

IBM Communications Server for Data Center
Deployment on Linux
バージョン 7.1

診断ガイド



注記

本書および本書で紹介する製品をご使用になる前に、[69 ページの『付録 F 特記事項』](#)に記載されている情報をお読みください。

第 7 版 (2021 年 1 月)

本書は、IBM Communications Server for Data Center Deployment on Linux バージョン 7.1 (プログラム番号 5725-H32) に適用されます。また、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。

資料のご注文方法については、<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)

IBM にお客様のご意見をお寄せください。本資料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> フォームが削除されている場合は、以下の住所までご意見をお送りください。

- International Business Machines Corporation
- Department CGMD
- P.O. Box 12195
- Research Triangle Park, North Carolina
- 27709-2195
- U.S.A.

ファックスまたはインターネットをご使用の場合は、下記宛てにお願いします。

- IBMLink: RALVM17 の CIBMORCF
- IBM Mail: IBMMAIL の USIB2HPD
- インターネット: USIB2HPD@vnet.ibm.com
- FAX: 1-800-227-5088

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。

© Copyright International Business Machines Corporation 1998, 2021.

目次

表.....	v
図.....	vii
本書について.....	ix
本書の対象読者.....	ix
本書の使用法.....	ix
本書の構成.....	ix
表記上の規則.....	x
本書の追加情報の入手先.....	xi
第 1 章問題解決の概要.....	1
診断情報のタイプ.....	1
プログラム・エラー・メッセージ.....	2
ログ・メッセージ.....	2
トレースの紹介.....	4
アラート.....	6
ヘルプ情報.....	6
第 2 章一般的な問題の解決.....	9
基本的な確認事項.....	9
最初に確認する事項.....	9
SNA デーモンが開始済みであることの確認.....	9
ローカル SNA ノードが活動状態であることの確認.....	10
その他の SNA ノードとの通信の確認.....	10
AIX または Linux リモート API クライアントが自分のサーバーを認識できるかどうかの確認.....	14
Windows におけるリモート API クライアントが自分のサーバーを認識できるかどうかの確認.....	15
システム構成情報の確認.....	16
特定の問題の解決.....	17
Motif を使用するプログラムに関する問題の解決.....	17
APPC または CPI-C アプリケーションの問題の解決.....	17
LUA 問題の解決.....	18
MS アプリケーションの問題の解決.....	19
NOF アプリケーションの問題の解決.....	19
TN Server での問題の解決.....	19
TN リダイレクターでの問題の解決.....	20
ネットワーク・ノード・セッションのルーティング問題の解決.....	21
SNA ゲートウェイ・セッションの問題の解決.....	21
サーバー管理問題の解決.....	21
オンライン・サポート情報.....	22
サポート担当員への問題の報告.....	22
サポート担当員のタイプ.....	22
担当者に提供する情報.....	22
第 3 章ログ記録とトレースの使用法.....	25
Motif 管理プログラムによるログ記録の制御.....	25
コマンド行管理プログラムによるログ記録の制御.....	26
ログ・ファイルの名前と位置の変更.....	26
中央ロギングまたはローカル・ロギングの使用可能化.....	26

ログに記録するメッセージの指定.....	27
ログ・ファイル・サイズの制御.....	27
ログの形式の選択.....	27
ロギングのフィルター処理.....	31
使用量のログ記録.....	32
使用量ログ・ファイル.....	32
ライセンス・マネージャーのトラッキング.....	33
リソースの使用量を確認するための管理ツールの使用.....	33
トレース.....	33
回線トレース.....	34
API トレース.....	37
クライアント/サーバー・トレース.....	43
TN Server トレース.....	45
内部トレース.....	46
付録 A snafilter と snatrcfmt の使用方法.....	49
バイナリー・トレースのフィルター処理.....	49
snafilter ユーティリティの実行.....	49
コマンドと出力の例.....	52
内部バイナリー・トレース出力をテキスト・ファイルにフォーマットする.....	52
回線トレース用の snatrcfmt ユーティリティの実行方法.....	53
内部トレース用の snatrcfmt ユーティリティの実行方法.....	53
snatrcfmt ユーティリティの出力.....	55
付録 B getsense の使用法.....	61
付録 C snagetpd の使用法.....	63
snagetpd の実行方法.....	63
コマンド構文とプログラム出力.....	63
コマンドの制限.....	64
付録 D Windows クライアント.....	65
Windows クライアントのロギング.....	65
Windows クライアントのトレースの制御.....	65
Windows クライアントの診断情報の収集.....	66
付録 E IBM へのご意見の送付方法.....	67
E メール・フィードバック・テンプレート.....	67
技術的な問題がある場合.....	67
付録 F 特記事項.....	69
商標.....	70
参考文献.....	73
CS Linux バージョン 7.1 の資料.....	73
システム・ネットワーク体系 (SNA) 関連資料.....	74
ホスト構成関連資料.....	74
z/OS Communications Server 関連資料.....	75
TCP/IP 関連資料.....	75
X.25 関連資料.....	75
APPC 関連資料.....	75
プログラミング関連資料.....	75
その他の IBM ネットワーキング・トピック.....	75
索引.....	77

表

1. 表記上の規則.....X



1. インターフェースのトレース	34
------------------------	----

本書について

本書では、IBM Communications Server for Data Center Deployment on Linux の使用時に発生する可能性のある最も一般的な問題について説明し、解決方法を示します。また、利用できる診断ツールと、サポート担当員に提出する診断データの収集方法についても説明します。

IBM Communications Server for Data Center Deployment on Linux (プログラム・プロダクト番号 5725-H32) は、Linux が稼働するコンピューターが SNA ネットワーク上にある他のノードと情報を交換できるようにする IBM® のソフトウェア製品です。

IBM Communications Server for Data Center Deployment on Linux には、この製品が動作するハードウェアに応じて、2つの異なるインストール方法があります。

CS Linux

CS Linux は、次のハードウェアで動作します。

- Linux が稼働する 64 ビット AMD64/Intel EM64T ワークステーション (x86_64)
- Linux が稼働する IBM Power コンピューター (ppc64le)

CS Linux for IBM Z

CS Linux for IBM Z は、Linux for IBM Z が稼働する IBM Z メインフレーム (s390x) で動作します。

本書では、相違が明示的に記述されていない限り、CS Linux という名称はこれらの2つのバリエーションのいずれかを示すために使用され、「CS Linux コンピューター」という用語は、CS Linux が稼働しているすべてのコンピューターの種類を示す場合に使用されます。

本書は CS Linux のバージョン 7.1 に適用されます。

本書の対象読者

本書は CS Linux を使用するシステム管理者およびアプリケーション・プログラマーを対象としています。

システム管理者

システム管理者は、CS Linux をインストールし、システムをネットワーク接続用に構成し、システムの保守を行います。システム管理者は CS Linux システムおよび CS Linux を実行するハードウェアに精通していなければなりません。また、システムの接続先ネットワークに関して知識があり、SNA の概念も理解している必要があります。

アプリケーション・プログラマー

アプリケーション・プログラマーは、CS Linux プログラミング・インターフェースを使用して SNA ネットワーク上でデータを送受信するトランザクション・プログラムおよびアプリケーション・プログラムの設計とコーディングを行います。また、SNA、トランザクション・プログラムまたはアプリケーション・プログラムの通信相手となるリモート・プログラム、および CS Linux システムのプログラミングおよび稼働環境についても精通していなければなりません。

アプリケーション・プログラムの作成についての詳細は、各 API の資料にそれぞれ記載されています。CS Linux の関連資料に関する追加情報については、『参考文献』の節を参照してください。

本書の使用法

この節では、本書の構成と表記について説明します。

本書の構成

本書は次の各章で構成されています。

- [1 ページの『第 1 章 問題解決の概要』](#)では、利用できる診断ツールとその重要性、および、それをいつ使用するかについて説明します。

- 9 ページの『[第 2 章 一般的な問題の解決](#)』では、発生する可能性の最も高い問題について説明し、問題解決の方法を順を追って説明します。
- 25 ページの『[第 3 章 ログ記録とトレースの使用法](#)』では、ロギングとトレースについて作業本位に説明し、システムからの情報の収集方法を提供します。また、最も頻繁に使用するトレースの実行方法についても詳しく説明します。
- 49 ページの『[付録 A snafilter と snatrcfmt の使用方法](#)』では、snafilter ユーティリティおよび snatrcfmt ユーティリティを使用してバイナリー・フォーマットでトレースを出力する方法について説明します。
- 61 ページの『[付録 B getsense の使用法](#)』では、getsense ユーティリティを使用してセンス・コードをオンラインで表示する方法について説明します。
- 63 ページの『[付録 C snagetpd の使用法](#)』では、診断情報収集ユーティリティ (snagetpd) について説明し、サポート担当員に提出する診断情報をこのユーティリティで収集する方法について説明します。
- 65 ページの『[付録 D Windows クライアント](#)』は CS Linux ユーザー用の Windows 固有の情報を含んでいます。

表記上の規則

本書では、[x ページの表 1](#) に示すような、表記上の規則を使用します。

表 1. 表記上の規則

内容	表記例
資料名	<i>IBM Communications Server for Data Center Deployment on Linux 管理ガイド</i>
ファイル名またはパス名	/var/opt/ibm/sna/sna.err
プログラムまたはアプリケーション	vi
コマンドまたは Linux ユーティリティ	define_default_pu
特定のタイプのすべてのコマンドへの一般的な参照	define_* (定義用管理コマンドすべてを指す)
オプションまたはフラグ	ALL
パラメーターまたは Motif フィールド	log_file_type
リテラル値またはユーザーが入力できる選択項目 (デフォルト値を含む)	USER, NODE
定数またはシグナル	ERROR
戻り値	Audit
指定する値を表す変数	server name
環境変数	\$DISPLAY
プログラミング verb	REGISTER_NMVT_APPLICATION
ユーザー入力	xснаadmin
コンピューターからの出力	+RSP
関数、コール、またはエントリー・ポイント	Set Session Parameters
Motif ボタン	「状況 (Status)」
Motif メニュー	「サービス (Services)」
Motif メニュー項目	「ノード・パラメーターの構成 (Configure node parameters)」

表 1. 表記上の規則 (続き)

内容	表記例
キーボード・キー	Enter
16 進値	0x0a

本書の追加情報の入手先

CS Linux ライブラリーに含まれている他の資料、および SNA と Linux ワークステーションに関連するトピックに関する追加情報が記載されている資料については、『参考文献』の節を参照してください。

第1章 問題解決の概要

CS Linux は、複合ソフトウェア製品です。したがって、CS Linux を稼働しているときに、CS Linux 自体または他のシステム・コンポーネントで問題が発生することがあります。

本書では、発生する可能性のある問題のうち、いくつかの一般的な問題について説明し、問題の調査方法を示し、詳細な診断情報の収集方法について説明します。本書は次のように構成されています。

- この章では、利用できる診断情報のタイプを示し、各情報の使用方法について説明します。
- 9 ページの『第2章 一般的な問題の解決』では、常に実行する必要がある基本チェックについて説明し、特定の問題をさらに詳しく調査する方法を順を追って説明します。発生した問題の調査方法について知りたい場合には、この章を参照してください。
- 25 ページの『第3章 ログ記録とトレースの使用法』では、CS Linux のトレース機能とログ記録機能を使用して、詳細な診断情報を収集する方法について説明します。特定のログまたは特定のトレースの収集方法についてのガイドが必要な場合には、この章を参照してください。

診断情報のタイプ

この節では、CS Linux システムの問題を解決するために利用できるさまざまな診断情報、および各情報の使用方法について説明します。

CS Linux 診断情報は、次のように分類されます。

- 「通知」情報は、いつでも使用できる情報です。オンまたはオフにすることはできません。通知情報は、エラーが発生し、調査する必要があることを示します。通知情報には、エラー・メッセージ、エラー・ログ、およびアラートがあります。
- 「診断」情報は、制御可能な情報です。診断情報は特定の問題に関してさらに詳細な情報を収集するために使用します。診断情報には、例外ログ、監査ログ、およびトレース・データがあります。

CS Linux は、問題の予防と解決に役立つオンライン・ヘルプ情報も備えています。

CS Linux の稼働時には発生した問題を解決するため、通常、さまざまな状況下でこれらの情報すべてを使用します。

たとえば、APPC アプリケーション・プログラムの実行中に、セッションの開始や停止などのような他の CS Linux イベントが発生することがあります。各イベントは、さらに小さな多数のイベントからなります。このため、セッション・イベントには、LU への接続、セッションの開始、セキュリティ検査、リンク初期化などのような内部イベントが含まれていることがあります。このような通常の小さいイベントのログをすべて記録する場合には、これらのログを監査ログ・ファイルと呼ばれるファイルに記録するように CS Linux を構成できます。

プログラムに障害が発生すると、CS Linux は診断用のリソースを数種類提供します。これらの診断用リソースは発生したイベントのそれぞれのタイプやレベルについての情報です。この情報は画面に表示できるほか、エラー・ログ・ファイルと呼ばれるファイルに記録できます。CS Linux の特定の機能についてトレースを活動化することにより生成され、ファイルに収集されるイベント情報もあります。このようなイベント情報は CS Linux の特定領域の情報です。

この項では、各種の情報と情報の使用方法について説明します。

プログラム・エラー・メッセージの情報

プログラム・エラー・メッセージは、重大なシステムの問題が検出されると必ず表示されます。

CS Linux ログ・メッセージの情報

ログ・メッセージには、プログラム・イベントに関する情報が含まれています。イベントはログに記録される情報により、問題、例外、監査の3つのタイプに分けられます。それぞれのイベントはログ・ファイルに記録されており、このログ・ファイルにアクセスして、問題解決に必要な情報を入手することができます。

CS Linux は、ローカル・ノードにおける SNA リソースの使用量に関する情報を記録する別個のログ・ファイルも保守します。詳細については、32 ページの『使用量のログ記録』を参照してください。

CS Linux トレースの情報

トレースとは、CS Linux の稼働中に CS Linux の特定の境界を越えて発生するイベントをトラッキングすることです。CS Linux は、診断のために活動化できる各種トレース・オプションを提供します。

アラートの情報

標準 SNA アラートが生成され、ホストへ送信されます。ホストでアラートを表示するには、NetView を使用します。

通常、プログラム・エラー・メッセージおよびログ・メッセージは問題の性質、原因、推奨処置を示しています。これらの情報だけで問題を解決できることがよくあります。利用できる情報量は、問題の性質とログ記録の設定方法によって異なります。例えば、以下のようになります。

- プログラム・エラー・メッセージは自動的に表示されます。このエラー・メッセージ機能を使用不可にすることはできません。
- 問題を示すイベント・ログも自動的に生成されます。この機能を使用不可にすることはできません。ただし、例外イベント・ログの記録は使用不可にできます。監査イベント・ログの記録は、ログを記録するように指定しない限り実行されません。

トレースの開始は、問題が発生した可能性がある場合のみ実行してください。そうすれば、問題のあると思われる領域でのアクティビティをトレースできます。さらに、収集するトレース情報の容量と形式を制御できます。

診断情報以外に、CS Linux は、次のプログラムに関する拡張オンライン・ヘルプ情報も備えています。

- Motif 管理プログラム
- コマンド行管理プログラム

この章のこれ以降は、各診断情報についての詳しい説明と、CS Linux オンライン・ヘルプ情報の参照方法についての説明です。

ユーザーが体験する一般的な問題の一覧とその解決手順については、[9 ページの『第 2 章 一般的な問題の解決』](#)を参照してください。ログ記録とトレースの使用法の詳細については、[25 ページの『第 3 章 ログ記録とトレースの使用法』](#)を参照してください。

プログラム・エラー・メッセージ

問題が発生したことが最も明確に分かるのが、プログラム・エラー・メッセージが表示されたときです。CS Linux では重大な問題を報告するために、プログラム・エラー・メッセージが自動的に生成されます。プログラム・エラー・メッセージは、画面またはコンソール上で報告されます。各メッセージは、プログラムに発生した問題について説明します。このメッセージ機能は使用不可にできません。

たとえば、サーバー上にノード構成ファイルが存在しないのに、管理者がコマンド行管理プログラムを使用してノードを開始しようとした場合は、次のようなメッセージが表示されます。

```
$: snaadmin init_node
init_node command failed:
primary_rc = STATE_CHECK, secondary_rc = NODE_NOT_CONFIGURED
```

ノードを実行中に、管理者が Motif 管理プログラムを使用して「ノード・パラメーター (Node Parameters)」ウィンドウでパラメーターを変更しようとした場合、次のようなポップアップ・メッセージが表示されます。

```
You cannot modify the node's parameters while it is not inactive.
```

プログラム・エラー・メッセージと共に、追加情報を示すログ・メッセージが表示されることがよくあります。エラー・ログ・ファイルの詳細については、[3 ページの『ログ情報のタイプ』](#)を参照してください。

ログ・メッセージ

プログラムの実行中には、セッションの開始や終了のような様々なイベントが発生します。CS Linux では、CS Linux の内部アクティビティに関する特定の情報を提供するため、このような様々なイベントに関するログ・メッセージやその他のイベントがログ・ファイルに記録されます。ログに記録されるその他の CS

Linux のその他の内部アクティビティには、ポートの初期化、セキュリティ検査、およびネットワーク・リンク・ステーションの障害があります。

CS Linux では、通常のイベント (セッション開始など) および異常なイベント (予期しないセッションの終了やリソースの不足など) に関するメッセージがログに記録されます。各イベントごとに、発生したイベントの内容、発生時刻、発生位置がメッセージに記録されます。いくつかのタイプのログ記録機能を使用不可にしたり (27 ページの『[ログに記録するメッセージの指定](#)』を参照)、ログに記録される詳細情報の量を制御することができます (27 ページの『[ログの形式の選択](#)』を参照)。1つのログ・ファイルに複数のタイプのメッセージを記録できます。

ログ情報のタイプ

CS Linux では、イベントは重大度によって次の 3 つのタイプに分類されます。

問題

システムのパフォーマンスを低下させ、パフォーマンスが低下したことをユーザーがすぐに認識できる異常なシステム・イベント。(たとえば、セッションの異常終了など)

CS Linux では、このようなイベントのログは常に記録されます。問題イベントのログ記録機能は使用不可にできません。

例外

例外イベントは、次の 2 つの種類に分類されます。

- システム・パフォーマンスを低下させるが、パフォーマンスが低下したことをユーザーはただちに認識できない異常なシステム・イベント (リソースの不足など)。
- システム・パフォーマンスは低下させないが、後に例外や問題を引き起こす可能性があるイベント。(たとえば、リモート・システムからの予期しないメッセージの受信など)。

デフォルトでは、CS Linux により、例外イベントのログが記録されます。

Motif 管理プログラムを使用して例外イベントのログ記録を制御するには、[25 ページの『Motif 管理プログラムによるログ記録の制御』](#)を参照してください。

すべてのサーバーのグローバル・デフォルト設定値を確立する場合は、`snaadmin set_global_log_type` コマンドを使用して、また、特定のサーバー・デフォルト値をオーバーライドする場合は、`snaadmin set_log_type` コマンドを使用して、これらのイベントのログ記録を制御することもできます。

管理コマンドを使用したログ記録の制御について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理コマンド解説書*」を参照してください。

監査

通常のシステム・イベント (セッションの開始など)。デフォルトで、CS Linux は監査イベントのログを記録しません。

Motif 管理プログラムを使用して監査イベントのログ記録を制御するには、[25 ページの『Motif 管理プログラムによるログ記録の制御』](#)を参照してください。

`snaadminset_global_log_type` コマンド、または `snaadmin set_log_type` コマンドを使用しても、これらのイベントのログ記録の制御を行うことができます。

修正処置が必要になることがある一部の問題メッセージおよび例外メッセージは、ログ・ファイルに記録されるだけでなく、Linux システム・コンソールにも表示されます。これらのコンソール・メッセージを表示しないようにするには、CS Linux ソフトウェアの開始時に `-s` オプションを使用します (詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください)。これにより、メッセージはログ・ファイルにのみ書き込まれるようになります。

ログ情報の使用法

一般に、CS Linux の使用中に発生した問題を解決するには、最初にエラー・ログ・ファイルの内容を調べる必要があります。

ログ・ファイルには、「メッセージに対する処置 (*Message action*)」フィールドが含まれています。このフィールドには、メッセージの結果としての推奨処置が記載されています。場合によっては処置を必要としないこともあります。たとえば、例外メッセージの情報がエラーを示してはいないものの、その後で、問題メッセージが出された場合にその原因を識別する際に役立つバックグラウンド情報を示していることがあります。

一般的な推奨処置には次のようなものがあります。

- ローカル CS Linux を確認して、リソースの追加、変更、活動化のいずれかを行う。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理コマンド解説書*」を参照してください。
- Linux コンピューターのリソース (メモリー、ハード・ディスク・スペース、アダプター・カードなど) を調べます。詳細については、使用している Linux オペレーティング・システムの文書を参照してください。
- 構成不一致の問題を解決するには、CS Linux の通信先システムのサポート 担当員に連絡します。詳しくは、[22 ページの『サポート担当員のタイプ』](#)を参照してください。
- CS Linux API を使用するアプリケーションから無効な API 呼び出しが実行された場合は、アプリケーションの開発者に連絡します。
- *Cause type* フィールドが CS Linux ソフトウェアの内部エラーを示している場合は、エラー状態をサポート担当員まで報告します。
- ローカル・システムおよびリモート・サーバーのロギング情報をチェックします。

ログの情報が想定していたよりも少ない場合には、簡略ロギングを使用可能にしていた可能性があります (使用可能なモードは簡略ロギングと詳細ロギングの 2 つです)。欠落した情報を回復するには、`snahelp` ユーティリティーを使用してください。(30 ページの『簡略ロギング・メッセージに `snahelp` を使用する方法』を参照してください。) 詳細ロギングの使用可能化については、[27 ページの『ログの形式の選択』](#)を参照してください。

トレースの紹介

CS Linux は、CS Linux の実行中に、CS Linux の内部アクティビティーに関する情報を収集および取得できるようにするためのトレース機能を提供します。この情報は、特定の問題を診断するのに役立ちます。例えば、DLC (データ・リンク制御) の問題が疑われる場合には、特定の DLC の回線トレースを活動化することができます。

管理プログラムを使用して、収集されるトレース・データのタイプおよび量を制御することができます。実行するトレースのタイプに応じて、出力は ASCII フォーマットまたはバイナリー・フォーマットのいずれかで生成されます。テキスト・エディターを使用して ASCII ファイルを表示します。バイナリー・データの場合は、バイナリー・ファイルをテキスト・ファイルに変換するためのスナット `cfmt` ユーティリティー。詳細については、[52 ページの『内部バイナリー・トレース出力をテキスト・ファイルにフォーマットする』](#)を参照してください。

以下のセクションでは、トレースのタイプについて簡単に説明し、トレースを使用する場合の例を示します。トレースの詳細については、[33 ページの『トレース』](#)を参照してください。

トレースのタイプ

CS Linux は、次のタイプのトレースを提供します。

回線トレース

回線トレースを使用して、ノードとリモート・システム間のメッセージをトレースします。リソース・タイプ (DLC、ポート、リンク・ステーション、またはセッション) を指定することによって、トレースの量を制御することができます。行トレースは、「DLC トレース」と呼ばれることもあります。

通常、回線トレース出力は、SNA 管理者がエンド・ユーザーの問題を解決するために使用します。これには、セッションまたはセッションの障害を起こすことができなくなります。形式は標準 SNA トレースです。回線トレースについて詳しくは、[34 ページの『回線トレース』](#)を参照してください。

回線トレースによって生成される出力は 2 進数です。以下を使用すると、行トレース・ファイルから特定の項目を選択できます。スナフイーター スナット `cfmt` ユーティリティーを使用して、バイナリー・トレース出力をテキスト・ファイルにフォーマット設定することができます。これらのユーティ

リティーについて詳しくは、[49 ページの『付録 A snafilter と snatrcfmt の使用方法』](#)を参照してください。

API トレース

API トレースを使用して、以下のいずれかの CS Linux API(拡張プログラム間通信機能 (APPC)、通信用共通プログラミング・インターフェース (CPI-C)、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 (LUA)、ノード・オペレーター機能 (NOF)、管理サービス (MS)、および共通サービス verb (CSV)) に関連する通信の問題を特定します。API トレースは、アプリケーションによって API ライブラリーまたはドライバーに提供されたすべてのパラメーター、および API ライブラリーによって戻されるすべてのパラメーターをトレースします。

API トレース・データはテキスト・ファイルに書き込まれます。詳しくは、[37 ページの『API トレース』](#)を参照してください。

クライアント/サーバーのトレース

クライアント / サーバー・トレースを使用して、CS Linux サーバーとクライアント間のメッセージ・フローをトレースし、同じドメイン内の CS Linux サーバー間でトレースすることができます。クライアント / サーバー・トレースを使用して、例えば、クライアントがサーバーに接続できない理由を判別することができます。

クライアント/サーバーのトレースは、テキスト・ファイルに書き込まれます。詳しくは、[43 ページの『クライアント/サーバー・トレース』](#)を参照してください。

TN サーバーのトレース

TN サーバー・トレースを使用して、CS Linux TN サーバーとその TN3270 クライアントの間を流れるメッセージを記録します。通常、システム管理者は、このトレースを開始して TN3270 関連の問題を解決します。

TN サーバーのトレース・データは、テキスト・ファイルに書き込まれます。詳しくは、[45 ページの『TN Server トレース』](#)を参照してください。

内部トレース

内部トレースを使用して、CS Linux ノードの内部コンポーネント間で流れるメッセージをトレースします。

通常は、大量のトレース出力がすぐに生成されます。問題を解決するために必要な情報を収集したら、トレースを停止して、ファイルが大きくなりすぎないようにしたり、上書きされたりすることがないようにします。内部トレースについて詳しくは、[46 ページの『内部トレース』](#)を参照してください。

内部トレース (通常、サポート担当者によって使用される) によって生成される出力はバイナリーです。スナフィルターユーティリティーを使用して、内部トレース・ファイルから特定の項目を選択することができます。また、スナット cfmt ユーティリティーを使用して、バイナリー・トレース出力をテキスト・ファイルにフォーマットすることができます。これらのユーティリティーについて詳しくは、[49 ページの『付録 A snafilter と snatrcfmt の使用方法』](#)を参照してください。

トレースの使用法

問題が発生した場合に CS Linux のトレース機能を使用すると、特定のインターフェース間で流れるメッセージに関するより詳細な情報を入手できます。例えば、以下のようになります。

- CS Linux API を使用するアプリケーションがあり、API 戻りコードで問題が示されている場合は、API トレースを使用します。
- リモート・システムに正常に接続できない場合、またはリモート・システムに接続しようとして CS Linux で例外ログおよび問題ログが生成される場合には、回線トレースまたはクライアント/サーバー・トレースを使用します。

注: CS Linux システムに問題がない場合は、CS Linux トレースを実行しないでください。トレースを使用可能にすると、システムの実行速度が遅くなります。

トレースの使用法の詳細については、[33 ページの『トレース』](#)を参照してください。

アラート

アラートは、自動生成される SNA メッセージです。アラートは外部ネットワーク上の複数のプログラムへ送信されて各プログラムで処理され、問題、または発生する可能性のある問題を識別するために使用されます。アラートは、接続コンポーネントや Management Services API を使用するアプリケーション・プログラムから出される可能性があります。

アラートは、検出された最初の活動状態であるホスト・リンク、または `snaadmin define_default_pu` コマンドにより指定されたリンク・ステーションへ送信されます。このリンク・ステーションが活動状態でない場合、アラートはディスクに保管され、リンク・ステーションが再度活動状態になった時点で送信されます。

アラートの表示と処理のために使用される最も一般的なプログラムは、通常ホスト上で稼働している NetView です。

ヘルプ情報

問題発生時に実行されていた CS Linux システム・アクティビティーに関するログ情報とトレース情報以外に、CS Linux には標準オンライン・ヘルプ情報もあります。オンライン・ヘルプの情報は、特定のプログラムを使用中に問題が発生した場合や、特定のトピックやコマンドの使用法について知りたい場合に役立ちます。

CS Linux には次のオンライン・ヘルプ情報があります。

- Motif 管理プログラムのオンライン・ヘルプ
- コマンド行管理プログラム・ヘルプ
- PDF 形式の CS Linux マニュアル
- man ページ
- 使用方法のヘルプ

Motif 管理プログラムのオンライン・ヘルプ

CS Linux Motif 管理プログラム (CS Linux の構成と管理に使用) には、オンライン・ヘルプが準備されています。オンライン・ヘルプにアクセスするには次の 2 つの方法があります。この 2 つの方法の違いは、ヘルプ情報の形式です。

- メイン・ウィンドウからは広範囲なヘルプ・トピックにアクセスできます。
 1. オンライン・ヘルプ・メニューを表示するには、管理ウィンドウの右上隅にある「ヘルプ (Help)」をクリックします。
 2. 「ヘルプ (Help)」メニューから「目次 (Contents)」を選択します。メイン・オンライン・ヘルプ・ダイアログにトピックのリストが表示されます。
- 各ダイアログから、そのダイアログに関するヘルプ情報にアクセスできます。
 1. ダイアログ・フィールドの詳細について知りたい場合には、「ヘルプ (Help)」ボタンをクリックします。

コマンド行管理ヘルプ

コマンドに関する管理コマンド行ヘルプへは、コマンド行管理プログラムからアクセスできます。コマンド行ヘルプでは、`snaadmin` コマンドについて、パラメーターとデフォルト値を含む完全な構文が説明されています。例えば、次のコマンドを入力すると、使用可能なヘルプ情報が表示されます。

```
snaadmin -h
```

PDF 形式の CS Linux マニュアル

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。PDF では、ハイパーテキスト・リンクを使用して関連情報を得ることができ、情報をより簡単に検索、表示、または印刷で

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。

PDF マニュアルは任意の PDF ビューアーを使用して読むことができます。たとえば、Linux 上で PDF ファイルを表示するには `xpdf` を使用できます。

```
xpdf filename.pdf
```

man ページ

man ページには、コマンドの使用方法に関する情報があります。トピックの概要や、特定のコマンドの使用法がすぐに必要な場合は、それらにアクセスしてください。

コマンド行から man ページにアクセスするには、`man - M /opt/ibm/sna/man pagename` と入力します。*pagename* は表示するページの名前を表します (一般には CS Linux コマンドの名前です)。例えば、man ページ「sna」を参照するには、`man - M /opt/ibm/sna/man sna` と入力します。

man ページでは、要求された情報だけでなく、関連コマンドのリストも表示されます。

使用方法のヘルプ

CS Linux プログラムの使用方法のヘルプを見るには、`-h` パラメーターを使用してプログラムを起動します。これにより、簡潔な構文ステートメントを取得できます。例えば、次のように入力します。

```
sna -h
```


第 2 章 一般的な問題の解決

この章では、CS Linux の稼働中に発生する可能性のある一般的な問題のいくつかを説明します。この章で説明する手順は、このような問題を解決する際に役立ちます。

この章では次の事項について説明します。

- 問題が発生した場合に必ず実行する必要がある基本的な確認事項。
- 特定の問題と、その問題をさらに詳しく調査するための手順。
- CS Linux のオンライン・サポート情報。
- この章で説明する手順を行っても解決できない問題を、サポート部門へ報告する手順。

一部の確認では、Motif 管理プログラムまたは コマンド行管理プログラムを使用して構成情報を調べる必要があります。本書では確認事項について説明していますが、確認方法の詳細については次の資料を参照してください。

- Motif 管理プログラムを使用している場合、特定のタスクおよびフィールドの詳細については 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。
- コマンド行管理プログラムを使用している場合、コマンドとそのパラメーターの詳細については、コマンド行管理プログラムのオンライン・ヘルプおよび「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

基本的な確認事項

問題が発生した場合には、次に示す基本的な確認事項のチェックを必ず行ってください。この節に記載されている各確認事項をすべて行ってから、[17 ページの『特定の問題の解決』](#)に進みます。

最初に確認する事項

SNA または CS Linux の問題であると想定する前に、次の事項を少し時間を割いて確認してください。

1. すべての通信ケーブルが、マシン、スイッチ、ハブに正しく接続されている。
2. モデム、スイッチ、ハブはすべて使用可能な状態になっている。
3. 通信しようとしている相手のリモート・システムが活動状態になっている。
4. 使用しているソフトウェアとオプションのソフトウェア・パッケージがすべて正しくインストールされている。
5. 該当するソフトウェアのすべての修正 (fix) またはパッチが適用済みである。
6. すでに認識されている問題についての情報は、使用しているシステムのインストール用資料を参照する。

SNA デーモンが開始済みであることの確認

管理ツールをはじめとする CS Linux の機能を使用する前に、SNA デーモン・プログラムを開始する必要があります。

CS Linux が正しくインストールされている場合には、Linux の再始動時に必ず SNA デーモンが自動的に開始されます。

SNA デーモンを開始する (またはすでに実行中であるかどうかを調べる) には、`sna start` コマンドを発行します。このコマンドにより、次の SNA デーモンが開始されます。

- `snadaemon`
- `snaerrlog`
- `snacfgdae`

存在しない構成レコードへの参照など構成にエラーがあると、SNA デーモンが開始できない場合があります。CS Linux が正しくインストールされていない場合、他のエラーが起きることがあります。例えば、以下ようになります。

- ドメイン構成ファイル (sna_domn.cfg) がない。
- LiS ストリームなどの必要なコンポーネントがインストールされていない。
- CS Linux が正常にインストールされていない。

詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」を参照してください。表示されるエラー・メッセージのヘルプについては、CS 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。

ローカル SNA ノードが活動状態であることの確認

CS Linux の全機能を有効にするには、ローカル SNA ノードを活動化していなければなりません。ノードが活動状態であるかどうかを確認するには、snaadmin status_node コマンドを使用します。このコマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください。

このコマンドは、次の事項を報告します。

- ノードが活動状態である。
- ノードが活動状態でない。

ローカル SNA ノードが活動状態である場合

snaadmin status_node コマンドでノードが活動状態であると報告される場合は、ノードは正しく活動状態になっています。

ローカル SNA ノードが非アクティブです。

If the snaadmin status_node command reports that the node is inactive then you must activate the node using either the Motif administration program or the command-line administration program. ノードの開始については、*IBM Communications Server for Linux 管理ガイド*上のデータ・センター・デプロイメントを参照してください。

ノードが開始に失敗した場合は、エラー・ログ・ファイルを確認してください。エラー・ログ・ファイルは、構成ファイル内にエラーがあることを示している可能性があります。この場合は、Motif 管理プログラムまたはコマンド行管理プログラムを使用して、エラーを訂正してください。

例えば、snaadmin ノードの define_node コマンドは、topology.dat (トポロジー) ファイルを削除せずにネットワーク・ノードからエンド・ノードに変更されました。その後、snaadmin init_node コマンドが発行されたときにエラーが報告されます。の ノード・タイプ パラメーターの値が次の値の場合は、

ノードの構成について詳しくは、*IBM Communications Server for Linux 管理ガイド*上のデータ・センター・デプロイメントを参照してください。

その他の SNA ノードとの通信の確認

その他の SNA ノードとの通信路を確認するには、次の事項を確認してください。

1. 他の SNA ノードとのリンクが確立している。
 - Motif 管理プログラムを使用している場合は、次の事項も確認してください。
 - a. 使用しようとするポートが、活動状態にマークされている。
 - b. 使用しようとしているリンク・ステーションが、活動状態にマークされている。リンク・ステーションを始動する時期は、次に示す構成値によって異なります。
 - 初期設定 (Initially)

リンク・ステーションはノード開始時に始動します。

– 管理者 (Administrator)

リンク・ステーションは管理者が始動します。(これは始動シェル・スクリプトに記述された `snaadmin start_ls` コマンドの場合があります。)

– 着呼 (Incoming)

リンク・ステーションはリモート・ノードにより始動されます。リモート・ノードがリンク・ステーションの始動を試行したことを確認してください。

– オンデマンド (On demand)

リンク・ステーションは、アプリケーションまたは端末エミュレーターから要求される場合にのみ始動します。(この場合、LS を明示的に起動して LS をテストするために、この値を変更することも可能です。これについては以降で説明します。)

- コマンド行管理プログラムを使用している場合は、`snaadmin status_connectivity` コマンドを入力して、状況を表示します。このコマンドの発行について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

Motif インターフェース・ウィンドウに表示される `snaadmin status_connectivity` コマンドの出力には、「DLC」によりグループ化されたポートが表示されます。各 DLC は 1 つ以上のポートをサポートする物理通信アダプターを表します。ポートが定義、開始されると、Motif 管理プログラムにより DLC コンポーネントが自動的に定義、開始されます。ただしコマンド行管理プログラムを使用している場合には、ポートが属する DLC を明示的に定義して開始する必要があります。

2. 他の SNA ノードとのリンクが確立されていない場合には、DLC/ポートまたはリンク・ステーションを起動します。DLC/ポートおよびリンク・ステーションの始動について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理ガイド」を参照してください。
3. DLC/ポートまたは LS が活動状態になっていない場合は、以下のセクションを参照してください。

DLC/ポートを始動できない

DLC/ポート を始動できない場合は、エラー・ログ・ファイルを調べてください。

さらに、CS Linux のエラー・ログ・ファイルも確認してください。このファイルは、通常、`/var/opt/ibm/sna/sna.err` です (管理ツールを使用して、別のファイル名またはディレクトリーを指定しなかった場合)。

それでも問題を解決できない場合には、次の事項を確認して、問題の共通原因として考えられることを見つけてください。

1. 必要なアダプター・カードと関連するデバイス・ドライバーがすべて正しくインストールされている。(製品に付属の資料を参照してください。)
2. 構成されたアダプター・カード番号 (パラメーター `adapter_number`) が、ハードウェアが使用しているアダプター・カード番号と一致している。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理ガイド」のポートと DLC の構成パラメーターの定義に関するセクションを参照してください。

DLC 構成を次のいずれかの方法で確認してください。

- Motif 管理プログラムを使用している場合は、DLC/ポートを調べます。
 - コマンド行管理プログラムを使用している場合は、`snaadmin query_dlc` コマンドを発行して、DLC 構成を表示します。
3. 複数のポートをサポートできるアダプター・カードを使用している場合は、構成されたポート番号 (パラメーター `port_number`) がハードウェアが使用しているアダプター番号と一致している。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理ガイド」のポートと DLC の構成パラメーターの定義に関するセクションを参照してください。

ポートの構成を次のいずれかの方法で確認してください。

- Motif 管理プログラムを使用している場合は、DLC/ポートを調べます。

- コマンド行管理プログラムを使用している場合は、`snaadmin query_port` コマンドを発行して、ポートの構成を表示します。
4. リモート機器を使用している場合は、該当するケーブルが差し込まれていて、接続しているリモート機器が活動状態である。
 5. SDLC 交換回線を使用している場合は、モデムにデータ・セット・レディー (DSR) およびキャリア検出 (CD) が表示されていることを確認してください。これらのシグナルが長い期間アクティブでない場合は、モデムか、または回線に問題がある可能性があります。
 6. MPC リンクを使用している場合は、チャンネル・アダプター・カードに提供されている構成および管理のツールを使用して、それがアクティブであることをチェックしてください。
 7. Enterprise Extender (HPR/IP) リンクを使用している場合は、ローカルの IP アドレスが CS Linux のポート構成とコンピューターの IP 構成の両方に正しく指定されているか確認してください。

リンク・ステーションを始動できない

リンク・ステーションを始動できない場合は、エラー・ログ・ファイルを調べてください。

さらに、CS Linux のエラー・ログ・ファイルも確認してください。このファイルは、通常、`/var/opt/ibm/sna/sna.err` です (管理ツールを使用して、別のファイル名またはディレクトリーを指定しなかった場合)。

それでも問題を解決できない場合には、次の事項を確認して、問題の共通原因として考えられることを見つけてください。

1. 中間ルーターまたはブリッジのみでなく、リモート・システムも活動状態であるか確認します。
2. リンク・ステーションの構成を次のいずれかの方法で調べます。
 - Motif 管理プログラムを使用している場合は、リンク・ステーションを調べます。
 - コマンド行管理プログラムを使用している場合は、`snaadmin query_ls` コマンドを発行して、リンク・ステーションの構成を表示します。

次の事項を確認します。

- a. 通信を行おうとしている 2 つのシステムのノード ID (パラメーター `node_id`) およびコントロール・ポイント名フィールド (パラメーター `adj_cp_name`) (XID で交換されます) が一致している。
- b. リンクの役割 (使用しているリンク・タイプに適用可能である、`ls_role` パラメーターの `primary`、`secondary` または `negotiable`) が、通信を行おうとしている 2 つのシステム間で互換性があることを確認します。 `primary` 対 `primary`、および `secondary` 対 `secondary` を除くすべての組み合わせが有効です。
3. それでもまだリンク・ステーションを始動できない場合は、使用しているリンク・タイプに応じて以降の項を確認します。

SDLC リンク・ステーションを始動できない

SDLC リンク・ステーションを始動できない場合は、[12 ページの『リンク・ステーションを始動できない』](#)に記載されている確認項目を実行してから、次の確認項目を実行してください。

1. 使用しているモデム構成を確認します。
 - a. CS Linux がデータ端末レディー (DTR) シグナルをオンにしていることを確認する。(この状態はモデムの状況ライトまたは表示で確認することができます)。CS Linux がデータ端末レディー (DTR) シグナルをオンにしていない場合は、カードとポートに構成されたポート番号が正しく設定されていることを確認します。
 - b. モデムが同期モードにプログラミングされていることを確認します (SDLC 通信には同期モードが必要です)。
 - c. ダイアル・ストリングが正しく構成されていることを確認します。
2. 次の構成パラメーターを確認します。
 - a. リンク・ステーションについて構成された `NRZ/NRZI` パラメーターが、リンクで使用される値に設定されていることを確認します。

- b. 全二重/半二重の固定キャリア構成設定値が、リンク用として正しいことを確認します。最新のモデムを装備したシステムでは、たいていは半二重および固定キャリアにセットアップされています。
- c. リンク・ステーションまたはポート構成において、リンク・アドレスが正しいことを確認します。このことは多重分岐の場合、特に重要です。

リンク・ステーションの構成を次のいずれかの方法で確認します。

- Motif 管理プログラムを使用している場合は、リンク・ステーションを調べます。
- コマンド行管理プログラムを使用している場合は、`snaadmin query_ls` コマンドを発行して、リンク・ステーションの構成を表示します。

QLLC リンク・ステーションを始動できない

QLLC リンク・ステーションを始動できない場合は、[12 ページの『リンク・ステーションを始動できない』](#)に記載されている確認項目を実行してから、次の確認項目を実行してください。さらに、基盤の X.25 ソフトウェアがアクティブであることも確認します。

スイッチド・バーチャル・サーキットを使用して QLLC リンク・ステーションを始動できない場合は、発呼要求パケット (リンク・ステーションを確立するために送信される) が誤りのパラメーターを伝送することがあります。したがって、以下の確認項目を実行する必要があります。

1. 次の構成パラメーターを確認します。
 - a. CS Linux が着呼呼び出しを受信するようにセットアップされている場合は、この呼び出しを CS Linux に経路指定するように X.25 ソフトウェアをセットアップします。リンク・ステーションのポートに対して、パラメーター `incoming_address` (使用している場合) がローカル・ノードの DTE アドレスと一致するよう正しく構成されていることを確認してください。

ポートの構成を次のいずれかの方法で確認してください。

 - Motif 管理プログラムを使用している場合は、ポートを調べます。
 - コマンド行管理プログラムを使用している場合は、`snaadmin query_port` コマンドを発行して、ポートの構成を表示します。
 - b. リモート・システムが特定の機能またはユーザー・データ・パラメーターを必要とするかどうかを確認します。必要な場合は、このリンク・ステーションに使用する CS Linux の構成で上記の機能またはパラメーターは適切に設定されていることを確認します。

CS Linux のリンク・ステーションの構成を次のいずれかの方法で確認します。

 - Motif 管理プログラムを使用している場合は、リンク・ステーションを調べます。
 - コマンド行管理プログラムを使用している場合は、`snaadmin query_ls` コマンドを発行して、リンク・ステーションの構成を表示します。
2. リモート DTE アドレスがリモート・システムのアドレスと一致していることを確認します。

トークンリング・リンク・ステーションまたはイーサネット・リンク・ステーションを始動できない

トークンリング・リンク・ステーションまたはイーサネット・リンク・ステーションを始動できない場合、まず [12 ページの『リンク・ステーションを始動できない』](#)に記載されている事項を確認してから、次に示す事項を確認してください。

次のリンク・ステーション・パラメーターを確認します。

1. `mac_address` パラメーターに構成されているリモート MAC アドレスがリモート・システムのアドレスと一致していることを確認します。
2. 次の手順に従って、ローカルおよびリモートの SAP 構成を確認します。
 - a. ローカル SAP (リンク・ステーションのポートのパラメーター `lsap_address` で構成されている) がリモート・システムのリモート SAP と一致していることを確認します。
 - b. リモート SAP (リンク・ステーションのパラメーター `lsap_address` で構成されている) がリモート・システムのローカル SAP と一致していることを確認します。

3. リンク・ステーションがイーサネット・リンク・ステーションの場合は、LAN のタイプ (802.3 あるいは標準) がリモート・システムの LAN のタイプと一致していることを確認します。

リンク・ステーションの構成を次のいずれかの方法で確認します。

- Motif 管理プログラムを使用している場合は、リンク・ステーションを調べます。
- コマンド行管理プログラムを使用している場合は、`snaadmin query_ls` コマンドを発行して、リンク・ステーションの構成を表示します。

MPC リンク・ステーションを始動できない

MPC リンク・ステーションを始動できない場合は、[12 ページの『リンク・ステーションを始動できない』](#)に記載されている確認項目を実行してください。

Enterprise Extender (HPR/IP) のリンク・ステーションを始動できない

Enterprise Extender (HPR/IP) のリンク・ステーションを始動できない場合は、[12 ページの『リンク・ステーションを始動できない』](#)に記載されている確認項目を実行してから、次の確認項目を実行してください。

次のリンク・ステーション・パラメーターを確認します。

1. `remote_hostname` パラメーターに対応する IP アドレスが有効かどうかを確認します。ホスト名は、その定義時点、または CS Linux ソフトウェアを停止してから再始動する時点で、IP アドレスが解決されます。リモート・コンピューターが DHCP を使用している場合は、IP アドレスが変更された可能性があります。その場合は、CS Linux を停止してから再始動するか、リンク・ステーションを削除して再定義し、新規の IP アドレスを取得します。
2. マシンに複数のネットワーク・インターフェース・カードがある場合は、リンク・ステーションが正しいポートに構成されているか確認します。LS が関連付けられているポートに対応するローカルの IP アドレスから、リモート・ホストの IP アドレスに到達可能でなければなりません。
3. 次の手順に従って、ローカルおよびリモートの SAP 構成を確認します。
 - a. ローカルの SAP (ポートの `lsap_address` パラメーターで構成されている) がリモート・システムのリモート SAP に一致していることを確認します。
 - b. リモート SAP (リンク・ステーションのパラメーター `lsap_address` で構成されている) がリモート・システムのローカル SAP と一致していることを確認します。
4. リモート・ホストが Enterprise Extender (HPR/IP) の機能をサポートしていることを確認します。

CS Linux のリンク・ステーションの構成を次のいずれかの方法で確認します。

- Motif 管理プログラムを使用している場合は、リンク・ステーションを調べます。
- コマンド行管理プログラムを使用している場合は、`snaadmin query_ls` コマンドを発行して、リンク・ステーションの構成を表示します。

注: Motif 管理プログラムでは、ローカルおよびリモートの SAP アドレスを表示することはできません。これらのパラメーターを確認するには、コマンド行管理プログラムを使用してください。

AIX または Linux リモート API クライアントが自分のサーバーを認識できるかどうかの確認

最初に、「*IBM Communications Server for Data Center Deployment on Linux* 管理ガイド」のクライアントの管理に関する章を参照してください。

クライアントは、操作の前に、サーバーに接続する必要があります。クライアントから `snaadmin query_node_all` コマンドを発行して、クライアントが 1 つ以上のサーバーを認識できるか確認します。(このコマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。) さらに、いずれかのクライアントが現在接続状態にある場合は、サーバーの名前を記録する `server.current` ファイルも確認します。Linux のリモート API クライアントの場合、このファイルは `/var/opt/ibm/sna` に保管され、AIX のリモート API クライアントの場合、このファイルは `/var/sna` に保管されます。

サーバーから、`snaadmin query_remote_api_clients` コマンドを発行して、クライアントがサーバーに接続されたクライアントとしてリストされていることを確認します。

次の節で `snaadmin query_node_all` コマンドに対する種々の応答および取るべきアクションについて説明します。

アクティブ・ノードのリストが表示される

`snaadmin query_node_all` によりすべてのアクティブ・ノード (つまり、サーバー) が報告される場合、`server.current` ファイルにその名前が記録されており、クライアントはサーバーを認識して、これが問題の原因ではないことを確認できます。

アクティブ・ノードのリストに1つまたは複数のノードがない場合、これらの各ノードがアクティブであることを確認します。詳しくは、[10 ページの『ローカル SNA ノードが活動状態であることの確認』](#)を参照してください。

各ノードはアクティブだが表示されるアクティブ・ノードのリストがまだ不完全な場合、CS Linux がクライアント/サーバー・コミュニケーションで使用している基盤の TCP/IP ネットワークに問題があるかもしれません。

サーバーがアクティブでないときでもクライアントが操作し続けられるようにするには、`snaadmin add_backup` コマンドを発行して、バックアップ・サーバーを構成するか、またはクライアントが使用できるサーバーをさらに追加します。このコマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

アクティブ・ノードが表示されない

`snaadmin query_node_all` がノードを報告しない場合、または `server.current` ファイルが空の場合 (クライアントがサーバーに接続されていないことを示す)、次のことを確認してください。

1. 次のことを行い、クライアントがインストールされたとき、ドメイン名およびサーバーの名前の両方が正しく指定されたことを確認します。
 - サーバーで `snaadmin query_sna_net` コマンドを発行します。
 - クライアントで `sna_clnt.net` ファイルをチェックします。

両方のロケーションで名前がマッチしない場合、クライアントの `sna_clnt.net` ファイルの名前を変更して一致するようにします。

2. クライアントからサーバーにブロードキャスト・メッセージをルーティングするようにセットアップされていない TCP/IP ネットワーク上で、ブロードキャストを使用してサーバーを見つけるようにクライアントが構成されているかを確認します。そのように構成されている場合、`sna_clnt.net` ファイルでサーバー名を明示的に指定します。
3. サーバー上で `sna start` の次に `snaadmin status_node` を発行してアクティブ・サーバーがあるかを確認します。

これらの確認をすべて行い、問題を修正しても、`snaadmin query_node_all` がそれでもノードを報告しない場合、CS Linux がクライアント/サーバー・コミュニケーションで使用している基盤の TCP/IP ネットワークに問題があるかもしれません。この場合は、お客様のシステム管理者に相談してください。

Windows におけるリモート API クライアントが自分のサーバーを認識できるかどうかの確認

Windows クライアントは 操作の前にサーバーのサービスが必要です。

Windows クライアントが開始していることを確認します。この確認および Windows クライアントの始動方法について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 入門」を参照してください。

Windows クライアントが始動したが、まだ作動していない場合、次のことを確認してください。

1. `snaadmin query_sna_net` コマンドを発行して、サーバー上にクライアント・サポートがインストールされて構成されていることを確認します。
2. Windows クライアントが TCP/IP ネットワークに接続され、サーバーを認識できることを確認します (`ping` コマンドを使用)。
3. Windows クライアントがインストールされ、正しく構成されていることを確認します (Windows クライアントのインストール方法については、「*IBM Communications Server for Data Center Deployment on Linux 入門*」を参照してください。Windows クライアントの構成方法については、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください)。特に、クライアントが Windows Vista で稼働している場合は、適切な TCP ポートでのトラフィックが許可されるようにファイアウォールが再構成されていることを確認してください。詳しくは、インストール CD に収められている README ファイルの Windows クライアント情報を参照してください。
4. Windows クライアントの TP 情報が正しく構成されていることを確認します (Windows クライアントの TP 情報の構成については、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください)。
5. CS Linux が Linux サーバー上で正常に開始したことを確認します。
6. マウスをツールバー上の CS Linux アイコンに移動し、Windows クライアントの状態を (クリックしないで) 確認します。次のツールチップ (クライアント・ステータスを示す) が表示されます。
 - `Not started` は、クライアントを始動する必要があることを示します。
 - `Not connected` は、クライアントが実行中だが、ネットワークに接続できなかったことを示します。クライアントがなぜネットワークに接続できなかったかを調査する必要があります。
 - `idle` はクライアントが現在サーバーにアクティブ・セッションをもっていないことを示します。セッションを開始しようとする、クライアントは自動的に再接続し、オペレーター介入は不要です。
 - `servername` はクライアントがアクティブであり、`servername` というサーバーがあることを示します。この後に、この Windows クライアントを使用しているアプリケーション (APPC、LUA、PCOMM など) の現在の数が括弧内に表示されます。サポートできるアプリケーションの最大数は、パラメーター `maximum_process_count` で示されます。これは「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」で説明されています。 `sxclsvce.exe` およびモニター・プログラム `sxclappl.exe` 自体がアプリケーションとしてカウントされるので、実行されているアプリケーションがない場合の最小値は 2 です。

システム構成情報の確認

コードに問題があると想定する前に、必ずシステム構成を確認してください。SNA 構成情報は、サブディレクトリー `/etc/opt/ibm/sna` に保管されています。

構成情報は、次のテキスト・ファイルに保管されています。

- `sna_node.cfg` ファイル。このファイルには、ノード構成情報が保管されています。
- `sna_domn.cfg` ファイル。このファイルには、ドメイン構成情報が保管されています。

上記のファイルに保管されている構成情報は、次のいずれかのプログラムで変更できます。

- CS Linux Motif 管理プログラム
- CS Linux コマンド行管理プログラム
- クライアント/サーバー構成はサーバー上の `sna.net` ファイルに保持されており、Motif 管理プログラムまたはコマンド行管理プログラムを使用して表示および変更ができます。
- Linux クライアント構成情報は `sna_c1nt.net` ファイルに保持されています。
- Windows クライアント構成情報は Windows レジストリーに保持されています。

システム構成について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください。

特定の問題の解決

9 ページの『基本的な確認事項』に記載されている基本チェックをすべて実行しても問題が解決できない場合のために、この項では特定の問題の解決方法について説明します。

発生している問題に最も関連性のある項を参照して、調査を続行してください。

Motif を使用するプログラムに関する問題の解決

Motif CS Linux プログラム (Motif 管理プログラムなど) を開始できない場合には、次の事項を確認してください。

1. Motif ソフトウェアがシステムにインストールされていることを確認します。必要な Motif のバージョンについての情報、およびそのインストール方法については、CS Linux インストール・イメージにある README ファイルを参照してください。
2. 使用している端末が X サーバーをサポートしていることを確認します。
3. Motif 管理プログラムが稼働している Linux システムの X ソフトウェアで、X サーバーのアドレスが構成されていることを確認します。この設定方法は次のいずれかです。

- DISPLAY 環境変数を設定します。たとえば、Korn シェルを使用して X サーバーに接続しており、この X サーバーの TCP/IP 名が「my_PC」の場合には、次のように入力します。

```
export DISPLAY=my_PC:0
```

- Motif プログラムを開始する場合は -d オプションを指定します。例えば、Motif 管理プログラムを開始して X サーバーに接続しており、この X サーバーの TCP/IP 名が「my_PC」の場合には、次のように入力します。

```
xsnaadmin -d my_PC:0
```

4. X サーバーによっては、デフォルトでは要求を受け入れない場合があります (Linux が稼働するサーバーなど)。Motif プログラムから接続不可能であると報告される場合は、要求を受け入れるよう構成する必要があります。たとえば、Linux が稼働している X サーバーの場合には次のように入力します。

```
xhost +
```

APPC または CPI-C アプリケーションの問題の解決

Java™ CPI-C アプリケーションを含め、APPC または CPI-C アプリケーションに問題がある場合は、次の手順に従います。

1. アプリケーションが正しくインストールされていることを確認します。
2. Java CPI-C アプリケーションの場合は、以下のように、環境変数が正しく設定されているか確認します。

Java CPI-C アプリケーションを コンパイル、リンクする前に、Java クラスが保管されるディレクトリーを指定する必要があります。これを行うには、環境変数 CLASSPATH を、/opt/ibm/sna/java/cpic.jar にセットし、エクスポートする必要があります。

Java CPI-C アプリケーションを実行する前に、アプリケーションが実行時にそれらを見つけられるようにライブラリーがストアされているディレクトリーを指定する必要があります。LiS ストリームで Java CPI-C が正常に動作することを確認するため、追加の環境変数をセットする必要があります。

これを行うには、環境変数を次のようにセットし、エクスポートします。

32 ビット・アプリケーションの場合:

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

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

64 ビット・アプリケーションの場合:

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

```
export LD_PRELOAD=/usr/lib64/libpLiS.so
export PATH=/opt/ibm/java2-ppc64-50/jre/bin:/opt/ibm/java2-ppc64-50/
bin:$PATH
```

「*IBM Communications Server for Data Center Deployment on AIX or Linux CPI-C プログラマーズ・ガイド*」に説明されているように、アプリケーションのローカル TP 名を指定するために APPCTPN を設定し、エクスポートすることが必要になる場合もあります。

- 必要なモード、LU、およびリンクの構成が正しいことを確認します。
- アプリケーションにより会話が発信されるときに問題が発生する場合は、次の手順に従います。
 - リンクが活動状態であることを確認します (リンクはオンデマンドでアクティブにされることもあります)。10 ページの『[その他の SNA ノードとの通信の確認](#)』を参照してください。
 - 関係のあるセッションが開始されていることを確認します (これは必要に応じて行われます)。
- アプリケーションが着呼会話要求を受信するときに問題が発生する場合は、次の手順に従います。
 - リンクが活動状態であることを確認します (リンクが着呼として構成されている場合、リンクはオンデマンドでアクティブにされることがあります)。
 - 関係のあるセッションが開始されていることを確認します (これは必要に応じて行われます)。
 - 受信側アプリケーションが動的にロードされている場合 (始動可能な場合) は、TP 構成情報が正しく構成されていることも確認します (TP 情報の構成について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください)。
- 必要なアプリケーションの数に対し、セッション限度が十分であることを確認します。
- LU-LU の組み合わせに対して単一セッション・モードと並列セッション・モードを混在していないことを確認します。
- APPN アーキテクチャーでは、XID 交換なしで (つまり SNRM と UA のみで) 確立されたリンク・ステーション上の (通常は SDLC) 独立 LU 6.2 (並列セッション) はサポートされません。したがって、ホストから XID が送信されないかぎり、独立 LU 6.2 は使用できません。
- 割り当てタイプが正しいことを確認します (たとえば、即時割り当てではコンテンション勝者セッションが必要です)。
- CPI-C アプリケーションの場合には、(snaadmin define_cplic_side_info コマンドまたは APPCLLU 環境変数を使用して) ローカル LU 名および TP 名 (APPCTPN) を設定するか、あるいはデフォルトの LU プールおよび TP 名を使用します。

LUA 問題の解決

LUA の場合は、以下を実行し、アプリケーションが作動しない原因を判別してください。

- アプリケーションが正しくインストールされており、ユーザーがこのアプリケーションの実行権限をもっていることを確認します。
- 次のいずれかの方法を使用して、アプリケーションが実行中であることを確認します。
 - Motif 管理プログラムを使用している場合は、アプリケーションの ID またはユーザー名が LU の隣に表示されていることを確認してください。
 - コマンド行管理プログラムを使用している場合は、snaadmin query_lu_0_to_3 を入力して、appl_conn_active パラメーターが YES に設定されていることを確認します。
 - Linux ps ユーティリティも使用できます。ps ユーティリティの使い方については、[23 ページの『サポート担当員に提出するその他の情報』](#)を参照してください。
- アプリケーションが正しい LU を使用していることを確認します。
- ホスト・リンクが活動状態であることを確認します。
- ホストによって、関係のある LU が活動化されていることを確認します。

MS アプリケーションの問題の解決

Management Services (MS) アプリケーションが作動しない原因を判別するには `snaadmin query_nmvt_application` コマンドを入力します。このコマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

このコマンドにより、次の情報が戻されます。

- 登録済みアプリケーションの名前
- アプリケーションにより受け入れられた MS ベクトル・キー

`REGISTER_NMVT_APPLICATION verb` を使用して、正しい MS ベクトル・キーを設定してください。詳しくは、「*IBM Communications Server for Data Center Deployment on AIX or Linux MS プログラマーズ・ガイド*」を参照してください。

NOF アプリケーションの問題の解決

ノード・オペレーター機能 (NOF) アプリケーションが作動しない原因を判別するには、NOF インターフェースでの API トレースを取得します。API トレースの取得方法の詳細については、[37 ページの『API トレース』](#)を参照してください。

構成を変更する `verb` の発行前に、アプリケーション・プログラムから `SET_PROCESSING_MODE verb` が発行されていることを確認します。この `Verb` について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux NOF プログラマーズ・ガイド*」を参照してください。

TN Server での問題の解決

ここでは、TN Server for 3270 で問題が発生した場合にどのようにすればよいのかについて説明します。

ホストに接続できない

TN Server for 3270 を使用していてホストに接続できない場合は、次のことを確認してください。

1. ノードが開始していること、およびホストとのリンクが「活動」状態、または「オンデマンド」状態にあることを確認します。
2. TN3270 アクセス・レコード内で構成されている LU が SSCP 状態にあることを確認します。
 - LU が活動状態になっていない場合は、リンクを停止してから再始動すると、その LU を活動状態にできます。リンクを停止する前にこのリンクを使用しているユーザーが他にいないことを確認してください。
 - LU がすでに活動状態になっている場合は、別のユーザーがその LU を使用中であるので、現時点ではこの TN3270 クライアントはその LU を使用することはできません。
 - 構成によって個々の LU ではなく、LU プールが指定されている場合は、プール内に 1 つ以上の LU が SSCP 状態にあるか確認します。すべての LU が活動状態である場合は、LU はすべて他のユーザーが使用中であるということなので、現時点ではこの TN3270 クライアントは LU を使用することはできません。

クライアントに接続できない

1. 構成に、TN3270 ユーザー用の適切な TN3270 アクセス・レコードが含まれていることを確認します。アクセス・レコードが、TN3270 クライアントのアドレスを正しく指定していること、またはデフォルト・レコードであることを確認します。また、アクセス・レコードで指定されている LU が、ホストとのリンクで定義されている有効な LU または LU プールであることを確認します。

TN3270 ユーザーおよび LU の構成について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理ガイド」を参照してください。

2. TN Server のセキュア・ソケット層 (SSL) 機能を使用する場合は、次の点について SSL の構成を確認します。

- TN3270 エミュレーターと TN3270 のアクセス・レコードは、いずれも SSL を使用するように構成されていなければなりません。あるいは、この TN3270 クライアントに SSL を使用しないのであれば、どちらも SSL を使用しないように構成されていなければなりません。
 - SSL を使用するように TN3270 アクセス・レコードを構成することができない場合 (SSL オプションが、Motif 管理プログラムで使用不可にされているか、あるいは SMIT または管理コマンドが失敗し、戻りコード FUNCTION_NOT_SUPPORTED が戻される場合)、SSL をサポートしているソフトウェアがサーバーにインストールされていません。このソフトウェアのインストールについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」を参照してください。
 - SSL ソフトウェアが正しくインストールされており、TN3270 エミュレーターと TN3270 アクセス・レコードが SSL ソフトウェアを使用するように構成されている場合は、SSL に対するセキュリティー要件が有効でない可能性があります。TN Server は、TN3270 エミュレーターが認める認証局からの最新の証明書を持っている、というのがセキュリティー要件の 1 つです。これにより、別のプログラムが TN3270 接続要求を代行受信し、有効な TN Server になりすますことができなくなります (その理由は、このプログラムは正式な証明書を持っていないと、自身を正しくエミュレーターに識別させることができないからです)。これを訂正するには、キー管理プログラム snakeyman を開始して、オンライン・ヘルプの指示に従います。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」を参照してください。
3. アクセス・レコードで定義されている TN3270 サーバー・ポートへのクライアント・アクセスをブロックするファイアウォールが使用可能になっていないことを確認します。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」のファイアウォールの構成方法を参照してください。

TN リダイレクターでの問題の解決

ここでは、TN リダイレクターで問題が発生した場合にどのようにすればよいのかについて説明します。

ホストに接続できない

TN リダイレクターを使用していてホストに接続できない場合は、以下を確認してください。

1. ノードが開始していることを確認します。
2. 構成に、TN リダイレクターのユーザー用に適切な TN リダイレクターのアクセス・レコードが含まれていることを確認します。アクセス・レコードが、TN クライアントのアドレスを正しく指定しているか、またはデフォルトのレコードであること、および指定された TCP/IP ポートがクライアントが使用しているポートであることを確認します。

TN リダイレクターの構成について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください。
3. アクセス・レコードがホストの IP アドレスを正しく指定し、指定された TCP/IP ポートがホストが使用しているポートであることを確認します。
4. ping ユーティリティーを使用して、クライアントの TCP/IP アドレスとの接続を確認します。ホストに対してもこれを繰り返します。
5. TN Server のセキュア・ソケット層 (SSL) 機能を使用する場合は、次の点について SSL の構成を確認します。
 - TN クライアントと TN リダイレクターのアクセス・レコードは、いずれも SSL を使用するように構成されていなければなりません。あるいは、このクライアントに SSL を使用しないのであれば、どちらも SSL を使用しないように構成されていなければなりません。
 - SSL を使用するように TN リダイレクターのアクセス・レコードを構成することができない場合 (SSL オプションが Motif 管理プログラムで使用不可にされているか、あるいは SMIT または管理コマンドが失敗し、戻りコード FUNCTION_NOT_SUPPORTED が戻される場合)、SSL をサポートしているソフトウェアがサーバーにインストールされていません。このソフトウェアのインストールについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」を参照してください。
 - SSL ソフトウェアが正しくインストールされており、エミュレーターとアクセス・レコードが SSL ソフトウェアを使用するように構成されている場合は、SSL に対するセキュリティー要件が有効でない可能性があります。TN リダイレクターは、TN クライアントが認める認証局からの最新の証明書を持

持っている、というのがセキュリティ要件の1つです。これにより、別のプログラムが接続要求を代行受信し、有効な TN リダイレクターになりすまることができなくなります (その理由は、このプログラムは正式な証明書を持っていないと、自身を正しくエミュレーターに識別させることができないからです)。これを訂正するには、キー管理プログラム `snakeyman` を開始して、オンライン・ヘルプの指示に従います。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」を参照してください。

6. TN リダイレクターへのクライアント・アクセスをブロックするファイアウォール、またはホストへのサーバー接続をブロックするファイアウォールが使用可能になっていないことを確認します。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」のファイアウォールの構成方法を参照してください。

ネットワーク・ノード・セッションのルーティング問題の解決

ネットワーク・ノードによるセッションのルーティングが行われない原因を判別するには、次のようにします。

1. `snaadmin query_isr_sessions` を入力して、活動セッションの最新情報を取得します。このコマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理コマンド解説書*」を参照してください。
2. ネットワーク上のすべてのマシンのネットワーク ID が同一であることを確認します。CS Linux には、異なるネットワーク ID をもつマシンにアクセスするために必要なボーダー・ノード・サポートはありません。
3. ISR (中間セッション・ルーティング) を使用して独立 APPC セッションだけをルーティングしようとしていることを確認します。他のセッションでは、SNA ゲートウェイや DLUR を使用できます。

SNA ゲートウェイ・セッションの問題の解決

SNA ゲートウェイ・セッションが接続しない原因を判別するには、次の事項を確認してください。

1. アップストリームおよびダウンストリームのリンク・ステーションが活動状態であることを確認します。
 - Motif 管理プログラムを使用している場合は、リンク・ステーションを調べます。
 - コマンド行管理プログラムを使用している場合は、`snaadmin status_connectivity` コマンドを入力します。
2. 構成システムに LU が接続されており、他のアプリケーションでこれらの LU が使用されていないことを確認します。
 - Motif 管理プログラムを使用している場合は、リンク・ステーションと LU プールを調べます。
 - コマンド行管理プログラムを使用している場合は、`snaadmin query_downstream_lu` コマンドを入力します。
3. プールを使用している場合は、次のいずれかの手順でプール内に使用可能な LU があるかどうかを確認します。
 - Motif 管理プログラムを使用している場合は、リンク・ステーションと LU プールを調べます。
 - コマンド行管理プログラムを使用している場合は、`snaadmin query_lu_0_to_3` コマンドを入力します。

サーバー管理問題の解決

サーバーが互いに管理できない理由を判別するには、次のことを行います。

1. 1つのサーバーがドメイン内でコントローラー・サーバーであることを確認します。
2. 接続形態の構成の情報を入手するために、各サーバー上で `snaadmin query_sna_net` コマンドを入力します。
3. サーバーが同一のドメイン内にあることを確認します。
4. Linux の `ping` コマンドを使用して TCP/IP 接続をチェックします。

オンライン・サポート情報

前の節で説明したステップを実行しても問題が解決できない場合、IBM の Web サイトで新たな情報を見つけることができる場合があります。CS Linux のサポート・ページでは、次の項目に関する情報を提供しています。

- コードの修正
- 有益なヒントと手法
- CS Linux についての議論のためのニュースグループ
- サポート・オプション

<https://www.ibm.com/mysupport/s/topic/0TO0z0000006v30GAQ/communication-server> でこれらの情報にアクセスできます。

サポート担当者への問題の報告

システムの問題には、解決できないものもあります。解決できないシステムの問題の場合には、「メッセージに対する処置 (Message action)」フィールドにサポート担当員に連絡するよう示されます。

サポート担当員のタイプ

問題解決を支援するサポート担当員は次のとおりです。

リモート・システムまたはネットワーク担当のサポート担当員

SNA ネットワークおよび CS Linux と通信しているリモート・システムのサポート担当員。たとえば、X.25 ネットワークのプロバイダー (X.25 関連の問題)、TCP/IP ネットワークの担当員 (TN Server 関連の問題)、ホストの担当員 (LUA 関連の問題)、さらにリモート・システムのシステム管理者 (APPC または CPI-C 関連の問題) などがこれに該当します。

サポート・サービス

IBM サポート担当員

担当者に提供する情報

お客様の問題について最初に提供する情報の詳細は、高速解像度を受け取る可能性が高いことをサポート・チームに提供します。収集する情報のタイプについては、以下のセクションを参照してください。

報告する問題の性質と範囲によっては、サポート担当員が実行依頼を依頼することがあります。スナゲト pd。コマンド行診断コレクション・ユーティリティー。このユーティリティーは、圧縮 tar フォーマットでファイルを自動的に作成します。このファイルは、問題の診断に使用できる包括的なデータを提供します。ここで説明するすべての情報が含まれます。

スナゲト pd、参照 63 ページの『付録 C snagetpd の使用法』の詳細については、

Readme ファイル

発生した問題についてサポート担当員に処理を依頼する際には、readme ファイルに次の情報を記述して提出してください。

- 発生した問題についての簡潔な説明。機能しなくなったものは何か、あるいは正しく機能しないものは何か、どのような動作を想定していたのか。
- 問題が発生する前に行なった操作
- 問題が発生した日付と時刻
- 問題が再発する頻度 (再現できる場合)
- 問題の機能はこれまでに正常に作動していたかどうか。正常に作動していた場合、最後に作動した後、どのような変更があったか。
- 問題に関連している SNA ログ・ファイルに書き込まれたメッセージのメッセージ番号とパラメーター。

readme ファイルに記述する情報のほかにサポート担当員に提出する情報については、[23 ページの『サポート担当員に提出するその他の情報』](#)を参照してください。

サポート担当員に提出するその他の情報

readme ファイルのほかに、次の情報を収集してサポート担当員に提出してください。

プログラム・エラー・メッセージ

プログラム・エラー・メッセージの内容を検討しても問題を解決できない場合には、次の手順に従ってください。

1. 画面に表示されたメッセージを書きとめます。
2. ログ・ファイルを保管します ([26 ページの『ログ・ファイルの名前と位置の変更』](#)を参照してください)。

エラー・ログとトレース・ファイルの情報

エラーが発生した場合は、エラー・ログ・ファイルおよび監査ログ・ファイルとして使用していたファイルを提出してください (これらのファイルは通常 /var/opt/ibm/sna/sna.err と /var/opt/ibm/sna/sna.aud です)。トレースを使用可能にして稼働していた場合には、トレース・ファイルも提出してください。

監査ログまたは例外ログを使用不可にして CS Linux を稼働していた場合は、すべてのカテゴリのログ機能を使用可能にして問題を再現してみてください。問題を再現できた場合には、新しいログ・ファイル (すべてのメッセージ・カテゴリが含まれている) を提出してください。

システム構成情報

担当のサポート・チームから構成情報の提出を要求された場合は、次のファイルをサポート・チームに提出してください。

- sna_node.cfg ファイル。ノードの構成情報が保管されています。
- sna_domn.cfg ファイル。ドメインの構成情報が保管されています。
- sna_tps ファイル。TP の構成情報が保管されています。
- environment ファイル。CS Linux に渡される環境変数が保管されています。
- logfilter.txt ファイル。フィルター操作されたログが保管されています。

ソフトウェアのバージョン情報

問題をローカルで解決できない場合には、サポート・チームは使用中のマシンで実行されているコードのレベルを明確に把握する必要があります。適切な Linux ユーティリティを使用して、ソフトウェア・パッケージの全部のバージョンを表示します。すべての CS Linux コードに、詳細なコード・レベルを識別する「タグ」が含まれています。この情報を抽出するには、snawhat ユーティリティを使用します。サード・パーティーのアプリケーションが静的ライブラリーとリンクしている場合、使用しているライブラリーのバージョンを判別するには snawhat を使用します。

ファイルのバージョン情報を取得するには、コマンド行に次のように入力します。

```
snawhat
```

ここではバージョン情報を取得する必要があるファイル (1つまたは複数) を表します。

たとえば、次のファイルのバージョン情報を取得する手順は次のとおりです。

- my_app1 というローカル・ディレクトリー・ファイルが使用している静的ライブラリーのバージョン情報を取得するには、コマンド行に次のように入力します。

```
snawhat my_app1
```

- ローカル・ディレクトリーにインストールされている sna* 実行可能ファイルのバージョン情報を取得するには、コマンド行に次のように入力します。

```
snawhat sna*
```

システム・リソース

どのプログラムが実行しているかを確認するだけでなく、Linux 処理環境も確認できます。サポート・チームから Linux プロセスとリソースの状況についての情報を得るために `ps` 標準 Linux ユーティリティーを実行するように求められることがあります。

プロセス状況 (ps) ユーティリティー

`ps` ユーティリティーは、Linux コンピューターにインストールされている標準ツールの 1 つです。このユーティリティーにより、マシン上の Linux プロセスの状況に関する情報を取得できます。次のような場合に `ps` を実行します。

- プログラムが開始しない。
- プログラムが「停止する」、クラッシュする、または実行速度が遅い。
- プログラム・エラー・メッセージにより、実行中のプログラムが他のプロセスに割り込んでいることが示される。

実行中の CS Linux プロセスに関する基本情報を取得するには、コマンド行に次のように入力します。

```
ps -ef | fgrep sna
```

`ps` の詳細については、使用している Linux の文書を参照してください。

サポート担当員のための情報収集のまとめ

ここでは、再現可能な問題を見つけた場合に、サポート担当員に提供する情報を収集するために取らなければならないステップを要約しています。

可能であれば、既存の診断ファイルをすべて削除してから診断情報の収集を開始することをお勧めします。このために CS Linux を停止することが必要になるため、一部のケースでは実際的でないことがあります。このような場合は、下の手順の最初から 3 ステップは省略します。

1. `sna stop` コマンドを発行して、CS Linux を停止します。
2. `/var/opt/ibm/sna` の内容を削除します。その内容を保管しておきたい場合はそれを別のディレクトリに移動します。
3. `sna start` コマンドを発行して、CS Linux を再始動します。
4. 次のように、トレース・ファイルのサイズを最大値に設定して、関連するとレース情報をすべて取り込めるようにします。

```
snaadmin set_trace_file, trace_file_size = 10000000
```

5. 次のように、監査ログ記録と例外ログ記録をオンにします。

```
snaadmin set_global_log_type, audit = YES, exception = YES, succinct_audits = NO, succinct_errors = NO
```

6. 次のように、すべてのトレースをオンにします。

```
snaadmin add_dlc_trace
```

```
snaadmin set_trace_type, trace_flags = ALL
```

7. 問題を再現するための一連のアクションを実行します。

8. 次のように、トレースをオフにします。

```
snaadmin remove_dlc_trace
```

```
snaadmin set_trace_type, trace_flags = NONE
```

9. 次のように `snagetpd` を実行して、ログ情報とトレース情報を収集します。

```
snagetpd
```

10. サポート担当員が推奨するメカニズムをすべて使用して、`snagetpd` の出力をサポート担当員に提出します。

第 3 章 ログ記録とトレースの使用法

ログ記録とトレースは、CS Linux の問題を解決する上で有用な情報をユーザーと支援チームに提供する優れた診断ツールです。この章では、Motif 管理プログラムまたはコマンド行管理プログラムを使用して、ログ記録とトレースを実行する方法について説明します (コマンドの詳細な解説については、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください)。また、ログ・ファイルに書き込まれる情報をフィルター処理して、同じログ・メッセージの複数のインスタンスを記録しないようにする方法についても説明します。

また、ログ記録メッセージとトレース出力の例も収録しています。

Motif 管理プログラムによるログ記録の制御

CS Linux のイベントのロギングを制御する最も簡単な方法は、Motif 管理プログラム (xsnaadmin) を使用することです。このプログラムのグラフィカル・ユーザー・インターフェースを使用して、システムでのロギングのタイプやログに記録するメッセージのタイプの選択などの診断タスクを実行できます。

Motif 管理プログラムを開始する手順は次のとおりです。

1. コマンド行に xsnaadmin と入力して、Enter キーを押します。メイン・ウィンドウが表示されます。
2. 「診断 (Diagnostics)」メニューを選択します。

注: ノード・ウィンドウから「診断 (Diagnostics)」メニューを選択することもできます。

3. 「ロギング (Logging)」を選択します。

「ロギング (Logging)」ダイアログが表示されます。このダイアログでは、次のログ記録アクティビティーを制御できます。

中央ロギング (Central logging) または ローカル・ロギング (Local logging)

すべてのサーバーおよびクライアントの中央エラー・ログが必要な場合、「中央ロギング (Central logging)」を選択します。ローカル側で各マシンにエラー・ログを作成する必要がある場合、「ローカル・ロギング (Local logging)」を選択します。中央ロギングはコントローラーにより実行されるので、Motif 管理プログラムがコントローラー・サーバーと連絡を取り合っている場合にのみ中央ロギングまたはローカル・ロギングを選択できます。

例外のログへの記録 (Log exceptions)

すべての例外イベントをログに記録するようシステムに指示する場合には、このオプションを選択します。このオプションを選択すると、ロギング・モードとして「詳細 (verbose)」または「簡略 (succinct)」のいずれかを選択するようプロンプトが表示されます。選択し終わったら、「了解 (OK)」をクリックします。

この選択によりドメイン内のすべてのマシンが影響を受けます (コマンド行管理プログラムを使用して構成したローカル・オーバーライドを持っていない場合)。

監査メッセージのログへの記録 (Log audit messages)

すべての監査イベントをログに記録するようシステムに指示する場合には、このオプションを選択します。このオプションを選択すると、ロギング・モードとして「詳細 (verbose)」または「簡略 (succinct)」のいずれかを選択するようプロンプトが表示されます。選択し終わったら、「了解 (OK)」をクリックします。

この選択によりドメイン内のすべてのマシンが影響を受けます (コマンド行管理プログラムを使用して構成したローカル・オーバーライドを持っていない場合)。

CS Linux Motif 管理プログラムのみがグローバル・ログ設定値をセットします。ローカル・ログ設定値はグローバル設定値をオーバーライドし、コマンド行管理プログラムを使用して特定のマシンで構成できます。

コマンド行管理プログラムによるログ記録の制御

コマンド行管理プログラムでは次の操作を行うことができます。

- ログ・ファイルとバックアップ・ログ・ファイルの名前と位置を変更します。
- 中央ロギングまたはローカル・ロギングを使用可能にします。
- 個々のサーバーの監査イベントおよび例外イベントのログ記録を使用可能または使用不可にします。
- ログ・ファイルの最大サイズを変更します。
- 詳細ロギングまたは簡略ロギングを使用可能にします。

ログ・ファイルの名前と位置の変更

CS Linux では、通常、ログは次の 2 つのファイルに記録されます。

- 問題 (常にログ記録されます) と例外 (ログ記録した場合) は、通常、`/var/opt/ibm/sna/sna.err` ファイルに記録されます。
- 監査 (ログ記録した場合) は、通常、`/var/opt/ibm/sna/sna.aud` ファイルに記録されます。

コマンド行管理プログラムを使用してこれらのファイルの名前と位置を変更する手順は次のとおりです。

1. `snaadmin set_log_file` コマンドを使用して、ログ・ファイルの名前を変更します。
2. `snaadmin query_log_file` コマンドを使用して、現在のファイル名を確認します。

これらのコマンドの発行について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

すべてのメッセージ (エラーおよび監査) を同じファイルに記録し、それらが相互にどのように関連しているのかわかりやすくする方が簡単です。同じファイルに記録するには、`snaadmin set_log_file` コマンドを 2 回発行します。1 つは `log_file_type` パラメーターを `ERROR` に設定し、もう 1 つは同じパラメーターを `AUDIT` に設定します。ただし両方とも同じファイル名を指定してください。

たとえば、エラー・ログ・メッセージと監査ログ・メッセージの両方を `sna.log` という名前のファイルに記録する場合は、コマンドを次のように指定します。

```
snaadmin set_log_file, log_file_type = ERROR, file_name = sna.log  
snaadmin set_log_file, log_file_type = AUDIT, file_name = sna.log
```

監査ログ・ファイルとエラー・ログ・ファイルは ASCII テキスト・ファイルです。これらのファイルを表示するには、`vi` などの標準の Linux テキスト・エディターを使用します。

中央ロギングまたはローカル・ロギングの使用可能化

クライアント/サーバー・システムでは、CS Linux は、デフォルトで各サーバーに対するログ・メッセージをそのサーバー上のファイルに送信します (ローカル・ロギング)。ただし、すべてのログ・メッセージをコントローラー・サーバー上のファイルに送信して、中央の 1 つのファイル・セットにすべてのログを保管することもできます (中央ロギング)。

サーバーが始動したとき、そのドメイン構成ファイルを見つけられない場合 (たとえば、コントローラー・サーバーおよびバックアップ・サーバーがアクティブでない)、サーバーは中央ロギングするかローカル・ロギングするか、またはどのサーバーが中央ロガーか判断できません。この場合は、サーバーはメッセージをローカルのログに記録します。後でサーバーがコントローラー・サーバーと接続を確立し、中央ロギングが使用されていると判断すると、サーバーは以後のメッセージを中央ロガーに送信し、ローカル・ロギングを停止します。

`set_central_logging` コマンドを使用して中央ロギングするかローカル・ロギングするかを指定します。現在中央ロガー (ここへすべてのログ・メッセージが送信される) と定義されているサーバー名をチェックするかまたは中央ロギングが現在使用可能であるかを判断するには、`query_central_logger` 管理コマンドまたは `query_central_logging` 管理コマンドを使用してください。

これらの管理コマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

ログに記録するメッセージの指定

問題メッセージは常にログに記録され、このログ記録を使用不可にすることはできません。ただし、例外メッセージと監査メッセージについてはログに記録するかどうかを指定できます。初期デフォルトでは、例外メッセージはログに記録されますが監査メッセージは記録されません。すべてのサーバーの例外メッセージと監査メッセージのログ記録の設定値を指定するには、`snaadmin set_global_log_type` コマンドを使用します。必要な場合、`snaadmin set_log_type` コマンドを使用することにより、特定のサーバーに対するこれらの設定値をオーバーライドできます。

どのロギング・オプションが有効になっているかを判別するには、次のコマンドを発行します。

- グローバル設定を使用するサーバーで記録されるメッセージのカテゴリをチェックするには、`snaadmin query_global_log_type` を使用します。
- 特定のサーバーで記録されているメッセージのカテゴリをチェックするには、`snaadmin query_log_type` を使用します。

これらの管理コマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

ログ・ファイル・サイズの制御

CS Linux では、ログ・ファイルのサイズが大きくなり過ぎてディスク・リソースを過大に消費することのないよう制御できます。`snaadmin set_log_file` 管理コマンドでは、ログ・ファイルの最大サイズと、ログ情報のタイプ別（監査またはエラー）のバックアップ・ファイルの名前を指定できます。デフォルトの最大ファイル・サイズは、10,000,000 バイトです。

ログ・ファイルが指定のサイズに達すると、CS Linux によりログ・ファイルの名前がバックアップ・ファイルの名前に変更され（既存のバックアップ・ファイルは上書きされます）、ログ・ファイルの内容が消去されます。つまり、一度に保管できるログ情報の最大量は、指定した最大ファイル・サイズの 2 倍となります（監査情報とエラー情報をそれぞれ個別のファイルに記録している場合には、最大ファイル・サイズの 4 倍になります）。

大量のログ情報を受け入れるために、ログ・ファイルのサイズを増やす必要がある場合があります（システムの容量が十分な場合）。次のような状況の場合には、ログ・ファイルのサイズの増加について検討してください。

- 多数のお客様またはユーザーが存在する場合（通信リンクで障害が発生すると、サーバー上でセッション障害に関連するログが大量に記録されるため）。
- 監査ログと例外ログを開始する場合。
- ローカル・ロギングのかわりに中央ロギングを使用する場合。
- 簡略ロギングではなく詳細ロギングを実行する場合。詳しくは、[28 ページの『詳細ロギング・メッセージの形式』](#)を参照してください。

`snaadmin set_log_file` 管理コマンドを使用すれば、いつでもログ・ファイルの最新の内容を消去できます（この場合、ログ・ファイルの情報をバックアップ・ファイルにコピーすることもコピーしないこともできます）。

詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

ログの形式の選択

次のログ記録形式のいずれかを選択することで、ログに記録する詳細項目の容量を制御できます。

詳細ロギング

各メッセージには、メッセージ番号、発生元のコンポーネント、メッセージのタイプ、メッセージ・テキスト、メッセージが生成されるに至った状況の原因、推奨処置が含まれています。

簡略ロギング

各メッセージには、省略形式のヘッダー情報(メッセージ番号、発生元のコンポーネント、メッセージのタイプ)とメッセージ・テキストのみが含まれます。snahelp コマンド行ユーティリティを使用すると、特定のメッセージ番号に対する原因と処置の情報を取得することができます(30 ページの『簡略ロギング・メッセージに snahelp を使用する方法』を参照してください)。

デフォルトは簡略ロギングです。監査メッセージ、エラー・メッセージ(問題および例外)、またはその両方のメッセージ・タイプに対して詳細ロギングを指定するには、set_global_log_type コマンドまたは snaadmin set_log_type を使用します。中央ロギングを使用する場合、簡略ロギングまたは詳細ロギングの選択は、中央ロガーとして活動しているサーバーの設定値により決まります。それで、同一タイプのすべてのメッセージ(監査またはエラー)は同一のフォーマットでファイルに書き込まれます。

簡略ロギングを使用している場合でも、API コンポーネントによるメッセージのいくつかは詳細形式でログ・ファイルに書き込まれることがあります。一般にこのような状況が発生するのは、CS Linux が終了中の場合、または特定のエラー状態になっている場合です。このような場合、コンポーネントはノードの構成に関する情報を取得できないため、使用するログ形式を判別できません。

詳細ロギング・メッセージの形式

次に、詳細ロギング形式による標準的なログ・メッセージの例を示します。この例のあとで、メッセージの各項目について説明します。

```
Verbose Logging Message Format
----- 13:55:16 EDT 15 May 1997 -----
CFG_DAEM Message 4097 - 132, Subcode: 1 - 1
Log category: PROBLEM Cause Type: External
System: sna18
Process ID: 17908 (snacfgdae)

The initially active port could not be started when starting the node.
Port name = SDLCPOCause: The config daemon could not start the port while loading the node's
configuration. The node will be started, but the port will not be started.
Action: Check for other logs which indicate why the port failed to start.
Check that the DLC has been started.
```

詳細ログのメッセージに記述される情報は次のとおりです。

タイム・スタンプ (Timestamp)

メッセージが生成された日付と時刻。中央ロギングを使用している場合、各メッセージのタイム・スタンプはメッセージが生成されるコンピューターのシステム・クロックから取られます。異なるコンピューターのシステム・クロックが同期していない場合(クロックの不正確さまたはタイム・ゾーンの違いのため)、中央ログ・ファイルのメッセージは、タイム・スタンプの順序ではなく、到着順に中央ロガーのファイルに追加されるので、順序が狂って表示されます。

ご使用のシステムに Windows クライアントが存在し、中央ロギングを使用している場合、Windows レジストリーの TZ エントリーが正しいタイム・ゾーンを示すように設定されていることを確認してください。これが正しく設定されていない場合、Windows クライアントからのログのタイム・スタンプは不正確です。TZ の設定の詳細については、Windows の資料を参照してください。

コンポーネント (component)

メッセージをログに記録した CS Linux コンポーネント(ローカル・ノード、リンク・ドライバー、APPC ライブラリーなど)

メッセージ番号 (Message number)

メッセージの ID。この ID は 2 つの番号で構成されます。

サブコード (Subcode)

CS Linux 内のメッセージのログ記録先を示す固有の ID。このサブコードを使用するのは、CS Linux サポート担当員のみです。

ログ・カテゴリー (Log category)

ログ・メッセージのイベント・カテゴリー。有効な値は Problem、Exception、Audit です。

原因タイプ (Cause type)

メッセージが発行された原因。可能な値は次のとおりです。

内部 (Internal)

CS Linux コンポーネントの内部エラー。このタイプのエラーは CS Linux サポート担当員に報告してください。

システムしきい値 (System limit)

CS Linux ソフトウェアの内部限界値 (固定サイズのテーブルの項目など)。このタイプのログが記録されることはほとんどありません。

外部 (External)

CS Linux 外部の原因 (通信リンク・ハードウェアの問題など) または CS Linux 以外のソフトウェア内部の原因 (通信リンク・ドライバーなど)

リソース (Resource)

リソースの不足 (たとえば、Linux コンピューターのメモリー不足など)

ユーザー (User)

ユーザー・エラー (たとえば、コマンド行で CS Linux プログラムに対して無効なパラメーターが指定された場合など)

SNA

リモート・システムによる SNA プロトコル違反、または他の SNA システムとのインターオペラビリティの問題

構成 (Config)

CS Linux 構成のエラー、または CS Linux 構成とリモート・システム間の不一致

監査 (Audit)

通常のイベント。通知の目的でのみ報告されます。

システム名 (System name)

メッセージの原因となった状況が検出されたコンピューターの名前。

プロセス ID とプロセス名 (Process ID and name)

Linux プロセス ID (システム名が表示されているコンピューターから) およびメッセージをログに記録したプロセスの実行可能プログラム名。プロセス ID は、ユーザー・スペースのコンポーネントについてのみ表示されます。Windows クライアントによりログ記録されたメッセージの場合、このパラメーターはプロセスの Windows タスク・ハンドルを識別します。

メッセージ・テキスト (Message text)

ログ記録されている状況について説明するテキスト。このフィールドには、メッセージを発生した特定の状況に関連する変数パラメーターが多く出力されることがあります。たとえば、APPC セッションの開始を報告するメッセージには、ローカル LU およびパートナー LU の名前と、このセッションのためにこれらの LU が使用するモードの名前が含まれることがあります。

このフィールドには、オペレーティング・システムの呼び出しからの戻りコードが出力されることがあります。Linux コンピューターでログ記録されたメッセージの場合、戻りコードは、シンボル名または数値で示されます。対応するシンボル名を知りたい場合には、エラーが発生したコンピューターの /usr/include/sys/errno.h ファイルの数値を確認します。シンボル名の一覧は、オペレーティング・システムの資料に記載されています。

Windows クライアントによりログ記録されたメッセージの場合、これらの戻りコードの説明は、Windows の資料を参照してください。

メッセージの原因 (Message cause)

ログに記録された状況の原因に関する追加情報。メッセージ・テキストに必要な情報がすべて含まれている場合には、このフィールドは出力されないことがあります。原因タイプが Internal の場合、通常このフィールドは使用されません。

メッセージに対する処置 (Message action)

メッセージの結果としての推奨処置。監査メッセージでは、エラー状況ではなくアカウントと進行に関する情報が報告されます。したがって、監査メッセージの場合は処置が必要でないため、このフィールドは出力されません。

簡略ロギング・メッセージの形式

以下に、簡略ロギングのメッセージ形式を示します。

```
13:55:16 EDT 15 May 1997 4097-132 (1-1) P sna18 PID 17908 (snacfgdae)
The initially active port could not be started when starting the node.
Port name = SDLCP0
```

1 行目に、メッセージ・テキストを除くすべてのフィールドが出力されます。メッセージ・テキストは 2 行目に出力されます。

タイム・スタンプ (Timestamp)

メッセージがログ記録された日付と時刻

メッセージ番号 (Message number)

メッセージの ID。この ID は 2 つの番号で構成され、番号と番号の間はハイフン (-) で区切られます。

サブコード (Subcode)

CS Linux 内のメッセージのログ記録先を示す固有の ID。メッセージ番号のあとに括弧で囲まれて示されます。このサブコードを使用するのは、CS Linux サポート担当員のみです。

ログ・カテゴリ (Log category)

ログ・メッセージのカテゴリ。1 文字で示されます。

- P (問題)
- E (例外)
- A (監査)

システム名 (System name)

メッセージの原因となった状況が検出されたコンピューターの名前。

プロセス ID とプロセス名 (Process ID and name)

Linux プロセス ID (PID という文字が先頭に付いている) と、メッセージをログに記録したプロセスの実行可能プログラム名。プロセス ID は API コンポーネントについてのみ表示されます。Windows クライアントによりログ記録されたメッセージの場合、このパラメーターはプロセスの Windows タスク・ハンドルを識別します。

メッセージ・テキスト (Message text)

ログ記録されている状況について説明するテキスト。このフィールドには、メッセージを発生した特定の状況に関連する変数パラメーターが多く出力されることがあります。たとえば、APPC セッションの開始を報告するメッセージには、ローカル LU およびパートナー LU の名前と、このセッションのためにこれらの LU が使用するモードの名前が含まれることがあります。

このフィールドには、オペレーティング・システムの呼び出しからの戻りコードが出力されることがあります。Linux コンピューターでログ記録されたメッセージの場合、戻りコードは、シンボル名または数値で示されます。対応するシンボル名を知りたい場合には、エラーが発生したコンピューターの /usr/include/sys/errno.h ファイルの数値を確認します。シンボル名の一覧は、オペレーティング・システムの資料に記載されています。

Windows クライアントによりログ記録されたメッセージの場合、これらの戻りコードの説明は、Windows の資料を参照してください。

簡略ロギング・メッセージに snahelp を使用する方法

簡略ロギング形式では、原因と処置に関する情報は出力されません。ただし、snahelp ユーティリティーを使用すれば、特定のメッセージ番号に対する原因と処置に関する詳細を取得することができます。このユーティリティーを使用するには、Linux のコマンド・プロンプトで以下のように入力します。

```
snahelp message_number
```

message_number は、CS Linux から戻されたメッセージ番号で、メッセージのヘッダー情報に戻されます。メッセージ番号は 2 つの番号で構成され、番号と番号の間はハイフン (-) で区切られます。

このユーティリティーでは、メッセージをログに記録したコンポーネントの名前と、詳細ロギングで出力されるメッセージ番号 (*Message number*)、原因タイプ (*Cause type*)、メッセージの原因 (*Message cause*)、メッセージに対する処置 (*Message action*) の情報が戻されます。

たとえば、30 ページの『[簡略ロギング・メッセージの形式](#)』の簡略ロギング・メッセージ (コンポーネント ID およびメッセージ番号 4097 - 132) の詳細情報を取得するには、次のコマンドを入力します。

```
snahelp 4097-132
```

snahelp の出力を以下に示します。

```
snahelp Output
CFG_DAEM Message: 4097 - 132, Cause Type: External

Cause: The config daemon could not start the port while loading the node's
configuration. The node will be started, but the port will not be started.
Action: Check for other logs which indicate why the port failed to start.
Check that the DLC has been started.
```

ロギングのフィルター処理

特定のイベントが頻繁に発生し、ログ・ファイルに同じログ・メッセージのインスタンスが多くあるのがわかった場合は、フィルターを設定して、特定の 1 つ以上のログ・メッセージを 1 回だけログに記録するように指定できます。同じログ・メッセージの後続のインスタンスは無視され、ログ・ファイルには書き込まれません。このフィルター処理は、監査、例外、および問題ログのすべてのログ・タイプに適用されます。

中央ロギングを使用している場合は、中央ロガーとして動作しているサーバーにフィルターをセットアップします。すなわち、メッセージは、複数のサーバーで発生した場合でも、ファイルには 1 回だけ書き込まれます。フィルターは、バックアップ・サーバーとして動作可能な他のサーバーに複製しておくことをお勧めします。これにより、中央ロガーが停止して、他のサーバーが引き継ぐ場合にフィルター処理を継続することができます。また、このフィルターはリモート API クライアントに設定することができ、指定されたメッセージがこのクライアントから 1 回だけ中央ロガーに送信されます。

中央ロギングを使用していない場合は、フィルターを各サーバーまたはリモート API クライアントに設定することができます。これにより、メッセージがそのシステムで 1 回だけログに記録されます。同じフィルター・オプションをすべてのサーバーとクライアントにセットアップする必要はありません。フィルターは、必要に応じて、各サーバーまたはクライアントに設定したり設定解除のままにしておくことができます。

1 つ以上のログ・メッセージにフィルターをセットアップするには、ASCII ファイル `logfilter.txt` を次のディレクトリーに作成します。

- Linux サーバーまたは Linux リモート API クライアントの場合: `/etc/opt/ibm/sna`
- AIX リモート API クライアントの場合: `/etc/sna`
- Windows リモート API クライアントの場合: Windows レジストリーで `Logging / log_directory` パラメーターに指定されたディレクトリー。このパラメーターが指定されていない場合、デフォルトはリモート API クライアントのインストール・ディレクトリー (`c:\ibmcs\w32cli` など) です。

このファイルの各行には、フィルター処理したい特定のログ・メッセージのメッセージ番号が含まれます。これはハイフン (ダッシュ) 文字で区切った 2 つの番号として指定します (28 ページの『[詳細ロギング・メッセージの形式](#)』を参照)。例えば、以下のようになります。

```
1024-15
2048-12
512-16
```

このファイルには、最大 20 個のメッセージ番号を含めることができます。それ以上のメッセージ番号を含めると、ファイルの最初の 20 行に指定されたメッセージ番号のみがフィルター処理され、ファイル内のその他の行は無視されます。

このファイルは、CS Linux ソフトウェアがこのファイルを収容するサーバーまたはクライアントで始動されたときに読み取られます。

- `logfilter.txt` ファイルを新規に作成した場合、または既存のファイルを変更した場合は、その変更を有効にするために CS Linux ソフトウェアをこのファイルを受容するサーバーまたはクライアントで停止してから再始動する必要があります。
- CS Linux ソフトウェアが停止して再始動された場合、または中央ロガーとして動作するサーバーが停止して他のサーバーが引き継ぐ場合は、フィルター処理される各ログ・メッセージのカウントがリセットされます。すなわち、CS Linux ソフトウェアの実行ごと、またはサーバーが中央ロガーを引き継ぐごとに、メッセージが 1 回だけログに記録されます。

使用量のログ記録

ローカル・ノードで、常にアクティブになっているリンクとセッションの数など、CS Linux のリソースの使用量を追跡したい場合があります。特に、CS Linux のリソースの使用量が、ライセンスで許可されている制限内にあることを確認するためには、この情報が必要になります。ライセンス交付要件について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 入門*」を参照してください。

CS Linux では、この情報にアクセスするために、次の 2 つの方法が用意されています。

- 30 分ごとに、現在の使用量とピーク使用量 (Linux コンピューターを再始動してからの任意時点での最大使用レベル) の詳細が、使用量のログ・ファイルというファイルに書き込まれます。このファイルの内容を調べることで、使用量の時間変化がわかります。
- 管理コマンド `query_node_limits` あるいは `NOF verb QUERY_NODE_LIMITS` を使用して任意時点における現在の使用量とピーク使用量の「スナップショット」を得ることができます。

使用量ログ・ファイル

CS Linux は、通常、使用量の情報をファイル `/var/opt/ibm/sna/sna.usage` に記録します。このログ・ファイルのサイズが 10,000,000 バイトになると、CS Linux はこのファイルの名前を `/var/opt/ibm/sna/bak.usage` に名前変更し (既存のバックアップ用の使用量のログ・ファイルは上書きされます)、このログ・ファイルを消去します。つまり、一度に保管できるログ情報の最大量は、指定された最大ファイル・サイズの 2 倍になります。

コマンド行管理プログラムを使用すると、監査ログ・ファイルやエラー・ログ・ファイルの場合と同様に、使用量のログ・ファイルの名前と保管場所、または最大ファイル・サイズを変更することができます。また、ログ・ファイルの現在の内容はいつでも消去することができます (この場合、情報をバックアップ・ファイルにコピーすることも、コピーしないこともできます)。以下のコマンドを使用します。

1. ログ・ファイルの名前または最大サイズを変更したり、ファイルの内容を消去するには、`snaadmin set_log_file` コマンドを使用します。
2. 現在の使用量のログ・ファイルの設定値を確認するには、`snaadmin query_log_file` コマンドを使用します。

これらのコマンドの発行について詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理コマンド解説書*」を参照してください。

使用量のログ・ファイルの形式は次のようになります。

- ファイルはいくつかの列に分割され、それぞれの列は次のような特定のリソース・タイプの使用量を記録します。
 - APPC アプリケーションと CPI-C アプリケーション
 - LUA アプリケーション
 - 活動状態であるリンク・ステーション
 - TN Server を使用する TN3270 セッション
 - TN リダイレクターを使用する Telnet セッション
 - データ・セッション (PLU-SLU セッション)
- 各列には 2 つの数値が表示されます。1 つは、記録時点における特定のリソース・タイプの現在の使用量、もう 1 つは、ピーク使用量 (Linux コンピューターを再始動してからの任意時点でのリソース・タイプの最大使用レベル) です。

- ファイル内の各行は、特定の時点におけるリソースの使用量の「スナップショット」で、行の最後にタイム・スタンプが表示されます。使用量は 30 分間隔で記録され、追加の項目は CS Linux が正常停止したときに含まれます。

ライセンス・マネージャーのトラッキング

CS Linux はライセンス・マネジメントのトラッキングの一環として、30 分ごとに最大同時ユーザー数を記録します。ログ・ファイルは `/var/opt/ibm/sna/ad8a7aebefee4d8e9554c14f5f734b7b.slmtag` です。各入力レコードには、以下のものが含まれます。

- ログ項目の時刻
- 対象の期間
- その期間のピーク・データと使用された TN リダイレクターの数。

このファイルは IBM Tivoli License Management (ITLM) エージェントにより読み取られて、同時ユーザー・ライセンスの使用をレポートします。CS Linux サーバーは同時ユーザーを使用してライセンス交付されない場合もあります。製品のライセンス交付の別の方法は、プロセッサ・バリュー・ユニット (PVU) です。CS Linux サーバーが PVU でライセンス交付されている場合、このメカニズムはライセンス・レポートで使用されません。

リソースの使用量を確認するための管理ツールの使用

特定の時点におけるリソースの使用量を確認するために、管理コマンド `query_node_limits` または `NOF verb QUERY_NODE_LIMITS` を使用することができます。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」または「*IBM Communications Server for Data Center Deployment on Linux NOF プログラマーズ・ガイド*」を参照してください。

コマンドまたは `verb` は、使用量のログ・ファイルの場合と同じリソース・タイプに関する情報(各リソースの現在の使用量と最大使用量)を戻します。また、CS Linux のライセンスが使用許可を与えている機能に関する情報も戻します。

トレース

この節では、CS Linux システムの稼働中に CS Linux トレース機能を使用して診断データを収集する方法と、トレース出力の作成方法について説明します。

注: トレース機能をオンにするのは、サポート担当員からトレース機能をオンにするよう要求された場合、または問題を診断するためにトレース出力を必要とする場合のみにしてください。トレースによりシステム・パフォーマンスが低下するため、このような状況以外ではすべてのトレースをオフにしてください。

次の操作は、Motif 管理プログラムとコマンド行管理プログラムのいずれでも実行できます。

- 様々なリンク・タイプの回線トレースを使用可能または使用不可にします。
- クライアント/サーバー・トレースを使用可能または使用不可にします。
- 内部トレースを使用可能または使用不可にします。

APPC、CPI-C、LUA、MS、CSV、および NOF API に対して API トレースを使用可能または使用不可にできます。通常、API トレースの設定には環境変数 `SNATRC` を使用しますが、環境変数、`verb`、プログラム・ファンクション能を組み合わせる必要がある場合もあります。

34 ページの図 1 は、トレースのそれぞれの主要なタイプが CS Linux の構造全体で発生するインターフェースを示しています。

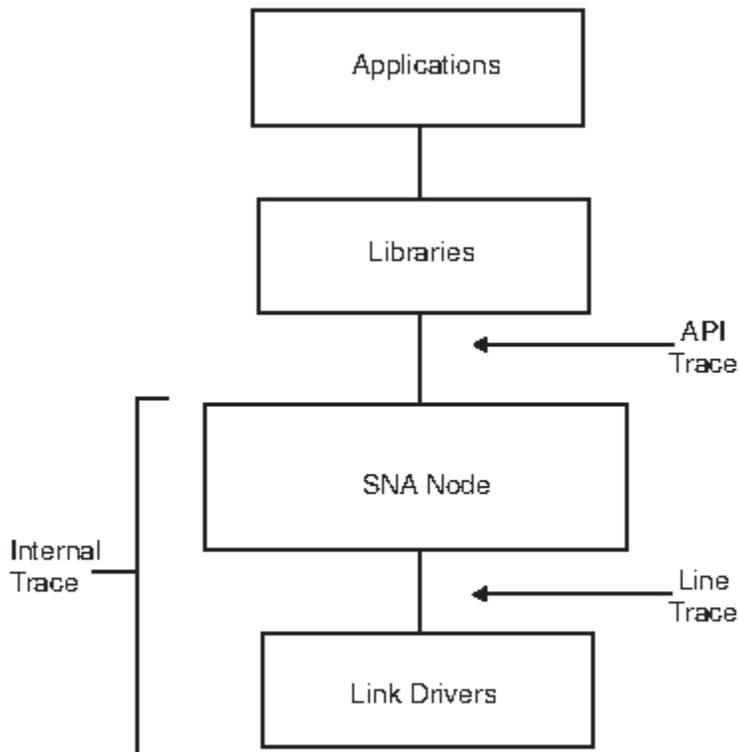


図 1. インターフェースのトレース

トレース機能では、データがテキスト・ファイルとバイナリー・ファイルに書き込まれます。書き込まれたテキスト・ファイルを表示するには、標準 ASCII テキスト・エディターを使用します。バイナリー出力の内容を ASCII テキスト・エディターで表示できるように、バイナリー出力をテキスト・ファイルに変換するには、`snatrcfmt` コマンド行ユーティリティーを使用します。詳しくは、52 ページの『内部バイナリー・トレース出力をテキスト・ファイルにフォーマットする』を参照してください。

以降の節では、CS Linux のトレース機能の各タイプについて説明します (最も有用なトレースから、エンド・ユーザーにとって実用性の高い順序で記載します)。

回線トレース

最も必要性の高いトレースが回線トレースです。回線トレースでは、リモート・システムへの通信リンクを流れる SNA メッセージがトレースの対象です。次の項では、Motif 管理プログラムおよびコマンド行管理プログラムでの回線トレースの実行方法について説明します。

回線トレースの実行方法

Motif 管理プログラムで回線トレースを実行する手順は次のとおりです。

1. コマンド行から `xsnaadmin` を入力します。Motif 管理プログラムが開始され、メイン・ウィンドウが表示されます。
2. 「診断 (Diagnostics)」メニューを選択し、次に「ノードのトレース (Node tracing)」を選択します。「トレース (Tracing)」ダイアログが表示されます。
3. 回線トレースの実行対象とする DLC を次から選択します。
 - トークンリング (token ring)
 - イーサネット (Ethernet)
 - SDLC

- X25
 - MPC チャネル・リンク (MPC Channel links) (マルチパス・チャネル)
 - Enterprise Extender のリンク (Enterprise Extender links)
4. オプションで「メッセージの切り捨て (Truncate messages)」を選択すれば、メッセージの最大サイズをバイト単位で入力することもできます。これは、大量のデータをトレースする際に、データそのものよりもプロトコルの交換を調べたい場合に役立ちます。プロトコル・データは、通常各メッセージの先頭に出力されます。
5. 終了したら、「了解 (OK)」をクリックします。

SDLC 回線をトレースするときに、さらに詳細な情報を必要とする場合は、回線トレースの場合と同様に SDLC についても内部トレースを使用して、この情報を取得します。この追加して取得した情報は回線トレースの出力の一部としてフォーマットされます。したがって、SDLC トレースのすべてが 1 つのファイルに収められます。詳しくは、[46 ページの『内部トレース』](#)を参照してください。

これでトレースが使用可能になりました。トレースは次のバイナリー・ファイルに書き込まれます。

- /var/opt/ibm/sna/sna1.trc
- /var/opt/ibm/sna/sna2.trc

コマンド行管理プログラムで回線トレースを実行する手順は次のとおりです。

1. デフォルト設定を使用しない場合は、`snaadmin set_trace_file` コマンドを使用してトレース・ファイル (1 つまたは複数) の属性を指定します。
 - トレースの出力先ファイルを 1 つまたは 2 つのどちらにするかを指定するには、パラメーター `dual_files` を追加します。
 - 2 つのファイルにトレースする場合は、次のように指定します。


```
snaadmin set_trace_file, dual_files = YES
```
 - 1 つのファイルにトレースするには、次のように指定します。


```
snaadmin set_trace_file, dual_files = NO
```
 - パラメーター `trace_file_size` を追加して、トレース・ファイルの最大サイズをバイト単位で指定します。(`dual_files` に「NO」が設定されていると、このパラメーターは無視されます。)例えば、以下のようになります。


```
snaadmin set_trace_file, trace_file_size = 10000000
```
 - パラメーター `file_name` と `file_name_2` を追加して、トレース・ファイルの名前を変更します。(`dual_files` に「NO」が設定されていると、パラメーター `file_name_2` は無視されます。)例えば、以下のようになります。


```
snaadmin set_trace_file, file_name = new1.trc, file_name_2 = new2.trc
```
2. 次のいずれかの手順を行います。
- 次のように指定して、すべての DLC、ポート、リンク・ステーション (LS) について回線トレースを開始します。


```
snaadmin add_dlc_trace
```
 - リソース・タイプ (ポート、リンク・ステーションなど) を指定して、回線トレースの容量を制御します。
 - トレースを特定のポートに制限するには、次のパラメーターをコマンドで指定します。


```
snaadmin add_dlc_trace, resource_type = PORT, resource_name = port_name
```
 - トレースを特定の LS に制限するには、次のパラメーターをコマンドで指定します。


```
snaadmin add_dlc_trace, resource_type = LS, resource_name = LS_name
```

これでトレースが使用可能になりました。トレース・ファイルの名前が変更されていない限り、トレースは次のバイナリー・ファイルに書き込まれます。

- /var/opt/ibm/sna/sna1.trc
- /var/opt/ibm/sna/sna2.trc

snaadmin add_dlc_trace コマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

SDLC 回線をトレースするときに、さらに詳細な情報を必要とする場合は、回線トレースの場合と同様に SDLC についても内部トレースを使用して、この情報を取得します。この追加して取得した情報は回線トレースの出力の一部としてフォーマットされます。したがって、SDLC トレースのすべてが 1 つのファイルに収められます。詳しくは、[46 ページの『内部トレース』](#)を参照してください。

バイナリー・トレース・ファイルのフォーマット設定

The snaadmin add_dlc_trace command generates a binary trace file or files containing only line trace messages. トレース・データはファイル /var/opt/ibm/sna/sna/sna1.trc および /var/opt/ibm/sna/sna2.trc に保管されます (snaadmin set_trace_file 管理コマンドを使用して別のトレース・ファイルまたはファイルを指定した場合を除く)。

バイナリー・ファイルを ASCII テキスト出力に変更するには、ディレクトリー /var/opt/ibm/sna を使用してに変更します。スナット cfmt ユーティリティーは、各トレース・ファイルにデフォルト・オプションを指定しています。

例えば、次のように指定します。

```
snastrcfmt -f sna1.trc -o sna1
```

```
snastrcfmt -f sna2.trc -o sna2
```

If you specified different trace files, replace sna1.trc and スクロウ 2.trc with the appropriate file names. For more information about スナット cfmt, see [52 ページの『内部バイナリー・トレース出力をテキスト・ファイルにフォーマットする』](#).

注: リモート・システムとの間でやり取りされるデータが (APPC モード、LS、またはセッションで使用される内部 PU の構成によって判別されたとおりに) 圧縮されている場合、トレース・フォーマッターはこのデータを圧縮解除したり、ASCII に変換したりしません。

ASCII テキスト・エディターを使用して、メッセージ・データ・ダンプ・ファイル (sna1.dmp および d2.dmp エビ) を表示します。これらのファイルには、SNA メッセージ・データが含まれます。各メッセージは 16 進データとしてリストされ、EBCDIC として解釈され、ASCII と In addition, the 木 and RH fields in the message header are interpreted as text strings preceding the message data.

対応するメッセージ・フロー・ドロー・ファイル (スネー・ドリユー および ドクロウ) は作成されません。トレースされるデータは、CS Linux からリモート・コンポーネント (ダイアグラムには表示されません) から、2 つの CS Linux コンポーネント間ではなく、リモート・コンポーネントに流れているため、メッセージ・フローの描画は行われません。

以下に示すのは、サンプル行トレース・ファイルです。

```
Sample Line Trace File
----- 17:21:04.040 PDT 22 Oct 1997
SND>> ACTPU +RSP LFSID:00000 TOKR0.TOKRP1.TOKRL1
TH: 2D000000BC9B BBIU EBIU EFI OAF:00 DAF:00 SNF:BC9B
RH: EB8000 SC FI
RU: 11124040 40404040 40400000 07010000 .. ..... @@@@.....
00000000 .....

----- 17:21:04.230 PDT 22 Oct 1997
SND>> ACTLU +RSP LFSID:02000 TOKR0.TOKRP1.TOKRL1
TH: 2D000002BC9C BBIU EBIU EFI OAF:02 DAF:00 SNF:BC9C
RH: EB8000 SC FI
RU: 0D020100 85800000 0C060100 01000000 ....e.....

----- 17:21:04.240 PDT 22 Oct 1997
SND>> NOTIFY RQD1 LFSID:02000 TOKR0.TOKRP1.TOKRL1
TH: 2C0000020000 BBIU EBIU OAF:02 DAF:00 SNF:0000
RH: 0B8000 FMD FI BC EC
```

```

RU: 8106200C 06030001 000000 a.....
----- 17:21:04.460 PDT 22 Oct 1997
<+RSP LFSID:02000 TOKR0.TOKRP1.TOKRL1
TH: 2C0000020001 BBIU EBIU OAF:02 DAF:00 SNF:0001
RH: 838000 FMD
----- 17:21:04.550 PDT 22 Oct 1997

```

API トレース

API トレースは、API を使用するプログラムに関連する通信の問題を検出する際に役立ちます。Motif 管理プログラムとコマンド行管理プログラムでは、API トレースを使用可能または使用不可にすることはできません。ほとんどの場合、SNATRC 環境変数を使用してトレースをセットアップします。アプリケーションの実行中にトレースを制御できるようにする場合は、環境変数、verb、およびプログラム・ファンクションを組み合わせて使用する必要があります。

CS Linux では、次の API のためのトレースがサポートされています。

- APPC
- CPI-C
- LUA
- MS
- CSV
- NOF

verb 制御ブロック (VCB) を使用する APPC、CSV、LUA、MS、および NOF API の場合は、トレース・ファイルには、VCB の内容のダンプ (16 進数、ASCII 変換、EBCDIC 変換) が含まれます。さらに詳細に VCB の内容を解釈する必要がある場合は、API トレース・フォーマッター・ユーティリティ `snaapitrcfmt` を使用して、トレース・ファイルを処理することができます。詳細については、[41 ページの『API トレース・フォーマッター』](#)を参照してください。snaapitrcfmt ユーティリティは CPI-C トレースの解釈は行ないません。CPI-C は VCB ではなく関数呼び出しを使用するからです。個々のパラメーターは、すでにトレース・ファイルに書き込まれており、それ以上解釈する必要はありません。

API トレースの実行方法

API トレースを設定して制御する場合は、API を使用するアプリケーション・プログラムを開始する前に、次の手順を行ってください。

1. 環境変数 SNATRC を使用して、1 つまたは 2 つのトレース・データ・ファイルを指定し、アプリケーション開始時にトレースを開始するかどうかを指定します。環境変数 SNATRC の構文は次のとおりです。

```
file1[:[file2][:]]
```

次のパラメーターを指定します。

file1

トレース・ファイルの名前

file2

2 番目のトレース・ファイル名。これは任意指定です。2 つのファイル名を区切るにはコロンを使用します。

最後のコロン (:)

最後のコロンは任意指定です。これは、アプリケーションの開始と同時にトレースを活動状態にすることを指定します。アプリケーションの開始時にトレースが活動状態にならない場合には、そのアプリケーション内から `CSV DEFINE_TRACE verb` を使用してトレースを開始します。ファイル名を 1 つのみ指定した場合は、コロンを 2 つ使用するとアプリケーションの開始と同時にトレースも活動状態になります。

例えば、以下のようになります。

- コマンド行に `export SNATRC=file1.trc:file2.trc` を入力すると、トレースの出力先が 2 つのファイルになり、API アプリケーションが開始しても、トレースは活動状態になりません。ただし、

あとでトレースを開始することもできます。これについては [38 ページの『アプリケーション内からの API トレースの制御』](#) で説明します。

- コマンド行に `export SNATRC=file.trc::` を入力すると、トレースの出力先は 1 つのファイルになり、API アプリケーションの開始時にトレースが開始します。
2. ファイル名を 2 つ指定した場合は、環境変数 `SNATRACESIZE` を使用して各トレース・ファイルの最大サイズを設定します。(ファイル名を 1 つのみ指定した場合は、トレース・ファイルのサイズに制限はありません。) `SNATRACESIZE` の使用法の詳細については、[38 ページの『トレース・ファイルの内容の制御』](#) を参照してください。

トレース・データの書き込み先テキスト・ファイルのパスを指定します。デバイス名 (`/dev/tty` など) や印刷スプーラーをトレース・ファイルとして指定しないでください。CS Linux ではトレース・ファイルに絶対パスを指定しないと、アプリケーションの開始元のディレクトリーが使用されます。

アプリケーション内からの API トレースの制御

CS Linux では、アプリケーション・プログラム内から API トレースを制御できます。プログラム内で問題が発生する特定のセクションをトレースすることで、アプリケーション全体をトレースする必要がなくなります。セクションの始めにトレースをアクティブ化し、セクションの終わりで非アクティブ化することができます。

この機能を使用するには、次の手順を行います。

1. アプリケーション・プログラムを開始する前に、環境変数 `SNATRC` を設定します ([37 ページの『API トレースの実行方法』](#) を参照してください)。環境変数を設定する際に、アプリケーションの開始時にトレースを活動状態にするかしないかを指定できます。必要に応じて、アプリケーション内からのトレースのアクティブ化または非アクティブ化ができます。
2. `CSV DEFINE_TRACE verb` を使用して、プログラム内でトレースを制御します (すなわち、トレースを使用可能または使用不可にします)。(詳しくは、「*IBM Communications Server for Data Center Deployment on AIX or Linux 共通サービス Verb プログラマーズ・ガイド*」を参照してください。)

アプリケーションによるトレース制御を使用不可にする

環境変数 `SNACTL` は、API トレース制御するアプリケーション・プログラムの機能をオーバーライドします。`SNACTL` は、通常はいくつかの機能のトレースをオフにする アプリケーション・プログラム全体のトレースを強制的に実行したり、通常はトレースを使用するアプリケーション・プログラムのトレースを停止させます。アプリケーション・プログラムの開始前に `SNACTL` を設定しておくで、アプリケーション・プログラムにより発行されるトレース制御コマンドは無視されます。トレースがオンになっている場合はオンの状態、オフになっている場合はオフの状態になります。ただし、発行したトレース制御コマンドからの戻りコードには、実行が正常に終了したことが示されます。

`SNACTL` を使用するには、この変数に任意のストリング (`export SNACTL=1` など) を設定します。`SNACTL` を取り消すには、ヌル・ストリングを設定します。

自動的に開始した起動可能 TP のトレースの制御

自動的に開始した起動可能トランザクション・プログラム (TP) をトレースするには、`snatpinstall` プログラムの実行時に適切な環境変数を構成します。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください。

トレース・ファイルの内容の制御

次の環境変数は、トレース・ファイルに保管されるデータの容量を制御します。

SNATRUNC

トレース・ファイルの各エントリーの最大長をバイト単位で指定します。この変数には 10 進数の数値を設定してください。メッセージの文字数がこの変数に設定された値よりも大きい場合、超過した分の文字は切り捨てられます。たとえば、`SNATRUNC` を 70 に設定すると、エントリー当たりのトレースのデータ長は 70 バイトに制限されます。デフォルトでは、API トレース・メッセージは切り捨てられません。

SNATRACESIZE

2つのファイルを使用する場合の各トレース・ファイルの最大サイズを、バイト単位で指定します。1つのファイルにトレース出力する場合は、トレース・ファイルのサイズに制限はありません。

この環境変数には10進数の数値を設定してください。file1のサイズが最大ファイル・サイズに達すると、CS Linuxによりfile2の内容が消去され、続いてfile2へトレースが出力されます。file2が同じ限界に達すると、CS Linuxによりfile1の内容が消去され、トレース情報が再度file1へ書き込まれます。これにより、一組のAPIトレース・ファイルにより占有される最大ディスク・スペースは、SNATRACESIZEの値の約2倍となります。2つのファイルを使用することで、トレース期間が拡大し、さらにディスク・スペースの使用量の制限もSNATRACESIZEで指定されている値の2倍になります。

SNATRACESIZEを設定しない場合のデフォルトは10,000,000バイトです。SNATRACESIZEの設定を取り消してデフォルトに戻す場合は、SNATRACESIZEにヌル・ストリングを設定してください。

SNATRCRESET

アプリケーションにより初めてトレース・ファイルへの書き込みを実行する時に、トレース・ファイルをリセットするかどうかを指定します。アプリケーションが最初のトレース・メッセージをファイルに書き込む時には、通常ファイルはリセットされ、内容はすべて破棄されます。複数のアプリケーションをトレースして1つのファイルに出力する場合、または同一のアプリケーションの複数の実行インスタンスをトレースして同一のファイルに出力する場合に、ファイルがリセットされるのを防ぐには、環境変数SNATRCRESETにNOを設定します。

2つのファイルへトレースを出力する場合、最大ファイル・サイズに達すると通常はリセットされますが、アプリケーションでトレースが初めて開始された時点ではリセットされません。1つのファイルへトレースを出力する場合は、SNATRCRESETをNOに設定すると、ファイルの自動リセットは行われません。ディスク・スペースを占有しすぎないようにするため、定期的にファイルを手動で削除してください。

アプリケーションが最初にトレースをファイルに出力する際にファイルをリセットするように、SNATRCRESETの設定値を取り消してデフォルトの設定値に戻すには、SNATRCRESETをヌル・ストリングに設定します。

API トレースのトレース・ファイルの形式

1つのメッセージのトレース・データは、トレース・ファイル内で1行以上の容量を占有できます。各メッセージの前には、トレース・エントリーの作成時を示す横線が出力されます。CS Linux API トレース・ファイルについて、次に説明します。

- トレースされるコンポーネントのプロセス ID が各行の先頭に出力されます。プロセス ID のあとにスレッド ID が出力されます(プロセス ID とスレッド ID はピリオドで区切られます)。シングルスレッドのアプリケーションの場合は、スレッド ID は常に 00 です。プロセス ID のあとに、トレース対象のコンポーネントのタイプを示すインジケータが示されます(たとえば APPC TP の場合は APPC)。
- メッセージ・データは次の形式で個別の列に出力されます。これにより、メッセージ・データの文字ストリングが、文字セットに応じて EBCDIC または ASCII のいずれかの列に可読テキストとして出力されます。
 - 16 進数
 - EBCDIC 変換
 - ASCII 変換

トレース・データの形式は、API によって多少異なります。

APPC、NOF、MS

verb が発行される時と戻される時に、アプリケーションから対応 API ライブラリーへ提供される verb 制御ブロックがトレースされます。非同期エントリー・ポイントを使用して発行される verb と、同期エントリー・ポイントを使用して発行される APPC [MC_]RECEIVE_AND_POST verb の場合、最初の戻り(verb が正常に発行され、処理中であることを示す)と(verb の完了時の)コールバック・ルーチンへの戻りがトレースされます。

トレースの各セクションの先頭には、発行された verb の名前と 戻りの結果 (1 次戻りコードから取得) が出力されます。verb 制御ブロック (VCB) に続いて、送信データまたは戻りデータもデータの保管先アドレス (verb 制御ブロックから取得) と共にトレースされます。

MS または NOF アプリケーションが指示を受信するように登録されている場合、各指示は非同期の verb の戻りと同様の形式でトレースされます。

VCB の構造体とこれらの各 API の内容については、「*IBM Communications Server for Data Center Deployment on AIX or Linux APPC プログラマーズ・ガイド*」、「*IBM Communications Server for Data Center Deployment on Linux NOF プログラマーズ・ガイド*」、および「*IBM Communications Server for Data Center Deployment on AIX or Linux MS プログラマーズ・ガイド*」を参照してください。

CSV

verb が発行される時と戻される時に、CSV ライブラリーへ 提供される verb 制御ブロックがトレースされます。トレースの各セクションの先頭には、発行された verb の名前と 戻りの結果 (1 次戻りコードから取得) が出力されます。verb 制御ブロックに続いて、verb に含まれる データ・ストリングもすべて、データの保管先アドレス (verb 制御ブロックから取得) と共にトレースされます。

詳しくは、「*IBM Communications Server for Data Center Deployment on AIX or Linux 共通サービス Verb プログラマーズ・ガイド*」を参照してください。

LUA

verb が発行される時と戻される時に、LUA ライブラリーへ 提供される verb 制御ブロックがトレースされます。

verb が非同期的に戻りコードを戻す (*lua_flag2.async* ビットが 1 にセットされ、1 次戻りコードが *LUA_IN_PROGRESS* にセットされる) 場合、verb 制御ブロック は完了時に 3 度トレースされます。この場合、最初の戻り VCB (*lua_flag2.async* が 1 にセットされ、1 次戻りコードが、非同期 verb 戻りコードであることを示す *LUA_IN_PROGRESS* にセットされている場合以外) 内のパラメーターを無視し、verb が完了したとき、最終の戻り VCB 内のパラメーターのみを考慮すべきです。この VCB 内で、*lua_flag2.async* は 1 にセットされたままになり、1 次戻りコードは *LUA_IN_PROGRESS* 以外の値にセットされます。

トレースの各セクションの先頭には、発行された LUA verb の名前と 戻りの結果 (1 次戻りコードおよび 2 次戻りコードから取得) が出力されます。verb 制御ブロックに続いて、送信データまたは戻りデータもすべて、データの保管先アドレス (verb 制御ブロックから取得) と共にトレースされます。

CS Linux は RUI verb を使用して SLI を実装しているので、LUA ライブラリーは SLI verb を対応する RUI verb (ここで各 SLI verb は 1 つまたは複数の RUI verb になる) に変換します。そのため、SLI トレースは SLI パラメーターおよび RUI パラメーターの双方を含みます。最初に SLI 要求がトレースされ、次に RUI 要求、および各 verb の戻りコード (該当する場合、後の非同期戻りコード) がトレースされ、最後に SLI 戻りコードがトレースされます。

詳しくは、「*IBM Communications Server for Data Center Deployment on AIX or Linux LUA プログラマーズ・ガイド*」を参照してください。

CPI-C

CS Linux では APPC を使用して CPI-C を実装するため、ほとんどの CPI-C 呼び出しが CPI-C ライブラリーによって対応する APPC verb に変換されます。したがって、CPI-C トレースには CPI-C パラメーターと APPC パラメーターが含まれています。トレースは、CPI-C 要求、APPC 要求、APPC 戻り、CPI-C 戻りの順序で行われます。ローカル情報のみを処理するその他の CPI-C 機能 (受信タイプや同期レベルの確認または設定など) の場合、APPC verb は実行されないため、トレースでは CPI-C パラメーターのみが示されます。

トレースの各セクションの先頭には、発行された CPI-C 呼び出しまたは APPC verb の名前と その戻りコードが出力されます。CPI-C パラメーターまたは APPC verb 制御ブロックに続いて、送信データまたは戻りデータもすべて、データの保管先アドレス (verb 制御ブロックから取得) と共にトレースされます。

詳しくは、「*IBM Communications Server for Data Center Deployment on AIX or Linux CPI-C プログラマーズ・ガイド*」を参照してください。

以下の例は、CPI-C アプリケーションおよび APPC アプリケーションの API トレース・ファイルの一部を示したものです。

```
Sample API Trace File Fragment: CPI-C
=====
===== Initialized 14:40:35 BST 15 Sep 1997 =====
=====
| 4849.00 CPIC ----- 14:40:35.07 BST 15 Sep 1997
| 4849.00 CPIC CMINIT request
| 4849.00 CPIC Sym dest name =
| 4849.00 CPIC ----- 14:40:35.08 BST 15 Sep 1997
| 4849.00 CPIC CMINIT response, result = CM_OK
| 4849.00 CPIC Conversation ID = 01000001
| 4849.00 CPIC Conversation characteristics
| 4849.00 CPIC Conversation type = CM_MAPPED_CONVERSATION
| 4849.00 CPIC Deallocate type = CM_DEALLOCATE_SYNC_LEVEL
| 4849.00 CPIC Error direction = CM_RECEIVE_ERROR
| 4849.00 CPIC Sync level = CM_NONE
| 4849.00 CPIC Fill type = CM_FILL_LL
| 4849.00 CPIC Prepare to receive type = CM_PREP_TO_RECEIVE_SYNC_LEVEL
| 4849.00 CPIC Receive type = CM_RECEIVE_AND_WAIT
| 4849.00 CPIC Send type = CM_BUFFER_DATA
| 4849.00 CPIC Conversation security type = XC_SECURITY_SAME
| 4849.00 CPIC Log data pointer = 0
| 4849.00 CPIC Log data length = 0
| 4849.00 CPIC Sym dest name =
| 4849.00 CPIC Partner LU name =
| 4849.00 CPIC 20202020 20202020 20202020 20202020 .....
| 4849.00 CPIC 20 .....@@@@@@@@@@@@@@@@
| 4849.00 CPIC Mode name =
| 4849.00 CPIC 40404040 40404040 @@@@@@@@@@@@@@@@@@
| 4849.00 CPIC Partner TP name =
| 4849.00 CPIC 40404040 40404040 40404040 40404040 @@@@@@@@@@@@@@@@@@
```

```
Sample API Trace Fragment: APPC
| 2511.00 APPC ----- 14:49:08.04 BST 20 Oct 1998
| 2511.00 APPC TP_STARTED request
| 2511.00 APPC ---- Verb Parameter Block at address 40001578 ----
| 2511.00 APPC 00140000 00000000 00000000 54504C55 .....&<.
| .....TPLU
| 2511.00 APPC 31202020 00000000 00000000 E3D7D5C1 .....TPNA 1 .....
| 2511.00 APPC D4C5F140 40404040 40404040 40404040 ME1 .....@@@@@@@@@@@@@@@@
| 2511.00 APPC 40404040 40404040 40404040 40404040 @@@@@@@@@@@@@@@@@@
| 2511.00 APPC 40404040 40404040 40404040 40404040 @@@@@@@@@@@@@@@@@@
| 2511.00 APPC 40404040 40404040 40404040 00000000 .....
@@@@@@@@@@@@@@@@....
```

APPC、CSV、LUA、MS、および NOF API の場合には、API トレース・フォーマッター・ユーティリティ `snaapitrcfmt` (Linux 上の 64 ビット・アプリケーションに対しては `snaapitrcfmt64`) を使用して、VCB の内容をさらに詳細に解釈することができます。詳細については、41 ページの『API トレース・フォーマッター』を参照してください。

API トレース・フォーマッター

`snaapitrcfmt` コマンド行ユーティリティを使用し、VCB 内の各パラメーターの内容を解釈して、それをプレーン・テキストとして提供することにより、APPC、CSV、LUA、MS、および NOF トレース・ファイルの VCB の内容をさらに詳細に展開することができます。このユーティリティは、標準の CS Linux API トレース・ファイルを入力として受け取り、詳細に展開されたトレースを新規のテキスト・ファイルに書き込みます。

注: トレース・ファイルが 64 ビット・アプリケーションで生成された場合 (該当する API に対する「プログラマーズ・ガイド」で説明されているように、これはアプリケーションをコンパイル、リンクするとき指定される)、`snaapitrcfmt` の代わりに `snaapitrcfmt64` を使用する必要があります。この項内の `snaapitrcfmt` はすべて `snaapitrcfmt64` と読み替えてください。

`snaapitrcfmt` ユーティリティは CPI-C トレースを展開しません。CPI-C に渡す関数パラメーターが、すでに標準のトレース・ファイル形式で解釈済みのためです。ただし、入力ファイルの CPI-C トレースはすべて未変更のまま出力ファイルに書き込まれるので、入力ファイルに複数のトレース・タイプが含まれ

ていてもその CPI-C トレースが失われることはありません。特に、CS Linux CPI-C は APPC インターフェース上で実装されるので、CPI-C トレース・ファイルにも基盤の APPC VCB のトレースが含まれます。snaapitrcfmt ユーティリティを使用して CPI-C トレース・ファイルを処理すると、出力ファイルには、オリジナルの CPI-C トレースが未変更のまま、APPC VCB の詳細に展開されたものと一緒に含まれます。

API トレース・フォーマット・ユーティリティを実行するためのコマンド構文は次のとおりです。

```
snaapitrcfmt [-f source_file_1[:source_file_2]] [-o output_file] [-h]
```

次のオプションとパラメーターを指定します。

-f source_filenames

このオプションは、1 つ以上の入力 API トレース・ファイルの名前を指定するのに使用します。SNATRC 環境変数が正しい入力ファイルの名前に設定されている場合は、このオプションを指定する必要はありません。

- フォーマットするトレースが単一のファイル、たとえば myapi.trc にある場合は、次の形式を使用します。

```
-f myapi.trc
```

- フォーマットするトレースが 2 つのトレース・ファイル、たとえば myapi1.trc と myapi2.trc にある場合は、次の形式を使用します。

```
-f myapi1.trc:myapi2.trc
```

トレース・ファイルの絶対パスを指定しないと、snaapitrcfmt はアプリケーションを開始したディレクトリーを使用します。-f オプションを使用しないと、snaapitrcfmt は SNATRC 環境変数で指定されたファイルを使用します。SNATRC が設定されていないと、このユーティリティは (現行のディレクトリー内の) snaapi.trc をデフォルトとして使用します。

-o output_file

snaapitrcfmt で作成される出力ファイルの名前。このファイルがすでに存在する場合は、そのファイルの内容は snaapitrcfmt の出力で置き換えられます。

出力ファイルの絶対パスを指定しないと、snaapitrcfmt はアプリケーションを開始したディレクトリーを使用します。-o オプションを指定しないと、snaapitrcfmt は (現在のディレクトリー内の) snaapi.dmp を出力ファイルとして使用します。

-h

snaapitrcfmt ユーティリティのヘルプ情報を表示します。

以下の例は、APPC トレース・ファイルの出力の一部を示したものです。

```
Sample Formatted API Trace Fragment: APPC
| 2511.00 APPC ----- 14:49:08.04 BST 20 Oct 1998
| 2511.00 APPC TP_STARTED request
| 2511.00 APPC ---- Verb Parameter Block at address 40001578 ----
| 2511.00 APPC 00140000 00000000 00000000 54504C55 .....&<.
| .....TPLU
| 2511.00 APPC 31202020 00000000 00000000 E3D7D5C1 .....TPNA 1
| 2511.00 APPC D4C5F140 40404040 40404040 40404040 ME1 .....
| 2511.00 APPC 40404040 40404040 40404040 40404040 @@@@@@@@@@@@@@@@
| 2511.00 APPC 40404040 40404040 40404040 40404040 @@@@@@@@@@@@@@@@
| 2511.00 APPC 40404040 40404040 40404040 00000000 .....
@@@@@@@@@@@@@...
| 2511.00 APPC
| 2511.00 APPC opcode = 14
| 2511.00 APPC opext = 0
| 2511.00 APPC format = 0
| 2511.00 APPC primary_rc = 0 OK
| 2511.00 APPC secondary_rc = 0 OK
| 2511.00 APPC lu_alias[8] = 54504C5531202020
| 2511.00 APPC . & < . . . . .
| 2511.00 APPC T P L U
| 2511.00 APPC tp_id[8] = 0000000000000000
| 2511.00 APPC @ @ @ @ @ @ @ @ @ @
```

```

| 2511.00 APPC
| 2511.00 APPC tp_name[64] = E3D7D5C1D4C5F140
| 2511.00 APPC T P N A M E 1
| 2511.00 APPC . . . @ @ @ @
| 2511.00 APPC delay_start = 0
| 2511.00 APPC enable_pool = 0
| 2511.00 APPC pip_dlen = 0

```

クライアント/サーバー・トレース

クライアント/サーバー・トレースは、同一ドメイン内の CS Linux サーバー間および CS Linux サーバーとクライアント間を流れるメッセージを記録します。トレースは 2 つの特定のコンピューター間、またはコンピューターと LAN 上の他のすべてのサーバー間を流れるデータに対してアクティブにできます。送信データあるいは受信データのみがアクティブになるかすべてのデータがアクティブになります。次の項では、Motif 管理プログラムおよびコマンド行管理プログラムでのクライアント/サーバー・トレースの実行方法について説明します。

Motif 管理プログラムによるクライアント/サーバー・トレースの実行

Motif 管理プログラムを使用してクライアント/サーバー・トレースを実行する手順は次のとおりです。

1. コマンド行から `xснаadmin` を入力します。Motif 管理プログラムが開始され、メイン・ウィンドウが表示されます。
2. 「診断 (Diagnostics)」メニューを選択し、次に「ノードのトレース (Node tracing)」を選択します。「トレース (Tracing)」ダイアログが表示されます。
3. このサーバー、ドメイン内のクライアントと他のサーバー間のメッセージのトレースをオンにするため、「クライアント/サーバー (client-server)」を選択します。
4. 「メッセージの切り捨て (*Truncate messages*)」を選択して、メッセージの最大サイズをバイト単位で入力することもできます。これは、大量のデータをトレースする際に、データそのものよりもプロトコルの交換を調べたい場合に役立ちます。プロトコル・データは、通常各メッセージの先頭に出力されません。
5. 終了したら、「了解 (OK)」をクリックします。

これでトレースが使用可能になりました。トレースは次のテキスト・ファイルに書き込まれます。

- `/var/opt/ibm/sna/snacs1.trc`
- `/var/opt/ibm/sna/snacs2.trc`

コマンド行管理プログラムを使用したクライアント/サーバーのトレースの実行

コマンド行管理プログラムを使用してクライアント/サーバーのトレースをセットアップおよび制御するには、以下のステップを実行します。

1. `snaadmin set_cs_trace` コマンド: の指定
2. Add the `dest_sys` parameter to the `snaadmin` command to specify the client or server name for which tracing is required (this is an ASCII string). `dest_sys` パラメーターを指定すると、このコマンドが発行されたコンピューターと LAN 上の他の 1 つのサーバーとの間に流れるメッセージのトレースを管理することができます。
このコマンドが発行されたコンピューターと、LAN 上の他のすべてのコンピューターとの間で流れるメッセージのトレースを管理するには、`dest_sys` パラメーターを指定しないでください。
3. `trace_flags` パラメーターを `snaadmin` コマンドは、すべてのトレースをオンまたはオフにするか、または特定のメッセージ・タイプのトレースを活動化します。に追加します。
 - すべてのトレースをオンまたはオフにするには、次のいずれかの値を指定します。

オール

すべてのタイプのトレース

なし

トレースなし

- 1つ以上のメッセージ・タイプのトレースをアクティブにするには、以下の値を1つ以上指定します (値を結合するには、+文字を使用します)。

CS_ADMIN_MSG

クライアント/サーバー・トポロジーに関連する内部メッセージ

CS_DATAGRAM

内部データグラム・メッセージ

CS_DATA

データ・メッセージ

4. 以下の値のいずれかを使用して *trace_direction* パラメーターを追加し、トレースが必要となる方向または方向を指示します (このパラメーターは、*trace_flags* が なしに設定されている場合は無視されます)。

CS_SEND

ローカル・コンピューターから流れるトレース・メッセージ

CS_RECEIVE

ローカル・コンピューターに流れるトレース・メッセージ

両方を指定

両方向に流れるトレース・メッセージ

5. コマンドへのパラメーターと値の追加が終了したら、Enter キー. を押します。

トレースが使用可能になり、以下のテキスト・ファイルに書き込まれるようになりました。

- /var/opt/ibm/sna/snacs1.trc
- /var/opt/ibm/sna/sn/snacs2.trc

Linux コンピューターでのクライアント/サーバー・トレースの制御の詳細については、*snaadmin set_cs_trace* and *snaadmin query_cs_trace* commands in the *IBM Communications Server for Data Center* デプロイメント on Linux 管理コマンド解説書」の説明を参照してください。(*snaadmin query_cs_trace* コマンドは、CS Linux LAN 上のコンピューター間で送信されるデータの現在のトレース・オプションに関する情報を戻します。) Windows クライアントでのクライアント/サーバー・トレースの制御については、*IBM Communications Server for Linux* 管理ガイド上の「データ・センター・デプロイメント」を参照してください。

クライアント/サーバー・トレース・ファイルの内容

クライアント/サーバー・トレース・データは、次のテキスト・ファイル (標準の ASCII テキスト・エディターで表示できる) に書き込まれます。

/var/opt/ibm/sna/snacs1.trc

クライアント/サーバー・トレース・ファイル

/var/opt/ibm/sna/sn/snacs2.trc

バックアップ・クライアント/サーバー・トレース・ファイル

各行の冒頭の省略語 SLM.BS は バークレー・ソフトウェア・ディストリビューション (BSD) ソケットの SNA LAN インターフェース・モジュール (SLIM) を示します。このプロセス ID のあとに、トレース・タイプのインジケーターが示されます (TCP または UDP)。各エントリーはイベント (接続の確立、メッセージの送信または受信) を説明し、関連するメッセージ・データを含みます。API verb タイプや戻りコードなど、メッセージに関する基本情報はデコードされます。

メッセージ・データは、16 進値、EBCDIC 変換、ASCII 変換の 3 つの列に表示されます。したがって、メッセージ・データのテキスト・ストリングは、テキストの文字セットによって EBCDIC 列または ASCII 列に可読文字として表示されます。

TN Server トレース

TN Server トレースを使用すると、LAN を介した CS Linux TN サーバーと、その TN3270 クライアントとの間のメッセージ・フローを記録できます。次の項では、コマンド行管理プログラムでの TN Server トレースの実行方法について説明します。

TN Server トレースの実行方法

コマンド行管理プログラムで TN Server トレースを実行する手順は次のとおりです。

1. デフォルト設定を使用しない場合は、`snaadmin set_trace_file` コマンドにパラメーター `trace_file_type` を指定して、TN Server トレース・パラメーターを指定します。
2. 次の管理コマンドを指定して、TN Server トレースを開始します。このコマンドは、稼働しているノードに対して発行してください。

```
snaadmin set_tn_server_trace
```

3. パラメーター `trace_flags` を `snaadmin` コマンドに追加して、全トレースをオンまたはオフにするか、あるいは特定のメッセージのタイプのトレースを開始します。

- 全トレースをオンまたはオフにするには、次のいずれかの値を指定します。

ALL

すべてのタイプをトレースします。

NONE

トレースを実行しません。

- 1つ以上のメッセージのタイプのトレースを開始するには、次に示す値を1つ以上指定します。値を結合する場合には、+ 文字を使用してください。

TCP

TCP/IP インターフェース・トレース : TN Server と TN3270 クライアントの間のメッセージ

FMAPI

ノード・インターフェース・トレース : 内部制御メッセージ、TN Server および TN3270 クライアント間のメッセージ (内部形式)

注 : 通常、第 1 レベルの診断では FMAPI を指定する必要はありません。

CFG

構成メッセージ・トレース : TN Server の構成に関連するメッセージ

4. コマンドにパラメーターと値を追加し終わったら、Enter を押します。

これでトレースが使用可能になりました。トレースは次のテキスト・ファイルに書き込まれます。

- /var/opt/ibm/sna/snatsv1.trc
- /var/opt/ibm/sna/snatsv2.trc

CS Linux TN Server フィーチャーの現行トレース・オプションについての情報は、`snaadmin query_tn_server_trace` コマンドを使用して入手することもできます。このコマンドと `snaadmin set_tn_server_trace` コマンドについて詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理コマンド解説書」を参照してください。

TN Server トレース・ファイルの内容

TN Server のトレース・データは、/var/opt/ibm/sna/snatsv1.trc と /var/opt/ibm/sna/snatsv2.trc に書き込まれます。これらの内容は、標準の ASCII テキスト・エディターを使用して表示することができます。

メッセージ・データは、16 進値、EBCDIC 変換、ASCII 変換の 3 つの列に表示されます。したがって、メッセージ・データのテキスト・ストリングは、テキストの文字セットによって EBCDIC 列または ASCII 列に可読文字として表示されます。

内部トレース

内部トレースでは、CS Linux プロセス間 (ローカル・ノードおよび接続コンポーネント間) のデータ・フローがトレースされます。CS Linux の全体構造と、`snaadmin set_trace_type` コマンドを使用して指定できる内部トレースのタイプを示す図については、[34 ページの図 1](#) を参照してください。次の項では、Motif 管理プログラムおよびコマンド行管理プログラムでの内部トレースの実行方法について説明します。

内部トレースの内容は詳細であるため、通常このトレースを使用するのは CS Linux サポート担当員のみです。

内部トレース・ファイルの制御

`sna start -t` を発行して SNA を開始した場合は、すべてのトレースを使用可能にできます。

Motif 管理プログラムで内部トレースを実行する手順は次のとおりです。

1. コマンド行から `xsnaadmin` を入力します。Motif 管理プログラムが開始され、メイン・ウィンドウが表示されます。
2. 「診断 (Diagnostics)」メニューを選択し、次に「ノードのトレース (Node tracing)」を選択します。「トレース (Tracing)」ダイアログが表示されます。
3. 「すべてのトレースをオンにする (Set all tracing on)」を選択して、すべての内部トレースをオンにします。または、ダイアログの「サーバー・メッセージ・トレース」セクションのオプションを選択し、CS Linux の特定の領域のトレースを 1 つ以上オンに指定します (関係のないトレース情報を大量に収集せずに、この領域での問題を診断するためです)。これらのオプションを指定すると、次の領域をトレースするよう指定することができます。
 - 特定の API または関連する API のグループ
 - TN Server
 - SDLC (これは内部トレースのその他のトレース・タイプから分離して制御されます。その他のトレース・タイプは、回線トレース・ファイルに追加の詳細なトレース情報を提供する場合に使用できるため、`Node` オプションを使用して制御されます。)
 - ノード: このダイアログに表示される特定のオプションに対するタイプを除く、内部トレースのすべてのタイプ
4. 「メッセージの切り捨て (Truncate messages)」を選択して、メッセージの最大サイズをバイト単位で入力することもできます。これは、大量のデータをトレースする際に、データそのものよりもプロトコルの交換を調べたい場合に役立ちます。プロトコル・データは、通常各メッセージの先頭に出力されません。
5. 選択し終わったら、「了解 (OK)」をクリックします。

これでトレースが使用可能になりました。トレースは次のバイナリー・ファイルに書き込まれます。

- `/var/opt/ibm/sna/sna1.trc`
- `/var/opt/ibm/sna/sna2.trc`

コマンド行管理プログラムで内部トレースを実行する手順は次のとおりです。

1. 必要であれば、`snaadmin set_trace_file` コマンドにパラメーター `trace_file_type` を指定します。

```
snaadmin set_trace_file, trace_file_type = IPS
```
2. パラメーターを指定したあとで、「Enter」を押します。
3. パラメーター `trace_flags` を `snaadmin set_trace_type` コマンドに追加して、全トレースをオンまたはオフにするか、あるいは特定のメッセージのタイプのトレースを活動状態にすることを指定します。
 - 全トレースをオンまたはオフにするには、パラメーター `trace_flags` に次のいずれかの値を指定します。

ALL

すべてのタイプをトレースします。

NONE

トレースを実行しません。

- 次のメッセージのタイプのうち 1 つ以上のタイプのトレースを活動状態にするには、次に示す値を 1 つ以上指定します。値を結合する場合には、+ 文字を使用してください。

APPC

APPC ライブラリーとノードの間で送信されるメッセージ

FM

3270 エミュレーション・プログラムとノードの間で送信されるメッセージ

LUA

LUA ライブラリーとノードの間で送信されるメッセージ

SLI アプリケーションの場合、ライブラリーは SLI verb をノードに送信する前に、対応する RUI verb に変換します。これは LUA の内部トレースは RUI verb のみを含むことを意味します。SLI verb のすべての問題を診断するには API トレースを使用してください。

NOF

NOF ライブラリーとノードの間で送信されるメッセージ

MS

MS ライブラリーとノードの間で送信されるメッセージ

NDLC

APPN ノードと DLC コンポーネントの間で送信されるメッセージ

LLC2

LLC2 ソフトウェアのレイヤーの間で送信されるメッセージ

MAC

LLC2 ソフトウェアのレイヤーの間で送信されるメッセージ

LLI

LLC2 ソフトウェアと MAC ドライバー間のアダプター・インターフェースを越えて送信されるメッセージ

SDLC

メッセージは SDLC コンポーネントと SDLC ドライバー間で送信されます。内部トレースの生成の場合と同様に、このオプションを設定すると、SDLC 回線トレースで追加の詳細な情報も入手できます。

NLI

メッセージは QLLC コンポーネントと X.25 ドライバー間で送信されます。

HPRIP

Enterprise Extender (HPR/IP) のコンポーネントとノードの間で送信されるメッセージ

NODE

APPN プロトコル・コード内のコンポーネント間で送信されるメッセージ

SLIM

クライアント/サーバー・システムのコントローラー・サーバーとバックアップ・サーバーの間で送信されるメッセージ

DGRM

システム・コンポーネント間で送信される内部制御メッセージ

4. 構文を指定し終わったら、「Enter」を押します。

snaadmin set_trace_file コマンドと snaadmin set_trace_type コマンドについて詳しくは、「IBM Communications Server for Data Center Deployment on Linux 管理コマンド解説書」を参照してください。

内部トレース・ファイルの内容

内部トレースにより作成されるバイナリー・トレース・ファイルの出力をテキスト・ファイル形式に変換するには、`snatrcfmt` ユーティリティを使用します。出力をフィルター処理するには、`snafilter` ユーティリティを使用します。

これらのユーティリティの詳細については、[49 ページの『付録 A snafilter と snatrcfmt の使用方法』](#)を参照してください。

付録 A snafilter と snatrcfmt の使用方法

使用するトレースのタイプによっては、バイナリー出力が生成されます。CS Linux には、このようなバイナリー・ファイルのフィルター処理とフォーマットのためのツールがあります。この付録では、それらの操作方法について説明します。

- `snafilter` ユーティリティーを使用してバイナリー・トレース出力をフィルター処理し、必要な情報のみを抽出します。
- `snatrcfmt` ユーティリティーを使用して、バイナリー・トレース出力をテキスト・ファイルにフォーマットします。

バイナリー・トレースのフィルター処理

`snafilter` ユーティリティーでは、形式が設定されていない内部トレース・ファイルから特定のエントリーを選択できます。このため、特定の問題を診断する際に必要となる情報のみを抽出できます。たとえば、トレース・ファイルに多数のトレースのタイプ (CS Linux 内のさまざまなインターフェースでトレースされたメッセージ) が含まれている場合に、特定のタイプのメッセージのみを選択したり、特定のタイプのメッセージをすべて除去することができます。複数の APPC セッションまたは LUA セッションからのデータがトレース・ファイルに出力されている場合、特定の APPC アプリケーションまたはセッションに関連するメッセージを対象にしたりあるいは除外することができます。

トレース・ファイルに HPR 接続からの NLP フレーム・データまたは RTP フレーム・データが含まれている場合は、`snafilter` はこれらのトレース・タイプをフィルター処理しないことに注意してください。

注: `snafilter` は、内部トレース・ファイルだけでなく、回線トレース・ファイル、または両方のタイプのトレースを含む単一ファイルにも使用できます。

一部の `snafilter` オプションは、内部トレースにしか適用できません。回線トレースには適用されません。このことは各オプションの説明箇所でも示されています。回線トレースに言及していない場合は、そのオプションは両方のトレース・タイプに適用されます。

`snafilter` の出力はバイナリー・フォーマットのため、`snatrcfmt` による処理に適しています。

snafilter ユーティリティーの実行

トレース・フィルター・ユーティリティーを実行するためのコマンドの構文は、以下のとおりです。

```
snafilter [-f infilename] [-o outfilename] [options]
```

以下のオプションとパラメーターを指定します。

-f 非ファイル名

入力トレース・ファイル。If you do not use this option, スナフィルター uses `sna1.trc` as the default.

-o 出力ファイル名

出力トレース・ファイル。If you do not use this option, スナフィルター uses `snafil.trc` as the default.

+点 トレース・タイプ

指定したタイプまたはタイプのメッセージのみを含めます (指定できるメッセージ・タイプは、[46 ページの『内部トレース・ファイルの制御』](#)で説明されているメッセージ・タイプに対応しています)。Set トレース・タイプ to オール to turn on tracing of all types, or specify one or more of the following values. 2 つ以上のトレース・タイプを指定する場合は、コンマで値を区切り、コンマの前後にスペース文字を含めないでください。

- アプリケーション・アプリケーション
- FM

- ルア
- ニオブ
- ミズ
- ドウルツ
- LLC2
- マック
- ルリ
- シュルツ
- ヌリ
- ヘプリップ (エンタープライズ・エクステンダー・リンク用)
- NDLC とは (DLC メッセージへのノード)
- ノード
- スリム (クライアント/サーバー・システム内のコントローラー・サーバーとバックアップ・サーバーの間で送信されるメッセージ)
- DGRM (CS Linux 内部制御メッセージ)

Do not specify both +点 and - 点. どちらのオプションも指定しない場合、デフォルトは +ポイント ALL になります。

トレース・ファイルに DLC 回線トレースと内部トレースの両方が含まれている場合は、+ポイント DLC には、DLC 回線トレースのみが組み込まれます。を使用できます。

- 点トレース・タイプ

指定されたタイプまたはタイプのメッセージを除外します。トレース・タイプオプションは、+点は、- ポイント ALL が無効であることを除いてあります。の場合と同じです。

Do not specify both +点 and - 点. どちらのオプションも指定しない場合、デフォルトは +ポイント ALL になります。

トレース・ファイルに DLC 回線トレースと内部トレースの両方が含まれている場合は、-point DLC は、DLC 回線トレースを除外します。を使用できます。

+tpid XXXXXXXXXXXXXXXXX

指定されたトランザクション・プログラム (TP) ID (16 進数) を持つ APPC メッセージを組み込みます。他の APPC メッセージは除外します。このオプションは、APPC メッセージ以外のメッセージには影響しません。複数の TP ID を指定するには、それらをコンマで区切ります。

+tpid および - tpid。の両方を指定しない

このオプションは、回線トレースには影響しません。

- tpid XXXXXXXXXXXXXXXXX

指定された TP ID を持つ APPC メッセージを (16 進数で) 除外します。

+tpid および - tpid。の両方を指定しない

このオプションは、回線トレースには影響しません。

+convid XXXXXXXX 号

指定された会話 ID (16 進数) に APPC または CPI-C メッセージを組み込みます。その他の APPC または CPI-C メッセージは除外します。このオプションは、APPC メッセージまたは CPI-C メッセージ以外のメッセージには影響しません。複数の会話 ID を指定するには、それらをコンマで区切ります。

+convid および - コンビド。の両方を指定しない

このオプションは、回線トレースには影響しません。

- コンビド XXXXXXXX 号

指定された会話 ID (16 進数) を持つ APPC または CPI-C メッセージを除外します。

+convid および - コンビド。の両方を指定しない

このオプションは、回線トレースには影響しません。

+セシッド XXXXXXXX 号

指定されたセッション ID (16 進数) を持つ LUA メッセージを組み込み、他の LUA メッセージを除外します。このオプションは、LUA メッセージ以外のメッセージには影響しません。複数のセッション ID を指定するには、それらをコンマで区切ります。

SLI アプリケーションの場合、ライブラリーは、SLI verb をノードに送信する前に、対応する RUI verb に変換することに注意してください。これは、LUA の内部トレースには RUI verb のみが含まれることを意味します。API トレースを使用して、SLI verb に関する問題を診断します。

+セシッド および - セシド。の両方を指定しない

このオプションは、回線トレースには影響しません。

- セシド XXXXXXXX 号

指定されたセッション ID (16 進数) を持つ LUA メッセージを除外します。

+セシッド および - セシド。の両方を指定しない

このオプションは、回線トレースには影響しません。

+lfsid aabbc

指定されたローカル・フォーム・セッション ID (LFSID) を持つ DLC メッセージを組み込みます。その他の DLC メッセージは除外します。このオプションは、DLC メッセージ以外のメッセージには影響しません。

LFSID は、16 進数の 2 文字の OAF (*aa*)、16 進数の 2 文字の DAF (*bb*)、および 1 文字の ODAI (*c*) で構成されます。これは、ローカル・ノードからのメッセージ・フローで使用されるのと同じようにフォーマット設定されます。*aa* または *bb* の代わりにワイルドカード文字 *xx* を使用することができ、*c* の代わりに *x* を使用することができます。複数の LFSID を指定するには、それらをコンマで区切ります。

このオプションは、セッション上の両方向のメッセージを制御します。単一方向のみでメッセージをフィルタリングすることはできません。例えば、ノードからのメッセージ・フローが OAF = 01、DAF = 02、および ODAI = 1 の場合、同じセッション上のノードに流れるメッセージは、OAF = 02、DAF = 01、および ODAI = 1 になります。+lfsid 01021 は、このセッションの両方向に流れるメッセージを含みます。の指定

以下の両方のオプションを使用できます +lfsid および - ルフシド。

- ルフシド aabbc

指定された LFSID を持つ DLC メッセージを除外します。

You can use both of the options +lfsid and - ルフシド. 例えば、OAF 0x0b 指定されている場合を除き、すべてのメッセージを DAF 0x0a に組み込むには、+lfsid xx0ax - lfsid 0bxxx を指定します。

+npid XXXXXXXX 号

指定されたコンポーネント・インスタンス ID (16 進数) を持つノードおよび SDLC メッセージを組み込み、他のノードおよび SDLC メッセージを除外します。このオプションは、ノードおよび SDLC メッセージ以外のメッセージには影響しません。

コンポーネント・インスタンス ID は CS Linux 内部 ID であり、同じコンポーネントを使用する異なるユーザーまたはプログラムを区別します。複数のコンポーネント・インスタンス ID を指定するには、それらをコンマで区切ります。

+npid および - npid。の両方を指定しない

このオプションは、回線トレースには影響しません。

- npid XXXXXXXX 号

指定されたコンポーネント・インスタンス ID (16 進数) を持つノードおよび SDLC メッセージを除外します。

内部バイナリー・トレース出力をテキスト・ファイルにフォーマットする

+npid および - npid。の両方を指定しない

このオプションは、回線トレースには影響しません。

-開始 イムムド・シュムス

指定された日付(年、月、日)、および時刻(時、分、秒)の後にトレースされたメッセージのみを含めます。例えば、午後3時45分です。(15.45) 1997年8月11日に970811154500.

- 開始と エンド の両方を指定して、指定された時間の間にメッセージのみを含めることができます 指定する終了時刻は、開始時刻よりも後でなければなりません。

エンド・イムムド・シュムス

指定された日付および時刻の後にトレースされたメッセージを除外 - 開始. と同じフォーマットを使用します。

- 開始と エンド の両方を指定して、指定された時間の間にメッセージのみを含めることができます 指定する終了時刻は、開始時刻よりも後でなければなりません。

コマンドと出力の例

次のトレース・コマンドの例に、いくつかのコマンド・オプションを示します。

```
snafilter -f new.trc -o newout.trc +point APPC,NOF,DLC -lfsid 0a021 +convid 0100000a
```

このコマンドにより、次の操作が行われます。

- ファイル new.trc から入力を受け入れる。
- 出力をファイル newout.trc に送る。
- APPC メッセージ、NOF メッセージ、DLC メッセージを対象にする。
- LFSID が 0a021 の DLC メッセージを除外する。
- 会話 ID が 0100000a の APPC メッセージを対象にする。

したがって、出力ファイルには次のメッセージが書き込まれています。

- 指定した会話 ID をもつすべての APPC メッセージ
- すべての NOF メッセージ
- 指定した LFSID の DLC メッセージを除くすべての DLC メッセージ

snatrcfmt コマンドを使用して、フィルター処理済みの出力をフォーマットできます。このコマンドについては、次の節で説明します。

内部バイナリー・トレース出力をテキスト・ファイルにフォーマットする

snatrcfmt コマンド行ユーティリティでは、バイナリー・トレース出力を1つまたは2つのテキスト・ファイルにフォーマットできます。このユーティリティは、回線トレース、あるいは内部トレースのどちらか1つに使用できますが、各トレースでオプションが少し異なります。

- 回線トレースの場合、バイナリー・トレース出力を、1つのテキスト・ファイル、つまりメッセージ・データ・ダンプ・ファイルにのみフォーマットできます。
- 内部トレースの場合は、バイナリー・トレース出力を、メッセージ・データ・ダンプ・ファイル(58ページの『メッセージ・データ・ダンプ』を参照) またはメッセージ・フロー・チャート(55ページの『メッセージ・フロー・チャート』を参照)のいずれか、あるいはその両方にフォーマットできます。メッセージ・フロー・チャートは、コンポーネント間のメッセージ・フローを示す概略図です。
- バイナリー・トレース・ファイルが回線トレースと内部トレースの双方を含む場合、双方のトレース・タイプが出力ファイルに含まれます。

回線トレース用の snatrcfmt ユーティリティの実行方法

回線トレース用のトレース・フォーマット・ユーティリティの構文は次のとおりです。

```
snatrcfmt [-f filename] [-o output_file_base] [-S | -D] [-m] [-1] [-M]
```

次のオプションとパラメーターを指定します。

-f filename

このオプションでは、入力バイナリー・トレース・ファイルの名前を指定します。このオプションを指定しないと、snatrcfmt ではデフォルトとして sna1.trc が使用されます。

-o output_file_base

出力ファイルに使用される基本名です。CS Linux は、メッセージ・データ・ダンプ・ファイルには拡張子 .dmp を付けます。このオプションを指定しないと、CS Linux では、出力ファイルとしてファイル snatrc.dmp が使用されます。

-S

要約トレース・ファイルを作成します。このファイルには、メッセージごとにトレースが 1 行だけ含まれます。トレース・ファイルに詳細な SDLC トレース情報が含まれていると、このオプションにより情報フィールドは削除され、アドレスと制御フィールドのみ含められます。

-D

各メッセージの詳細なレポートを作成します。

-m

メッセージごとに、絶対時刻ではなく最後のメッセージ以後の時間間隔を表示します。このオプションを選択しないと、各メッセージには、それがファイルに書き込まれた日付と時刻が表示されます。

-1

トレース・ファイルに詳細な SDLC トレース情報が含まれていると、この情報は削除されます(ただし、標準の SDLC 回線トレースは含められます)。このオプションを選択しないと、トレース・ファイルのいずれの詳細な SDLC トレース情報も出力に含められます。

-M

詳細な SDLC トレース情報がファイルに含められる場合は、フレームは Modulo 128 形式にデコードされます。このオプションを選択しない場合は、詳細な SDLC トレース情報は Modulo 8 形式にデコードされます。

内部トレース用の snatrcfmt ユーティリティの実行方法

内部トレース用のトレース・フォーマット・ユーティリティの構文は次のとおりです。

```
snatrcfmt -i [-m] [-f filename] [-o output_file_base] [options]
```

次のオプションとパラメーターを指定します。

-i

このオプションは、snatrcfmt を内部トレースのフォーマットに使用することを示すのに使用します。

-m

メッセージごとに、絶対時刻ではなく最後のメッセージ以後の時間間隔を表示します。このオプションを選択しないと、各メッセージには、それがファイルに書き込まれた日付と時刻が表示されます。

-f filename

このオプションでは、入力バイナリー・トレース・ファイルの名前を指定します。このオプションを指定しないと、snatrcfmt ではデフォルトとして sna1.trc が使用されます。

-o output_file_base

出力ファイルに使用される基本名です。CS Linux は、メッセージ・フロー・ドローには拡張子 .drw を付け、メッセージ・データ・ダンプには拡張子 .dmp を付けます。このオプションを指定しないと、CS Linux では、出力ファイルとしてファイル snatrc.drw および snatrc.dmp が使用されます。

上の options で示されるオプションは次のとおりです。

-w

メッセージ・データ・ダンプ・ファイルのみを作成します。

以下に示したオプションは、メッセージ・データ・ダンプ・ファイルにしか使用できません。-w オプションを指定する場合は、以下のオプションを指定しないでください。

-b

あるコンポーネントから送信され、別のコンポーネントで受信されたメッセージのリストを対象にします。このオプションを指定しないと、CS Linux により、送信メッセージのみが対象にされます。

-r

各メッセージの 16 進ロウ・データのみを対象にします。メッセージ・データの解釈は行われません。

-d

メッセージを詳細形式で取り込みます。CS Linux では、データは 16 進値、EBCDIC 変換、ASCII 変換の 3 つの列でリスト表示されます (このため、メッセージ・データのテキスト・ストリングは、テキストの文字セットに応じて EBCDIC 列または ASCII 列に可読文字で表示されます)。さらに、CS Linux では、メッセージ・データの複数のフィールドが変換処理され、この変換処理されたものが可読テキストとして印刷されます。

-D

-d と同一、しかし各メッセージの RH および TH に詳細リストを表示します。

-r オプション、-d オプション、および -D オプションは複数指定しないでください。これらのオプションを全く指定しないと、CS Linux は 16 進値、EBCDIC、および ASCII のリストは取り込みますが、変換処理されるメッセージ・フィールドの数は制限されます。

-w

メッセージ・フロー・ドローのみが作成されます。

以下に示したオプションは、メッセージ・フロー・ドロー・ファイルにしか使用できません。-w オプションを指定する場合は、次のオプションは指定しないでください。

-c component_group

一連のコンポーネントを 1 列に圧縮します。これにより、圧縮されたコンポーネント間のメッセージがドローに示されなくなります。次の値のうち 1 つ以上を指定します。値と値の間はコンマで区切ります。

CP

制御点を 1 列に圧縮します。

LU

LU 6.2 コンポーネント (CPI-C および APPC) を 1 列に圧縮します。

OL

LU 0 から 3 コンポーネントを 1 列に圧縮します。

NODE

APPN プロトコル・コード内のすべてのコンポーネント (CP、LU、旧 LU (OL) グループなど) を 1 列に圧縮します。これにより、ノード間の内部メッセージが表示されなくなります。

NONE

圧縮しません。このオプションを指定する場合は、その他のコンポーネント・オプションは指定しないでください。

USER

すべての API コンポーネントを 1 列に圧縮します。

SIX

APPN プロトコル・コードの外のすべての内部コンポーネント (DLC を除く) を 1 列に圧縮します。

DLC

すべての DLC コンポーネントを 1 列に圧縮します。

このオプションを指定しないと、デフォルトにより `-c USER,NODE` が指定されます。`-c` オプションおよび `-s` オプションの両方を使用する場合は、必ず `-c` の後に `-s` を指定してください。

-s components

1つ以上の個別のコンポーネントに対応する列をドローに表示しません。指定されたコンポーネントからのメッセージ・フロー、あるいはこのコンポーネントへのメッセージ・フローは、「unknown」列として表示されます。このため、詳細情報を表示する場合に、あるグループ内のコンポーネントをすべて表示する必要がなくなります。例えば、PS コンポーネントおよび HS コンポーネントを表示し、RM コンポーネントを表示しないようにできます。

各コンポーネントを指定する場合は、55 ページの『メッセージ・フロー・ドロー』にリストされている 2 文字のコンポーネント ID を使用します。複数のコンポーネントを指定する場合は、コンポーネントとコンポーネントの間をコマンドで区切ってください。

注: コンポーネント・グループには `-c` オプションを指定し、各コンポーネントには `-s` オプションを指定してください。

`-c` オプションおよび `-s` オプションの両方を使用する場合は、必ず `-c` の後に `-s` を指定してください。

-p nn

メッセージ・フロー・ドローのページ長として `nn` 行を使用します (`nn` は 10 進数です)。CS Linux では、ページは改ページ文字で区切られ、各ページの先頭には列ヘッダーが組み込まれます。

`-p` オプションと `-P` オプションの両方を同時に指定しないでください。このオプションを指定しないと、デフォルトにより `-p 66` が指定されます。

-P

出力を 1 ページにまとめて作成します。(ページ区切りはありません。列ヘッダーはデータの先頭のみ出力されます。)

`-p` オプションと `-P` オプションの両方を同時に指定しないでください。このオプションを指定しないと、デフォルトにより `-p 66` が指定されます。

snatrcfmt ユーティリティの出力

snatrcfmt ユーティリティでは、次のいずれかのフォーマットまたは両方のフォーマットでテキスト出力が生成されます。

メッセージ・フロー・ドロー

異なる CS Linux コンポーネント間のメッセージ・フローを示す図 (この形式は回線トレースには適用されません)。

メッセージ・データ・ダンプ

各トレース・メッセージのデータのリスト。

メッセージ・フロー・ドロー

メッセージ・フロー・ドローの各列は、特定の CS Linux コンポーネントまたはコンポーネント・グループに対応しています。各列のヘッダーには、コンポーネントまたはコンポーネント・グループの名前を 1 文字もしくは 2 文字で表した省略語が表示されます。ファイルの各行は、CS Linux コンポーネント間の特定のメッセージ・フローに対応しています。

デフォルトのオプションでは、全 API コンポーネントと APPN プロトコル・コード内の全コンポーネントはそれぞれ 1 列ずつにグループ化されます。API コンポーネントとノードの間でやり取りされるメッセージと、ノードと DLC コンポーネントの間でやり取りされるメッセージのみが表示されます。ドローには、各 DLC コンポーネントを表す個別の列も示されます。必要であれば、これらのグループを個別のコンポーネントに分割して、より詳細なドローを示すこともできます。このような追加詳細ドローは、主に CS Linux サポート担当員用です。

注: 選択するオプションによって、出力の形式も異なります。状況によってサポート担当員が異なるオプションを使用することがあるため、サポート担当員には常に元のバイナリー・トレース・ファイルを提出してください。

トレースされるデータは 2 つの CS Linux コンポーネント間に流れるデータではなく、CS Linux とリモート・コンポーネント (ダイアグラムには示されません) の間に流れるデータであるため、メッセージ・フロー・ドローにはノード回線トレースは示されません。

メッセージ・フロー・ドローに表示されるコンポーネント・グループは次のとおりです。

US

次の API コンポーネントです。

AL

APPC ライブラリー

CL

CPI-C ライブラリー

CV

CSV ライブラリー

RL

RUI (LUA) ライブラリー

SLI アプリケーションの場合、ライブラリーは SLI verb をノードに送信する前に、対応する RUI verb に変換します。これは LUA の内部トレースは RUI verb のみを含むことを意味します。SLI verb のすべての問題を診断するには API トレースを使用してください。

ML

MS ライブラリー

NL

NOF ライブラリー

CD

構成デーモン

LD

ログ・デーモン

SD

SNA デーモン

RD

RCF デーモン

RS

リモート・システム (クライアント/サーバー・メッセージ)

SIX

次の APPN プロトコル・コードの外の内部コンポーネントです (DLC を除く)。

PM

パス管理プログラム

SV

サービス・マネージャー

AP、TP

APPC 内部コンポーネント

LO

ログ内部コンポーネント

M

MS 内部コンポーネント

N

NOF 内部コンポーネント

L1、L2

LUA 内部コンポーネント

FM

FM 内部コンポーネント

CP

制御点。次のコンポーネントからなります。

CM

セッション・コネクタ・マネージャー

NF

NOF ノード・コンポーネント

AM、AS

アドレス・スペース・マネージャー

CS

構成サービス

DS

ディレクトリー・サービス

MD

Management Services の複数ドメイン・サポート (MDS) コンポーネント

MS

Management Services

SS

セッション・サービス

TR

トポロジー・ルーティング・サービス

DR、ES

従属 LU リクエスター (DLUR)

LU

LU 6.2。次のコンポーネントからなります。

PS

表示サービス

HS

ハーフセッション

RM

リソース管理プログラム

SM

セッション・マネージャー

OL

旧 LU (LU 0 から 3)。次のコンポーネントからなります。

RU

RUI (LUA)

CH

標準的なハーフセッション

LM

LU 管理プログラム

PU

PU 管理プログラム

PX

SNA ゲートウェイ

NO

次のノード・コンポーネント (APPN プロトコル・コードのコンポーネント) です。

BM

バッファ管理プログラム

- D**
ノード内の DLC コンポーネント
- PC**
バス制御
- SC**
セッション・コネクタ
- HP、RT**
高性能ルーティング
- DL**
DLC コンポーネント。次のコンポーネントからなります。
 - L、LL、MT、M1**
LLC2 ドライバー
 - SL**
SDLC ドライバー
 - S2**
SDLC デバイス・ドライバー
 - HM**
SDLC ハードウェア・インターフェース
 - QL**
QLLC ドライバー
 - X2**
X.25 インターフェース (NLI)
 - IP**
Enterprise Extender (HPR/IP)

コンポーネントをグループ化するだけでなく、特定のコンポーネントまたはコンポーネント・グループの列の表示を抑制することもできます。このコンポーネントに流れ込む、あるいはそこから出ていくメッセージは、?? で示される「不明 (unknown)」列に流れ込む、またはそこから出ていくものとして表示されます。

ドロウ内の各行の最後には、\$ 文字で始まる番号とタイム・スタンプが出力されます。番号は、対応するメッセージ・データ・ダンプ・ファイルで該当メッセージが表示されている行番号を示します。タイム・スタンプは、メッセージの生成時刻を示します。

次の例に、メッセージ・フロー・ドロウの形式を示します。

```

Message Flow Drawing Example
File: sna1.trc      Page 16      Trace started: Tue Apr  4 10:56:41.250 GMT 2000
+-----+-----+-----+-----+-----+-----+-----+-----+
|US||CP||PS  HS  RM  SM||OL||SC  PC  D   BM||GG|
+-----+-----+-----+-----+-----+-----+-----+-----+
.  .ASSIGN_LFSID | . . . . . $013795 10:45:48.120
.  o<-----+ . . . . . $013795 10:45:48.120
.  |ASSIGN_LFSID_RSP . . . . . $013815 10:45:48.120
.  +----->o . . . . . $013815 10:45:48.120
.  . . . .CREATE | . . . . . $013835 10:45:48.120
.  . . . .o<-----+ . . . . . $013835 10:45:48.120
.  .MU(MU_BIND_RQ_SEND)RQD1. . . . . $013845 10:45:48.120
.  o<-----+ . . . . . $013845 10:45:48.120
.  |MU(MU_BIND_RQ_SEND)RQD1,PI . . . . $013900 10:45:48.310
.  +----->o . . . . . $013900 10:45:48.310
.  . . . . . |DLC_MU . . . . $014010 10:45:48.310
.  . . . . . +--->o . . . . $014010 10:45:48.310
.  . . . . . |DLC_MU . . . . $014065 10:45:48.310
.  . . . . . +----->o . . . . $014065 10:45:48.310
    
```

メッセージ・データ・ダンプ

メッセージ・データ・ダンプ・ファイルの先頭の数行に、稼働システムの識別とフィールド位置合わせに関する情報が記述されています。この情報を使用するのは、CS Linux サポート担当員のみです。

ファイルには、各メッセージの送信元、宛先、メッセージのタイプなどの見出し情報、16進値のメッセージ・データがこの順序で出力されます。メッセージ・データの詳細レベルとして次の3つのレベルから選択できます。

- 未変換 16 進データ
- 16 進データを EBCDIC または ASCII に解釈
- 16 進データを EBCDIC または ASCII に解釈、複数のメッセージ・フィールドをテキスト変換処理

メッセージ・データ・ダンプは回線トレースを含むトレース・ファイルからのもので、かつ、詳細な SDLC トレースを含んでいる場合は、以下の追加情報が SDLC フレーム用に含められます。

- 「TX」または「RX」。フレームが送信されているか、または CS Linux により受信されているかどうかを表します。
- アドレスおよび制御フィールド (および情報フィールドがあれば、このフィールドも対象にされる) のデコードされたバージョン。

次の例は、回線トレースのメッセージ・データ・ダンプのフォーマットを示したものです。

```

Message Data Dump Example for Line Tracing
-----
SND>> CNCT_OUT REQ                               16:54:33.950 BST   20 Apr 1999
                                                ETHER0.ETSAP0.ETHL0
-----
SND>> CNCT_OUT REQ                               16:54:33.950 BST   20 Apr 1999
                                                TOKEN0.TRSAP0.TRL0
-----
SND>> CNCT_OUT REQ                               16:54:33.950 BST   20 Apr 1999
<:54:33.950 BST   20 Apr 1999
SND>> XID (NULL)                                  ETHER0.ETSAP0.ETHL0
-----
SND>> XID (NULL)                                  16:54:33.960 BST   20 Apr 1999
<:54:33.960 BST   20 Apr 1999
SND>> XID (NULL)                                  TOKEN0.TRSAP0.TRL0
-----
SND>> XID (NULL)                                  16:54:33.960 BST   20 Apr 1999
<(NULL)                                           ETHER0.ETSAP0.ETHL0
-----
SND>> XID FMT:3 ID:01100002 ESI:PRE_NEG LR:SEC   ETHER0.ETSAP0.ETHL0
XID 32540110 00020000 000AD100 00000000 .....J.....2T.....
00010B41 00040900 00000007 000E0AF4 .....4...A.....
      C1D7D7D5 4BD4D6D6 D5102900 28110C08 APPN.MOON.....K.....).....
      04F0F6F0 F0F0F609 06E2D5C1 E2E3C1D9 .060006..SNASTAR .....
03084011 0FE2D5C1 E2E3C1D9 40D3C9D4 .. ..SNASTAR LIM ..@.....@...
      C9E3C5C4                                ITED .....
-----
SND>> XID (NULL)                                  16:54:33.970 BST   20 Apr 1999
<(NULL)                                           TOKEN0.TRSAP0.TRL0

```

```

Message Data Dump Example for Line Tracing with Maximum Detail
-----
SND>> CNCT_OUT REQ                               16:54:33.950 BST   20 Apr 1999
IPS: 00000000 4554484C 30202020 010000EE .....<.....ETHL0 ....
      01000008 00000007 000629EA BC670400 .....g..
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 040900FF 8100009C .....@a.....
      FFFFFFFF 00020040 00060000 00000000 @@@@.....@.....
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 00000000 00000000 .....
      0000000A 0005000A 00051388 13880064 .....h.h.....d
      000A001E 001E0003 .....
-----
SND>> CNCT_OUT REQ                               16:54:33.950 BST   20 Apr 1999
IPS: 00000000 54524C30 20202020 010000F0 .....<.....TRL0 ....
      50182F10 00000007 08005AFD 90B30400 &.....!.....P./.....Z.....
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 1009000D 810000A5 .....a.v .....
      000E0007 00020040 00050000 00000000 .....@.....
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 00000000 00000000 .....
      0000000A 0005000A 00051388 13880064 .....h.h.....d
      000A001E 001E0003 .....
-----
<: 00000001 4554484C 30202020 010000EE .....<.....ETHL0 ....
      410000EF 00000007 000629EA BC670400 .....A.....)g..
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 040900FF 0100011E .....@.....
      FFFFFFFF 00020040 00060000 00000000 @@@@.....@.....
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 00000000 00000000 .....
      0000000A 0005000A 00051388 13880064 .....h.h.....d
      000A001E 001E0003 .....

```

内部バイナリー・トレース出力をテキスト・ファイルにフォーマットする

```

----- 16:54:33.950 BST 20 Apr 1999
SND>> XID (NULL)                ETHER0.ETSAP0.ETHL0
IPS: 410000EF 00020000          .....A.....
----- 16:54:33.960 BST 20 Apr 1999
<: 00000001 54524C30 20202020 010000F0 .....<.....0 ....TRL0 ....
      410000F1 00000007 08005AFD 90B30400 ...1.....!.....A.....Z.....
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 1009000D 0100012D .....-
      000E0007 00020040 00050000 00000000 .....@.....
      00000000 00000000 00000000 00000000 .....
      00000000 00000000 00000000 00000000 .....
      0000000A 0005000A 00051388 13880064 .....h.h.....d
      000A001E 001E0003 .....
----- 16:54:33.960 BST 20 Apr 1999
SND>> XID (NULL)                TOKEN0.TRSAP0.TRL0
IPS: 410000F1 00020019          ...1....A.....
----- 16:54:33.960 BST 20 Apr 1999
<(NULL)                          ETHER0.ETSAP0.ETHL0
IPS: 010000EE 80370070          .....7.p
----- 16:54:33.960 BST 20 Apr 1999
SND>> XID FMT:3 ID:01100002 ESI:PRE_NEG LR:SEC ETHER0.ETSAP0.ETHL0
IPS: 410000EF 00025400          .....A.....T.
XID decode:

XID format type           = 03
Node type                 = 02
XID length                = 54
block/ID number          = 01100002
                          . . @ .
                          . . . .
Bytes 8-9                 = 000A
                          Init self may be sent to the XID sender
                          XID sender supports independent-PLU session partners
                          This node can generate BIND PIU segments
                          This node can receive BIND PIU segments
                          ACTPU for an SSCP-PU session requested
                          The XID sender is not a network node
                          CP services not requested or supported
                          CP-CP sessions not supported on this TG
                          Secondary initiated non-activation exchange supported
                          XID sender does not supported CP name change
                          Prenegotiation exchange
Byte 10                   = D1
                          Adaptive BIND pacing as a BIND sender supported
                          Adaptive BIND pacing as a BIND receiver supported
                          This TG is operative
                          XID sender supports receipt of ACTPU containing PU cap cv
                          XID sender is not a peripheral border node
                          Adaptive pacing on both, negotiable
Byte 15                   = 00
                          XID sender does NOT support parallel TGs
TG number                 = 00
DLC type                  = 01
DLC type is SDLC         = 00
DLC data length          = 0B
Byte 19                   = 41
                          XID sender can be an ABM combined station
                          XID sender not already using short-hold mode
                          Short-hold mode not supported
                          Sender is secondary link station (non-negotiatiable)
                          Link-station transmit-receive capability: two-way simultaneous
Byte 20                   = 00
                          XID sender is not the sender of a nonactivation XID
Maximum BTU length        = 409
Byte 23                   = 00
                          SNA link profile
Byte 24                   = 00
                          SIM and RIM not supported
I-frame number           = 07
XID 32540110 00020000 000AD100 00000000 .....J.....2T.....
      00010B41 00040900 00000007 000E0AF4 .....4...A.....
      C1D7D7D5 4BD4D6D6 D5102900 28110C08 APPN.MOON.....K.....)(...
      04F0F6F0 F0F0F609 06E2D5C1 E2E3C1D9 .060006..SNASTAR .....`..
      03084011 0FE2D5C1 E2E3C1D9 40D3C9D4 .. ..SNASTAR LIM ..@.....@...
      C9E3C5C4 .....ITED.....
----- 16:54:33.970 BST 20 Apr 1999
<(NULL)                          TOKEN0.TRSAP0.TRL0
IPS: 010000F0 5A000000          ...0!....Z...

```

付録 B getsense の使用法

SNA ネットワーク障害は、アプリケーション・プログラムに戻されるセンス・コードで示されます。SNA センス・コードは、内部サービス・ログに 8 桁の 16 進値 (4 バイト) で表示されます。

- 最初の 2 桁は障害カテゴリーを示します。
- 次の 2 桁は障害カテゴリー修飾子を示します。
- 最後の 4 桁は障害サブカテゴリーを示します。障害サブカテゴリーには、障害の性質に関する詳細かつ具体的な情報が示されます。

CS Linux コンピューター上で生成された特定の SNA センス・コードに関する情報を取得するには、コマンド行で `sna -getsense` と入力し、続けてカテゴリーと修飾子 (先頭の 4 桁)、またはセンス・コード全体 (全 8 桁) を入力します。

たとえば、センス・コード 08170001 の情報を取得するには、次のように入力します。

```
sna -getsense 08170001
```

コマンドの `sna -getsense` は短縮形で `sna -g` を使用できます。

コマンドからの出力は次のようになります。

```
# sna -getsense 08170001
REQUEST REJECT (CATEGORY CODE = X'08')
```

```
-----
This category indicates that the request was delivered to
the intended component and was understood and supported,
but not executed.
```

```
0817      Link or Link Resource Inactive: A request
          requires the use of a link or link resource
          that is not active.
```

```
0001      Link inactive.
```

`sna -getsense` コマンドは、指定されたセンス・コードを認識できない場合でも、障害カテゴリーおよび障害カテゴリー修飾子情報 (先頭の 4 桁) を検索しようとします。`sna -getsense` コマンドを使用してもこの情報を検索できない場合には、「*IBM Systems Network Architecture: Formats*」の資料を参照してください。

SNA のセンス・コードがリモート・コンピューター上で生成されたものである場合は、そのリモート・コンピューター上で `sna -getsense` と同等のものを使用して、その意味を判別する必要があります。

付録 C snagetpd の使用法

サポート担当員が問題を診断できるように、ファイルをサポート担当員へ提出しなければならない場合があります。診断情報収集ユーティリティ `snagetpd` は、サポート担当員に必要な情報を簡単に単一のファイルに収集するコマンド行管理プログラムです。

`snagetpd` の収集内容は以下のとおりです。

- ログ・ユーティリティおよびトレース・ユーティリティの現在の設定に関する情報。監査ログが活動状態であったかどうか、およびログ・ファイルのサイズなどがこれに該当します。
- ログ・ファイルとトレース・ファイル
- コア・ファイル (必要な場合)
- 診断ファイル・ディレクトリー `/var/opt/ibm/sna` の内容
- AIX または Linux におけるリモート API クライアントの場合: クライアント・ネットワーク・データ・ファイルの `sna_clnt.net`、およびクライアントが現在接続されているサーバーの詳細を記録する `server.current` ファイル

この付録では、`snagetpd` の使用法について説明します。

snagetpd の実行方法

CS Linux の稼働中に発生した問題を解決できない場合には、サポート担当員から `snagetpd` ユーティリティを実行し、診断データを含む出力ファイルを送信するように求められることがあります。

`snagetpd` ユーティリティは `root` ログインから実行する必要があります。このユーティリティにより、CS Linux トレース・ユーティリティおよびログ・ユーティリティの出力ファイルを含む複数のファイルを圧縮した `tar` ファイルが作成されます。

このユーティリティがファイルの収集を開始する前に `snagetpd` は、管理ツールによって収集されるあらゆるタイプの CS Linux のトレース (回線トレース、クライアント/サーバー・トレース、TN Server トレース、内部トレースなど) を非活動化します。このユーティリティはログの設定またはユーザー・スペースの API トレースの設定 (環境変数で制御されます) は変更しません。

プログラム実行時に、`snagetpd` は「問題の症状を説明してください (Please describe the symptoms of the problem)」というメッセージを表示して、問題を記述するようプロンプトを出します。問題についての記述を入力し終わったら、`CTRL+D` を押すか、または `$` を入力し、`Enter` を押してプログラムを続行します。

プログラムが終了したら、出力ファイル (63 ページの『コマンド構文とプログラム出力』を参照してください) をサポート担当員に提出して診断してください。

注:

1. `snagetpd` ユーティリティは `tar` ファイル内の一部の診断ファイルの名前を変更するので、その内容が元のファイル名に一致しないことがあります。このユーティリティはシステム上の元の診断ファイルの変更や、名前変更は行ないません。
2. Linux クライアント・コンピューターで問題が発生する場合は、クライアント・コンピューターおよびそれが使用しているリソースを含むサーバー上で、`snagetpd` を実行し、可能な限り多くの問題判別情報を得ることを確認してください。

コマンド構文とプログラム出力

スナゲト pd は以下のとおりです。 のコマンド構文

スナゲト pd [- q][ファイル名]

以下のオプションとパラメーターを指定します。

- q

静止モードを指定する。このオプションを指定すると、スナゲト pd は情報を求めるプロンプトを出さずに実行されます。ログ・ファイルおよびトレース・ファイルを収集するほか、ログ・ユーティリティーおよびトレース・ユーティリティーの現在の設定に関する情報に加えて、スナゲト pd はローカル・ディレクトリーにある コア* というコア・ファイルを収集します。

- q、スナゲト pd は、問題の説明を入力し、収集するコア・ファイルのパスを求めるプロンプトを出し、関連する実行可能ファイルを収集するように通知します。を指定しない場合

ファイル名

問題判別情報が入っている出力ファイルの名前を指定します。この出力ファイルは、ファイル名パラメーターを使用するときにはパスを指定しない限り、現行ディレクトリーに配置されます。

ファイル名を指定すると、スナゲト pd は、ファイル名.tar.gz の出力が出力されます。ファイル名を指定しない場合、CS Linux はデフォルトのファイル名 pd.tar.gz. を割り当てます。

pd サブディレクトリーに、コマンド行に次のように入力します。スナップショット .tar.gz という名前の出力ファイルを配置する

```
スナゲト pd pd/snaperr
```

コマンドの制限

スナゲト pd: には以下の制約事項が適用

- スナゲト pd は、システムから必要なすべての情報を収集するために、根 ユーザー ID から実行する必要があります。
- 出力ファイルの内容全体を保管するのに十分なディスク・スペースがないマシン上でスナゲト pd を実行すると、スナゲト pd はエラー・メッセージ ディスク・スペース不足を表示します。これが発生した場合、出力は、ディスク・スペースを使い切れないうちに収集されたファイルで構成される tar ファイルです。The file is named ファイル名 0.tar ファイル。
- 中央ロギングがアクティブであり、中央ロガーとして機能しているノード以外のノードでスナゲト pd を実行する場合、中央ロガーで収集されたロギング情報は、出力ファイルには含まれません。ただし、スナゲト pd を実行するノードがアクセスできる情報はすべて、出力ファイルに含まれています。
- CS Linux が稼働していないときスナゲト pd 出力ファイルに以下のデータが含まれていません。を実行する場合
 - 監査ロギングがアクティブであったかどうか、およびログ・ファイルのサイズなど、スナゲト pd が実行されました。の場合のログおよびトレース・ユーティリティーの設定に関する情報
 - デフォルト・ディレクトリー内にないログ・ファイルおよびトレース・ファイルは、デフォルトのファイル名になります。(例えば、以下のようにスナゲト pd は、/var/opt/ibm/sna サブディレクトリー内のトレース情報を検索して、名前の付いたファイルを *.trc.)

付録 D Windows クライアント

この付録では、Windows クライアントに固有のロギング情報およびトレース情報を説明します。

Windows クライアントのロギング

Windows クライアントのロギングは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」で説明されているように、レジストリーのオプションで制御されます。問題メッセージおよび例外メッセージはエラー・ログ・ファイルに記録され、監査メッセージは監査ログ・ファイルに記録されます。

問題メッセージは常にログに記録され、このログ記録を使用不可にすることはできません。ただし、例外メッセージと監査メッセージについてはログに記録するかどうかを指定できます。これらのオプションのいずれかが明示的に指定されない場合、サーバーで `snaadmin set_global_log_type` コマンドを発行してください。

中央ロギングが (サーバーで `snaadmin set_central_logging` コマンドを発行することにより) 使用可能である場合、クライアントからのすべてのメッセージは中央ログ・ファイルに書き込まれます。そうでなければ、次のように指定します。

- Windows クライアント にエラー・メッセージおよび監査メッセージを保持するためのローカル・ファイル
- ログ情報をバックアップするために使用されるファイル
- ログ・ファイルをバックアップし、リセットするためのサイズ
- 詳細ロギングまたは簡潔ロギングのいずれを使用するか

ログ・ファイルは、Windows ログ・ファイルの場合のデフォルト最大サイズが 10,000 バイトであるというものを除けば (Linux では 1,000,000 バイトではない)、Linux コンピューターと同様にバックアップおよびリセットされます。

注：「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」で説明されているクライアント構成ユーティリティーは、クライアント上のエラー・メッセージまたは監査メッセージ (あるいはその両方) をログに記録するかどうかを制御する簡単な方法を提供します。ロギングをオンまたはオフにするだけでよい場合は、このユーティリティーを使用してそれを実行できます。例えば、ログ・ファイル名、またはファイルをバックアップおよびリセットするためのサイズを指定する場合など、より詳細にロギングを制御する必要がある場合のみ、レジストリーの使用が必要になります。

Windows クライアントのトレースの制御

Windows クライアントでは、CS Linux は API トレースおよびクライアント/サーバー・トレースの機能を提供します。レジストリーのオプションにより、これらのすべてのトレース・タイプが制御されます。詳しくは、「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」を参照してください。

注：「*IBM Communications Server for Data Center Deployment on Linux 管理ガイド*」で説明されているクライアント構成ユーティリティーは、クライアント上のトレースを制御する簡単な方法を提供します。API のトレースおよびクライアント/サーバーのトレースの両方を、オンまたはオフにするだけでよい場合は、このユーティリティーを使用してそれを実行できます。例えば、トレース・ファイル名を指定する場合や、特定の API に対してのみトレースをオンにする場合など、より詳細にトレースを制御する必要がある場合のみ、レジストリーの使用が必要になります。

これらのトレース・タイプの各フォーマットは、Linux コンピューターのコンポーネントと同様です (詳細は、[43 ページの『クライアント/サーバー・トレース』](#)、[39 ページの『API トレースのトレース・ファイルの形式』](#)を参照してください)。

CS Linux には、5250 エミュレーション・プログラム用、または Windows クライアントでの HLLAPI アプリケーション用のトレース機能はありません。しかし、5250 データは APPC を使用して転送されるので、

5250 エミュレーション・プログラムからノードに送信されるデータをトレースする場合は、クライアントで APPC API トレースを使用できます (詳しくは、「*IBM Communications Server for Data Center Deployment on Linux* 管理ガイド」を参照してください)。追加のトレース機能は、5250 エミュレーション・ソフトウェアまたは 3270 エミュレーション・ソフトウェアで提供されます。詳細は、プログラムに添付される資料を参照してください。

Windows クライアントの診断情報の収集

サポート担当員が問題を診断できるように、診断ファイルをサポート担当員へ提出しなければならない場合もあります。診断情報収集ユーティリティ `snagetpd` は、サポート担当員に必要な情報を簡単に単一のファイルに収集するコマンド行管理プログラムです。

Windows クライアントで `snagetpd` を実行するには、コマンド・ウィンドウで次のコマンドを入力するか、「Start/Run (開始/実行)」アイコンから行います。

```
snagetpd
```

`snagetpd` ユーティリティは Windows トレース・ファイルおよびその他の Windows クライアント・システム情報のすべてを一緒に収集し、これを `snapd.exe` という名前の単一自己解凍型 ZIP ファイルに集めます。この自己解凍型 ZIP ファイルをサポート担当員に送付します。

付録 E IBM へのご意見の送付方法

本資料に関するご意見をお待ちしております。情報の明確性、正確性、および完全性に関するご意見およびその他の情報をご自由にお寄せください。

以下のいずれかの方法でご意見をお送りください。

1. Knowledge Center の下部にあるフィードバック・リンクを使用してください。
2. 以下のフィードバック・テンプレートを 사용하여 "mhvrcfs@us.ibm.com" に E メールを送信してください。
3. 次の住所にご意見をお送りください。

IBM Corporation
Attention: MHVRCFS Reader's Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
US

E メール・フィードバック・テンプレート

以下のテンプレートをメールにカット・アンド・ペーストしてください。その後、必要な情報を入力します。

- お名前:
- 企業、大学、機関名:
- ご意見の対象となるトピックまたは Web ページの URL:
- ご意見の本文

ご意見を口頭でいただける場合は、お客様の電話番号とご都合の良い時間も記入していただけます。

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、適切な方法で、使用もしくは配布することができるものとします。

IBM または他の組織は、お客様からいただいた個人情報、問題解決のためのお客様への連絡以外の目的で使用することはありません。

技術的な問題がある場合

リストされたご意見の送付方法は使用しないでください。代わりに、以下のいずれかのアクションを実行してください。

- IBM サービス担当員にお問い合わせください。
- IBM 技術サポートに連絡してください。
- IBM サポート・ポータル (<https://www.ibm.com/support/home/>) にアクセスしてください。

付録 F 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す。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。

- 〒 106-8711
- 東京都港区六本木 3-2-12
- 日本アイ・ビー・エム株式会社
- 法務・知的財産
- 知的財産権ライセンス 渉外

- 〒 106-8711
- 東京都港区六本木 3-2-12
- 日本アイ・ビー・エム株式会社
- 法務・知的財産
- 知的財産権ライセンス 渉外

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すが、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。

著作権使用許諾: 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。© (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, 2007, 2008, 2021. All rights reserved.

商標

以下は、世界の多くの国で登録された International Business Machines Corporation の商標です。

- ACF/VTAM
- Advanced Peer-to-Peer Networking
- AIX
- Application System/400
- APPN
- AS/400
- CICS
- DATABASE 2
- DB2
- Enterprise System/3090
- Enterprise System/4381
- Enterprise System/9000
- ES/3090
- ES/9000
- eServer
- IBM
- IBMLink
- IMS
- MVS
- MVS/ESA
- Operating System/2
- Operating System/400
- OS/2
- OS/400
- PowerPC
- PowerPC Architecture
- S/390
- System/390
- System p5
- System z
- System z9
- VSE/ESA
- VTAM
- WebSphere

以下は、各々の会社の商標または登録商標です。

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。

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。

Intel および EM64T は、Intel Corporation または子会社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。

AMD64 は Advanced Micro Devices, Inc の米国およびその他の国における商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Microsoft、Windows、Windows 10、Windows Server 2012、Windows Server 2016、Windows Server 2019 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。

参考文献

以下の IBM 資料では、本書で説明しているトピックについての情報を記載しています。資料は、以下のトピック別に大きく分けてあります。

- CS Linux バージョン 7.1
- システム・ネットワーク体系 (SNA)
- ホスト構成
- z/OS Communications Server
- Transmission Control Protocol/Internet Protocol (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・トピック

CS 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、タイトルおよび資料番号のみをここに記しています。

CS Linux バージョン 7.1 の資料

CS 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 Data Center Deployment on Linux 入門*」を参照してください。これらのソフトコピー・ブックをシステムにインストールするには、9～15MB のハード・ディスク・スペースが必要になります (このスペースは、インストールする各国語バージョンによって異なります)。

- *IBM Communications Server for Data Center Deployment on Linux 入門* (GC88-9996 および GC88-9997)
この資料は CS 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。この資料には、以下の 2 つのバージョンがあります。
 - IBM CS Linux 入門 (GC88-9996)
 - IBM CS Linux for IBM Z 入門 (GC88-9997)
- *IBM Communications Server for Data Center Deployment on Linux 管理ガイド* (SC88-9999)
この資料には、SNA および CS Linux の概要、および CS 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 Data Center Deployment on Linux 管理コマンド解説書* (SC88-9998)
この資料には、SNA および CS 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 Data Center Deployment on AIX or Linux CPI-C プログラマーズ・ガイド* (SC88-5826)
この資料では、「C」または Java の経験があるプログラマーを対象として、CS Linux CPI 通信 API を使用する SNA トランザクション・プログラムの作成に関する情報を提供します。
- *IBM Communications Server for Data Center Deployment on AIX or Linux APPC プログラマーズ・ガイド* (SC88-5825)
この資料では、拡張プログラム間通信機能 (APPC) を使用する アプリケーション・プログラムを作成するために必要な情報を記載しています。
- *IBM Communications Server for Data Center Deployment on AIX or Linux LUA プログラマーズ・ガイド* (SC88-5827)
この資料では、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 (LUA) を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。

- *IBM Communications Server for Data Center Deployment on AIX or Linux 共通サービス Verb プログラマーズ・ガイド (SC88-5824)*
この資料では、Common Service Verb (CSV) アプリケーション・プログラミング・インターフェース (API) を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。
- *IBM Communications Server for Data Center Deployment on AIX or Linux MS プログラマーズ・ガイド (SC88-5829)*
この資料では、Management Services (MS) API を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。
- *IBM Communications Server for Data Center Deployment on Linux NOF プログラマーズ・ガイド (SC88-8591)*
この資料では、Node Operator Facility (NOF) API を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。
- *IBM Communications Server for Data Center Deployment on Linux 診断ガイド (GC88-8601)*
この資料では、SNA ネットワークの問題解決について説明しています。
- *IBM Communications Server for Data Center Deployment on AIX or Linux APPC アプリケーション・スイート ユーザーズ・ガイド (SC88-5828)*
この資料には、CS Linux で使用される 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 用語集 (GC88-8602)*
この資料は、CS Linux ライブラリーで頻繁に使用される用語とその定義を包括的に収録しています。

システム・ネットワーク体系 (SNA) 関連資料

以下の資料では、SNA ネットワークについての情報を記載しています。

- *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)*

ホスト構成関連資料

以下の資料では、ホスト構成についての情報を記載しています。

- *ES-9000, ES-3090 IOCP ユーザーズ・ガイド・ボリューム A04 (GC38-0097)*
- *3174 Establishment Controller インストール・ガイド (GG24-3061)*
- *3270 Information Display System 3174 Establishment Controller: 計画ガイド (GA27-3918)*
- *OS/390 ハードウェア構成定義 (HCD) ユーザーズ・ガイド (SC88-6630)*

z/OS Communications Server 関連資料

以下の資料には、z/OS Communications Server に関する情報について記載されています。

- *z/OS V1R7 Communications Server: SNA ネットワーク・インプリメンテーション・ガイド* (SC88-8928)
- *z/OS V1R7 Communications Server: SNA 診断* (Vol 1: GC31-6850、Vol 2: GC31-6851)
- *z/OS V1R6 Communications Server: リソース定義解説書* (SC88-8929)

TCP/IP 関連資料

以下の資料には、Transmission Control Protocol/Internet Protocol (TCP/IP) ネットワーク・プロトコルに関する情報について記載されています。

- *z/OS V1R7 Communications Server: IP 構成ガイド* (SC88-8926)
- *z/OS V1R7 Communications Server: IP 構成解説書* (SC88-8927)
- *z/VM V5R1 TCP/IP 計画およびカスタマイズ* (SD88-6453)

X.25 関連資料

以下の資料には、X.25 ネットワーク・プロトコルに関する情報について記載されています。

- *Communications Server for OS/2 Version 4 X.25 Programming* (SC31-8150)

APPC 関連資料

以下の資料では、拡張プログラム間通信機能 (APPC) についての情報を記載しています。

- *APPC Application Suite V1 ユーザーズ・ガイド* (SC31-6532)
- *APPC Application Suite V1 管理* (SC31-6533)
- *APPC Application Suite V1 プログラミング* (SC31-6534)
- *APPC Application Suite V1 オンライン製品ライブラリー* (SK2T-2680)
- *APPC Application Suite ライセンス・プログラムの資料* (GC31-6535)
- *z/OS V1R2.0 Communications Server: APPC Application Suite User's Guide* (SC31-8809)

プログラミング関連資料

次の資料では、プログラミングについての情報を記載しています。

- *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) 解説書* (SC26-4399)
- *Communications Server for OS/2 Version 4 Application Programming Guide* (SC31-8152)

その他の IBM ネットワーキング・トピック

次の資料では、CS Linux に関連する その他のトピックについての情報を記載しています。

- *SDLC Concepts* (GA27-3093)
- *Local Area Network Concepts and Products: LAN アーキテクチャー* (SG24-4753)
- *Local Area Network Concepts and Products: LAN アダプター、ハブおよび ATM* (SG24-4754)
- *Local Area Network Concepts and Products: ルーターおよびゲートウェイ* (SG24-4755)
- *Local Area Network Concepts and Products: LAN オペレーティング・システムおよび管理* (SG24-4756)
- *IBM ネットワーク制御プログラム リソース定義ガイド* (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]

- アラート [2, 6](#)
- イーサネット・リンク・ステーションを始動できない場合 [13](#)
- 一般的な問題の解決
 - TN Redirector [20](#)
 - MS アプリケーションが作動しない [19](#)
 - NOF アプリケーションが作動しない [19](#)
 - SNA ゲートウェイ・セッションの問題 [21](#)
 - サーバー管理問題 [21](#)
 - 最初に確認する事項 [9](#)
 - 初期ステップ [9](#)
 - その他の SNA ノードとの通信の確認 [10](#)
 - ネットワーク・ノードの経路が指定されない [21](#)
 - ノードが活動状態であることの確認 [10](#)
 - APPC アプリケーションが作動しない [17](#)
 - CPI-C アプリケーションが作動しない [17](#)
 - LUA アプリケーションが作動しない [18](#)
 - SNA デーモン開始の確認 [9](#)
 - TN Server [19](#)
- イベント
 - 監査 [3](#)
 - 定義 [1](#)
 - 問題 [3](#)
 - 例外 [3](#)
- オペレーティング・システムの戻りコード [29, 30](#)
- オンライン・ヘルプ
 - コマンド行管理プログラム [6](#)
 - 使用可能なタイプ [6](#)
 - 使用方法のヘルプ [7](#)
 - man ページ [7](#)
 - Motif 管理プログラム [6](#)

[カ行]

- 回線トレース
 - 概要 [4](#)
 - 制御 [35](#)
 - バイナリー・トレース・ファイルを [36](#)
- 環境変数
 - DISPLAY [17](#)
 - SNACTL [38](#)
 - SNATRC [33, 37](#)
- 監査イベント [3](#)
- 簡略ロギング [28](#)
- 技術的な問題
 - 解決方法 [67](#)
- 基本的な確認事項 [9](#)
- 共通の問題、解決 [9](#)
- クライアント/サーバー・トレース [5](#)
- クライアント/サーバー・トレース
 - 収集 [43](#)
 - トレース・ファイルの内容 [44](#)

- クライアント/サーバー問題 [14](#)
- クライアントの問題
 - Windows [15](#)
- 構成ファイル [16](#)
- コマンド
 - ps [24](#)
 - sna -getsense [61](#)
 - snaapitrdfmt [42](#)
 - snafilter [49](#)
 - snagetpd [63](#)
 - snahelp [28](#)
 - snatrcfmt [49, 53](#)
 - snawhat [23](#)

[サ行]

- サポート担当員
 - ログ/トレース・ファイルの提出 [23](#)
 - システム構成情報の提出 [23](#)
 - 情報の提供 [22](#)
 - 診断情報収集ユーティリティ [23](#)
 - ソフトウェア・バージョンの提出 [23](#)
 - タイプ [22](#)
 - プロセス情報の提出 [24](#)
 - 問題の報告 [22](#)
- システム構成情報
 - サポートのためのファイル [16](#)
 - サポートへ送付 [23](#)
- 詳細ロギング [28](#)
- 詳細ロギング・メッセージの形式 [28](#)
- 使用量ログ・ファイル [32](#)
- 資料に関するコメント
 - フィードバックの送信 [67](#)
- 診断情報、タイプ [1](#)
- 指揮
 - スナゲット pd [22](#)
 - スナット cfmt [4, 36](#)
 - スナファイター [4](#)
- センス・コード、getsense を使用した情報の取得 [61](#)

[タ行]

- 中央ロギング [26](#)
- デーモン、SNA、開始 [9](#)
- トークンリング・リンク・ステーションを始動できない場合 [13](#)
- 読者のコメント
 - フィードバックの送信方法 [67](#)
- トレース
 - アプリケーション内からの API トレースの制御 [38](#)
 - アプリケーション・プログラム [37, 38](#)
 - 回線トレース [4](#)
 - 回線トレースの実行 [34](#)
 - 概要 [2](#)
 - クライアント/サーバー・トレース [5](#)
 - クライアント/サーバー・トレース [43](#)
 - 使用 [25, 33](#)

トレース (続き)

- トレース・タイプ [4](#)
- トレースの使用法 [5](#)
- 内部トレース [5](#), [46](#)
- バイナリー・トレース・ファイルを [36](#)
- API トレース [5](#)
- API トレースの収集 [37](#)
- DLC トレース [4](#)
- Motif 管理プログラムを使用 [33](#)
- TN サーバーのトレース [5](#)

トレース機能 [33](#)

トレース出力

- 形式設定、snaapitrcfmt [41](#)
- フォーマット、snatrcfmt [52](#)
- 例 [39](#)

トレース出力のフォーマット [41](#), [52](#)

トレース・ファイル

- サイズ [39](#)
- フォーマット [39](#)

トレース・フォーマット・ユーティリティ

- 回線トレースのコマンド形式 [53](#)
- 出力オプション [54](#)
- 内部トレースのコマンド形式 [53](#)
- メッセージ・データ・ダンプ・オプション [54](#)
- メッセージ・フロー・ドロウ・オプション [54](#)
- API トレースのコマンド形式 [42](#)

[ナ行]

内部トレース

- トレース・ファイルの制御 [46](#)
- トレース・ファイルの内容 [48](#)
- フィルター処理 [49](#)

内部トレース・データのフィルター処理 [49](#)

内部トレースのフィルター処理 [49](#)

ネットワーク・ノード・セッションのルーティング問題の解決 [21](#)

ノードが非アクティブ [10](#)

[ハ行]

フィードバック

- 読者のコメントの送信 [67](#)
- Eメール・テンプレート [67](#)

プログラム・エラー・メッセージ [1](#), [2](#), [23](#)

プロセス ID [39](#)

ポートを始動できない [11](#)

[マ行]

メッセージ・データ・ダンプ

- 説明 [58](#)
- 例 [59](#)

「メッセージに対する処置 (Message action)」フィールド [4](#)

メッセージ・フロー・ドロウ

- 説明 [55](#)
- 例 [58](#)

戻りコード、オペレーティング・システム [29](#), [30](#)

問題イベント [3](#)

問題解決の概要 [1](#)

問題の解決

- TN Redirector [20](#)
- MS アプリケーションが作動しない [19](#)

問題の解決 (続き)

- NOF アプリケーションが作動しない [19](#)
- SNA ゲートウェイ・セッションの問題 [21](#)
- サーバー管理問題 [21](#)
- 最初に確認する事項 [9](#)
- その他の SNA ノードとの通信の確認 [10](#)
- ネットワーク・ノードの経路が指定されない [21](#)
- ノードが活動状態であることの確認 [10](#)
- APPC アプリケーションが作動しない [17](#)
- CPI-C アプリケーションが作動しない [17](#)
- LUA アプリケーションが作動しない [18](#)
- Motif が機能しない [17](#)
- SNA デーモン開始の確認 [9](#)
- TN Server [19](#)

問題の報告 [22](#)

[ヤ行]

ユーティリティ

- スナゲット pd [22](#)
- スナット cfmt [4](#), [36](#)
- スナフィーター [4](#)

ユーティリティ

- sna -getsense [61](#)
- snaapitrcfmt [42](#)
- snafilter [49](#)
- snagetpd [63](#)
- snahelp [28](#)
- snatrcfmt [49](#), [53](#)
- snawhat [23](#)

[ラ行]

リンク・ステーションを始動できない

- イーサネット [13](#)
- エンタープライズ・エクステンダー [14](#)
- すべてのリンク・タイプに対するアクション [12](#)
- トークンリング [13](#)
- IP [14](#)
- MPC [14](#)
- QLLC [13](#)
- SDLC [12](#)

例外イベント [3](#)

ローカル・ロギング [26](#)

ロギング

- 簡略ロギング [27](#), [28](#)
- コマンド行管理プログラム [26](#)
- 使用 [25](#)
- 詳細ロギング [27](#), [28](#)
- 使用量ログ・ファイル [32](#)
- 中央 [26](#)
- ライセンス・マネージャーのトラッキング [33](#)
- ローカル [26](#)
- ログ情報のタイプ [3](#)
- ログの使用法 [3](#)
- ログ・ファイルの名前と位置の変更 [26](#)
- Motif 管理プログラム [25](#)

ログの形式 [27](#)

ログ・ファイル

- サイズの制御 [27](#)
- バックアップ [27](#)
- リセット [27](#)
- ログ・メッセージ

ログ・メッセージ (続き)
概要 [2](#)
原因と処置の情報 [30](#)
推奨処置 [4](#)
ログに記録する [27](#)

A

API トレース
収集 [37](#)
設定 [37](#)
トレース・ファイルの形式 [39](#)
の使用 [5](#)
API トレース・ファイルの一部の例 [41](#)
API トレース・フォーマット・ユーティリティ [42](#)
APPC アプリケーションの問題 [17](#)

C

CPI-C アプリケーションの問題 [17](#)

D

DISPLAY 環境変数 [17](#)
DLC トレース [4](#)
DLC を始動できない [11](#)
DLC/ポートを始動できない [11](#)

E

Enterprise Extender のリンク・ステーションを始動できない [14](#)

G

getsense、使用法 [61](#)

I

IP リンク・ステーションを始動できない [14](#)

L

LS が活動状態であることの確認 [10](#)
LUA アプリケーションの問題 [18](#)

M

Motif 管理プログラム、ログ記録の制御のための使用 [25](#)
Motif の問題 [17](#)
MPC リンク・ステーションを始動できない [14](#)
MS アプリケーションの問題 [19](#)

N

NOF アプリケーションの問題 [19](#)

P

PDF ブック、表示 [6](#)
ps ユーティリティ [24](#)

Q

QLLC リンク・ステーションを始動できない [13](#)

S

SDLC リンク・ステーションを始動できない [12](#)
SNA ゲートウェイ・セッションの問題 [21](#)
SNA デーモン、開始 [9](#)
SNACTL 環境変数 [38](#)
snafilter ユーティリティ [49](#)
snagetpd ユーティリティ
概要 [63](#)
コマンド構文 [63](#)
コマンドの制限 [64](#)
実行方法 [63](#)
プログラム出力 [63](#)
Windows [66](#)
snahelp ユーティリティ [30](#)
SNATRC 環境変数 [33](#), [37](#)
snatrc.dmp ファイル [53](#)
snatrc.drw ファイル [53](#)
snatrcfmt ユーティリティ
回線トレースのコマンド形式 [53](#)
出力オプション [54](#)
内部トレースのコマンド形式 [53](#)
メッセージ・データ・ダンプ・オプション [54](#)
メッセージ・フロー・ドロー・オプション [54](#)
SNATRUNC [38](#)
snawhat ユーティリティ [23](#)

T

TN Server [19](#)
TN Server トレース
概要 [45](#)
収集 [45](#)
トレース・ファイルの内容 [45](#)
TN サーバーのトレース [5](#)
TZ エントリー、Windows レジストリー [28](#)

W

Windows クライアント
トレース [65](#)
問題 [15](#)
ロギング [65](#)
Windows クライアントの確認 [15](#)

X

xsnadmin プログラム [25](#)

[特殊文字]

TN Redirector [20](#)
snaapitrcfmt ユーティリティ
回線トレースのコマンド形式 [42](#)
概要 [41](#)
出力形式のサンプル [42](#)



GC88-8601-05

