-8 个字符的字符串。

使用此方式的 APPC 应用程序 (包括本地和远程应用程序) 也可以使用此名称, 所以, 请与您的应用程序开发者一起检查该名称 (或参阅第三方应用程序的产品文档)。

COS 名

此方式的服务等级的名称。该名称是 1-8 个字符的字符串。通常, 对于用于交互式数据交换的方式, 可以简单地指定 #INTER, 对于用于批处理数据传输的方式, 可以简单地指定 #BATCH。

此字段只适用于网络节点。

如果不知道要指定什么值, 则咨询 SNA 网络设计者。

会话限制

使用下列字段来指定会话限制:

初始会话限制

一对使用此方式的 LU 可以具有的最大会话数 (最多为最大会话限制), 除非使用 CNOS 协商了不同的最大值。

通常将值 8 用于此字段。如果有疑问, 则咨询 SNA 网络设计者或 APPC 应用程序开发者 (对于第三方应用程序, 请参阅产品文档)。

最大会话限制

一对使用此方式的 LU 之间允许的最大会话数 (最多为 32,767), 与 CNOS 协商值相等。

通常将此字段设置为与初始会话限制的值相同。如果有疑问, 则咨询 SNA 网络设计者或 APPC 应用程序开发者 (对于第三方应用程序, 请参阅产品文档)。

最小争用胜方会话数

Communications Server for Linux 必须保留以供作为争用胜方的本地 LU 使用的会话的数目 (最大为会话限制)。

通常可将此字段安全地设置为 0, 但如果您不能肯定, 则咨询 SNA 网络设计者。

最小争用胜方会话数和最小争用败方会话数的和不得超过初始会话限制。

最小争用败方会话数

Communications Server for Linux 必须保留以供作为争用败方的本地 LU 使用的会话的最小数目。此值与最小争用胜方会话数字段中的值一起确定如何解决会话争用情况。

通常可将此字段安全地设置为 0, 但如果您不能肯定, 则咨询 SNA 网络设计者。

最小争用胜方会话数和最小争用败方会话数的和不得超过初始会话限制。

自动激活会话

在为使用此方式的本地 LU 和伙伴 LU 之间的会话进行了 CNOS 协商之后自动激活的会话数（最多为最小争用胜方计数）。指定此字段的值将允许使用此方式的 LU 在响应来自 TP 的请求时自动启动会话，以便立即分配对话。

接收调步窗口

使用这些字段来指定在发送 SNA 调步响应之前可以接收多少 RU:

初始窗口大小

本地 LU 在必须将调步响应发送至远程 LU 之前可以接收的请求单元（RU）数的初始设置。可以将此值安全地设置为 4。

在某些情况下将它设置得高些可以提高性能，但这样做也会增加内存使用率。

最大窗口大小

本地 LU 在必须将调步响应发送至远程 LU 之前可以接收的最大请求单元数。（RU）数的初始设置。

此值是可选的。如果未提供它，则最大接收调步窗口是无限制的。如果提供了一个值，则用该值限制自适应调步的接收调步窗口的大小。如果未使用自适应调步，则忽略此值。

调步窗口可以在 0 至 32767 字节之间。值 0 指定无限制的窗口。

如果相邻节点只支持固定调步，则这些值确定固定调步窗口大小；但相邻节点仍然可以通过协商设置窗口大小。如果相邻节点使用自适应调步，则这些值设置初始窗口大小。

指定超时

如果要指定使用此方式的 LU 6.2 会话在可以超时之前必须处于不活动状态的秒数（0 - 65535），则选择此选项。更改此值只影响使用此定义激活的会话（而不会影响已活动的会话）。

如果使用值 0，则只要会话一空闲就超时。

限制最大 RU 大小

如果要指定最大 RU 大小（它确定在将数据发送至伙伴 LU 之前要缓冲的数据量），则选择此选项。

上限可以在 256 至 62440 字节之间。您可以安全地将上限设置为 1024 个字节。在某些情况下将它设置得高些可以提高性能，但这样做也会增加内存使用率。

下限可以为 0 或一个从 256 至您指定的上限之间的值。

如果此字段中的值与为远程节点定义的 RU 大小不同，用于与那个节点的会话的最大 RU 大小可以经协商后建立该会话的适当 RU 大小。实际值不能低于下限字段。

这些数与发送和接收调步值一起可用来调整本地 LU 和伙伴 LU 之间的会话级吞吐量。如果不知道要使用什么值，则从缺省值开始并按需要调整它们以使吞吐量最大化。

支持压缩

使用此方式的会话是否支持数据压缩。如果未设置此选项，则不使用压缩。

如果设置此选项，则可以指定用于入站数据和出站数据的最大压缩级别。这些选项是相互独立的，因此，可以对两个方向指定不同的级别，也可以在一个方向上使用压缩，但在另一个方向上不使用压缩。在每个方向上，可以选择无（表示不进行压缩）或 RLE（最小压缩）、LZ9 或 LZ10（最大压缩）这些值中的其中一个值。

复位为 SNA 定义值

如果您正在使用 Motif 对话框修改标准方式，则可以单击此按钮将方式参数的值复位为 SNA 定义值。

其他配置

在执行方式配置之后，继续执行下列配置任务：

- 要定义 CPI-C 辅助信息，请参阅『定义 CPI-C 辅助信息』。
- 要定义 APPC 安全性，请参阅第 86 页的『配置 APPC 安全性』。
- 要配置 5250 通信，请参阅第 89 页的第 7 章，『配置用户应用程序』。

定义 CPI-C 辅助信息

若打算支持使用 CPI-C 符号目标名称的 CPI-C 应用程序，则需要定义 CPI-C 辅助信息。该辅助信息使符号目标名称与关于对话的伙伴 TP、伙伴 LU、方式以及安全性的信息关联。

要确定 CPI-C 的符号目标名称，请教应用程序开发者（对于第三方应用程序，则查阅产品文档）。

要配置 CPI-C 辅助信息，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC** 和 **CPI-C**。

命令行管理程序

发出以下命令：

```
define_cplic_side_info
```

CPI-C 配置参数

对于由应用程序使用的每个 CPI-C 符号目标名称，收集下列信息：

名称 您想要运行的 CPI-C 应用程序（也称为 TP）所使用的符号目标名称。此名称的长度可为 1–8 个字符。

应用程序开发者（对于第三方应用程序，则为产品文档）可提供此名称。

本地 LU

由使用此辅助信息的 TP 使用下列其中一个方法启动的任何对话的本地 LU：

本地 LU 别名

本地 LU 的别名。

使用缺省 LU

指定此选项以使用缺省池（如果存在）的成员或节点控制点 LU（如果未定义缺省池）。

如果设置了 APPCLLU 环境变量，则忽略您提供的本地 LU 信息，而是使用为该环境变量指定的 LU。

伙伴 LU

由使用此辅助信息的本地 TP 启动的对话的别名或标准伙伴 LU 名称。伙伴 LU 必须是在运行伙伴 TP 的计算机上配置的 LU。

方式 用来访问伙伴 LU 的 APPC 方式的名称。在大多数情况下，该方式是下列其中一种预定义方式：

- 空白名
- #BATCH
- #BATCHSC
- #INTER
- #INTERSC
- QPCSUPP

伙伴 TP

CPI-C 应用程序与之通信的事务程序的名称：

- 如果 TP 是用户应用程序，则以正常字符指定该名称（最大长度为 64 个字符）。
- 如果 TP 是服务 TP，则以十六进制指定该名称（最多 8 个十六进制位，表示 4 个字节）。

应用程序开发者（对于第三方应用程序，则为产品文档）可提供此信息。

安全性 您想要使用的对话级安全性的级别。选项如下：

无 伙伴 TP 不需要检查安全性参数。

相同 伙伴 TP 使用安全性，但接受由启动 TP 提供的用户标识和密码的本地 TP 的验证。如果您选择安全级别“相同”，则还需要指定伙伴 TP 接受的有效用户标识。

程序 伙伴 TP 需要用户标识和密码。如果您选择安全级别“程序”，则需要指定伙伴 TP 接受的有效用户标识和密码。

强健程序

伙伴 TP 需要用户标识和密码。本地和远程节点都必须支持安全性增强以便对密码进行加密。

请参阅 CPI-C 应用程序的文档或咨询应用程序员以了解要使用什么安全性参数。

用户标识

如果您已选择安全级别相同、程序或可靠程序，则指定要在启动消息上发送至远程应用程序的用户标识。此值必须与定义应用程序接受的用户标识匹配。

此用户标识与本地或远程节点上的 Linux 登录用户标识不相关。如果远程节点正在运行 Communications Server for Linux，则必须在远程节点上使用“对话安全性配置”对话框来配置用户标识。

密码 如果将安全级别指定为程序或可靠程序，则指定分配对话时要发送的密码。此值必须与在远程应用程序处定义的密码匹配，才能供提供的用户名使用。

此密码与本地或远程节点上的 Linux 登录密码不相关。如果远程节点正在运行 Communications Server for Linux, 则必须在远程节点上使用“对话安全性配置”对话框来配置密码。

其他配置

在执行 CPI-C 配置之后, 继续执行下列配置任务:

- 要定义 APPC 安全性, 请参阅『配置 APPC 安全性』。
- 要配置 5250 通信, 请参阅第 89 页的第 7 章, 『配置用户应用程序』。

配置 APPC 安全性

可以为 APPC 安全性执行下列配置任务:

- 按『配置会话安全性』所述配置会话安全性
- 按第 87 页的『配置对话安全性』所述配置对话安全性
- 按第 87 页的『配置安全性访问列表』所述配置安全性访问列表

配置会话安全性

会话级安全性用来验证 LU-LU 会话。每个定义由一个本地 LU 名、一个伙伴 LU 名以及一个密码组成。

Communications Server for Linux 使用密码来验证本地 LU 与伙伴 LU 之间的会话。(这些密码与 Linux 登录密码不相关。)

要配置会话安全性, 使用下列其中一个方法:

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC、安全性以及会话级安全性**。

命令行管理程序

发出以下命令:

```
define_lu_lu_password
```

会话安全性配置参数

下列参数是会话安全性配置所必需的:

本地 LU

本地 LU 的 LU 名。该名称是 1-8 个字符的字符串。

伙伴 LU

伙伴 LU 的标准 LU 名称。

密码

Communications Server for Linux 可用于验证本地 LU 与伙伴 LU 之间的会话的密码。该密码是一个用来创建键的 16 位十六进制数, 建立会话时交换该密码。此密码与本地或远程节点上的 Linux 登录密码不相关。

其他配置

在执行会话安全性配置之后, 继续执行下列配置任务:

- 要配置对话安全性, 请参阅第 87 页的『配置对话安全性』。
- 要配置 5250 通信, 请参阅第 89 页的第 7 章, 『配置用户应用程序』。

配置对话安全性

对话安全性用来验证入局对话。每个定义由一个用户标识和一个密码组成。此用户标识与本地或远程节点上的 Linux 登录用户标识不相关。

要配置对话安全性，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC**、**安全性**以及**对话级安全性**。

命令行管理程序

发出以下命令：

```
define_userid_password
```

对话安全性配置参数

下列参数是对话安全性配置所必需的：

用户标识

在来自远程节点的入局对话中要被接受的用户标识。用户标识的最大长度可以是 10 个字符。

密码 在来自远程节点的入局对话中要被接受的密码。密码的最大长度可以是 10 个字符。

其他配置

在配置对话安全性之后，可以配置 5250 通信，如第 89 页的第 7 章，『配置用户应用程序』所述。

配置安全性访问列表

可以定义 APPC 安全性访问列表来控制对 LU 或 TP（或这两者）的访问权。APPC 本地 LU 或 TP 的定义可以引用此列表。

要配置安全性访问列表，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **APPC**、**安全性**以及**对话级安全性**，然后选择“安全性访问列表”窗格并选择**新建**。

命令行管理程序

发出以下命令：

```
define_security_access_list
```

安全性访问列表配置参数

下列参数是安全性访问列表配置所必需的：

名称 安全性访问列表的名称。APPC TP 或本地 LU 的定义可以使用此名称来引用访问列表。

访问表中的用户

包括在安全性访问列表中的用户名。

配置 APPC 安全性

其他配置

在执行安全性访问列表配置之后，继续执行下列配置任务：

- 配置 TP 访问权，如第 75 页的『定义 TP』所述。

第 7 章 配置用户应用程序

本章提供有关配置 SNA 资源以支持使用下列任何通信的用户应用程序的指示信息：3270、5250 和 LUA。此类应用程序需要的 SNA 资源为 LU。

对于 3270、LUA 以及从属 APPC 通信，必须配置从属 LU。对于独立 APPC 和 5250 通信，可以使用缺省控制点 LU（配置本地节点时自动定义）或定义独立 LU。

在可以配置本章中描述的资源之前，必须执行下列配置：

- 按第 48 页的『配置节点』所述配置节点。
- 按第 51 页的第 4 章，『定义连接组件』所述配置连接。对于 3270、LUA 以及从属 APPC 通信，必须配置链路来支持从属 LU 数据传输。对于独立 APPC 和 5250 通信，链路必须支持独立 LU 数据传输。

如果正在使用上游 SNA 网关或 DLUR，则不需要配置至主机的直接链路。有关更多信息，请参阅第 98 页的『配置 SNA 网关』和第 61 页的『定义 DLUR PU』。

以下列表描述了每种类型的用户应用程序所需的配置任务：

3270 应用程序

对于 3270 通信，配置下列资源：

1. 对于 3270 显示器或打印机，请定义从属 LU，如第 65 页的『定义 0-3 型 LU』所述。
2. 要使 3270 显示器能够从 LU 池中进行选择，请定义 LU 池，如第 67 页的『定义 LU 池』所述。如果显示器使用专用 LU，则可以跳过此步骤。

5250 应用程序

对于 5250 通信，配置下列资源：

1. 配置用于 APPC 通信的节点：
 - a. 如果可以使用本地节点的控制点 LU，则不需要配置本地 LU。如果需要本地 LU 定义（例如，要使用会话安全性），则定义本地 LU，如第 69 页的『定义本地 LU』所述。
 - b. 如果本地节点是低入口联网节点，则必须将 AS/400 系统定义为远程节点，如第 71 页的『定义远程节点』所述。

如果本地节点是 APPN 端节点或网络节点，则可以将 AS/400 系统上的控制点 LU 用作伙伴 LU，因此不需要配置任何其他伙伴 LU。

由于 5250 使用标准方式 QPCSUPP，所以不需要定义任何方式。

LUA 应用程序

要支持 LUA 应用程序，配置下列资源：

1. 按第 65 页的『定义 0-3 型 LU』所述定义从属 LU。
2. 要使 LUA 应用程序能够从 LU 池中进行选择，则按第 67 页的『定义 LU 池』所述定义 LU 池。如果应用程序使用专用 LU，则可以跳过此步骤。

配置用户应用程序

LUA 应用程序使用节点的 LU 0-3 资源与主机应用程序通信。不需要定义任何其他资源。

第 8 章 配置传递服务

运行 Communications Server for Linux 的服务器上的传递服务在 SNA 主机与未直接连接至该主机的本地系统之间启动了通信。

Communications Server for Linux 包含用于 TN3270、TN3287 和 TN3270E 客户机（统称为“TN3270 客户机”）的 TN 服务器支持。要配置此功能，请参阅『配置 TN 服务器』。

Communications Server for Linux 还包含“TN 重定向器”支持，用于传递对 TN3270、TN3270E、TN5250 和 VT 客户机（统称为“Telnet 客户机”）的 TCP/IP 主机访问。要配置此功能，请参阅第 95 页的『配置 TN 重定向器』。

SNA 网关在主机和本地系统之间提供连接。可以在本地节点上配置 LU 来支持此功能（请参阅第 98 页的『配置 SNA 网关』），或者可以定义用来支持尚未显式配置的下流 LU 的模板（请参阅第 52 页的『定义 DLC、端口和连接网络』）。

DLUR 支持 APPN 网络中主机和节点之间的从属 LU 会话。要配置此功能，请参阅第 99 页的『配置 DLUR』。

配置 TN 服务器

TN 服务器使 TN3270 客户机能够通过实现了 TN 服务器的 Communications Server for Linux 中间节点来与主机通信。TN3270 客户机使用 TCP/IP 与 TN 服务器连接，并使用 TN 服务器上定义的 LU。这些 TN 服务器 LU 建立与主机处的 LU 的会话来支持客户机的 TN3270 会话。

在可以配置 TN 服务器之前，必须执行下列配置任务：

- 按第 48 页的『配置节点』所述定义本地节点。
- 按第 51 页的第 4 章，『定义连接组件』所述配置本地节点与主机之间的从属通信的端口和链路站。
- 定义本地节点上用于与主机通信的 TN3270 LU。要添加这些 LU，请参阅第 65 页的『定义 0-3 型 LU』。
- 如果将要使用任何 LU 池，则按第 67 页的『定义 LU 池』所述定义它们。

要配置 TN 服务器，执行下列任务：

- 为将使用该服务器的每个 TN3270 客户机配置 TN 服务器访问记录，或配置允许任何客户机访问该服务器的缺省记录（请参阅第 92 页的『配置 TN 服务器访问记录』）。
- 如果要支持 TN3270E 或 TN3287 客户机，则可以为显示器 LU 和打印机 LU 定义一个关联记录（请参阅第 94 页的『配置 TN 服务器关联记录』）。此记录使 TN3270E 或 TN3287 客户机能够选择特定的打印机（通过选择关联的显示器 LU）。必须授权该客户机选择 TN 服务器访问记录中的 LU。

TN 服务器的其他选项允许强制打印机响应、对所有 TN3270 会话指定“保持活动”方法以及指定如何访问外部 LDAP 服务器（该服务器保存一个用于检查 TN3270 客户机权限的撤销列表）。要访问这些选项，使用“TN 服务器”窗口上的**服务**菜单。

配置 TN 服务器访问记录

TN 服务器访问记录指示哪些 TN3270 客户机可以访问 TN 服务器以及这些客户机应该使用哪些 LU。每个访问记录标识一个被允许访问 TN 服务器的 TN3270 客户机、该客户机所连接的 TCP/IP 端口以及该客户机所使用的 LU 或 LU 池。

也可以定义允许任何 TN3270 客户机访问的缺省记录（对所有客户机使用相同 LU 或 LU 池）。

仅当节点、端口以及链路站是活动的时候，TN3270 客户机才可以使用 TN 服务器。

要配置 TN 服务器访问记录，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的**服务**菜单中选择 **TN 服务器**，从子菜单选择 **TN 服务器**。在出现的窗口上，选择“TN 服务器客户机访问许可权”窗格并选择**新建**。

命令行管理程序

发出以下命令：

```
define_tn3270_access
```

TN 服务器访问记录配置参数

下列参数是 TN 服务器访问记录配置所必需的：

TN3270 客户机地址

标识访问记录所适用的 TN3270 客户机的地址：

缺省记录

允许任何 TN3270 客户机访问。

TCP/IP 名或别名

允许指定的 TN3270 客户机访问。如果知道该客户机的 TCP/IP 名，则选择此选项并输入该名称。在许多计算机上，可以使用 **hostname** 命令来了解计算机的 TCP/IP 名。

TCP/IP 地址

允许从特定的 TCP/IP 地址访问。如果知道 TN3270 客户机的 TCP/IP 地址，则选择此选项并以标准 TCP/IP 点分十进制地址格式输入该地址。

支持 TN3270E

由该节点提供的 TN3270 支持的级别：

TN3270 仅支持 TN3270 协议。选择此选项禁用 TN3270E 协议的服务器支持，即使这些协议在客户机上受支持的。

TN3270E

支持 TN3270 和 TN3270E 协议（缺省值）。

无论选择哪个选项，TN3270 和 TN3287 协议总是受支持的。

对于 AS/400 TN3270 客户机，必须将此选项设置为 TN3270E。

TCP/IP 端口号

TN3270 客户机所连接的端口的 TCP/IP 端口号（在 TN 服务器上）。

注： TCP/IP 端口与 SNA 端口完全无关。

TN3270 服务的熟知端口号为 23，但使用此端口号可能会与 Linux 系统的 TELNET 服务冲突。Communications Server for Linux 包含允许在 **telnet** 与 TN3270 守护程序之间共享此端口的实用程序；您必须安装该实用程序才能使这两个守护程序起作用。有关此实用程序的详细信息，请参阅 *Communications Server for Linux Administration Command Reference* 中 **define_tn3270_access** 命令的信息。

如果选择 TN 服务器上不在使用的其他端口号，则还需要在 TN3270 客户机上配置那个端口号（或使用指定端口号的选项启动 TN3270 客户机）。2000 以上的端口号大概是可用的。256–1023 范围内的端口号可能提供稍好的安全性，但多半在使用中。

如果想要 TN3270 客户机能够使用多个 LU 或 LU 池，则定义多个访问记录，每个具有不同的 TCP/IP 端口号，以便可以通过指定不同的端口号来标识不同的 LU 或 LU 池。

指定的显示器 LU

TN3270 客户机在活动时所访问的 LU 的名称。该 LU 必须是本地节点上的从属 LU。您可以指定 LU 池的名称而不是特定 LU 的名称。

指定的打印机 LU

使用此访问记录的客户机的缺省打印机 LU 或 LU 池的名称。必须将此 LU 定义为本地节点上的从属 LU。

允许访问特定 LU

指定此选项以使 TN3270E 和 TN3287 客户机能够为会话请求特定 LU。（此选项对于 TN3270 是不可用的。）

SSL 安全会话

指定此选项以指示此会话使用安全套接字层（SSL）来访问服务器。

仅当在服务器上安装了支持 SSL 所需要的其他软件时，此选项才可用；否则不能选择此选项。

注： 如果此会话的 *TCP/IP* 端口号参数指示它使用 Telnet 守护程序的 TCP/IP 端口，则不要对此会话使用 SSL。如果对使用 Telnet 守护程序的 TCP/IP 端口的会话使用 SSL，则当节点处于活动状态时，Telnet 客户机将无法使用 **telnet** 来访问 Communications Server for Linux 计算机。

执行客户机认证

仅当您选择了 SSL 安全会话选项时，此选项才会出现。

指定此选项以指示 TN 服务器要求会话使用客户机认证。客户机必须发送有效的证书（用于将它标识成有权使用 TN 服务器的有效客户机的信息）。

除了检查证书是否有效以外，TN 服务器可能还需要对照外部 LDAP 服务器上的证书撤销列表来检查证书，以确保用户的授权尚未被撤销。在这种情况下，您还必须使用“TN 服务器高级参数”对话框来指定如何访问此服务器。

安全级别

指示此会话需要的 SSL 安全级别。该会话将使用客户机和服务器可支持的最高安全级别；如果客户机不能支持请求的安全级别或更高安全级别，则不会启动该会话。

仅当您选择了 SSL 安全会话选项时，此选项才会出现。

可能的值为：

仅认证 必须交换证书，但将不使用加密。当客户机通过安全内部网进行连接时，一般使用此选项来避免加密的开销。

最小认证

客户机必须从服务器请求证书来检查其有效性；不需要加密（但如果客户机请求的话，也可以使用加密）。

最少 40 位

客户机必须支持最少 40 位加密。

最少 56 位

客户机必须支持最少 56 位加密。

最少 128 位

客户机必须支持最少 128 位加密。

最少 168 位

客户机必须支持最少 168 位加密。

注：使用加密要求将其他软件与 Communications Server for Linux 一起安装；有关更多信息，请参阅《*Communications Server for Linux 快速入门*》。根据您的位置不同，可能并非列示的所有加密级别您都能使用，因为支持它们所需要的软件在您所在的国家或地区可能不可用。

其他配置

在执行 TN 服务器访问配置之后，继续执行下列配置任务：

- 按『配置 TN 服务器关联记录』所述配置 TN 服务器关联记录。

配置 TN 服务器关联记录

TN 服务器关联记录定义了打印机 LU 与显示器 LU 之间的关联，以便 TN3270E 或 TN3287 协议可将两者连接在一起。如果客户机的访问记录允许选择特定的 LU，则此记录使客户机能够通过指定关联的显示器 LU 来选择特定的打印机。

要配置 TN 服务器关联记录，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的**服务**菜单中选择 **TN 服务器**，然后在“TN 服务器”窗口上选择“关联记录”窗格并选择**新建**。

命令行管理程序

发出以下命令：

```
define_tn3270_association
```

TN 服务器关联记录配置参数

下列参数是 TN 服务器关联记录配置所必需的：

显示器 LU

显示器 LU 的名称（必须在本地节点上定义它）。

打印机 LU

打印机 LU 的名称（必须在本地节点上定义它）。不要指定已在另一 TN 服务器关联记录上输入的打印机 LU。

配置 TN 重定向器

TN 重定向器使 TN3270、TN3270E、TN5250 和 VT 客户机（统称为 Telnet 客户机）能够通过实现了 TN 重定向器的 Communications Server for Linux 中间节点来与主机通信。客户机使用 TCP/IP 连接至 TN 重定向器；然后，TN 重定向器建立与主机的独立 TCP/IP 连接。

要配置 TN 重定向器，执行下列任务：

- 为将使用该服务器的每个 Telnet 客户机配置 TN 重定向器访问记录，或配置允许任何客户机访问该服务器的缺省记录（请参阅『配置 TN 重定向器访问记录』）。

配置 TN 重定向器访问记录

TN 重定向器访问记录指示哪些 Telnet 客户机可以通过 TCP/IP 链路访问 TN 重定向器。每个访问记录都标识了一个被允许访问 TN 重定向器的 Telnet 客户机、该客户机连接至 Communications Server for Linux 所使用的 TCP/IP 端口、Communications Server for Linux 连接至主机所使用的 TCP/IP 端口以及 SSL 安全性设置。还可以定义允许任何客户机访问的缺省记录。

如果想允许任何客户机使用 TN 重定向器且想要所有客户机使用相同的主机访问配置，则可以定义一个缺省记录。

Telnet 客户机只能在节点是活动的时候使用 TN 重定向器。

要配置 TN 重定向器访问记录，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **TN 服务器**，从子菜单选择 **TN 服务器**。在出现的窗口上，选择“TN 重定向器客户机访问许可权”窗格并选择**新建**。

命令行管理程序

发出以下命令：

```
define_tn_redirect
```

TN 重定向器访问记录配置参数

对于客户机和主机 TCP/IP 连接，TN 重定向器访问记录配置由两组参数组成。

客户机参数如下所示：

Telnet 客户机地址

标识访问记录所适用的 Telnet 客户机的地址：

缺省记录

允许任何 Telnet 客户机访问。

TCP/IP 名或别名

允许指定的 Telnet 客户机访问。如果知道该客户机的 TCP/IP 名，则选择此选项并输入该名称。在许多计算机上，可以使用 **hostname** 命令来了解计算机的 TCP/IP 名。

TCP/IP 地址

允许从特定的 TCP/IP 地址访问。如果知道客户机的 TCP/IP 地址，则选择此选项并以标准 TCP/IP 点分十进制地址格式输入该地址。

TCP/IP 端口号

客户机所连接的端口的 TCP/IP 端口号（在 TN 服务器上）。

注： TCP/IP 端口与 SNA 端口完全无关。

您还需要在客户机上配置此端口号（或使用指定端口号的选项启动客户机）。2000 以上的端口号大概是可用的。256–1023 范围内的端口号可能提供稍好的安全性，但多半在使用中。

SSL 安全会话

指定此选项以指示此会话使用安全套接字层（SSL）来访问服务器。

仅在服务器上安装了支持 SSL 所需要的其他软件时，此选项才可用；否则不能选择此选项。

执行客户机认证

仅当您选择了 SSL 安全会话选项时，此选项才会出现。

指定此选项以指示 TN 服务器要求会话使用客户机认证。客户机必须发送有效的证书（用于将它标识成有权使用 TN 服务器的有效客户机的信息）。

除了检查证书是否有效以外，TN 服务器可能还需要对照外部 LDAP 服务器上的证书撤销列表来检查证书，以确保用户的授权尚未被撤销。在这种情况下，您还必须使用“TN 服务器高级参数”对话框（从“TN 服务器”窗口上的**服务**菜单上访问）来指定如何访问此服务器。

安全级别

指示客户机会话需要的 SSL 安全级别。该会话将使用客户机和服务器可支持的最高安全级别；如果客户机不能支持请求的安全级别或更高安全级别，则不会启动该会话。

仅当您选择了 SSL 安全会话选项时，此选项才会出现。

可能的值为：

仅认证 必须交换证书，但将不使用加密。当客户机通过安全内部网进行连接时，一般使用此选项来避免加密的开销。

最小认证

客户机必须从服务器请求证书来检查其有效性；不需要加密（但如果客户机请求的话，也可以使用加密）。

最少 40 位

客户机必须支持最少 40 位加密。

最少 56 位

客户机必须支持最少 56 位加密。

最少 128 位

客户机必须支持最少 128 位加密。

最少 168 位

客户机必须支持最少 168 位加密。

注：使用加密要求将其他软件与 Communications Server for Linux 一起安装；有关更多信息，请参阅《*Communications Server for Linux 快速入门*》。根据您的位置不同，可能并非列示的所有加密级别您都能使用，因为支持它们所需要的软件在您所在的国家或地区可能不可用。

目标主机参数如下：

地址 标识访问记录所适用的主机的地址：

TCP/IP 名或别名

访问指定的主机。如果知道该主机的 TCP/IP 名，则选择此选项并输入该名称。在许多计算机上，可以使用 **hostname** 命令来了解计算机的 TCP/IP 名。

TCP/IP 地址

访问特定的 TCP/IP 地址。如果知道主机的 TCP/IP 地址，则选择此选项并以标准 TCP/IP 点分十进制地址格式输入该地址。

TCP/IP 端口号

TN 重定向器用来访问主机的 TCP/IP 端口号。

注：TCP/IP 端口与 SNA 端口完全无关。

您还需要在主机上配置此端口号。2000 以上的端口号大概是可用的。256–1023 范围内的端口号可能提供稍好的安全性，但多半在使用中。

SSL 安全会话

指定此选项以指示 TN 重定向器使用安全套接字层（SSL）来访问主机。

仅当主机支持 SSL 时此选项才可用。

安全级别

指示主机会话需要的 SSL 安全级别。该会话将使用主机和服务器可支持的最高安全级别；如果主机不能支持请求的安全级别或更高安全级别，则不会启动该会话。

仅当您选择了 SSL 安全会话选项时，此选项才会出现。

可能的值为：

仅认证 必须交换证书，但将不使用加密。当主机连接是在安全内部网上时，一般使用此选项来避免加密的开销。

最小认证

主机必须从服务器请求证书来检查其有效性；不需要加密（但如果主机请求的话，也可以使用加密）。

最少 40 位

主机必须支持最少 40 位加密。

最少 56 位

主机必须支持最少 56 位加密。

最少 128 位

主机必须支持最少 128 位加密。

最少 168 位

主机必须支持最少 168 位加密。

注：使用加密要求将其他软件与 Communications Server for Linux 一起安装；有关更多信息，请参阅《*Communications Server for Linux 快速入门*》。根据您的位置不同，可能并非列示的所有加密级别您都能使用，因为支持它们所需要的软件在您所在的国家或地区可能不可用。

配置 SNA 网关

通常，从属 LU 会话需要一条至主机的直接通信链路。然而，运行 Communications Server for Linux 并具有至主机的直接通信链路的节点也可以为下游计算机上的 LU 提供 SNA 网关设施，从而使它们能够通过该通信链路从 Communications Server for Linux 节点访问主机。下游计算机必须包含 2.0 或 2.1 型 SNA PU 才能支持与主机进行从属通信。例如，下游计算机可以是单机配置中的另一台运行 Communications Server for Linux 的计算机。

使用 SNA 网关功能部件，通过 Communications Server for Linux 本地节点路由在主机与下游计算机之间传送的所有数据。这使下游计算机能够与 Communications Server for Linux 或其他下游计算机共享主机连接，而不需要直接链路。例如，您可设置通过本地令牌环网络连接至 Communications Server for Linux 的若干台下游计算机，以便它们都可访问从 Communications Server for Linux 至主机的同一长途 SDLC 租用线路。

使用 SNA 网关还简化了主机的配置。主机配置仅需要包含 Communications Server for Linux 计算机及其主机通信链路；下游计算机上的 LU 将配置为 Communications Server for Linux 计算机的资源的一部分。主机不知道正在使用 SNA 网关。

在配置 SNA 网关之前，必须执行下列配置任务：

- 按第 48 页的『配置节点』所述定义本地节点。
- 按第 51 页的第 4 章，『定义连接组件』所述配置本地节点与主机之间的从属通信的端口和链路站。另外，配置本地节点与下游节点之间的从属数据传输的端口和链路站。对于下游链路，可以在端口上配置模板来支持隐式下游 LU（未在本地节点上显式定义的 LU）。
- 定义本地节点上用于与主机通信的 LU（上游 LU）。必须使用“0-3 型 LU 配置”对话框来定义上游 LU（包括从属 LU 6.2 LU），指定未限制（未知）的 LU 类型。要添加这些 LU，请参阅第 65 页的『定义 0-3 型 LU』。
- 如果将要使用任何 LU 池，则按第 67 页的『定义 LU 池』所述定义它们。

要启用 SNA 网关，必须在本地节点上配置 LU 来支持与下游工作站的会话。（如果在端口上配置了模板来支持隐式下游 LU，则可能不需要显式定义下游 LU）。在本地节点上定义的 LU 称为“下游 LU”。要配置下游 LU，需要在下游节点上使用的 LU 号以及主机 LU 的名称。（在下游节点上定义的 LU 可以是任何类型的从属 LU。）

要配置下游 LU，使用下列其中一个方法：

Motif 管理程序

从“节点”窗口上的服务菜单中选择 **SNA 网关和新建下游 LU**。

命令行管理程序

发出下列其中一个命令：

define_downstream_lu

define_downstream_lu_range

下游 LU 配置参数

下列参数是下游 LU 配置所必需的:

下游 LU 名

每个下游 LU 的名称。该 LU 名仅用来在本地标识 LU，不需要与下游节点上的任何配置匹配。

如果正在定义 LU 的范围，则指定 1-5 个字符的基本名称。Communications Server for Linux 将一个三位十进制字符串添加至基本名称来为您指定的每个 LU 号创建 LU 名。

下游 PU 名

至下游节点的链路站点的名称。

LU 号

该 LU 号必须与下游节点上定义的 LU 号匹配。如果不知道要使用什么 LU 号，则与 SNA 网络设计者联系。

可以通过定义 LU 的范围来配置几个具有连续 LU 号的 LU。

上游 LU 名

下游 LU 将与之通信的主机 LU 或 LU 池的名称。

延迟登录

为了缩短用户启动时间，Communications Server for Linux 会显示未指定上游 LU 的登录屏幕；3270 用户在与上游 LU 相关联之前必须击键。

允许超时

为了减少需要的 LU 数，没有活动 PLU-SLU 会话的 LU 在此秒数之后断开与上游 LU 的联系。

其他配置

在执行 SNA 网关的下游 LU 配置之后，继续执行下列配置任务:

- 要配置用户应用程序，请参阅第 89 页的第 7 章，『配置用户应用程序』。

配置 DLUR

通常，从属 LU 会话需要一条至主机的直接通信链路。如果许多节点（包括主机节点）在 APPN 网络中连接在一起，则它们中的某些结点可能有通过另一节点的间接连接，而没有至主机的直接连接。没有直接连接，从这些间接连接的节点中的 LU 建立与主机的从属 LU 会话是不可能的。

从属 LU 请求者（DLUR）是为克服此限制而设计的 APPN 功能部件。您可以在 APPN 节点（如运行 Communications Server for Linux 的节点）上配置 DLUR。它与主机处的从属 LU 服务器（DLUS）一起工作，将会话从 DLUR 节点上的从属 LU 通过 APPN 网络路由至 DLUS 主机。

至主机路由可以跨多个节点，并且可以利用 APPN 的网络管理、动态资源定位以及路由计算等设施。DLUR 必须在定义这些 LU 的节点上是可用的，并且 DLUR 必须在主机节点上是可用的，但不必在会话路由中的任何中间节点上启用 DLUR。

注：不能在低入口联网节点上配置 DLUR。

如果 Communications Server for Linux DLUR 节点是网络节点，则它还可以为与 Communications Server for Linux 节点连接的下游计算机上的从属 LU 提供传递 DLUR 设施。（仅网络节点支持此功能。）这些下游 LU 可以使用 Communications Server for Linux 节点上的 DLUR 以节点内部的 LU 所用的同一方式在网络上访问主机。

注：不能在端节点上配置传递 DLUR。

配置 DLUR 所需要执行的任务取决于从属 LU 是在本地节点上还是在下游节点上。

要在本地节点上配置 DLUR 支持，必须执行下列配置任务：

1. 按第 48 页的『配置节点』所述定义本地节点。如果要为下游节点提供传递 DLUR 支持，则将该节点定义为 APPN 网络节点。
2. 配置与 APPN 网络的连接。APPN 连接至少需要一个端口和链路站用于本地节点与相邻 APPN 网络节点之间的独立通信，如第 51 页的第 4 章，『定义连接组件』所述。
3. 按第 61 页的『定义 DLUR PU』所述在本地节点上定义 DLUR PU。（DLUR PU 支持与主机的连接。）
4. 要配置 DLUR 以支持本地节点上的 LU，必须按第 89 页的第 7 章，『配置用户应用程序』所述在本地节点上添加 LU。可配置这些 LU 来支持 3270 显示器、3270 打印机或 LUA。根据这些 LU 所支持的用户应用程序的需求，也可能需要执行进一步的配置。

要为下游节点配置传递 DLUR 支持，必须执行下列配置任务：

1. 将本地节点定义为 APPN 网络节点（请参阅第 48 页的『配置节点』）。
2. 配置至下游节点的连接。按第 51 页的第 4 章，『定义连接组件』所述配置本地节点与每个下游节点之间从属通信的端口和链路站。（不需要定义 DLUR PU 就可以对下游节点支持 DLUR。）
3. 一个下游节点可以支持多个 PU。在这种情况下，每个下游 PU 都与不同的链路相关联，所以需要在 Communications Server for Linux DLUR 节点和下游节点之间配置多个链路，并且需要知道每个链路的下游 PU 名称。

第 9 章 从 NetView 中管理 Communications Server for Linux

Communications Server for Linux 包含与主机上的 NetView 程序一起运行的远程命令设施 (RCF)，从而使 NetView 操作员能够从主机 NetView 程序向 Communications Server for Linux 计算机发出命令。(有关 NetView 和 RCF 命令的简要概述，请参阅『使用主机 NetView 程序』。)

Communications Server for Linux RCF 提供了下列两项功能：

- 服务点命令设施 (SPCF) 使 NetView 操作员能够从 NetView 中使用命令行管理程序 `snaadmin` 所用的语法来发出 Communications Server for Linux 管理命令。第 103 页的『使用 SPCF』描述了此设施。
- UNIX 命令设施 (UCF) 使 NetView 操作员能够从 NetView 中发出 Linux 操作系统命令。第 105 页的『使用 UCF』描述了此设施。

可以用相同的方法从 NetView 控制台访问这两项功能，并且用于发出命令的总的语法是相同的。

使用主机 NetView 程序

Communications Server for Linux RCF 与主机上的 NetView 程序一起运行。主机必须正在运行 NetView V1R2 或更高版本；Communications Server for Linux 不支持 NetView V1R1。

要使用 NetView 程序，需要下列各项：

- 主机 NetView 程序的登录标识和密码（与主机人员联系以获取此信息）
- Communications Server for Linux 的服务点名，这是在主机上为 NetView 程序定义的（与主机人员联系以获取此信息）
- 用于访问正在运行 NetView 程序的主机的 DLC、端口以及链路站

为了测试 RCF 功能，您可能想通过使用 3270 仿真来从 Communications Server for Linux 访问 NetView，而不是从主机直接访问它。在此情况下，您还需要下列各项：

- 在主机上配置的 3270 LU
- 使用此 LU 的 3270 会话

请咨询主机管理员以获取必需的配置信息。

要访问 NetView 程序，请执行下列步骤：

1. 确保已使用节点配置文件启动了 Communications Server for Linux 软件，该节点配置文件包含 RCF 访问参数定义 (`define_rcf_access` 记录)。
2. 如果正在使用 3270 仿真来访问 NetView 程序，则启动 3270 仿真程序并激活与主机的会话。
3. 按照主机管理员提供给您指示信息来启动 NetView 并登录。(操作序列可能随 NetView 版本的不同而有所变化。)

4. 根据需要发出 SPCF 或 UCF 命令。
5. 如果正在使用 3270 仿真来访问 NetView，则发出命令之后按照 3270 文档的指示信息来结束 3270 仿真。

NetView 屏幕显示

NetView 屏幕布局随不同主机上 NetView 版本的不同而有所变化。图 19 显示了一般布局。

显示内容包括屏幕底部的输入区；这是可输入命令的区域。??? 一行将主屏幕区（NetView 在其中显示命令响应的区域）与输入区分隔开。

```
NCCF          N E T V I E W          [SCAN DDAC12  07/18/95  13:52:24 A
RUNCMD SP=ADCDPU01,APL=NODE,START_DLC,DLC_NAME= TOKR01
COMMAND ISSUED SUCCESSFULLY

???
runcmd sp=abcdpu01,appl=node,query_node
```

图 19. NetView 屏幕的示例

更改命令输入区的大小

缺省情况下，输入区为一行，但对于某些较长的命令则需要多行。在 NetView 的某些版本上，可以使用 **input** 命令来指定输入区为一行、两行或三行。为此，输入以下命令：

input *n*

在此命令中，*n* 是 1、2 或 3，指示所要的行数。如果此命令在您正在使用的 NetView 版本中不起作用，则与 NetView 支持人员联系。

RCF 命令语法概述

SPCF 命令和 UCF 命令都使用 RCF 命令语法：

runcmd *sp=spname, appl=component, commandtext*

NetView 使用 **runcmd** 实用程序来将命令字符串发送至远程系统。该命令包括下列参数：

sp= spname

指示对应于 Communications Server for Linux 节点的服务点名（此名称是在 NetView 中定义的）。主机 NetView 人员可为您提供此信息。

appl=component

指示 NetView 应该将该命令发送至的 Communications Server for Linux 组件的名称，如下所示：

节点 与服务点名 *sname* 相关联的 Communications Server for Linux 节点（对于 SPCF 命令）

unix 在与服务点名 *sname* 相关联的 Communications Server for Linux 计算机上运行的 UCF 守护程序（对于 UCF 命令）

commandtext

提供所发出的命令的文本。对于 SPCF，这是对 Communications Server for Linux 命令行管理程序发出的命令。对于 UCF，这是 Linux 操作系统命令。有关可使用的命令的更多信息，请参阅第 104 页的『对用于 SPCF 的管理命令的限制』或第 106 页的『允许的命令』。

大写字符和转义字符

虽然 Linux 能够对大小写字母字符进行区分，但 NetView 程序不能。相反，它在将字符发送至 Linux 计算机之前将所有字符转换成大写字符。由于 RCF 不能确定从主机接收的字母字符最初是大写的还是小写的，所以它假设接收到的字符是小写的。另外，主机字符集可能不支持方括号字符 [和]，这两个字符在某些命令中是必需的。

RCF 使用反斜杠字符 \ 来提供对大写字符和方括号字符的支持，如下所述：

- 要将大写字符包括在命令字符串中，在该大写字符前面包括一个反斜杠。前面没有反斜杠的字母字符被解释为小写字符。
- 要包括方括号字符 [和]，分别使用序列 \[和 \]。
- 要包括反斜杠字符 \ 本身，输入反斜杠两次。

如果单个反斜杠后面跟着任何其他非字母字符，则忽略反斜杠，该字符保持不变。

表 3 中显示了一些示例。

表 3. 在 RCF 命令中使用转义字符

| 要生成的字符 | 输入 |
|--------|--------|
| ABcd | \a\bcd |
| [] | \[\] |
| \a | \\a |
| \[| \\[|

通常在 Linux 命令行上使用的用于防止 Linux shell 解释特殊字符的转义字符对于 RCF 而言是不需要的。例如，对于包含字符 * 或 \$ 的字符串，不要像在 Linux 命令行上输入它们那样使用转义字符。而且，当使用 SPCF 发出管理命令时，应知道常量名如 LIST_FROM_NEXT 等是不区分大小写的。不必对这些字符进行换码来使它们变成大写字符。

使用 SPCF

SPCF 使您能够从 NetView 控制台发出命令来管理正在运行的 Communications Server for Linux 系统。这些命令与那些可使用 Communications Server for Linux 命令行管理程序 **snaadmin** 发出的命令相同（如 *Communications Server for Linux Administration Command Reference* 所述）。

有关 SPCF 命令的语法的信息，请参阅第 102 页的『RCF 命令语法概述』。跟在 `appl=node` 参数后面的命令文本是一个向 Communications Server for Linux 命令行管理程序发出的命令，使用的格式与您在 Linux 命令行上对 `snaadmin` 程序指定该命令时使用的格式相同。有关管理命令的语法和各个命令的参数的信息，请参阅 *Communications Server for Linux Administration Command Reference*。

对用于 SPCF 的管理命令的限制

与特定节点的资源相关联的管理命令（例如 `query_node` 和 `define_local_lu` 命令）被发送至与 SPCF 命令上指定的服务点名相关联的节点。不能使用 `-n` 选项来指定另一个节点名；因此，除非特定节点与 NetView 上的服务点名相关联，否则不能对此节点发出命令。始终可以发出与域资源或与 SNA 网络数据文件相关联但与特定节点不相关联的命令。有关某个命令是与节点、与域资源还是与 SNA 网络数据文件关联的信息，请参阅 *Communications Server for Linux Administration Command Reference* 中每个命令的描述。

不能使用命令行选项 `-i` 来指定从文件还是从标准输入中输入。必须直接在 NetView 控制台上输入所有命令。

对于 `query_*` 命令，可以使用在 Linux 命令行上输入命令时使用的方法来使用命令行选项 `-a`（返回所有条目）和 `-d`（返回详细信息）。

要提供安全性，可以设置 Communications Server for Linux 配置，以便某些类型的命令只允许从 SPCF 发出。例如，可允许远程用户发出 `query_*` 命令，但不允许激活或释放 Communications Server for Linux 组件。可分别控制对下列每个命令组的访问权：

- `define_*`、`set_*`、`delete_*`、`add_*` 和 `remove_*` 命令以及 `init_node`
- `query_*` 命令
- “操作”命令：`start_*`、`stop_*`、`activate_*`、`deactivate_*` 以及 `aping`、`initialize_session_limit`、`change_session_limit` 和 `reset_session_limit`

有关设置 SPCF 的安全性选项的更多信息，请参阅 *Communications Server for Linux Administration Command Reference* 中 `define_rcf_access` 命令的描述。

SPCF 命令的示例

下列示例显示了如何使用 SPCF 发出 `define_lu_0_to_3` 命令。此示例使用反斜杠字符来指示两个字符串 LU\$01 和 PU2 中的大写字母。不必使常量名 `3270_display_model_2` 中的字符成为大写字母，因为 `snaadmin` 程序以小写格式接受此字符串。

```
runcmd sp=mypname, appl=node, define_lu_0_to_3, lu_name=\lu$01,  
  
nau_address=1, pu_name=\pu2, lu_model=3270_display_model_2
```

下列示例显示如何使用 SPCF 发出 `query_lu_0_to_3` 命令。`-a` 选项指示“返回所有条目”，所以不必指定 LU 名或 PU 名。`-d` 选项指示“返回详细信息”，所以不必使用 `list_options` 参数来指定这一点。这两个选项所起的作用与 `snaadmin` 程序的这两个选项所起的作用完全相同。

```
runcmd sp=mypname, appl=node, -a -d query_lu_0_to_3
```

使用 UCF

UCF 使 NetView 操作员能够在运行 Communications Server for Linux 的计算机上通过在 NetView 控制台上输入命令文本来发出 Linux 命令，并且能够查看这些命令的输出。该设施不限于只能运行与 Communications Server for Linux 相关的命令；只要符合第 106 页的『允许的命令』中的限制，就可以发出任何类型的命令。

通过使用 UCF，远程操作员可监视 Communications Server for Linux 计算机上的活动和诊断问题，在某些情况下还可以执行更正操作。

可通过使用 **define_rcf_access** 命令（请参阅 *Communications Server for Linux Administration Command Reference*）来指定 Communications Server for Linux 是否支持 UCF。如果配置指定支持 UCF，则当节点启动时，Communications Server for Linux 启动 UCF 守护程序。通过对每个命令启动一个新的 Linux shell 并在该 shell 中运行该命令，UCF 守护程序可处理 UCF 发出的 Linux 命令。如果未包含 UCF 支持，则 Communications Server for Linux 不启动此程序。

该配置指定 UCF 用户的名称，该名称必须是 Communications Server for Linux 计算机上的有效登录名。UCF shell 是通过使用为该用户指定的 shell 程序、登录标识、许可权以及 **.login** 或 **.profile** 来启动的。（如果未指定 shell 程序，则使用 **/bin/sh**。）这意味着可使用一般的 Linux 系统安全性功能部件来限制 UCF 用户对文件和命令的访问，从而限制可从 UCF 中运行的命令的范围。

有关设置 UCF 配置的更多信息，请参阅 *Communications Server for Linux Administration Command Reference* 中 **define_rcf_access** 命令的描述。

UCF 命令语法

UCF 命令的语法如下所示：

```
runcmd sp=spname, appl=unix, unix_command
```

NetView 使用 **runcmd** 实用程序来将命令发送至远程系统。该命令包括下列参数：

sp= spname

指定 *spname*，它是在 NetView 中定义的服务点的名称。主机 NetView 人员可为您提供此信息。

appl=**unix**

指示 NetView 将命令发送至与服务点名 *spname* 相关联的 Communications Server for Linux 计算机上的 UCF 守护程序。

unix_command

提供 Linux 操作系统命令。此命令的输入与在 Linux 命令行上输入时一样，但需要使用转义字符来指示大写字母或方括号字符（如第 102 页的『RCF 命令语法概述』所述）。

通常在 Linux 命令行上使用的用于防止 Linux shell 解释特殊字符的转义字符对于 UCF 而言是不需要的。例如，对于包含字符 ***** 或 **\$** 的字符串，不要像在 Linux 命令行上输入它们那样使用转义字符。

允许的命令

UCF 是为了与那些不需要用户任何进一步交互作用就能完成（无论是否生成输出）的命令配合使用而设计的。例如，可以发出命令 `cat filename`（此命令在显示 `filename` 的内容后完成）或 `mv filename1 filename2`（除非发生错误，否则此命令在不生成输出的情况下完成）。

UCF 命令生成的输出在 Linux 操作系统命令完成时被返回至 UCF。这导致下列限制：

- 在命令完成后生成的任何输出都不会被返回至 UCF。例如，如果发出后面跟着 `&` 的命令以便在后台运行该命令，则 UCF 将接收到提供了后台命令的进程标识的操作系统消息，但不会接收到所生成的任何后续输出。同样，可以使用 UCF 来启动守护进程，但您不会看到该进程生成的任何输出。
- UCF 不能与在完成前要求用户进一步进行输入的命令配合使用（例如，启动交互式进程的命令，如 `vi filename`，或者是在被用户停止之前不完成的命令，如 `tail -f filename`）。

由于所有 Linux 命令都是使用已配置的 UCF 用户的登录标识和许可权运行，所以有效命令受 UCF 用户登录的访问权限制。特别是不允许运行 root 用户或超级用户命令。有关更多信息，请参阅第 107 页的『UCF 安全性』。

UCF 命令的示例

以下是从 NetView 中输入的 UCF 命令的一个示例：

```
runcmd sp=mypsname, appl=unix, grep \temp \{ab\}*.c >\t\l\m\p.out
```

在 Linux 计算机上运行的命令是：

```
grep Temp [ab]*.c >TEMP.out
```

Linux 系统命令的输出

成功发出命令后，以下消息将显示在 NetView 屏幕上：

```
=== 正在执行 UNIX 命令 ===
（该命令的任何输出，包括错误消息）
=== UNIX 命令已完成 ===
```

这些消息可能不会同时出现在 NetView 屏幕上。只要 UCF 守护程序接收到命令并将控制返回给 NetView 操作员，正在执行 UNIX 命令消息就立即出现。该命令的任何输出在生成时都会被发送至 NetView，并且可能作为一系列独立消息出现；UNIX 命令已完成消息在 Linux 命令完成并且其 shell 结束后出现。

如果 Linux 命令的输出包含制表符字符，则 Communications Server for Linux 在将输出发送至 NetView 之前会将每个制表符转换为空格字符。否则，不加更改地发送输出。

如果在上一个命令仍在执行时（即在接收到 UNIX 命令已完成消息之前）发出命令，则显示以下消息：

```
=== 命令已排队 ===
```

第二个命令将排队，并在上一个命令完成后执行。

取消命令

UCF 提供了一种取消仍在执行的命令的方法。可使用此方法来停止执行当前命令，或取消不进一步进行输入就不能完成的交互式命令（如 `vi filename`）。此方法相当于使用中断序列（如 **Ctrl + C**）来停止正在终端上运行的进程，或者相当于使用 Linux `kill` 命令来停止进程。

除了取消当前正在执行的命令之外，Communications Server for Linux 还可以取消在该命令之后排队的任何命令。

该命令语法与 Linux 命令的语法相同，但使用字符串 `ux-cancel`，而不是使用命令文本。例如：

```
runcmd sp=mypname, appl=unix, ux-cancel
```

对于每个未完成的命令（当前正在执行的命令以及任何已排队的命令），显示以下消息：

```
=== UNIX 命令已取消 ===
```

此消息指示正在其中运行命令的 Linux shell 已被停止。必要时可以进一步发出 Linux 命令。

如果某个命令在 Linux 计算机上启动守护进程，则不能用 `ux-cancel` 来停止此进程。您可能需要使用 Linux `kill` 命令（在终端上执行此命令，或使用 UCF 执行此命令）来显式地停止此类进程。

如果使用 `ux-cancel` 时没有正在运行的 UCF 命令，则 UCF 显示以下消息：

```
没有未完成的命令
```

在此情况下，`ux-cancel` 命令被忽略。不需要执行任何操作。在上一个命令完成之后但在接收到 **UNIX 命令已完成** 消息之前发出 `ux-cancel` 命令时，会显示此消息。

UCF 安全性

因为 UCF 使远程操作员能够在 Linux 计算机上发出命令并接收这些命令的输出，所以考虑对安全性的影响非常重要。例如，您需要确保操作员不能访问私人信息或发出可能会干扰其他用户的 Linux 命令。

Communications Server for Linux 配置包含作为 UCF 用户的特定 Linux 系统用户名；这必须是 Communications Server for Linux 计算机上的有效登录标识。所有 UCF 命令以此用户的标识运行，因此具有此用户的访问许可权。

其意图是使用 Linux 提供的一般安全性功能部件来限制 UCF 用户可访问的命令，以便只允许访问您认为适于从 UCF 中使用的那些命令。下列准则也许有用：

- UCF 用户名应该是只用于 UCF 的名称；不应该使用一个还用于其他用途的现有登录名。这使得将此用户的特权定义为只包括那些适于 UCF 的特权更容易；它还使您能够标识那些使用 UCF 启动的进程。
- 可能需要限制 UCF 用户可更改其用户标识或组标识的用户和组。特别是一定不能允许 UCF 用户执行下列操作：
 - 成为 root 用户或超级用户。

使用 UCF

- 使用组标识 `system`，这会启用对 `snaadmin` 程序的访问权。（应该使用 `SPCF` 而不是 `UCF` 来访问此程序的功能，如本章前面所述。）

第 10 章 管理 Communications Server for Linux 客户机 / 服务器系统

Communications Server for Linux 可以作为单机系统与所有 SNA 组件和应用程序一起在单个 Linux 系统上运行，也可以作为客户机 / 服务器域的一部分运行。客户机 / 服务器域包含服务器（SNA 节点）和 IBM Remote API Client（它们可以通过服务器访问 SNA 连接）。

在具有多个 Communications Server for Linux 服务器的域中，一个服务器保存 Communications Server for Linux 域配置文件的主副本。此服务器称为主控服务器。您可将域中的其他服务器定义为备份服务器。域配置文件在备份服务器启动或主副本更改时复制到备份服务器，以便所有备份服务器保存最新信息的副本。

Remote API Client 可以是运行 AIX、Linux、Linux for pSeries、Linux for System z 或 Microsoft Windows 的计算机。

服务器和客户机使用 TCP/IP 来通过 Communications Server for Linux 域进行通信。客户机可同时访问一个或多个服务器，也可在需要时运行并发的应用程序。有关客户机 / 服务器配置的联网需求的信息，请参阅第 111 页的『IP 联网需求』。

AIX, LINUX

对于 AIX 或 Linux 上的 Remote API Client，您必须提供有关 Communications Server for Linux 网络和服务器的信息。有关此功能的信息以及有关在客户机上启用和禁用 Communications Server for Linux 软件的指示信息，请参阅第 126 页的『在 AIX 或 Linux 上管理 Remote API Client』。

Communications Server for Linux Administration Command Reference 中定义的一些管理命令显式地指出它们可以从 Linux 客户机中发出。对于所有其他命令，如果指定带有服务器名的 **-n**，则只能从客户机发出命令行管理命令。这样的命令与从指定的服务器上发出该命令效果相同。

WINDOWS

对于 Windows 客户机，必须提供可以由 Communications Server for Linux 用来启用客户机软件的信息。如果您计划将可调用 TP 部署在 Windows 客户机上，则还必须提供关于 TP 的信息。有关这些功能的信息以及有关在 Windows 客户机上启用和禁用 Communications Server for Linux 软件的指示信息，请参阅第 113 页的『在 Windows 上管理 Remote API Client』。

更改客户机 / 服务器配置

在安装 Communications Server for Linux 软件（如《*Communications Server for Linux 快速入门*》中所述）时，最初它是以单机方式安装（所有组件都位于一台 Linux 计算机上）。如果您要将 Communications Server for Linux 作为客户机 / 服务器系统运行，则可以将一台服务器配置为主控服务器，并将其他任何服务器配置为备份服务器。（建议您将主控服务器之外的所有服务器配置为备份服务器。）

Communications Server for Linux 包含命令行应用程序 **snanetutil**，可充当客户机 / 服务器域的服务器角色。为此，请在每台服务器上使用下列命令（从主控服务器开始）：

```
sna stop  
snanetutil master_name [domain_name]  
sna start
```

snanetutil 命令中的参数如下所示。

master_name

该服务器所属域中的主控服务器的名称。如果正在将服务器移至现有的域中，则此名称必须与该域中的现有主控服务器的名称匹配。

domain_name

该服务器所属的域的名称。此参数是可选的；如果您未指定此参数，则 Communications Server for Linux 使用缺省域名 `ibmcs_domain`。

要将主控服务器之外的每台服务器都配置为备份服务器，请发出下列命令。您可以在备份服务器自身或主控服务器上执行此操作，但在这两种情况下，Communications Server for Linux 软件均必须在主控服务器上运行。

```
snaadmin add_backup, backup_name=server_name
```

server_name 是要添加为备份服务器的服务器的名称。

您还可以使用 **snanetutil** 程序将服务器移出有域，以便它作为单机系统运行。

注：除非要停止将 Communications Server for Linux 作为客户机 / 服务器系统运行，并仅仅将其用作单机节点，否则请勿使用此选项。如果从现有的域中除去所有服务器，则留在该域中的任何客户机都无法访问 SNA 资源。

要将服务器移到域外部以使其作为单机系统运行，请使用下列命令：

```
snanetutil -d
```

将客户机移到另一个域中

snanetutil 程序允许在不同的客户机 / 服务器域之间移动服务器。如果要在各个域之间移动客户机，则需要通过修改客户机配置来执行此操作。

在 Windows 上每台将要移动的 Remote API Client 上，使用客户机配置实用程序将 *domain* 参数更改为与新域名匹配。有关更多信息，请参阅第 115 页的『Windows 上的 Remote API Client 配置』。

在 AIX 或 Linux 上每台将要移动的 Remote API Client 上，将客户机网络数据文件的 Configuration 部分中的 *domain* 条目更改为与新域名匹配。有关更多信息，请参阅第 126 页的『客户机网络数据文件 (sna_clnt.net)』。

IP 联网需求

Remote API Client 可使用 TCP/IP 与 Communications Server for Linux 服务器进行通信，或使用 HTTPS 通过 WebSphere 服务器来与这些服务器进行通信。有关使用 HTTPS 连接的更多详细信息，请参阅第 112 页的『对 Remote API Client 的 HTTPS 访问』。

在可以运行 Remote API Client 之前，必须在网络中的客户机和服务器上配置 TCP/IP 端口地址。如果在使用缺省端口指定时遇到问题，则可能需要解决冲突，如『设置 IP 端口号』所述。

另外，您可能希望设置客户机，以便在客户机结束使用 Communications Server for Linux 时自动断开 TCP/IP 连接，如第 112 页的『LAN 访问超时』所述。

设置 IP 端口号

Communications Server for Linux 使用 TCP/IP 和 UDP/IP 通信来通过 LAN 发送客户机 / 服务器数据。缺省情况下，它使用端口号 1553 来进行这两种类型的通信。对于大多数的安装，此端口号应该是合适的；不需要对其进行更改。

如果在启用 Communications Server for Linux 软件时遇到问题，请检查错误日志文件，以了解是否有任何消息指示 Communications Server for Linux 使用的端口号与另一程序使用的端口号冲突。如果找到这样的消息，请执行下列步骤：

1. 检查发生错误的计算机上的 **/etc/services** 文件，了解是否列示了另一个使用端口号 1553 来进行 TCP/IP 或 UDP/IP 通信的程序。如果有的话，首先尝试将该程序更改为使用另一个端口。
2. 如果无法执行此操作，或者没有列示任何使用端口 1553 的程序，则查找另一个在文件中未列示为正在由任何程序使用的端口号。检查同一个域中的所有其他 Communications Server for Linux 计算机上的 **/etc/services** 文件，以确保在任何其他计算机上都未使用该端口号。
3. 在域中每台计算机上的 **/etc/services** 文件中，按以下格式添加两行：

```
sna-cs      nnnn/tcp
sna-cs      nnnn/udp
```

nnnn 项是新端口号。在 Communications Server for Linux 域中的所有计算机上，必须将此端口号设置为相同的值。

- 4.

WINDOWS

如果 Communications Server for Linux 域包含 Windows 客户机，则将这两行添加到每台 Windows 计算机上的 **services** 文件中。**services** 文件与 Linux 文件具有相同的格式，并且通常存储在 Windows TCP/IP 软件的主目录中；如有必要，请参阅 Windows TCP/IP 文档以了解更多信息。

5. 重新启用 Communications Server for Linux 服务器和 Remote API Client 软件。

LAN 访问超时

如果客户机正在通过需要支付连接费用的网络与 Communications Server for Linux 服务器通信，则您可能想确保在客户机上的应用程序停止使用 Communications Server for Linux 资源后自动断开客户机的 TCP/IP 连接。这并不会自动禁用客户机上的 SNA 软件；它仍保持活动状态，如果某个应用程序以后需要访问服务器，则该软件将尝试重新建立与服务器的联系。

`lan_access_timeout` 参数（对于 AIX 或 Linux 上的 Remote API Client，位于 `sna_clnt.net` 文件中；对于 Windows 上的 Remote API Client，位于注册表中）使您能够在客户机上禁用 SNA 软件。当客户机上在指定的时间内没有发生下列任何事件时，TCP/IP 连接将断开：

- APPC 或 CPI-C 对话活动（或尝试启动对话）
- LUA 会话活动
- 来自 Windows 客户机的 CSV TRANSFER_MS_DATA 动词
- MS 动词（仅限于 Linux 客户机）
- NOF 动词（`query_central_logger` 或 `query_node_all` 动词除外）
- 管理命令（下列事件除外，它们不会导致客户机重新启动连接）：
 - 客户机记录的错误或审计消息（这些消息以本地方式记录在客户机上，即使正在使用中央记录亦如此）
 - 管理命令 `query_central_logger` 或 `query_node_all`（这些命令返回 TCP/IP 连接断开前获得的信息，因此可能与 LAN 的当前状态不匹配）
 - NOF 动词 `query_central_logger` 或 `query_node_all`（与等效的管理命令相同）

特别是如果启用了 SNA 软件，但未在指定的超时时间内在客户机上启动任何 Communications Server for Linux 应用程序，则 TCP/IP 连接将断开。

TCP/IP 连接断开后，当发生这些事件中的一个事件时，客户机将重新开始尝试与服务器联系，如第 126 页的『客户机网络数据文件（`sna_clnt.net`）』或第 117 页的『Servers』中的 * 和 `servername` 参数描述所述。

TCP/IP 连接断开后，就不能接受此客户机上的被调用 TP 的入局 Attach；Attach 将被拒绝，就像是目标系统不处于活动状态一样。这表示如果在客户机上没有运行任何其他应用程序，并且 TCP/IP 连接已超时，则客户机上自动启动的 TP 将不可用。但是，客户机上由操作员启动的 TP 是始终可以使用的，这是因为该 TP 发出的 `Receive_Allocate` 动词将重新建立 TCP/IP 连接。

对 Remote API Client 的 HTTPS 访问

如果您将要运行客户机 / 服务器系统，在这个系统中，Remote API Client 使用 HTTPS 连接至 Communications Server for Linux 服务器，则将需要运行 WebSphere Application Server 的计算机，以提供从这些客户机至服务器的 HTTPS 访问。有关如何安装和配置此服务器的指示信息，请参阅《Communications Server for Linux 快速入门》。

如果您将新服务器添加至 Communications Server for Linux 域，并想让 Remote API Client 能够使用 HTTPS 访问这些服务器，则将需要更新 WebSphere 服务器配置文件以便包含这些服务器。此文件命名为 **snahttpsrv.cfg**，并存储在 WebSphere 服务器上由 **USER_INSTALL_ROOT** 环境变量指定的目录中。如果您不确定其所在位置，则执行下列步骤。

1. 启动 WebSphere 管理控制台。
2. 在管理控制台菜单栏中，选择“环境”、“管理 WebSphere 变量”。
3. 在此列表中查找 **USER_INSTALL_ROOT** 变量，并记下它的值（这是 WebSphere 服务器上的目录路径）。环境变量的列表可能分为两个或多个页面，所以您可能需要使用“下一页”按钮来滚动浏览此列表。

使用文本编辑器来编辑配置文件，以包括 Remote API Client 使用 HTTPS 可访问的 Communications Server for Linux 服务器的列表。您必须采用下列格式在文件的单独行上指定每个服务器：

```
server=servername.domainname.com
```

在将访问新服务器的每个 Remote API Client 上，您还需要将新服务器名添加至客户机网络数据文件（或 Windows 客户机的 Windows 注册表）中的服务器列表中。请参阅本章稍后部分的小节以了解适当的客户机类型。

在 Windows 上管理 Remote API Client

WINDOWS

Communications Server for Linux 使运行 Microsoft Windows 的机器能够充当 Communications Server for Linux 域中的客户机。Communications Server for Linux 客户机软件包含与 Microsoft Host Integration Server、Windows 开放式系统体系结构（WOSA）以及 IBM Personal Communications and Communications Server for Windows 提供的接口相兼容的 API 库。这使得为这些实现编写的应用程序不加更改就可以在 Windows 上的 Remote API Client 上运行。

Windows 上的 Remote API Client 支持下列 WOSA API：

- Windows APPC
- Windows CPI-C
- Windows LUA
- Windows CSV

有关 Windows SNA API 的更多信息，请参阅随 Microsoft Host Integration Server 附带提供的文档。

SNA 网络信息以及 Windows 上的 Remote API Client 所需的其他信息均保存在 Windows 注册表中。

您必须先启用客户机，然后才可以在该客户机上使用 Communications Server for Linux 应用程序或仿真程序。有关更多信息，请参阅第 114 页的『在 Windows 上启用 Remote API Client』。启用客户机后，它将通过 TCP/IP 网络与运行 Communications Server for Linux 的服务器联系，以访问 Communications Server for Linux 功能。

在 Windows 上管理 Remote API Client

客户机的操作还受 Windows 注册表中的信息控制。Windows 注册表包含关于下列各项的信息：

- 特定于 Windows 上的 Remote API Client 的配置信息
- 客户机可以访问的服务器
- 在客户机上运行的应用程序的记录和跟踪选项
- 在客户机上运行的 CPI-C 和 CSV 应用程序的其他选项
- 可以在客户机上运行的可调用 TP (APPC 或 CPI-C)

您还可以使用客户机配置实用程序来修改最常用的参数（这是修改参数的首选方法）。有关更多信息，请参阅第 115 页的『Windows 上的 Remote API Client 配置』。

注：如果客户机使用 HTTPS 来访问其服务器，则您需要修改客户机配置，以指定这些服务器的名称以及用于提供对服务器的 HTTPS 访问权的 WebSphere 服务器的名称，然后才可以使用客户机。有关更多信息，请参阅第 115 页的『Windows 上的 Remote API Client 配置』。

在 Windows 上启用 Remote API Client

Windows 上的 Remote API Client 作为 Windows 服务运行。安装程序将其配置为在计算机启动时自动启动。如果有必要的话，可以通过下列其中一个方法手工启动它。

- 在控制面板中，从“管理工具”中的“服务” applet 启动客户机服务。
- 从命令窗口或从“开始/运行”图标输入 `net start sxclient`。

然后，客户机使用 Windows 注册表中的信息（已使用客户机配置实用程序定义，并在第 115 页的『Windows 上的 Remote API Client 配置』中描述），以找到运行 Communications Server for Linux 的服务器。

在 Windows 上查看 Remote API Client 的状态

客户机监视器在系统任务栏中放置一个图标，当您鼠标指针移至该图标上时，该图标将显示客户机的状态。此监视器已被设置成在计算机启动时自动运行，但是，也可以根据需要从“开始”菜单手工运行它。状态为下列其中一种状态：

未活动 客户机尚未启动。

未连接 客户机已启动，但尚未与服务器取得联系（或已失去联系）。

服务器名

客户机已连接至指定的服务器。

在 Windows 上禁用 Remote API Client

在禁用客户机之前，确保客户机上的所有 Communications Server for Linux 应用程序（3270 和 5250 仿真程序或使用 Communications Server for Linux API 的应用程序）都已停止。

要禁用客户机，请通过下列其中一个方法来停止客户机服务。

- 在控制面板中，从“管理工具”中的“服务” applet 停止客户机服务。
- 从命令窗口或从“开始/运行”图标输入 `net stop sxclient`。

在运行 Windows 终端服务的计算机上，这表示不允许所有用户使用客户机。

Windows 上的 Remote API Client 配置

在 Windows 上的 Remote API Client 上，配置信息保存在 Windows 注册表中。该注册表包含 SNA 网络信息（与 AIX 或 Linux 上的 Remote API Client 上保存在客户机网络数据文件中的信息类似）。它还包含一些特定于 Windows 上的 Remote API Client 的其他配置信息。

注：CPI-C 应用程序的配置信息（本地 TP 名和本地 LU 别名）可通过环境变量或在注册表中指定。如果您要使用 Windows Terminal Server 并需要使用不同本地 LU 来运行同一应用程序的多个副本，则可能需要使用环境变量。有关更多详细信息，请参阅第 124 页的『Appl_Name』。

客户机配置实用程序提供了修改最常用客户机配置参数的简单方法，它是修改这些参数的首选方法。要使用此程序，请运行位于 IBM 远程 API 程序组中的**配置实用程序**。该程序显示的“配置”窗口与初始安装过程中显示的“配置”窗口相同。有关如何修改这些配置参数的更多详细信息，请参阅《Communications Server for Linux 快速入门》中的『Windows 上的 Remote API Client 安装』章节。

注：在更改客户机配置参数后，您需要停止并重新启动该客户机，然后所作的更改才会生效。有关如何执行此操作的详细信息，请参阅第 114 页的『在 Windows 上使用 Remote API Client』和第 114 页的『在 Windows 上启用 Remote API Client』。

注册表中下列键的子键下配置的值包含该信息：

```
\\HKEY_LOCAL_MACHINE\SOFTWARE\SNA Client\SxClient\Parameters
```

每个注册表子键的可能值如下：

```
Configuration
domain = domain_name
invoked_tps = YES | NO
lan_access_timeout = nn
broadcast_attempt_count = nn
server_lost_timeout = nn
  client_start_timeout = nn

Servers
Server1 = * | [ webservername : [ portnumber : ] ]servername1
Server2 = [ webservername : [ portnumber : ] ]servername2
.
.
.
Server9 = [ webservername : [ portnumber : ] ]servername9

Logging
exception_logging_enabled = YES | NO
audit_logging_enabled = YES | NO
log_directory = directory
error_file = error_filename
backup_error_file = backup_error_filename
error_file_wrap_size = error_file_size
audit_file = audit_filename
backup_audit_file = backup_audit_filename
audit_file_wrap_size = audit_file_size
succinct_errors = YES | NO
succinct_audits = YES | NO

API_tracing
file1 = trace_filename_1
file2 = trace_filename_2
flip_size = filesize
truncation_length = length
```


在 Windows 上管理 Remote API Client

```
all_api = YES | NO
appc = YES | NO
cpic = YES | NO
csv = YES | NO
rui = YES | NO
nof = YES | NO

CS_tracing
file1 = cs_trace_filename_1
file2 = cs_trace_filename_2
flip_size = filesize
admin_msg = YES | NO
datagram = YES | NO
data = YES | NO
send = YES | NO
receive = YES | NO

Internal_tracing
file1 = internal_trace_filename_1
file2 = internal_trace_filename_2
flip_size = filesize
trace_level = nn
trace_flushing = YES | NO

Appl_Name
APPCTPN = tp_name
APPCLLU = lu_name

CSV_data
CSVTBLG = table_G_filename
```

注: domain = *domain_name* 值是注册表中唯一的必需值。

下列各节说明配置参数。在参数值是 YES 或 NO 的地方，任何以 Y 或 y 开头的字符串都被解释为 YES，任何以 N 或 n 开头的字符串都被解释为 NO。

Configuration

Configuration 子键包含客户机的配置信息，如下所示：

domain 此值的注册表数据类型是 REG_SZ。

domain_name 值指示 Communications Server for Linux LAN 的域名，该名称在客户机安装期间指定。此行是必需的。

invoked_tps

此值的注册表数据类型是 REG_SZ。

指定下列其中一个值：

YES 此客户机用来运行被调用 TP（发出 RECEIVE_ALLOCATE 的 APPC TP，或者是发出 Accept_Conversation 或 Accept_Incoming 的 CPI-C 应用程序）。在此情况下，可能还需要在此客户机上定义该 TP。有关更多信息，请参阅第 75 页的『定义 TP』或第 151 页的附录 B，『从命令行配置可调用 TP』。

NO 此客户机并非用来运行被调用 TP。

此行是可选的。如果未指定此行，则缺省值是 NO。

lan_access_timeout

此值的注册表数据类型是 REG_SZ。

指定当客户机上没有应用程序使用 Communications Server for Linux 资源时应该让客户机到服务器的 IP 或 HTTPS 连接保持活动的时间（以秒计）。有关更多信息，请参阅第 112 页的『LAN 访问超时』。

有效范围是 0–65535。最短超时是 60 秒（更小的值被上舍入到 60 秒）。要更快地释放该连接，请禁用客户机。

此参数是可选的。如果未指定此参数，则缺省值是没有超时，只要客户机在运行，该连接就保持活动。

broadcast_attempt_count

此值的注册表数据类型是 REG_SZ。

如果客户机使用广播方法来与服务器联系（由『Servers』描述的 * 条目指定），则此参数指定要在一次服务器联系尝试中进行的广播的最大数目。

有效范围是 1–65535。最小值是 1；如果指定更大的值，则客户机每 10 秒钟重试一次，直到与服务器取得联系或直到达到此计数为止。如果达到此计数，但没有与服务器取得联系，则客户机接着尝试访问指定的服务器（如『Servers』所述）。

此参数是可选的。如果未指定此参数，则缺省值是 5。

server_lost_timeout

此值的注册表数据类型是 REG_SZ。

如果客户机与服务器失去联系并需要重新连接，或者它无法使用广播或指定的服务器来与服务器取得联系（如『Servers』所述），则此参数指定客户机在尝试与服务器联系前要等待的时间（以秒计）。如果客户机与服务器失去联系，则 Communications Server for Linux 并不是等待完整的超时周期，而是在长度介于 5 秒与指定超时之间的随机时间段过后进行重试；这是为了避免由于大量客户机同时尝试与服务器联系而造成网络流量爆增。

此参数是可选的。有效范围是 5–65535。如果未指定此项，则缺省值是 200（秒）。

client_start_timeout

此值的注册表数据类型是 REG_DWORD。

指定当客户机启动并尝试与服务器联系时应用程序要等待的时间（以秒计）。有效值介于 0 与 300 之间；此范围以外的值将被强制设置为在此范围内。缺省值是 10 秒。

此参数可用于控制一些事件，这些事件是在应用程序和客户机都被配置为随系统启动而启动（通过位于“启动”文件夹中，或通过作为自动启动的服务）时发生的。应用程序将等待此字段中指定的秒数，以便客户机能够首先启动。这样，客户机就可以在应用程序由于缺少所需的资源而失败前连接至服务器以提供那些资源。

Servers

Servers 子键包含有关客户机为了获取配置信息而应该联系的 Communications Server for Linux 配置服务器的信息。此列表应包含与该客户机处于同一域中的主控配置服务器与所有备份服务器的名称。有关配置主控服务器与备份服务器的信息，请参阅第 47 页的『配置客户机/服务器功能』。

注：此子键的格式和含义取决于该客户机是与其服务器在同一个专用网络，还是如下所述通过公用网络使用 HTTPS 进行连接。

Server1

此值的注册表数据类型是 REG_SZ。

输入星号 (*) 或服务器名:

- 要指示客户机与其服务器在同一专用网络以及客户机应该通过将 UDP 广播消息发送至它的 TCP/IP 子网中的所有计算机（如果客户机包含多块 LAN 适配卡，则将 UDP 广播消息发送至它可以访问的所有子网中的全部计算机）来尝试查找运行 Communications Server for Linux 的服务器，请指定 *。

客户机每 10 秒钟重试广播一次，直到达到 *broadcast_attempt_count* 参数指定的尝试次数为止，或者直到与服务器取得联系为止。如果在与服务器取得联系之前达到 *broadcast_attempt_count* 指定的限制，则客户机接着尝试使用发送至一台或多台指定的服务器的定向消息（这些服务器由文件中接下来的各行指定）。

- 在客户机与其服务器位于同一专用网络但却无法使用 UDP 广播访问到任何服务器，并且必须使用定向消息的情况下，指定客户机应该尝试联系的第一台服务器的名称。*webservername* 和 *portnumber* 参数将不被使用，因此您不应该指定它们。这在下列情况下适用：
 - 当 Communications Server for Linux LAN 跨多个 TCP/IP 子网，并且在客户机使用 UDP 可以访问的任何 TCP/IP 子网中都没有 Communications Server for Linux 服务器时。
 - 当客户机上未安装 UDP 支持时。

在其他情况下，UDP 广播的使用是可选的；要指定不应该尝试广播，请指定第一台服务器的名称，而不是指定 *。

- 如果客户机使用 HTTPS 来访问其服务器，则 UDP 广播不受支持。在此情况下，请采用下列格式指定提供 HTTPS 支持的 WebSphere 服务器的名称以及 Communications Server for Linux 服务器的名称：

webservername : *servername1*

这假定将 WebSphere 设置为使用缺省端口 443 进行 HTTPS 连接。如果您的网络管理员已将 WebSphere 配置为使用其他端口号，则采用下列格式来包含该端口号：

webservername : *portnumber* : *servername1*

有关配置 WebSphere 以支持 HTTPS 连接的更多详细信息，请参阅《Communications Server for Linux 快速入门》。

Server2–Server9

此值的注册表数据类型是 REG_SZ。

按优先顺序指定客户机应该联系的其他 Communications Server for Linux 配置服务器的名称。使用与 *Server1* 相同的格式。

如果客户机已尝试使用 UDP 广播来联系服务器（或已尝试联系 *Server1* 中指定的服务器），但未接收到响应，则它接着尝试使用定向消息来联系 *Server2* 指定的服务器。如果此操作失败，则它尝试联系 *Server3* 指定的服务器，依此类

推。尽管这些服务器名是可选的，但如果用于找到服务器的广播方法失败或者 *Server1* 指定的服务器不可用，这些服务器名就提供了一种备份机制。

如果客户机尝试了列示的所有服务器，但没有一次成功，则它等待由 *server_lost_timeout* 参数指定的秒数，然后重新开始尝试联系服务器的过程（使用 UDP 广播或使用列示的第一台服务器）。

不能将参数 *Server2–Server9* 设置为 * 以指示使用 UDP 广播。由于 * 值必须位于文件中的任何服务器名之前，所以，只有 *Server1* 参数可用来指示使用广播。

Logging

Logging 子键指定客户机的记录选项。这些选项可用来指定客户机记录设置，这些设置将覆盖对整个域指定的记录选项。有关指定域记录选项的更多信息，请参阅第 49 页的『配置记录』。

如果启用了中央记录，则所有日志消息都将被写至服务器上的中央文件。在此情况下，只使用在此指定的 *exception_logging_enabled* 和 *audit_logging_enabled* 参数；其余参数被忽略。

记录选项的指定方法如下：

exception_logging_enabled

此值的注册表数据类型是 REG_SZ。

将此参数设置为下列其中一个值：

YES 记录异常消息。

NO 不记录异常消息。

此参数是可选的。如果未指定此参数，则客户机使用全局域设置来确定是否记录异常消息。（初始缺省值是记录异常消息。）

audit_logging_enabled

此值的注册表数据类型是 REG_SZ。

将此参数设置为下列其中一个值：

YES 记录审计消息。

NO 不记录审计消息。

此参数是可选的。如果未指定此参数，则客户机使用全局域设置来确定是否记录审计消息。（初始缺省值是记录审计消息。）

log_directory

此值的注册表数据类型是 REG_SZ。

此客户机上存储日志文件所用的目录的完整路径。所有日志文件和备份日志文件（在下列参数中指定）都存储在此目录中。

此参数是可选的。如果未指定此参数，则文件存储在 Windows 安装目录中。

error_file

此值的注册表数据类型是 REG_SZ。

将错误消息写至的文件的名称。此参数是可选的。如果未指定此参数，则缺省名称是 **sna.err**。

要将错误和审计消息记录到单个文件中，请对此参数和 *audit_file* 参数指定同一个文件名。

backup_error_file

此值的注册表数据类型是 REG_SZ。

备份错误日志文件的名称。当错误日志文件达到 *error_file_wrap_size* 指定的大小时，Communications Server for Linux 便将它的内容复制到备份文件（覆盖任何现有文件），然后清除错误日志文件。

此参数是可选的。如果未指定此参数，则缺省名称是 **bak.err**。

要将错误和审计消息记录到单个文件中，请对此参数和 *backup_audit_file* 参数指定同一个文件名。

error_file_wrap_size

此值的注册表数据类型是 REG_DWORD。

error_file 指定的日志文件的最大大小。当写至文件的消息导致文件大小超出此限制时，Communications Server for Linux 便将日志文件的当前内容复制到备份日志文件，然后清除日志文件。这表示错误日志文件占用的最大磁盘空间容量大约是 *error_file_wrap_size* 参数值的两倍。

此参数是可选的。如果未指定此参数，则缺省值是 1000000（字节）。如果正在将错误和审计消息记录到同一个文件中，则必须将此参数设置为与 *audit_file_wrap_size* 参数具有相同的值。

audit_file

此值的注册表数据类型是 REG_SZ。

将审计消息写至的文件的名称。此参数是可选的。如果未指定此参数，则缺省名称是 **sna.aud**。

要将错误和审计消息记录到单个文件中，请对此参数和 *error_file* 参数指定同一个文件名。

backup_audit_file

此值的注册表数据类型是 REG_SZ。

备份审计日志文件的名称。当审计日志文件达到 *audit_file_wrap_size* 指定的大小时，Communications Server for Linux 便将它的内容复制到备份文件（覆盖任何当前文件），然后清除审计日志文件。

此参数是可选的。如果未指定此参数，则缺省名称是 **bak.aud**。

要将错误和审计消息记录到单个文件中，请对此参数和 *backup_error_file* 参数指定同一个文件名。

audit_file_wrap_size

此值的注册表数据类型是 REG_DWORD。

audit_file 指定的日志文件的最大大小。当写至文件的消息导致文件大小超出此限制时，Communications Server for Linux 便将日志文件的当前内容复制到备份日志文件，然后清除日志文件。这表示审计日志文件占用的最大磁盘空间容量大约是 *audit_file_wrap_size* 参数值的两倍。

此参数是可选的。如果未指定此参数，则缺省值是 1000000（字节）。如果正在将错误和审计消息记录到同一个文件中，则必须将此参数设置为与 *error_file_wrap_size* 参数具有相同的值。

succinct_errors

此值的注册表数据类型是 REG_SZ。

指定在错误日志文件中是使用简要记录还是使用详细记录。此设置既应用于异常日志也应用于问题日志。可以指定下列其中一个值：

YES 使用简要记录：日志文件中的每条消息都包含消息头信息的摘要（如消息号和日志类型）以及消息文本字符串和参数。要获取有关日志原因以及所需执行的任何操作的更多详细信息，可以在运行 Linux 的计算机上使用 **snahelp** 实用程序。

NO 使用详细记录：日志文件中的每条消息都包含消息头信息的完整列表、消息文本字符串和参数以及有关日志原因和所需执行的任何操作的其他信息。

此参数是可选的。如果未指定此参数，则缺省值是从对主控服务器发出的上一个 **set_global_log_type** 命令中获取的（或者是使用 Motif 管理程序设置的）。在发出任何 **set_global_log_type** 命令之前，初始缺省值是使用简要记录。

如果正在使用中央记录，则来自所有计算机的消息的简要或详细记录选项是由作为中央记录器的服务器上的此参数的设置确定的；此设置可以来自 **set_global_log_type** 命令，也可以来自对该服务器发出的用于覆盖缺省值的 **set_log_type** 命令。

succinct_audits

此值的注册表数据类型是 REG_SZ。

指定在审计日志文件中是使用简要记录还是使用详细记录。允许的值及其含义与 *succinct_errors* 参数相同。

API_tracing

API_tracing 子键指定在客户机上运行的应用程序的 API 跟踪选项。有关跟踪的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。跟踪选项的指定方法如下：

file1 此值的注册表数据类型是 REG_SZ。

跟踪文件的全路径名，或者如果要跟踪信息写至两个文件的话，则是第一个跟踪文件的全路径名（请参阅 *file2* 参数的描述）。

如果要启用 API 跟踪，则此参数是必需的。

file2 此值的注册表数据类型是 REG_SZ。

第二个跟踪文件的全路径名。此参数是可选的；要指示应将跟踪信息写至一个文件而不是两个文件，请不要包括此行。

如果同时指定 *file1* 和 *file2*，则将把跟踪信息写入两个文件。当第一个文件达到 *flip_size* 参数指定的大小时，将清除第二个文件，并接着将跟踪信息写至第二个文件。当此文件达到 *flip_size* 指定的大小时，将清除第一个文件，并接着将跟踪信息写至第一个文件。这确保可以长时间地进行跟踪，而不会使用过多的磁盘空间；所需的最大空间容量大约是 *flip_size* 参数值的两倍。

flip_size

此值的注册表数据类型是 REG_DWORD。

跟踪文件的最大大小。如果指定了两个文件名，则当前文件达到此大小时，跟踪操作将在这两个文件之间切换。如果只指定一个文件名，则此参数将被忽略；文件大小不受限制。

此参数是可选的。如果未指定此参数，则缺省值是 1000000（字节）。

truncation_length

此值的注册表数据类型是 REG_DWORD。

对每条消息写至跟踪文件的信息的最大长度（以字节计）。如果消息的长度超出此限制，则 Communications Server for Linux 只将消息的开头写入跟踪文件，并废弃位于 *truncation_length* 之后的数据。这使您能够记录每条消息的最重要的信息，而避免文件被过长的消息充满。

此参数是可选的。如果未指定此参数，则 Communications Server for Linux 不截断消息（将每条消息的所有数据都写入文件）。

all_api

此值的注册表数据类型是 REG_SZ。

要跟踪所有 API 的消息，请将此参数设置为 YES。在此情况下，Communications Server for Linux 忽略从 *appc* 到 *nof* 的参数。

要对所有 API 禁用跟踪，请将 *all_api* 以及从 *appc* 到 *nof* 的所有参数都设置为 NO。

要仅跟踪特定 API 的消息，请将 *all_api* 设置为 NO，并使用从 *appc* 到 *nof* 的参数来指示要跟踪的 API。

此参数是可选的。如果未指定此参数，则缺省值是 NO。

appc

此值的注册表数据类型是 REG_SZ。

要跟踪 APPC API 消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。如果将 *all_api* 参数设置为 YES，则将忽略此参数，并跟踪 APPC 消息。

cpic

此值的注册表数据类型是 REG_SZ。

要跟踪 CPI-C API 消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。如果将 *all_api* 参数设置为 YES，则将忽略此参数，并跟踪 CPI-C 消息。

csv

此值的注册表数据类型是 REG_SZ。

要跟踪 CSV API 消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。如果将 *all_api* 参数设置为 YES，则将忽略此参数，并跟踪 CSV 消息。

rui

此值的注册表数据类型是 REG_SZ。

要跟踪 LUA RUI 消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。如果将 *all_api* 参数设置为 YES，则将忽略此参数，并跟踪 LUA RUI 消息。

nof

此值的注册表数据类型是 REG_SZ。

要跟踪 NOF API 消息，请将此参数设置为 YES；否则，将其设置为 NO。Windows 客户机上的应用程序不直接使用 NOF 消息，但 Communications Server for Linux 组件以内部方式使用此消息以获取配置信息。

此参数是可选的。如果未指定此参数，则缺省值是 NO。如果将 *all_api* 参数设置为 YES，则将忽略此参数，并跟踪 NOF 消息。

CS_tracing

CS_tracing 子键指定客户机 / 服务器跟踪（对客户机与 Communications Server for Linux 服务器之间的消息进行的跟踪）的选项。有关跟踪的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。跟踪选项的指定方法如下：

file1 此值的注册表数据类型是 REG_SZ。

跟踪文件的全路径名，或者如果要将跟踪信息写至两个文件的话，则是第一个跟踪文件的全路径名（请参阅 *file2* 参数的描述）。

如果要启用客户机 / 服务器跟踪，则此参数是必需的；还需要设置 *trace_flags* 参数。

file2 此值的注册表数据类型是 REG_SZ。

第二个跟踪文件的全路径名。此参数是可选的；要指示应将跟踪信息写至一个文件而不是两个文件，请不要包括此行。

如果同时指定 *file1* 和 *file2*，则将把跟踪信息写入两个文件。当第一个文件达到 *flip_size* 参数指定的大小时，将清除第二个文件，并接着将跟踪信息写至第二个文件。当此文件达到 *flip_size* 指定的大小时，将清除第一个文件，并接着将跟踪信息写至第一个文件。这确保可以长时间地进行跟踪，而不会使用过多的磁盘空间；所需的最大空间容量大约是 *flip_size* 参数值的两倍。

flip_size

此值的注册表数据类型是 REG_DWORD。

跟踪文件的最大大小。如果指定了两个文件名，则当前文件达到此大小时，跟踪操作将在这两个文件之间切换。如果只指定一个文件名，则此参数将被忽略；文件大小不受限制。

此参数是可选的。如果未指定此参数，则缺省值是 1000000（字节）。

admin_msg

此值的注册表数据类型是 REG_SZ。

要跟踪与客户机 / 服务器拓扑相关的内部消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。

datagram

此值的注册表数据类型是 REG_SZ。

要跟踪数据报消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。

data

此值的注册表数据类型是 REG_SZ。

要跟踪数据消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。

send

此值的注册表数据类型是 REG_SZ。

要跟踪从客户机发送至服务器的所有数据消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。

receive 此值的注册表数据类型是 REG_SZ。

要跟踪客户机从服务器接收到的所有数据消息，请将此参数设置为 YES；否则，将其设置为 NO。

此参数是可选的。如果未指定此参数，则缺省值是 NO。

Internal_tracing

Internal_tracing 子键指定客户机内部操作跟踪选项。有关跟踪的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。跟踪选项的指定方法如下：

file1 此值的注册表数据类型是 REG_SZ。

跟踪文件的全路径名，或者如果要跟踪信息写至两个文件的话，则是第一个跟踪文件的全路径名（请参阅 *file2* 参数的描述）。

如果要启用内部跟踪，则此参数是必需的；还需要设置 *trace_level* 参数。

file2 此值的注册表数据类型是 REG_SZ。

第二个跟踪文件的全路径名。此参数是可选的；要指示应将跟踪信息写至一个文件而不是两个文件，请不要包括此行。

如果同时指定 *file1* 和 *file2*，则将把跟踪信息写入两个文件。当第一个文件达到 *flip_size* 参数指定的大小时，将清除第二个文件，并接着将跟踪信息写至第二个文件。当此文件达到 *flip_size* 指定的大小时，将清除第一个文件，并接着将跟踪信息写至第一个文件。这确保可以长时间地进行跟踪，而不会使用过多的磁盘空间；所需的最大空间容量大约是 *flip_size* 参数值的两倍。

flip_size

此值的注册表数据类型是 REG_DWORD。

跟踪文件的最大大小。如果指定了两个文件名，则当当前文件达到此大小时，跟踪操作将在这两个文件之间切换。如果只指定一个文件名，则此参数将被忽略；文件大小不受限制。

此参数是可选的。如果未指定此参数，则缺省值是 1000000（字节）。

trace_level

此值的注册表数据类型是 REG_DWORD。

跟踪所包括的详细信息的级别。有效值的范围是从 0（所有跟踪）到 20（没有跟踪）。

此参数是可选的。如果未指定此参数，则缺省值是 20（没有跟踪）。

trace_flushing

此值的注册表数据类型是 REG_SZ。

如果将此参数设置为 YES，则立即将每个跟踪语句写至磁盘。这将大幅降低操作速度，但能确保即使发生崩溃也不会丢失跟踪数据。

此参数是可选的。如果未指定此参数，则缺省值是 NO。

Appl_Name

Appl_Name 子键指定 CPI-C 应用程序选项。

注：可以在环境变量或注册表中指定这些选项。Communications Server for Linux 首先检查环境变量，如果已指定环境变量，则使用此信息；如果未指定环境变量，则仅使用注册表条目。如果您要使用 Windows Terminal Server 并需要使用不同本地 LU 来运行同一应用程序的多个副本，则可能需要使用环境变量。

要在注册表中为一个或多个应用程序设置这些选项，请采用此格式对每个应用程序都提供一个节，并将 *Appl_Name* 变量替换为该应用程序的可执行文件名（不包括 **.exe** 文件扩展名）。

有关 CPI-C 的更多信息，请参阅 *Communications Server for Linux CPI-C Programmer's Guide*。

选项的指定方法如下：

APPCLLU

可以使用 APPCLLU 环境变量（而不用在注册表中）指定此选项。

此值的注册表数据类型是 REG_SZ。

此应用程序使用的本地 LU 的名称。

此参数是可选的。如果未指定此参数，则应用程序将尝试使用缺省 LU（与本地节点的控制点相关联的 LU）。

APPCTPN

可以使用 APPCTPN 环境变量（而不用在注册表中）指定此选项。

此值的注册表数据类型是 REG_SZ。

应用程序的 TP 名。在日志和跟踪文件中，使用此名称来标识应用程序。对于被调用应用程序（发出 *Accept_Conversation* 的应用程序），此名称还用来将入局 *Allocate* 请求中的 TP 名与正确的应用程序相关联；被调用应用程序还可以使用 *Specify_Local_TP_Name* 调用来指定要与入局 *Allocate* 请求匹配的其他名称。

此参数是可选的。如果未指定此参数，则缺省值是 *CPIC_DEFAULT_TPNAME*。

CSV_data

CSV_data 子键指定使用 CSV 接口的应用程序的选项。它仅适用于使用 *CONVERT* 动词来通过用户定义的转换表（G 表）来执行字符转换的应用程序。有关 *CONVERT* 动词的更多信息，请参阅 *Communications Server for Linux CSV Programmer's Guide*。

如果客户机上没有任何应用程序使用此功能，则不需要包括此节。

此节中的唯一选项如下：

CSVTLG

此值的注册表数据类型是 REG_SZ。

包含用户定义的“G 表”转换表的文件的全路径名。如果 CSV 应用程序需要执行“G 表”字符转换，则此参数是必需的（没有缺省值）；否则，此参数是可选的。



在 AIX 或 Linux 上管理 Remote API Client

AIX, LINUX

Remote API Client 可以在 AIX、Linux、Linux for pSeries 或 Linux for System z 上运行。

AIX 或 Linux 上的 Remote API Client 的客户机信息存储在 **sna_clnt.net** 文件中，该文件是在客户机上安装 SNA 软件时创建的。在可以启用客户机软件之前，该文件必须存在。

注：如果客户机使用 HTTPS 来访问其服务器，则您需要修改 **sna_clnt.net** 文件，以指定这些服务器的名称以及用于提供对服务器的 HTTPS 访问权的 WebSphere 服务器的名称，然后才可以使用客户机。有关更多信息，请参阅『客户机网络数据文件（sna_clnt.net）』。

在 AIX 或 Linux 上启用和禁用 Remote API Client

要在 AIX 或 Linux 上启用 Remote API Client 软件，请在命令提示符处输入下列命令：

```
sna start [ -t ]
```

当您安装客户机时，安装实用程序自动更新启动文件 **/etc/rc.sna**（AIX）或 **/etc/rc.d/init.d/snastart**（Linux），以便包含 **sna start** 命令。这可确保在系统启动时自动启动客户机。如果不想让它自动启动，则可以除去或注释掉此行，然后遵循本节中的指示信息来手工启用该软件。

唯一的选项如下：

-t 激活客户机 / 服务器跟踪。这使您能够对客户机尝试连接至服务器时发生的问题进行诊断。如果未使用此选项，则客户机 / 服务器跟踪在所有接口上都是不活动的；您可以根据需要使用命令行管理程序 **snaadmin** 来激活跟踪。

此选项相当于在 Motif 管理程序中选择将所有跟踪设置为打开字段，但此选项不启用 DLC 跟踪。

跟踪会降低 Communications Server for Linux 组件的性能。启用软件后，当不再需要进行跟踪时，可以使用命令行管理程序 **snaadmin** 来停止跟踪。有关跟踪的更多信息，请参阅 *Communications Server for Linux Diagnostics Guide*。

要停止 Remote API Client，请在命令提示符处输入下列命令：

```
sna stop
```

客户机网络数据文件（sna_clnt.net）

sna_clnt.net 文件定义在 AIX 或 Linux 上的 Remote API Client 上可用的 Communications Server for Linux 设施以及该客户机可以访问的服务器。（有关 Windows 客户机上的等效文件的信息，请参阅第 109 页的第 10 章，『管理 Communications Server for Linux 客户机 / 服务器系统』。）

它还包含有关设置 Communications Server for Linux 用于进行客户机 / 服务器通信的 IP 端口号的信息。缺省端口号在大多数情况下应该是合适的；仅当 Communications Server for Linux 记录了指示与同一台计算机上的另一程序发生端口号冲突的错误消息时才需要参阅此信息。

客户机不保存域配置文件或 SNA 网络数据文件的副本；它只保存访问 Communications Server for Linux LAN 中的服务器所需的信息，并且依靠服务器提供必需的配置信息。

所需的 SNA 网络信息保存在 **sna_clnt.net** 文件中，该文件存储于目录 **/etc/sna**（对于 AIX）或 **/etc/opt/ibm/sna**（对于 Linux）。此文件是在客户机安装过程中设置的；它是一个 ASCII 文本文件，以后可以根据需要使用标准文本编辑器对其进行修改。

注：在此文件中更改参数后，您需要停止并重新启动该客户机，然后所作的更改才会生效。有关如何执行此操作的详细信息，请参阅第 126 页的『在 AIX 或 Linux 上启用和禁用 Remote API Client』。

此文件的内容如下：

```
domain = domain_name
invoked_tps = YES | NO
lan_access_timeout = nn
broadcast_attempt_count = nn
server_lost_timeout = nn
*
[ webservername : [ portnumber : ] ]servername1
[ webservername : [ portnumber : ] ]servername2
.
.
.
```

以下列表描述了此文件的每一行中的参数：

domain domain_name 参数值指示 Communications Server for Linux LAN 的域名；在客户机安装期间，此名称被设置为 **ibmcs_domain**。此行是必需的。

invoked_tps

如果此客户机用来运行被调用 TP（发出 **RECEIVE_ALLOCATE** 动词的 APPC TP，或者是发出 **Accept_Conversation** 或 **Accept_Incoming** 动词的 CPI-C 应用程序），则指定 **invoked_tps = YES**。在此情况下，可能还需要在此客户机上定义该 TP。有关更多信息，请参阅第 75 页的『定义 TP』。

如果此客户机并非用来运行被调用 TP，则指定 **invoked_tps = NO**。

此行是可选的；如果未包括此行，则缺省值为 **NO**。

lan_access_timeout

指定当客户机上没有应用程序使用 Communications Server for Linux 资源时应该让客户机到服务器的 IP 或 HTTPS 连接保持活动的时间（以秒计）。有关更多信息，请参阅第 112 页的『LAN 访问超时』。

最短超时是 60 秒（更小的值被上舍入到 60 秒）。要更快地断开该连接，请在客户机上禁用 Communications Server for Linux 软件。

要指示无超时，以便只要 Communications Server for Linux 软件在客户机上运行就保持连接处于活动状态，请不要指定此参数。

此参数是可选的；如果未指定此参数，则缺省值是没有超时。

broadcast_attempt_count

如果客户机使用广播方法来与服务器联系（由 * 条目指定），则此参数指定要在一次服务器联系尝试中进行的广播的最大数目。最小值是 1；如果指定更大的值，则客户机每 10 秒钟重试一次，直到与服务器取得联系或直达到此计数为止。如果达到此计数，而没有与服务器取得联系，则客户机接着尝试与指定的服务器联系。

此参数是可选的；如果未指定此参数，则缺省值是 5。

server_lost_timeout

如果客户机与服务器失去联系并需要进行重新连接，或者它无法使用广播或指定的服务器来与服务器取得联系，则此参数指定客户机在开始或重新开始尝试与服务器联系前要等待的时间（以秒计）。如果客户机与服务器失去联系，则 Communications Server for Linux 并不是等待完整的超时周期，而是在长度介于 5 秒与指定超时之间的随机时间段过后进行重试；这是为了避免由于大量客户机同时尝试与服务器联系而造成网络流量爆增。

此参数是可选的；如果未指定此参数，则缺省值是 200 秒。

- * 此行指示客户机与其服务器位于同一专用网络以及客户机应该通过将 UDP 广播消息发送至它的 TCP/IP 子网中的所有计算机（如果客户机包含多块 LAN 适配卡，则将 UDP 广播消息发送至它可以访问的所有子网中的全部计算机）来尝试与运行 Communications Server for Linux 的服务器联系。

客户机每 10 秒钟重试广播一次，直达到 *broadcast_attempt_count* 参数指定的尝试次数为止，或者直到与服务器取得联系为止。如果在与服务器取得联系之前达到 *broadcast_attempt_count* 指定的限制，则客户机接着尝试使用发送至一台或多台指定的服务器的定向消息（这些服务器由文件中接下来的各行指定）。

UDP 广播的使用是可选的；要指定不应尝试广播，请不要包括此行。如果包括此行，则它必须位于文件中的任何服务器名之前。

在客户机与其服务器位于同一专用网络但却无法使用 UDP 广播访问任何服务器的情况下，请不要包括此行。这在下列情况下适用：

- 当 Communications Server for Linux LAN 跨多个 TCP/IP 子网，并且在客户机使用 UDP 可以访问的任何 TCP/IP 子网中都没有 Communications Server for Linux 服务器时
- 当客户机上未安装 UDP 支持时

如果客户机使用 HTTPS 来访问其服务器，则 UDP 广播不受支持。在此情况下，请显式地指定服务器名称，如下所述。

服务器名

指定客户机为了获取配置信息而应该联系的一个或多个 Communications Server for Linux 配置服务器的名称。此列表应包含与该客户机处于同一域中的主控配置服务器与所有备份服务器的名称。有关配置主控服务器与备份服务器的信息，请参阅第 47 页的『配置客户机 / 服务器功能』。

如果客户机使用 HTTPS 来访问其服务器，则 UDP 广播不受支持。在此情况下，请采用下列格式指定提供 HTTPS 支持的 WebSphere 服务器的名称以及 Communications Server for Linux 服务器的名称：

webservername : *servername1*

这假定将 WebSphere 设置为使用缺省端口 443 进行 HTTPS 连接。如果您的网络管理员已将 WebSphere 配置为使用其他端口号，则采用下列格式来包含该端口号：

```
webservname : portnumber : servname1
```

有关配置 WebSphere 以支持 HTTPS 连接的更多详细信息，请参阅《*Communications Server for Linux 快速入门*》。

如果未包括 * 行（指示使用 UDP 广播），或者客户机已尝试使用此方法来与服务器联系，但未接收到响应，则客户机尝试使用定向消息来与列示的第一台服务器联系。如果此操作失败，则客户机尝试与列示的第二台服务器联系，依此类推。这意味着您可以通过更改服务器的列示顺序来平衡两台或多台配置服务器之间的负载。

如果未包括 * 行（指示使用 UDP 广播），则必须指定至少一个服务器名；否则，服务器名是可选的。

如果客户机尝试了列示的所有服务器，但没有一次成功，则它等待由以上 *server_lost_timeout* 指定的时间，然后重新开始尝试与服务器联系的过程（使用 UDP 广播或使用列示的第一台服务器）。

除了 **sna_clnt.net** 之外，其他文件 **server.current** 也存储在同一个目录（对于 AIX，它是 **/etc/sna**；对于 Linux，它是 **/etc/opt/ibm/sna**）。如果有的话，这是包含客户机当前连接的服务器的名称的文本文件。您可以查看此文件来确定哪台服务器在充当客户机在域中的连接点。



定义客户机 TP

有关在 Remote API Client 系统上定义 TP 的信息，请参阅第 75 页的『定义 TP』或第 151 页的附录 B，『从命令行配置可调用 TP』。

附录 A. 配置计划工作表

本附录提供用于配置 Communications Server for Linux 的特定功能的工作表。这些工作表概述了启用每项功能所需的基本配置参数；有关高级配置参数的信息，请参阅本书正文中的相应章节，或参阅 *Communications Server for Linux Administration Command Reference*。

要收集配置节点所需的所有信息，必须完成下列类别的工作表：

节点配置

根据节点的能力以及节点工作所在的网络的特征，完成『节点工作表』中包含的其中一个工作表。

连接配置

根据用来与网络中的其他系统通信的链路协议的不同，完成第 133 页的『连接工作表』中包含的一个或多个工作表。

传递服务配置

完成第 142 页的『传递服务工作表』中的工作表，以使节点支持任何传递服务。

应用程序支持配置

根据节点要支持的用户应用程序的类型不同，完成第 145 页的『用户应用程序支持工作表』中包含的一个或多个工作表。

节点工作表

仅完成下列其中一个工作表：

- 『APPN 网络节点』
- 第 132 页的『APPN 端节点』
- 第 132 页的『APPN 分支网络节点』
- 第 133 页的『低入口联网节点』

APPN 网络节点

如果本地节点是 APPN 网络节点（在 APPN 网络中提供路由服务的节点），则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------|---|-----|
| “节点参数”对话框 | | |
| APPN 支持 | 网络节点 | |
| 控制点名称 | NETNAME.CPNAME（每个都是 1-8 个 A 型 EBCDIC 字符） | |
| | 要连接至 VTAM 主机，此名称必须与 VTAM PU 语句中的 NETID= 和 CPNAME= 条目匹配。 | |
| 控制点别名 | 限长 8 个字符 | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------------------------------|---|-----|
| 节点标识 | 8 个十六进制位 | |
| 连接配置: 请参阅第 133 页的『连接工作表』。 | | |
| 客户机 / 服务器配置: 对于单机节点, 此项不是必需的。 | | |
| 配置服务器? | 节点是否应该作为配置服务器, 以将有关域资源的信息存储在 Communications Server for Linux LAN 中? | |
| 应用程序配置: 请参阅第 145 页的『用户应用程序支持工作表』。 | | |

APPN 端节点

如果本地节点是 APPN 端节点 (可以使用动态路由信息但不为其他节点提供路由服务的节点), 则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------------------------------|---|-----|
| “节点参数”对话框 | | |
| APPN 支持 | 端节点 | |
| 控制点名称 | NETNAME.CPNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| | 要连接至 VTAM 主机, 此名称必须与 VTAM PU 语句中的 NETID= 和 CPNAME= 条目匹配。 | |
| 控制点别名 | 限长 8 个字符 | |
| 节点标识 | 8 个十六进制位 | |
| 连接配置: 请参阅第 133 页的『连接工作表』。 | | |
| 客户机 / 服务器配置: 对于单机节点, 此项不是必需的。 | | |
| 配置服务器? | 节点是否应该作为配置服务器, 以将有关域资源的信息存储在 Communications Server for Linux LAN 中? | |
| 应用程序配置: 请参阅第 145 页的『用户应用程序支持工作表』。 | | |

APPN 分支网络节点

如果本地节点是 APPN 分支网络节点 (此节点为从 APPN 主网络中分离出来的分支网络中的端节点提供网络节点功能, 而同时又在主网络中充当端节点), 则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------|--|-----|
| “节点参数”对话框 | | |
| APPN 支持 | 分支网络节点 | |
| 控制点名称 | NETNAME.CPNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| | 要连接至 VTAM 主机, 此名称必须与 VTAM PU 语句中的 NETID= 和 CPNAME= 条目匹配。 | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------------------------------|---|-----|
| 控制点别名 | 限长 8 个字符 | |
| 节点标识 | 8 个十六进制位 | |
| 连接配置: 请参阅『连接工作表』。 | | |
| 客户机 / 服务器配置: 对于单机节点, 此项不是必需的。 | | |
| 配置服务器? | 节点是否应该作为配置服务器, 以将有关域资源的信息存储在 Communications Server for Linux LAN 中? | |
| 应用程序配置: 请参阅第 145 页的『用户应用程序支持工作表』。 | | |

低入口联网节点

如果本地节点是低入口联网节点（不支持 APPN 功能的节点或只与主机通信的单机系统），则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------------------------------|---|-----|
| “节点参数”对话框 | | |
| APPN 支持 | 低入口联网节点 | |
| 控制点名称 | NETNAME.CPNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| | 要连接至 VTAM 主机, 此名称必须与 VTAM PU 语句中的 NETID= 和 CPNAME= 条目匹配。 | |
| 控制点别名 | 限长 8 个字符 | |
| 节点标识 | 8 个十六进制位 | |
| 连接配置: 请参阅『连接工作表』。 | | |
| 客户机 / 服务器配置: 对于单机节点, 此项不是必需的。 | | |
| 配置服务器? | 节点是否应该作为配置服务器, 以将有关域资源的信息存储在 Communications Server for Linux LAN 中? | |
| 应用程序配置: 请参阅第 145 页的『用户应用程序支持工作表』。 | | |

连接工作表

对于每个用来与另一个节点通信的链路协议, 完成下列其中一个工作表。必要时, 可以在一个端口上配置多个链路站。

- 第 134 页的『SDLC』
- 第 136 页的『令牌环』
- 第 137 页的『以太网』
- 第 139 页的『QLLC (X.25)』
- 第 140 页的『多路径通道』
- 第 141 页的『企业扩展程序 (HPR/IP)』

SDLC

完成此工作表以支持使用 SDLC 链路协议的连接。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---------------|---|-----|
| “SDLC 端口”对话框 | | |
| SNA 端口名 | 限长 8 个字符 | |
| SDLC 卡号 | 0 至 <i>number_of_cards_minus_1</i> | |
| 端口号 | 0 至 <i>number_of_ports_on_card_minus_1</i> | |
| 最初处于活动状态 | 根据需要选择 | |
| 线路详细信息 | | |
| 类型 | 租用线路 | |
| | 出局交换 | |
| | 入局交换 | |
| 链路角色 | 可协商 | |
| | 主 | |
| | 主多分支 | |
| | 辅助 | |
| | 辅助多 PU | |
| 对于入局交换或租用线路: | | |
| 轮询地址 | 仅适用于非主人局交换端口 | |
| | 在 VTAM 主机上, 轮询地址在 VTAM PU 定义中被配置为 <i>ADDR=</i> 参数。 | |
| | 在 AS/400 系统上, 轮询地址是线路描 述的 <i>STNADR</i> 参数。 | |
| “SDLC 链路站”对话框 | | |
| 链路站字段 | | |
| 名称 | 限长 8 个字符 | |
| SNA 端口名 | 限长 8 个字符 | |
| 激活 | 由管理员 | |
| | 在节点启动时 | |
| LU 数据传输 | 按要求 | |
| | 任何 | |
| | 仅独立 | |
| | 仅从属 | |
| 独立 LU 数据传输 | | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|------------|---|-----|
| 远程节点 | <i>NETNAME.CPNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符; 可选) | |
| 远程节点类型 | 如果远程系统是 VTAM 主机, 则可以在 VTAM 启动列表的 <i>NETID</i> 参数中找到网络名 (标准名称的前八个字符)。后八个字符位于 VTAM 启动列表的 <i>SSCPNAME</i> 参数中。 发现 网络节点 端节点或低入口联网节点 | |
| 从属 LU 数据传输 | | |
| 远程节点角色 | 主机 下游 (SNA 网关) 下游 (DLUR) | |
| 本地节点标识 | 8 个十六进制位 (缺省值为节点名) 在 VTAM 配置中, 前三位应该与 PU 定义中的 <i>IDBLK</i> 参数匹配, 后五位应该与 <i>IDNUM</i> 参数匹配。 在 AS/400 系统上, 节点标识是在 <i>EXCHID</i> 参数中配置的。 | |
| 远程节点标识 | 8 个十六进制位 (可选) | |
| 下游 PU 名 | 1-8 个 A 型 EBCDIC 字符 | |
| 上游 DLUS 名 | <i>NETNAME.LUNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 联系信息 | | |
| 轮询地址 | 对于入局交换端口, 仅对该端口配置。 2 个十六进制数字: <ul style="list-style-type: none"> • C1 (对于点到点) • 0xFF (对于主出局交换) (目标地址未知) • 主至多分支的特殊地址 在 VTAM 主机上, 轮询地址在 VTAM PU 定义中被配置为 <i>ADDR=</i> 参数。 在 AS/400 系统上, 轮询地址是线路描述的 <i>STNADR</i> 参数。 | |

令牌环

完成此工作表以支持使用令牌环链路协议的连接。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---------------|---|-----|
| “令牌环 SAP” 对话框 | | |
| SNA 端口名 | 限长 8 个字符 | |
| 令牌环卡号 | 0 至 <i>number_of_cards_minus_1</i> | |
| 本地 SAP 号 | 十六进制 (4 的倍数) | |
| 最初处于活动状态 | 根据需要选择 | |
| 定义连接网络 | 根据需要选择 | |
| CN 名 | NETNAME.CNNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| “令牌环链路站” 对话框 | | |
| 链路站字段 | | |
| 名称 | 限长 8 个字符 | |
| SNA 端口名 | 限长 8 个字符 | |
| 激活 | 由管理员 | |
| | 在节点启动时 | |
| | 按要求 | |
| LU 数据传输 | 任何 | |
| | 仅独立 | |
| | 仅从属 | |
| 独立 LU 数据传输 | | |
| 远程节点 | NETNAME.CPNAME (每个都是 1-8 个 A 型 EBCDIC 字符; 可选) | |
| | 如果远程系统是 VTAM 主机, 则可以在 VTAM 启动列表的 NETID 参数中找到网络名 (标准名称的前八个字符)。后八个字符位于 VTAM 启动列表的 SSCPNAME 参数中。 | |
| 远程节点类型 | 发现 | |
| | 端节点或低入口联网节点 | |
| | 网络节点 | |
| 从属 LU 数据传输 | | |
| 远程节点角色 | 主机 | |
| | 下游 (SNA 网关) | |
| | 下游 (DLUR) | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------|---|-----|
| 本地节点标识 | 8 个十六进制位 (缺省值为节点名) | |
| | 在 VTAM 配置中, 前三位应该与 PU 定义中的 <i>IDBLK</i> 参数匹配, 后五位应该与 <i>IDNUM</i> 参数匹配。 | |
| | 在 AS/400 系统上, 节点标识是在 <i>EXCHID</i> 参数中配置的。 | |
| 远程节点标识 | 8 个十六进制位 (可选) | |
| 下游 PU 名 | 1-8 个 A 型 EBCDIC 字符 | |
| 上游 DLUS 名 | <i>NETNAME.LUNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 联系信息 | | |
| MAC 地址 | 十六进制位 | |
| | 如果此链路的远程端是 VTAM 主机, 则可以在“VTAM 端口”定义的 <i>MACADDR=</i> 参数中找到远程端的 MAC 地址。 | |
| | 如果正在配置至 AS/400 系统的链路, 则 MAC 地址是线路描述中的 <i>ADPTADR</i> 参数。 | |
| SAP 号 | 十六进制 (4 的倍数) | |
| | 如果此链路的远程端是 VTAM 主机, 则 SAP 号是 VTAM PU 定义的 <i>SAPADDR=</i> 参数。 | |
| | 如果正在配置至 AS/400 系统的链路, 则 SAP 号是线路描述中的 <i>SSAP</i> 参数。 | |

以太网

完成此工作表以支持使用以太网链路协议的连接。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|--------------|--|-----|
| “以太网 SAP”对话框 | | |
| SNA 端口名 | 限长 8 个字符 | |
| 以太网卡号 | 0 至 <i>number_of_cards_minus_1</i> | |
| 本地 SAP 号 | 十六进制 (4 的倍数) | |
| 最初处于活动状态 | 根据需要选择 | |
| 定义连接网络 | 根据需要选择 | |
| CN 名 | <i>NETNAME.CNNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 以太网类型 | 选择“标准”或 802.3 | |
| “以太网链路站”对话框 | | |
| 链路站字段 | | |

连接工作表

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|------------|---|-----|
| 名称 | 限长 8 个字符 | |
| SNA 端口名 | 限长 8 个字符 | |
| 激活 | 由管理员 | |
| | 在节点启动时 | |
| LU 数据传输 | 按要求 | |
| | 任何 | |
| | 仅独立 | |
| | 仅从属 | |
| 独立 LU 数据传输 | | |
| 远程节点 | <i>NETNAME.CPNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符; 可选) | |
| | 如果远程系统是 VTAM 主机, 则可以在 VTAM 启动列表的 <i>NETID</i> 参数中找到网络名 (标准名称的前八个字符)。后八个字符位于 VTAM 启动列表的 <i>SSCPNAME</i> 参数中。 | |
| 远程节点类型 | 发现 | |
| | 网络节点 | |
| | 端节点或低入口联网节点 | |
| 从属 LU 数据传输 | | |
| 远程节点角色 | 主机 | |
| | 下游 (SNA 网关) | |
| | 下游 (DLUR) | |
| 本地节点标识 | 8 个十六进制位 (缺省值为节点名) | |
| | 在 VTAM 配置中, 前三位应该与 PU 定义中的 <i>IDBLK</i> 参数匹配, 后五位应该与 <i>IDNUM</i> 参数匹配。 | |
| | 在 AS/400 系统上, 节点标识是在 <i>EXCHID</i> 参数中配置的。 | |
| 远程节点标识 | 8 个十六进制位 (可选) | |
| 下游 PU 名 | 1-8 个 A 型 EBCDIC 字符 | |
| 上游 DLUS 名 | <i>NETNAME.LUNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 联系信息 | | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|----------|---|-----|
| MAC 地址 | 十六进制位 | |
| | 如果此链路的远程端是 VTAM 主机，则可以在“VTAM 端口”定义的 <i>MACADDR=</i> 参数中找到远程端的 MAC 地址。 | |
| | 如果正在配置至 AS/400 系统的链路，则 MAC 地址是线路描述中的 <i>ADPTADR</i> 参数。 | |
| SAP 号 | 十六进制 (4 的倍数) | |
| | 如果此链路的远程端是 VTAM 主机，则 SAP 号是 VTAM PU 定义的 <i>SAPADDR=</i> 参数。 | |
| | 如果正在配置至 AS/400 系统的链路，则 SAP 号是线路描述中的 <i>SSAP</i> 参数。 | |

QLLC (X.25)

完成此工作表以支持使用 QLLC (X.25) 链路协议的连接。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---------------|------------------------------------|-----|
| “QLLC 端口”对话框 | | |
| SNA 端口名 | 限长 8 个字符 | |
| X.25 卡号 | 0 至 <i>number_of_cards_minus_1</i> | |
| 最初处于活动状态 | 根据需要选择 | |
| “QLLC 链路站”对话框 | | |
| 链路站字段 | | |
| 名称 | 限长 8 个字符 | |
| SNA 端口名 | 限长 8 个字符 | |
| 激活 | 由管理员 | |
| | 在节点启动时 | |
| | 按要求 | |
| LU 数据传输 | 任何 | |
| | 仅独立 | |
| | 仅从属 | |
| 独立 LU 数据传输 | | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|------------|--|-----|
| 远程节点 | <i>NETNAME.CPNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符; 可选) | |
| 远程节点类型 | 如果远程系统是 VTAM 主机, 则可以在 VTAM 启动列表的 <i>NETID</i> 参数中找到网络名 (标准名称的前八个字符)。后八个字符位于 VTAM 启动列表的 <i>SSCPNAME</i> 参数中。 发现 网络节点 端节点或低入口联网节点 | |
| 从属 LU 数据传输 | | |
| 远程节点角色 | 主机 下游 (SNA 网关) | |
| 本地节点标识 | 下游 (DLUR) 8 个十六进制位 (缺省值为节点名) | |
| 远程节点标识 | 在 VTAM 配置中, 前三位应该与 PU 定义中的 <i>IDBLK</i> 参数匹配, 后五位应该与 <i>IDNUM</i> 参数匹配。 | |
| 下游 PU 名 | 在 AS/400 系统上, 节点标识是在 <i>EXCHID</i> 参数中配置的。 | |
| 上游 DLUS 名 | 8 个十六进制位 (可选) 1-8 个 A 型 EBCDIC 字符 <i>NETNAME.LUNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 联系信息 | | |
| 远程 X.25 地址 | 十六进制数字 (仅适用于 SVC); 1-4096 (仅适用于 PVC) | |

多路径通道

完成此工作表以支持使用“多路径通道”链路协议的连接。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---------------|--------------------|-----|
| “多路径通道端口”对话框 | | |
| SNA 端口名 | 限长 8 个字符 | |
| 端口号 | 必须与“多路径通道”设备的设备号匹配 | |
| 最初处于活动状态 | 根据需要选择 | |
| “多路径通道链路站”对话框 | | |
| 链路站字段 | | |
| 名称 | 限长 8 个字符 | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|------------|--|-----|
| SNA 端口名 | 限长 8 个字符 | |
| 激活 | 由管理员 在节点启动时 按要求 | |
| 独立 LU 数据传输 | | |
| 远程节点 | NETNAME.CPNAME (每个都是 1-8 个 A 型 EBCDIC 字符; 可选) 如果远程系统是 VTAM 主机, 则可以在 VTAM 启动列表的 NETID 参数中找到网络名 (标准名称的前八个字符)。后八个字符位于 VTAM 启动列表的 SSCPNAME 参数中。 | |
| 远程节点类型 | 发现 网络节点 端节点或低入口联网节点 | |

企业扩展程序 (HPR/IP)

完成此工作表以支持使用“企业扩展程序”链路协议的连接。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|----------------|---|-----|
| “企业扩展程序端口”对话框 | | |
| SNA 端口名 | 限长 8 个字符 | |
| 最初处于活动状态 | 根据需要选择 | |
| 本地 IP 接口 | 用于 IP 链路的本地网络适配卡的标识。 | |
| 定义连接网络 | 根据需要选择 | |
| CN 名 | NETNAME.CNNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| “企业扩展程序链路站”对话框 | | |
| 链路站字段 | | |
| 名称 | 限长 8 个字符 | |
| SNA 端口名 | 限长 8 个字符 | |
| 激活 | 由管理员 在节点启动时 按要求 | |
| 独立 LU 数据传输 | | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------|---|-----|
| 远程节点 | <i>NETNAME.CPNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符; 可选) | |
| | 如果远程系统是 VTAM 主机, 则可以在 VTAM 启动列表的 <i>NETID</i> 参数中找到网络名 (标准名称的前八个字符)。后八个字符位于 VTAM 启动列表的 <i>SSCPNAME</i> 参数中。 | |
| 远程节点类型 | 发现 | |
| | 端节点或低入口联网节点 | |
| | 网络节点 | |
| 联系信息 | | |
| 远程 IP 主机名 | 点分十进制 IP 地址 (如 193.1.11.100)、名称 (如 newbox.this.co.uk) 或别名 (如 newbox)。如果您指定名称或别名, 则 Linux 系统必须能够 (通过使用本地 TCP/IP 配置或使用域名服务器) 将它解析为标准名称。 | |

传递服务工作表

完成下列各节描述的任何传递服务的工作表 (如果本地节点将要支持该服务的话):

- 『本地节点上的 DLUR 』
- 第 143 页的 『下游节点的传递 DLUR 』
- 第 143 页的 『SNA 网关 』
- 第 144 页的 『TN 服务器 』
- 第 144 页的 『TN 重定向器 』

本地节点上的 DLUR

完成此工作表以在本地节点上支持 DLUR。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---|--|-----|
| 节点配置: 请参阅第 131 页的 『节点工作表 』。 | | |
| 连接配置: 请参阅第 133 页的 『连接工作表 』。要在本地节点上支持 DLUR, 请配置与 APPN 网络的连接。 | | |
| DLUR PU: | | |
| <i>PU</i> 名 | 1-8 个 A 型 EBCDIC 字符 | |
| <i>DLUS</i> 名 | <i>NETNAME.LUNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 备份 <i>DLUS</i> 名 | 此参数是可选的。 | |
| | <i>NETNAME.LUNAME</i> (每个都是 1-8 个 A 型 EBCDIC 字符) | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---|---|-----|
| PU 标识 | 8 个十六进制位 | |
| | 在 VTAM 配置中，前三位应该与 PU 定义中的 <i>IDBLK</i> 参数匹配，后五位应该与 <i>IDNUM</i> 设置匹配。 | |
| | 在 AS/400 系统上，PU 标识是在 <i>EXCHID</i> 参数中配置的。 | |
| 最初处于活动状态 | 根据需要选择 | |
| 支持压缩 | 根据需要选择 | |
| 无限地重试联系 <i>DLUS</i> | 根据需要选择 | |
| 本地 LU 和应用程序配置：请参阅第 145 页的『用户应用程序支持工作表』。必须配置本地从属 LU 以及所需的任何应用程序支持。 | | |

下游节点的传递 DLUR

如果本地节点是 APPN 网络节点，则可以为下游节点提供传递 DLUR 服务。完成此工作表以支持 DLUR。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|--|-----------|-----|
| 节点配置：将节点配置为网络节点（请参阅第 131 页的『APPN 网络节点』）。 | | |
| 连接配置：请参阅第 133 页的『连接工作表』。配置与 APPN 网络的连接，并为至下游节点的从属通信配置连接。 | | |

SNA 网关

如果本地节点要支持 SNA 网关，则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---|--|-----|
| 节点配置：请参阅第 131 页的『节点工作表』。 | | |
| 连接配置：请参阅第 133 页的『连接工作表』。为至主机的从属通信配置连接，并为至每个下游节点的从属通信配置链路。 | | |
| 本地 LU 和应用程序配置：请参阅第 145 页的『用户应用程序支持工作表』。 | | |
| “LU 池”对话框 | | |
| 池名 | 1-8 个 AE 型 EBCDIC 字符 | |
| LU 列表 | 指定给池的 LU (0-3 型) 的名称 | |
| “下游 LU”对话框 | | |
| 下游 LU 名 | 1-8 个 A 型 EBCDIC 字符（对于一系列 LU 的基本名称，则为 1-5 个字符） | |
| 下游 PU 名 | A 型 EBCDIC 字符串 | |
| LU 号 | 1-255（对于范围，请提供第一个和最后一个编号） | |
| 上游 LU 名 | A 型 EBCDIC 字符串（对于 LU 名） 或 AE 型 EBCDIC 字符串（对于 LU 池名） | |

TN 服务器

如果本地节点要支持 TN3270 客户机，则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|--|--|-----|
| 节点配置: 请参阅第 131 页的『节点工作表』。 | | |
| 连接配置: 请参阅第 133 页的『连接工作表』(为至主机的从属 LU 数据传输进行配置)。 | | |
| 本地 LU 和应用程序配置: 请参阅第 145 页的『用户应用程序支持工作表』。 | | |
| “LU 池”对话框 | | |
| 池名 | 1-8 个 AE 型 EBCDIC 字符 | |
| LU 列表 | 指定给池的 LU (0-3 型) 的名称 | |
| “TN 服务器访问”对话框 | | |
| TN3270 客户机地址 | 指定下列其中一个值: <ul style="list-style-type: none"> • 缺省记录 (任何 TN3270 客户机) • TCP/IP 地址 (客户机的点分十进制地址) • TCP/IP 名或别名 | |
| 支持 TN3270E | 选择此项以支持 TN3270E (同时还支持 TN3270 和 TN3287) | |
| TN3270 端口和 LU | | |
| TCP/IP 端口号 | 通常为 23。 | |
| 指定的显示器 LU | LU 或池名 | |
| 指定的打印机 LU | LU 或池名 | |
| 允许访问特定的 LU | 根据需要选择 | |
| SSL 安全会话 | 根据需要选择 | |
| 执行客户机认证 | 根据需要选择 | |
| 加密强度 | 指定下列其中一个值: <ul style="list-style-type: none"> • 仅认证 • 最小认证 • 最少 40 位 • 最少 56 位 • 最少 128 位 • 最少 168 位 | |
| “TN 服务器关联”对话框 | | |
| 显示器 LU | LU 名 | |
| 打印机 LU | LU 名 | |

TN 重定向器

如果本地节点要支持使用 TN 重定向器的 Telnet 客户机，则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---------------------------|-----------|-----|
| 节点配置: 请参阅第 131 页的『节点工作表』。 | | |
| “TN 重定向器访问”对话框 | | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---|---|-----|
| <i>Telnet</i> 客户机地址 | 指定下列其中一个值: <ul style="list-style-type: none"> • 缺省记录 (任何 Telnet 客户机) • TCP/IP 地址 (客户机的点分十进制地址) • TCP/IP 名或别名 | |
| TCP/IP 端口号 SSL 安全会话 执行客户机认证 加密强度 | 通常为 23。 根据需要选择 根据需要选择 指定下列其中一个值: <ul style="list-style-type: none"> • 仅认证 • 最小认证 • 最少 40 位 • 最少 56 位 • 最少 128 位 • 最少 168 位 | |
| 主机地址 | 指定下列其中一个值: <ul style="list-style-type: none"> • TCP/IP 地址 (主机的点分十进制地址) • TCP/IP 名或别名 | |
| TCP/IP 端口号 SSL 安全会话 加密强度 | 根据需要选择 指定下列其中一个值: <ul style="list-style-type: none"> • 仅认证 • 最小认证 • 最少 40 位 • 最少 56 位 • 最少 128 位 • 最少 168 位 | |

用户应用程序支持工作表

如果本地节点要支持相应的用户级别应用程序, 则完成下列工作表:

- 第 146 页的『APPC』
- 第 148 页的『CPI-C』
- 第 149 页的『5250』
- 第 149 页的『3270』
- 第 150 页的『LUA』

APPC

如果本地节点要支持 APPC 应用程序，则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---|--|-----|
| 节点配置: 请参阅第 131 页的『节点工作表』。 | | |
| 连接配置: 请参阅第 133 页的『连接工作表』。 | | |
| “本地 LU”对话框: 如果可以使用缺省控制点 LU, 则不需要进行配置。 | | |
| LU 名 | 1-8 个 A 型 EBCDIC 字符 | |
| LU 别名 | 限长 8 个字符 | |
| 从属 LU 参数 | | |
| 主机 LS/DLUR PU | 至主机或 DLUR PU 的从属链路站的名 称 (必须在定义 LU 之前定义) | |
| LU 号 | 1-255 | |
| | 此值必须与 VTAM/NCP LU 资源定义 语句中的 LOCADDR 参数匹配。 | |
| 缺省池的成员 | 根据需要选择 (仅对于从属 LU) | |
| 本地 LU 参数 | | |
| 支持同步点 | 根据需要选择 | |
| 禁用密码替换 | 根据需要选择 | |
| 限于特定的 SSCP | 根据需要选择 (仅对于从属 LU)。 | |
| | SSCP 标识是在 VTAM 启动列表的 SSCPID= 字段中定义的。 | |
| “远程节点”对话框: 仅当本地节点为低入口联网节点时才需要进行配置。 | | |
| 节点的 SNA 网络名 | NETNAME.CPNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| “伙伴 LU”对话框: 仅当与低入口联网节点通信、定义伙伴 LU 别名或本地节点是低入口联网节点时才需要进行配置。 | | |
| 伙伴 LU 名 | NETNAME.LUNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 别名 | 限长 8 个字符 | |
| 未解释名称 | 1-8 个 AE 型 EBCDIC 字符 (如果主 机 LU 名与本地使用的 PLU 名不同的 话) | |
| 支持并行会话 位置 | 如果支持则选择 NETNAME.CPNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| “LS 路由”对话框: 仅当伙伴 LU 由链路站找到时才需要进行配置。 | | |
| LU 名 | 1-8 个 A 型 EBCDIC 字符 | |
| LS 名 | 限长 8 个字符 | |
| 伙伴 LU 名 | NETNAME.LUNAME (每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 将伙伴 LU 名用作通 配符 | 根据需要选择 | |
| “方式”对话框: 仅当使用非标准方式时才需要进行配置。 | | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---|---|-----|
| 名称 | 1-8 个 A 型 EBCDIC 字符 | |
| COS 名 | 1-8 个 A 型 EBCDIC 字符 | |
| 会话限制 | | |
| 初始会话限制 | 最大为最大会话限制; 建议值为 8 | |
| 最大会话限制 | 最大为 32767 | |
| 最小争用胜方会话数 | 最大为最大会话限制; 建议值为 0。 | |
| 最小争用败方会话数 | 建议值为 0。 | |
| 自动激活会话数 | 0 至 <i>minimum_contention_winners</i> | |
| 接收调步窗口 | | |
| 初始窗口大小 | 建议值为 4 | |
| 最大窗口大小 | 可选 | |
| 会话超时 | | |
| 最大 RU 大小 | 建议上限为 1024。 | |
| 支持压缩 | | |
| 最大入站压缩 | 无 | |
| | RLE | |
| | LZ9 | |
| | LZ10 | |
| 最大出站压缩 | 无 | |
| | RLE | |
| | LZ9 | |
| | LZ10 | |
| “会话安全性”对话框: 仅当特定的本地 LU 与伙伴 LU 之间的会话需要会话安全性时才需要进行配置。 | | |
| 本地 LU | 1-8 个 A 型 EBCDIC 字符 | |
| 伙伴 LU | 1-8 个 A 型 EBCDIC 字符 | |
| 密码 | 16 位十六进制数 | |
| “TP 调用”对话框: 仅当要启动本地 TP 来响应远程系统的请求时才需要进行配置。 | | |
| TP 名 | 用户应用程序: 最多 64 个 ASCII 字符 | |
| 限于特定的 LU | 服务 TP: 最多 8 个十六进制位 | |
| LU 别名 | 根据需要选择 | |
| 支持多个实例 | 限长 8 个字符 | |
| | 为不排队 TP 选择此项; 如果未选择此项, 则将入局 Allocate 请求排队 (如果该 TP 已处于运行状态的话)。 | |
| 将入局 Allocate 路由至正在运行的 TP | 为广播排队 TP 选择此项 | |
| TP 可执行文件的全路径 | 可执行文件的路径和文件名 (缺省为 TP 名) | |
| 自变量 | 可执行文件的任何有效自变量 | |
| 用户标识 | 限长 64 个字符 | |
| 组标识 | 限长 64 个字符 | |

用户应用程序支持工作表

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|--|--|-----|
| “TP 定义”对话框: 定义 APPC 特征。 | | |
| TP 名 | 用户应用程序: 最多 64 个 ASCII 字符 | |
| 需要对话级安全性 | 服务 TP: 最多 8 个十六进制位 如果要求分配请求提供有效的用户名和密码, 则选择此项 | |
| 限制访问权 | 如果要求将用户名包括在安全性访问列表中, 则选择此项 | |
| 安全性访问列表 对话类型 | 安全性访问列表的名称 基本的 映射的 | |
| 同步级别 | 两者 无 确认 同步点 “无”或“确认” “无”、“确认”或“同步点” | |
| 允许 PIP | 根据需要选择 | |
| “对话安全性”对话框: 仅当响应远程系统的请求时要启动的本地 TP 需要对话安全性时才需要进行配置。 | | |
| 用户标识 | 限长 10 个字符 | |
| 密码 | 限长 10 个字符 | |

CPI-C

如果本地节点要支持 CPI-C 应用程序, 则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------------------------|---|-----|
| 节点配置: 请参阅第 131 页的『节点工作表』。 | | |
| 连接配置: 请参阅第 133 页的『连接工作表』。 | | |
| APPC 配置: 请参阅第 146 页的『APPC』。 | | |
| “CPI-C 目标”对话框 | | |
| 符号目标名称 | 1-8 个字符 | |
| 本地 LU | 别名 (限长 8 个字符) 或标准名称 (NETNAME.LUNAME, 每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 伙伴 LU | 别名 (限长 8 个字符) 或标准名称 (NETNAME.LUNAME, 每个都是 1-8 个 A 型 EBCDIC 字符) | |
| 方式 | A 型 EBCDIC 字符串 | |
| 伙伴 TP 名 | 用户应用程序: 限长 64 个字符 服务 TP: 最多 8 个十六进制位 | |

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|----------|---------------------------------------|-----|
| 安全性 | 无 相同 | |
| 用户标识 | 程序 仅用于安全级别“相同”或“程序” (与用户登录标识无关) | |
| 密码 | 仅用于安全级别“程序”(与用户登录 密码无关) | |

5250

如果本地节点要支持 5250 通信，则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|----------|--------------------------------|-----|
| 节点配置: | 请参阅第 131 页的『节点工作表』。 | |
| 连接配置: | 请参阅第 133 页的『连接工作表』(为独立通信进行配置)。 | |
| APPC 配置: | 请参阅第 146 页的『APPC』。 | |

3270

如果本地节点要支持 3270 通信，则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|---------------|--|-----|
| 节点配置: | 请参阅第 131 页的『节点工作表』。 | |
| 连接配置: | 请参阅第 133 页的『连接工作表』(为从属通信进行配置)。 | |
| “0-3 型 LU”对话框 | | |
| LU 名 | 1-8 个 A 型 EBCDIC 字符(或者对于 一系列 LU 的基本名称,则为 1-5 个字 符) | |
| 主机 LS/DLUR PU | 至主机或 DLUR PU 的从属链路站的名 称(必须在定义 LU 之前定义) | |
| LU 号 | 1-255(对于范围,请提供第一个和最 后一个编号) | |
| LU 类型 | 此值必须与 VTAM/NCP LU 资源定义 语句中的 LOCADDR 参数匹配。 3270 型号 2 (80x24) 显示器 3270 型号 3 (80x32) 显示器 3270 型号 4 (80x43) 显示器 3270 型号 5 (132x27) 显示器 3270 打印机 SCS 打印机 | |

用户应用程序支持工作表

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|-----------|-----------------------------|-----|
| 池中的 LU | 选择期望的选项（仅对于显示器 LU 和未限制 LU）。 | |
| 池名 | 1-8 个 AE 型 EBCDIC 字符 | |
| “LU 池”对话框 | | |
| 池名 | 1-8 个 AE 型 EBCDIC 字符 | |
| LU 列表 | 指定给池的 LU (0-3 型) 的名称 | |

LUA

如果本地节点要支持 LUA 应用程序，则完成此工作表。

| Motif 字段 | 有效条目 / 注释 | 实现值 |
|--------------------------------------|--|-----|
| 节点配置: 请参阅第 131 页的『节点工作表』。 | | |
| 连接配置: 请参阅第 133 页的『连接工作表』（为从属通信进行配置）。 | | |
| “0-3 型 LU”对话框 | | |
| LU 名 | 1-8 个 A 型 EBCDIC 字符（或者对于一系列 LU 的基本名称，则为 1-5 个字符） | |
| 主机 LS/DLUR PU | 至主机或 DLUR PU 的从属链路站点的名称（必须在定义 LU 之前定义） | |
| LU 号 | 1-255（对于范围，请提供第一个和最后一个编号） | |
| LU 类型 | 此值必须与 VTAM/NCP LU 资源定义语句中的 LOCADDR 参数匹配。 | |
| 池中的 LU | 未限制 | |
| 池名 | 选择期望的选项（仅对于显示器 LU 和未限制 LU） | |
| 池名 | 1-8 个 AE 型 EBCDIC 字符 | |
| “LU 池”对话框 | | |
| 池名 | 1-8 个 AE 型 EBCDIC 字符 | |
| LU 列表 | 指定给池的 LU (0-3 型) 的名称 | |

附录 B. 从命令行配置可调用 TP

Communications Server for Linux 包含一个命令行实用程序，该实用程序使用户或 TP 安装程序的编写者能够定义可调用 TP。可以在服务器或客户机上运行此实用程序。

根据您是在定义、除去还是查询 TP 定义，此命令的语法有所不同，而且对于 Windows 上的 Remote API Client，它是不同的：

定义可调用 TP:

AIX, LINUX

snatpinstall -a *file_name*

WINDOWS

tpinst32 -a *file_name*

注: **tpinst32** 命令适用于 32 位和 x64 版本的 Windows。



此命令从指定的 *file_name* 添加一个或多个 TP 定义。如果该文件中指定的 TP 已定义，则该文件中的信息替换现有定义。有关必需的文件格式的信息，请参阅第 152 页的『可调用 TP 定义的文件格式』。

除去可调用 TP 定义:

AIX, LINUX

snatpinstall -r -t *TP_name* [-l *LU_alias*]

此命令除去具有指定 TP 名和（如果使用同一 TP 名定义了多个 APPC TP 的话）指定 LU 别名的条目。如果该条目是用于 CPI-C 应用程序的，或者只存在一个用指定 TP 名定义的 APPC TP，则省略 **-l *LU_alias*** 选项。

WINDOWS

tpinst32 -r -t *TP_name*

此命令除去具有指定 TP 名的条目。



从命令行配置可调用 TP

查询可调用 TP 定义:

AIX, LINUX

snatpinstall -q [-t *TP_name*] [-l *LU_alias*]

此命令查询具有指定 TP 名和（如果使用同一 TP 名定义了多个 APPC TP 的话）指定 LU 别名的条目。如果该条目是用于 CPI-C 应用程序的，或者只存在一个用指定 TP 名定义的 APPC TP，则省略 **-l *LU_alias*** 选项。如果未包括 **-t *TP_name*** 选项，则此命令查询所有可调用 TP 定义。

WINDOWS

tpinst32 -q -t *TP_name*

此命令查询具有指定 TP 名的条目。如果未包括 **-t *TP_name*** 选项，则此命令查询所有可调用 TP 定义。

可调用 TP 定义的文件格式

提供可调用 TP 的配置信息的文件是一个 ASCII 文本文件，可以使用任何标准文本编辑器修改该文件。该文件中的每个条目都具有以下格式：

```
[TPname]  
PATH           = full_pathname_of_executable_file  
ARGUMENTS     = command-line_arguments_separated_by_spaces  
TYPE          = QUEUED | QUEUED-BROADCAST | NON-QUEUED  
TIMEOUT       = nnn
```

AIX, LINUX

```
USERID         = user_ID  
GROUP         = group_ID  
LUALIAS       = LU_alias  
ENV           = environment_variable=value  
              .  
              .  
              .  
ENV           = environment_variable=value
```

WINDOWS

```
SHOW           = MAXIMIZED|MINIMIZED|HIDDEN|NORMAL|NOACTIVATE|MINNOACTIVATE  
SECURITY_TYPE = APPLICATION | SERVICE  
SERVICE_NAME = name_of_installed_service  
USERID        = domain_name\user_ID
```

这些参数如下所示。对于操作员启动的 TP，使用的参数只有 TP 名、TP 类型、超时值和（对于 AIX 或 Linux 上的 APPC TP）LU 别名；其他参数仅适用于自动启动的 TP。

AIX, LINUX

在 AIX 或 Linux 上，当输入无效参数时，Communications Server for Linux 将返回错误消息。

WINDOWS

在 Windows 机器上，Communications Server for Linux 忽略无效的参数。

TPname

TP 的名称（1–64 个字符，不能有嵌入的空格字符）。在 Receive_Allocate 或入局 Allocate 请求中指定的 TP 名与此名称匹配。如果该 TP 是自动启动的 TP，则它启动时必须在 Receive_Allocate 中指定此 TP 名，以使 Communications Server for Linux 能够将入局 Attach 路由至正确的 TP。

必须用方括号将此名称括起来。可以用 ASCII 字符串指定该名称，并用双引号括起来（例如，["TPNAME1"]）。另外，还可以用表示该 TP 名的 EBCDIC 字符的十六进制数组（例如，[<53504E414D45F1>]）或这两种方式的组合（例如，[<3f>"TP1"]）指定该名称。在此示例中，第一个字符是不可打印字符 0x3f，后面跟着的字符是 "TP1"。

Communications Server for Linux 将提供的 ASCII 字符串转换为 EBCDIC，但不将十六进制字符串执行任何转换（已假定它是 EBCDIC 格式）。然后，在与指定的 TP 名进行匹配之前，在该名称的右边用 EBCDIC 空格填充该名称（直到总长为 64 个字符为止）。

PATH 此 TP 的可执行文件的路径和文件名。

此行是可选的。如果未包括此行，则 Communications Server for Linux 假定可执行文件名与 TP 名相同。如果指定不带路径的文件名，则 AIX 或 Linux 系统的缺省路径是 **/etc/opt/ibm/sna**；对于 Windows 客户机，Communications Server for Linux 使用正常的 Windows 机制来找到可执行文件。

ARGUMENTS

将被传递至 TP 的任何命令行自变量，由空格分隔。这些自变量按它们在命令行中出现的顺序被传递至 TP。

此行是可选的。如果未包括此行，则在不指定任何命令行自变量的情况下调用 TP。

TYPE 指定下列其中一个值：

QUEUED 该 TP 是排队 TP。当 TP 正在运行时到达的任何入局 Allocate 请求都会排队，直到该 TP 发出另一个 Receive_Allocate 或直到它运行完成

并可以重新启动为止。仅当配置成能将入局 Allocate 请求路由至此计算机的 LU 接收到入局 Allocate 请求时，才将该入局 Allocate 请求路由至此 TP。

QUEUED-BROADCAST

该 TP 是广播排队 TP。当 TP 正在运行时到达的任何入局 Allocate 请求都会排队，直到该 TP 发出另一个 Receive_Allocate 或直到它运行完成并可以重新启动为止。当 TP 启动时，将关于 TP 的信息广播至 LAN 中的所有服务器；如果另一台计算机上的 LU 接收到入局 Allocate 请求并且未配置路由信息，则它可以动态地找到 TP 并将 Allocate 请求路由至该 TP。

使用 QUEUED-BROADCAST 来代替 QUEUED 时，不必为 LU 配置显式的路由信息，并且通过在不同计算机上运行同一个 TP 的多个副本启用了负载均衡。但是，如果要避免广播信息以减少 LAN 数据传输，或者需要确保始终将到达特定 LU 的入局 Allocate 请求路由至 TP 的同一个副本，则应该使用 QUEUED。

NON-QUEUED

该 TP 是不排队 TP。每当 Communications Server for Linux 的入局 Allocate 请求到达时，Communications Server for Linux 就启动该 TP 的新副本。不要对不排队 TP 指定 *TIMEOUT* 参数。

定义为不排队类型的 TP 不能由操作员启动；它总是由 Communications Server for Linux 自动启动。如果该 TP 是要由操作员启动的，则不要指定 NON-QUEUED。如果用户尝试启动不排队 TP，则 Communications Server for Linux 拒绝 Receive_Allocate，这是因为没有入局 Allocate 请求正在等待它。

在不排队 TP 结束了一个对话之后，该 TP 可以终止，也可以发出另一个 Receive_Allocate。对于频繁使用的程序，这提供了一种方法来避免为每个对话启动新程序实例时的性能开销。每当自动启动的不排队 TP 接收到 Attach 时，Communications Server for Linux 检查此 TP 的实例中是否已经有未完成的 Receive_Allocate。如果已经有的话，则将此 TP 用于入局对话；否则 Communications Server for Linux 启动程序的新实例。

如果使用 NON-QUEUED，则可以同时运行 TP 的多个副本。如果 TP 写文件，则需要确保 TP 的不同副本不会覆盖彼此的文件。为此，使用下列其中一个方法：

- 确保 TP 将数据追加至现有文件而不是创建文件（以使 TP 的所有副本都将数据追加至同一个文件）
- 将 TP 设计成在运行时能够根据运行 TP 所用的进程标识来生成文件名（以使 TP 的每个副本写至不同的文件）。

此行是可选的。如果未包括此行，或者指定了无效的值，则缺省值为 QUEUED。

TIMEOUT

在没有入局 Allocate 请求处于暂挂状态时，由 TP 发出的 Receive_Allocate 调用应该阻塞的最大时间长度（以秒计）。如果在此时间内没有接收到入局 Allocate，则调用失败，并返回指示“状态检查 - Allocate 未暂挂”的返回码。

超时值 0 指示除非发出调用时入局 Allocate 已处于暂挂状态，否则调用总是失败。超时值 -1 指示调用无限期地等待入局 Allocate，并且不超时。

此行是可选的。如果未包括此行，或者指定了无效的值（非数字值），则缺省值为 -1（无限）。

如果 *TYPE* 参数设置为 NON-QUEUED，则不要指定此参数。Communications Server for Linux 将超时值 0 用于不排队 TP，这是因为，作为对入局 Allocate 的响应，总是启动 TP，因而总是有一个入局 Allocate 处于暂挂状态。

AIX, LINUX

USERID

指定 Communications Server for Linux 启动 TP 时所用的用户标识。TP 是在与此用户标识相关联的主目录中启动的。此主目录也是跟踪文件和由 TP 访问的任何其他文件的缺省路径（除非应用程序通过指定全路径来覆盖此缺省路径）。如果应用程序指定不带路径的文件名，则 Communications Server for Linux 在此主目录中搜索该文件；如果应用程序指定带有相对路径的文件名，则 Communications Server for Linux 在以此主目录为基准的指定目录中搜索该文件。

此行是必需的，必须指定此行。该标识必须是 Communications Server for Linux 计算机上的有效登录标识；除非 AIX 或 Linux 配置将用户名限定为更少字符，否则它最长可以是 64 个字符。

指定的用户对 *PATH* 参数指定的 TP 可执行文件必须具有执行许可权。另外，如果 *USERID* 设置为 root，则该文件必须由 root 用户拥有，并且必须具有 *setuid* 和 *setgid* 许可权，这样才能由 Communications Server for Linux 自动启动。

GROUP

指定 Communications Server for Linux 启动 TP 时所用的组标识。此标识必须是 Communications Server for Linux 计算机上的有效组标识；除非 AIX 或 Linux 配置将组名限定为更少字符，否则它最长可以是 64 个字符。

此行是可选的；如果未包括此行，则缺省值为 *other*。

LUALIAS

指定 TP 将从其接受入局 Attach 的本地 LU 别名。

注：仅当 TP 是 APPC TP 时才能使用此参数。如果 TP 是 CPI-C 应用程序，则不要指定此参数。CPI-C 不支持从特定本地 LU 接受入局 Attach；对 CPI-C 应用程序指定 LU 别名（即使是空白的 LU 别名）会导致在将入局 Attach 路由到 TP 时出错。

这是一个 8 字符的名称，它必须与 Communications Server for Linux 本地 APPC LU 的名称匹配。

要指示 TP 接受来自任何本地 LU 的入局 Attach，请将此参数设置为两个双引号字符 ""，这指示空白的 LU 别名。如果可调用 TP 数据文件包含同一个 TP 名的多个条目，则在这些条目中只有一个条目可以指定空白的 LU 别名；其他每个条目都必须指定不同的显式 LU 别名。Communications Server for Linux 将此 TP 名的入局 Attach 与指定了适当 LU 别名的 TP 匹配（如果可能的话），或者与指定了空白 LU 别名的 TP 匹配（如果找不到 LU 别名匹配项的话）。

可调用 TP 定义的文件格式

如果在文件中指定了非空 LU 别名，则 TP 必须使用 APPC RECEIVE_ALLOCATE 动词的扩展格式并指定此 LU 别名作为该动词的参数。这使 Communications Server for Linux 能够将入局 Attach 路由至正确的 TP。有关不同格式的 RECEIVE_ALLOCATE 的更多信息，请参阅 *Communications Server for Linux APPC Programmer's Guide*。如果需要允许 TP 在运行时确定正确的 LU 别名，而不是将该别名构建到应用程序中，则可以这样实现：设置一个包含适当 LU 别名的环境变量（使用 ENV 参数）并将应用程序设计成读取此环境变量来确定如何发出 RECEIVE_ALLOCATE。

此行是可选的；如果未包括此行，则缺省值为接受来自任何本地 LU 的入局 Attach，并且 TP 可以使用任何一种格式的 APPC RECEIVE_ALLOCATE 动词。

ENV 指定 TP 需要的任何环境变量。每个变量都是在单独的 ENV 行上以 *environment_variable=value* 格式指定的。最多可以包括 64 个 ENV 行；这些变量是按它们在此处出现的顺序设置的。

字符串 *environment_variable=value* 在 = 字符之前或之后一定不能包含空格或制表符字符。

WINDOWS

SHOW 仅当应用程序是 GUI 应用程序时，此参数才适用；如果该应用程序是控制台应用程序，则将忽略此参数。指定当应用程序启动时应该以何种方式显示。此参数被传递给应用程序，Communications Server for Linux 不对其进行处理；应用程序将负责对其进行解释以及对其执行操作。可以输入下列任何值：

MAXIMIZED

应用程序是最大化的。

MINIMIZED

应用程序是最小化的。

HIDDEN 应用程序不在屏幕上出现。

NORMAL 应用程序以正常的大小和位置显示。

NOACTIVATE

应用程序以正常的大小和位置显示，焦点仍位于先前的活动窗口中。此应用程序的窗口不会成为活动窗口。

MINNOACTIVATE

应用程序是最小化的，焦点仍位于先前的活动窗口中。

此参数是可选的。如果未包括此参数，则缺省值为 NORMAL。

SECURITY_TYPE

指定 TP 可执行文件的安全性类型：

APPLICATION

TP 可执行文件是使用 CreateProcess 系统调用作为应用程序启动的。

SERVICE

TP 可执行文件是使用 StartService 系统调用作为服务启动的。在此情况下，先前必须已使用由 SERVICE_NAME 参数指定的名称通过服务控制管理器安装了该服务。

此值将正在运行的 TP 作为 Windows 服务（而不是名称包含 4 个十六进制字符的 SNA 服务 TP）引用。Windows 只允许每次运行服务的一个副本，因此，不应该将 *TYPE* 参数设置为 NON-QUEUED；如果指定此值，则将使用 QUEUED-BROADCAST 值。

SERVICE_NAME

使用服务控制管理器安装的服务的名称。仅当 *SECURITY_TYPE* 是 SERVICE 时才使用此参数。

USERID

指定当 *SECURITY_TYPE* 是 APPLICATION 时客户机在启动 TP 时应该使用的域和用户标识。如果 Windows 客户机是域的一部分，则此参数的格式为 *domain_name\user_ID*；如果 Windows 客户机不是域的一部分，则此参数的格式为 *computer_name\user_ID*（指示 Windows 客户机自己的计算机名，而不是域名）。

客户机尝试在指定的用户的登录会话中启动 TP。如果 *USERID* 是空白的，或者未指定，则在控制台会话中启动 TP。如果指定的用户未登录，或者没有任何用户在控制台上登录，则不启动 TP，并且将向 Communications Server for Linux 服务器通报故障。

注意有关这些条目的格式的下列几点：

- 您可以包括注释行，方法是包括 # 字符作为该行的首字符；然后 Communications Server for Linux 就会忽略此行。Communications Server for Linux 还将完全忽略空白行。
- 每个 *parameter=value* 条目都必须在一行上；它不能包含换行字符。一行的最大长度为 255 个字符；其他的字符将被忽略。
- 位于一行开头或末尾处或者位于 = 字符之前或之后的空格（空格字符和制表符字符）将被忽略，ENV 参数的 *environment_variable=value* 字符串中的空格除外。
- 每个 TP 定义都以标识该 TP 名的行开始，以文件末尾或下一个 TP 名结束。
- 除 ENV 行以外（此行最多可以出现 64 次），不要对同一个 TP 多次指定同一参数。如果多次指定同一参数，则仅使用每个关键字的最后一个实例。

可调用 **TP** 定义的文件格式

附录 C. 声明

本信息是为在美国提供的产品和服务编写的。IBM 可能在其他国家或地区不提供本文中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

有关双字节（DBCS）信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区： International Business Machines Corporation “按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些国家或地区在某些交易中不允许免除明示或暗含的保证，因此本条款可能不适用于您。

本信息中可能有技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本出版物的新版本中。IBM 可以随时对本出版物中描述的产品和 / 或程序进行改进和 / 或更改，而不另行通知。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：（i）允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及（ii）允许对已经交换的信息进行相互使用，请与下列地址联系：

IBM Corporation
P.O. Box 12195
3039 Cornwallis Road

Research Triangle Park, NC 27709-2195
U.S.A.

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的。实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

本信息包含在日常业务操作中使用的数据和报告的示例。为了尽可能完整地说明这些示例，这些示例中可能会包括个人、公司、品牌和产品的名称。所有这些名称都是虚构的，与实际商业企业所用的名称和地址的任何雷同纯属巧合。

版权许可：本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口（API）进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。用户如果是为了按照 IBM 应用程序编程接口开发、使用、经销或分发应用程序，则可以任何形式复制、修改和分发这些样本程序，而无须向 IBM 付费。

凡这些样本程序的每份拷贝或其任何部分或任何衍生产品，都必须包括如下版权声明：®（贵公司名称）（年）。此部分代码是根据 IBM 公司的样本程序衍生出来的。®
Copyright IBM Corp. 2000, 2005, 2006. All rights reserved.

注册商标

下列各项是 IBM Corporation 在美国和 / 或其他国家或地区的商标:

| | |
|----------------------------------|----------------------------------|
| Advanced Peer-to-Peer Networking | Power5 |
| AIX | pSeries |
| Application System/400 | S/390 |
| AS/400 | SP |
| CICS | System/370 |
| IBM | System/390 |
| MQSeries | SAA |
| MVS | Systems Application Architecture |
| MVS/ESA | VTAM |
| MVS/XA | WebSphere |
| NetView | z/OS |
| OpenPower | z9 |
| OS/2 | zSeries |

下列各项是其他公司的商标或注册商标:

Java 和所有基于 Java 的商标是 Sun Microsystems Inc. 在美国和 / 或其他国家或地区的商标。

Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。

Intel、Intel 徽标、Intel Inside、Intel Inside 徽标、Intel Centrino、Intel Centrino 徽标、Celeron、Intel Xeon、Intel SpeedStep、Itanium 和 Pentium 是 Intel Corporation 或其子公司在美国和 / 或其他国家或地区的商标或注册商标。

UNIX 是 The Open Group 在美国和其他国家或地区的注册商标。

Linux 是 Linus Torvalds 在美国和 / 或其他国家或地区的商标。

其他公司、产品和服务名称可能是其他公司的商标或服务标记。

书目提要

下列 IBM 出版物提供关于此库中讨论的主题的信息。这些出版物分为以下主要的主题区：

- Communications Server for Linux V6.2.2
- 系统网络体系结构 (SNA)
- 主机配置
- z/OS® Communications Server
- 传输控制协议 / 英特网协议 (TCP/IP)
- X.25
- 高级程序间通信 (APPC)
- 编程
- 其他 IBM Networking 主题

对于 Communications Server for Linux 资料库中的书籍，提供了简短描述。对于其他书籍，此处只显示书名和出版编号，某些情况下还显示在本书的文本中使用的缩写书名。

Communications Server for Linux V6.2.2 出版物

Communications Server for Linux 资料库包含以下书籍。另外，CD-ROM 上提供了这些文档的软拷贝版本。有关访问 CD-ROM 上的软拷贝文件的信息，请参阅 *IBM Communications Server for Linux Quick Beginnings*。要在系统上安装这些软拷贝书籍，需要 9 至 15 MB 的硬盘空间（取决于您安装的本地语言版本）。

- 《*IBM Communications Server for Linux 快速入门*》(G152-0885-01)

此书对 Communications Server for Linux 进行了一般介绍，包括有关受支持的网络特征、安装、配置和操作的信息。

- 《*IBM Communications Server for Linux 管理指南*》(S152-0886-01)

此书提供有关 Communications Server for Linux 配置和操作的 SNA 和 Communications Server for Linux 概述和信息。

- *IBM Communications Server for Linux Administration Command Reference* (SC31-6770-01)

此书提供有关 SNA 和 Communications Server for Linux 命令的信息。

- *IBM Communications Server for Linux CPI-C Programmer's Guide* (SC31-6774-01)

此书向有经验的“C”或 Java™ 程序员提供有关使用 Communications Server for Linux 用于通信的 CPI API 来编写 SNA 事务程序的信息。

- *IBM Communications Server for Linux APPC Programmer's Guide* (SC31-6773-01)

此书包含使用“高级程序间通信”（APPC）编写应用程序所需要的信息。

- *IBM Communications Server for Linux LUA Programmer's Guide* (SC31-6776-01)

此书包含使用“常规 LU 应用程序编程接口”（LUA）编写应用程序所需要的信息。

- *IBM Communications Server for Linux CSV Programmer's Guide* (SC31-6775-01)

此书包含使用“公共服务动词”（CSV）应用程序接口（API）编写应用程序所需要的信息。

- *IBM Communications Server for Linux MS Programmer's Guide* (SC31-6777-01)

此书包含使用“管理服务”（MS）API 编写应用程序所需要的信息。

- *IBM Communications Server for Linux NOF Programmer's Guide* (SC31-6778-01)

此书包含使用“节点操作员设施”（NOF）API 编写应用程序所需要的信息。

- *IBM Communications Server for Linux Diagnostics Guide* (SC31-6779-01)

此书提供关于 SNA 网络问题的信息。

- *IBM Communications Server for Linux APPC Application Suite User's Guide* (SC31-6772-01)

此书提供有关与 Communications Server for Linux 配合使用的 APPC 应用程序的信息。

- *《IBM Communications Server for Linux 词汇表》* (G152-0888-01)

本书提供在整个 IBM Communications Server for Linux 库中使用的术语和定义的综合列表。

系统网络体系结构（SNA）出版物

下列书包含关于 SNA 网络的信息：

- *Systems Network Architecture: Format and Protocol Reference Manual – Architecture Logic for LU Type 6.2* (SC30-3269)
- *Systems Network Architecture: Formats* (GA27-3136)
- *Systems Network Architecture: Guide to SNA Publications* (GC30-3438)
- *Systems Network Architecture: Network Product Formats* (LY43-0081)
- *Systems Network Architecture: Technical Overview* (GC30-3073)
- *Systems Network Architecture: APPN Architecture Reference* (SC30-3422)
- *Systems Network Architecture: Sessions between Logical Units* (GC20-1868)
- *Systems Network Architecture: LU 6.2 Reference – Peer Protocols* (SC31-6808)
- *Systems Network Architecture: Transaction Programmer's Reference Manual for LU Type 6.2* (GC30-3084)
- *Systems Network Architecture: 3270 Datastream Programmer's Reference* (GA23-0059)
- *Networking Blueprint Executive Overview* (GC31-7057)
- *Systems Network Architecture: Management Services Reference* (SC30-3346)

主机配置出版物

下列书包含关于主机配置的信息:

- *ES/9000, ES/3090 IOCP User's Guide Volume A04* (GC38-0097)
- *3174 Establishment Controller Installation Guide* (GG24-3061)
- *3270 Information Display System 3174 Establishment Controller: Planning Guide* (GA27-3918)
- *OS/390 Hardware Configuration Definition (HCD) User's Guide* (SC28-1848)

z/OS Communications Server 出版物

以下书籍包含有关 z/OS Communications Server 的信息:

- *z/OS V1R7 Communications Server: SNA Network Implementation Guide* (SC31-8777)
- *z/OS V1R7 Communications Server: SNA Diagnostics* (Vol 1: GC31-6850, Vol 2: GC31-6851)
- *z/OS V1R6 Communications Server: Resource Definition Reference* (SC31-8778)

TCP/IP 出版物

以下书籍包含有关传输控制协议 / 英特网协议 (TCP/IP) 网络协议的信息:

- *z/OS V1R7 Communications Server: IP Configuration Guide* (SC31-8775)
- *z/OS V1R7 Communications Server: IP Configuration Reference* (SC31-8776)
- *z/VM V5R1 TCP/IP Planning and Customization* (SC24-6125)

X.25 出版物

以下书籍包含有关 X.25 网络协议的信息:

- *Communications Server for OS/2 Version 4 X.25 Programming* (SC31-8150)

APPC 出版物

下列书籍包含关于“高级程序间通信”(APPC)的信息:

- *APPC Application Suite V1 User's Guide* (SC31-6532)
- *APPC Application Suite V1 Administration* (SC31-6533)
- *APPC Application Suite V1 Programming* (SC31-6534)
- *APPC Application Suite V1 Online Product Library* (SK2T-2680)
- *APPC Application Suite Licensed Program Specifications* (GC31-6535)
- *z/OS V1R2.0 Communications Server: APPC Application Suite User's Guide* (SC31-8809)

编程出版物

下列书包含关于编程的信息:

- *Common Programming Interface Communications CPI-C Reference* (SC26-4399)

- *Communications Server for OS/2 Version 4 Application Programming Guide* (SC31-8152)

其他 IBM Networking 出版物

以下书籍包含有关与 Communications Server for Linux 相关的其他主题的信息:

- *SDLC Concepts* (GA27-3093)
- *Local Area Network Concepts and Products: LAN Architecture* (SG24-4753)
- *Local Area Network Concepts and Products: LAN Adapters, Hubs and ATM* (SG24-4754)
- *Local Area Network Concepts and Products: Routers and Gateways* (SG24-4755)
- *Local Area Network Concepts and Products: LAN Operating Systems and Management* (SG24-4756)
- *IBM Network Control Program Resource Definition Guide* (SC30-3349)

索引

[A]

- 安全套接字层 (SSL) 93, 96
 - 服务器认证 94, 96, 97
 - 客户机认证 93, 96
 - 数据加密 94, 96
- 安全性
 - 对话 87
 - 会话 86
 - APPC 86
 - UCF 105, 107
- 安全性参数 85
- 安全性访问列表
 - 参数 87
 - 目的 87
 - 配置方法 87
 - 其他配置需求 88
- 安全性访问列表参数 80

[B]

- 版本号, NetView 101
- 帮助
 - 命令行管理程序 46
 - Motif 管理程序 44
- 备份服务器 109
- 备份主控服务器 47
- 备份 DLUS 名参数 62
- 本地节点
 - LU 7
- 本地节点标识参数 60
- 本地 IP 接口参数 55
- 本地 LU
 - 参数 70, 71
 - 定义 69
 - 描述 7
 - 配置方法 70
 - 其他配置需求 71
- 本地 LU 别名参数 84
- 本地 LU 参数 84, 86
- 本地 LU 名参数 74
- 本地 SAP 号参数 54
- 边界节点 2
- 标准错误参数 79
- 标准输出参数 79
- 标准输入参数 79
- 标准 CP 名称 14
- 标准 LU 名称 14
- 别名参数 73
- 别名, 伙伴 LU 72
- 并行会话 8

[C]

- 参数用于在任何 LU 参数上的调用 77
- 池名参数 66
- 池中的 LU 参数 66
- 初始窗口大小参数 83
- 初始会话限制参数 82
- 传递服务
 - 工作表 131, 142
 - 配置 91
- 传递 DLUR 63
- 传输网络 11
- 传输组 17
- 窗口
 - 菜单 37
 - 工具栏按钮 41
 - 节点 37, 39
 - 描述 37
 - 域 36, 37
 - 资源 37
 - 资源项 40
 - CPI-C 目标名称 37
 - LU 池 37
- 从属节点 2
- 从属于 SSCP 的 LU 8
- 从属 LU
 - 描述 8
 - 配置 65
- 从属 LU 服务器 (DLUS) 24
- 错误日志文件 31

[D]

- 打印机 3
- 打印机 LU 参数 95
- 低入口联网节点
 - 功能 12
 - 工作表 133
 - 描述 4, 13
 - 目录 14, 15
- 低入口联网 (LEN) 节点 4
- 调用 TP 9, 75
- 定向搜索 17
- 定义连接网络参数 54, 55
- 独立 LU
 - 描述 8
 - 配置 69
- 端节点
 - 描述 4
 - 目录 15, 16
 - 在样本 APPN 网络中 12

- 端节点 (续)
 - APPN 13
- 端口
 - 参数 53, 54, 55
 - 配置 52, 53
 - 其他配置需求 55
- 端口号参数 53
- 对等通信 1
- 对话
 - 安全性 87
 - 描述 9
- 对话安全性
 - 参数 87
 - 配置方法 87
- 对话框 37
- 对话类型参数 80
- 多个会话 8
- 多路径通道
 - 工作表 140
- 多路径通道 (MPC)
 - 端口配置 52

[F]

- 方式 80
 - 标准 81
 - 参数 82, 83, 84
 - 描述 10
 - 配置 81, 82
 - 其他配置需求 84
- 方式参数 85
- 访问表中的用户参数 87
- 分支扩展程序 23
- 分支链路类型参数 59
- 分支网络节点 4, 23
- 服务等级 (COS) 11
- 服务点 101
- 服务点命令设施 (SPCF) 30, 101
- 服务访问点 (SAP) 23
- 服务器
 - 除去 47
 - 禁用 35
 - 启用 34
 - 添加 47
 - 与客户机的关系 109
- 辅助 LU 8
- 复位为 SNA 定义值参数 84

[G]

- 高级对等联网 (APPN) 1
- 高性能路由 (HPR) 11, 17
- 格式 1
- 跟踪
 - 客户机 / 服务器 126

- 跟踪 (续)
 - LAN 126
- 跟踪内核组件 35
- 跟踪文件 32
- 共享访问传输设施 (SATF) 21
- 工作表 33
- 故障诊断 31
- 管理
 - 工具 28
 - 责任 27
- 管理服务 (MS) 13
- 管理服务 (MS) API 5
- 广播搜索 17
- 规划工作表 33

[H]

- 环境参数 79
- 会话
 - 类型 7
 - 路由 17
 - 描述 7
- 会话安全性
 - 参数 86
 - 配置方法 86
 - 其他配置需求 86
- 会话超时参数 83
- 混合网络 2, 24
- 伙伴 LU 7
 - 别名, 定义 72
 - 参数 73, 74
 - 多个, 用通配符定义 73
 - 配置方法 72
 - 其他配置需求 75
 - 远程节点, 定义 73
- 伙伴 LU 参数 85, 86
- 伙伴 LU 名参数 73, 74
- 伙伴 TP 参数 85

[J]

- 基本对话 10
- 激活参数 57
- 记录 50
 - Windows 上的 Remote API Client 119
- 简明记录 32
- 将入局 Allocate 路由至正在运行的 TP 参数 78
- 节点
 - 参数 48, 49
 - 工作表 131
 - 类型 2, 4
 - 目的 48
 - 配置方法 48
 - 配置文件 30
 - 其他配置需求 49

- 节点 (续)
 - 同级 2
 - 外围 2
 - 子区域 2
 - SNA 2
- 节点标识参数 49
- 节点操作员设施 (NOF) API 5
- 节点窗口 39
- 节点的 SNA 网络名参数 72
- 节点资源 47
- 接收调步窗口参数 83
- 介质访问控制 (MAC) 23
- 禁用软件 36
- 禁用 Communications Server for Linux 软件
 - Windows 上的 Remote API Client 114
- 局部拓扑数据库 17

[K]

- 可调用 TP 9
 - 定义到 Communications Server for Linux 75
 - 使用 snatpinstall 151
 - 数据文件 30
- 客户机
 - 定义 TP, 于 129
 - 管理 109
 - 可调用 TP 配置 151
 - 联网需求 111
 - 网络数据文件 31
 - ARGUMENTS 参数 153
 - PATH 参数 153
 - SECURITY_TYPE 参数 156
 - SERVICE_NAME 参数 157
 - SHOW 参数 156
 - TIMEOUT 参数 154
 - TPname 参数 153
 - TYPE 参数 153
- 客户机 / 服务器
 - 跟踪 123, 126
 - 配置 47
- 客户信息控制系统 (CICS) 6
- 控制点别名参数 49
- 控制点名称参数 48
- 控制点 (CP) 7
- 控制数据 7
- 快速传输协议 (RTP) 11, 17

[L]

- 连接
 - 工作表 131, 133
 - 描述 4
 - 配置 51
 - 直接 21

- 连接网络
 - 描述 11
 - 配置 52, 53
 - 配置方法 52
 - 其他配置需求 55
 - 拓扑信息 18
 - APPN 22
- 链路站
 - 参数 57, 58, 59, 60, 61
 - 描述 4
 - 配置 55, 56
 - 其他配置需求 61
- 链路站路由
 - 参数 74
- 链路站名参数 74
- 令牌环
 - 端口配置 52
 - 工作表 136
- 路由 10
- 路由选择 10, 17, 20
- 轮询地址参数 57
- 逻辑单元 (LU) 5
- 逻辑记录 10

[M]

- 密码参数 85, 86, 87
- 名称参数
 - 安全性访问列表 87
 - 方式 82
- 链路站 57
- CPI-C 符号目标 84
- LU 池 67
- 命令
 - 修改配置服务器 47
- 命令行管理程序
 - 帮助 46
 - 从客户机 46
 - 描述 29
 - 命令类型 46
 - 使用 45
- 目标主机地址参数 97
- 目标 TP 9, 75
- 目录
 - 低入口联网节点 14, 15
 - 端节点 15, 16
 - 网络节点 15, 16
 - 用于 Communications Server for Linux 可执行程序 34

[N]

- 内部跟踪
 - Windows 上的 Remote API Client 124
- 内核内存限制 34
- 内核组件, 跟踪 35

[P]

配置

- 安全性访问列表 87
- 传递服务 91
- 从属 LU 65
- 端口 52
- 节点 48
- 连接 51
- 连接网络 52
- 任务 47
- 文件 30
- APPC 安全性 86
- APPC 通信 69
- CPI-C 辅助信息 84
- DLC 52
- DLUR 99
- SNA 网关 98
- TN 重定向器访问记录 95
- TN 服务器访问记录 92
- TN 服务器 关联记录 94
- TP 75
- 配置服务器 47
 - 除去 47
 - 添加 47

[Q]

- 启动命令 34
- 企业扩展程序 (HPR/IP)
 - 端口配置 52
 - 工作表 141
- 启用 Communications Server for Linux 软件
 - 初始化期间的问题 35
 - 在服务器上 34
 - Windows 上的 Remote API Client 114
- 启用 SNA 软件
 - AIX 或 Linux 上的 Remote API Client 126
- 前端处理机 (FEP) 3
- 请求单元 (RU) 83
- 缺省池的成员参数 71
- 群集控制器 3

[R]

- 任务表 33
- 日志文件
 - 类型 49
 - 配置 49
- 日志消息 31

[S]

- 上游 DLUS 名参数 61
- 上游 LU 名参数 99

- 审计日志文件 31
- 使用缺省 LU 参数 84
- 适配卡号参数 53
- 事务程序 (TP) 4
- 数据链路控制 (DLC) 51
- 数据文件
 - 节点配置 30
 - 可调用 TP 30
 - 客户机网络 31
 - 域配置 30
 - SNA 网络 31
 - TP 定义 30

[T]

- 停止命令 36
- 通配符 73
- 通配符伙伴 LU 名参数 73
- 通信控制器 3
- 通信控制器节点 2
- 通信链路 3
- 通用数据流 (GDS) 6
- 同步级别参数 80
- 同级网络 2
 - 节点类型 4
 - 路由选择 11
- 拓扑和路由服务 (TRS) 17
- 拓扑数据库更新 (TDU) 19
- 拓扑信息 8
 - 本地 18
 - 连接网络 18

[W]

- 外围节点 2
- 完全记录 32
- 网络
 - 管理 101
 - 混合 24
 - 类型 2
 - 拓扑数据库 17
- 网络访问过程 (NAP) 113
- 网络管理数据 7
- 网络节点
 - 目录 15, 16
 - 样本配置 12
- 网络节点服务器 4, 12
- 网络可访问单元 (NAU) 5
- 网络可寻址单元 5
- 网络数据文件
 - 描述 31
 - AIX 或 Linux 上的 Remote API Client 126
- 为隐式 PU 访问配置下游 LU 参数 55
- 未解释名称参数 73
- 位置参数 74

文档内容体系结构 (DCA) 6
问题确定帮助
 概述 31
 记录 49
无限地重试联系 DLUS 参数 63
物理部件控制点 (PUCP) 7
物理部件 (PU) 5

[X]

系统服务控制点 (SSCP) 7
下游 LU 名参数 99
下游 PU 名参数 60, 63
显示器 LU 参数 95
线路详细信息参数 53
限制访问权参数 80
限制最大 RU 大小参数 83
相邻节点 11
协议 1
信息管理系统 / 虚拟存储 (IMS/VS) 6
虚路由节点 (VRN) 22
虚拟终端访问方法 (VTAM) 11
需要对话级安全性参数 80

[Y]

延迟登录参数 99
以太网
 端口配置 52
 工作表 137
以太网类型参数 54
隐式链路上的链路级别错误恢复参数 55
应用程序
 程序 4
 工作表 131
应用程序编程接口 (API) 5
映射对话 10
用户标识参数 79, 85, 87
用户应用程序支持工作表 145
域
 描述 2
 配置文件 30
域窗口 37
域名
 更改 110
域资源 47
源 TP 9, 75
远程节点
 定义 71
 伙伴 LU 73
 节点的 SNA 网络名参数 72
 配置方法 71
 其他配置需求 72
 LU 7
远程节点标识参数 60

远程节点角色参数 60
远程节点类型参数 59
远程节点名参数 59
远程命令设施 (RCF) 30
远程作业输入 (RJE) 6
远程 X.25 地址参数 58
允许超时参数 99
允许访问特定 LU 参数 93
允许 PIP 参数 80

[Z]

找到资源 14
诊断工具 31
支持并行会话参数 74
支持多个实例参数 78
支持压缩参数 62, 83
支持 TN3270E 参数 92
指定超时参数 83
指定的打印机 LU 参数 93
指定的显示器 LU 参数 93
指定的 LU 参数 67
直接连接 21
终端 3
终端控制器 3
中间会话路由 (ISR) 17, 20
中间路由 20
中央记录 31
主机 3
主机节点 2
主机 LS/DLUR PU 参数 65, 70
主控服务器 47, 109
 指定 110
主 LU 8
转义字符, RCF 103
状态
 Windows 上的 Remote API Client 114
资源名称 14
资源, 找到 14
子区域节点 2
子区域网络
 节点类型 2
 路由选择 10
 描述 2
 示例 3
子区域 SNA 1
自变量参数 78
自动激活会话参数 83
自动网络路由 (ANR) 11, 17
字符, 在 RCF 命令中 103
组标识参数 79
最初处于活动状态参数 53, 62
最大窗口大小参数 83
最大会话限制参数 82
最大活动模板实例数参数 55
最大 RU 大小参数 83

最小争用败方会话数参数 82
最小争用胜方会话数参数 82

[数字]

0-3 型 LU
 参数 65, 66
 配置方法 65
 其他配置需求 66
2.0 型节点 2
2.1 型节点 2
3270
 池配置 67
 工作表 149
 LU 配置 65
4 型节点 2
5 型节点 2
5250
 工作表 149

A

AIX 或 Linux 上的 Remote API Client
 服务器名 128
 管理 126
 broadcast_attempt_count 128
 invoked_tps 127
 lan_access_timeout 127
 server_lost_timeout 128
 * 128
ANR
 动态路由 20
 描述 11, 17
API
 描述 5
 随 Communications Server for Linux 提供 5
 专用 5
API 跟踪
 Windows 上的 Remote API Client 121
APPC
 安全性 86
 工作表 146
 配置 69
APPCLLU
 Windows 上的 Remote API Client 125
APPCTPN
 Windows 上的 Remote API Client 125
Application System/400 (AS/400) 11
APPN
 端节点 4, 13, 132
 分支网络节点 4, 132
 函数 11
 节点类型 12
 控制点 13
 连接网络 22

APPN (续)
 路由选择 20
 描述 1, 11
 网络 11, 21
 网络节点 4, 12, 131
 网络示例 12
APPN 支持参数 48
AS/400 (Application System/400) 11

B

BIND 请求 8

C

CICS (客户信息控制系统) 6
CN (连接网络) 11
CN 名参数 54, 55
Communications Server for Linux 可执行程序的路径 34
COS
 类型 81
 描述 11
 目的 80
COS 名参数 82
CP (控制点) 7
CPI-C (用于通信的公共编程接口)
 辅助信息 84
 工作表 148
CPI-C 辅助信息
 参数 84, 85
 配置方法 84
 其他配置需求 86
CP-CP 会话 8
CSVTBLG
 Windows 上的 Remote API Client 125

D

DCA (文档内容体系结构) 6
DLC
 配置 51, 53
 配置方法 52
 其他配置需求 55
DLUR
 工作表 142
 描述 25
 配置 99
 其他配置需求 63
DLUR 下游节点 63
DLUR PU
 参数 62, 63
 配置方法 62
DLUS
 描述 24
DLUS 名参数 62, 63

E

EN (端节点) 4
ENV 参数 156

F

FEP (前端处理机) 3

G

GDS (通用数据流) 6
GROUP 参数 155

H

HPR
描述 11, 17
HPR 在隐式链路上受支持参数 55
HTTPS 112

I

IMS/VS (信息管理系统 / 虚拟存储) 6
IP 端口号 111
ISR 17, 20

L

LAN 访问超时 112
LAN 跟踪
在客户机上 126
Linux 客户机
域名 127
Linux 命令 101
LS (链路站) 55
LU
类型 6
描述 5
LU 别名参数 70, 77
LU 池
参数 67
查看 67
定义 67
配置方法 67
LU 号参数 66, 70, 99
LU 类型参数 66
LU 名参数 65, 70
LU 数据传输参数 57
LU 0
描述 6
LU 1 6
LU 2 6
LU 3 6

LU 6.2
描述 6
配置 69
LUA
池配置 67
工作表 150
配置 65
LUALIAS 参数 155
LU-LU 会话 7

M

MAC (介质访问控制) 23
MAC 地址参数 58
Motif 管理程序
帮助 44
调用 36
对话框 42, 44
工具栏按钮 41
节点窗口 39
描述 28
使用 36
域窗口 37
资源窗口 37
资源项 40
MPC
端口配置 52
MPC 组参数 58
MS (管理服务) 13

N

NAP (网络访问过程) 113
NAU (网络可访问单元) 5
NetView
版本号 101
程序 101
服务点 101
更改命令输入区的大小 102
描述 101
命令 101
屏幕显示 102
NN (网络节点) 4
NOF (节点操作员设施) API 31

P

PU
描述 5
DLUR 61
PU 标识参数 62
PU 名参数 62
PUCP (物理部件控制点) 7

Q

QLLC

- 端口配置 52
- 工作表 139

R

RCF

- 命令语法 102
- 设施 30
- 有效字符 103

RJE (远程作业输入) 6

RTP

- 端点 21
- 描述 11, 17

RU (请求单元) 83

S

SAP (服务访问点) 23

SAP 号参数 58

SATF

- 在 APPN 网络中 22
- 直接连接 21

SDLC

- 端口配置 52
- 工作表 134

SEND 函数 10

SNA

- 层 2
- 层次结构 2
- 基本概念 1
- 描述 1
- 网络 1
- 网络类型 2
- 网络数据文件 31, 126
- 子区域 1
- APPN 概念 11

SNA 端口名参数 53, 57

SNA 网关

- 工作表 143
- 目的 98

SNA 网关的下游 LU

- 参数 99
- 配置方法 98
- 其他配置需求 99

SNA 网络信息

- Windows 上的 Remote API Client 115

snaadmin 程序 29

snanetutil 程序 110

sna_clnt.net 文件 126

SPCF

- 描述 30, 101
- 命令 103
- 命令语法 102

SSCP (系统服务控制点) 7

SSCP-LU 会话 7

SSCP-PU 会话 7

T

TCP/IP 端口号参数 93, 96, 97

TDU (拓扑数据库更新) 19

Telnet 客户机地址参数 95

TN 重定向器

- 访问记录 95
- 访问记录参数 95, 96, 97
- 工作表 144

TN 服务器

- 访问记录 92, 94
- 访问记录参数 92, 93
- 工作表 144
- 关联记录 94
- 关联记录参数 95

TN3270 客户机地址参数 92

TP

- 调用 9, 75
- 调用参数 77, 78, 79
- 可调用 9, 75
- 客户机 129
- 描述 4
- 目标 9, 75
- 配置 75
- 配置方法 76
- 源 9, 75
- APPC 定义参数 79, 80

TP 可执行文件的全路径参数 78

TP 名参数 77, 79

TP 配置参数

- ENV 156
- GROUP 155
- LUALIAS 155
- USERID, AIX 或 Linux 155
- USERID, Windows 157

TRS (拓扑和路由服务) 17

U

UCF

- 安全性 105, 107
- 对文件的访问权 107
- 描述 30, 101
- 命令语法 102, 105
- 取消命令 107
- 使用 105
- 守护程序 105
- 输出 106
- 许可权 105
- 样本命令 106
- 用户 105

UCF (续)

- 用户名 107
- 有效命令 106
- 允许的命令 106

UDP/IP 通信 111

UNIX 命令设施 (UCF) 30

USERID 参数

- AIX 或 Linux 155
- Windows 157

ux-cancel 命令 107

V

VRN

- 描述 22

VTAM (虚拟终端访问方法) 11

W

WebSphere Application Server 112

Windows 开放式系统体系结构 (WOSA) 113

Windows 客户机

- 网络访问过程 (NAP) 113

Windows 上的 Remote API Client

- 被调用 TP 116
- 服务器信息 117
- 记录信息 119
- 禁用 114
- 客户机 / 服务器跟踪信息 123
- 内部跟踪信息 124
- 配置 115
- 配置信息 116
- 启用 114
- 数据 123
- 域 116
- 状态 114
- admin_msg 123
- all_api 122
- API 跟踪信息 121
- appc 122
- APPCLU 125
- APPCTPN 125
- audit_file 120
- audit_file_wrap_size 120
- audit_logging_enabled 119
- backup_audit_file 120
- backup_error_file 120
- broadcast_attempt_count 117
- client_start_timeout 117
- cpic 122
- CPI-C 应用程序数据 125
- csv 122
- CSV 应用程序数据 125
- CSVTBLG 125
- datagram 123

Windows 上的 Remote API Client (续)

- error_file 119
- error_file_wrap_size 120
- exception_logging_enabled 119
- file1 121
- file1 (CS_tracing) 123
- file1 (Internal_tracing) 124
- file2 121
- file2 (CS_tracing) 123
- file2 (Internal_tracing) 124
- flip_size 121
- flip_size (CS_tracing) 123
- flip_size (Internal_tracing) 124
- lan_access_timeout 116
- log_directory 119
- nof 122
- receive 124
- rui 122
- send 123
- Server1 118
- Server2-Server9 118
- server_lost_timeout 117
- succinct_audits 121
- succinct_errors 121
- trace_flushing 124
- trace_level 124
- truncation_length 122

WOSA (Windows 开放式系统体系结构) 113

X

x snaadmin 程序 28



程序号: 5724-i33 和 5724-i34

中国印刷

S152-0886-01

