

IBM Communications Server for Linux on System z



入門

バージョン 6.2.2

IBM Communications Server for Linux on System z



入門

バージョン 6.2.2

お願い

本書および本書で紹介する製品をご使用になる前に、115 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM Communications Server for Linux、バージョン 6.2.2 および新しい版またはテクニカル・ニュースレターで明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： GC31-6769-01
IBM Communications Server for Linux on System z
Quick Beginnings
Version 6.2.2
Version 9.1

発 行： 日本アイ・ピー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2006.5

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、平成角ゴシック体™W7

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

© Copyright IBM Japan 2006

目次

表	vii
図	ix
IBM Communications Server for Linux へようこそ	xi
本書の用法	xi
参照リスト	xi
表記上の規則	xii
本書で使用される省略名	xiii
本リリースでの新機能	xiv
新しい機能	xiv
サポートしなくなった機能	xv
第 1 章 IBM Communications Server for Linux について	1
IBM Communications Server for Linux の機能とパッケージ化	1
IBM Communications Server for Linux	1
拡張ネットワーキング機能	8
機能および利点	14
多用途のビルディング・ブロック	14
クライアント/サーバー操作	14
容易な構成	15
追加の管理用ユーザー・インターフェースの選択	15
パフォーマンスの改善	16
セキュリティー・オプション	16
ネットワーク管理の柔軟性	17
信頼性、可用性、および保守容易性	17
ネットワークの統合、拡張および変更	18
第 2 章 ご使用のネットワークと Communications Server for Linux on System z の計画	19
ネットワーク計画の段階	19
ネットワークの機能要件の確認	19
Communications Server for Linux の構成方法の決定	20
インストールおよび操作のリソース要件の確認	20
命名規則	24
第 3 章 Communications Server for Linux の Linux サーバーへのインストール	25
Communications Server for Linux ライセンス・プログラムのパッケージ内容	25
Communications Server for Linux インストールの準備	26
前提ソフトウェアのインストール	26
製品インストール詳細の表示	26
言語環境変数の変更	26
旧レベルの Communications Server for Linux からのマイグレーション	27
Communications Server for Linux のライセンス・プログラムのインストール	29
Communications Server for Linux のインストール	29
Communications Server for Linux オンライン資料	31
Host Access Class Library (HACL)	31
WebSphere Application Server の構成	31
WebSphere Application Server のセキュア証明書のセットアップ	32
WebSphere Application Server の構成	32

サーバー構成ファイルのインストール	33
インストール後の作業	33
クライアント/サーバー操作	34
インストール後のクリーンアップ	34
PDF 資料の表示	34
現行リリース情報の検討	34
TN サーバーまたは TN リダイレクターで使用するための SSL の構成	34
Communications Server for Linux 構成ファイルのバックアップ	35
Communications Server for Linux 構成ファイルのバックアップ・コピーの復元	36
構成ファイルの再初期化	36
Communications Server for Linux のアンインストール	38
第 4 章 IBM Remote API Client on Linux のインストール	39
ハードウェアおよびソフトウェア要件	39
ハードウェア要件	39
Linux オペレーティング・システムのバージョン	40
Java	40
GSKIT	40
製品インストール詳細の表示	40
言語環境変数の設定	41
Remote API Client on Linux のインストール	41
GSKIT を使用した HTTPS セキュリティ証明書セットアップ	42
Remote API Client on Linux のアンインストール	44
第 5 章 System z に対する IBM Remote API Client on Linux のインストール	45
ハードウェアおよびソフトウェア要件	45
ハードウェア要件	45
Linux オペレーティング・システムのバージョン	45
Java	45
GSKIT	46
製品インストール詳細の表示	46
言語環境変数の設定	46
System z に対する Remote API Client on Linux のインストール	47
GSKIT を使用した HTTPS セキュリティ証明書セットアップ	48
System z 用の Remote API Client on Linux のアンインストール	50
第 6 章 IBM Remote API Client on AIX システムのインストール	51
ハードウェアおよびソフトウェア要件	51
ハードウェア要件	51
オペレーティング・システムのバージョン	51
Java	51
GSKIT	51
言語環境変数の変更	52
Remote API Client on AIX のインストール	52
AIX ワークステーションにファイルをコピーして Remote API Client をインストールする方法	52
CD からの Remote API Client のインストール	53
GSKIT を使用した HTTPS セキュリティ証明書セットアップ	54
Remote API Client on AIX のアンインストール	55
第 7 章 Remote API Client on Windows の計画とインストール	57
ハードウェアおよびソフトウェア要件	57
セットアップ・プログラムへのアクセス	58
セットアップ・プログラムを使用した Remote API Client on Windows のインストール	59
Remote API Client 構成用の拡張オプション	62
コマンド行からの Remote API Client ソフトウェアのインストール	63
GSKIT ソフトウェアのインストールとセキュリティ証明書セットアップ	65

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ	66
インストール後の Remote API Client ソフトウェアのカスタマイズ	67
Remote API Client ソフトウェアの再インストール	68
Remote API Client ソフトウェアのアンインストール	68
GSKIT ソフトウェアのアンインストール	69
ヘルプ	69
第 8 章 Communications Server for Linux の構成と使用	71
Communications Server for Linux の構成の計画	72
計画ワークシート	72
作業シート	73
Motif 管理プログラムの使用	73
Communications Server for Linux プログラムのパスの指定	73
Communications Server for Linux の使用可能にする方法	74
Motif 管理プログラムによる Communications Server for Linux の管理	74
クライアント/サーバー機能の構成	79
ノードの構成	80
接続の構成	81
従属トラフィックに対する SDLC リンクの構成	82
従属トラフィックと独立トラフィックをサポートするイーサネット・リンクの構成	83
Enterprise Extender リンクの構成	85
タイプ 0-3 LU の構成	86
タイプ 0-3 LU の定義	86
LU プールの定義	87
APPC 通信の構成	88
単純な APPN ネットワークの構成	89
従属 APPC の構成	94
CPI 通信のための構成	95
LUA の構成	96
LU プールの定義	96
SNA ゲートウェイの構成	98
暗黙のダウンストリーム LU のサポート	99
ダウンストリーム LU の定義	100
DLUR の構成	101
ローカル・ノードでの DLUR サポートの構成	102
ダウンストリーム・ノードをサポートするパススルー DLUR の構成	103
TN サーバーの構成	104
3270 LU の定義	105
LU プールの定義	105
TN3270 サーバーの構成	107
TN リダイレクターの構成	108
TN リダイレクターの構成	108
Communications Server for Linux を使用不可にする方法	109
Communications Server for Linux の自動的な始動	110
Communications Server for Linux の使用可能にする方法	110
SNA ノードの初期化	111
ポートとリンク・ステーションの活動化	111
レポート時の Communications Server for Linux の開始	111
第 9 章 Communications Server for Linux と SNA の情報リソース	113
SNA ライブラリー	113
ネットワークで入手可能な情報	113
推薦資料	114
付録. 特記事項	115
商標	117

参考文献	119
Communications Server for Linux バージョン 6.2.2 資料	119
システム・ネットワーク体系 (SNA) 資料	120
ホスト構成資料	121
z/OS Communications Server 資料	121
TCP/IP 資料	121
X.25 資料	121
APPC 資料	122
プログラミング資料	122
他の IBM ネットワーキング資料	122
索引	123

表

1. 本書内容の参照リスト	xii
2. 表記上の規則	xii



1.	Communications Server for Linux on System z のホスト通信.	2
2.	複数のダウンストリーム Linux コンピューターをホスト・コンピューターにリンクする SNA ゲートウェイ	9
3.	Branch Extender	10
4.	TN サーバー	12
5.	「ノード (Node)」ウィンドウ	76
6.	Communications Server for Linux ツールバー	79
7.	SNA ゲートウェイ	98
8.	DLUR を提供する Communications Server for Linux ノード	102

IBM Communications Server for Linux へようこそ

本書では、IBM® Communications Server for Linux on System z (Communications Server for Linux®) について紹介します。この IBM ソフトウェア製品は、Linux を実行するコンピューターがシステム・ネットワーク体系 (SNA) ネットワークのほかのノードと情報を交換できるようにするものです。

IBM Communications Server for Linux に関して 2 つの異なるインストール方法があり、この製品が稼働するハードウェアにより異なります。

Communications Server for Linux

Communications Server for Linux (プログラム・プロダクト番号 5724-i33) は、以下の環境で稼働します。

- Linux で稼働する 32 ビット Intel® ワークステーション (i686)
- Linux で稼働する 64 ビット AMD64/Intel EM64T ワークステーション (x86_64)
- Linux で稼働する IBM pSeries® コンピューター (ppc64)

Communications Server for Linux on System z

Communications Server for Linux on System z (プログラム・プロダクト番号 5724-i34) は、Linux for System z (s390 または s390x) System z を実行するメインフレームで稼働します。

「*Communications Server for Linux 入門*」の資料には 2 つの異なるタイプがあり、それぞれが上記のシステムごとに対応しています。ご使用の Communications Server for Linux システムに適した資料をご用意してください。本書は、Communications Server for Linux on System z に適用されます。

Communications Server for Linux は、多種多様なネットワークのニーズとソリューションのためのビルディング・ブロックを提供します。これらを使用すると、SNA ネットワークのノード間で情報を交換でき、TCP プロトコル/インターネット・プロトコル (TCP/IP) を介して通信する Telnet プログラムに対してホスト・アクセスを提供できます。

本書の使用法

この節では、本書の構成と表記法について説明します。

参照リスト

本書は、ネットワーク計画に関する管理およびテクニカル担当、および Linux オペレーティング・システム用の Communications Server に関心を持つ方々を対象としています。

Communications Server for Linux を初めてご使用になる際に必要な情報を xii ページの表 1 で探してください。

表 1. 本書内容の参照リスト

内容	参照先
Communications Server for Linux について読む	1 ページの『第 1 章 IBM Communications Server for Linux について』
ネットワーク内での Communications Server for Linux の使用計画	19 ページの『第 2 章 ご使用のネットワークと Communications Server for Linux on System z の計画』
Linux サーバー上での Communications Server for Linux のインストール	25 ページの『第 3 章 Communications Server for Linux の Linux サーバーへのインストール』
Linux (32 ビット Intel、64 ビット Intel/AMD、または pSeries) 上に Remote API Client のインストール	39 ページの『第 4 章 IBM Remote API Client on Linux のインストール』
Linux for System z 上に Remote API Client のインストール	45 ページの『第 5 章 System z に対する IBM Remote API Client on Linux のインストール』
Remote API Client on AIX® のインストール	51 ページの『第 6 章 IBM Remote API Client on AIX システムのインストール』
Remote API Client on Windows® のインストール	57 ページの『第 7 章 Remote API Client on Windows の計画とインストール』
Communications Server for Linux の構成	71 ページの『第 8 章 Communications Server for Linux の構成と使用』
オンライン情報を含む Communications Server for Linux 資料およびその他の資料の検索	113 ページの『第 9 章 Communications Server for Linux と SNA の情報リソース』
特記事項および商標情報について	115 ページの『特記事項』

表記上の規則

表 2 に、本書で使用されている表記上の形式を示します。

表 2. 表記上の規則

内容	表記例
強調	削除する前に、ファイルのバックアップをとります。
資料名	<i>Communications Server for Linux 管理ガイド</i>
ファイル名、パス名	<code>/usr/spool/uucp/myfile.bkp</code>
プログラム、アプリケーション	<code>snaadmin</code>
ユーザーの入力	0p1
コンピューターからの出力	CLOSE

本書で使用される省略名

本書で使用されている省略名を以下に示します。

AIX	AIX
API	アプリケーション・プログラミング・インターフェース
APPC	拡張プログラム間通信機能
APPN	拡張対等通信ネットワーク (Advanced Peer-to-Peer Networking [®])
BrNN	ブランチ・ネットワーク・ノード
COS	サービス・クラス
CPI-C	共通プログラミング・インターフェース・コミュニケーション
CSV	共通サービス verb
DDDLU	従属 LU の動的定義
DES	データ暗号化規格
DLC	データ・リンク制御
DLUR	従属 LU リクエスター
DLUS	従属 LU サーバー
FTP	ファイル転送プロトコル
HPR	高性能ルーティング
IETF	Internet Engineering Task Force
ISO	国際標準化機構
ISR	中間セッション・ルーティング
LAN	ローカル・エリア・ネットワーク
LDAP	Lightweight Directory Access Protocol
LEN	ローエントリー・ネットワーク
LLC2	論理リンク制御 2
LU	論理装置
LUA	従来型 LU アプリケーション・プログラミング・インターフェース
MDS-NMVT	マルチドメイン・サポート - ネットワーク管理ベクトル・トランスポート
MPC	マルチパス・チャネル
MS	Management Services
NMVT	ネットワーク管理ベクトル・トランスポート
NOF	Node Operator Facility
OSI	オープン・システム相互接続
PU	物理装置
RFC	Request For Comments
RLE	ラン・レンダス・エンコード
SAA	システム・アプリケーション体系 (Systems Application Architecture [®])
SAP	サービス・アクセス・ポイント
SNA	システム・ネットワーク体系
SSL	Secure Sockets Layer
TCP/IP	TCP/IP プロトコル
TN	Telnet
TP	トランザクション・プログラム
VT	仮想端末
WAN	広域ネットワーク

本リリースでの新機能

Communications Server for Linux バージョン 6.2.2 は、Communications Server for Linux 6.2 と Communications Server for Linux Version 6.2.1 を置き換えます。

まだサポートされている当製品のリリースは、以下のとおりです。

- Communications Server for Linux バージョン 6.2
- Communications Server for Linux バージョン 6.2.1

当製品の以下のリリースは最早サポートされません。

- Communications Server for Linux バージョン 6.0.1 (V6.0.1)。このバージョンは、PRPQ 5799-RXL として入手可能だったものです。

新しい機能

Communications Server for Linux のこのリリースで追加された機能は、以下のとおりです。

- クライアント/サーバー・サポートが、現在、広範囲なサーバーとクライアント・コンピューターを包含するようになりました。
 - サーバーとしては、Linux for System z (s390 または s390x) を稼働する System z メインフレームです。
 - クライアントとしては、i686、x86_64、ppc64、s390 または s390x システム、AIX ワークステーション、または 32 ビット・バージョンの Microsoft® Windows (Microsoft Windows 2000、2003、または XP) または x64 バージョンの Microsoft Windows (Microsoft Windows Server 2003 x64 Edition または Microsoft Windows XP Professional x64 Edition) を稼働する PC です。上記のクライアントは Communications Server for Linux サーバー (または CS/AIX サーバー) との通信を行います。この通信には、TCP/IP を使用するか、または WebSphere® サーバー経由の HTTPS を使用します。
 - Linux、Linux for System z、および AIX クライアントは、Linux または Linux for System z サーバー上で使用可能な API ライブラリーのすべてをサポートします。(以前の CS/AIX v4.2 製品に含まれていたこの API は、当バージョンではサポートされません。)
 - Windows クライアント (32 ビットまたは x64 バージョンの Microsoft Windows 上のクライアント) は、APPC、CPI-C、CSV、LUA (LUA RUI および SLI の両方) および NOF API をサポートします。
 - NOF API に対するクライアント・サポート (すべてのクライアント・タイプで) が提供するものは、照会機能だけです。リソースの定義、削除、開始、または停止の呼び出しはサポートされません。
 - Windows クライアントは、Windows Terminal Server (WTS) 環境で稼働します。この環境では、複数のユーザーが同じクライアントを共有しますが、それぞれのアプリケーションは相互に干渉することなく独立して実行されます。
- SLES10 Linux ディストリビューション現在サポートされています (サーバーとクライアントの両方に対して)。
- Communications Server for Linux は、現在、Linux サーバー上の WAN アダプターを使って SDLC および X.25 (QLLC) 接続をサポートします。これには、アダ

プター・ベンダーからのデバイス・ドライバーと DLC コードが必要です。IBM からはこれらのアダプターまたはデバイス・ドライバーは提供されません。適切な WAN アダプター・ベンダーの詳細は、Communications Server for Linux サポートの Web ページを参照してください。

- TN サーバーと TN リダイレクターは、現在、特別なローカル・アドレスを指定する機能が含まれています。このローカル・アドレス上で TN パススルー・サービスがクライアント接続を `listen` することになります。このフィーチャーはオプションです。このサービスを構成して、どのローカル・アドレス上でも TN3270 クライアント接続をサポートでき、あるいは指定されたアドレスに TN3270 クライアントを制限できます。

サポートしなくなった機能

このクライアントとサーバーのコードは RedHat Advanced Server 2.1 上では最早サポートされません。

第 1 章 IBM Communications Server for Linux について

この章では、Communications Server for Linux のパッケージ化の方法、ならびに機能、フィーチャー、および利点について説明します。

IBM Communications Server for Linux の機能とパッケージ化

Communications Server for Linux は、Linux オペレーティング・システムで稼働する通信用ソフトウェアです。これは、『IBM Communications Server for Linux』および 8 ページの『拡張ネットワーキング機能』で説明している機能から構成されます。

Communications Server for Linux on System z は、サーバーを統合してネットワークを単純化することができるソリューションです。

- System z テクノロジーを使用して、サーバーを単一のハードウェア・プラットフォームに統合できます。このテクノロジーにより、複数の仮想サーバー間でプロセッサおよびメモリーを共用できるなどの利点が得られます。
- SNA と IP のデュアル・ネットワークを IP のみのネットワークに単純化できます。これにより、SNA トラフィックが、Communications Server for Linux on System z へ IP ネットワーク経由で流れ、SNA に変換されて高信頼性のセキュア・インターフェースを介して z/OS® CS に送信されます。さらに、ネットワークの単純化による利点として、SNA スキルが 1 つの場所でしか必要とされなくなるということもあります。

IBM Communications Server for Linux

IBM Communications Server for Linux は、SNA ネットワークと TCP/IP ネットワーク間を横断的にアプリケーションを接続します。Linux を実行している System z VM または LPAR に SNA リソースおよびプロトコルを提供することにより、SNA ノードに変換します。これにより、SNA ネットワーク内のその他のコンピューターと通信できるようになります。TCP/IP 機能も提供されるので、TCP/IP ネットワーク内において、あるいは TCP/IP ネットワークと SNA ネットワークの境界においても IBM Communications Server for Linux を使用できます。

Communications Server for Linux は SNA ホスト・コンピューターと通信する場合に、さまざまな方法で作動します。2 ページの図 1 に、Communications Server for Linux の配置例を 2 つ図示します。

- 第 1 の例では、Communications Server for Linux はメインの z/OS システムをオフロードするために、別の z800 システムにインストールされています。この 2 つのシステムを接続するために、IP を使用する Enterprise Extender リンクまたは LLC2 リンクが使用されています。
- 第 2 の例では、Communications Server for Linux はメインの z/OS システム内の 1 つ以上の VM または LPAR にインストールされています。Communications Server for Linux および z/OS Communications Server は同じメインフレームにイ

IBM Communications Server for Linux の機能とパッケージ化

インストールされていますが、これらは別々の SNA ノードです。そのため、これらを接続するために、HyperSockets IP を使用する Enterprise Extender リンクまたは LLC2 リンクが必要です。

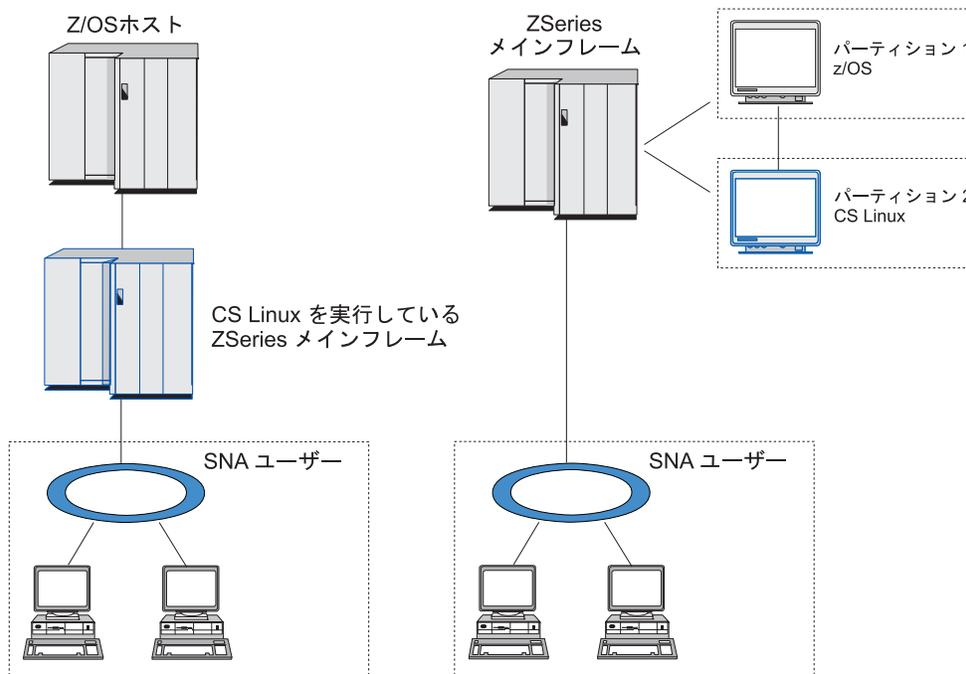


図1. Communications Server for Linux on System z のホスト通信

図に示される 2 つの配置は概念上は同じものであり、双方に同じ Communications Server for Linux 構成 (Communications Server for Linux と SNA ホスト間の通信リンクなど) が必要です。説明を明確にするために、本書の図では、Communications Server for Linux と SNA ホストが別々のコンピューターに置かれている第 1 の配置例を示します。

Communications Server for Linux では、以下のサービスが提供されます。

ネットワーク・サポート

Communications Server for Linux は、サブエリアと対等ネットワーク機能を以下のようにサポートします。

SNA サブエリア・ネットワーク

これらのネットワーク (ホスト介在ネットワークとも呼ばれる) では、1 つ以上のホスト・コンピューターが階層的に編成されます。各ホスト・コンピューターは、コンピューター間の通信を制御し、ネットワークを管理して、処理サービスおよび大容量データ・ストレージを提供します。ネットワーク内のその他のすべてのノードは、ホストの制御に依存しています。

Linux コンピューターは、ホストに依存するノードとして構成すれば、サブエリア・ネットワークに組み入れることができます。

対等ネットワーク

分散処理環境の場合、Communications Server for Linux は APPN ネットワーク

IBM Communications Server for Linux の機能とパッケージ化

ットワークをサポートします。これらの対等ネットワーク内で、Linux コンピューターは処理機能を保持し、互いに対等に直接通信します。

APPN ネットワークは、以下のタイプの対等ノードで構成されます。

- APPN ネットワーク・ノード (トラフィック制御、動的経路計算および選択サービス、ならびにネットワーク管理サービスを提供します)
- APPN エンド・ノード (APPN ネットワーク・ノード・サービスを使用して、対等ノードと通信します)
- LEN ノード (隣接ノードまたは隣接しているように構成されているノードと直接に通信します)

注: ホスト・コンピューターは、独立 LU 6.2 を使用してネットワーク内の Linux コンピューターおよびその他のホストと通信し、APPN ネットワーク内の対等ノードとして機能することができます。

APPN ネットワーク内でのサブエリア機能の提供

従属 LU リクエスター (DLUR) 機能により、ホストとホスト従属ノードの間のトラフィックを、APPN ネットワーク内で受け渡すことができます。

データ・リンク制御オプション

リンク・レベルで、Communications Server for Linux により多様な接続オプションが提供され、ネットワークのサイズ、速度、セキュリティ、およびコストに関する要件を満たすのに役立ちます。(サポートされるリンク・タイプの詳細リストについては、22 ページの『インストール要件』を参照してください。) これは、以下のように、異なるネットワーク・タイプのデータ・リンクをサポートします。

ローカル・エリア・ネットワーク

LAN 接続の場合、トークンリング、標準イーサネット、および 802.3 イーサネット・プロトコルを使用して、適切な通信リンクをインストールできます。

広域ネットワーク

Communications Server for Linux は SDLC および X.25 (QLLC) 接続をサポートします。これは、各プラットフォーム上での OEM アダプター・サポートに依存します。

ローカル接続

Communications Server for Linux は、ローカル接続用のマルチパス・チャンネル (MPC) 接続をサポートします (Communications Server for Linux on System z のみ)。

IP 統合

企業のバックボーン・ネットワークが IP に基づいている場合、Communications Server for Linux の Enterprise Extender (HPR/IP) 機能を使用して、これを SNA に統合するので、SNA アプリケーションが IP ネットワークを通して通信できるようになります。

LU サポート

論理装置 (LU) は、SNA ネットワークの各ノードに常駐するアプリケーション固有のネットワーク・リソースです。各 LU は、ネットワークを介してその他のノード上のパートナー・アプリケーションと通信するために、アプリケーションがリンクにアクセスする際のインターフェースとして機能します。

Communications Server for Linux は、アプリケーションの異なるクラスごとに異なるタイプの LU をサポートします。

- サブエリア・ネットワークでは、Communications Server for Linux は、以下のタイプのいずれかの従属 LU をサポートしています。
 - LU 0
 - LU 1
 - LU 2
 - LU 3
 - LU 6.2

LU 0 は、基本的なプログラム間の通信をサポートします。これは、通常、小売業の POS トランザクションや銀行業務で使用します。LU 2 は、Linux コンピューターが IBM 3270 ファミリー端末をエミュレートできるように、端末エミュレーション・アプリケーションをサポートします。その他の LU タイプは、アプリケーションを分散処理に組み込んだり、各種プリンターや対話式ディスプレイ端末と通信したりできるようにします。

Communications Server for Linux は、従属 LU の動的定義 (DDDLU) を使用するホスト・システムをサポートします。DDDLU は、SNA システムからホストへの通信リンクが確立されたとき、SNA システムの従属 LU をホスト構成に追加できるホスト機能です。DDDLU を指定しておくと、LU をホストに静的に構成する必要はありません。(しかし、Communications Server for Linux ノードで従属 LU を定義していなければなりません。) これにより、ホストに必要な初期構成が少なくなり、その後の拡張が容易になります。

Communications Server for Linux は、構成を変えなくても、DDDLU 可能なホストと DDDLU 可能でないホストの両方と通信できます。

Communications Server for Linux ノードからホストへの通信リンクが確立されると、DDDLU 可能なホストは、DDDLU をサポートしていることをノードに通知します。ノードは、次に、リンクを使用する従属 LU を定義するために必要な情報を送信します。ホストが DDDLU 可能でない場合、Communications Server for Linux はこの情報を送信しません。これは、LU がホストで既に静的に定義されていることを前提としています。

- 独立 LU 6.2 は、APPN ネットワークの独立トラフィックをサポートします。独立 LU 6.2 は、自立走行式通信およびネットワーク管理ならびに分散処理をサポートします。

IBM Communications Server for Linux の機能とパッケージ化

さらに、Communications Server for Linux の DLUR 機能により、従属 LU のトラフィックが APPN ネットワーク上を通ることができるようになります。

- プライマリー RUI サポートを使用すると、Communications Server for Linux アプリケーションがダウンストリーム LAN/WAN 接続の従属 LU 装置を管理できるようになります。それは、このアプリケーションがメインフレームであるかのように機能します。この機能には接続上の制約がいくつかありますが、完全なメインフレーム・アプリケーションに要求されることを必要とせず、アプリケーションが従属 LU 装置間でのデータの受け渡しを行えるようになります。

セッション・サポート

セッションは、パートナー LU 間の一時的な論理チャネルです。通常、各 LU に関連するパートナー・アプリケーションは、セッションを介して通信を行います。Communications Server for Linux は何千ものセッションをサポートできます。Communications Server for Linux は、1 次 LU と 2 次 LU の両方が同じ Linux コンピューターに置かれる U 型セッション（「ローカル/リモート透過型」とも呼ばれます）もサポートできます。これにより、リンク接続しなくても、1 台のコンピューターでソースとターゲットのトランザクション・プログラムのペアの開発やテストを行うことができます。

2 つのパートナー LU 間のセッション上のデータ・フローが圧縮されて、必要な帯域幅が減少する場合があります。

- LU タイプ 6.2 の場合、Communications Server for Linux を使用すると、セッションが使用するモードの構成で圧縮の使用を指定できます。さまざまな圧縮レベル（RLE、LZ9、または LZ10）を提供する圧縮アルゴリズムから、使用するアルゴリズムを指定できます。また、セッション上の異なる方向のデータ・フローに対して別の圧縮レベルを指定したり、一方向の圧縮レベルは指定して他の方向の圧縮レベルを指定しないこともできます。
- LU タイプ 0 から 3 の場合、Communications Server for Linux を使用すると、セッションが使用するリンク・ステーションまたは PU の構成で圧縮の使用を指定できます。インバウンド方向では RLE 圧縮が使用され、アウトバウンド方向では LZ9 圧縮が使用されます。

API サポート

Communications Server for Linux には、アプリケーション・プログラミング・インターフェース（API）があり、特定のタイプの LU、分散処理、ネットワーク管理、および Communications Server for Linux 自身を管理するアプリケーション開発に用いられます。Communications Server for Linux は、その他のオペレーティング・システム上で稼働する Communications Server ファミリーのメンバーが提供する API と互換性のある一連の API を提供します。

API は、トランザクション・プログラム（TP）がサポートする LU と通信できるインターフェースです。API は、verb（関数、呼び出し、サブルーチンともいう）のライブラリーから構成されています。ここから、TP は LU に渡す必要のあるものを選択し、SEND_DATA などのアクションを要求します。LU は、適切なプロトコルに従って、順に verb を処理し、デー

タ・ストリームを作成して、宛先アドレスを示すヘッダーを付加し、パートナー LU へのリンクを通してデータを送信します。

共通プログラミング・インターフェース・コミュニケーション (CPI-C) は、移植性があるため、最も効果的な API の 1 つになっています。従属および独立 LU 6.2 をサポートするために導入された CPI-C は、システム・アプリケーション体系 (SAA[®]) の指示に準拠し、異なるプラットフォームおよびオペレーティング・システムを統合します。CPI-C は、すべてのシステムに共通の構文規則のセットを使用します。そのため、これが標準となりました。

Communications Server for Linux には、標準 C 言語 CPI-C API の他に Java[™] アプリケーション用の CPI-C API もあります。詳しくは、「*Communications Server for Linux CPI-C プログラマーズ・ガイド*」を参照してください。Communications Server for Linux 資料には、特に断りのない限り、CPI-C のリファレンスすべてに Java CPI-C が含まれています。

その他の Communications Server for Linux API には、以下のものが含まれます。

- APPC API。LU 6.2 を使用するアプリケーション・プログラム間の対等通信を行います。この API には、非ブロッキングにするオプションがあります。TP が非ブロッキング verb を使用すると、要求されたアクションが完了していなくても、API は TP に制御を戻すことができます。後にアクションが完了した時点で、TP はその通知を受け取ります。
- LUA API。ホスト・アプリケーションと通信を行います。
- CSV (共通サービス verb) API。文字変換やアプリケーション・トレース制御などのユーティリティー関数を提供します。

さらに、Communications Server for Linux には、所有権のある以下のプログラミング・インターフェースがあります。

- MS (管理サービス) API。ネットワーク・メッセージ処理機能を持ちます。
- NOF (ノード・オペレーター機能) API。アプリケーションによって Communications Server for Linux リソースを構成および管理します。

API について、詳しくは、API のプログラミング・ガイドを参照してください。

クライアント/サーバー・サポート

Communications Server for Linux を実行するコンピューターは、クライアント/サーバー・プロトコルを使用して通信するように構成できます。ネットワーク内でクライアント/サーバー・プロトコルが使用されている場合、そのネットワーク内のクライアント/サーバー・プロトコルを使用して通信を行うすべてのコンピューターは「ドメイン」と呼ばれます。

クライアント/サーバー構成で Communications Server for Linux を実行するコンピューターは、次の役割を担うことができます。

- サーバー。SNA ノードおよび関連する接続コンポーネントが収容されず。サーバーは、ローカル・システム上または Communications Server for

IBM Communications Server for Linux の機能とパッケージ化

Linux ドメイン内のその他のコンピューター上のアプリケーションに SNA 接続を提供します。サーバーは、Linux システムである必要があります。

- Remote API Client には SNA ノード・コンポーネントは収容されず、サーバーを介して SNA ノード・コンポーネントにアクセスします。クライアントは、同時に 1 つ以上のサーバーにアクセスでき、必要に応じて同時アプリケーションを実行できます。クライアントは、AIX、Linux、または Windows で実行することができます。(Linux コンピューターは、サーバーまたはクライアントのいずれかであることができますが、両方は不可能です。このため、同一コンピューター上にサーバーとクライアントの両方をインストールできません。

サーバーとクライアントは、TCP/IP を使用して Communications Server for Linux ドメインを通じて通信します。代替方法として、サーバーとクライアントは WebSphere サーバー経由の HTTPS を使用して通信可能です。これは、セキュリティ証明書を使用してクライアント接続を認証します。クライアントがパブリック・ネットワークを横断して接続する場合は、通常は HTTPS を使用する必要が生じます。

複数の Communications Server for Linux サーバーがあるドメインでは、1 つのサーバーが Communications Server for Linux ドメイン構成ファイルのマスター・コピーを保持します。このサーバーは、マスター・サーバーと呼ばれます。ドメイン内のその他のサーバーは、バックアップ・サーバーとして定義するか、ピア・サーバーとして残すこともできます。ドメイン構成ファイルは (バックアップ・サーバーの始動時またはマスター・コピーの変更時に) バックアップ・サーバーにコピーされるため、すべてのバックアップ・サーバーが最新情報のコピーを保持することになります。ピア・サーバーは、必要に応じてドメイン構成情報をマスター・サーバーから取得しますが、バックアップ・サーバーとして機能することはできません。

マスター・サーバーに障害が発生した場合、ドメインに定義されたサーバーのリストで先頭のバックアップ・サーバーがマスターの役割を担当します。このサーバー上のドメイン構成ファイルがマスター・コピーとして使用され、必要に応じてその他のサーバーにコピーされます。マスター・サーバーが再始動されると、現在マスターとして機能しているバックアップ・サーバーからドメイン構成ファイルのコピーを受け取り、マスターとしての役割を継承します。

分散アプリケーションのサポート

クライアント/サーバーの Communications Server for Linux システムでは、Remote API Client 上で実行されるアプリケーションが、サーバー上の接続リソースと連携して単一のタスクを実行します。その他の (Communications Server for Linux 以外の) コンピューター上で実行されるアプリケーションも、Communications Server for Linux コンピューター上のアプリケーションと連携して分散処理を実行できます。

Communications Server for Linux は、APPC (LU 6.2 とも呼ばれます) を使用する分散アプリケーションをサポートします。

拡張ネットワーク機能

Communications Server for Linux のベース製品には、拡張ネットワーク機能を追加するための機能セットが組み込まれています。これらの機能には、以下のものがあります。

- SNA ゲートウェイは、LAN をサブエリア SNA ネットワークに接続します。
- プライマリー LU サポートを使用すると、ホスト・メインフレーム・アプリケーションと同じ方法でダウンストリームの従属 LU 装置を制御するためのサポートが提供されます。
- Branch Extender は、リソースを異なるロケーション (たとえば、大規模な組織内の異なる事業所) に分割して、大規模 APPN ネットワークの規模を単純化します。これにより、効率の良いリソース・ロケーションを維持しながら、保管に必要なトポロジーに関する情報量を削減できます。
- APPC Application Suite は、APPN ネットワークで使用するための各種アプリケーションを提供します。
- Enterprise Extender (EE、HPR/IP とも呼ばれます) は、SNA トラフィックが IP ネットワークを経由してそのままトランスポートできるようにします。
- TN サーバーは、TN3270 クライアントと総称される TN3270 クライアントと TN3270E クライアントが、SNA 経由でホストにアクセスできるようにします。
- TN リダイレクターにより TCP/IP をパススルーして、TN3270、TN3270E、TN5250 および VT クライアント (これらは、Telnet クライアントと総称されま) がホストにアクセスできるようにします。

SNA ゲートウェイ

ゲートウェイは、異種のネットワークまたはコンピューター・システムを接続する、ユーザーに透過的な装置であり、接続されている両方の環境をサポートします。エンド・ユーザー同士は、互いに同じネットワーク内に存在しているように認識します。

SNA ゲートウェイを使用すると、Communications Server for Linux コンピューターは、SNA ネットワーク内の複数のダウンストリーム・コンピューターを 1 つ以上のホスト物理装置 (PU) にリンクさせるゲートウェイとして動作できるようになります。これについては、9 ページの図 2 で説明しています。ホスト接続を単純化して、余分なリンクを減らすため、SNA ゲートウェイは PU コンセントレーターとして動作します。これは、複数のコンピューターを単一の PU として扱い (この PU は SNA ゲートウェイ・ノードにあると認識されます)、1 つの物理接続でホストと通信します。

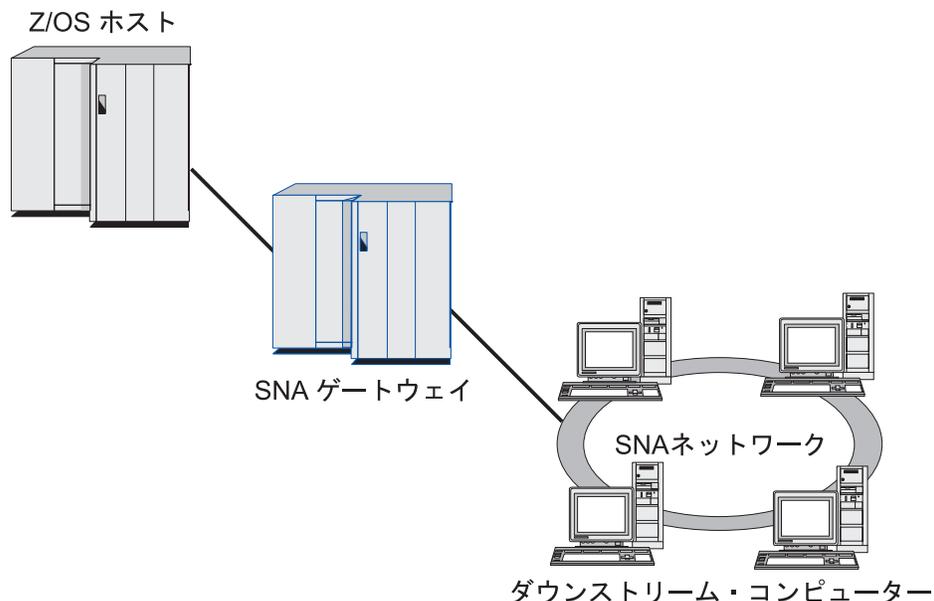


図2. 複数のダウンストリーム Linux コンピューターをホスト・コンピューターにリンクする SNA ゲートウェイ

プライマリー LU サポート

プライマリー LU サポートを使用すると、Linux アプリケーションが、あたかもホスト・メインフレーム・アプリケーションであるかのように、ダウンストリーム従属 LU 装置を制御できるようになります。

通常、LUA アプリケーションはセカンダリー LU としてホスト・メインフレームに接続します。それにより、ホスト・アプリケーションはそのセッション用の定義を制御し、セッション開始のための BIND を送信する責任があります。また、Communications Server for Linux には、プライマリー RUI インターフェースを使用して、LAN 経由のダウンストリーム従属の SNA 装置に対してプライマリー LU として機能する能力が含まれます。このインターフェースを使用して、アプリケーションがホスト・メインフレームを使用せずにダウンストリーム従属 LU セッションを接続できます。

プライマリー LU アプリケーションを使用するには、ホスト LU 名 #PRIRUI# を使ってダウンストリーム LU (または、ダウンストリーム PU テンプレート) を指定してそのノードを構成する必要があります。これにより、プライマリー RUI を使用するアプリケーションはこれらの PU とその PU に割り当てられた LU リソースを制御することをサーバーに示します。この PU が使用可能なのは、LAN ポート上だけです。プライマリー RUI を使用するアプリケーションをプログラミングする方法は、「*Communications Server for Linux LUA Programmer's Guide*」を参照してください。

Branch Extender

APPN ネットワーク内のネットワーク・ノードは、トポロジー情報 (ネットワーク内のその他のノードのロケーションとノード間の通信リンク) を保持しており、トポロジーの変更時にネットワークに関するこの情報を転送します。ネットワークの

IBM Communications Server for Linux の機能とパッケージ化

サイズが大きくなると、保管する情報量やトポロジー関連のネットワーク・トラフィックが大きくなり、管理が難しくなる可能性があります。

ネットワークを複数のサブネットワークに分けると、こうした問題は避けられ、各ノードは、そのサブネットワーク内のノードに関するトポロジー情報を保持しているのみで済みます。ただし、こうすると、他のサブネットワーク内のリソースを探し出そうとするときに、ネットワーク・トラフィックを増加させることとなります。

図3 に示す、APPN の Branch Extender 機能を使用すれば、こうした問題は解消されます。

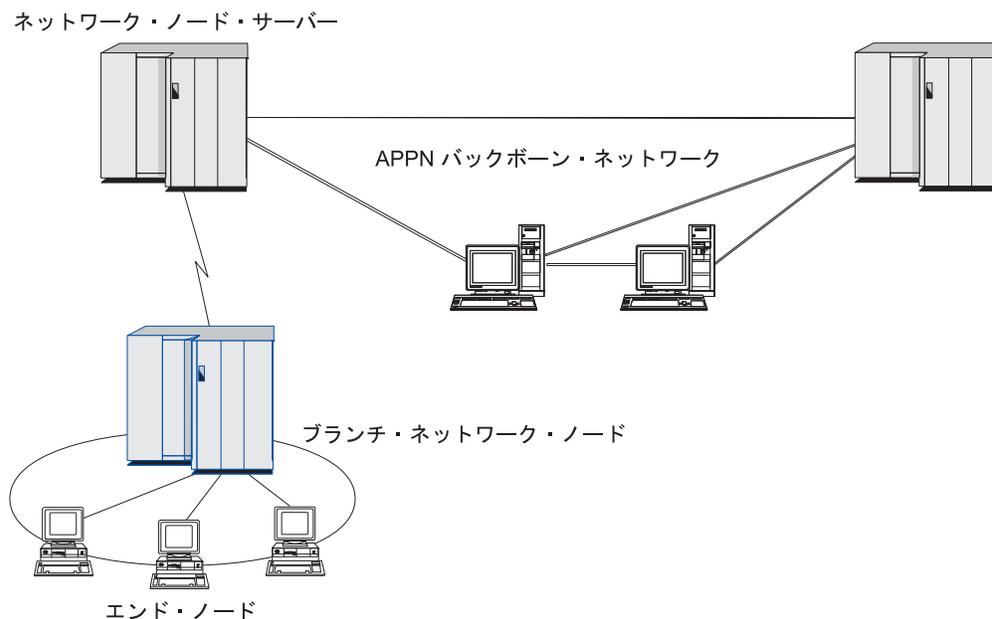


図3. Branch Extender

名前のおり、Branch Extender は、大規模な組織から分割した部門のように、ネットワークを個々の領域に分割できるよう設計されています。これは、メインのバックボーン APPN ネットワーク (たとえば、組織の本社内のネットワーク) から各ブランチに分割することで機能します。

各ブランチは、ブランチ・ネットワーク・ノード (BrNN) という新しいタイプのノードを持ち、メインの APPN バックボーン・ネットワークのネットワーク・ノードに接続されています。BrNN は、APPN ネットワーク・ノードと APPN エンド・ノードの機能を結合させます。

- 以下に説明するように、バックボーン・ネットワークからは、BrNN はバックボーン・ネットワーク内のネットワーク・ノード・サーバー (NNS) に接続されたエンド・ノードとして認識されます。
 - バックボーン・ネットワーク内のノードは、ブランチ内のノードを認識しないため、保管すべきトポロジー情報量が削減されます。
 - BrNN はエンド・ノードとして認識されるため、バックボーン・ネットワークからトポロジー情報を受信することはありません (トポロジー情報は、ネットワーク・ノード間でしか送信されません)。

- BrNN は、ブランチ内のすべてのリソースを、BrNN 自身が保有しているものとして NNS に登録します。つまり、バックボーン・ネットワーク内のノードは、ブランチ内の別のノードにあることを認識していなくても、ブランチ内のリソースを探し出すことができます。
- ブランチ・ネットワークからは、BrNN はネットワーク・ノードと認識され、ブランチ内のエンド・ノードの NNS として動作します。ブランチ内の各ノードは、ネットワーク他のノードも標準 NNS と同様に NNS を経由して接続されているものと認識します。

APPC Application Suite

APPC Application Suite は、APPN ネットワークの分散処理能力を実証する一連のアプリケーションです。これは、構成の検証および問題の判別に役立ちます。APPC Application Suite を使用すると、ファイル転送など、ネットワークを介して頻繁に実行される操作をサポートできます。

APPC Application Suite は、以下のアプリケーションで構成されています。

- **ACOPY** (APPC COPY)
- **AFTP** (APPC File Transfer Protocol)
- **ANAME** (APPC Name Server)
- **APING** (APPC Ping)
- **AREXEC** (APPC Remote EXECution)
- **ATELL** (APPC TELL)

これらのアプリケーションには、サーバー、もしくは Linux または Windows クライアントからアクセスできます。

Enterprise Extender

Enterprise Extender (HPR/IP と呼ばれます) は、SNA アプリケーションを IP ネットワークに組み込む機構を提供します。

SNA アプリケーションは、SNA プロトコルを使用し、SNA ネットワークを介して、他の SNA アプリケーションと通信するよう設計されています。Enterprise Extender を使用して TCP/IP ネットワークにインストールすれば、SNA アプリケーションは引き続き通信可能です。Enterprise Extender 機能により、IP ネットワークを経由して SNA プロトコルをトランスポートする機構が提供されるためです。特に、APPN 高性能ルーティング (HPR) 機能が提供されているので、アプリケーションに APPN と IP の両方の接続の利点が与えられます。

Communications Server for Linux の Enterprise Extender は、単に通信リンクとしてインプリメントされます。IP 経由で 2 つの SNA アプリケーションを接続するには、SDLC またはイーサネットなどの他のリンク・タイプの場合のように、Enterprise Extender リンクを定義します。

TN サーバー

SNA ネットワークの代わりに TCP/IP を介して通信を行う 3270 エミュレーション・プログラムは、TN3270 プログラム (Telnet 3270 エミュレーション・プログラム) と呼ばれます。

IBM Communications Server for Linux の機能とパッケージ化

TN3270 プログラムには、TN3270E (Telnet 3270 標準拡張) サポートを組み込むこともできます。TN3270E は、Telnet を使用して 3270 装置エミュレーション (端末とプリンターの両方を含む) をサポートします。これにより、Telnet クライアントから (LU 名または LU プールの名前を指定して) 特定の装置を選択できるようになります。また、ATTN キーおよび SYSREQ キー、SNA 応答処理などの各種機能も拡張サポートされます。

注: 本書では、TN3270、TN3287、TN3270E プロトコルのすべてに適用できる情報については、TN3270 という用語を使用します。

Communications Server for Linux TN サーバーを使用すると、3270 ホスト・コンピューター以外のコンピューターの TN3270 ユーザーが、この 3270 ホスト・コンピューターへアクセスできるようになります。TN サーバーにより、直接リンクを要求せずに TN3270 ユーザーは Communications Server for Linux、またはその他の TN3270 ユーザーとホスト接続を共有できます。また、TN3270 ユーザーは、TCP/IP を実行していないホストへもアクセスできます。

Communications Server for Linux TN サーバー機能を、図 4 に示します。

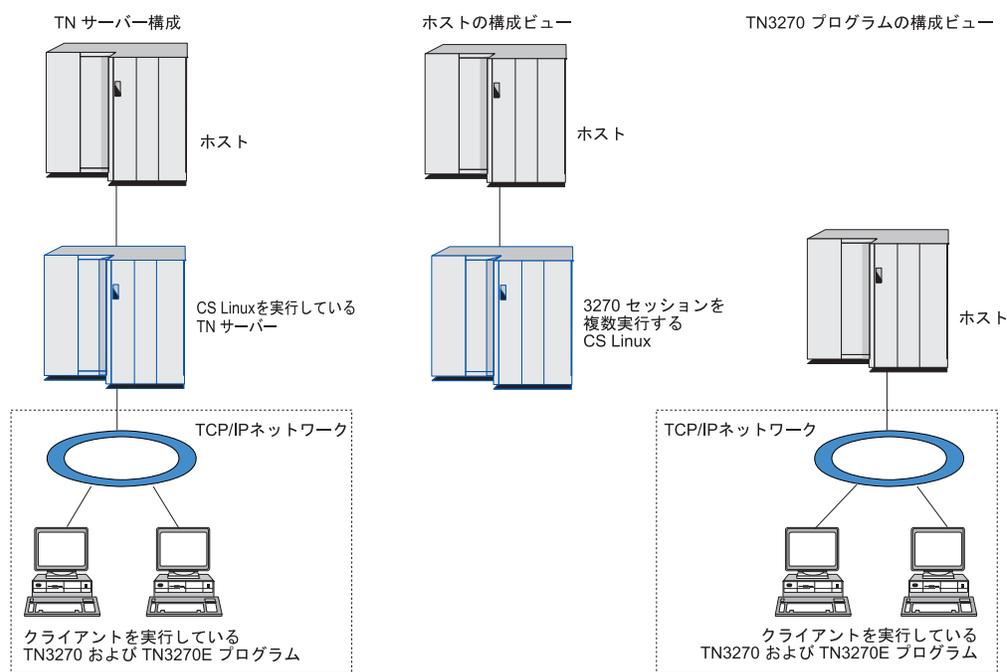


図 4. TN サーバー

Communications Server for Linux TN サーバー機能は、TN3270 ユーザーと Communications Server for Linux 3270 LU との関連付けを行います。TN3270 ユーザーからのデータはすべて、関連付けられた LU へ転送されます。つまり、ホストと TN3270 ユーザーのどちらの構成でも、両者が直接接続しているように構成されています。したがってホストもユーザーも、データが TN サーバーを介して転送されることを認識する必要はありません。

Communications Server for Linux TN サーバーは、IETF RFC

1123、1576、1646、1647、2355 で定義されているプロトコルを正確にインプリメントしている、TN3270 クライアント・エミュレーション・プログラムをすべてサポートします。

セキュリティー機能: Communications Server for Linux TN サーバーは、Secure Sockets Layer (SSL) ソフトウェアを使用して、データの暗号化、サーバー認証、クライアント認証、および高速ログオンをサポートします。

- データの暗号化とは、TN サーバーと TN3270 エミュレーター間のデータ・フローが暗号化された形式になっていることです。
- サーバー認証により、TN3270 クライアントは、接続されている TN サーバーが適切であるかどうか検証できます。
- クライアント認証により、TN サーバーは、接続されている TN3270 クライアントが適切であるかどうか検証できます。TN サーバーは、クライアントの許可が取り消されていないか確認するため、外部ディレクトリー・サーバーの取り消しリストも検査できます。
- 高速ログオンはクライアント認証と連動するため、TN3270 クライアントがホストに接続する際にユーザー ID とパスワードを入力する必要がありません。その代わりに、クライアントのセキュリティー証明書が使用されて、必要なユーザー ID とパスワード情報が検索されます。

TN リダイレクター

Communications Server for Linux TN リダイレクター機能は、TCP/IP による 3270、5250 または VT セッションにパススルー・サービスを提供します。Telnet ユーザーは、TCP/IP 接続を介して Communications Server for Linux との通信を行います。Communications Server for Linux は次に、別の TCP/IP 接続を介してホストとの通信を行います。

Communications Server for Linux TN リダイレクターは、3270 用の TN サーバーの場合と同じように、Secure Sockets Layer (SSL) ソフトウェアを使用して、データの暗号化、サーバー認証およびクライアント認証をサポートします。これにより、ユーザーからホストへの接続完了時でなくても、必要な場合に、Secure Sockets Layer (SSL) セキュリティー検査を実施できます。たとえば、以下のとおりです。

- クライアントが、検査を必要としない TCP/IP LAN を経由して Communications Server for Linux に接続しており、しかも SSL が必要なリモート・ホストに接続している場合、Communications Server for Linux とホスト間の TCP/IP を介して SSL を使用できます。つまり、すべてのクライアントに対するセキュリティーを一度検査すれば、個々のクライアントがセキュリティー情報を提供する必要はないということです。
- Communications Server for Linux がホストと同じサイトにインストールされていても、クライアントが外部サイトから接続している場合、ホストに SSL ソフトウェアをインストールしなくても、Communications Server for Linux にクライアント接続すれば SSL を使用できます。

機能および利点

Communications Server for Linux の機能および利点は、構成の単純化から問題診断の改善、ネットワーク・パフォーマンスの強化まで多岐にわたっています。

多用途のビルディング・ブロック

Communications Server for Linux は、ほとんどの環境およびノード機能をサポートします。サブエリアまたは APPN など、どのタイプのネットワークであっても、Linux コンピューターは、以下のいずれかのノードあるいはノードの組み合わせとして機能できます。

- ホスト従属ノード
- 対等ノード (APPN 対等ノードについては、1 ページの『IBM Communications Server for Linux』の対等ネットワークの説明を参照してください)
- 分散アプリケーションにおけるパートナー (送信元または宛先)
- SNA ネットワークを相互接続するゲートウェイ・ノード

Linux コンピューターは、ネットワーク管理 API を介して、管理サービス (MS) エントリー・ポイントとして機能し、分散ネットワーク管理をサポートするように構成できます。リンク・レベルでは、Linux コンピューターは、サポートされる任意のリンク・タイプ (1 ページの『IBM Communications Server for Linux』および 22 ページの『インストール要件』で説明) を使用して、各種 LAN および WAN に接続できます。

クライアント/サーバー操作

クライアント/サーバー構成により、次のような利点が得られます。

- SNA リソースをサーバーに集中させることによりクライアントの負荷が軽減するため、クライアントのパフォーマンスが向上すると同時に、SNA サービスをクライアントに提供するために必要なストレージが最小化します。
- 単一のデータ・リンクを異なるマシン上の複数のユーザーで共用できるため、各マシンに物理 SNA ネットワーク接続を確立する必要がありません。
- 複数のサーバーにより冗長接続が提供されます (たとえば、複数のサーバーが同じホストへのアクセスを提供します)。SNA リソースへの複数のパスにより、異なるサーバー間でロード・バランシングが行われ、特定のサーバーまたはリンクに障害が発生した場合に即時のバックアップが提供されます。
- 複数のサーバー間で LU プールを使用することにより、管理者は簡単にサーバーとユーザーを構成および追加できます。
- ホスト接続用のリンクおよび PU の数が少ないため、ホスト VTAM[®] 定義のサイズが削減されます。
- 管理ユーティリティを使用して、(ドメイン内のすべてのコンピューターの) ノード・リソースおよび共用リソースを構成および管理できます。Communications Server for Linux 管理ツールにより提供されるクライアント/サーバー・サポートを使用すると、すべてのドメイン・リソースをドメイン内のどのコンピューターからも透過的に管理できます。

- SNA アプリケーションをインターネット・プロトコル経由で接続可能です。この接続には、ファイアウォールの横断、および認証とセキュリティーのために TCP/IP と HTTPS を使用します。

容易な構成

Communications Server for Linux は、構成に要する時間およびネットワークの複雑性を低減するための構成オプションおよび機能を備えて設計されています。たとえば、以下のとおりです。

Motif 管理プログラム

Communications Server for Linux 構成を定義および変更する場合、Motif 管理プログラム (`xснаadmin`) を使用方法が最も簡単です。このプログラムのグラフィカル・ユーザー・インターフェースにより、Communications Server for Linux リソースを表示および管理できます。このプログラムは、通常、他のインストールと異なる値を持つインストールのフィールドのみを表示し、その他のフィールドにはデフォルト値を使用して、構成を単純化します。

Motif 管理プログラムには、ヘルプ画面があります。この画面では、SNA および Communications Server for Linux に関する概要、Communications Server for Linux ダイアログの参照情報、特定の操作の説明が表示されます。

APPN ネットワークの動的構成

ノードまたはネットワークの構成も、APPN ネットワークの動的構成を使用すると簡単になります。たとえば、APPN エンド・ノードおよびアプリケーションは、LU 6.2 セッションをサポートする構成データを動的に登録するので、セッション構成は必要なときのみ行えばよいようになります。さらに、ノード制御点をデフォルトのローカル LU として機能させると、LU 6.2 構成に手を煩わす必要がなくなります。

APPN も、構成されたリンク・ステーションがない場合、ダイナミック・リンク・ステーション構成をサポートします。

追加の管理用ユーザー・インターフェースの選択

Motif 管理プログラムは、Communications Server for Linux を構成および管理するために推奨されるインターフェースです。ただし、ユーザーは Communications Server for Linux のインターフェースを選択できます。これにより、装置、ニーズ、およびプリファレンスに合ったインターフェースで作業できるようになります。

コマンド行管理プログラム

コマンド行管理プログラム (`sнаadmin`) を使用して、個々の Communications Server for Linux リソースを管理するためのコマンドを発行できます。

`sнаadmin` は、Linux コマンド・プロンプトから直接使用することも、シェル・スクリプト内で使用することもできます。

NOF API

Communications Server for Linux の NOF API は、コマンド行管理プログラムと同じ機能を提供します。これは、(コマンド・スクリプトではなく) プロ

グラム内で使用するのに適したインターフェースを提供します。NOF API を使用して、Communications Server for Linux を管理するためのアプリケーション・プログラムを作成できます。

パフォーマンスの改善

Communications Server for Linux は、SNA ネットワークの本来のハイパフォーマンスを強化し、サービス・オペレーションのクラスを使用します。Communications Server for Linux は、以下のように、LU 0 から 3 セッション・データの SNA データ圧縮、またはネットワーク・サイズに従ってトラフィック・フローのバランスをとる各種トラフィック管理メソッドを使用して、ネットワーク速度を最適化します。

- APPN ネットワークでは、Communications Server for Linux は、高性能ルーティング (HPR) と中間セッション・ルーティング (ISR) の両方をサポートし、接続ネットワーク・オプションを提供します。ISR は、小規模ネットワークの場合は効率的に機能しますが、大規模ネットワークの場合はパフォーマンスが低下します。
- LAN 接続オプション (トークンリングまたはイーサネット) または Enterprise Extender を使用している大規模ネットワークの場合、接続ネットワーク・オプションを使用すれば、通信効率を上げることができます。接続ネットワーク・オプションは、ノード間に通信パスを直接作成します。これにより、トラフィックは中間ネットワーク・ノードをバイパスできるようになります。
- もう 1 つのトラフィック制御機構である適応セッション・レベル・ペーシングは、LU がメッセージ単位をパートナー LU に送信する速度を調節して、輻輳を自動的に調整します。

セキュリティー・オプション

ネットワークがさらに複雑化するにつれ、また、オープン・アーキテクチャーへ移行するにつれて、セキュリティーが重要な問題として取り上げられるようになってきました。Communications Server for Linux が稼働する SNA ネットワークでは、構成で各種レベルのセキュリティーを定義して、特定のタイプのリンクをインプリメントすることにより、お客様の資産を保護します。たとえば、以下のとおりです。

- クライアント/サーバー・システムでは、WebSphere サーバーをセットアップして、Remote API Client からサーバーへの HTTPS アクセスを可能にします。この意味は、クライアント接続がセキュリティー証明書を使って認証されることを示します。(このフィーチャーには、標準 Communications Server for Linux 製品のほかに追加のソフトウェアが必要です。詳しくは、22 ページの『インストール要件』を参照してください。)
- LU 6.2 ユーザーは、3 つまでのレベルのセキュリティー (セッション、リソース、会話) を定義できます。具体的には、セッション・レベル・セキュリティーは、セッションで適切な LU を確実に使用できるようにし、リソース・レベル・セキュリティーは、特定の LU に関連するすべてのアプリケーションへのアクセスを制限し、会話レベル・セキュリティーは、特定のアプリケーションへのアクセスを制限します。追加のセキュリティーは、データ暗号化ルーチンによって可能になります。

- Communications Server for Linux TN サーバーおよび TN リダイレクターは、Secure Sockets Layer (SSL) ソフトウェアを使用して、Communications Server for Linux と TN3270 または Telnet クライアントの間で、データの暗号化、サーバー認証およびクライアント認証をサポートすることができます。(このフィーチャーには、標準 Communications Server for Linux 製品のほかに追加のソフトウェアが必要です。詳しくは、22 ページの『インストール要件』を参照してください。)

ネットワーク管理の柔軟性

Communications Server for Linux はマルチドメイン・サポート・ネットワーク管理ベクトル・トランスポート (MDS-NMVT) ネットワーク管理方式をサポートします。これは、集中、分散、または階層管理体系として機能します。これは、フォーカル・ポイント / エントリー・ポイント・アーキテクチャーに基づいて高度な柔軟性を提供するものです。

フォーカル・ポイントは、エントリー・ポイント (ネットワーク内の他のすべてのノードにある管理アプリケーション) から収集するデータに基づいて、ネットワークを管理する制御ノードです。

- 集中管理では、単一フォーカル・ポイントが、ネットワーク全体の制御点として機能します。
- 分散管理では、複数フォーカル・ポイントがネットワーク管理を共有します。
- 階層管理では、フォーカル・ポイントは機能に従ってネスト構造を取ります。

MDS-NMVT は、こうして、サブエリア、標準 APPN、および超大規模ネットワークを管理するように適合させることができます。

信頼性、可用性、および保守容易性

信頼性のあるシステム操作を保守しやすくするように、Communications Server for Linux は、表示機能から問題診断ツールまでの一連の機能を提供します。

- Motif 管理プログラムは、以下のように、優れた構成および管理ツールを提供します。
 - 構成情報の即時更新
 - リンク、セッション、およびノード・リソースの状況情報
- query コマンドと status コマンドは、以下の情報を提供します。
 - LU-LU セッション
 - APPN 中間セッション
 - アクティブ・リンク
 - APPN トポロジー・データベース。これは、リンク情報を保管します。
- 問題診断ツールは、構成および操作のさまざまな段階で役に立ちます。これらのツールには、以下のものがあります。
 - 診断情報収集ツール (snagetpd)。サービス情報を容易に収集できます。
 - 状況およびエラー・メッセージ。構成ルーチンおよびシステム操作の問題解決に役立ちます。
 - ログ。ネットワークのエラー、障害、および監査情報を収集します。

- トレース機能。詳細な問題情報を収集し、形式設定します。

その他のユーティリティーは、アプリケーション間のリンク接続および通信のテストに役立ちます。

Communications Server for Linux には、管理サービス API があります。これは、ネットワーク・アラートや問題データの送受信のためのツールを開発します。

これらのすべての管理ツールおよび問題診断ツールが Communications Server for Linux クライアント/サーバー・モデルに完全に統合されているため、ネットワーク内の単一ポイントから、Communications Server for Linux ドメイン全体の管理、または診断情報の収集ができます。

ネットワークの統合、拡張および変更

ネットワークの統合、拡張、および変更をサポートするため、Communications Server for Linux API を使用して、ビジネス・ニーズに合った特定の LU、プラットフォーム、またはオペレーティング・システム用のアプリケーションを開発できます。CPI-C は、異種のプラットフォームおよびオペレーティング・システムを通じて一貫性があるため、特に重要な API です。これを使用して、任意のシステムで実行可能なアプリケーションを開発できます。

Enterprise Extender も、SNA と TCP/IP ネットワークを統合するための機構を提供します。

第 2 章 ご使用のネットワークと Communications Server for Linux on System z の計画

この章では、Communications Server for Linux が稼働するネットワーク計画の段階の概要を説明します。また、Linux コンピューターについて構成できる機能を要約して、機能をサポートするために必要なリソースの見積りのガイドラインも示しています。

ネットワーク計画の段階

このセクションでは、各種 Communications Server for Linux および Linux ユーティリティを使用して、ネットワークの計画、構成および管理を行う一般的なガイドラインを示します。

ネットワークの計画には、平衡化機能、パフォーマンス、リソースおよびコストが関係します。ネットワークに最適な計画を 1 つだけ示すことはできませんが、いくつかの一般ガイドラインおよび技法が、確実にニーズに合う計画を立てる際に役立ちます。ネットワークを計画するために、以下の作業を行います。

- ネットワークが提供する機能（ファイル転送や 3270 エミュレーションなど）およびパフォーマンス要件を決定します。
- 必要な機能を提供するための Communications Server for Linux の構成方法を決定します。
- Communications Server for Linux のインストール、パフォーマンスおよび容量要件のサポート、Communications Server for Linux 機能のサポートを行うために必要なリソースを見積もり、そのために要するコストを算出します。

ネットワークの機能要件の確認

ネットワークで提供する機能を判別するには、以下の問題点を考慮する必要があります。

- ネットワークは APPN である必要があるか。
- Communications Server for Linux はクライアント/サーバー・システムとして稼働するか。その場合、すべてのコンピューターが単一の Communications Server for Linux ドメイン内で稼働するのか、もしくは複数のドメインを個別に定義する必要があるのか。
- 接続リソースのロード・バランシングを実行するために、Communications Server for Linux ドメイン内に複数のサーバーが必要か。その場合、どのサーバーがマスター構成サーバーになるのか。1 つ以上のバックアップ構成サーバーが必要か。
- HTTPS を使って Communications Server for Linux に接続する Remote API Client をサポートする必要があるか。
- ユーザー・アプリケーションは、サーバーまたは Linux クライアント・コンピューターのどちらで実行されるか。

ネットワーク計画の段階

- サーバーは、Windows クライアント上で実行される Windows アプリケーション (たとえば、API トランザクション・プログラムなど) に対して接続リソースを提供するのか。
- サーバーをセッションのエンドポイントにするか、または以下のタイプのゲートウェイのいずれかにするか。
 - APPN
 - LU 0
 - LU 2
 - TN サーバーまたは TN リダイレクター
- ネットワークが使用する物理リンクのタイプ

上記の質問の回答は、ネットワークに必要な Communications Server for Linux の機能を決定するのに役立ちます。

Communications Server for Linux の構成方法の決定

Communications Server for Linux の機能を決定するには、まず、ネットワークでの作業フローを決定する必要があります。考えられる質問は以下のとおりです。

- ネットワークを介して使用する必要のあるリソース (アプリケーション) は何か
- リモート・リソースにアクセスする必要のあるユーザー数
- 各リソースへのアクセス頻度
- ユーザーのネットワークへのアクセス取得方法
- ユーザー要求のネットワークにおける経路指定方法

Communications Server for Linux は、たとえば、以下のノードなど、多数の機能をサポートするように構成できます。

- 中間セッション・ルーティング (ISR) の APPN ネットワーク・ノード
- APPN エンド・ノード (隣接ノードとは自発的に通信しますが、隣接していない対等ノードとは、APPN ネットワーク・ノード・サービスを使用して通信します。)
- ローエントリー・ネットワーキング (LEN) ノード (隣接ノードまたは隣接していると認識されるノードと直接に通信します。)
- LU 0、LU 1、LU 2、LU 3 および LU 6.2 (従属および独立) の使用
- ホストへの SNA ゲートウェイ接続

ユーザーのニーズに応じて、指定されたノードには 1 つ以上の機能を構成できます。たとえば、Communications Server for Linux を APPN ネットワーク・ノードとして構成し、ルーティング・サービスおよび ISR を提供したり、SNA ゲートウェイに対して同じノードを使用して LU 0 および LU 2 などの従属 LU セッションの経路を指定できます。同様に、TN サーバーを稼働するように Communications Server for Linux を構成し、共用データベース、およびホストの MQSeries® への独立 LU 6.2 接続をサポートすることができます。

インストールおよび操作のリソース要件の確認

Communications Server for Linux の機能のサポートを見積もるには、以下の問題の回答が必要です。

- 必要な個人スキルは何か
- どのトランスポート・メディアを使用したいか
- 選択する構成のインストール要件は何か
- 操作に必要なメモリーおよびページング・スペース量はどのくらいか

19 ページの『ネットワークの機能要件の確認』で説明している多くの機能のうち、1 つ以上をサポートするように Communications Server for Linux を構成する場合、上記の質問の回答は使用するリソース・タイプの確認に役立ちます。質問の回答は、Communications Server for Linux 機能、Linux リソース、およびネットワーク・リソースの関係を理解するのにも役立ちます。

ノードに対するリソースの割り振り方法により、ネットワークの実現方法が決定されます。

要員の要件

Communications Server for Linux のインストール、操作、および調整を行うには、以下の担当者が必要です。

- ネットワーク管理者。ネットワークの計画、新しい装置の追加、ネットワーク全体のパフォーマンスの保守および拡張を行います。
- システム管理者。Communications Server for Linux およびこの Linux を実行するハードウェアのインストールおよび保守、ネットワーク接続のシステム構成を行います。
- プログラマー。トランザクション・プログラムやネットワーク管理ルーチンなど、カスタマイズされたアプリケーションを開発します。

ネットワーク管理者およびシステム管理者は、Communications Server for Linux を実行するハードウェアと Linux オペレーティング・システムに十分に精通していなければなりません。各種システムが接続されるネットワークについての知識があり、一般的な SNA の概念を理解していなければなりません。また、以下のことにも精通している必要があります。

- Motif インターフェース
- rpm。Linux のインストール・ツール
- TCP/IP (クライアント/サーバー機能、TN サーバー または Enterprise Extender の使用を計画している場合)
- Windows 2000、Windows XP、または Windows Server 2003 オペレーティング・システム (Communications Server for Linux システムに Windows 上の Remote API Client が組み込まれている場合)
- WebSphere Application Server (Communications Server for Linux システムに HTTPS を使用するサーバーに接続する Remote API Client が組み込まれている場合)

SNA 用にカスタマイズされたアプリケーションを開発するプログラマーは、C 言語 (または Java CPI-C を使用する場合は Java) の経験者で、Communications Server for Linux で使用可能な API に精通している必要があります。

トランスポート・メディア

Communications Server for Linux は、基盤のトランスポート・メディア (SDLC、トークンリングなど) を、他の通信プロトコルと共用する必要がある場合があります。したがって、物理層の帯域幅要件は、トランスポート・メディアを共用する、すべてのプロトコルおよびアプリケーションを収容できなければなりません。

注: Communications Server for Linux では、トークンリングおよびイーサネットのアダプターを TCP/IP などの他のプロトコルと共用できます。使用するプロトコルごとに、固有のサービス・アクセス・ポイント (SAP) アドレスを指定する必要があります。

インストール要件

Communications Server for Linux に (19 ページの『ネットワークの機能要件の確認』から) 割り当てる機能によってもインストール要件が決定づけられます。このセクションでは、Communications Server for Linux のインストールに必要なコンピューター・リソースの概要について説明します。詳しくは、製品ごとに同梱されている資料を参照してください。

ハードウェア

Communications Server for Linux には、以下の Linux ディストリビューションのいずれかがサポートするコンピューターが必要です。

コマンド `uname -m` を使用して、ご使用のターゲット・コンピューターの CPU クラスをチェックします。下表には、各サーバー・タイプごとの適切なハードウェアと、このハードウェアに対する `uname -m` からの応答を記載してあります。

サーバー・タイプ	ハードウェア	uname 応答
System z	31 ビット S/390® または zSeries®	s390
System z	64 ビット zSeries または System z9™	s390x

Linux オペレーティング・システム

Communications Server for Linux は、以下のタイプの各種 Linux をサポートします。各タイプのバージョン番号およびカーネル・ビルドに関する最新情報は、Communications Server for Linux インストール・イメージの **README** ファイルを参照してください。

- RedHat Enterprise Linux 3 for S/390 (RHEL3-s390)
- RedHat Enterprise Linux 3 for zSeries (RHEL3-s390x)
- RedHat Enterprise Linux 4 for S/390 (RHEL4-s390)
- RedHat Enterprise Linux 4 for zSeries (RHEL4-s390x)
- SUSE Linux Enterprise Server 8 for IBM Mainframe (SLES8-s390*)
- SUSE Linux Enterprise Server 9 for IBM Mainframe (SLES9-s390*)
- SUSE Linux Enterprise Server 10 for IBM Mainframe (SLES10-s390x)

リンク・ハードウェア

リンク・ハードウェアはサーバーでのみ必要です。クライアントでは必要ありません。

Communications Server for Linux を使用可能なインターフェースは次のとおりです。すなわち、Enterprise Extender を使用する TCP/IP インターフェース経由、Linux for System z のマルチパス・チャンネル・デバイス・ドライバ (Communications Server for Linux on System z のみ) を使用する仮想マルチパス・チャンネル (MPC) インターフェース経由、Linux **lcs** デバイス・ドライバ (OSE **chpid** モードで OSA2 が必要) を使用する 802.2 イーサネットまたはトークンリング OSA 接続経由、または layer2/VSwitch サポートを持った Linux **qeth** デバイス・ドライバを使用する 802.2 イーサネット OSA 接続経由です。

追加ソフトウェア: Linux

Communications Server for Linux には、以下の追加ソフトウェアが必要です。具体的なバージョン要件 (ご使用の各種 Linux タイプにより異なります)、およびこれらのソフトウェア・パッケージのインストール手順については、Communications Server for Linux インストール・イメージの **README** ファイルを参照してください。

- Linux Streams (LiS).
- OpenMotif (サーバーのみで必要です。クライアントでは必要ありません)。これは、Motif 管理プログラムを使用するために必要です。Communications Server for Linux の構成および管理を行うには、この方法が推奨されます。
- Java (Java CPI-C を使用する場合に必要です)。Java ランタイム環境 (JRE) が必要になります。新しい Java クラスをコンパイルして Java CPI-C アプリケーションとして使用する場合は、Java SDK も必要になります。
- Linux for System z MultiPath Channel デバイス・ドライバ (Communications Server for Linux on System z でのみ必要 (サーバー上でのみ必要で、クライアント上には不要))。このコンポーネントは、MPC 接続を使用して VM/VTAM システムに接続する場合に必要です。)

WebSphere Application Server (HTTPS アクセス用)

HTTPS を使用して Remote API Client が Communications Server for Linux サーバーに接続するクライアント/サーバー・システムを実行しようとしている場合、WebSphere Application Server を稼働してこれらのクライアントからサーバーへの HTTPS アクセスを行う必要があります。

Communications Server for Linux は WebSphere Application Server バージョン 5 と一緒に稼働します。この製品は、WebSphere によりサポートされるすべてのオペレーティング・システムで稼働するコンピューター上にインストール可能です。(必要な場合、Communications Server for Linux サーバーと同じ Linux コンピューター上にインストール可能です。)この製品のインストール方法の詳細は、WebSphere Application Server 資料を参照してください。また、追加の Communications Server for Linux プラグインをこのコンピューター上にインストールして、Communications Server for Linux と一緒に WebSphere を使用する必要があります。これは、31 ページの『WebSphere Application Server の構成』に記載したとおりです。

メモリーおよびストレージ

構成とサービスの全範囲をサポートするには、Communications Server for Linux に、Linux ディストリビューションが必要とする最小メモリーに加え

ネットワーク計画の段階

て 32 MB のメモリー、および 120 MB のディスク・スペースが必要です。また、インストール時には 250 MB の一時ストレージが必要となります。

Communications Server for Linux の資料をソフトコピー (PDF) 形式でインストールする場合は、ハード・ディスク・ストレージがさらに必要となります。すべてのソフトコピー・ブックをインストールするには、75 MB のディスク・スペースが必要です。

注: その他のライセンス・プログラム、ユーザー・アプリケーション、データを使用するためのメモリーおよびハード・ディスクの要件は、上記に含まれていません。システム、メモリー、およびハード・ディスクの、すべての要件については、IBM 担当員または業界の IBM 認定販売店と共に慎重に検討してください。

命名規則

ネットワーク ID を使用すると、物理ネットワークを論理的にセグメント化することができます。また、他のネットワークに接続する計画がある場合は、ネットワーク名の競合を避けるためにも、ネットワーク ID を登録することを強くお勧めします。

ネットワーク名および LU 名は、次のように定義できます。

ネットワーク名

それぞれ異なるネットワーク名 (ネットワーク ID) を定義することによって、APPN ネットワークのセグメンテーションを行うことができます。セグメンテーションにより、ネットワーク・トポロジー・データベースのサイズ、および個々のネットワークを流れるブロードキャスト LOCATE 要求の頻度を規制できます。

ネットワーク ID の固有性を保証するために、ネットワーク管理者は、IBM のワールド・ワイド・レジストリーを使用して、ネットワーク ID を登録することができます。IBM レジストリーは、そこに登録されているネットワーク ID が、それぞれ固有の ID であることを保証します。登録規格は、国際標準化機構 (ISO) が確立した、オープン・システム間相互接続 (OSI) の規格 (OSI 国別コードを含む) に準拠しています。登録に関する詳細は、

「*User's Guide for SNA Network Registry*」を参照してください。

LU 名 LU 名にワイルドカードを使用すると、システム定義とネットワーク検索を最小限にとどめることができます。

第 3 章 Communications Server for Linux の Linux サーバーへのインストール

この章では、Communications Server for Linux を Linux サーバーにインストールするために必要な手順について簡単に説明します。インストール・プロセスの段階的な手順については、Communications Server for Linux インストール・イメージの **README** ファイルを参照してください。ご使用の各種 Linux タイプに適した最新の詳細情報が記載されています。**README** ファイルには以下の情報が含まれています。

- 必要な追加ソフトウェア・パッケージの正確なパッケージ名およびまたはバージョン番号
- インストール・コマンドおよびセットアップ・コマンドの詳細

Communications Server for Linux ライセンス・プログラムのパッケージ内容

Communications Server for Linux ライセンス・プログラムは 3 枚の CD イメージとして提供され、以下のものが収容されます。

CD #1: クイック・スタート

この CD には Communications Server for Linux の PDF 資料一式 (ディレクトリー **/DOCS**) が入っています。

CD #2: サーバー

この CD には、サーバーをインストールするのに必要な各ファイルが入っています。

- **README** ファイル。Communications Server for Linux 資料の編成後に行われた製品の変更に関する情報が収録されています。
- インストール・スクリプト
- このサーバー用のイメージおよび PDF マニュアル用のイメージをインストールします。

CD #3: クライアント

この CD には、以下のような、各タイプのクライアントのインストールに必要な **README** ファイル、インストール・スクリプト、およびインストール・イメージが入っています。

- 32 ビット Intel (i686) 用 Linux クライアント
- 64 ビット AMD64/Intel EM64T (x86_64) 用 Linux クライアント
- pSeries (ppc64) 用 Linux クライアント
- System z9 または System z (s390 / s390x) 用 Linux クライアント
- AIX クライアント
- 32 ビット Windows クライアント
- 64 ビット Windows クライアント

CD イメージに収容されているファイルの詳細については、**README** ファイルを参照してください。

注: 必要なストレージの容量については、22 ページの『インストール要件』を参照してください。

Communications Server for Linux インストールの準備

前提ソフトウェアのインストール

Communications Server for Linux をインストールする前に、22 ページの『インストール要件』にリストされている前提ソフトウェアをインストールする必要があります。

- LiS Streams
- OpenMotif
- Java
- Linux for System z MultiPath Channel デバイス・ドライバー (Communications Server for Linux on System z のみ。MPC 接続を使用して VM/VTAM システムと接続する場合にのみ必要です)。

必要なソフトウェア・パッケージの詳細およびその段階的なインストール手順については、Communications Server for Linux インストール・イメージの **README** ファイルを参照してください。

製品インストール詳細の表示

既にインストールされている Communications Server for Linux および関連ソフトウェア・パッケージに関する情報を表示できます。すべてのインストール済みパッケージをリストするには、次のコマンドを使用します。

```
rpm -q -a
```

特定のパッケージの詳細を表示するには、次のコマンドを使用します。

```
rpm -q -i packagename
```

packagename は、インストール済みパッケージのベース・ネームです。たとえば、**ibm-commserver** です。

言語環境変数の変更

Communications Server for Linux を使用する場合、使用する言語を示す LANG 変数が正確に設定されていることを確認してください。

LANG 変数を変更するには、次のコマンドを使用します。

```
export LANG=language
```

language を、使用する言語の ID に置き換えてください。言語は以下から、1 つが使用できます。

ID	言語
en_US	英語 (US)
ja_JP	日本語 (PC)
de_DE	ドイツ語
es_ES	スペイン語
fr_FR	フランス語
ko_KR	韓国語
pt_BR	ポルトガル語
zh_CN	中国語 (簡体字 EUC)
zh_TW	中国語 (繁体字)

旧レベルの Communications Server for Linux からのマイグレーション

Communications Server for Linux の旧バージョンが既にインストールされており、バージョン 6.2.2 にマイグレーションする場合、次の手順に従ってください。

カスタマイズされたすべての構成ファイルの保管

下記にリストするファイルが存在する場合、一時ディレクトリーに保管してください。すべてのシステムに以下のすべてのファイルが存在するわけではありません。

```

/etc/opt/ibm/sna/sna_node.cfg
/etc/opt/ibm/sna/sna_domn.cfg
/etc/opt/sna/sna.net
/etc/opt/ibm/sna/sna_tps
/etc/opt/ibm/sna/ibmcs.kdb
/etc/opt/ibm/sna/ibmcs.sth
/etc/opt/ibm/sna/ibmcs.rdb
/etc/opt/ibm/sna/ibmcs.crl

```

さらに、74 ページの『Communications Server for Linux の使用可能にする方法』に記載したようにスタートアップ・ファイル `/etc/rc.d/init.d/snastart` をカスタマイズして `sna start` コマンドを削除し、それにより Communications Server for Linux がシステム始動時に自動開始しない場合、このファイルに対して行った変更内容を書き留めてください。

旧リリースのアンインストール

次のコマンドを使用して、Communications Server for Linux の停止およびアンインストールを実行します。どの旧バージョンがインストールされているか、およびそのインストール方法によっては、下記のすべての RPM パッケージが必ずしもご使用のシステム上に存在するとは限りません。

```

sna stop
rpm -e ibm-conmserver-ptf
rpm -e ibm-commserver-docs
rpm -e ibm-commserver-ecl
rpm -e ibm-commserver
rpm -e gsk6bas

```

LiS のアンインストール

次のコマンドを使用して、現行レベルの LiS オープン・ソース・パッケージをアンインストールします。

旧レベルの **Communications Server for Linux** からのマイグレーション

```
PATH=$PATH:/sbin
unset LD_PRELOAD
rmmod streams
cd /usr/src/LiS
make uninstall
make very-clean
cd /usr/src
rm -rf LiS*
```

PATH および他の環境変数の変更

Communications Server for Linux の旧バージョンに対して以下の環境変数のいずれかを変更していた場合、行ったこの変更を除去する必要があります。その理由は、これらのパスは Communications Server for Linux バージョン 6.2.2 プログラム・プロダクトでは異なっている可能性があるからです。

```
PATH
LD_LIBRARY_PATH
LD_RUN_PATH
LD_PRELOAD
CLASSPATH
```

env コマンドを使用して、すべての環境変数をチェックして **sna** への参照がないか調べても構いません。

```
env | grep sna
```

その他のパッケージ

ご使用の Java システムを確認して、必要な場合は最新レベルに更新することをお勧めします。

最終クリーンアップ

次のコマンドにより、製品の旧レベルの残りの全項目が除去されます。

```
rm -rf /etc/opt/ibm/sna /var/opt/ibm/sna /opt/ibm/sna
```

新規レベルの **Communications Server for Linux** バージョン 6.2.2 プログラム・プロダクトのインストール

本書および **README** ファイルの手順に従って、製品をインストールします。

保管構成の復元

このプロセスの最初に構成ファイルを保管した場合は、ここで復元します。まず最初に、Communications Server for Linux ソフトウェアを次のコマンドで停止します。

```
/opt/ibm/sna/bin/sna stop
```

このプロセスでの最初のステップですべての **ibmcs.*** ファイルを保管済みの場合、ここで、**/etc/opt/ibm/sna** ディレクトリーからすべての **ibmcs.*** ファイルを削除します。たとえば、**ibmcs.kdb** と **ibmcs.sth** を保管済みの場合、**ibmcs.crl** と **ibmcs.rdb** ファイルを除去する必要があります。この作業は、これらのファイルを置き換えるための保管されたファイルがない場合でも除去します。重要なことは、保管済みファイルと新規ファイルが混在した状態で実行しないことです。

旧レベルの Communications Server for Linux からのマイグレーション

保管済みのファイルを `/etc/opt/ibm/sna` ディレクトリーに復元します。

さらに、スタートアップ・ファイル `/etc/rc.d/init.d/snastart` への変更を保管済みの場合、74 ページの『Communications Server for Linux の使用可能にする方法』に記載したようにこのスタートアップ・ファイルの新規コピーに対して同様の変更を行って、Communications Server for Linux が確実にシステム始動時に自動開始しないようにします。

ここで、Communications Server for Linux ソフトウェアを次のコマンドで再始動します。

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

Communications Server for Linux のライセンス・プログラムのインストール

Communications Server for Linux のインストール

旧レベルの Communications Server for Linux が既にインストールされている場合は、この新規レベルをインストールする前に、27 ページの『旧レベルの Communications Server for Linux からのマイグレーション』の手順に従って旧レベルを除去してください。

Communications Server for Linux をインストールするには、次の手順に従います。

1. **ibm-commserver-6.2.0.0-s390.tgz** ファイルを CD-ROM から Linux System z システムにコピーまたは FTP でファイル転送する。ファイルをコピーまたは FTP でファイル転送する際は、必ずバイナリー・モードを使用してください。
2. Linux System z システムにルートとしてログインします。
3. `tar` ファイルを空の一時ディレクトリーに解凍する。

```
mkdir /tmp/ibmcs
```

```
cd /tmp/ibmcs
```

```
zcat ibm-commserver-6.2.2.0-s390.tgz | tar -xf -
```

4. `installibmcs` シェル・スクリプトを実行する。

```
./installibmcs
```

このシェル・スクリプトは一定の前提条件を検査し、条件が満たされない場合は警告メッセージを出します。また、ユーザーに対して以下に関するプロンプトが出されます。プロンプトに応答した後、シェル・スクリプトは **rpm** パッケージをインストールします。

- Communications Server for Linux ライセンス条件を読み、受け入れることの確認。
- Communications Server for Linux ドメイン内でマスター・サーバーとなるサーバーの名前。Communications Server for Linux をスタンドアロン・ノードとして実行する場合は、このパラメーターを指定しないでください。その場合、このノードはクライアント/サーバー機能をサポートしません。

Communications Server for Linux ライセンス・プログラムのインストール

これらのプロンプトは、以下に説明されるように **installibmcs** コマンドで追加パラメーターを指定することによりオーバーライドできます。

シェル・スクリプトで Communications Server for Linux が正常にインストールされないようなエラーが発生した場合、標準出力 (通常、画面) にエラー・メッセージが書き込まれます。このようなエラーの解決に関するヘルプは、Communications Server for Linux インストール・イメージの **README** ファイルを参照してください。

メモリーが限られているシステムの場合、Communications Server for Linux をインストールしてから、SNA ノードを開始する前にリブートする必要があります。大容量システムの場合、この操作は必要ありません。Communications Server for Linux ノードが開始できない場合、**/var/log/messages** ファイルで次のような項目を確認してください。

kernel: SNA Trace Driver can only get X blocks of memory — please reboot

これらのメッセージがリブートした後も出される場合、メモリーを増やす必要があります。

5. Communications Server for Linux バイナリー・ディレクトリーを **PATH** に追加する。このプロセスを自動化する場合は、次の方法でプロファイルを変更します。

```
export PATH="$PATH:/opt/ibm/sna/bin:/opt/ibm/sna/bin/X11"
```

```
export LD_LIBRARY_PATH=/usr/lib:/opt/ibm/sna/lib
```

```
export LD_RUN_PATH=/usr/lib:/opt/ibm/sna/lib
```

Java CPI-C アプリケーションの場合、次の環境変数も設定する必要があります。

```
export CLASSPATH=$CLASSPATH:/opt/ibm/sna/java/cpic.jar
```

一部のアプリケーションでは、**LD_PRELOAD** 環境変数も変更する必要があります。ただし、これをプロファイルで一括変更してはなりません。

```
export LD_PRELOAD=/usr/lib/libpLiS.so
```

6. Communications Server for Linux を始動する。インストール後は、このプロセスがマシンのリポート時に自動的に行われることにご注意ください。

```
cd /
```

```
sna start
```

7. Communications Server for Linux Motif 管理プログラムを実行する。Communications Server for Linux の操作に慣れるまで、このプログラムを使用することをお勧めします。

Linux System z システムには X Window システムのクライアント機能だけが組み込まれているため、リモートの X Window システムのサーバーを使用する必要があります。X Window システム・サーバーで、次のコマンドを使用します。

Communications Server for Linux ライセンス・プログラムのインストール

```
xhost +XXXX
```

XXXX は、Linux System z システムの TCP/IP 名またはアドレスです。

ここで、xsnaadmin クライアントにサーバーの場所を指示して、開始します。

```
export DISPLAY=YYYY:Z
```

```
xsnaadmin &
```

YYYY は X Window システム・サーバーの TCP/IP 名またはアドレスであり、Z は仮想表示番号 (通常は 0) です。

無人インストールを行う場合、**installibmcs** コマンドで追加パラメーターを指定して、Communications Server for Linux ライセンス条件の受け入れを確認して、マスター・サーバーの名前を指定します。この場合、シェル・スクリプトはそれ以上の情報についてプロンプトを出さずに実行されます。次のコマンドを使用します。

```
./installibmcs license_accepted [ master_name ]
```

master_name は、マスター・サーバーの名前です。Communications Server for Linux をスタンドアロン・ノードとして実行する場合は、このパラメーターを指定しないでください。その場合、このノードはクライアント/サーバー機能をサポートしません。

Communications Server for Linux オンライン資料

29 ページの『Communications Server for Linux のインストール』の順に従って、**tgz** ファイルを解凍してから、**installibmcsdocs** シェル・スクリプトを実行します。

```
./installibmcsdocs
```

Host Access Class Library (HACL)

HACL ファイルは、Communications Server for Linux のインストール時に自動的にインストールされます。このライブラリー・コードは、**rpm** 下の **ibm-commserver-ecl** パッケージにあります。これらのファイル (README を含む) は **/opt/ibm/sna/ecl** にあります。あるいは、以下のコマンドを発行して見つけることができます。

```
rpm -ql ibm-commserver-ecl
```

WebSphere Application Server の構成

HTTPS を使用して Remote API Client が Communications Server for Linux サーバーに接続するクライアント/サーバー・システムを実行しようとしている場合、WebSphere Application Server を稼働するコンピューターが必要です。これは、22 ページの『インストール要件』に記載したとおり、これらのクライアントからサーバーへの HTTPS アクセスを行うためです。

このセクションでは、Communications Server for Linux と一緒に使用するための WebSphere のセットアップ方法を以下に記載してあります。

WebSphere Application Server の構成

- WebSphere 上でクライアントに提示されるセキュア証明書のセットアップ
- Communications Server for Linux と一緒に稼働する WebSphere Application Server の構成
- WebSphere サーバー上でサーバー構成ファイルのインストール

また、クライアントのセキュリティー証明書およびネットワーク・データ・ファイルを各 Remote API Client 上にセットアップして、WebSphere Application Server にアクセスする必要もあります。詳細は、該当のクライアント・タイプのインストールに関する章を参照してください。

WebSphere Application Server のセキュア証明書のセットアップ

このサーバー上でのセキュア証明書のセットアップ方法は、WebSphere Application Server 資料を参照してください。これは、サーバーの証明書であり、HTTPS を使って接続しようとする時点で認証プロセス中に Remote API Client に提示されることとなります。

お勧めすることは、WebSphere を構成して、それによりクライアント認証を強制することです。この詳細は、WebSphere Application Server 資料を参照してください。この意味は、認証プロセス中に WebSphere が Remote API Client からセキュリティー証明書を要求し、Remote API Client からの着信接続を受け入れるのはクライアント証明書の信憑性を WebSphere が検証可能な場合に限るということです。

WebSphere Application Server の構成

WebSphere Application Server を構成して Communications Server for Linux と一緒に稼働するには、次の手順に従います。詳細は、WebSphere Application Server 資料を参照してください。

1. 2 つのファイル (**snahttpsrv.ear** と **snahttpsrv.cfg**) を、Remote API Client のインストール CD 上の **ibm-commserver-https** ディレクトリーから、WebSphere 管理コンソールが稼働するコンピューター上のディレクトリーに、またはこのコンピューターからアクセス可能なネットワーク・ディレクトリーにコピーまたは FTP します。

この管理コンソールが Windows 上で稼働している場合、これらのファイルのコピーは不要です。その理由は、これらのファイルを CD から直接アクセス可能だからです。単に、Remote API Client のインストール CD を Windows コンピューターの CD ドライブに挿入する必要があるだけです。

2. WebSphere 管理コンソールを始動します。
3. WebSphere 資料に従って、SSL のセキュア接続経由でのみアクセス可能な仮想ホストを作成します。この仮想ホストは、SNA HTTPS 接続を管理する Java プラグイン用に使用することになります。
4. メニュー・バーで、「アプリケーション (Applications)」、「新規アプリケーションのインストール (Install New Application)」を選択します。
5. **snahttpsrv.ear** ファイルの場所を指定します。「次へ (Next)」を選択します。
6. 最初の 2 つの画面で仮想ホスト名を指定するようにプロンプトが出された場合、HTTPS 用にセットアップ済みの仮想ホスト名を入力します。いずれか特定

の WebSphere 構成の使用が必要となる場合を除き、その他のすべてのパラメーターに対してデフォルト・オプションを受け入れることができます。「次へ (Next)」ボタンが「終了 (Finish)」ボタンに変わるまで「次へ (Next)」ボタンを後続ダイアログ上で選択してゆき、「終了 (Finish)」ボタンを選択します。次に、画面上に「アプリケーションが正常にインストールされました (Application installed successfully)」が表示されることになります。

7. 「マスター構成への保管 (Save to Master Configuration)」をクリックしてから、「保管 (Save)」ボタンをクリックします。
8. メニュー・バーで、「アプリケーション (Applications)」、「エンタープライズ・アプリケーション (Enterprise Applications)」を選択します。
9. アプリケーション・リストの中で「**SnaHttpTransport**」を見つけて、その横にあるチェック・ボックスをクリックし、「開始 (Start)」ボタンを押してこのアプリケーションを開始します。(この後では、このアプリケーションは WebSphere Application Server の開始時に自動開始します。)
10. メニュー・バーで、「環境 (Environment)」、「Web サーバー・プラグインの更新 (Update Web Server Plugin)」を選択してから、「OK」ボタンをクリックします。これにより、WebSphere 構成が更新されます。

サーバー構成ファイルのインストール

WebSphere Application Server が、Communications Server for Linux と一緒に稼働するには、HTTPS を使ってアクセスされる Communications Server for Linux リストが必要です。以下のステップを使用してこのリストの作成とインストールを行います。

1. 「WebSphere 管理コンソール (WebSphere administration console)」メニュー・バーで、「環境 (Environment)」、「WebSphere 変数の管理 (Manage WebSphere Variables)」を選択します。
2. このリストで **USER_INSTALL_ROOT** 変数を探し、その値 (これは WebSphere サーバー上のディレクトリーのパスです) をメモします。環境変数のリストは 2 ページ以上にわたって表示される場合があり、「次へ (Next)」ボタンを使用してこのリスト全体をスクロールする必要性が生じる可能性があります。
3. 32 ページの『WebSphere Application Server の構成』で **snahttpsrv.cfg** ファイルを保管済みの場所 (または、インストール CD から) からこのファイルをコピーして、**USER_INSTALL_ROOT** 変数で指定したディレクトリーに入れます。次に、テキスト・エディターを使ってこのファイルを編集して、HTTPS を使用する Remote API Client がアクセス可能な Communications Server for Linux サーバー・リストに組み込みます。各サーバーはこのファイルの独立した行に指定する必要があります。この形式は以下のとおりです。

```
server=servername.domainname.com
```

インストール後の作業

この項では、Communications Server for Linux をインストールしたあとに必要な保守作業の実行方法について説明します。

クライアント/サーバー操作

インストール後、Communications Server for Linux は最初にスタンドアロンのサーバーとして稼働します (単一の Linux システム上の全コンポーネントを持つ)。クライアント/サーバー・ドメイン内のサーバーとして稼働させたい場合、その説明は「*Communications Server for Linux 管理ガイド*」の『Communications Server for Linux クライアント/サーバー・システムの管理』に関する章を参照してください。

インストール後のクリーンアップ

インストールが完了すると、インストール・プロセス中に作成された **tgz** ファイルおよび一時ディレクトリーを削除できます。

PDF 資料の表示

この製品のインストール・メディアには、PDF 形式のマニュアルが収録されています。ソフトコピー形式では、関連情報へのハイパーテキスト・リンクを使用して情報を簡単に検索または表示できます。また、PDF ビューアーが多くの各種プラットフォームで使用できるので、システム内でのライブラリーの共有も簡単に行うことができます。

製品のインストール時に PDF マニュアルのインストールを選択すると、それらのマニュアルはディレクトリー **/opt/ibm/sna/docs** にインストールされます。また、マニュアルは Communications Server for Linux インストール・メディアのディレクトリー **/DOCS**、および **tgz** ファイルにも収容されています。

PDF マニュアルは、PDF ビューアーを使用して読むことができます。たとえば、Adobe Acrobat を使用して Windows で、または **xpdf** を使用して Intel Linux で読むことができます。

現行リリース情報の検討

/opt/ibm/sna ディレクトリーに入っている本製品の最新の **README** ファイルには、Communications Server for Linux ライブラリー資料の編成後に行われた製品の変更に関する情報が収録されています。このファイルは、Communications Server for Linux インストール・メディアのルート・ディレクトリー、および **tgz** ファイルにも収容されています。製品の更新を受け取ったら、必ず **README** ファイルを調べてください。

TN サーバーまたは TN リダイレクターで使用するための SSL の構成

SSL 機能をもつ TN サーバーまたは TN リダイレクターを使用する場合は、Communications Server for Linux をインストールした後に SSL ソフトウェアを構成する必要があります。

SSL ソフトウェアには以下の 2 つのコンポーネントが必要です。

- データ暗号化および暗号化解除を実行できるようにするには、鍵ペアが必要です。
- サーバー認証を許可するには、証明書が必要です。

証明書および鍵ペアは、鍵リング・データベース内に単一のレコードを作成します。このレコードは、TN サーバーまたは TN リダイレクターを実行している Communications Server for Linux サーバーに保管されます。Communications Server for Linux はこのデータベースを使用して、SSL をインプリメントします。

鍵リング・データベースを管理するには、OpenMotif をインストールする必要があります。OpenMotif のインストール手順については、Communications Server for Linux インストール・イメージの **README** ファイルを参照してください。

鍵リング・データベースを管理するには、Linux コマンド・プロンプトに以下のコマンドを入力します。

snakeyman

snakeyman コマンドは、Java プログラムを立ち上げます。詳しい指示については、このプログラムと共に提供されるヘルプを参照してください。

データベース内の各レコードは、ラベルと呼ばれる固有名で識別されます。異なる TN サーバー・セッションまたは TN リダイレクター・セッションで使用するレコードが複数ある場合には、データベースをセットアップするときに、割り当てるラベルをメモしておく必要があります。これらのラベルは、各セッションで使用されるレコードを識別するために使用します。レコードの 1 つをデフォルトに決定することもできます。その結果、別のレコードのラベルを明示的に指定しない限り、セッションではこのレコードが使用されます。

snakeyman を使用してサーバー証明書の更新後、**snakeyman** プログラムを終了してから、Communications Server for Linux ノードの停止と再始動を行って更新された証明書を使用する必要があります。以下のコマンドを使用してノードの停止と再始動を行います。

```
snaadmin term_node
```

```
snaadmin init_node
```

Communications Server for Linux 構成ファイルのバックアップ

Communications Server for Linux では、Communications Server for Linux 管理ツールを使用してノード、ドメイン、および TP の構成ファイルに影響する変更を行うと、必ずこれらのファイルのバックアップが自動的に作成されます。たとえば、ノード構成ファイル (**sna_node.cfg**) に影響のある変更を行うと、Communications Server for Linux により **sna_node.bk n** という名前のバックアップ・ファイルが作成されます。この *n* は、1 または 2 のどちらかです。

- 初めてファイルを変更すると、既存の構成は **sna_node.bk1** に保管されます。
- 2 度目にファイルを変更すると、既存の構成は **sna_node.bk2** に保管され、**sna_node.bk1** は変更されません。
- 3 度目以降にファイルを変更すると、**sna_node.bk1** は破棄され、**sna_node.bk2** は **sna_node.bk1** に名前が変更されて、既存の構成が **sna_node.bk2** に保管されます。

インストール後の作業

このプロセスは、ノード構成ファイルのバックアップ・ファイルは、常に、最大でも 2 つしかないことを意味します。他のバックアップ・ファイルのファイル名拡張子も、同じプロセスで生成されます。

次の状況では、自動バックアップの他に、データの消失を防ぐために構成ファイルのバックアップを各自作成しておく必要があります。

- 新しいレベルの Linux オペレーティング・システムをインストールする前
- 新しいリリースの Communications Server for Linux をインストールする前
- 新しい構成を作成したあと

以下のコマンドを使用して、構成ファイルをバックアップします。

```
cd /etc/opt/ibm/sna
tar cvf Devicename sna_node.cfg sna.net sna_tps sna_domn.cfg
ibmcs.*
```

Communications Server for Linux 構成ファイルのバックアップ・コピーの復元

35 ページの『Communications Server for Linux 構成ファイルのバックアップ』に説明されている方法でバックアップされた Communications Server for Linux 構成ファイルを復元するには、次の手順に従います。

1. Communications Server for Linux がアクティブでないことを確認する。アクティブであるかどうかを確認するには、次のコマンドを入力します。

```
snaadmin status_node
```

Communications Server for Linux がアクティブである場合には、このコマンドは、ローカル・ノードの状態に関する情報を表示します。そうでない場合には、Communications Server for Linux がアクティブでないことを示すメッセージを表示します。

Communications Server for Linux がアクティブである場合は、次のコマンドを入力して活動停止します。

```
sna stop
```

2. 以下のコマンドを入力する。

```
cd /etc/opt/ibm/sna
tar xvf Devicename
```

上記のコマンドの *Devicename* は、ファイルのバックアップで使用したデバイスのパスとファイル名です。

このコマンドは、*/etc/opt/ibm/sna* ディレクトリーにある同じ名前の既存の構成ファイルを上書きします。

構成ファイルの再初期化

Communications Server for Linux 構成ファイルが誤って変更されたために、ファイル内の情報を使用できなくなった場合には、構成ファイルを再初期化する必要があります。

ります。構成ファイルを再初期化すると、Communications Server for Linux を新たにインストールしたかのように再構成できます。この操作は、構成情報が修復できない場合にのみ行ってください。

注: 有効なバックアップ構成ファイルがある場合は、まずこれらのバックアップ・ファイルを `/etc/opt/ibm/sna` ディレクトリーにコピーし、`sna start` コマンドを使用してノードを初期化する際にこれらのファイルを使用します。

再初期化できる構成ファイルを次に示します。

- ノード構成ファイル `sna_node.cfg`
- ドメイン構成ファイル `sna_domn.cfg`
- TP 構成ファイル `sna_tps`
- SSL 鍵リング・データベース・ファイルおよびパスワード・スタッシュ・ファイル

構成ファイルを再初期化する手順は次のとおりです。

1. 管理プログラムがアクティブである場合は管理プログラムを終了し、次のコマンドを発行して Communications Server for Linux を使用不可にする。

sna stop

2. 既存の構成ファイルのバックアップを作成するため、再初期化するファイルをすべて別の位置へコピーする。
3. 再初期化するファイルを削除する。
4. ドメイン構成ファイルを削除した場合は、以下のコマンドを発行して、その構成ファイルを再作成する (Communications Server for Linux に同梱された空のドメイン構成ファイルからコピーして)。

```
cp -p /opt/ibm/sna/samples/empty.cfg /etc/opt/ibm/sna/sna_domn.cfg
```

上記のコマンドにより、Communications Server for Linux の始動に必要な新しいドメイン構成ファイルが作成されます。

5. SSL 鍵リング・データベース・ファイルを削除した場合は、以下のコマンドを発行して、そのデータベース・ファイルを再作成する (Communications Server for Linux とともに送達されたサンプル・ファイルからコピーして)。

```
cp -p /opt/ibm/sna/samples/ibmcs.* /etc/opt/ibm/sna
```

6. 次のコマンドを入力して、Communications Server for Linux を再始動する。

sna start

7. Motif 管理プログラムを開始する。

xsnaadmin &

`sna_node.cfg` ファイルが存在しない場合、ノードを構成するように管理プログラムからプロンプトが指示されます。ノードおよびその他のリソースの構成を続けます。詳細については、71 ページの『第 8 章 Communications Server for Linux の構成と使用』または「*Communications Server for Linux* 管理者用ガイド」を参照してください。

有効な `sna_node.cfg` ファイルを使用した場合は、ノードを初期化するとき新しい構成ファイルが使用されます。

Communications Server for Linux のアンインストール

Communications Server for Linux 製品はいつでもアンインストールできます。次の手順に従ってください。

1. root 権限でログインする。
2. Communications Server for Linux がアクティブでないことを確認する。アクティブであるかどうかを確認するには、次のコマンドを入力します。

snaadmin status_node

Communications Server for Linux がアクティブである場合には、このコマンドは、ローカル・ノードの状態に関する情報を表示します。そうでない場合には、Communications Server for Linux がアクティブでないことを示すメッセージを表示します。

Communications Server for Linux がアクティブである場合は、次のコマンドを入力して活動停止します。

sna stop

3. 以下の手順を使用して、Communications Server for Linux パッケージおよび関連するソフトウェア・パッケージを除去する。

```
rpm -e ibm-commsserver-ptf
```

```
rpm -e ibm-commsserver-docs
```

```
rpm -e ibm-commsserver-ecl
```

```
rpm -e ibm-commsserver
```

```
/sbin/shutdown -r now
```

第 4 章 IBM Remote API Client on Linux のインストール

本章では IBM Remote API Client on Linux のインストール方法を記載してあります。これにより、Linux ワークステーションで完全な SNA スタックのインストール環境を保有せずに SNA アプリケーションを実行可能となります。Remote API Client on Linux は 1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー、ただし両方同時は不可) と TCP/IP ネットワークを介して接続可能です。(CS Linux サーバーは、CS/AIX サーバーと同じドメイン内で稼働できません。)

本章の記述が適用対象とするコンピューターは、IBM Remote API Client が 32 ビット Intel (i686)、64 ビット AMD64/Intel EM64T (x86_64)、および pSeries (ppc64) 上で稼働する場合です。System z コンピューター (s390 / s390x) 上に IBM Remote API Client をインストールする場合は、45 ページの『第 5 章 System z に対する IBM Remote API Client on Linux のインストール』を参照してください。

インストール・プログラムおよび関連ファイル (IBM Remote API Client README ファイルを含む) は、インストール CD に収録されており、以下のご使用のクライアント・タイプごとに該当のライブラリーが異なります。

クライアント・タイプ CD 上のディレクトリー名

32 ビット Intel (i686)	/ibm-commserver-clients/linux
64 ビット AMD64/ Intel EM64T (x86_64)	/ibm-commserver-clients/linux-x86_64
pSeries (ppc64)	/ibm-commserver-clients/linux-ppc64

このソフトウェアのインストール前に IBM Remote API Client README ファイルを読んでください。

ハードウェアおよびソフトウェア要件

ハードウェア要件

IBM Remote API Client には、以下の Linux ディストリビューションのいずれかがサポートするコンピューターが必要です。

コマンド `uname -m` を使用して、ご使用のターゲット・コンピューターの CPU クラスをチェックします。下表には、各クライアント・タイプごとの適切なハードウェアと、このハードウェアに対する `uname -m` からの応答を記載してあります。

クライアント・タイプ	ハードウェア	uname 応答
32 ビット Intel	Pentium® II またはそれ以降の 32 ビット Intel システム、または Opteron ベースのシステム	i686

クライアント・タイプ	ハードウェア	uname 応答
64 ビット AMD64/Intel EM64T	x86_64 (AMD64 または Intel EM64T) システム	x86_64
pSeries	pSeries POWER5™ または OpenPower™ システム	ppc64

Linux オペレーティング・システムのバージョン

現行バージョンの IBM Remote API Client は、以下の Linux オペレーティング・システム・バージョンでテスト済みです。このソフトウェアは、他の Linux ディストリビューション上でも正常に稼働します。

- RedHat Enterprise Linux 3 (RHEL3): i686 のみ
- RedHat Enterprise Linux 4 (RHEL4)
- SUSE Linux Enterprise Server 8 (SLES8): i686 のみ
- SUSE Linux Enterprise Server 9 (SLES9)
- SUSE Linux Enterprise Server 10 (SLES10)

どのオプション・パッケージが必要とされる可能性があるかの詳細は、インストール CD に収録された **README** ファイルを参照してください。

Java

Java CPI-C API を使用する場合、Java ソフトウェアが必要になります。詳細は、インストール CD 上の **README** ファイルを参照してください。

GSKIT

HTTPS を使用して Communications Server for Linux サーバーと接続する場合、WebSphere サーバー経由で HTTPS がこのサーバーにアクセス可能となるために GSKIT ソフトウェアが必要となります。この GSKIT ソフトウェアはインストール CD 上に収録されていますが、一部のオプションの Linux オペレーティング・システム・パッケージが GSKIT ソフトウェアのインストールに必要です。どのオプション・パッケージが必要とされる可能性があるかの詳細は、インストール CD に収録された **README** ファイルを参照してください。

クライアントのインストール・プロセス実行時に前提条件となる全パッケージをインストールすると、GSKIT ソフトウェアがこのプロセスの一部として自動的にインストールされます。そうでない場合は、GSKIT ソフトウェアを後でインストールしても構いません。

製品インストール詳細の表示

既にインストールされている Remote API Client および関連ソフトウェア・パッケージに関する情報を表示できます。すべてのインストール済みパッケージをリストするには、次のコマンドを使用します。

```
rpm -q -a
```

特定のパッケージの詳細を表示するには、次のコマンドを使用します。

```
rpm -q -i packagename
```

packagename は、インストール済みパッケージのベース・ネームです。たとえば、**ibm-commsserver-client** です。

言語環境変数の設定

LANG 変数を変更するには、次のコマンドを使用します。これにより使用したい言語を指定します。

```
export LANG=language
```

language を、使用する言語の ID に置き換えてください。言語は以下から、1 つが使用できます。

ID	言語
en_US	英語 (US)
ja_JP	日本語 (PC)
de_DE	ドイツ語
es_ES	スペイン語
fr_FR	フランス語
ko_KR	韓国語
pt_BR	ポルトガル語
zh_CN	中国語 (簡体字 EUC)
zh_TW	中国語 (繁体字)

Remote API Client on Linux のインストール

前提ソフトウェアをインストールした後で、IBM Remote API Client をインストールできる状態となります。

旧レベルの IBM Remote API Client が既にインストールされている場合は、この新規レベルをインストールする前に、44 ページの『Remote API Client on Linux のアンインストール』のセクションの手順に従って旧レベルを除去してください。すべての構成情報は、新規インストールで使用するために残ったままの状態になります。

1. root 権限でログインする。
2. CD をマウントして現行ディレクトリーにする。

```
mount /dev/cdrom
cd /media/cdrom
```

RHEL3 (32 ビット Intel クライアントのみ) の場合、**/media/cdrom** を **/mnt/cdrom** で置き換えてください。そうでない場合は、DVD ドライブ使用時はディレクトリー名 **/media/cdrom** が異なっている可能性があります。コマンド **df** を使用して、Linux がその CD をどこにマウントしたかを表示します。

3. この CD 上の適切なサブディレクトリーに変わり、シェル・スクリプトを実行してこのクライアントをインストールします。以下の例では、32 ビット Intel (i686) クライアント用の **/linux** サブディレクトリーを示しています。必要に応じて、これを **/linux-x86_64** または **/linux-ppc64** に置き換えます。

Remote API Client on Linux のインストール

```
cd ibm-commserver-clients/linux
./installibmcscli
```

このシェル・スクリプトは一定の前提条件を検査し、条件が満たされない場合は警告メッセージを出します。ご使用条件を読んで受諾するようにプロンプトが出されます。その後、この RPM をスクリプトがインストールします。適切な前提条件となるソフトウェアがすでにインストール済みの場合、このスクリプトは GSKIT ソフトウェアもインストールします。

4. IBM Remote API Client バイナリー・ディレクトリーを PATH に追加します。このプロセスを自動化する場合は、次の方法でプロファイルを変更します。

```
export PATH="$PATH:/opt/ibm/sna/bin"
```

```
export LD_LIBRARY_PATH=/usr/lib:/opt/ibm/sna/lib
```

```
export LD_RUN_PATH=/usr/lib:/opt/ibm/sna/lib
```

Java CPI-C アプリケーションの場合、次の環境変数も設定する必要があります。

```
export CLASSPATH=$CLASSPATH:/opt/ibm/sna/java/cpic.jar
```

一部のアプリケーションでは、LD_PRELOAD 環境変数も変更する必要があります。ただし、これをプロファイルで一括変更してはなりません。

```
export LD_PRELOAD=/usr/lib/libpLiS.so
```

5. IBM Remote API Client を始動します。インストール後は、このプロセスがマシンのリブート時に自動的に行われます。これを行う時点で CD のディレクトリーになっていないことを確認してください。

```
cd /
sna start
```

注: HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティー証明書をセットアップする必要があります。詳しくは、『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、このクライアントが接続対象とする Communications Server for Linux サーバー、および HTTPS をサポートする WebSphere サーバー名を指定する必要があります。詳細は、「*Communications Server for Linux 管理ガイド*」にある Remote API Client の管理に関するセクションを参照してください。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

HTTPS を使用して Communications Server for Linux サーバーと接続する場合、GSKIT 鍵管理機能ソフトウェアをインストールしておく必要があります。これは、Linux オペレーティング・システムの前条件がインストール CD 上の README ファイルに記載されたとおりインストールされているという条件で、通常クライアント・インストールの一部として行われます。GSKIT がクライアント・インスト

GSKIT を使用した HTTPS セキュリティ証明書セットアップ

ールの一部としてインストール済みではなく、ここでこの前提条件のソフトウェアをインストールする場合、以下のステップで GSKIT をインストールできます。

1. root 権限でログインする。
2. CD をマウントして現行ディレクトリーにする。

```
mount /dev/cdrom
```

```
cd /media/cdrom
```

RHEL3 (32 ビット Intel クライアントのみ) の場合、**/media/cdrom** を **/mnt/cdrom** で置き換えてください。そうでない場合は、DVD ドライブ使用時はディレクトリー名 **/media/cdrom** が異なっている可能性があります。コマンド **df** を使用して、Linux がその CD をどこにマウントしたかを表示します。

3. この CD 上の適切なサブディレクトリーに変わり、シェル・スクリプトを実行してこの GSKIT ソフトウェアをインストールします。以下の例では、32 ビット Intel (i686) クライアント用の **/linux** サブディレクトリーを示しています。必要に応じて、これを **/linux-x86_64** または **/linux-ppc64** に置き換えます。

```
cd ibm-commserver-clients/linux
```

```
./installgskit
```

HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティ証明書をセットアップする必要があります。以下のステップで行います。

1. 以下のコマンドで GSKIT 鍵管理機能を実行します。

```
/opt/ibm/sna/bin/snakeyman
```

この鍵管理機能のユーザー・インターフェース内から鍵データベース・ファイル (**/etc/opt/ibm/sna/ibmcs.kdb**) をオープンします。このファイルは CMS 形式です。

2. この鍵データベース用の初期パスワードは **ibmcs** です。セキュリティ証明書のセットアップ前に、このパスワードを変更して、ご使用の構成をセキュア状態に保つ必要があります。パスワード変更用ダイアログ上で、「ファイルに対するパスワードを隠しておく? (Stash the password to a file?)」チェック・ボックスにマークを付ける必要があります。これにより、新規のパスワードが保管されて、クライアントがこの鍵データベースをオープンできるようにします。
3. Web サーバーのセキュリティ証明書を署名するのに使用された Certificate Authority (CA) 証明書のコピーを入手して、それを鍵データベースにインストールします。これを行うには、鍵管理機能のユーザー・インターフェースで「署名者証明書 (Signer Certificates)」を選択して「追加 (Add)」をクリックします。
4. WebSphere サーバーがクライアントのセキュリティ証明書を必要とするように構成されている場合、そのクライアントは CA 発行の証明書を保有する必要があります。この CA 所有の証明書は Web サーバーのセキュリティ証明書データベースにあります。新規の証明書が必要な場合は以下のようにします。
 - a. 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規認証要求 (New Certificate Request)」を選択して、要求された詳細内容を入力します。
 - b. 証明書を保管し、それをファイルに抜き出し、CA に送信します。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

- c. この証明書を発行する場合、それを Web サーバーのデータベースに保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。

お客様自身の内部的なテストのための一時的な手段として、CA から証明書入手する代わりに自己署名のクライアント証明書を作成できます。ただし、この手段では必要なレベルのセキュリティーを提供しないため、本格稼働システムでは使用しないでください。自己署名証明書を作成するには以下を行います。

- a. 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規自己署名証明書 (New Self-Signed Certificate)」を選択して、要求された詳細内容を入力します。
 - b. この証明書を保管し、それをファイルに抜き出します。
 - c. Web サーバーのデータベースに証明書ファイルを保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。
5. 証明書の構成が完了したら、GSKIT 鍵管理機能を終了します。

Remote API Client on Linux のアンインストール

Remote API Client on Linux をアンインストールするには、以下のコマンドを使用します。

```
/opt/ibm/sna/bin/sna stop  
rpm -e ibm-commsserver-ptf  
rpm -e ibm-commsserver-docs  
rpm -e ibm-commsserver-ecl  
rpm -e ibm-commsserver-cli  
rpm -e ibm-commsserver  
rpm -e gsk7bas  
/sbin/shutdown -r now
```

上記コマンドにリストされたパッケージすべてが、必ずしもすべてのシステムにインストールされているとは限りません。

IBM Remote API Client on Linux をアンインストールしても、すべてのカスタマイズされた構成情報がその後のインストールで使用する目的で残ったままになります。

第 5 章 System z に対する IBM Remote API Client on Linux のインストール

本章では IBM Remote API Client on Linux のインストール方法を記載してあります。これにより、System z メインフレームで完全な SNA スタックのインストール環境を保有せずに SNA アプリケーションを実行可能となります。System z 用の Remote API Client on Linux は 1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー) と TCP/IP ネットワークを介して接続可能です。

このソフトウェアのインストール前に IBM Remote API Client README ファイルを読んでください。このファイルは、インストール CD 上の `/ibm-commserver-clients/linux-systemz` ディレクトリにあります。

ハードウェアおよびソフトウェア要件

ハードウェア要件

IBM Remote API Client には、『Linux オペレーティング・システムのバージョン』にリストした Linux ディストリビューションのいずれかがサポートする 31 ビットまたは 64 ビットの System z システムが必要です。

コマンド `uname -m` を使用して、CPU クラスをチェックします。このコマンドで、31 ビット環境を示す `s390`、または 64 ビット環境を示す `s390x` がレポートされる必要があります。

Linux オペレーティング・システムのバージョン

現行バージョンの IBM Remote API Client は、以下の Linux オペレーティング・システム・バージョンでテスト済みです。このソフトウェアは、他の Linux ディストリビューション上でも正常に稼働します。

- RedHat Enterprise Linux 3 for S/390 (RHEL3-s390)
- RedHat Enterprise Linux 3 for zSeries (RHEL3-s390x)
- RedHat Enterprise Linux 4 for S/390 (RHEL4-s390)
- RedHat Enterprise Linux 4 for zSeries (RHEL4-s390x)
- SUSE Linux Enterprise Server 8 for IBM Mainframe (SLES8-s390*)
- SUSE Linux Enterprise Server 9 for IBM Mainframe (SLES9-s390*)
- SUSE Linux Enterprise Server 10 for IBM Mainframe (SLES10-s390x)

どのオプション・パッケージが必要とされる可能性があるかの詳細は、インストール CD に収録された **README** ファイルを参照してください。

Java

Java CPI-C API を使用する場合、Java ソフトウェアが必要になります。詳細は、インストール CD 上の **README** ファイルを参照してください。

GSKIT

HTTPS を使用して Communications Server for Linux サーバーと接続する場合、WebSphere サーバー経由で HTTPS がこのサーバーにアクセス可能となるために GSKIT ソフトウェアが必要となります。この GSKIT ソフトウェアはインストール CD 上に収録されていますが、一部のオプションの Linux オペレーティング・システム・パッケージが GSKIT ソフトウェアのインストールに必要です。どのオプション・パッケージが必要とされる可能性があるかの詳細は、インストール CD 上の `/ibm-commsrver-clients/linux-systemz` ディレクトリー収録された **README** ファイルを参照してください。

クライアントのインストール・プロセス実行時に前提条件となる全パッケージをインストールすると、GSKIT ソフトウェアがこのプロセスの一部として自動的にインストールされます。そうでない場合は、GSKIT ソフトウェアを後でインストールしても構いません。

製品インストール詳細の表示

既にインストールされている Remote API Client および関連ソフトウェア・パッケージに関する情報を表示できます。すべてのインストール済みパッケージをリストするには、次のコマンドを使用します。

```
rpm -q -a
```

特定のパッケージの詳細を表示するには、次のコマンドを使用します。

```
rpm -q -i packagename
```

packagename は、インストール済みパッケージのベース・ネームです。たとえば、**ibm-commsrver-client** です。

言語環境変数の設定

LANG 変数を変更するには、次のコマンドを使用します。これにより使用したい言語を指定します。

```
export LANG=language
```

language を、使用する言語の ID に置き換えてください。言語は以下から、1 つが使用できます。

ID	言語
en_US	英語 (US)
ja_JP	日本語 (PC)
de_DE	ドイツ語
es_ES	スペイン語
fr_FR	フランス語
ko_KR	韓国語
pt_BR	ポルトガル語
zh_CN	中国語 (簡体字 EUC)
zh_TW	中国語 (繁体字)

System z に対する Remote API Client on Linux のインストール

前提ソフトウェアをインストールした後で、IBM Remote API Client をインストールできる状態となります。

旧レベルの IBM Remote API Client が既にインストールされている場合は、この新規レベルをインストールする前に、50 ページの『System z 用の Remote API Client on Linux のアンインストール』のセクションの手順に従って旧レベルを除去してください。すべての構成情報は、新規インストールで使用するために残ったままの状態になります。

1. CD-ROM に収録された **ibm-commserver-client-6.3.0.1-s390.tgz** ファイルを **/ibm-commserver-clients/linux-systemz** ディレクトリーから Linux System z システムにコピーまたは FTP します。ファイルをコピーまたは FTP でファイル転送する際は、必ずバイナリー・モードを使用してください。
2. Linux System z システムにルートとしてログインします。
3. tar ファイルを空の一時ディレクトリーに解凍する。

```
mkdir /tmp/ibmcs
```

```
cd /tmp/ibmcs
```

```
tar -xzf ibm-commserver-client-6.3.0.1-s390.tgz
```

4. **installibmcscli** シェル・スクリプトを実行する。

```
./installibmcscli
```

このシェル・スクリプトは一定の前提条件を検査し、条件が満たされない場合は警告メッセージを出します。また、Communications Server for Linux ライセンス条件を読んで受け入れることを確認するプロンプトが出されます。このプロンプトは、以下に説明される方法により **installibmcscli** コマンドで追加パラメーターを指定してオーバーライドできます。プロンプトに回答した後、シェル・スクリプトは **rpm** パッケージをインストールします。適切な前提条件となるソフトウェアがすでにインストール済みの場合、このスクリプトは GSKIT ソフトウェアもインストールします。

5. IBM Remote API Client バイナリー・ディレクトリーを **PATH** に追加します。このプロセスを自動化する場合は、次の方法でプロファイルを変更します。

```
export PATH="$PATH:/opt/ibm/sna/bin"
```

```
export LD_LIBRARY_PATH=/usr/lib:/opt/ibm/sna/lib
```

```
export LD_RUN_PATH=/usr/lib:/opt/ibm/sna/lib
```

64 ビット・アプリケーションを実行予定の場合は、以下のコマンドを入力します。

```
export LD_LIBRARY_PATH=/usr/lib64:/opt/ibm/sna/lib64
```

```
export LD_RUN_PATH=/usr/lib64:/opt/ibm/sna/lib64
```

System z に対する Remote API Client on Linux のインストール

Java CPI-C アプリケーションの場合、次の環境変数も設定する必要があります。

```
export CLASSPATH=$CLASSPATH:/opt/ibm/sna/java/cpic.jar
```

一部のアプリケーションでは、LD_PRELOAD 環境変数も変更する必要があります。ただし、これをプロファイルで一括変更してはなりません。

```
export LD_PRELOAD=/usr/lib/libpLiS.so
```

6. IBM Remote API Client を始動します。インストール後は、このプロセスがマシンのリブート時に自動的に行われます。これを行う時点で CD のディレクトリになっていないことを確認してください。

```
cd /
```

```
sna start
```

7. インストールが完了すると、インストール・プロセス中に作成された **tgz** ファイルおよび一時ディレクトリを削除できます。

注: HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティー証明書をセットアップする必要があります。詳しくは、『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、このクライアントが接続対象とする Communications Server for Linux サーバー、および HTTPS をサポートする WebSphere サーバー名を指定する必要があります。詳細は、「*Communications Server for Linux 管理ガイド*」にある Remote API Client の管理に関するセクションを参照してください。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

HTTPS を使用して Communications Server for Linux サーバーと接続する場合、GSKIT 鍵管理機能ソフトウェアをインストールしておく必要となります。これは、Linux オペレーティング・システムの前提条件がインストール CD 上の **README** ファイルに記載されたとおりインストールされているという条件で、通常クライアント・インストールの一部として行われます。GSKIT がクライアント・インストールの一部としてインストール済みではなく、ここでこの前提条件のソフトウェアをインストールする場合、以下のステップで GSKIT をインストールできます。

1. CD-ROM に収録された **ibm-commserver-client-6.3.0.1-s390.tgz** ファイルを **/ibm-commserver-clients/linux-systemz** ディレクトリから Linux System z システムにコピーまたは FTP します。ファイルをコピーまたは FTP でファイル転送する際は、必ずバイナリー・モードを使用してください。
2. Linux System z システムにルートとしてログインします。
3. tar ファイルを空の一時ディレクトリに解凍します。

```
mkdir /tmp/ibmcs
```

```
cd /tmp/ibmcs
```

```
tar -xzf ibm-commserver-client-6.3.0.1-s390.tgz
```

GSKIT を使用した HTTPS セキュリティ証明書セットアップ

4. `installgskit` シェル・スクリプトを実行します。

`./installgskit`

5. インストールが完了すると、インストール・プロセス中に作成された `tgz` ファイルおよび一時ディレクトリーを削除できます。

HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティ証明書をセットアップする必要があります。以下のステップで行います。

1. 以下のコマンドで GSKIT 鍵管理機能を実行します。

`/opt/ibm/sna/bin/snakeyman`

この鍵管理機能のユーザー・インターフェース内から鍵データベース・ファイル (`/etc/opt/ibm/sna/ibmcs.kdb`) をオープンします。このファイルは CMS 形式です。

2. この鍵データベース用の初期パスワードは `ibmcs` です。セキュリティ証明書のセットアップ前に、このパスワードを変更して、ご使用の構成をセキュア状態に保つ必要があります。パスワード変更用ダイアログ上で、「ファイルに対するパスワードを隠しておく? (Stash the password to a file?)」チェック・ボックスにマークを付ける必要があります。これにより、新規のパスワードが保管されて、クライアントがこの鍵データベースをオープンできるようにします。
3. Web サーバーのセキュリティ証明書を署名するのに使用された Certificate Authority (CA) 証明書のコピーを入手して、それを鍵データベースにインストールします。これを行うには、鍵管理機能のユーザー・インターフェースで「署名者証明書 (Signer Certificates)」を選択して「追加 (Add)」をクリックします。
4. WebSphere サーバーがクライアントのセキュリティ証明書を必要とするように構成されている場合、そのクライアントは CA 発行の証明書を保有する必要があります。この CA 所有の証明書は Web サーバーのセキュリティ証明書データベースにあります。新規の証明書が必要な場合は以下のようにします。
 - a. 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規認証要求 (New Certificate Request)」を選択して、要求された詳細内容を入力します。
 - b. 証明書を保管し、それをファイルに抜き出し、CA に送信します。
 - c. この証明書を発行する場合、それを Web サーバーのデータベースに保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。

お客様自身の内部的なテストのための一時的な手段として、CA から証明書を手にする代わりに自己署名のクライアント証明書を作成できます。ただし、この手段では必要なレベルのセキュリティを提供しないため、本格稼働システムでは使用しないでください。自己署名証明書を作成するには以下を行います。

- a. 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規自己署名証明書 (New Self-Signed Certificate)」を選択して、要求された詳細内容を入力します。
- b. この証明書を保管し、それをファイルに抜き出します。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

- c. Web サーバーのデータベースに証明書ファイルを保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。
5. 証明書の構成が完了したら、GSKIT 鍵管理機能を終了します。

System z 用の Remote API Client on Linux のアンインストール

System z 用の Remote API Client on Linux をアンインストールするには、以下のコマンドを使用します。

```
/opt/ibm/sna/bin/sna stop  
rpm -e ibm-commsserver-ptf  
rpm -e ibm-commsserver-docs  
rpm -e ibm-commsserver-ecl  
rpm -e ibm-commsserver-cli  
rpm -e ibm-commsserver  
rpm -e gsk7bas  
/sbin/shutdown -r now
```

上記コマンドにリストされたパッケージのすべてが、必ずしもすべてのシステムにインストールされているとは限りません。

System z 用の IBM Remote API Client on Linux をアンインストールしても、すべてのカスタマイズされた構成情報がその後のインストールで使用する目的で残ったままになります。

第 6 章 IBM Remote API Client on AIX システムのインストール

本章では IBM Remote API Client on AIX のインストール方法を記載してあります。これにより、AIX ワークステーションで完全な SNA スタックのインストール環境を保有せずに SNA アプリケーションを実行可能となります。Remote API Client on AIX は 1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー) と TCP/IP ネットワークを介して接続可能です。

このソフトウェアのインストール前に IBM Remote API Client README ファイルを読んでください。このファイルは、インストール CD 上の `/ibm-commserver-clients/aix` ディレクトリーにあります。

ハードウェアおよびソフトウェア要件

ハードウェア要件

IBM Remote API Client には、『オペレーティング・システムのバージョン』にリストした AIX オペレーティング・システムのいずれかがサポートする pSeries システムが必要です。

オペレーティング・システムのバージョン

現行バージョンの IBM Remote API Client は、以下のオペレーティング・システム・バージョンでテスト済みです。

- AIX v5.2-ML5
- AIX v5.3-ML1

Java

Java CPI-C API を使用する場合、Java ソフトウェアが必要になります。

<http://www.ibm.com/developerworks/java/jdk> から入手可能な Java 1.4.2 SDK は、すべての要件を満足します。64 ビット Java もまた入手可能ですが、Java CPI-C API には 32 ビット JVM が必要です。

`installp` コマンドを使用して Java SDK パッケージをインストールします。

GSKIT

HTTPS を使用して Communications Server for Linux サーバーと接続する場合、WebSphere サーバー経由で HTTPS がこのサーバーにアクセス可能となるために GSKIT ソフトウェアが必要となります。詳細は、インストール CD 上の `/ibm-commserver-clients/aix` ディレクトリーにある **README** ファイルを参照してください。本章で後述するとおり、GSKIT ソフトウェアがメインのクライアント・インストール・プロセスの一部としてインストールされます。

言語環境変数の変更

Remote API Client を使用する場合、LANG 変数が C に設定されていないことを確認してください。

以下の手順に従って、どの LANG 変数が使用中になっているかを表示するか、または LANG 変数を変更します。

1. 「SMIT」メイン・メニューで「システム環境 (System Environments)」を選択します。
2. 「SMIT」の後続メニューで「言語環境の管理 (Manage Language Environment)」を選択します。
3. 「SMIT」の後続メニューで「主要言語環境の変更/表示 (Change/Show Primary Language Environment)」を選択します。
4. 「SMIT」の後続メニューで「国/地域別情報、言語、またはキーボードの変更/表示 (Change/Show Cultural Convention, Language, or Keyboard)」を選択します。
5. 使用したい言語を選択します。たとえば、U.S. 英語メッセージを使おうとする場合、「en_US」を選択します。

Remote API Client on AIX のインストール

前提ソフトウェアをインストールした後で、IBM Remote API Client をインストールできる状態となります。

旧レベルの IBM Remote API Client が既にインストールされている場合は、この新規レベルをインストールする前に、55 ページの『Remote API Client on AIX のインストール』のセクションの手順に従って旧レベルを除去してください。すべての構成情報は、新規インストールで使用するために残ったままの状態になります。

AIX ワークステーションにファイルをコピーして Remote API Client をインストールする方法

Remote API Client をインストールするには、次の手順に従います。

1. **sna.client.6.3.0.1.bff** ファイルを CD-ROM 上の **/ibm-commserver-clients/aix** ディレクトリーから AIX ワークステーションにコピーまたは FTP します。ファイルをコピーまたは FTP でファイル転送する際は、必ずバイナリー・モードを使用してください。

このクライアントが Communications Server for Linux サーバーに HTTPS を使って接続する場合、2 つのファイル (**gskta.*.I** と **gkska.*.I**) を CD 上のこの同じディレクトリーからコピーまたは FTP する必要があります。これらのファイルには、クライアントからの HTTPS アクセスに必要な GSKIT ソフトウェアが入っています。

2. AIX ワークステーションにルートとしてログインします。
3. **smit** または **installp** のいずれかを使用して AIX クライアントをインストールします。これを行う方法は、インストール CD 上の **/ibm-commserver-clients/aix** ディレクトリーにある **README** ファイルを参照してください。

4. HTTPS を使用して Communications Server for Linux サーバーと接続する場合、**README** ファイルにある手順に従って GSKIT ファイルをインストールします。
5. このインストール・プロセス完了後は、作業ディレクトリーから **sna.client.6.3.0.1.bff** ファイルと GSKIT ファイルを削除しても構いません。
6. IBM Remote API Client を始動します。インストール後は、このプロセスがマシンのリブート時に自動的に行われます。

```
cd /
sna start
```

注: HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティー証明書をセットアップする必要があります。詳しくは、54 ページの『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、このクライアントが接続対象とする Communications Server for Linux サーバー、および HTTPS をサポートする WebSphere サーバー名を指定する必要があります。詳細は、「*Communications Server for Linux 管理ガイド*」にある Remote API Client の管理に関するセクションを参照してください。

CD からの Remote API Client のインストール

Remote API Client をインストールするには、次の手順に従います。

1. AIX ワークステーションにルートとしてログインします。
2. 以下のコマンドを使用して AIX ワークステーション上で CD をマウントします。

```
mount -o ro /dev/cd0 /mnt
```

3. **smit** または **installp** のいずれかを使用して AIX クライアントをインストールします。これを行う方法は、インストール CD 上の **/ibm-commserver-clients/aix** ディレクトリーにある **README** ファイルを参照してください。
4. HTTPS を使用して Communications Server for Linux サーバーと接続する場合、**README** ファイルにある手順に従って GSKIT ファイルをインストールします。
5. このインストール・プロセス完了後に以下のコマンドを使用してその CD をアンマウントします。

```
umount /mnt
```

6. IBM Remote API Client を始動します。インストール後は、このプロセスがマシンのリブート時に自動的に行われます。これを行う時点で CD のディレクトリーになっていないことを確認してください。

```
cd /
sna start
```

注: HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティー証

明書をセットアップする必要があります。詳しくは、『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、このクライアントが接続対象とする Communications Server for Linux サーバー、および HTTPS をサポートする WebSphere サーバー名を指定する必要があります。詳細は、「*Communications Server for Linux 管理ガイド*」にある Remote API Client の管理に関するセクションを参照してください。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティ証明書をセットアップする必要があります。以下のステップで行います。

1. 以下のコマンドで GSKIT 鍵管理機能を実行します。

```
/usr/bin/snakeyman
```

この鍵管理機能のユーザー・インターフェース内から鍵データベース・ファイル (*/etc/sna/ibmcs.kdb*) をオープンします。このファイルは CMS 形式です。

2. この鍵データベース用の初期パスワードは *ibmcs* です。セキュリティ証明書のセットアップ前に、このパスワードを変更して、ご使用の構成をセキュア状態に保つ必要があります。パスワード変更用ダイアログ上で、「ファイルに対するパスワードを隠しておく? (Stash the password to a file?)」チェック・ボックスにマークを付ける必要があります。これにより、新規のパスワードが保管されて、クライアントがこの鍵データベースをオープンできるようにします。
3. Web サーバーのセキュリティ証明書を署名するのに使用された Certificate Authority (CA) 証明書のコピーを入手して、それを鍵データベースにインストールします。これを行うには、鍵管理機能のユーザー・インターフェースで「署名者証明書 (Signer Certificates)」を選択して「追加 (Add)」をクリックします。
4. WebSphere サーバーがクライアントのセキュリティ証明書を必要とするように構成されている場合、そのクライアントは CA 発行の証明書を保有する必要があります。この CA 所有の証明書は Web サーバーのセキュリティ証明書データベースにあります。新規の証明書が必要な場合は以下のようにします。
 - a. 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規認証要求 (New Certificate Request)」を選択して、要求された詳細内容を入力します。
 - b. 証明書を保管し、それをファイルに抜き出し、CA に送信します。
 - c. この証明書を発行する場合、それを Web サーバーのデータベースに保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。

お客様自身の内部的なテストのための一時的な手段として、CA から証明書を入手する代わりに自己署名のクライアント証明書を作成できます。ただし、この手段では必要なレベルのセキュリティを提供しないため、本格稼働システムでは使用しないでください。自己署名証明書を作成するには以下を行います。

GSKIT を使用した HTTPS セキュリティ証明書のセットアップ

- a. 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規自己署名証明書 (New Self-Signed Certificate)」を選択して、要求された詳細内容を入力します。
 - b. この証明書を保管し、それをファイルに抜き出します。
 - c. Web サーバーのデータベースに証明書ファイルを保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。
5. 証明書の構成が完了したら、GSKIT 鍵管理機能を終了します。

Remote API Client on AIX のアンインストール

以下のコマンドを使用して、Remote API Client をアンインストールできます。

1. クライアント・ソフトウェアが稼働中の場合は以下のコマンドで停止します。

sna stop

2. root 権限でログインします。
3. 以下のコマンドのいずれかを使用して、Remote API Client パッケージおよび関連するソフトウェア・パッケージを除去します。

installp を使用してパッケージを除去するには、以下のコマンドを入力します。

installp -u sna.client

smit を使用してパッケージを除去するには、以下のコマンドを入力します。

smit remove

Remote API Client on AIX のアンインストール

第 7 章 Remote API Client on Windows の計画とインストール

本章では IBM Remote API Client on Windows のインストール方法を記載してあります。これにより、PC 上で完全な SNA スタックのインストール環境を保有せずに SNA アプリケーションを PC が実行可能となります。Remote API Client on Windows は 1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー) と TCP/IP ネットワークを介して接続可能です。

IBM Remote API Client on Windows には 2 種類の異なる形態があります。これは、お客様が使用している具体的なハードウェアと Windows バージョンにより異なります。本章の内容の適用対象は、その 2 つの差が明確に注意書きしている部分を除き、両方の形態を対象としています。

- 32 ビット・クライアントは Intel ベースのコンピューター上で実行します。このクライアントのオペレーティング・システムは Microsoft Windows 2000、2003、または XP です。
- x64 クライアントは AMD64 または Intel EM64T コンピューター上で実行します。このクライアントのオペレーティング・システムは Microsoft Windows Server 2003 x64 Edition または Microsoft Windows XP Professional x64 Edition です。

IBM Remote API Client on Windows が提供するインターフェースは、IBM Communications Server for Windows、および Microsoft Host Integration Server 製品により提供されるインターフェースと広範囲な互換性があります。

IBM Remote API Client on Windows Software Development Kit (SDK) は、Remote API Client を使用して APPC、CPI-C、LUA、および CSV API を使用するアプリケーション・プログラムを開発することができるオプションのパッケージです。これらの API についての詳細は、適切なプログラマー向け参照ガイドを参照してください。Remote API Client を既存のアプリケーションを実行するためのみに使用する場合 (新規アプリケーションを開発しない場合)、このパッケージをインストールする必要はありません。

ハードウェアおよびソフトウェア要件

セットアップ プログラムおよび Remote API Client on Windows を実行するには、そのコンピューターは次の要件を満たしている必要があります。

- 以下のいずれかのオペレーティング・システムを実行している必要があります。
 - 32 ビット Windows クライアントの場合:
 - Windows 2000
 - Windows XP
 - Windows 2003
 - x64 Windows クライアントの場合:
 - Microsoft Windows XP Professional x64 Edition

- Microsoft Windows Server 2003 x64 Edition
- 以下のいずれかの手段で 1 つ以上の Communications Server for Linux サーバーにアクセスする必要があります。
 - TCP/IP ネットワーク経由でこのサーバーにアクセス
 - Communications Server for Linux サーバーに HTTPS アクセスを提供する WebSphere へのアクセス (このケースでは、クライアント・セットアップ・プロセスの一部として GSKIT ソフトウェアもインストールされている必要あり)。

セットアップ・プログラムへのアクセス

Remote API Client と SDK ソフトウェア、GSKIT ソフトウェア、および **Setup** プログラムは、インストール CD に Windows 形式で収容されているため、これらのソフトウェアを CD から Windows コンピューターにインストールできます。Remote API Client ソフトウェアをそれぞれの Windows クライアント PC にインストールする必要があります。SDK が必要になるのは、このクライアントを使って Windows リモート API を使用する新規アプリケーションを開発する場合にのみです。このクライアントを既存のアプリケーションの実行のみに使用する場合は不要です。この GSKIT ソフトウェアが必要となるのは、このクライアントが HTTPS を使用する Communications Server for Linux サーバーにアクセスする場合だけです。

Remote API Client on Windows のインストール・イメージは自己解凍型 ZIP 実行可能ファイルで、インストール CD 上で提供されます。

- 32 ビットの場合、このイメージはこの CD 上のディレクトリー `/ibm-commserver-clients/windows` 中にある `i_w32cli.exe` です。
- x64 クライアントの場合、このイメージはこの CD 上のディレクトリー `/ibm-commserver-clients/win-x64` 中にある `i_w64cli.exe` です。

このファイルはネットワーク内のその他の Windows PC にコピーできるため、Communications Server for Linux 配布 CD に直接アクセスせずにインストールできます。この実行可能ファイルを実行すると、インストール・イメージを unzip し、**Setup** プログラムを自動的に実行します。このインストール・イメージを単に unzip して一時的ディレクトリーに入れたいだけの場合は (例えば、コマンド行から **Setup** プログラムを実行する目的で)、自己解凍型 ZIP 実行ファイルを unzip プログラムにロードすることによりそれを行うことができます。

初めて **Setup** プログラムを実行するコンピューターでは、プログラムは選択されたソースから実行されます。プログラムは、インストール・プロセス全体を処理し、基本構成をセットアップし、このプログラムのアイコンをインストールおよび作成します。インストールが完了した後に、このソフトウェアを再インストールする必要がある場合は、(ファイル・マネージャーからプログラムを選択するか、作成されたアイコンを選択して) **Setup** プログラムを使用できます。

Remote API Client インストール・イメージを一時ディレクトリーに解凍完了後、以下の 2 つの方法のいずれかでこのソフトウェアをインストールできます。

- **Setup** プログラムを Windows を介して実行します (『セットアップ・プログラムを使用した Remote API Client on Windows のインストール』を参照してください)。SDK をインストールする場合は、この方法でインストールする必要があります。
- コマンド行から **setup** コマンドを入力します (63 ページの『コマンド行からの Remote API Client ソフトウェアのインストール』を参照してください)。この方法では、SDK をインストールできません。

注: HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT ソフトウェアをインストールし、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティ証明書構成をセットアップする必要があります。詳しくは、65 ページの『GSKIT ソフトウェアのインストールとセキュリティ証明書のセットアップ』と 66 ページの『GSKIT を使用した HTTPS セキュリティ証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、このクライアントが接続対象とする Communications Server for Linux サーバー、および HTTPS をサポートする WebSphere サーバー名を指定する必要があります。詳細は、「*Communications Server for Linux 管理ガイド*」にある Remote API Client の管理に関するセクションを参照してください。

セットアップ・プログラムを使用した Remote API Client on Windows のインストール

setup プログラムを実行します。それを行うには、自己解凍型 ZIP 実行可能ファイル **i_w32cli.exe** (32 ビット・クライアント) または **i_w64cli.exe** (x64 クライアント) の実行の一部として自動的に行うか、またはコマンド行から手動で行います。このプログラムでは、最初に「セットアップ言語の選択 (Choose Setup Language)」画面が表示されます。

1. Remote API Client クライアントのインストールおよび構成に使用する言語を選択して、「**OK**」を選択する。

Setup プログラムに導く、プログラムの「ウェルカム (Welcome)」画面が表示されます。

2. 「次へ (**Next**)」を選択して、インストールを続行する。

プログラムの「ソフトウェア・ライセンス契約 (Software Licensing Agreement)」が表示されます。内容を読んで理解してください。

3. ライセンス条件を受け入れる場合、「**同意する (Accept)**」を選択して続行する。

プログラムにより、ファイルをインストールする宛先ディレクトリーを指定するようにプロンプトが出されます。

4. 宛先ディレクトリーを入力する。

プログラムにより、インストールのタイプを尋ねられます。

標準 (Standard)

このオプションを選択するのは、この SDK のインストールが不要な場

セットアップ・プログラムを使用した Remote API Client on Windows のインストール

合です。SDK が必要になるのは、このクライアントを使って Windows リモート API を使用する新規アプリケーションを開発する場合にのみです。このクライアントを既存のアプリケーションの実行のみに使用する場合は不要です。

開発者 (Developer)

このオプションを選択するのは、この SDK をインストールする必要がある場合です。すなわち、Windows Remote API を使用する新規アプリケーションを開発するのにこのクライアントを使用予定の場合です。

注: この SDK をインストールしたい場合、「開発者 (Developer)」を選択する必要があります。

5. インストール・タイプを選択する。

次にこのプログラムは Remote API Client on Windows のアイコンを表示させたいプログラム・フォルダーの名前を入力するよう求めます。

6. フォルダー名を入力する。

7. システム・ディレクトリーに、この Setup プログラムに使用されるファイルと同じ名前だが、Remote API Client ファイルではない .DLL ファイルが既にある場合 (たとえば、他の SNA ソフトウェアのファイルなど)、プログラムから、以下のいずれかを行うようプロンプトが出される。

- Remote API Client の .DLL ファイルを既存の .DLL ファイルに上書きコピーする。
- 既存の .DLL ファイルをインストール・ディレクトリー内の OTHERSNA というサブディレクトリーにコピーしてから、Remote API Client の .DLL ファイルをインストールする。このオプションを選択すると、Remote API Client ファイルを後でアンインストールする場合に、Remote API Client をインストールする前のオリジナル・セットアップを復元できます (68 ページの『Remote API Client ソフトウェアのアンインストール』を参照してください)。
- クライアント・ソフトウェアのインストールを取り消す。

Remote API Client の .DLL ファイルが既に存在する場合は、Setup プログラムにより、そのことを示すメッセージが表示されます。既存の .DLL ファイルのバージョン番号が Setup プログラム .DLL ファイルよりも前である場合のみ、新規の .DLL ファイルが既存ファイルを上書きします。

8. この時点で、Setup プログラムは指定されたソースからファイルをコピーして、適切な場所にインストールする。この処理が行われる間、インストールが完了した部分を示す情報バーが表示されます。 .DLL ファイルはシステムまたは同等のディレクトリーにコピーされ、その他のファイルはステップ 2 で指定した宛先ディレクトリーにコピーされます。各ファイルの転送操作の間、指定したディレクトリーに作成された setup.log ファイルにレコードが書き込まれます。書き込みが行われるファイルが「読み取り専用」である場合、またはその他の理由でコピーできないファイルがある場合、新規ファイルは除去され、setup.log ファイルを確認するよう求めるメッセージを受け取ります。

9. Setup プログラムを実行するソースに必要なすべてのファイルがない場合、プログラムによりディレクトリー名を求めるプロンプトが出される。必要なファイルが置かれているディレクトリーの名前を入力してください。

セットアップ・プログラムを使用した Remote API Client on Windows のインストール

指定された情報で Remote API Client ファイルのコピーが見付からなかった場合、プログラムはこの画面を再度表示します。

10. 必要なファイルがコピーされると、**Setup** プログラムは「構成 (Configuration)」ウィンドウを表示する。

デフォルト構成値はドメイン構成ファイルから取られます。詳しくは、「*Communications Server for Linux 管理ガイド*」を参照してください。これらのデフォルト値を使用しない場合は、以下のように値を構成できます。

ドメイン (Domain)

Communications Server for Linux クライアント/サーバーのドメイン名を指定します。

以下の設定値はオプションです。

サーバー名 (Server Name)

画面に、このクライアントが接続できる最大 9 つのサーバーのリストが表示されます。このリストに表示されるサーバーの順序は、クライアントがこれらのサーバーを選択する順序です。クライアントが先頭のサーバーに接続できない場合は、次のサーバーで試行されます。

- リストに新規サーバーを追加するには、「**追加 (Add)**」ボタンを使用します。
- リストからサーバーを除去するには、該当するサーバーを選択して「**除去 (Remove)**」ボタンを使用します。
- リスト内でサーバーの順序を変えるには、該当するサーバーを選択して、リストの横にあるスライド・ボタンを使用します。

このクライアントがそのサーバーと同じプライベート・ネットワーク上にあり、TCP/IP を使用してそのサーバーにアクセスする場合、各サーバーは単にそのサーバー名により識別されます。

このクライアントが HTTPS を使ってそのサーバーにアクセスする場合、各サーバーを識別するには、HTTPS サポートを提供する

WebSphere サーバー名と、Communications Server for Linux サーバー名を以下の形式で指定する必要があります。

webservername : servername1

この前提として、WebSphere は HTTPS 接続に対してデフォルト・ポート 443 を使用してセットアップされているとします。お客様のネットワーク管理者がこれとは異なるポート番号で WebSphere を構成済みの場合、以下の形式でポート番号を組み込みます。

webservername : portnumber : servername1

HTTPS 接続をサポートする WebSphere の構成方法の詳細は、31 ページの『WebSphere Application Server の構成』を参照してください。

UDP ブロードキャスト (UDP broadcasts)

このクライアントがサーバーへの接続に UDP ブロードキャストを使用するかどうかを指定します。このオプションが選択されると、クライアントは、特定のサーバーに直接接続しようとは試行するのではなく、ネットワークを介して UDP ブロードキャストを送信してサーバー接続を探索します。

セットアップ・プログラムを使用した Remote API Client on Windows のインストール

デフォルト設定では、UDP ブロードキャストを使用します。この設定を変更するには、ボックスをクリックします。

拡張 (Advanced)

Setup プログラムにより提供されるデフォルトの代わりに追加の値を入力する場合は、ウィンドウの下部にある「**拡張 (Advanced)**」ボタンをクリックします。**Setup** プログラムにより、「**拡張オプション (Advanced Options)**」ウィンドウが表示されます。このウィンドウには、Windows クライアント構成の拡張設定が表示されています。ほとんどのユーザーは、これらのパラメーターにデフォルトの設定値を使用することができ、このダイアログで設定値を変更する必要はありません。

これらのパラメーターについて詳しくは、『Remote API Client 構成用の拡張オプション』を参照してください。

構成パラメーターまたは設定値について詳しくは、「**ヘルプ (Help)**」をクリックしてください。

11. 「**構成 (Configuration)**」ウィンドウで設定が完了したら、「**OK**」をクリックする。この画面での設定が完了していない場合は、**Setup** プログラムによりメッセージが表示されます。
12. インストールが正常に完了すると、「**終了 (Finish)**」ウィンドウが表示される。インストール・プログラムの終了後のアクションとして、以下のいずれか、もしくは両方を選択できます。

README ファイルの表示 (View README file)

README ファイルを表示します。

クライアントの始動 (Start client)

Communications Server for Linux クライアントの実行を開始する。

「**終了 (Finish)**」を選択して、インストール・プログラムを終了します。

Remote API Client 構成用の拡張オプション

「**拡張オプション (Advanced Options)**」ウィンドウを使用して、Remote API Client の一部の拡張パラメーターを構成できます。ほとんどのユーザーはこれらのパラメーターを変更する必要はありませんが、必要な場合はデフォルトの設定値を調整できます。

LAN アクセスのタイムアウト (LAN access time-out)

サーバーへのクライアント接続がクローズされる前にアイドル状態のままにいられる時間を秒単位で指定します。このチェック・ボックスが空である場合、LAN アクセスのタイムアウトは指定されません (タイムアウトは無限大になります)。このボックスにチェックマークを付けると、隣接するフィールドにタイムアウト値を秒単位で入力できます。最小値は 60 (60 秒間) です。このボックスをブランクのままにしたり、60 未満の値を指定すると、Remote API Client は最小値の 60 を使用します。

ブロードキャスト試行の最大回数 (Max. broadcast attempts)

クライアントがブロードキャストを使用してサーバーへの接続を試行する最大回数を指定します。「**拡張オプション (Advanced Options)**」ウィンドウが

オープンされている場合、デフォルト値の 5 が表示されます。このボックスの値は、メインの「構成 (Configuration)」ウィンドウで UDP ブロードキャストのチェック・ボックスにチェックマークが付けられている場合のみ使用されます。

再接続タイムアウト (Reconnect time-out)

サーバーがダウンしてからサーバーへの再接続を試行するまでクライアントが待機する時間を秒単位で指定します。「拡張オプション (Advanced Options)」ウィンドウがオープンされている場合、デフォルト値の 200 が表示されます。

これらのパラメーターについて詳しくは、「ヘルプ (Help)」を押してください。

「拡張オプション (Advanced Options)」ウィンドウで設定が完了したら、「OK」をクリックします。画面の設定が正常に完了すると、**Setup** プログラムは「構成 (Configuration)」ウィンドウに戻ります。新しい Remote API Client をインストールする場合は、ステップ 11 (62 ページ) に戻ってください。そうでない場合は、「構成 (Configuration)」ダイアログの「OK」ボタンをクリックして構成を完了します。

コマンド行からの Remote API Client ソフトウェアのインストール

注: この SDK をインストールする場合は、**Setup** プログラムを使用する必要があります。59 ページの『セットアップ・プログラムを使用した Remote API Client on Windows のインストール』を参照してください。この SDK はコマンド行からインストールできません。

Remote API Client のインストール・イメージを一時ディレクトリーに解凍した後、Windows で **Setup** プログラムを使用する代わりに、コマンド行から Remote API Client ソフトウェアをインストールできます。コマンド行で、1 つ以上のオプションを指定して **setup** コマンドを入力します。オプションは大文字でも小文字でも入力でき、オプションの前に / (スラッシュ) または - (ハイフン) のいずれかを付けることができます。*folder* などのパラメーターが、スペースを含む文字列である場合、ストリングを二重引用符で囲む必要があります。

setup コマンドを入力した後、**Setup** プログラムによりコマンド行に含まれない情報を求めるプロンプトが出され、セットアップのさまざまな段階で確認メッセージが表示されます。**Setup** プログラムによるプロンプトを出したくない場合は、**-accept -s** オプションを使用してプログラムをサイレント・モードで実行し、ソフトウェア・ライセンス契約を受け入れます。

以下に、**setup** コマンド・オプションを示します。

-? コマンド行オプションのリストを表示します。これは、**-h** オプションと同じです。

-h コマンド行オプションのリストを表示します。これは、**-?** オプションと同じです。

-accept -s

サイレント・モードでインストールを実行し、ソフトウェア・ライセンス契約を受け入れます。この契約の内容は、Windows インストール・イメージの **license** サブディレクトリーで見ることができます。

コマンド行からの Remote API Client ソフトウェアのインストール

-s オプションは、コマンド行で最後に指定する必要があります。また、ドメイン名 (**-i** オプションを使用して)、およびその他の該当するパラメーターを指定したことを確認してください。サイレント・モードでインストールが実行される場合、すべてのパラメーターに関してプロンプトは出されず、確認メッセージも表示されません。**-s** の後のコマンド行引数は無視されません。

-f2 サイレント・モード・インストール (**-s** オプションを使用) の間に作成されたインストール・ログ・ファイルの絶対パス名を指定します。

このオプションを指定しないと、このファイルは、インストール・プログラムの実行元のディレクトリーに **setup.log** という名前で作成されます。CD ドライブからサイレント・モードでインストールする場合、このオプションを指定してファイルがご使用のコンピューター上に作成されるようにしてください (CD ドライブ上には作成できないためです)。

-kfolder

プログラム・フォルダーを指定します。

-pdirectory

インストール・ディレクトリーを指定します。

-idomain

このクライアントのドメイン名を指定します。このパラメーターは必須です。デフォルトはありません。

-wdirectory

ソースがディスクまたは CD にある場合に、Communications Server for Linux クライアント・ソフトウェア・ファイルがあるソース・ディレクトリーを指定します。それ以外の場合は、**-v** オプションを使用します。

-vserver

クライアント・ソフトウェア・ファイルのダウンロード元のサーバーを指定します。サーバー名または TCP/IP アドレスを指定できます。ソース・ファイルをディスクまたは CD からコピーする場合は、**-v** オプションではなく **-w** オプションを使用してください。

-lserver

このクライアントがアクセスできるサーバーのリストに含めるサーバーを指定します。

このクライアントがそのサーバーと同じプライベート・ネットワーク上にあり、TCP/IP を使用してそのサーバーにアクセスする場合、各サーバーは単にそのサーバー名により識別されます。

このクライアントが HTTPS を使ってそのサーバーにアクセスする場合、各サーバーを識別するには、HTTPS サポートを提供する WebSphere サーバー名と、Communications Server for Linux サーバー名を以下の形式で指定する必要があります。

webservname : servname1

この前提として、WebSphere は HTTPS 接続に対してデフォルト・ポート 443 を使用してセットアップされているとします。お客様のネットワーク管理者がこれとは異なるポート番号で WebSphere を構成済みの場合、以下の形式でポート番号を組み込みます。

コマンド行からの Remote API Client ソフトウェアのインストール

webservername : portnumber : servername |

HTTPS 接続をサポートする WebSphere の構成方法の詳細は、31 ページの『WebSphere Application Server の構成』を参照してください。

- o 既存の **.DLL** ファイルを上書きします。Remote API Client の **.DLL** ファイルが既に存在する場合、これらのファイルのバージョン番号が **Setup** プログラムの **.DLL** ファイルより高位の場合でも、**Setup** プログラムはこれらのファイルを上書きします。
- y 既存の **.DLL** ファイルを保管します。必要なディレクトリーに Remote API Client の **.DLL** ファイルが既に存在する場合、**Setup** プログラムは既存の **.DLL** ファイルをインストール・ディレクトリーのサブディレクトリーにコピーしてから、Remote API Client の **.DLL** ファイルをインストールします。サブディレクトリーにコピーされると、Remote API Client ソフトウェアをアンインストールする際にアンインストール・プロセスが確実に行われます。
- n 既存の Remote API Client の **.DLL** ファイルが検出される場合に、インストールを取り消します。

-atimeout

LAN アクセスのタイムアウトを秒単位で指定します。これは、サーバーへのクライアント接続がクローズされる前にアイドル状態でいられる時間の長さです。値 0 は、タイムアウトが指定されないことを示します。

-bmax-broadcast

UDP ブロードキャスト試行の最大回数を指定します。UDP ブロードキャストは、クライアントが特定のサーバーではなくドメイン内の不特定のサーバーに接続しようと試みることです。値 0 は、ブロードキャストが試行されないことを示します。

-jreconnect-timeout

サーバーがダウンしてからサーバーへの再接続を試行するまでクライアントが待機する時間を秒単位で指定します。

GSKIT ソフトウェアのインストールとセキュリティ証明書のセットアップ

注: 初期リリースの x64 Remote API Client on Windows は、サーバーへの HTTPS 接続をサポートしません。このフィーチャーは PTF 形式で提供されることになります。詳細なインストール手順と構成手順も一緒に提供されます。以下の手順の適用対象は、32 ビット・クライアントに限定されます。

IBM Remote API Client が HTTPS を使用するサーバーに接続する場合、GSKIT ソフトウェアをインストールしてセキュリティ証明書を管理します。以下のステップを使用します。

1. Windows コンピューター上で一時ディレクトリーを作成して、インストール・ファイルを収容します。
2. 32 ビットクライアントの場合:

GSKIT ソフトウェアのインストールとセキュリティー証明書のセットアップ

- インストール CD 上の `/ibm-commserver-clients/windows` ディレクトリーから `gsk7bas.exe` ファイルをコピーして一時ディレクトリーに入れます。
- コマンド・ウィンドウで、この一時ディレクトリーに変わります。
- 以下のコマンドを実行してインストール・ファイルを抽出します。

```
gsk7bas tempdir /D
```

`tempdir` はこの一時ディレクトリーの絶対パスです。

- 以下のコマンドを実行してインストール・プログラムを開始します。

```
setup sxclient
```

3. スクリーン内の指示に従って GSKIT ソフトウェアをインストールします。
4. インストール完了後、最初のステップで作成した一時ディレクトリーを削除してその内容を廃棄します。

これで、以下のセクションに記載したセキュリティー証明書をセットアップできます。

32 ビットクライアント上での無人インストールを行う必要がある場合 (例えば、多くのクライアント・コンピューターに GSKIT ソフトウェアをインストールしようとして、処理を自動化するためにバッチ・ファイルを作成したい場合)、以下のコマンドを使用してサイレント・モードでインストール・プログラムを実行します。

```
setup gskit -s -fl "tempdir¥SETUP.ISS"
```

`tempdir` は、最初のステップで作成済みの一時ディレクトリーの絶対パスです。

このコマンドを使用すると、インストール・プログラムのユーザー・インターフェースを表示せずにソフトウェアがインストールされ、ユーザー入力は不要です。このインストールのデフォルト指定が **SETUP.ISS** ファイルから使用されます。このファイルは、インストール・プロセスの最初に一時ディレクトリーに抽出したファイルの 1 つです。GSKIT ソフトウェアをインストールするディレクトリーを変更する必要がある場合は、一時ディレクトリーへの抽出後にテキスト・エディターを使用して **SETUP.ISS** を編集して変更します。このディレクトリー内の各ファイル (この更新されたファイルを含む) は、次に他のクライアント・コンピューター上でのインストールに使用可能です。その都度、このファイルを抽出してインストール・ディレクトリーを変更する必要はありません。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

HTTPS を使用するサーバーに IBM Remote API Client を接続する前に、GSKIT の鍵管理機能プログラムを使用してクライアント上でセキュリティー証明書をセットアップする必要があります。以下のステップで行います。

1. GSKIT 鍵管理機能プログラムを実行します。このプログラムは `installdir¥snakeyman.exe` です。`installdir` はこのクライアント・ソフトウェアをインストールしたディレクトリーを意味します。クライアント・インストール時に別の場所を指定した場合を除き、これは **C:¥IBMCS¥w32cli** (32 ビット・クライアント) です。

GSKIT を使用した HTTPS セキュリティ証明書セットアップ

この鍵管理機能のユーザー・インターフェース内から鍵データベース・ファイル (`installdir\ibmcs.kdb`) をオープンします。このファイルは CMS 形式です。

- この鍵データベース用の初期パスワードは `ibmcs` です。セキュリティ証明書のセットアップ前に、このパスワードを変更して、ご使用の構成をセキュア状態に保つ必要があります。パスワード変更用ダイアログ上で、「ファイルに対するパスワードを隠しておく? (Stash the password to a file?)」チェック・ボックスにマークを付ける必要があります。これにより、新規のパスワードが保管されて、クライアントがこの鍵データベースをオープンできるようにします。
- Web サーバーのセキュリティ証明書を署名するのに使用された Certificate Authority (CA) 証明書のコピーを入手して、それを鍵データベースにインストールします。これを行うには、鍵管理機能のユーザー・インターフェースで「署名者証明書 (Signer Certificates)」を選択して「追加 (Add)」をクリックします。
- WebSphere サーバーがクライアントのセキュリティ証明書を必要とするように構成されている場合、そのクライアントは CA 発行の証明書を保有する必要があります。この CA 所有の証明書は Web サーバーのセキュリティ証明書データベースにあります。新規の証明書が必要な場合は以下のようにします。
 - 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規認証要求 (New Certificate Request)」を選択して、要求された詳細内容を入力します。
 - 証明書を保管し、それをファイルに抜き出し、CA に送信します。
 - この証明書を発行する場合、それを Web サーバーのデータベースに保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。

お客様自身の内部的なテストのための一時的な手段として、CA から証明書を入手する代わりに自己署名のクライアント証明書を作成できます。ただし、この手段では必要なレベルのセキュリティを提供しないため、本格稼働システムでは使用しないでください。自己署名証明書を作成するには以下を行います。

- 鍵管理機能のユーザー・インターフェースで「作成 (Create)」、「新規自己署名証明書 (New Self-Signed Certificate)」を選択して、要求された詳細内容を入力します。
 - この証明書を保管し、それをファイルに抜き出します。
 - Web サーバーのデータベースに証明書ファイルを保管します。これを行うには、鍵管理機能のユーザー・インターフェースで「個人証明書 (Personal Certificates)」を選択して「受け入れ (Receive)」をクリックします。
- 証明書の構成が完了したら、GSKIT 鍵管理機能を終了します。

インストール後の Remote API Client ソフトウェアのカスタマイズ

初期インストールの後、いつでも Communications Server for Linux プログラム・グループにある **Configuration Utility** プログラムを実行して、カスタマイズされた設定を変更することができます。このプログラムでは、初期インストール・プロセスで表示されたのと同じ「構成 (Configuration)」ウィンドウが表示されます。59 ページの『セットアップ・プログラムを使用した Remote API Client on Windows のインストール』の手順に従って、任意のフィールドの情報を変更できます。

インストール後の Remote API Client ソフトウェアのカスタマイズ

初期インストールでこの SDK ファイルをインストールせず、後で追加する場合、Setup プログラムを再実行して追加することができます。インストール・タイプとして「**カスタム (Custom)**」を選択して、この SDK パッケージを選択します。

Remote API Client ソフトウェアの再インストール

ソフトウェアのアップグレードなど、必要な場合にいつでも Remote API Client ソフトウェアを再インストールできます。

この操作を行うには、59 ページの『セットアップ・プログラムを使用した Remote API Client on Windows のインストール』、または 63 ページの『コマンド行からの Remote API Client ソフトウェアのインストール』の手順に従って、以前と同じ方法で Setup プログラムを実行します。Setup プログラムは、初期インストール時のクライアント・ソフトウェア・ファイルのコピー元の場所を表示します。「OK」をクリックして、この同じ場所からファイルの新規コピーを取得します。「OK」をクリックすると、Setup プログラムはファイルをコピーし、「オプション (Options)」画面に戻ります。

注: Remote API Client ソフトウェアをサイレント・モードでインストールする場合 (63 ページの『コマンド行からの Remote API Client ソフトウェアのインストール』に説明されている方法で)、インストールを完了するにはコンピューターを再始動する必要があります。これは、一部のプログラム・ファイルがインストール・プロセス中に使用中で (たとえば、Remote API Client が実行中である場合)、新規ファイルに置き換えられないことがあるためです。その場合、新規ファイルは一時ディレクトリーにコピーされ、コンピューターが次に再始動されると自動的に置き換わります。

コンピューターの再始動が必要かどうかを確認するには、インストール・プロセスが完了してからインストール・ログ・ファイルの内容をメモ帳などのテキスト・エディターで表示してください。インストール・ログ・ファイルは **setup.log** と呼ばれ、Setup プログラムを実行したディレクトリーに作成されます。ただし、**-f2** コマンド行オプションを使用して、異なるパスとファイル名を指定した場合は除きます。

ファイルの終わりにあるヘッディング **Response Result** の下に、テキスト **Result Code** に続いて値 0 (ゼロ) または -12 があります。値が 0 の場合、コンピューターを再始動する必要はありません。値が -12 の場合、Windows クライアントを使用する前にコンピューターを再始動してください。

Remote API Client ソフトウェアのアンインストール

Remote API Client ソフトウェアは、Windows の「コントロール パネル (Control Panel)」から「**削除 (Remove)**」オプションを使用して、いつでもアンインストールできます。アンインストール・プロセスが確認されると、Windows は以下のことを行います。

- すべてのインストール済みファイルを削除する。
- 初期インストール時にサブディレクトリーに保管された **.DLL** ファイルが存在する場合、これらのファイルを元の場所に復元する。

- 保管された **.DLL** ファイルが格納されていたサブディレクトリーが空になる場合、削除する。
- プログラム・フォルダーおよび作成されたディレクトリーが空である場合、除去する。
- アンインストールが正常に行われた場合、すべてのファイルの転送と削除が記録されている **setup.log** ファイルを削除する。
- アンインストールが正常に行われたことを示すメッセージ、あるいはインストールの一部に障害があるため **setup.log** ファイルを確認するよう求めるメッセージのいずれかを表示する。

「終了 (Exit)」ボタンで Windows に戻ります。

32 ビットの Windows Remote API Client で使用するために GSKIT ソフトウェアをインストール済みの場合で、他のアプリケーションは何もこの GSKIT を使用しない場合、『GSKIT ソフトウェアのアンインストール』に記載したようにこのソフトウェアをアンインストールしても構いません。

GSKIT ソフトウェアのアンインストール

32 ビットの Windows Remote API Client で使用するために GSKIT ソフトウェアをインストール済みの場合で、他のアプリケーションは何もこの GSKIT を使用しない場合、Windows Remote API Client のアンインストール完了後にこのソフトウェアをアンインストールしても構いません。以下のステップで行います。

1. コマンド・ウィンドウで、Windows をインストールしたディレクトリーに変わります。これは、通常は **C:¥Windows** です。
2. 次のコマンドを実行します。

gskbui sxclient

他のアプリケーションが何も GSKIT を使用していない場合、Windows はこれらのファイルを削除することになります。いずれかのアプリケーションが依然として GSKIT を使用している場合、これらのファイルは、その他のアプリケーションもアンインストールされるまでインストールされたままになります。

ヘルプ

F1 キーを押すと、いつでもヘルプにアクセスできます。「構成 (Configuration)」ウィンドウおよび「拡張オプション (Advanced Options)」ウィンドウにも、「ヘルプ (Help)」ボタンがあります。

第 8 章 Communications Server for Linux の構成と使用

Communications Server for Linux 構成を定義および変更する場合、Motif 管理プログラム (**xsnaadmin**) を使用する方法が最も簡単です。このプログラムのグラフィカル・ユーザー・インターフェースから、ローカル・ノード上の SNA リソースを表示および管理できます。コマンド行管理などの他の管理ツールを使用することもできますが、Motif プログラムの使用をお勧めします。

Motif 管理プログラムには、ヘルプ画面があります。この画面では、SNA および Communications Server for Linux に関する概要、Communications Server for Linux ダイアログの参照情報、特定の操作の説明が表示されます。このプログラムは、ノード構成などの各操作、あるいは TN3270 または APPC などの通信タイプに必要なリソースの構成操作を支援します。

Motif 管理プログラムでは、標準 Communications Server for Linux 構成に必要なすべてのパラメーターを設定できます。拡張パラメーターには、Motif 管理プログラムによりデフォルト値が設定されます。指定する必要があるのは主要な構成情報のみなので、SNA 通信を簡単かつ迅速に設定できます。

また、Motif 管理プログラムを使用して、稼働中の Communications Server for Linux システムを管理することもできます。この管理プログラムでは、Communications Server for Linux がアクティブであるときでも、変更内容を構成に適用できます。また、ノード・リソースの状況情報へも簡単にアクセスできます。

Motif 管理プログラムでは、Communications Server for Linux リソースに関する状況情報が自動的に表示されます。状況情報の大部分は、「ノード (Node)」ウィンドウに表示されます (74 ページの『Motif 管理プログラムによる Communications Server for Linux の管理』を参照してください)。さらに「ノード (Node)」ウィンドウの、「開始 (Start)」ボタンと「停止 (Stop)」ボタンを使用して、ノードやリンク・ステーションなどの特定のリソースを管理することもできます。その他のリソースの始動と停止は常に自動的に実行されるため、手動で制御する必要はありません。

注:

1. Communications Server for Linux のリソースを定義または変更するには、ログイン・グループ `sna` のメンバーである必要があります。
2. Motif 管理プログラムを使用するには、X 端末が必要です。
3. ウィンドウ内に表示されるボタンやアイコンなど、Motif 管理プログラムのユーザー・インターフェースに関する詳細については、そのプログラムのヘルプ画面または、「*Communications Server for Linux 管理ガイド*」を参照してください。
4. 特定のダイアログで選択した項目によっては、Motif 管理プログラムのウィンドウやダイアログが、本書に記載されているものと異なる場合があります。

コマンド行管理および NOF アプリケーション・プログラムなどその他の Communications Server for Linux 管理ツールについては、「*Communications Server*

for Linux 管理ガイド」、*「Communications Server for Linux 管理コマンド解説書」*、または*「Communications Server for Linux NOF プログラマーズ・ガイド」*を参照してください。

Communications Server for Linux の構成の計画

構成を変更する場合は、事前に全体的な計画を立てておくことが大切です。変更内容によっては、ローカル・ノードのユーザーのみでなく、ネットワーク全体のユーザーの混乱を招く恐れがあります。

ネットワーク・トポロジーに対するすべての変更内容を表すダイアグラムを作成すると便利です。他のノードへの接続を追加または除去する場合には、使用しているローカル・ノードと他のノードの関係図を作成してください。Motif 管理プログラムを使用して、既存のすべての接続に関する構成情報を入手し、この情報をノード・ダイアグラムに付け加えることができます。

ダイアグラムに新しいリソースを追加する場合には、新しいリソースが既存のリソースと重複していないかどうか、名前が競合していないかどうかを簡単に確認できます。同様に、ダイアグラムを使用すれば、除去する必要のあるリソースを判別し、重要なリソースの削除を防止できます。

複数のノードを使用してクライアント/サーバー Communications Server for Linux システムを構成する場合、必ずすべての Communications Server for Linux ノードおよびその接続リソースをダイアグラムに組み込んでください。その後、本章の説明に従ってスタンドアロン・ノードの構成と同じ方法で、各ノードを順番に構成できます。

実行する必要がある変更を判別したら、次にこの変更に必要な構成情報を収集します。特定の Communications Server for Linux 機能の構成情報を収集する際には、Motif 管理プログラムのオンライン・ヘルプにあるタスク・シート、あるいは「*Communications Server for Linux 管理ガイド*」の計画ワークシートを使用します。

この章では、Communications Server for Linux で使用可能な機能のうち、最も頻繁に使用される機能を構成する方法について説明します。各構成手順では、リソースを構成する前に収集する必要がある情報についても記述しています。

注: 本書では、Communications Server for Linux ダイアログに入力する必要がある構成情報の詳細については説明していません。特定のダイアログのフィールドの詳細については、Motif 管理プログラムの該当ダイアログのオンライン・ヘルプを参照してください。

計画ワークシート

Communications Server for Linux のリソースを構成する前に、新しいリソースの構成データをすべて収集します。サポートが必要な特定の機能またはアプリケーションの情報をすべて記録するため、「*Communications Server for Linux 管理ガイド*」の計画ワークシートを使用します。

ネットワーク管理者、ホスト管理者、アプリケーション・プログラマー、エンド・ユーザーなど、さまざまなソースの構成情報を収集しなければならない場合があります。

他のノードへ接続しようとしている場合、そのノードの管理者と連絡をとることが大切です。ノード上のすべてのリソースについて、名前、アドレス、特性をそのノードの管理者から入手できます。また、ローカル・ノードで入力される構成パラメーターと、リモート・ノードで入力される構成パラメーターが一致していることを、確認する必要がある場合があります。

作業シート

Motif 管理プログラムのオンライン・ヘルプ画面には、特定の構成タスクに関する説明を提供する作業シートがあります。作業シートには、構成情報を入力するために使用される、各ダイアログのヘルプ画面へのポインターが示されます。このポインターを使用してヘルプ情報を表示し、収集する必要のあるデータを確認することができます。

作業シートでは、構成情報を入力するために使用する、ウィンドウやダイアログごとの詳細なヘルプも参照できます。これらのヘルプ画面では、入力または選択する必要のある各フィールドの説明が表示されます。

Motif 管理プログラムの使用

Motif 管理プログラムを使用する前に、システムが実行可能プログラムを検出できるように、各自の **.login** ファイルまたは **.profile** ファイルに、パス情報を追加します (『Communications Server for Linux プログラムのパスの指定』を参照してください)。また、管理プログラムを使用する前に、必ず Communications Server for Linux ソフトウェアを使用可能にしてください (74 ページの『Communications Server for Linux の使用可能にする方法』を参照してください)。

Motif 管理プログラムの起動方法の詳細については、74 ページの『Motif 管理プログラムによる Communications Server for Linux の管理』を参照してください。

Communications Server for Linux プログラムのパスの指定

Communications Server for Linux プログラムを実行するには、Communications Server for Linux 実行可能プログラムが入っているディレクトリーのパスを指定する必要があります。パスを指定するには、プログラムを初めて実行する前に、環境変数 **PATH** にディレクトリーを追加するか、プログラムを実行するたびにディレクトリー名を指定します。

Motif 管理プログラムは **/opt/ibm/sna/bin/X11** ディレクトリーに保管され、その他のプログラムは **/opt/ibm/sna/bin** に保管されています。**.login** ファイルまたは **.profile** ファイルの、環境変数 **PATH** の定義にこれらのディレクトリーを追加すると、Communications Server for Linux により自動的にプログラムが検出されます。あるいは次のように、プログラムの実行時にディレクトリー名を指定することもできます。

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

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

本書に記載されているコマンド行の例では、環境変数 `PATH` にディレクトリーを追加していることを前提としているため、ディレクトリー名は含まれていません。

Communications Server for Linux の使用可能にする方法

ローカル・ノードの構成や管理を行うには、Communications Server for Linux がローカル・システム上で使用可能な状態でなければなりません。他の X/Motif アプリケーションと同様、適切な X サーバーを指定するため、環境変数 `DISPLAY` を設定する必要があります。

Communications Server for Linux を使用可能にするには、Linux コマンド・プロンプトに次のコマンドを入力します。

sna start

Communications Server for Linux をインストールすると、インストール・ユーティリティーは、スタートアップ・ファイル `/etc/rc.d/init.d/snastart` を自動的に更新して **sna start** コマンドを組み込みます。これによって、Communications Server for Linux は確実に、システム起動時に自動的に始動します。Communications Server for Linux が自動的に始動しないようにしたい場合は、この行を除去するかコメント化し、次にこの節の指示に従って、Communications Server for Linux ソフトウェアを手動で使用可能にしてください。

標準エラー表示 (通常は端末の画面) に、Communications Server for Linux が初期化中であることを示すメッセージと、初期化が正常に終了したかどうかを示すメッセージが書き込まれます。

Motif 管理プログラムによる Communications Server for Linux の管理

Communications Server for Linux で Motif 管理プログラムを使用するには、まず最初に、『Communications Server for Linux の使用可能にする方法』で説明されているように、Communications Server for Linux が初期化されていることを確認します。(適切な X サーバーを指定するために、環境変数 `DISPLAY` も設定する必要がある場合もあります。)

Motif 管理プログラムをバックグラウンドで始動するには、次のコマンドを入力します。

xsnaadmin &

Communications Server for Linux が「ドメイン (Domain)」ウィンドウを表示します。このウィンドウにはすべての定義済みノードが表示され、ユーザーはノードの始動と停止を行うことができます。ノードをダブルクリックすると、76 ページの図 5 に示されるように、そのノードの「ノード (Node)」ウィンドウが開きます。

「ノード (Node)」ウィンドウには、ノードとそのリソースに関する情報が表示されます。ノードを構成していない場合は、管理プログラムから、ノードを構成するという指示が出されます。ノードの構成については、80 ページの『ノードの構成』で説明します。

注: 本書では、ウィンドウという用語は、Communications Server for Linux リソースの情報を表示する Motif ウィンドウを意味します。ウィンドウは、1 つ以上のセクションあるいはペイン (画面区画) からなります。ダイアログは、ユーザーが情報を入力できる Motif ウィンドウです。

「ノード (Node)」ウィンドウには必要な情報のほとんどが表示されており、どの情報へも簡単にアクセスできるようになっています。このウィンドウには、ローカル・ノード上の主要なリソースがすべて表示されています。

複数のノードを使用してクライアント/サーバーの Communications Server for Linux システムを構成する場合、本章の手順に従って各ノードを順番に (「ドメイン (Domain)」ウィンドウに戻って次のノードを選択して) 構成してください。

その他のウィンドウへは、「ノード (Node)」ウィンドウの「**ウィンドウ (Windows)**」メニューからアクセスできます。それらのウィンドウには、次のものがあります。

- 「LU プール (LU Pools)」ウィンドウ
- 「CPI-C 宛先名 (CPI-C Destination Names)」ウィンドウ

「ノード (Node)」ウィンドウの「**サービス (Services)**」メニューでは、リソースの追加が容易に実行できるようになっています。また構成作業および管理作業のヘルプも表示されます。「**診断 (Diagnostics)**」メニューでは、「**ログ記録 (Logging)**」ダイアログと「**トレース (Tracing)**」ダイアログが表示されます。

「ノード (Node)」ウィンドウ

「ノード (Node)」ウィンドウの例を 76 ページの図 5 に示します。タイトル・バーに Linux システムの名前が表示されます。

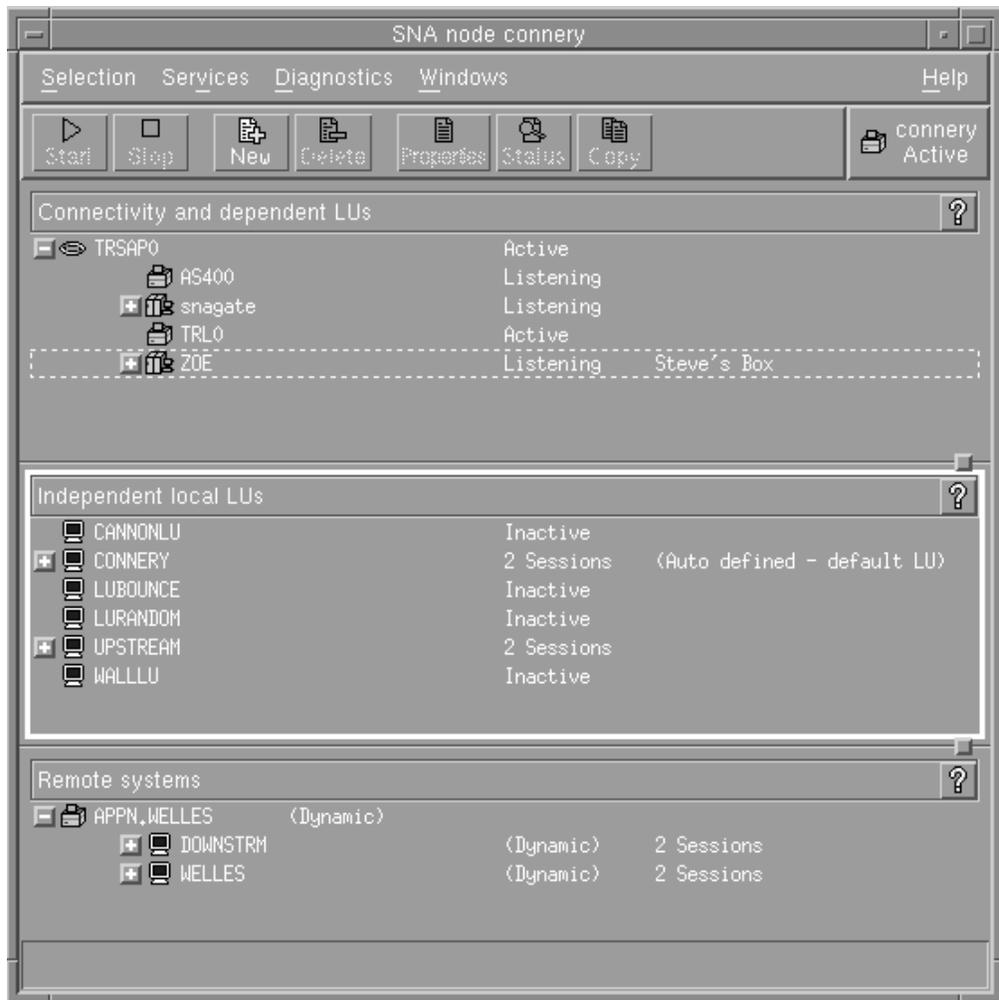


図5. 「ノード (Node)」 ウィンドウ

「ノード (Node)」 ウィンドウでは、Communications Server for Linux ノードのすべてのリソースとコンポーネントの構成と管理を行うことができます。

- ポート
- リンク・ステーション
- タイプ 0 から 3 の LU およびタイプ 6.2 の従属 LU
- DLUR 内部 PU
- 独立型ローカル LU
- リモート・ノード
- パートナー LU

「ノード (Node)」 ウィンドウでは、上記のリソースの追加、削除、変更、および管理を行うことができます。ウィンドウに表示されるリソースのレイアウトには、リソース間の関係が示されます。また、レイアウト中のリソースの表示を制御することができます。

ポート、ローカル LU、リモート・ノードは常に表示されます。「ノード (Node)」 ウィンドウには、各リンク・ステーションがその親ポートの下に表示され、各従属

LU がその親リンク・ステーションの下に表示されます。また、パートナー LU はローカル LU とリモート・ノードの下に表示されます。

「ノード (Node)」ウィンドウでは、ノードのリソースがタイプ別に各セクションにまとめられています。

- 「ノード (Node)」ウィンドウの右上隅にある「ノード (Node)」ボックスには、ノードが Active または Inactive のいずれであるかが示されます。
- 「ノード (Node)」ウィンドウの最上部のペイン (「接続 (Connectivity)」ペイン) には、ポート、リンク・ステーション、各ポートの PU、特定のリンク・ステーションまたは PU の従属 LU など、ノードの接続リソースのリストが表示されます。各リソースごとの現行の状況情報も表示されます。
- 中央のペイン (「独立型ローカル LU (Independent Local LUs)」ペイン) には、ローカル・ノード上で定義されている独立 LU が表示されます。このウィンドウには、特定の LU を使用するセッションに関する情報と、パートナー LU にアクセスするために使用されるリンク・ステーションによる、パートナー LU の位置を定義するレコードも表示されます。
- 最下部のペイン (「リモート・システム (Remote Systems)」ペイン) には、リモート・ノードとパートナー LU に関する情報が表示されます。また、各リモート・ノードまたは各パートナー LU のセッション情報も表示されます。

ペインを選択するには、ペインをクリックします。ペインの中に表示されている特定のリソースを選択するには、そのリソースの行をクリックします。特定の項目の構成を表示または変更するには、その項目をダブルクリックします。(特定のリソースの構成情報にアクセスする場合は、このウィンドウのボタンとメニューを使用してアクセスできます。)

リストに表示されている項目ごとに、その項目に属するリソースが、項目の情報にネストされています。たとえば、リンク・ステーションは、属するポートごとにまとめられます。項目のリソースが表示されていない場合には、項目の隣にある「**展開 (Expand)**」ボタン (+) をクリックすると、リソースを表示できます。また、「**縮小 (Contract)**」ボタン (-) をクリックすると、項目のリソースを隠すことができます。

「ノード (Node)」ウィンドウから次の管理タスクを実行できます。

リソースの始動と停止

リソースを選択して、「**開始 (Start)**」ボタン、または「**停止 (Stop)**」ボタンをクリックします。(あるいは「**選択 (Selection)**」メニューから、「**項目の開始 (Start item)**」または、「**項目の停止 (Stop item)**」を選択します。)

項目にリソースを追加

項目を選択して、「**新規 (New)**」ボタンをクリックします (あるいは、「**選択 (Selection)**」メニューから「**新規 (New)**」を選択します)。たとえば、ポートのリンク・ステーションを追加するには、ポートを選択してから「**新規 (New)**」ボタンをクリックします。

リソースの削除

項目を選択して、「**削除 (Delete)**」ボタンをクリックします (あるいは、「**選択 (Selection)**」メニューから「**削除 (Delete)**」を選択します)。

リソース構成の表示または変更

リソースを選択して、「プロパティ (Properties)」ボタンをクリックします (あるいは、「選択 (Selection)」メニューから、「プロパティ (Properties)」を選択します)。

リソースの状況情報の取得

項目を選択して、「状況 (Status)」ボタンをクリックします (あるいは、「選択 (Selection)」メニューから「状況 (Status)」を選択します)。

リソース構成のコピー

リソースを選択して、「コピー (Copy)」ボタンをクリックします (あるいは、「選択 (Selection)」メニューから「コピー (Copy)」を選択します)。

さらに、ノードの特定の構成タスクの選択 (「サービス (Services)」メニュー)、ログ記録 (ドメインの場合) とトレース (ノードの場合) の制御 (「診断 (Diagnostics)」メニュー)、ドメイン・リソースの表示、または変更 (「ウィンドウ (Windows)」メニューで項目を 1 つ選択) などの操作が可能です。

リソース項目

ウィンドウに表示されるリソースのレイアウトには、リソース間の関係が示されません。

項目に 1 つ以上の子項目が関連付けられている場合、項目の隣に「展開 (Expand)」記号 (+)、または「縮小 (Contract)」記号 (-) が表示されます。

- 「展開 (Expand)」記号は、関連付けられている子項目が隠されていることを示します。隠れている子項目を表示するには、「展開 (Expand)」記号をクリックするか、数字キーパッドの + キーを押します。
- 「縮小 (Contract)」記号は、子項目が表示されていることを示します。表示されている子項目を隠すには、「縮小 (Contract)」記号をクリックするか、数字キーパッドの - キーを押します。
- 項目の隣にどちらの記号も表示されていない場合は、関連する子リソースがないことを示します。

たとえば、特定のポートに関連付けられているリンク・ステーションがあるとして、 「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインでは、このリンク・ステーションが、同じ親ポートに関連付けられているその他のすべてのリンク・ステーションと共に、親ポートの下に表示されます。ポートは常に表示されますが、関連付けられているリンク・ステーションについては、表示または非表示を選択できます。同様に、関連付けられている LU のリストがあるリンク・ステーションの場合も、展開して LU を表示することも、縮小して LU を隠すこともできます。

親リソースは必ず子リソースよりも前に構成しなければなりません。親リソースを削除すると、その子リソースもすべて削除されます。

ツールバー・ボタン

リソース・ウィンドウには、主な機能を簡単に実行できるように、ツールバー・ボタンが組み込まれています。79 ページの図 6 に、Communications Server for Linux のツールバーを示します。



図 6. Communications Server for Linux ツールバー

各リソース・ウィンドウのツールバーに、すべてのボタンが表示されるわけではありません。現在選択している項目に対して、ボタンの操作が無効な場合（あるいは、操作対象の項目を選択する必要があるが、選択されていない場合）は、ボタンの枠線がぼかし表示されます。この場合、ボタンの機能は選択できません（ボタンをクリックしても何も実行されません）。リソース・ウィンドウには、次のボタンが表示されます。

開始 (Start)

選択した項目を始動します。

停止 (Stop)

選択した項目を停止します。

新規 (New)

新しいリソース項目を追加します。

削除 (Delete)

選択したリソースを削除します。

プロパティ (Properties)

選択した項目の構成を表示および変更できるダイアログをオープンします。

状況 (Status)

選択した項目の現在の状況が表示されます。

コピー (Copy)

選択した項目をコピーします。このボタンをクリックすると、表示されるダイアログのフィールドに、選択された項目の構成が複製されます。新しいリソースを追加するには、ダイアログのフィールドに情報をすべて入力します（新しい項目の名前を入力します）。

ポートやリンク・ステーションなどの多くのリソースは、アクティブである場合には変更できません。ただし、アクティブであるリソースのパラメータを表示することはできます。パラメータを表示するには、リソースを選択して、「**プロパティ (Properties)**」ボタンをクリックし、ダイアログをオープンします。パラメータを確認したら、「**クローズ (Close)**」ボタンをクリックしてください。

クライアント/サーバー機能の構成

このセクションは、Communications Server for Linux をクライアント/サーバー環境（同一ネットワーク内に複数の Communications Server for Linux ノードがある環境）で実行するためにインストールした場合にのみ関連します。

クライアント/サーバー環境では、サーバーを構成サーバーとしてマークすることができます。Communications Server for Linux はこれらの構成サーバーのリストを保守します。最初にリストされるサーバーはマスター・サーバーで、リスト内のその他のサーバーはバックアップ・サーバーです。サーバーは、マスター・サーバーが使用不可になった場合に 2 番目にリストされたサーバー（第 1 のバックアップ・

クライアント/サーバー機能の構成

サーバー) が継承し、マスターも第 1 のバックアップ・サーバーも使用不可になった場合に 3 番目にリストされたサーバー (2 番目のバックアップ・サーバー) が継承する、というように順序に従ってリストされています。

ドメイン内にアクティブなノードがある場合、ドメイン内で最初に使用可能な構成サーバー (接続可能で、Communications Server for Linux ソフトウェアを実行している第 1 のサーバー) がマスター・サーバーになります。現行のマスター・サーバーが使用不可になった場合 (ネットワーク障害または実行中の SNA ソフトウェアの停止により接続できなくなった場合)、リストの次にある使用可能な構成サーバーが新しいマスターになります。

Communications Server for Linux はマスターなしでも実行できます。この状態は、構成サーバー・リストにあるどのサーバーにも接続できない場合に発生します。この状態が発生すると、接続できるサーバーでのみノード・リソースの表示および構成を行うことができます。

注: どのノードがマスター・サーバーとして機能するか直接的に指示することはできません。マスター・サーバーは、ノードが構成サーバー・リストに追加された順番に選択されます。あるサーバーをリストの先頭に移動したい場合、リストからその他のすべてのノードを除去してから、再度それらのノードを追加してください。

Motif 管理プログラムの「ドメイン (Domain)」ウィンドウで、「**選択 (Selection)**」メニューから「**構成サーバーにする (Make configuration server)**」を選択して、構成サーバーを追加できます。そのサーバーはリストの最後に追加されます。このサーバーがマスター・サーバーになるのは、その他のすべての構成サーバーが使用不可である場合のみです。サーバーを除去するには、「**選択 (Selection)**」メニューから「**構成サーバーの除去 (Remove configuration server)**」を選択します。

注: Communications Server for Linux ソフトウェアが実行されているサーバーが 1 台しかない場合は、そのサーバーを削除できません。この場合、マスター・サーバーとしてテークオーバーできるサーバーが他にないためです。クライアント/サーバー構成では、少なくとも 1 つの使用可能なマスター・サーバーが必要です。

クライアント/サーバー Communications Server for Linux システムの構成および管理について詳しくは、「*Communications Server for Linux 管理ガイド*」を参照してください。このマニュアルには、クライアントとサーバーを別の Communications Server for Linux ドメインに移動する方法や、クライアント操作の詳細を構成する方法など、高度なクライアント/サーバー構成に関する情報も記載されています。

ノードの構成

システム上で Communications Server for Linux を構成する最初の手順は、ローカル・ノードを構成することです。ノードの構成により、SNA ネットワーク内で通信するためにノードに必要な基本情報が作成されます。ノードの接続やその他のリソースを定義する前に、まずノードを構成する必要があります。

ノードがすでに構成されている場合には、この項で説明する手順に従ってノードの構成を変更できます。ただし、構成を変更する前に、必ずノードを停止させてください。

Communications Server for Linux ノードを APPN ネットワーク・ノードまたは APPN エンド・ノードとして構成できます (Communications Server for Linux でホストとの通信にのみ SNA を使用する場合には、Communications Server for Linux ノードをエンド・ノードまたはブランチ・ネットワーク・ノードとして構成できません)。

ローカル・ノードが APPN ネットワークに接続され、このノードが他のノードに対して APPN ルーティング・サービスを提供している場合には、ローカル・ノードを APPN ネットワーク・ノードとして構成します。他のノードがルーティング・サービスを提供している場合は、ローカル・ノードを APPN エンド・ノードとして構成します。

ノードを構成する前に、次の情報を収集してください。

- APPN サポートのタイプ (ネットワーク・ノード、ブランチ・ネットワーク・ノード、エンド・ノード)
- 制御点名 (別名が異なる場合は、別名も)。制御点名を確認するには、ネットワーク計画担当者に問い合わせてください。
- デフォルト・ノード ID (個別の通信リンクを構成する場合は、このデフォルトをオーバーライドできます。)

ノードを構成するには、「ノード (Node)」ウィンドウから次の手順を行います。

1. 「サービス (Services)」メニューから「ノード・パラメーターの構成 (Configure node parameters)」を選択するか、「ノード (Node)」ウィンドウの右上隅にある「ノード (Node)」ボックスをダブルクリックする。Communications Server for Linux が「ノード・パラメーター (Node Parameters)」ダイアログを表示します。
2. APPN サポートのレベル、制御点名、ノード ID (必要な場合) を指定する。
3. 「OK」ボタンをクリックする。これでノードが定義されます。このノードの定義時に、Communications Server for Linux は制御点と同じ名前が付いたデフォルトの LU を自動定義します。

入力した値を保管しないで終了するには、「取り消し (Cancel)」ボタンをクリックします。

接続の構成

他のノードと通信する Communications Server for Linux ノードの場合、少なくとも 1 つの隣接ノードとの接続を構成する必要があります。接続リンクは、従属トラフィックと独立トラフィックのいずれか、または両方をサポートするように構成できます。

コンピューターには、1 つ以上のリンク・プロトコル用にアダプター・カードを装着できます。接続の構成に入力する必要がある情報の大部分は、使用しているリンク・プロトコルに応じて決まります。Communications Server for Linux でサポートされているリンク・プロトコルのリストについては、22 ページの『インストール要件』を参照してください。

リンクを構成するには、ポートとリンク・ステーション (ほとんどの場合) を定義する必要があります。 Motif 管理プログラムを使用している場合は、ポート構成の一部として DLC (データ・リンク制御) が自動的に構成されます。 さらに、接続ネットワークの一部としてポートを定義することもできます。

構成する必要があるリンクは、使用しているネットワークが APPN ネットワークであるかどうか、および現在の構成しようとしている内容によって異なります。 必要な情報は、リンク・プロトコルによって異なり、また、リンクが従属トラフィックと独立トラフィックのいずれかまたはその両方をサポートするかによっても異なります。

この項では、上記の例として次のタイプのリンクの構成方法について説明します。

- SDLC 回線を使用するホスト・システムとの従属トラフィックをサポートするリンク。
- イーサネット・リンク・プロトコルを使用する APPN ネットワークに接続し、従属トラフィックと独立トラフィックの両方をサポートするリンク。 この例では、イーサネット・ポート上で接続ネットワークも定義します。
- APPN ネットワークへの Enterprise Extender リンク (ただし、Enterprise Extender リンクがサポートするのは、独立トラフィックのみであることにご注意ください)。

その他のリンク・プロトコルについては、「*Communications Server for Linux 管理ガイド*」、または Motif 管理プログラムのオンライン・ヘルプを参照してください。

従属トラフィックに対する SDLC リンクの構成

SDLC (同期データ・リンク制御) ポートについて、次の情報を収集する必要があります。

- SNA ポート名 (通常、デフォルト名を使用できます)。また、SDLC 装置番号も提供する必要があります。
- ノード始動時にポートを自動的に活動化するかどうか。
- 回線タイプ (交換発信、交換着信、または専用回線)。
- リンクの役割 (プライマリー、セカンダリー、ネゴシエーション可能、プライマリー・マルチドロップ、またはセカンダリー・マルチ PU)。
- ポーリング・アドレス (非プライマリー・ポート上の交換着信回線に対してのみ)。他のタイプのポートの場合、リンク・ステーション上のポーリング・アドレスを構成します。

SDLC リンク・ステーションについて、次の追加情報を収集する必要があります。

- 活動化の方法 (管理者による活動化、ノード始動時の活動化、要求に応じて行う活動化)
- サポートされているトラフィックのタイプ (この例の場合、従属のみ)。
- リモート・ノード・ロール (この例の場合、ホスト)。

SDLC リンクを構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ポートを構成する。

- a. ウィンドウの「接続 (Connectivity)」ペインを選択する。
- b. 「サービス (Services)」メニューの、「接続 (Connectivity)」サブメニューから、「新規ポート (New port)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックします)。
- c. その結果表示されるダイアログのオプション・メニューから、プロトコルのタイプを選択し、次にポートの定義を選択する。

「OK」ボタンをクリックすると、Communications Server for Linux は「SDLC ポート (SDLC Port)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力する。
- e. 「OK」ボタンをクリックする。これでポートが定義されます。

「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインに、ポートが表示されます。

2. ポートのリンク・ステーションを定義する。
 - a. 「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインで、リンク・ステーションの追加先のポートが選択されていることを確認する。
 - b. 「サービス (Services)」メニューの、「接続 (Connectivity)」サブメニューから、「新規リンク・ステーション (New link station)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックします)。
 - c. 「OK」ボタンをクリックする。

Communications Server for Linux が「SDLC リンク・ステーション (SDLC Link Station)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力する。
- e. 「OK」ボタンをクリックする。リンク・ステーションが定義されます。

リンク・ステーションは、「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインに表示されている、従属先ポートの下に表示されます。

従属トラフィックと独立トラフィックをサポートするイーサネット・リンクの構成

この例では、APPN ネットワークと接続し、従属トラフィックと独立トラフィックの両方をサポートするイーサネット・リンクの構成方法を説明します。また、イーサネット・ポート上で接続ネットワークも定義します。

イーサネット・ポートについて次の情報を収集する必要があります。

- SNA ポート名 (通常、デフォルト名を使用できます)。複数のイーサネット・ネットワーク・アダプター・カードがある場合は、イーサネット・カード番号も確認しておく必要があります。また、ローカル SAP (サービス・アクセス・ポイント) 番号も、指定する必要があります (Intel および OSA2 アダプターの場合は通常、04 です)。OSA-Express アダプターの場合、ローカル SAP 番号は、この Linux イメージの ethX インターフェースに対応する入出力装置アドレスの OSA/SF で定義された番号と一致する必要があります。
- ノード始動時にポートを自動的に活性化するかどうか。

- 接続ネットワーク名 (1 つの接続ネットワーク内では、すべてのポートについて同一の名前が使用されていなければなりません)。

イーサネット・リンク・ステーションについて、次の追加情報を収集する必要があります。

- 活動化の方法 (管理者による活動化、ノード始動時の活動化、要求に応じて行う活動化)
- サポートされているトラフィックのタイプ (従属トラフィック、および独立トラフィック)
- リモート・ノードの制御点名 (LEN ノードの場合のみ必要)
- リモート・ノードのタイプ (ネットワーク・ノード、エンド・ノード、または自動検出)
- リモート・ノード・ロール (ダウンストリーム SNA ゲートウェイ、またはパススルー DLUR など)
- 選択リンク・ステーションを構成するには、リモート・ステーションの MAC (メディア・アクセス制御) アドレスと、SAP 番号 (通常は 04) が必要です。アドレス情報がない場合に、「活動化 (Activation)」フィールドに、「管理者 (By administrator)」を指定すると、リンク・ステーションは非選択 listen リンク・ステーションになります。

イーサネット・リンクを構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ポートを構成する。
 - a. ウィンドウの「接続 (Connectivity)」ペインを選択する。
 - b. 「サービス (Services)」メニューの、「接続 (Connectivity)」サブメニューから、「新規ポート (New port)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックします)。
 - c. その結果表示されるダイアログのオプション・メニューから、プロトコルのタイプを選択し、次にポートの定義を選択する。

「OK」ボタンをクリックすると、Communications Server for Linux は「イーサネット SAP (Ethernet SAP)」ダイアログを表示します。
 - d. ダイアログのフィールドに適切な値を入力する。
 - e. 「OK」ボタンをクリックする。これでポートが定義されます。

「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインに、ポートが表示されます。
2. ポートのリンク・ステーションを定義する。
 - a. 「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインで、リンク・ステーションの追加先のポートが選択されていることを確認する。
 - b. 「サービス (Services)」メニューの、「接続 (Connectivity)」サブメニューから、「新規リンク・ステーション (New link station)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックします)。
 - c. 「OK」ボタンをクリックする。

Communications Server for Linux が「イーサネット・リンク・ステーション (Ethernet Link Station)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力する。
- e. 「OK」 ボタンをクリックする。リンク・ステーションが定義されます。

リンク・ステーションは、「ノード (Node)」 ウィンドウの「接続 (Connectivity)」 ペインに表示されている、従属先ポートの下に表示されます。

Enterprise Extender リンクの構成

この例では、APPN ネットワークへの、 Enterprise Extender リンクの構成方法を説明します。 Enterprise Extender リンクがサポートするのは、独立 LU トラフィックのみであることに注意してください。

Enterprise Extender ポートについて、次の情報を収集する必要があります。

- SNA ポート名 (通常、デフォルト名を使用できます)。 IP を実行するネットワーク・アダプター・カードが複数ある場合は、使用したい IP インターフェース名 (eth0 など) も、提供する必要があります。
- ノード始動時にポートを自動的に活動化するかどうか。

Enterprise Extender リンク・ステーションについて、次の追加情報を収集する必要があります。

- 活動化の方法 (管理者による活動化、ノード始動時の活動化、要求に応じて行う活動化)
- リモート・ノードのタイプ (ネットワーク・ノード、エンド・ノード、または自動検出)
- 選択リンク・ステーションを構成するには、リモート・ステーションの IP ホスト名または IP アドレスが必要です。この情報を提供せず、「活動化 (Activation)」 フィールドで「管理者 (By administrator)」を指定すると、リンク・ステーションは非選択 listen リンク・ステーションになります。

Enterprise Extender リンクを構成するには、「ノード (Node)」 ウィンドウで次の手順を行います。

1. ポートを構成する。
 - a. ウィンドウの「接続 (Connectivity)」 ペインを選択する。
 - b. 「サービス (Services)」 メニューの、「接続 (Connectivity)」 サブメニューから、「新規ポート (New port)」を選択する (あるいは、ボタン・バーの「新規 (New)」 ボタンをクリックします)。
 - c. その結果表示されるダイアログのオプション・メニューから、プロトコルのタイプを選択し、次にポートの定義を選択する。

「OK」 ボタンをクリックすると、Communications Server for Linux は「IP ポート (IP Port)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力する。
- e. 「OK」 ボタンをクリックする。これでポートが定義されます。

「ノード (Node)」 ウィンドウの「接続 (Connectivity)」 ペインに、ポートが表示されます。

2. ポートのリンク・ステーションを定義する。
 - a. 「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインで、リンク・ステーションの追加先のポートが選択されていることを確認する。
 - b. 「サービス (Services)」メニューの、「接続 (Connectivity)」サブメニューから、「新規リンク・ステーション (New link station)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックします)。
 - c. 「OK」ボタンをクリックする。

Communications Server for Linux が「IP リンク・ステーション (IP Link Station)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力する。
- e. 「OK」ボタンをクリックする。リンク・ステーションが定義されます。

リンク・ステーションは、「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインに表示されている、従属先ポートの下に表示されます。

タイプ 0-3 LU の構成

タイプ 0-3 LU を使用するユーザー・アプリケーションをサポートするには、従属 LU を構成する必要があります。そうする前に、次の構成作業を行ってください。

- 80 ページの『ノードの構成』の説明に従って、ノードを構成する。
- 81 ページの『接続の構成』に記載したように、リンクを構成して従属 LU トラフィックをサポートする。

SNA ゲートウェイを使用する別ノードへのアップストリーム・リンクがある場合、または DLUR を使用している場合、ホストに対する直接リンクの構成は不要です。詳細は、98 ページの『SNA ゲートウェイの構成』と 101 ページの『DLUR の構成』を参照してください。

タイプ 0-3 の従属 LU を構成して、ホスト・システムとの通信をサポートする必要があります。当セクションの内容を使用して、LUA、DLUR、または PU 集信をサポートする LU を定義します。また、LU の範囲を定義して、単一の操作で同一タイプの複数 LU を構成することもできます。

さらに、必要に応じて使用される LU のプールを定義できます。これを行うには、ある LU の定義時にそれをプールに割り当てるか、または以前に定義済みの LU をプールに割り当てるかのいずれかです。

タイプ 0-3 LU の定義

3270 LU を構成する前に、次の情報を収集してください。

- LU 名 (これはローカル ID で、ホスト構成と一致する必要はありません。)
- LU 番号 (または、一連の LU の数)
- LU タイプ (3270 ディスプレイ・モデルまたは 3270 プリンター)
- プール名 (LU をプールに追加する場合)

既に定義されているリンク・ステーションについて、タイプ 0 から 3 の LU を構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ウィンドウの「接続 (Connectivity)」ペインから、ホスト接続リンク・ステーションを選択する。
2. 「新規 (New)」ボタンをクリックする。
3. 表示されるダイアログで、LU タイプ (「新規 3270 ディスプレイ LU (New 3270 display LU)」、または「新規 3270 プリンター LU (New 3270 printer LU)」) を選択する。

この項目を選択して「OK」をクリックすると、Communications Server for Linux は「LU タイプ 0 から 3 (LU Type 0-3)」ダイアログを表示します。

4. ダイアログのフィールドに適切な値を入力する。
5. 「OK」をクリックする。LU が定義されます。

LU は、「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインに表示されている、ホスト接続リンク・ステーションの下に表示されます。

LU プールの定義

LU タイプ 0-3 では、LU プールを定義することにより、ユーザー構成が単純化され、ホスト・セッションの確立を非常に柔軟に行うことができます。たとえば、複数の LU を 1 つの LU プールに定義しておくこと、この LU プールを使用して複数ユーザーを構成できます。これにより、ユーザーのセッションの構成が容易になり、またどのセッションでもプール内の LU をすべて使用できるようになります。

注: ユーザーのセッションを特定の LU または LU プールのいずれにも割り当てることができます。

- ユーザーのセッションをプール内の特定の LU に割り当てられる場合、この LU が使用可能であればセッションで使用されます。そうでない場合、プール内のその他の空いている LU が、特定の LU の代わりにセッションに割り当てられているかのように、セッションで使用されます。
- ユーザーに特定の LU のみを使用させ、LU が既に使用中のときはユーザーのセッションを確立できないようにする場合、その LU がプール内にあることを確認してください。

ローカル Communications Server for Linux ノードの LU プールは、「LU プール (LU Pools)」ウィンドウで確認できます。このウィンドウには、ローカル・システムで構成されている LU プールのリストが表示されます。また、LU プールに追加する LU を選択することもできます。

プールに追加できる LU タイプを次に示します (ただし、1 つのプールには異なる種類の LU を混在させないでください)。

- 3270 ディスプレイ LU
- 無制限 LU

LU は、プールに追加する前に必ずローカル・ノードで定義してください。

LU プールを構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. 「ウィンドウ (Windows)」メニューから、「LU プール (LU Pools)」を選択する。

タイプ 0-3 LU の構成

Communications Server for Linux が「LU プール (LU Pools)」ウィンドウを表示します。

2. 「新規 (New)」ボタンをクリックする。

Communications Server for Linux が「LU プール構成 (LU Pool Configuration)」ダイアログを表示します。

右側のボックスに、まだプールに割り振られていない LU のリストが表示されます。このリストに表示されている LU はすべて、新しいプールに割り振ることができます。

3. プールに追加する LU (1 つ以上) を選択してから、「新規 (New)」ボタンをクリックする。選択された LU が左側のボックスへ移動します。

左側のボックスから LU を除去するには、LU を選択して「除去 (Remove)」ボタンをクリックします。

4. 「OK」をクリックする。LU プールが定義されます。

左側のボックスに表示されている LU がすべて LU プールに追加されます。

定義されたプールが「LU プール (LU Pools)」ウィンドウに表示されます。

APPC 通信の構成

APPC アプリケーションと CPI-C アプリケーションを使用する場合には、まず最初に APPC を構成する必要があります。APPC アプリケーションは、ホストまたは対等コンピューター上の、他の APPC アプリケーションまたは CPI-C アプリケーションと、指定されたモードで通信を行う際に、ノードの LU タイプ 6.2 リソースを使用します。

APPC 通信を構成する前に、次の構成を行ってください。

1. 80 ページの『ノードの構成』の説明に従って、ノードを構成する。
2. 81 ページの『接続の構成』の説明に従って、接続を構成する。

これ以降の構成手順は、構成でサポートされているトラフィックの種類 (従属トラフィックと独立トラフィックのいずれか、または両方) によって異なります。

独立 APPC

独立 APPC では独立 LU が使用されます。各 LU-LU セッションでは、ローカル LU とパートナー LU が使用されます。

ローカル LU の場合、ノード制御点に関連付けられている事前定義のデフォルト LU を使用できます。あるいは新しいローカル LU を構成することもできます。

Communications Server for Linux ノードが APPN ネットワーク内のエンド・ノードまたはネットワーク・ノードの場合、APPN によりパートナー LU が動的に検出されるため、パートナー LU を構成する必要はありません。ただし、APPN ネットワーク以外のネットワークの場合や LEN ノードの場合は、パートナー LU を定義する必要があります。この場合、まずパートナー LU があるリモート・ノードを構成してから、このリモート・ノードのパートナー LU を定義してください。

従属 APPC

リモート・ノードで独立 LU 6.2 がサポートされていない場合は、従属トラフィックをサポートするように構成します。従属 APPC では、ローカル LU を構成する必要があります。

アプリケーションで CPI-C が使用される場合には、APPC の構成後にさらに、CPI-C を構成しなければならないことがあります (95 ページの『CPI 通信のための構成』を参照してください)。CPI-C アプリケーションは、ホストまたは対等コンピュータ上の、他の APPC アプリケーションまたは CPI-C アプリケーションと通信を行う際に、ノードの LU タイプ 6.2 を使用します。CPI-C アプリケーションと APPC アプリケーションには、同じリソースを定義します。また、Communications Server for Linux コンピューターの TP が始動 TP、つまりソース TP (会話を開始する TP) の場合、TP に対して 1 つ以上のサイド情報エントリーを定義しなければならないことがあります。この手順については 95 ページの『CPI 通信のための構成』で説明します。各エントリーには、パートナー TP の情報、パートナー TP にアクセスするために使用される LU やその他のリソースの情報、必須セキュリティ情報が示されます。

この節では、単純な APPN ネットワーク (独立 LU 6.2 を使用) の構成方法について説明します。このネットワークは、『単純な APPN ネットワークの構成』に説明があるように、1 つのネットワーク・ノード、1 つのエンド・ノード、1 つの LEN ノードで構成されています。(この例では、2 つのノード間の CP-CP セッションの状況情報の取得方法についても説明します。)

また、この節では、94 ページの『従属 APPC の構成』で説明されている、従属 APPC 通信の構成方法についても説明します。

どちらの例でも、APPC セッションで、標準モードとサービス・クラス (COS) が使用されていることを前提としています。

モード、セキュリティ、始動可能な (ターゲット) TP など、その他の APPC 情報の構成方法の詳細については、「*Communications Server for Linux 管理ガイド*」を参照してください。

単純な APPN ネットワークの構成

最も単純な APPN ネットワークの構成は、APPN ネットワーク・ノードと APPN エンド・ノードの 2 つのノードのみで構成できます。ネットワーク・ノードは、エンド・ノード用セッションのルーティングを処理します。

ネットワーク・ノードの構成

この例では、制御点 LU と標準モードが使用されていること、および LAN リンク・タイプ (トークンリング、イーサネット) が使用されていることを前提としています。この場合にネットワーク・ノードを構成する手順は次のとおりです。

1. 80 ページの『ノードの構成』の説明に従って、ノードを構成する。「APPN サポート (APPN support)」フィールドでは、値「ネットワーク・ノード (Network node)」を選択します。制御点名をメモしておいてください。
2. 81 ページの『接続の構成』の説明に従って、接続を構成する。従属トラフィックをサポートするリンクを構成します。

隣接するエンド・ノードからこのネットワーク・ノードへ通信するには、ネットワーク・ノードのポートの MAC アドレスと SAP 番号が必要です。Communications Server for Linux ノードの MAC アドレスを取得する手順は次のとおりです。

1. 「ノード (Node)」ウィンドウでポートを選択する。
2. 「開始 (Start)」ボタンをクリックする。ポートが始動します。
3. ポートの状況情報を取得するため、「状況 (Status)」ボタンをクリックする。「ポートの状況 (Port Status)」ダイアログに、MAC アドレスと SAP 番号が表示されます。
4. MAC アドレスと SAP 番号をメモしておく。これらの値は、後でエンド・ノードのリンク・ステーション構成ダイアログで入力する必要があるためです。

エンド・ノードの構成

この例では、制御点 LU と標準モードが使用されていること、および LAN リンク・タイプ (トークンリング、イーサネット) が使用されていることを前提としています。この場合にネットワーク・ノードを構成する手順は次のとおりです。

1. 80 ページの『ノードの構成』の説明に従って、ノードを構成する。「APPN サポート (APPN support)」フィールドへの値として、「エンド・ノード (End node)」を選択します。
2. 81 ページの『接続の構成』の説明に従って、接続を構成する。従属トラフィックをサポートするリンクを構成して、リンク・ステーション用に次の情報を入力します。
 - 「リモート・ノード (Remote node)」フィールドへの値として、ネットワーク・ノードの名前 (89 ページの『ネットワーク・ノードの構成』を参照) を入力する。
 - リンク・ステーション構成ダイアログの、「接続情報 (Contact Information)」ペインに、ネットワーク・ノードのポートの MAC アドレスと SAP 番号を入力する。

APPN ネットワークでは、隣接するネットワーク・ノードに接続する 1 つのリンク・ステーションを使用して、ネットワーク内の他のリモート・ノードと通信できます。このため、リモート・ノードごとに個別のリンク・ステーションを構成する必要はありません。

2 つのノード間の接続の確認

この例では、89 ページの『ネットワーク・ノードの構成』と『エンド・ノードの構成』の説明に従って、すでにネットワーク・ノードとエンド・ノードが構成済みであることを前提としています。エンド・ノードから次の手順を行います。

1. 「ノード (Node)」ウィンドウで、隣接するネットワーク・ノードに接続するリンク・ステーションを選択する。
2. 「開始 (Start)」ボタンをクリックする。リンク・ステーションが始動します。

リンク・ステーションの始動時に、2 つのノード間での CP-CP セッションも自動的に確立されます。これらのセッションは、「ノード (Node)」ウィンドウの、「独立型ローカル LU (Independent Local LUs)」ペインに表示されます。

3. セッションの状況情報を取得するには、「ノード (Node)」ウィンドウでセッションを選択してから、「状況 (Status)」ボタンをクリックする。

独立 APPC LU の構成

多くの場合、アプリケーションではローカル・ノードの制御点 LU を使用できます。この制御点 LU は、ノードの構成時に自動的に定義されます。これはデフォルトの LU です。アプリケーションでは、特定の LU が指定されない場合には、このデフォルト LU を使用できます。アプリケーションでデフォルト LU を使用する場合は、ローカル LU を定義する必要はありません。使用している APPC アプリケーションの資料を参照するか、アプリケーション・プログラマーに問い合わせてください。

独立 LU 6.2 を構成するには、次の情報を収集する必要があります。

- ローカル LU 名
- ローカル LU の別名 (この LU がサポートする TP で別名が使用されている場合)

独立型ローカル LU を構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ウィンドウの「独立型ローカル LU (Independent Local LUs)」ペインを選択する。
2. 「サービス (Services)」メニューの「APPC」サブメニューから、「新しい独立型ローカル LU (New independent local LU)」を選択する (あるいは、「新規 (New)」ボタンをクリックします)。

Communications Server for Linux が「ローカル LU (Local LU)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力する。
4. 「OK」ボタンをクリックする。ローカル LU が定義されます。「ノード (Node)」ウィンドウの「独立型ローカル LU (Independent Local LUs)」ペインに、独立型ローカル LU が表示されます。

LEN ノードのパートナー LU の構成

次の状況では、リモート・ノード (およびこのリモート・ノードのパートナー LU) を定義する必要があります。

- ローカル・ノードが LEN ノードの場合、すべてのリモート・ノードとローカル・ノードが、APPC を使用して通信を行うリモート・ノードのパートナー LU を定義しなければなりません。LEN ノードからはパートナー LU を動的に検出することはできませんが、リモート・ノードの定義により、LEN ノードでこのような LU の検出が可能になります。
- ローカル・ノードが APPN ネットワークの一部ではない場合 (たとえば、ネットワーク・ノード・サーバーを使用せずに、2 つのエンド・ノードを直接接続している場合) は、LU を動的に検出できません。この場合には、各パートナー LU を個別に構成する必要があります。
- リモート・ノードが LEN ノードで、ローカル・ノードが、この LEN ノードのネットワーク・ノード・サーバーとして動作するネットワーク・ノードの場合、LEN ノード (およびそのパートナー LU) をネットワーク・ノード・サーバーのリモート・ノードとして定義しなければなりません。このように定義することで、APPN ネットワークのその他のノードから、LEN ノードの LU を検出できるようになります。

- リモート・ノードが別の APPN ネットワークにある場合、このリモート・ノードは動的に検出できないため、定義しておく必要があります。

ローカル・ノードとリモート・ノードの両方が、同じ APPN ネットワークに接続している場合は、パートナー LU を定義しないでください。

リモート・ノード定義を追加すると、リモート・ノードと同じ名前のパートナー LU が自動的に追加されます。これは、リモート・ノードの制御点 LU となります。このパートナー LU をアプリケーションで使用する場合、その他のパートナー LU を追加する必要はありませんが、このパートナー LU に LU 別名を追加できます。別名を追加するには、パートナー LU をダブルクリックし、「パートナー LU の構成 (Partner LU Configuration)」ダイアログに別名を入力します。

アプリケーションで、LU 別名を使用してパートナー LU を参照する場合には、パートナー LU 別名の定義を追加する必要があります。

ローカル・ノードまたはリモート・ノードのいずれかが LEN ノードの場合、LEN ノードでは LU を動的に検出できないため、パートナー LU をリモート・ノードの子として定義しなければなりません。アプリケーションで、リモート・ノードの制御点 LU をそのパートナー LU として使用する場合、リモート・ノードの定義時に自動的に制御点 LU を定義します。

Motif 管理プログラムでは、パートナー LU 別名の追加 (93 ページの『パートナー LU 別名の定義』を参照)、特定のリモート・ノードのパートナー LU 定義の追加 (93 ページの『リモート・ノードでのパートナー LU の定義』を参照)、ワイルドカードを使用した、複数のパートナー LU の定義 (93 ページの『ワイルドカードを使用した複数のパートナー LU の定義』を参照) を行うことができます。

リモート・ノードの定義: リモート・ノードを構成する前に、次の情報を収集する必要があります。

- ノードの SNA ネットワークの完全修飾名

リモート・ノードを構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ウィンドウの「リモート・システム (Remote Systems)」ペインを選択する。
2. 「サービス (Services)」メニューの「APPC」サブメニューから、「新規リモート・ノード (New remote node)」を選択する (あるいは、ボタン・バーの「新規 (New)」をクリックし、次に「リモート・ノードの定義 (Define remote node)」を選択します)。

Communications Server for Linux が「リモート・ノード構成 (Remote Node Configuration)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力する。
4. 「OK」ボタンをクリックする。リモート・ノードが定義されます。「ノード (Node)」ウィンドウの「リモート・システム (Remote Systems)」ペインに、リモート・ノードが表示されます。

リモート・システムを定義すると、Communications Server for Linux はリモート・ノードの制御点 LU をローカル・ノードのパートナー LU として自動的に定義します。

パートナー LU 別名の定義: パートナー LU の別名を定義するには、次の情報を収集する必要があります。

- パートナー LU の完全修飾名 (SNA ネットワーク名と LU 名)
- ローカル TP が使用するパートナー LU 別名

パートナー LU の別名を追加するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ウィンドウの「リモート・システム (Remote Systems)」ペインを選択する。
2. 「サービス (Services)」メニューから、「APPC」、「新規パートナー LU (New partner LUs)」、「パートナー LU の別名 (Partner LU alias)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックし、次に「パートナー LU 別名の定義 (Define partner LU alias)」を選択します)。

Communications Server for Linux が「パートナー LU エイリアス構成 (Partner LU Alias Configuration)」ダイアログを表示します。

3. ダイアログにパートナー LU の名前と別名を入力する。
4. 「OK」ボタンをクリックする。パートナー LU の別名が定義されます。「ノード (Node)」ウィンドウの「リモート・システム (Remote Systems)」ペインに、(ネットワーク定義の一部として) パートナー LU の別名が表示されます。

リモート・ノードでのパートナー LU の定義: 特定のリモート・ノードのパートナー LU を定義するには、次の情報を収集する必要があります。

- パートナーの完全修飾 LU 名
- パートナー LU の別名 (ローカル TP で別名が使用されている場合)
- パートナー LU のディレクトリー情報を含むノードの完全修飾名

特定のリモート・ノードのパートナー LU 定義を追加するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. リモート・ノードを選択する。
2. 「サービス (Services)」メニューから、「APPC」、「新規パートナー LU (New partner LUs)」、「リモート・ノードのパートナー LU (Partner LU on remote node)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックし、次に、「リモート・ノードのパートナー LU の定義 (Define partner LU on remote node)」を選択します)。

Communications Server for Linux が「パートナー LU 構成 (Partner LU Configuration)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力する。
4. 「OK」ボタンをクリックする。パートナー LU が定義されます。パートナー LU の別名が、「ノード (Node)」ウィンドウの「リモート・システム (Remote Systems)」ペインで、この LU が属するリモート・システムの下に表示されます。

ワイルドカードを使用した複数のパートナー LU の定義: 1 つのリモート・ノードにあり、名前が同じ文字で始まる複数のパートナー LU を構成する場合には、ワイルドカードを使用できます。ワイルドカードを使用すれば、パートナー LU を個別に構成する必要がなくなります。

ワイルドカードを使用して複数のパートナー LU を定義する際には、次の情報を入力する必要があります。

- ワイルドカード・パートナー LU 名。ワイルドカード・パートナー LU 名は、定義する複数のパートナー LU の完全修飾名に一致する、1 から 8 文字のタイプ A の EBCDIC ストリング 2 つからなります。

最初のストリングには、パートナー LU の SNA ネットワーク名と厳密に一致する完全な SNA ネットワーク名か、ネットワーク名の先頭の文字に一致するワイルドカード接頭部のいずれかを使用できます。ネットワーク名にワイルドカード接頭部を入力する場合は、2 番目のストリングをブランクにしてください。

最初のストリングに完全な SNA ネットワーク名を入力した場合は、2 番目の値も入力できます。(この場合、最初のストリングに有効な SNA ネットワーク名を指定していないと、2 番目のストリングに入力できません。) 2 番目のストリングは、パートナー LU の完全修飾名の後半の部分の先頭に一致する、ワイルドカード接頭部として処理されます。

- パートナー LU があるノードの名前。

複数のパートナー LU を追加するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. 定義するパートナー LU があるリモート・ノードを選択する。
2. 「サービス (Services)」メニューから、「APPC」、「新規パートナー LU (New partner LUs)」、「リモート・ノードのワイルドカード・パートナー (Wildcard partner on remote node)」を選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックし、次に、「リモート・ノードのワイルドカード・パートナー LU の定義 (Define wildcard partner LUs on remote node)」を選択します)。

Communications Server for Linux が「ワイルドカード・パートナー LU 構成 (Wildcard Partner LU Configuration)」ダイアログを表示します。

3. ダイアログのフィールドに、適切な情報を入力する。
4. 「OK」ボタンをクリックする。パートナー LU が定義されます。パートナー LU が、「ノード (Node)」ウィンドウの「リモート・システム (Remote Systems)」ペインで、この LU が属するリモート・ノードの下に表示されます。

従属 APPC の構成

従属 LU 6.2 を構成するには、次の情報を収集する必要があります。

- ローカル LU 名
- ローカル LU の別名 (この LU がサポートする TP で別名が使用されている場合)
- ホストへの接続を提供するリンク・ステーションの名前
- LU 番号
- LU を従属 LU 6.2 のデフォルト・プールに割り当てる必要があるかどうか

APPC アプリケーションまたは CPI-C アプリケーションで使用するために、タイプ 6.2 の従属 LU を構成している場合は、これらの LU をデフォルト・プールのメン

バーとして定義することがあります。特定のローカル LU を指定しないアプリケーションには、デフォルト LU として定義されている LU のプールから、未使用の LU が割り当てられます。

従属型ローカル LU を構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ウィンドウの「接続 (Connectivity)」ペインでリンク・ステーションを選択する。
2. 「サービス (Services)」メニューの「APPC」サブメニューから、「新しい従属型ローカル LU (New dependent Local LU)」を選択する (あるいは、ボタンの「新規 (New)」をクリックし、次に「新しい従属型ローカル LU (New dependent local LU)」を選択します)。

Communications Server for Linux が「ローカル LU (Local LU)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力する。
4. 「OK」ボタンをクリックする。ローカル LU が定義されます。従属型ローカル LU は、「接続 (Connectivity)」ペインでこの LU が属するリンク・ステーションの下に表示されます。

CPI 通信のための構成

サポートしている CPI-C アプリケーションで CPI-C 記号宛先名を使用する場合には、CPI-C サイド情報を定義する必要があります。サイド情報により、記号宛先名と、会話のパートナー TP、パートナー LU、モード、セキュリティーに関する情報が関連付けられます。

CPI-C 用の記号宛先名を判別するには、アプリケーション開発者に相談してください (サード・パーティーのアプリケーションの場合は、その製品の資料を参照してください)。

CPI-C サイド情報を構成する前に、次の情報を収集する必要があります。

- TP により使用される記号宛先名
- パートナー TP 名
- パートナー LU 名または別名
- モード名

CPI-C サイド情報を構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. 「サービス (Services)」メニューの「APPC」サブメニューから、「CPI-C」を選択する。

Communications Server for Linux が「CPI-C 宛先名 (CPI-C Destination Names)」ウィンドウを表示します。

2. 「新規 (New)」ボタンをクリックする。

Communications Server for Linux が「CPI-C 宛先構成 (CPI-C Destination Configuration)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力する。
4. 「OK」ボタンをクリックする。CPI-C サイド情報が定義されます。

LUA の構成

LUA API は、LU タイプ 0 から 3 を使用して、ホスト・コンピューターと通信を行うアプリケーションに使用できます。(LUA API の詳細については、「*Communications Server for Linux LUA プログラマーズ・ガイド*」を参照してください。)

LUA を構成する前に、次の構成を行ってください。

1. 80 ページの『ノードの構成』の説明に従って、ノードを構成する。
2. 81 ページの『接続の構成』の説明に従って、従属トラフィックの接続を構成する。(アップストリーム SNA ゲートウェイまたは DLUR を使用している場合は、ホストへの直接リンクではなく、アップストリーム・ノードへのリンクを構成してください。)

LUA を構成するには、次の情報を収集する必要があります。

- LU 名または LU プール名。
- 各 LU の LU 番号。LU 番号は、ホストで構成されている LU 番号と一致していなければなりません。

LUA を構成するため、次の手順に従って LU を定義します。

1. 「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインで、ホストへのリンク・ステーションを選択する。
2. 「新規 (New)」ボタンをクリックする。
3. 表示されるダイアログから、「LUA の新しい LU (New LU for LUA)」を選択する。
4. ダイアログのフィールドに適切な値を入力する。LU タイプとして「制限なし (Unrestricted)」を選択します。
5. 「OK」ボタンをクリックする。LU は、「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインに表示されている、ホスト接続リンク・ステーションの下に表示されます。
6. LU プールを使用する場合は、87 ページの『LU プールの定義』の説明に従って LU プールを定義する。

LU プールの定義

LU プールを定義することにより、ユーザー構成が単純化され、ホスト・セッションの確立を非常に柔軟に行うことができます。たとえば、複数の LU を 1 つの LU プールに定義しておくこと、この LU プールを使用して複数の LUA アプリケーションを構成できます。これにより、アプリケーションの構成が容易になり、またアプリケーションでプール内の LU をすべて使用できるようになります。

注: ユーザーのセッションを特定の LU または LU プールのいずれにも割り当てることができます。

- ユーザーのセッションをプール内の特定の LU に割り当てる場合、この LU が使用可能であればセッションで使用されます。そうでない場合、プール内のその他の空いている LU が、特定の LU の代わりにセッションに割り当てられているかのように、セッションで使用されます。
- ユーザーに特定の LU のみを使用させ、LU が既に使用中のときはユーザーのセッションを確立できないようにする場合、その LU がプール内にないことを確認してください。

ローカル Communications Server for Linux ノードの LU プールは、「LU プール (LU Pools)」ウィンドウで確認できます。このウィンドウには、ローカル・システムで構成されている LU プールのリストが表示されます。また、LU プールに追加する LU を選択することもできます。

3270 が使用するプールに追加できる LU タイプを次に示します (ただし、1 つのプールには異なる種類の LU を混在させないでください)。

- 3270 ディスプレイ LU
- 無制限 LU

LU は、プールに追加する前に必ずローカル・ノードで定義してください。

LU プールを構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. 「**ウィンドウ (Windows)**」メニューから、「**LU プール (LU Pools)**」を選択する。

Communications Server for Linux が「LU プール (LU Pools)」ウィンドウを表示します。

2. 「**新規 (New)**」ボタンをクリックする。

Communications Server for Linux が「LU プール構成 (LU Pool Configuration)」ダイアログを表示します。

右側のボックスに、まだプールに割り振られていない LU のリストが表示されます。制限なし (Unrestricted) タイプのこれらの LU はすべて、LUA の新しいプールに含めることができます。

3. プールに追加する LU (1 つ以上) を選択してから、「**新規 (New)**」ボタンをクリックする。選択された LU が左側のボックスへ移動します。

左側のボックスから LU を除去するには、LU を選択して「**除去 (Remove)**」ボタンをクリックします。

4. 「**OK**」をクリックする。LU プールが定義されます。

左側のボックスに表示されている LU がすべて LU プールに追加されます。

定義されたプールが「LU プール (LU Pools)」ウィンドウに表示されます。

SNA ゲートウェイの構成

Communications Server for Linux には、ホスト・コンピューターへの直接アクセス機能のみでなく、SNA ゲートウェイ機能が組み込まれています。この機能により、他のコンピューターが、Communications Server for Linux ノードを介してホスト・コンピューターに接続できます。各コンピューターからホストへ個別に接続する必要はありません。

図7 に SNA ゲートウェイの機能を示します。

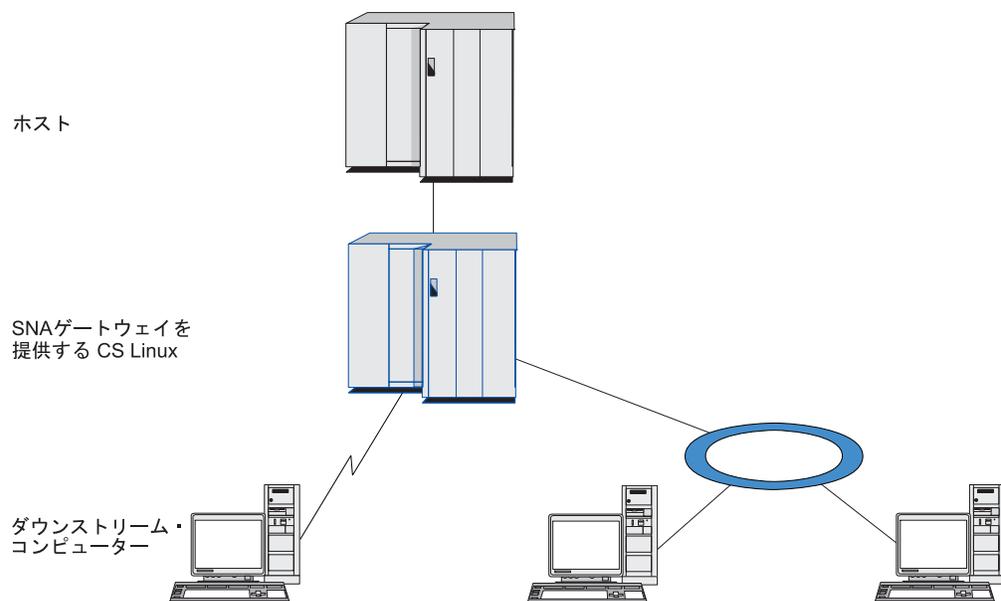


図7. SNA ゲートウェイ

ダウンストリーム・コンピューターには、従属 LU をサポートする、タイプ 2.0 または 2.1 の SNA PU がインストールされていなければなりません。たとえば、別の Communications Server for Linux コンピューターや、Communications Server for Windows NT[®] が稼働する PC を、ダウンストリーム・コンピューターとして使用できます。

ローカルの Communications Server for Linux ノードで SNA ゲートウェイ機能を使用すると、ホストとダウンストリーム・コンピューター間で送受信されるデータはすべて、ローカル・ノードを介して転送されます。これにより、ダウンストリーム・コンピューターは、直接リンクを要求せずに、Communications Server for Linux、またはその他のダウンストリーム・コンピューターとホスト接続を共有できます。たとえば、複数のダウンストリーム・コンピューターを、ローカル・トークンリング・ネットワークを介して Communications Server for Linux に接続するように設定すると、これらのコンピューターはすべて、同一の長距離専用回線を介して、Communications Server for Linux からホストにアクセスできるようになります。

SNA ゲートウェイを使用すると、ダウンストリーム・コンピューターと、それらのコンピューターとの通信リンクを定義する必要がなくなるため、ホスト側での構成が単純化されます。ホストの構成では、Communications Server for Linux コンピュ

ーターとそのホスト通信リンクのみを含めます。ダウンストリーム・コンピューターの LU は、Communications Server for Linux コンピューターのリソースの一部として構成されます。ホスト・コンピューターでは、SNA ゲートウェイの使用を認識しません。

SNA ゲートウェイを構成する前に次の構成作業を行ってください。

- 80 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義します。
- 81 ページの『接続の構成』の説明に従って、ローカル・ノードとホスト間の、従属トラフィック用のポートとリンク・ステーションを構成します。また、ローカル・ノードとダウンストリーム・ノード間の、従属トラフィック用のポートとリンク・ステーションを構成します。事前に定義されていないダウンストリーム LU をサポートする必要がある場合は、暗黙の PU とダウンストリーム LU をサポートするように、ポートのテンプレートを定義できます (『暗黙のダウンストリーム LU のサポート』を参照してください)。
- ホストとの通信に使用する、ローカル・ノードの LU (アップストリーム LU) を定義します。アップストリーム LU は、LU タイプが 制限なし (Unrestricted) (不明 (unknown)) の、LU タイプ 0 から 3 として定義してください。(ダウンストリーム・ノードの LU にはどの LU タイプでも定義できます。)
- LU プールを使用する場合は、96 ページの『LU プールの定義』の説明に従って LU プールを定義します。

暗黙のダウンストリーム LU のサポート

Communications Server for Linux に対して事前に定義されていないダウンストリーム LU をサポートするには、暗黙のダウンストリーム PU と LU に対して、ポートのテンプレートを定義できます (基本的なポート構成については、81 ページの『接続の構成』を参照してください)。これらのテンプレートにより、ダウンストリーム LU がサポートされます。この際、ダウンストリーム・ノードのすべての LU をサポートするように、ローカル・ノードに LU を 1 つ構成する必要はありません。

SNA ゲートウェイに対してダウンストリーム LU を構成する前に、次の情報を収集する必要があります。

- ダウンストリーム LU をサポートする LU の数の範囲
- ホスト LU 名

暗黙のダウンストリーム LU についてテンプレートを定義する手順は、次のとおりです。

1. ポートをすでに構成している場合は、「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインの、ポート定義をダブルクリックする。Communications Server for Linux が「ポート構成 (port configuration)」ダイアログを表示します。

ポートをまだ構成していない場合は、ポートを構成します。

- a. 「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインを選択する。
- b. 「新規 (New)」ボタンをクリックする。
- c. 表示されたダイアログから、ポートを定義してリンク・プロトコルのタイプを選択する。

Communications Server for Linux が「ポート構成 (port configuration)」ダイアログを表示します。

- d. 基本ポート・パラメーターを入力する。詳細については、81 ページの『接続の構成』を参照してください。
2. ダイアログの下部にある、「**詳細設定 (Advanced)**」ボタンをクリックする。

Communications Server for Linux が「ポート・パラメーター (Port Parameters)」ダイアログを表示します。下のペインに表示されている設定が、ダウンストリーム LU テンプレートに反映されます。

3. 「**暗黙 PU アクセスをサポートするようダウンストリーム LU を構成 (Configure downstream LUs for implicit PU access)**」オプションを選択する。
4. 「**OK**」をクリックする。

Communications Server for Linux が「ダウンストリーム LU テンプレート構成 (Downstream LU Template Configuration)」ダイアログを表示します。

5. ダイアログのフィールドに適切な値を入力する。
6. 「**OK**」をクリックする。暗黙のダウンストリーム LU テンプレートが定義されます。

ダウンストリーム LU の定義

SNA ゲートウェイに対してダウンストリーム LU を構成する前に、次の情報を収集する必要があります。

- 各ダウンストリーム LU の LU 名 (これはローカル ID で、ダウンストリーム・システムの構成と一致する必要はありません。)
- 各ダウンストリーム LU の LU 番号
- ダウンストリーム・ノードへのリンク・ステーション
- アップストリーム LU 名 (ホスト LU の場合)

SNA ゲートウェイに対してダウンストリーム LU を構成する手順は次のとおりです。

1. 「**ノード (Node)**」ウィンドウの「**接続 (Connectivity)**」ペインから、ダウンストリーム・ノードへのリンク・ステーションを選択する。
2. 「**新規 (New)**」ボタンをクリックする。
3. 「**新しいダウンストリーム LU (New downstream LU)**」を選択してから、「**OK**」をクリックする。

Communications Server for Linux が「ダウンストリーム LU (Downstream LU)」ダイアログを表示します。

4. ダイアログのフィールドに適切な値を入力する。
5. 「**OK**」をクリックする。ダウンストリーム LU が定義されます。

LU 定義は、「**ノード (Node)**」ウィンドウの「**接続 (Connectivity)**」ペインで、ダウンストリーム・ノード接続リンク・ステーションの下に表示されます。

DLUR の構成

Communications Server for Linux は、ホスト・コンピューターへの直接アクセス機能のみでなく、従属 LU リクエスター (DLUR) 機能を備えています。この機能により、ホストへ直接接続しなくても、APPN ネットワーク内の複数のノードを介して従属 LU のセッションを実行できます。

従属 LU セッションでは、通常ホスト・コンピューターとの直接通信リンクが必要となります。APPN ネットワークで多数のノード (ホスト・ノードも含む) が接続されている場合、一部のノードがホストと直接接続されておらず、他のノードを介して間接的に接続していることがあります。このような間接接続ノードの LU からホストへ、従属 LU セッションを確立することはできません。

従属 LU リクエスター (DLUR) は、このような制限を解決するように設計された APPN 機能です。

APPN ノード (Communications Server for Linux ノードなど) 上の DLUR は、ホストの従属 LU サーバー (DLUS) と連携して動作し、DLUR ノードの従属 LU から、APPN ネットワークを介した DLUS ホストまでのセッションの経路を指定します。ホストへの経路では、複数のノードを経由できるため、APPN のネットワーク管理、ダイナミック・リソース検出、経路計算などの機能を利用できます。DLUR は LU のあるノードで使用可能な状態であり、DLUS はホスト・ノードで使用可能な状態でなければなりません。ただし、セッション経路の中間ノードでは DLUR は必要ありません。

Communications Server for Linux DLUR ノードがネットワーク・ノードまたはブランチ・ネットワーク・ノードの場合は、Communications Server for Linux ノードに接続しているダウンストリーム・コンピューターの従属 LU に、パススルー DLUR 機能を提供することもできます。このような LU では、ネットワークを介してホストにアクセスするために、Communications Server for Linux ノード内部の LU と同様に Communications Server for Linux ノードの DLUR を使用できます。ダウンストリーム・コンピューターは DLUR を実行しないので、DLUR が使用されていることを認識する必要はまったくありません。

102 ページの図 8 に示すのは、APPN ネットワーク・ノードとして構成されている Communications Server for Linux サーバーで、パススルー DLUR をインプリメントしているため、ホスト (アップストリーム・ノード) の LU と、APPN ネットワーク内のノード (ダウンストリーム・ノード) の LU とのセッションをサポートしています。

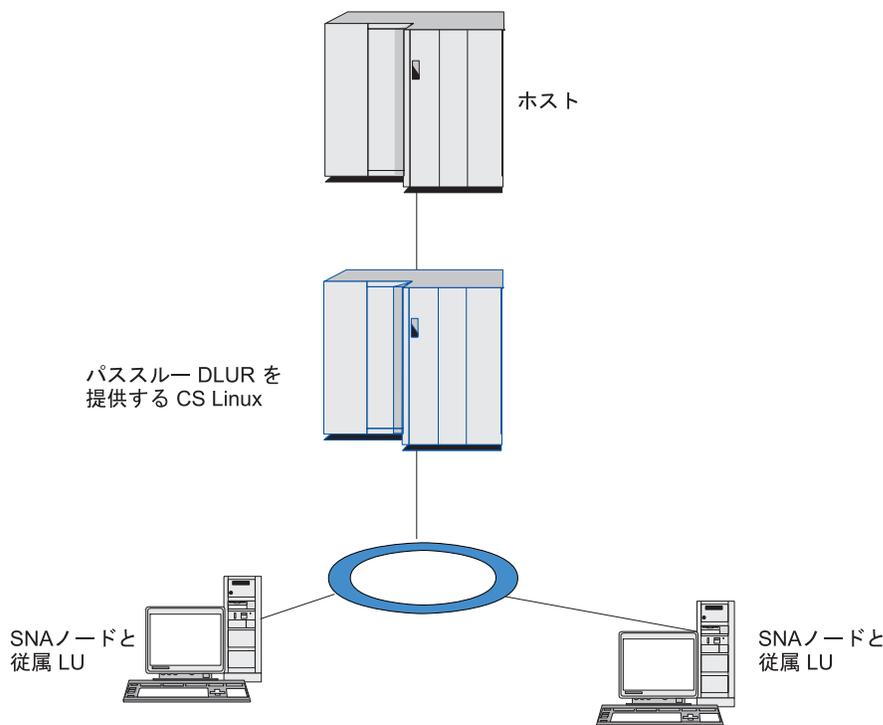


図8. DLUR を提供する *Communications Server for Linux* ノード

注:

1. LEN ノードでは DLUR を構成できません。
2. パススルー DLUR は、ネットワーク・ノードまたはブランチ・ネットワーク・ノード上でしか構成できません。
3. Branch Extender を使用する場合は、そのブランチのエンド・ノード (ブランチ・ネットワーク・ノードを、ネットワーク・ノード・サーバーとして使用する) 上で DLUR を構成することはできません。しかし、ブランチ・ネットワーク・ノード上でパススルー DLUR を構成すれば、このノードから従属 LU アプリケーションをサポートすることができます (つまり、ブランチ上のエンド・ノードは DLUR は実行しませんが、ブランチ・ネットワーク・ノード上のパススルー DLUR は使用するということです)。

DLUR の構成手順は、従属 LU が、ローカル・ノードとダウンストリーム・ノードのどちらにあるかによって異なります。

ローカル・ノードでの DLUR サポートの構成

この手順を実行するために、次の情報を収集する必要があります。

- ローカル・ノードの PU の PU ID。
- PU 名 (これはローカル ID で、ホスト構成と一致する必要はありません。)
- ホストの DLUS の名前 (およびバックアップ DLUS がある場合はその名前)。
- 各ダウンストリーム LU の LU 名、LU 番号、LU タイプ。LU 番号は、ホストで構成されている番号と一致していなければなりません。

ローカル・ノードで DLUR サポートを構成する手順は次のとおりです。

1. 80 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義する。ダウンストリーム・ノードにパススルー DLUR サポートを設定している場合には、ローカル・ノードを APPN ネットワーク・ノードまたはブランチ・ネットワーク・ノードとして定義してください。
2. APPN ネットワークとの接続を構成する。APPN 接続では、ローカル・ノードと隣接する APPN ネットワーク・ノード間の、独立トラフィック用のポートとリンク・ステーションが、少なくとも 1 つは必要になります。詳細については、81 ページの『接続の構成』を参照してください。
3. ローカル・ノードの DLUR PU を定義する (DLUR PU により、ホストとの接続がサポートされます)。

DLUR PU を構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

- a. 「サービス (Services)」メニューから、「接続 (Connectivity)」サブメニュー、「新規 DLUR PU (New DLUR PU)」の順に選択する (あるいは、ボタン・バーの「新規 (New)」ボタンをクリックして、「DLUR PU」を選択します)。

「OK」ボタンをクリックすると、Communications Server for Linux は「DLUR PU 構成 (DLUR PU Configuration)」ダイアログを表示します。

- b. ダイアログのフィールドに適切な値を入力する。
- c. 「OK」ボタンをクリックする。DLUR PU が定義されます。

「接続 (Connectivity)」ペインの DLUR 項目の下に DLUR PU が表示されます。

4. DLUR をローカル・ノードの LU をサポートするように構成するには、ローカル・ノードに LU を追加する。96 ページの『LUA の構成』で説明される方法で、LU が LUA をサポートするように構成する必要があります。LU によりサポートされているユーザー・アプリケーションの要件によっては、さらに他の項目を構成する必要がある場合もあります。

ダウンストリーム・ノードをサポートするパススルー DLUR の構成

この手順を実行するために、次の情報を収集する必要があります。

- 各ダウンストリーム・ノードのダウンストリーム PU 名、またはダウンストリーム・ノードの各 PU のダウンストリーム PU 名。(これはローカル ID で、ホスト構成と一致する必要はありません。)
- ホストの DLUS の名前

ダウンストリーム・ノードでパススルー DLUR サポートを構成する手順は次のとおりです。

1. ローカル・ノードを、APPN ネットワーク・ノードとして定義する (80 ページの『ノードの構成』を参照してください)。
2. ダウンストリーム・ノードとの接続を構成する。81 ページの『接続の構成』の手順に従って、ローカル・ノードと各ダウンストリーム・ノード間の、従属トラ

フィック用のポートとリンク・ステーションを構成します。(ダウンストリーム・ノードのパススルー DLUR をサポートするために、DLUR PU を定義する必要はありません。)

3. ダウンストリーム・ノードでは、複数の PU をサポートできる。この場合、各ダウンストリーム PU には、それぞれ異なるリンクが関連付けられているため、Communications Server for Linux DLUR ノードとダウンストリーム・ノード間に、複数のリンクを構成する必要があります。また、各リンクのダウンストリーム PU 名を確認しておく必要があります。

TN サーバーの構成

SNA ネットワークの代わりに TCP/IP を介して通信を行う 3270 エミュレーション・プログラムは、TN3270 プログラム (Telnet 3270 エミュレーション・プログラム) と呼ばれます。

TN3270 プログラムには、TN3270E (Telnet 3270 標準拡張) サポートを組み込むこともできます。TN3270E は、Telnet を使用して、3270 装置エミュレーション (端末およびプリンターの両方を含む) をサポートするオープン・プロトコルです。これにより、Telnet クライアントから、(LU 名を指定することにより) 特定の装置を選択できるようになります。また、ATTN キーおよび SYSREQ キー、SNA 応答処理などの各種機能も拡張サポートされます。

注: 本書では、TN3270、TN3287、TN3270E プロトコルのすべてに適用できる情報については、TN3270 という用語を使用します。

Communications Server for Linux TN サーバーを使用すると、3270 ホスト・コンピューター以外のコンピューターの TN3270 ユーザーが、この 3270 ホスト・コンピューターへアクセスできるようになります。TN サーバーにより、直接リンクを要求せずに TN3270 ユーザーは Communications Server for Linux、またはその他の TN3270 ユーザーとホスト接続を共有できます。また、TN3270 ユーザーは、TCP/IP を実行していないホストへもアクセスできます。

Communications Server for Linux TN サーバー機能は、TN3270 ユーザーと Communications Server for Linux 3270 LU との関連付けを行います。TN3270 ユーザーからのデータはすべて、関連付けられた LU へ転送されます。つまり、ホストと TN3270 ユーザーのどちらの構成でも、両者が直接接続しているように構成されています。したがってホストもユーザーも、データが TN サーバーを介して転送されることを認識する必要はありません。

Communications Server for Linux TN サーバーは、IETF RFC 1123、1576、1646、1647、2355 で定義されているプロトコルを正確にインプリメントしている、TN3270 クライアント・エミュレーション・プログラムをすべてサポートします。

TN3270 プログラムが TN サーバーと通信を行うと、Communications Server for Linux では、このプログラムが実行されているコンピューターの TCP/IP アドレスにより、プログラムを識別します。Communications Server for Linux では、同じコンピューター上で異なるユーザーに使用されている、2 つの TN3270 プログラムを区別することはできません。Communications Server for Linux のマニュアルでは、TN サーバー・ユーザーという用語は、TN3270 プログラムを使用している個人ユーザーではなく、TN3270 プログラムが実行されているコンピューターを指します。

通常、TN3270 サーバー機能を使用して Communications Server for Linux に接続する TN サーバー・ユーザーは、それぞれ、1 つの 3270 LU にアクセスするように構成されているため、1 回に確立できるホスト・セッションは 1 つに制限されます。ただし、TN サーバー・ユーザーが、それぞれ専用の 3270 LU を 1 つずつ持つように構成するのではなく、3270 LU プールにアクセスするよう構成することもできます。このように設定すると、ユーザーは、プール内で使用可能な LU と同数のセッションにアクセスできます。

TN サーバーのアクセスを構成する前に、次の構成作業を行ってください。

- 80 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義する。
- 81 ページの『接続の構成』の説明に従って、ローカル・ノードとホスト間の、従属トラフィック用のポートとリンク・ステーションを構成する。

TN サーバーのアクセスを構成するには、次の構成作業を行ってください。

- ホストとの通信に使用するローカル・ノードの 3270 LU を定義する。LU を追加する場合は、『3270 LU の定義』を参照してください。
- LU プールを使用する場合は、『LU プールの定義』の説明に従って LU プールを定義する。

3270 LU の定義

3270 LU を構成する前に、次の情報を収集してください。

- LU 名 (これはローカル ID で、ホスト構成と一致する必要はありません。)
- LU 番号 (または、一連の LU の数)
- LU タイプ (3270 ディスプレイ・モデルまたは 3270 プリンター)
- プール名 (LU をプールに追加する場合)

既に定義されているリンク・ステーションについて、タイプ 0 から 3 の LU を構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. ウィンドウの「接続 (Connectivity)」ペインから、ホスト接続リンク・ステーションを選択する。
2. 「新規 (New)」ボタンをクリックする。
3. 表示されるダイアログで、LU タイプ (「新規 3270 ディスプレイ LU (New 3270 display LU)」、または「新規 3270 プリンター LU (New 3270 printer LU)」) を選択する。

この項目を選択して「OK」をクリックすると、Communications Server for Linux は「LU タイプ 0 から 3 (LU Type 0-3)」ダイアログを表示します。

4. ダイアログのフィールドに適切な値を入力する。
5. 「OK」をクリックする。LU が定義されます。

LU は、「ノード (Node)」ウィンドウの「接続 (Connectivity)」ペインに表示されている、ホスト接続リンク・ステーションの下に表示されます。

LU プールの定義

3270 では、LU プールを定義することにより、ユーザー構成が単純化され、ホスト・セッションの確立を非常に柔軟に行うことができます。たとえば、複数の 3270

LU を 1 つの LU プールに定義しておくこと、この LU プールを使用して複数の TN3270 クライアントを構成できます。これにより、3270 セッションの構成が容易になり、またクライアントでプール内の LU をすべて使用できるようになります。

注: TN3270 クライアントを特定の LU または LU プールのいずれにも割り当てることができる。

- クライアントをプール内の特定の LU に割り当てる場合、この LU が使用可能であればクライアントで使用されます。そうでない場合、プール内のその他の空いている LU が、特定の LU の代わりに、クライアントに割り当てられているかのように使用されます。
- クライアントに特定の LU のみを使用させ、LU が既に使用中のときはクライアントのセッションを確立できないようにする場合、その LU がプール内にないことを確認してください。

ローカル Communications Server for Linux ノードの LU プールは、「LU プール (LU Pools)」ウィンドウで確認できます。このウィンドウには、ローカル・システムで構成されている LU プールのリストが表示されます。また、LU プールに追加する LU を選択することもできます。

3270 が使用するプールに追加できる LU タイプを次に示します (ただし、1 つのプールには異なる種類の LU を混在させないでください)。

- 3270 ディスプレイ LU
- 無制限 LU

LU は、プールに追加する前に必ずローカル・ノードで定義してください。

LU プールを構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. 「**ウィンドウ (Windows)**」メニューから、「**LU プール (LU Pools)**」を選択する。

Communications Server for Linux が「LU プール (LU Pools)」ウィンドウを表示します。

2. 「**新規 (New)**」ボタンをクリックする。

Communications Server for Linux が「LU プール構成 (LU Pool Configuration)」ダイアログを表示します。

右側のボックスに、まだプールに割り振られていない LU のリストが表示されます。このリストに表示されている LU はすべて、新しいプールに割り振ることができます。

3. プールに追加する LU (1 つ以上) を選択してから、「**新規 (New)**」ボタンをクリックする。選択された LU が左側のボックスへ移動します。

左側のボックスから LU を除去するには、LU を選択して「**除去 (Remove)**」ボタンをクリックします。

4. 「**OK**」をクリックする。LU プールが定義されます。

左側のボックスに表示されている LU がすべて LU プールに追加されます。

定義されたプールが「LU プール (LU Pools)」ウィンドウに表示されます。

TN3270 サーバーの構成

TN3270 サーバーを構成する前に、次の情報を収集する必要があります。

- サーバーが TN3270 のみをサポートしているのか、TN3270E (TN3270 サポートも含まれます) もサポートしているかどうか。
- TN3270E クライアントが特定の LU を要求できるかどうか。
- 各クライアントのディスプレイとプリンターの LU 名 (あるいは LU プール名)。(プリンター LU 名が必要となるのは、TN3270E をサポートしている場合のみです。)
- 特定のクライアントのみが許可されている場合や、特定の LU に対してクライアントを制限する場合には、対象となるクライアントの TCP/IP 名またはアドレスが必要です。
- TN サーバー・ノードの TCP/IP ポート番号。
- SSL データの暗号化、クライアント認証、およびサーバー認証が必要かどうか (このオプションは、このサポートに必要な追加のソフトウェアがインストールされている場合にのみ使用可能です)。

ディスプレイ LU とプリンター LU を関連付ける場合は、これらの LU の名前も必要です。TN サーバーのアソシエーション・レコードにより、プリンター LU とディスプレイ LU の関連付けが定義されるため、この 2 つの LU は TN3270E プロトコルによって接続できます。TN3270E またはプリンター LU をサポートしていない場合は、アソシエーション・レコードを定義する必要はありません。

TN サーバーのデフォルト・レコードで定義されるパラメーターは、すべての TN3270 クライアント・セッションで使用されます。各サーバーにデフォルト・レコードを 1 つずつ定義できます。

TN3270 サーバーを構成するには、「ノード (Node)」ウィンドウで次の手順を行います。

1. TN サーバーのアクセス・レコードを定義する。

- a. 「サービス (Services)」メニューから、「TN サーバー (TN Server)」を選択する。

Communications Server for Linux は「TN サーバー (TN Server)」ウィンドウを表示します。このウィンドウの上部のペインには、TN サーバーの構成済みアクセス・レコードが表示され、下部のペインには、TN サーバーのアソシエーション・レコードが表示されます。

- b. TN3270 サーバーのアクセス・レコードが表示されているペインを選択してから、「新規 (New)」ボタンをクリックする。

Communications Server for Linux が「TN サーバー・アクセス (TN Server Access)」ダイアログを表示します。

- c. ダイアログのフィールドに適切な値を入力する。
- d. 「OK」をクリックする。TN サーバー・アクセス・レコードが定義されます。定義されたレコードが「TN サーバー (TN Server)」ウィンドウに表示されます。

TN サーバーの構成

2. TN サーバーのアソシエーション・レコードを定義する。
 - a. 「TN サーバー (TN Server)」ウィンドウで、アソシエーション・レコードが表示されているペインを選択してから、「新規 (New)」ボタンをクリックする。

Communications Server for Linux が「TN サーバー・アソシエーション・レコード (TN Server Association Record)」ダイアログを表示します。
 - b. ダイアログのフィールドに適切な値を入力する。
 - c. 「OK」をクリックする。TN サーバーのアソシエーション・レコードが定義されます。定義されたレコードが「TN サーバー (TN Server)」ウィンドウに表示されます。
3. プリンター応答を強制する必要がある場合は、TN3270 のすべてのセッションにキープアライブ・メソッドを指定するか、TN3270 クライアントの許可検査に使用される取り消しリストを保有する、外部 LDAP サーバーへのアクセス方法を指定するか、あるいは、TN3270 SLP (サービス・ロケーション・プロトコル)、または TN サーバー拡張パラメーター・ダイアログを使用する。

TN サーバーで SSL サポートを構成する詳細については、IBM Communications Server Support Web ページの <http://www.ibm.com/software/network/commserver/support/> を参照してください。

TN リダイレクターの構成

Communications Server for Linux TN リダイレクター機能により、TN3270、TN3270E、TN5250、および VT クライアント (これらは Telnet クライアントと総称されます) への、パススルー TCP/IP ホスト・アクセスが可能になります。Telnet ユーザーは、TCP/IP 接続を介して Communications Server for Linux との通信を行います。Communications Server for Linux は次に、別の TCP/IP 接続を介してホストとの通信を行います。これにより、ユーザーからホストへの接続完了時でなくても、必要な場合に、Secure Sockets Layer (SSL) セキュリティー検査を実施できます。たとえば、以下のとおりです。

- クライアントが、検査を必要としない TCP/IP LAN を経由して Communications Server for Linux に接続しており、しかも SSL が必要なリモート・ホストに接続している場合、Communications Server for Linux とホストの間の TCP/IP を介して SSL を使用できます。つまり、すべてのクライアントに対するセキュリティーを一度検査すれば、個々のクライアントがセキュリティー情報を提供する必要はないということです。
- Communications Server for Linux がホストと同じサイトにインストールされていても、クライアントが外部サイトから接続している場合、ホストに SSL ソフトウェアをインストールしなくても、Communications Server for Linux にクライアント接続すれば SSL を使用できます。

TN リダイレクターの構成

TN リダイレクター・アクセスの構成を行う前には、80 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義しなければなりません。さらに、次の情報を収集する必要もあります。

- 特定のクライアントのみが許可されている場合は、そのクライアントの TCP/IP 名またはアドレスが必要です。
- クライアントが TN リダイレクター・ノードに接続する際に使用する、TCP/IP ポート番号。
- ホストの TCP/IP 名またはアドレス。
- TN リダイレクター・ノードがホストに接続する際に使用する、TCP/IP ポート番号。
- クライアントと TN リダイレクター・ノードの間で、SSL データの暗号化、クライアント認証、およびサーバー認証が必要かどうか (このオプションは、そのサポートに必要な追加のソフトウェアがインストールされている場合にのみ使用可能です)。
- TN リダイレクター・ノードとホストの間で、SSL データの暗号化が必要かどうか。

TN リダイレクターのデフォルト・レコードで定義されるパラメーターは、すべての TN リダイレクター・クライアント・セッションで使用されます。個々のクライアント TCP/IP ポート番号には、デフォルト・レコードを 1 つずつ定義することができます。

TN リダイレクターを構成するには、「ノード (Node)」ウィンドウで次の手順を実行して、TN リダイレクター・アクセス・レコードを定義します。

1. 「サービス (Services)」メニューから、「TN サーバー (TN Server)」を選択する。

Communications Server for Linux は「TN サーバー (TN Server)」ウィンドウを表示します。このウィンドウには、TN3270 サーバーの構成済みアクセス・レコード、TN3270 サーバーのアソシエーション・レコード、および TN リダイレクターのアクセス・レコードが表示されます。

2. TN リダイレクターのアクセス・レコードが表示されているペインを選択してから、「新規 (New)」ボタンをクリックする。

Communications Server for Linux が「TN リダイレクター・アクセス (TN Redirector Access)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力する。
4. 「OK」をクリックする。TN リダイレクターのアクセス・レコードが定義されます。定義されたレコードは、「TN サーバー (TN Server)」ウィンドウの「TN リダイレクター (TN Redirector)」ペインに表示されます。

注: TN リダイレクターを使用するために、SNA ノードはアクティブである必要があります。ただし、ノードの SNA リソースはまったく使用されません。

Communications Server for Linux を使用不可にする方法

Communications Server for Linux ソフトウェアを使用不可にすると、自動的に Communications Server for Linux ノードと関連する接続コンポーネントも停止します。Communications Server for Linux を使用不可にすると、その他のプロセス (LUA アプリケーションなど) も、このサーバー上の Communications Server for Linux リソースを使用しなくなります。

Communications Server for Linux を使用不可にする方法

通常、サービスを終了するユーザーとして、サービスを個別に停止して Communications Server for Linux 上で何も実行されていない状態にしてから、システムを使用不可にしてください。

ユーザーがアクティブであるときに Communications Server for Linux ソフトウェアを使用不可にする必要がある場合は、Communications Server for Linux が停止する前に、これらのユーザーに Communications Server for Linux が停止しようとしている旨を通知し、各ユーザーがアクティビティを終了するために時間を与えてください。

Communications Server for Linux ソフトウェアを使用不可にすると、APPC、CSV、LUA、NOF、または MS API を使用しているアプリケーションには、戻りコード COMM_SUBSYSTEM_ABENDED によって通知され、CPI-C アプリケーションには、戻りコード CM_PRODUCT_SPECIFIC_ERROR によって通知されます。

Communications Server for Linux ソフトウェアを使用不可にするには、Linux コマンド・プロンプトに次のコマンドを入力します。

sna stop

Communications Server for Linux が正常に使用不可になると、**sna stop** により終了コード 0 が戻されます。これ以外の終了コードは、エラーが発生して、Communications Server for Linux ソフトウェアを使用不可にできなかったことを示します。終了コード値の詳細は、「*Communications Server for Linux 診断用ガイド*」を参照してください。

Communications Server for Linux の自動的な始動

Communications Server for Linux を完全に作動している状態に起動するまでには、いくつかの段階に分類されます。

- Communications Server for Linux ソフトウェアの使用可能化
- SNA ノードの初期化
- ノードで構成済みのポートおよびリンク・ステーションの活動化

上記の各段階は個別に管理され、必要な場合、すべての段階がブート時に行われます。このセクションでは、これらの段階について説明し、ブート時に制御する方法について示します。

Communications Server for Linux の使用可能にする方法

Communications Server for Linux が作動するには、いくつかのカーネル・モジュールがロードされる必要があります。これらのモジュールは、Communications Server for Linux ソフトウェアが最初に使用可能にされるときにロードされ、カーネルの実行レベルが変更された場合にのみアンロードされます。

また、Communications Server for Linux では多くのデーモン (プログラム) が実行される必要があります。これらのデーモンが実行されるまで、Communications Server for Linux を構成または使用できません。

- デーモンを開始して、Communications Server for Linux ソフトウェアを使用可能にするには、コマンド **sna start** を実行します。

- デーモンを停止して、Communications Server for Linux ソフトウェアを使用不可にするには、コマンド **sna stop** を実行します。

SNA ノードの初期化

Communications Server for Linux ソフトウェアが使用可能にされると、通常は Motif 管理プログラム **xsnaadmin** を使用して、SNA ノードおよびそのリソースを構成できるようになります。ただし、ノードは初期化されるまで使用できません。ノードの初期化は、コマンド行管理プログラムからコマンド **snaadmin init_node** を使用して、または Motif 管理プログラム **xsnaadmin** から行うことができます。

ポートとリンク・ステーションの活動化

ポートとリンク・ステーションは、さまざまな状況で開始するように構成できます。

- オペレーター介入のみ
- オンデマンド (リンク上のリソースを使用するアプリケーションの開始時)
- ノードの起動時 (**snaadmin init_node** が実行される場合、またはノードが Motif 管理プログラムから開始される場合)。

デフォルトはオペレーターによる始動ですが、コマンド行管理プログラムまたは Motif 管理プログラムを使用して、特定のポートまたはリンクに対して変更することができます。

注: ポートを開始すると、そのポートは他のコンピューターからの呼び出しを受信することができますが、発呼を行うことはできません。リンク・ステーションを開始すると、Communications Server for Linux がリモート・コンピューターに接続しようとします。

リポート時の Communications Server for Linux の開始

その他の Linux サービスと同様に、Communications Server for Linux はリポート時に使用可能になります。つまり、デフォルトでは、リポートの後にコマンド **sna start** が実行されますが、SNA ノードは開始されません。

この初期化は、Communications Server for Linux ブート時の初期化スクリプト **/etc/rc.d/init.d/snastart** により行われます。始動スクリプトは従来と同じように、さまざまなブート・レベルに対応するように **/etc/rc?.d/init.d/snastart** にリンクされています。

/etc/rc.d/init.d/snastart を編集して、リポート時の方法を変更することができます。もっとも一般的な変更方法は、ノードの初期化を追加する方法です。このためのコマンド **snaadmin init_node** は既にファイルに含まれていますが、コメント化されているため、アンコメントする必要があります。このコマンドを組み込むことにより、ノードの始動時に活動化されるように構成されたポートまたはリンク・ステーションを活動化させるトリガーとすることもできます。

Communications Server for Linux を使用するアプリケーションは、ノードが初期化されるまで開始できません。必要に応じて、これらのアプリケーションはブート時に自動的に開始して、ブート時に実行する必要があるその他の **snaadmin** コマンドを追加できます。その操作は、以下の 2 つの方法のいずれかから実行できます。

Communications Server for Linux の自動的な始動

- コマンドを `/etc/rc.d/init.d/snastart` の終わりで、`snaadmin init_node` コマンドの後に追加します。
- `/etc/rc?.d/init.d` スクリプトが確実に Communications Server for Linux が開始された後に実行されるように、95 より大きい番号を指定して作成し、このスクリプトにコマンドを追加します。

注: `/etc/rc.d/init.d/snastart` ファイルへの変更は、Communications Server for Linux を以降のバージョンにアップグレードする際に保管されません。アップグレードの後にこの変更を再適用できるように、常に、この変更のコピーを保持していることを確認してください。

第 9 章 Communications Server for Linux と SNA の情報リソース

この章では、SNA テクノロジーに関する情報を提供する SNA ライブラリーのリソースと、IBM が提供するさまざまなネットワーク製品およびサービスについて説明します。また、ネットワーク・フォーラムで入手できる情報についても説明します。

SNA ライブラリー

SNA ライブラリーには、マーケティング・パンフレット、資料、ユーザー・ガイド、チュートリアルが含まれ、以下のトピックに関する入門情報および詳細情報を提供します。

- SNA 理論
- SNA 製品
- 製品のインプリメンテーション
- システムおよびネットワークの構成
- SNA アプリケーション・プログラムおよび API
- 全体の計画、パフォーマンスおよびチューニング
- 問題診断
- ネットワーク管理
- ネットワーク・セキュリティ

すべての IBM の資料は、IBM 担当員または地区担当の IBM 営業所を通じて注文できます。

それ以外の資料に関する情報は、IBM 担当員にお尋ねください。

ネットワークで入手可能な情報

情報の交換を促進するため、IBM は電子フォーラムと電子掲示板のスポンサーを務めています。また、インターネット上にホーム・ページを開設し、WWW でアクセスできるオンライン資料を公開しています。

IBMLink™ を使用した製品サポート

IBMLink フォーラムは、IBM が所有するネットワーク上に開設されています。このフォーラムは、IBM 製品のライセンスを持つお客様に対して、技術的な問題の解決方法、あるいは、お客様のシステムやネットワークに関するその他の問題の解決方法を提供することを目的としています。IBM の従業員が質問にお答えし、IBM 製品のユーザー間のオンライン・ディスカッションを介在します。

IBMLink に関する詳細は、<http://www.ibm.link.ibm.com> をご覧ください。

IBM ホーム・ページの情報

インターネット上では、さまざまな IBM ホーム・ページを用意し、フォーラムにアクセスできるようにしています。包括的なヘルプ情報について、IBM のメイン・ホーム・ページを使用して、インターネット上および WWW 上の情報センター検索できます。メイン・ホーム・ページにアクセスするには、<http://www.ibm.com> (日本語の情報は、<http://www.ibm.co.jp>) を使用します。

<http://www.ibm.com/software/network> にアクセスすると、Communications Server for Linux も含めて、IBM ネットワーク・ソフトウェアに関する情報を入手することができます。Communications Server for Linux に関する情報は、<http://www.ibm.com/software/network/commsserver> (日本語の情報は、<http://www.ibm.co.jp/software/secureway/cms/cmsAIX.html>) にあります。

Communications Server for Linux のサポートに関する詳細情報は、<http://www.ibm.com/software/network/commsserver/support> (英語サイト) をご覧ください。

ダウンロードに関する情報

ユーザーは、WWW 上で <http://www.redbooks.ibm.com> にアクセスして、レッドブック資料をダウンロードすることができます。

IBM ソフトウェアについての情報は、<http://www.ibm.com/software> (英語サイト) でアクセスできます。ここからは、Communications Server for Linux およびすべての IBM ソフトウェア・サーバーに関するページに、リンクすることができます。

推薦資料

SNA の知識を深めたい場合は、SNA の理論と Communications Server for Linux の実践方法を扱っている以下の資料が参考になります。これらの資料は、SNA に関する知識を第一歩から学ぶ初心者にも、綿密な処理方法を知りたい専門家にも、役立ちます。

- *Systems Network Architecture: Technical Overview* (GC30-3073)
- *IBM Communications Server for Linux 管理ガイド*

さらに特定の関心がある場合は、IBM 担当員にお問い合わせください。

付録. 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものであり、本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation
P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, NC 27709-2195
U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができませんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権: 本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。お客様は、IBM のアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生した創作物にも、次のように、著作権表示を入れていただく必要があります。「(C)

(お客様の会社名) (西暦年)」。このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。® Copyright IBM Corp. 2000, 2005, 2006. All rights reserved.

商標

以下は、IBM Corporation の商標です。

Advanced Peer-to-Peer Networking	Power5
AIX	pSeries
Application System/400	S/390
AS/400	SP
CICS	System/370
IBM	System/390
MQSeries	Systems Application Architecture
MVS	VTAM
MVS/ESA	WebSphere
MVS/XA	z/OS
NetView	z9
OpenPower	zSeries
OS/2	

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Intel、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Intel Centrino、Intel Centrino ロゴ、Celeron、Intel Xeon、Intel SpeedStep、Itanium、および Pentium は、Intel Corporation の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

参考文献

以下の IBM 資料には、このライブラリーで説明しているトピックに関する情報が記載されています。資料は、次のトピック別に大きく分けてあります。

- Communications Server for Linux、バージョン 6.2.2
- システム・ネットワーク体系 (SNA)
- ホスト構成
- z/OS Communications Server
- TCP/IP プロトコル (TCP/IP)
- X.25
- 拡張プログラム間通信機能 (APPC)
- プログラミング
- その他の IBM ネットワーキングに関するトピック

Communications Server for Linux ライブラリーの資料については、その要旨が説明されています。その他の資料については、資料タイトル、資料番号、および本書の本文で使用される資料タイトルの省略名 (記載されていない場合もあります) が記載されています。

Communications Server for Linux バージョン 6.2.2 資料

Communications Server for Linux ライブラリーは、以下の資料により構成されています。なお、これらの資料のソフトコピー版が CD-ROM で提供されています。CD-ROM のソフトコピー・ファイルへのアクセス方法については、「*IBM Communications Server for Linux 入門*」を参照してください。ご使用のシステムにこれらのソフトコピー・ブックをインストールするには、9 から 15 MB のハード・ディスク・スペースが必要です (インストールする各国語バージョンにより異なります)。

- *IBM Communications Server for Linux 入門* (GC88-9996-01)

この資料は Communications Server for Linux の概要を示すもので、サポートされているネットワークの特性、インストール、構成、および操作について説明しています。

- *IBM Communications Server for Linux 管理ガイド* (SC88-9999-01)

この資料には、SNA および Communications Server for Linux の概要、および Communications Server for Linux の構成および操作に関する情報が記載されています。

- *IBM Communications Server for Linux Administration Command Reference* (SC31-6770-02)

この資料には、SNA および Communications Server for Linux のコマンドに関する情報が記載されています。

- *IBM Communications Server for Linux CPI-C Programmer's Guide* (SC31-6774-02)

この資料には、熟練した “C” または Java プログラマー向けに、Communications Server for Linux CPI コミュニケーション API を使用した SNA トランザクション・プログラムの作成に関する情報が記載されています。

- *IBM Communications Server for Linux APPC Programmer's Guide* (SC31-6773-02)

この資料には、拡張プログラム間通信機能 (APPC) を使用したアプリケーション・プログラムの作成に必要な情報が記載されています。

- *IBM Communications Server for Linux LUA Programmer's Guide* (SC31-6776-02)

この資料には、標準的な LU アプリケーション・プログラミング・インターフェース (LUA) を使用したアプリケーションの作成に必要な情報が記載されています。

- *IBM Communications Server for Linux CSV Programmer's Guide* (SC31-6775-02)

この資料には、Common Service Verbs (CSV) アプリケーション・プログラム・インターフェース (API) を使用したアプリケーション・プログラムの作成に必要な情報が記載されています。

- *IBM Communications Server for Linux MS Programmer's Guide* (SC31-67770-02)

この資料には、管理サービス (MS) API を使用したアプリケーションの作成に必要な情報が記載されています。

- *IBM Communications Server for Linux NOF Programmer's Guide* (SC31-6778-02)

この資料には、ノード・オペレーター機能 (NOF) API を使用したアプリケーションの作成に必要な情報が記載されています。

- *IBM Communications Server for Linux Diagnostics Guide* (SC31-6779-02)

この資料には、SNA ネットワーク問題の解決に関する情報が記載されています。

- *IBM Communications Server for Linux APPC Application Suite User's Guide* (SC31-6772-02)

この資料には、Communications Server for Linux を使用する APPC アプリケーションに関する情報が記載されています。

- *IBM Communications Server for Linux Glossary* (GC31-6780-02)

この資料には、IBM Communications Server for Linux ライブラリー全体で使用される用語および定義の包括的なリストが記載されています。

システム・ネットワーク体系 (SNA) 資料

次の資料には、SNA ネットワークに関する情報が記載されています。

- *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)

ホスト構成資料

次の資料には、ホスト構成に関する情報が記載されています。

- *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 資料

以下の資料には、z/OS Communications Server に関する情報が記載されています。

- *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 資料

以下のマニュアルには TCP/IP ネットワーク・プロトコルに関する情報が記載されています。

- *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 資料

以下のマニュアルには X.25 ネットワーク・プロトコルに関する情報が記載されています。

- *Communications Server for OS/2 Version 4 X.25 Programming* (SC31-8150)

APPC 資料

次の資料には、拡張プログラム間通信機能 (APPC) に関する情報が記載されています。

- *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)

プログラミング資料

次の資料には、プログラミングに関する情報が記載されています。

- *共通プログラミング・インターフェース Communications CPI-C Reference* (SC26-4399)
- *Communications Server for OS/2 Version 4 Application Programming Guide* (SC31-8152)

他の IBM ネットワーキング資料

次の資料には、Communications Server for Linux に関連するその他のトピックについての情報が記載されています。

- *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

日本語、数字、英字、特殊文字の順に配列されています。なお、濁音と半濁音は清音と同等に扱われています。

[ア行]

圧縮、LU セッション・データ 5
アップストリーム・ノード 101
アプリケーション・プログラミング・インターフェース (API) 5
アラート 18
暗黙のダウンストリーム LU の構成 99
イーサネット
 リンクの構成 83
 SAP ダイアログ 84
インストール
 既存のパッケージの詳細 26
 実行後の保守作業 33
 準備 26
 Remote API Client on AIX 52, 53
 Remote API Client on Linux 41
 System z 用の Remote API Client on Linux 47
インストール後の作業 33
インストール前の作業 26
インストール要件 22
インターフェースの選択 15
ウィンドウ 75
エントリー・ポイント 17
オープン・システム間相互接続 (OSI) 24
オペレーティング・システム要件 22
オンライン
 ディスカッション・グループ 113
 フォーラム 114
 文書 113
 ヘルプ 15
 APPC 114

[カ行]

拡張プログラム間通信機能 (APPC) 7
活動化、ポートとリンク・ステーションの 111
関数 5
管理プログラム
 コマンド行 15
 Motif 15, 17
 NOF API 15
機能要件 19
旧レベルの Communications Server for Linux からのマイグレーション 27

共通プログラミング・インターフェース・コミュニケーション (CPI-C) 6
クライアント 7
クライアント/サーバー
 構成 79
クライアント/サーバーのインストール 34
クライアント/サーバー・サポート 6
クライアント・ソフトウェア
 アップグレード 68
 アンインストール 68
 カスタマイズ 67
 再インストール 68
ゲートウェイ
 定義 8
 SNA ゲートウェイ 8
計画ワークシート 72
言語環境変数 26, 52
 Remote API Client on Linux 41
 System z 用の Remote API Client on Linux 46
広域ネットワーク (WAN) 3
構成 20
 暗黙のダウンストリーム LU 99
 計画 72
 接続 81
 ノード 80
 バックアップ 35
 表示 78
 ファイル 35, 36
 変更 78
 ポート 82
 リモート・ノード 92
 例 101
 APPC 通信 88
 CPI-C サイド情報 95
 DLUR 101
 LEN ノードのパートナー LU 91
 LU 6.2 91, 94
 LU タイプ 0-3 86
 SNA ゲートウェイのダウンストリーム LU 98
 TN サーバーのアソシエーション・レコード 107
 TN サーバーのデフォルト 107
 TN リダイレクターのデフォルト 109
「構成 (Configuration)」ウィンドウ、Remote API Client on Windows
 拡張パラメーター 62
 パラメーター 61
構成サーバー 79
 除去 80
 追加 80
構成の表示 78
構成の変更 78

国際標準化機構 (ISO) 24
コマンド行管理プログラム 15
コンポーネントの管理 75

[サ行]

サーバー 6
 除去 80
 追加 80
 Linux ハードウェア要件 22
サーバー、Telnet 11
サーバー名 61
サービス・アクセス・ポイント (SAP) 22, 83
再接続タイムアウト・パラメーター 63
作業シート 73
サブルーチン 5
システム・アプリケーション体系 (SAA) 6
始動コマンド 74
従属 LU 6.2 94
従属 LU の動的定義 (DDDLU) 4
従属 LU リクエスト (DLUR) 3
従属論理装置サーバー (DLUS) 101
情報リソース 113
初期化、SNA ノードの 111
推薦資料 114
スキル要件 21
ストレージ容量 23
セキュリティー・オプション 16
セッション
 サポート 5
 ベーシング 16
 ルーティング 16
 U 型 5
接続
 オプション 3
 構成 81
接続ネットワーク 16
接続ネットワーク、構成 84
ソフトウェア機能 8
ソフトウェア要件 22, 23
 Remote API Client on AIX 51
 Remote API Client on Linux 40
 Remote API Client on Windows 57
 System z 用の Remote API Client on Linux 45

[タ行]

ダイアログ 75
対等
 ネットワーク 2
ダウンストリーム LU
 構成 100
 ハードウェアの例 98
 SNA ゲートウェイ 98
ダウンストリーム・コンピューター 98

ダウンストリーム・ノード 101
中間セッション・ルーティング (ISR) 16
ツールバー・ボタン 78
データ・ストリーム 6
データ・リンク制御オプション 3
停止コマンド 110
ディスカッション・グループ、オンラインの 113
適応セッション・レベル・ベーシング 16
手順
 Remote API Client on Windows のインストール 59, 63
透過性、ローカル / リモートの 5
同期データ・リンク制御 (SDLC) 82
動的構成 15
独立 LU 6.2 の構成 91
ドメイン 6
ドメイン・パラメーター 61
トラブルシューティング・ツール 17
トランザクション・プログラム (TP) 5
トランスポート・メディア 22

[ナ行]

ネットワーク
 アラート 18
 管理 17
 計画 19, 20, 24
 サブエリア 2
 サポート 2
 対等 2
 入手可能な情報 113
 命名規則 24
ノード
 アップストリーム 101
 構成 80
 初期化 111
 ダウンストリーム 103
 リモート 91, 92
 ローカル (local) 102
「ノード (Node)」ウィンドウ 75

[ハ行]

ハードウェア
 リンク 22
ハードウェア要件
 サーバー 22
 Remote API Client on AIX 51
 Remote API Client on Linux 39
 Remote API Client on Windows 57
 System z 用の Remote API Client on Linux 45
パートナー LU
 構成 91, 92
 別名 93
 リモート・ノード 93
 ワイルドカードを使用した定義 93

- パートナー・アプリケーション 7
- ハード・ディスク・ストレージ 23
- バックアップ
 - 構成ファイル 35
 - 復元 36
- バックアップ、マスター・サーバー 79
- バックアップ・サーバー 7
- ピア・サーバー 7
- フォーカル・ポイント 17
- フォーラム、オンラインの 114
- 複数のサーバー、ドメイン内の 7
- 物理装置 (PU) 8
- プライマリー RUI 5
- ブランチ・ネットワーク・ノード 9
- ブロードキャスト試行の最大回数パラメーター 62
- 分散処理
 - アプリケーションのサポート 7
 - 環境 3
- 文書、オンラインの 113
- ペーシング、セッション・レベル 16
- 別名、パートナー LU の定義 93
- ヘルプ
 - Motif 管理プログラム 15
- ポート
 - 活動化 111
- ポートの構成 82, 84, 85
- ホスト
 - サブエリア・ネットワーク内で 2
 - APPN ネットワーク内での 3
 - LU サポート 4
- ボタン、リソース・ウィンドウの 78

[マ行]

- マスター・サーバー 7
- マルチドメイン・サポート・ネットワーク管理ベクトル・トランスポート (MDS-NMVT) 17
- 命名規則 24
- メディア・アクセス制御 (MAC) 84
- メモリー所要量 23
- 問題
 - 診断ツール 17
 - データ 18

[ヤ行]

- 要員の要件 21
- 要件
 - インストール 22
 - オペレーティング・システム 22
 - ソフトウェア 22, 23
 - メモリーおよびストレージ 23
 - 要員とスキル 21
 - HTTPS 23
 - WebSphere Application Server 23

呼び出し 5

[ラ行]

- ライセンス・プログラム 29
 - Remote API Client on AIX 52
 - Remote API Client on Linux 41
 - System z 用の Remote API Client on Linux 47
- リソース
 - 管理 75
 - 項目 78
 - 削除 77
 - 始動 77
 - 情報 113
 - 定義 77
 - 停止 77
 - 要件 20
- リソースの開始 77
- リソースの削除 77
- リソースの追加 77
- リソースの停止 77
- リポート時の Communications Server for Linux の開始 111
- リモート・ノード
 - 構成 91, 92
 - パートナー LU の構成 93
- リリース情報 34
- リンク・ステーション
 - 活動化 111
 - ポート上での定義 83
- リンク・ハードウェア 22
- ローエントリー・ネットワークキング (LEN) ノード 84
- ローカル LU の定義 91
- ローカル / リモート透過性 5
- ローカル・エリア・ネットワーク (LAN) 3
- 論理装置 (LU) 4, 100

[ワ行]

- ワークシート、計画 72
- ワイルドカード 93

[数字]

- 3270 LU
 - 定義 105
 - TN サーバーの 12, 104

A

- API
 - サポート 5
 - Communications Server for Linux 管理用 15
 - Communications Server for Linux のタイプ 6
- APPC
 - オンライン 114

APPC (続き)

- 構成 88
- 従属 LU 6.2 94
- 独立 LU 6.2 91
- 分散アプリケーションのサポート 7
- application suite 8, 11

APPC Application Suite 11

APPN

- アプリケーション 8, 11
- エンド・ノード 3
- 構成 89, 90
- サブエリア機能 3
- 接続ネットワーク 16
- 動的構成 15
- 独立 LU サポート 4
- ネットワークのセグメンテーション 24
- ネットワーク・ノード 3
- ノード・タイプ 3
- ホスト・サポート 3
- ルーティング 16
- DLUR サポート 3

B

- Branch Extender 9

C

- Communications Server for Linux 実行可能プログラム用のディレクトリー 73
- Communications Server for Linux 実行可能プログラム用のパス 73
- Communications Server for Linux ソフトウェアを使用可能にする方法 110
- Communications Server for Linux のアンインストール 38
- Communications Server for Linux のインストール 29
- Communications Server for Linux の始動
 - 自動的、システム起動時に 74
- Communications Server for Linux の自動的な始動 110
 - 活性化、ポートとリンク・ステーションの 111
 - 始動、リブート時の 111
 - 初期化、SNA ノードの 111
- Communications Server for Linux ソフトウェアを使用可能にする方法 110
- Communications Server for Linux 用のオンライン資料 31
- Communications Server for Linux 用の資料 31
- Communications Server for Linux を使用可能にする方法
 - ローカル・システムでの 74
- Communications Server for Linux を使用不可にする方法 110
- CPI コミュニケーション (CPI-C) 6
- CPI-C
 - 構成 95
 - 相互運用性 18
 - API 6

D

- DDDLU (従属 LU の動的定義) 4
- DLC
 - 構成 84, 85
 - ポート構成の 82
- DLUR
 - 構成 101
 - 説明 3
 - ダウンストリーム・ノードのサポート 103
 - ローカル・ノード上の 102
 - PU 構成 103
- DLUS 101

E

- Enterprise Extender
 - 概要 11
 - ポート・ダイアログ 85
 - リンクの構成 85

G

- GSKIT
 - アンインストール、Remote API Client on Windows 69
 - Remote API Client on AIX 51, 54
 - Remote API Client on Linux 40, 42
 - Remote API Client on Windows 65, 66
 - System z 用の Remote API Client on Linux 46, 48

H

- HACL 31
- Host Access Class Library 31
- HPR
 - ISR との比較 16
- HPR/IP 11
- HTTPS
 - 構成 31
 - 要件 23
 - Remote API Client on AIX 54
 - Remote API Client on Linux 42
 - Remote API Client on Windows 65, 66
 - System z 用の Remote API Client on Linux 48

I

- IP ポート・ダイアログ 85
- ISO (国際標準化機構) 24
- ISR (中間セッション・ルーティング) 16

J

- Java
 - Remote API Client on AIX 51

Java (続き)

Remote API Client on Linux 40

System z 用の Remote API Client on Linux 45

L

LAN アクセスのタイムアウト・パラメーター 62

LAN (ローカル・エリア・ネットワーク) 3

LEN ノード

説明 3

パートナー LU の構成 91

リモート・ノードの識別 84

LU

構成 91, 94, 105

サポート 4

ダウンストリーム 98, 100

パートナーの定義 91

プール 96, 97, 105, 106

命名規則 24

LU タイプ 0-3 86

LU プール

構成 87

定義 87

表示 87

M

MAC (メディア・アクセス制御) 84

MDS-NMVT (マルチドメイン・サポート・ネットワーク管理ベ
クトル・トランスポート) 17

Motif 管理プログラム

管理機能 17

使用 71

説明 15

ヘルプ 15

N

NOF API 15

O

OSI (オープン・システム間相互接続) 24

P

PDF 資料の表示 34

PU からの集線 98

PU コンセントレーター 8

PU (物理装置) 8

Q

query コマンド 17

R

Remote API Client

AIX ソフトウェア要件 51

AIX ハードウェア要件 51

Linux ソフトウェア要件 40

Linux ハードウェア要件 39

System z 用 Linux ソフトウェア要件 45

System z 用 Linux ハードウェア要件 45

Remote API Client on AIX のアンインストール 55

Remote API Client on AIX のインストール 52, 53

Remote API Client on Linux

既存のパッケージの詳細 40

Remote API Client on Linux のアンインストール 44

Remote API Client on Linux のインストール 41

Remote API Client on Windows

インストール 58

コマンド行からのインストール 63

GSKIT のアンインストール 69

Setup プログラムを使用したインストール 59

rpm 21

S

SAA (システム・アプリケーション体系) 6

SAP (サービス・アクセス・ポイント) 22, 83

SDK ソフトウェア

Remote API Client on Windows 58

SDLC 3

構成 82

従属トラフィック用 82

Secure Sockets Layer (SSL)

クライアント認証 107, 109

サーバー認証 34, 107, 109

データ暗号化 34, 107, 109

SNA

ライブラリー 113

SNA ゲートウェイ

概要 8

構成 98

snaadmin プログラム 15

snastart ファイル 74, 111

status コマンド 17

System z に対する Remote API Client on Linux のインストール
47

System z 用の Remote API Client on Linux

既存のパッケージの詳細 46

System z 用の Remote API Client on Linux のアンインストー
ル 50

T

TN サーバー

アクセス・レコードの構成 107

アソシエーション・レコードの構成 107, 108

概要 11

TN サーバー (続き)
構成 104
デフォルト構成 107
複数セッションのサポート 105
ユーザー 104
TN リダイレクター
アクセス・レコードの構成 109
概要 13
構成 108
デフォルト構成 109
TN3270
サーバー 11
プログラム 11
TN3270 プログラム 104
TN3270 ユーザー 12, 104
TP (トランザクション・プログラム) 5

U

U 型セッション 5
UDP ブロードキャスト・パラメーター 61

V

verb 5

W

WAN (広域ネットワーク) 3
WebSphere Application Server
構成 31
要件 23

X

xснаadmin プログラム 15, 71



プログラム番号: 5724-i34

Printed in Japan

GC88-9997-01



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12