

# IBM Sterling Connect:Direct for UNIX

管理ガイド

バージョン 4.1



本書は、新しい版で明記されていない限り、IBM® Sterling Connect:Direct® for UNIX バージョン 4.1、また、それ以降のすべてのリリースおよびモディフィケーションに適用されます。

本書および本書で紹介する製品をご使用になる前に、98ページの『特記事項』に記載されている情報をお読みください。

Licensed Materials - Property of IBM

IBM® Sterling Connect:Direct® for UNIX

© Copyright IBM Corp. 1999, 2011. All Rights Reserved.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP

Schedule Contract with IBM Corp.

## 第 1 章 Sterling Connect:Direct for UNIX 7

サーバー・コンポーネント .....	7
Process マネージャー .....	7
コマンド・マネージャー .....	8
セッション・マネージャー .....	8
Sterling Connect:Direct for UNIX の概念 .....	8
ユーザー認証 .....	9
Process の再始動 .....	9
アーカイブ統計ファイル .....	11
Process、シェル・スクリプト、プログラムのサンプル .....	11
Sterling Connect:Direct for UNIXファイル .....	13
Sterling Connect:Direct for UNIX の構成ファイル .....	13
Sterling Connect:Direct for UNIX のディレクトリー構成 .....	14
Sterling Connect:Direct for UNIX 文書 .....	16
本書について .....	16
タスクの概要 .....	16

## 第 2 章 構成ファイルの保守 18

構成ファイルについて .....	18
構成ファイルの修正 .....	19

## 第 3 章 初期化パラメーター・ファイルの保守 20

初期化パラメーター・ファイル .....	20
各種パラメーターの更新 .....	22
パス・レコードの更新 .....	22
SNODE 作業パスの更新 .....	23
ノード名レコードの更新 .....	23
PAM サービス・レコードの更新 .....	23
停止/再開レコードの更新 .....	24
優先順位レコードの更新 .....	24
実行ディレクトリー内の特殊文字の使用制限 .....	24
リモート・ノード接続レコードの更新 .....	25

伝送制御キュー (TCQ) レコードの更新 .....	26
Sterling Connect:Direct Secure Plus レコード (Secure+ レコード) の追加と更新 .....	26
グローバルなコピー (copy) レコードの更新 .....	27
グローバルなタスク実行 (run task) レコードの更新 .....	30
統計ファイル情報レコードの更新 .....	31
サーバー認証レコードの更新 .....	32
ユーザー出口レコードの更新 .....	34
ファイアウォール・ナビゲーション・レコードの更新 .....	35
<b>第 4 章 クライアント構成ファイルの保守</b> .....	<b>37</b>
クライアント構成ファイル .....	37
API 構成レコードの更新 .....	38
CLI 構成レコードの更新 .....	38
クライアント認証レコードの更新 .....	39
<b>第 5 章 ネットワーク・マップファイルの保守</b> .....	<b>41</b>
ネットワーク・マップファイル .....	41
リモート・ノードのネットワーク・マップ・エントリーのサンプル .....	42
ローカル・ノード接続レコードの更新 .....	42
ローカル・ノードの TCP/IP 設定の更新 .....	49
リモート・ノード接続情報の更新 .....	51
<b>第 6 章 アクセス情報ファイルの保守</b> .....	<b>56</b>
ユーザー認証情報ファイル .....	56
ユーザー認証ファイルのサンプル .....	57
ローカル・ユーザー情報レコード・フォーマットの更新 .....	58
リモート・ユーザー情報レコードの更新 .....	62
強固なアクセス制御ファイルの更新 .....	65
シャドー・パスワードの自動検出 .....	65
プログラム・ディレクトリーへのアクセス制限 .....	66
セキュリティ出口 .....	66
<b>第 7 章 クライアントおよびサーバー認証キー・ファイルの保守</b> .....	<b>68</b>
クライアントおよびサーバー認証キー・ファイル .....	68
キー・ファイルのフォーマット .....	68
キー・ファイル・パラメーターの更新 .....	69
クライアント認証キー・ファイルのサンプル .....	69
認証手順 .....	70
サーバー認証パラメーター .....	71
クライアント認証パラメーター .....	71

<b>付録 A ファイアウォール・ナビゲーションの構成</b>	<b>72</b>
ファイアウォール・ナビゲーションの実装 .....	72
ファイアウォール・ナビゲーション .....	72
TCP ファイアウォール・ナビゲーション・ルール .....	73
UDT ファイアウォール・ナビゲーション・ルール .....	73
ファイアウォールの構成例 .....	74
TCP ファイアウォールの構成例 .....	74
UDT ファイアウォールの構成例 .....	74
アウトバウンド・パケットのブロック .....	75
セッションの確立 .....	75
TCP セッションの確立 .....	75
UDT セッションの確立 .....	76
<b>付録 B IP アドレス、ホスト名およびポートの指定</b>	<b>77</b>
IP アドレス .....	77
IPv4 アドレス .....	77
IPv6 アドレス .....	77
ホスト名 .....	78
ポート番号 .....	79
複数のアドレス、ホスト名、およびポート .....	79
IP アドレス範囲へのマスクの使用 .....	79
<b>付録 C テスト・モードでのSterling Connect:Direct の使用</b>	<b>81</b>
テスト・モードの処理フロー .....	81
NDMPXTBL パラメーター・テーブルの準備 .....	82
テスト・シナリオのサンプル .....	84
<b>用語集</b>	<b>86</b>
<b>索引</b>	<b>92</b>
<b>特記事項</b>	<b>98</b>
商標 .....	100



---

# Sterling Connect:Direct for UNIX

IBM® Sterling Connect:Direct® for UNIX は、複数の技術を関連付けて、ネットワーク化されたシステムおよびコンピューター間において、あらゆるタイプの情報を移動します。また、自動化、信頼性、効率的なリソース使用、アプリケーション統合、使いやすさといった特性を提供することにより、ハイパフォーマンスな転送を管理します。Sterling Connect: Direct for UNIX は、通信プロトコル、ハードウェア・プラットフォーム、およびオペレーティング・システムにおいて複数の選択肢を提供するとともに、メインフレーム・システム、ミッドレンジ・システム、デスクトップ・システム、および LAN ベースのワークステーション間での情報の移動を可能とする柔軟性も備えています。

Sterling Connect:Direct for UNIX は、クライアント/サーバー・アーキテクチャーに基づいています。Sterling Connect:Direct for UNIX のサーバー・コンポーネントは、ユーザー・インターフェース (API、CLI、IBM® Sterling Connect:Direct® Browser User Interface、IBM® Sterling Control Center) と通信することで、Sterling Connect:Direct for UNIX のステートメントおよびコマンドの実行依頼、実行、監視することができるようになります。

---

## サーバー・コンポーネント

Sterling Connect:Direct for UNIX には、以下のサーバー・コンポーネントが搭載されています。

### Process マネージャー

Process マネージャー (PMGR) は、Sterling Connect:Direct for UNIX サーバー環境を初期化するデーモンです。PMGR は、以下の機能を提供します。

- ◆ Sterling Connect:Direct for UNIX の初期化
- ◆ Sterling Connect:Direct for UNIX のクライアント API およびリモート・ノードからの接続要求の受け入れ
- ◆ API およびリモート・ノードと通信するための、コマンド・マネージャーおよびセッション・マネージャーの子 Process の作成
- ◆ 一元化された Sterling Connect:Direct for UNIX 機能が必要とされる場合の、コマンド・マネージャーおよびセッション・マネージャーからの要求の受け入れ
- ◆ Sterling Connect:Direct for UNIX の停止

---

注: PMGR に接続可能なすべてのコンピュータにおいて、エンド・ユーザー・アプリケーション (End User Application: EUA) を含むすべてのアプリケーションを実行することができます。

---

## コマンド・マネージャー

正常に確立された API 接続のすべてに対して、コマンド・マネージャー (CMGR) が作成されます。PMGR が作成できるコマンド・マネージャーの数は、システムによって異なり、各 UNIX Process で使用可能なファイル記述子の数によって制限されます。UNIX オペレーティング・システムによってセットアップされるファイル記述子の数は、Sterling Connect:Direct for UNIX の運用に影響する可能性があります。許可される同時 Sterling Connect:Direct for UNIX セッションの数 (最大 999 個) を処理するのに十分な数のファイル記述子を定義する必要があります。

CMGR は、以下の機能を提供します。

- ◆ API によって送信されるコマンドを実行し、その結果を API に対して送信
- ◆ API と連動して、Sterling Connect:Direct for UNIX の認証手順を実行し、Sterling Connect:Direct for UNIX に対するアクセスを判断
- ◆ コマンド実行時の PMGR との通信

## セッション・マネージャー

セッション・マネージャー (SMGR) は、十分なリソースが利用可能であり、Process の実行準備ができていて、またはリモート・ノードがローカル・ノードとの接続を要求する場合に、PMGR によって作成および呼び出しされます。SMGR は、以下の機能を提供します。

- ◆ 必要な Sterling Connect:Direct for UNIX 作業の実施
- ◆ 1 次ノード (PNODE) として機能し、Process の実行を開始
- ◆ 2 次ノード (SNODE) として機能し、PNODE によって開始された Process に参加

ノードに実行依頼される Process を実行するために SMGR が作成されると、リモート・ノードへの接続が作成されます。ローカル Process を実行するために PMGR によって SMGR が開始されると、SMGR は全 Process が完了するまで、このセッション上で各 Process を実行します。

リモート・ノードが接続を開始したことによって SMGR が作成される場合、SMGR が接続を完了させます。リモート Process を実行するために、PMGR によって SMGR が開始される場合、SMGR は、リモート SMGR が保持する Process をすべて完了させるまで、リモート SMGR によって提供されるリモート Process ステップを実行します。

SMGR は、伝送制御キュー (TCQ) サービスおよびその他の一元化されたサービスの PMGR に依存します。TCQ の概要については、15 ページの『伝送制御キュー (TCQ)』を参照してください。

---

## Sterling Connect:Direct for UNIX の概念

本セクションでは、Sterling Connect:Direct for UNIX システム・オペレーションの理解に役立つ、概念および定義についてご紹介します。Sterling Connect:Direct の一般的な概念については、「IBM Sterling Connect:Direct for UNIX 製品概要 (IBM Sterling Connect:Direct for UNIX Product Overview)」を参照してください。



## ユーザー認証

Sterling Connect:Direct for UNIX では、Sterling Connect:Direct for UNIX の特定のタスクの実行において、ローカルおよびリモートのユーザーを認証することができます。Sterling Connect:Direct for UNIX を使用するにあたって、各ユーザーは `userfile.cfg` と呼ばれるユーザー認証ファイルで定義されたレコードを持つ必要があります。各ローカル・ユーザーは、ユーザー認証ファイル内にレコードを持つ必要があり、リモート・ユーザーは、プロキシ関係でローカル・ユーザー ID にマッピングされる可能性があります。

一般ユーザーが、Sterling Connect:Direct for UNIX を通して root アクセス権を得ることを防止する対策として、Sterling Connect:Direct for UNIX インストール時に、強固なアクセス制御 (SACL) ファイルと呼ばれる 2 つ目のアクセス・ファイルが作成されます。このファイルは `sysacl.cfg` と命名されます。`sysacl.cfg` ファイルで指定される **root:deny.access** パラメーターは、Sterling Connect:Direct for UNIX に対するアクセスを許可、拒否、または制限します。SACL ファイルが削除されている、または破損している場合には、全ユーザーに対する Sterling Connect:Direct for UNIX へのアクセスが拒否されます。

`userfile.cfg` および `sysacl.cfg` ファイルでのユーザー認証の指定に関する詳細については、『Sterling Connect:Direct for UNIX 管理ガイド (Sterling Connect:Direct for UNIX Administration Guide)』の『アクセス情報ファイルの保守 (Maintaining Access Information Files)』を参照してください。

## Process の再始動

システム障害後の Process リカバリー用に複数の機能が提供されます。Process リカバリーは、システム障害発生後、できるだけ早急の実行を再開して、無駄なデータ送信を最小限に抑えることを目的としています。Process リカバリーを可能とするために、以下の Sterling Connect:Direct for UNIX 機能を利用することができます。

- ◆ **Process ステップ・リスタート:** Process の実行に伴い、ステップが TCQ に記録されます。何らかの理由によって Process が停止されると、その Process は TCQ 内で保留されます。実行を継続するために Process が解放されると、その Process は停止されたステップから自動的に開始します。
- ◆ **自動セッション再試行:** ネットワーク・マップ・ファイルのリモート・ノード情報レコード内に `short-term` および `long-term` の 2 つの接続再試行パラメーター・セットが定義されます。これらのパラメーター値を、リモート・ノード情報レコード内で指定しない場合は、ネットワーク・マップ・ファイルの `local.node` エントリーからデフォルト値が使用されます。`short-term` パラメーターは、即時の再試行を可能とします。`long-term` パラメーターは、すべての `short-term` の再試行が実施された後に使用されます。`long-term` の試行では、接続の問題を迅速に修正することができず、再試行を実施するまでにより長い時間を要することが想定されるため、接続再試行のオーバーヘッドが保存されます。
- ◆ **チェックポイント・リスタート:** この機能は、**copy** ステートメントによって使用できます。チェックポイント・リスタートは、**ckpt** パラメーターを通して、**copy** ステップ内で明示的に構成することができます。次の URL から IBM Sterling Connect:Direct for UNIX Process の Web サイトを参照してください:  
<http://www.sterlingcommerce.com/documentation/processes/processhome.html>  
 チェックポイント・リスタートが **copy** ステップで構成されない場合は、**ckpt.interval** パラメーターを使用して、`Initparms` で構成することができます。(このパラメーターに関する詳細

については、「*Sterling Connect:Direct for UNIX 管理ガイド (Sterling Connect:Direct for UNIX Administration Guide)*」の『*初期化パラメーター・ファイルの保守 (Maintaining Configuration Files)*』を参照してください。

- ◆ Run Task リスタート: SNODE 上の run task ステップの実行時に Process が中断された場合、Sterling Connect:Direct for UNIX は SNODE の前の run task ステップと現行の run task ステップとの同期化を図ります。同期化は、以下のいずれかの方法で実施されます。
  - Process の再始動時に、SNODE がタスクを実行中の場合には、そのタスクの完了を待ってから、PNODE に対して task completion ステータスを返します。処理が継続されます。
  - Process が再始動される前に SNODE のタスクが完了している場合には、タスク結果を保存します。Process の再始動時に、SNODE は結果を報告し、処理が継続されます。

同期化に失敗した場合、Sterling Connect:Direct for UNIX は、**run task** ステップ内の **restart** パラメーターまたは初期化パラメーター・ファイルを読み込み、**run task** ステップを再度実施するかどうかを決定します。**run task** ステップの **restart** パラメーターは、初期化パラメーターの設定をオーバーライドします。

例えば、SNODE が Sterling Connect:Direct for UNIX のコールド・リスタートによって、run task ステップの結果を失った場合、Sterling Connect:Direct for UNIX は、**restart** パラメーター内で定義されている値を確認して、**run task** を再度実行するかどうかを決定します。

---

注: run task リスタートは、Sterling Connect:Direct for UNIX が接続ロード・バランサーの裏で稼働している場合には、異なった機能を行います。ロード・バランシング環境で Sterling Connect:Direct for UNIX を稼働する場合に知っておく必要のある考慮事項についての詳細な情報については、「*IBM Sterling Connect:Direct for UNIX リリース・ノート (IBM Sterling Connect:Direct for UNIX Release Notes)*」、「*IBM Sterling Connect:Direct for UNIX 管理ガイド (IBM Sterling Connect:Direct for UNIX Administration Guide)*」、および「*Sterling Connect:Direct for UNIX スタートアップ・ガイド (Sterling Connect:Direct for UNIX Getting Started Guide)*」を参照してください。

---

- ◆ SNODE が Sterling Connect:Direct for UNIX ノードの場合の Process アクティビティーの中断: SNODE が Sterling Connect:Direct for UNIX ノードであり、PNODE が、Process アクティビティーの停止コマンドを実行したり、実行している Process を削除したりすることによってプロセス・アクティビティーを中断させた場合、または、リンクが失敗したり、転送時に I/O エラーが発生したりした場合には、その Process は、WS ステータスで待機キューに設定されます。

Process アクティビティーが継続されない場合、その Process を TCQ から手動で削除する必要があります。**delete process** コマンドにおけるコマンド構文およびパラメーター記述については、「*Sterling Connect:Direct for UNIX ユーザー・ガイド (Sterling Connect:Direct for UNIX User Guide)*」を参照してください。

---

注: SNODE から **change process** コマンドを実行して、Process アクティビティーを継続させることはできません。Process は、常にセッションの制御を行っている PNODE によってのみ再始動させることができます。

---

## アーカイブ統計ファイル

Sterling Connect:Direct for UNIX は、統計ファイルをアーカイブして削除するユーティリティを提供します。Sterling Connect:Direct for UNIX を構成する際に、初期化パラメーター・ファイルの統計レコードに **max.age** のパラメーターを設定することによって、統計ファイルをアーカイブするタイミングを特定します。**max.age** パラメーターは、統計ファイルをアーカイブの対象する際の要件となる、ファイルの古さを定義します。

毎日 1 回、statarch.sh と呼ばれるスクリプトが開始されます。このスクリプトは、**max.age** に一致する統計ファイルを特定した後、tar コマンドおよび compress コマンドを実行して、**max.age** パラメーターに一致する統計レコードのすべてをアーカイブした圧縮ファイルを作成します。統計ファイルは、アーカイブ後、パーズされます。Linux コンピューター上でアーカイブされるファイルに関しては、アーカイブされる統計ファイルは gzip 形式で圧縮されるため、.gz の接尾辞が付加されます。その他すべての UNIX プラットフォーム上でアーカイブされるファイルは、compress 形式を使用して圧縮されていることを示す .Z の接尾辞が付加されます。

アーカイブ・ファイルは、統計ファイルおよび TCQ が格納されているディレクトリーに保存されます。statarch.sh のシェル・スクリプトは、ndm/bin ディレクトリーに置かれています。必要に応じて、スクリプトを修正し、使用する環境用にカスタマイズしてください。

アーカイブされた統計ファイルをリストアしたい場合には、**statrestore.sh** スクリプトを実行します。このスクリプトは、**uncompress** および **tar** コマンドを使用して、アーカイブ内のすべての統計ファイルをリストアします。この際、**statrestore** コマンドに対して 2 つの引数を提供する必要があります。1 つ目の引数は統計ファイルが格納されているディレクトリーのパスになります。2 つ目の引数は、リストアしたいだけの数のアーカイブ・ファイル名が続く、アーカイブ・ファイル名を特定します。以下は、**statrestore** コマンドのサンプルです。

```
qa160sol: ./statrestore.sh/export/home/users/cd4000/ndm/bin archive1
```

ファイルのリストア後、select statistics コマンドを使用することにより、統計レコードを確認することができます。

## Process、シェル・スクリプト、および API プログラムのサンプル

Sterling Connect:Direct for UNIX は、*d\_dir*/ndm/src 内で Process およびシェル・スクリプトのサンプルを提供します。*d\_dir* は、Sterling Connect:Direct for UNIX ソフトウェアの宛先ディレクトリーを示しています。テキスト・エディターで同様のファイルを作成することができます。また、Process およびシェル・スクリプトのサンプルの作成手順に関しては、同じディレクトリー内の README ファイルに記載されています。

以下の表は、サンプル Process のファイル名を記載しています。必要に応じて Process を修正してください。

ファイル名	Process タイプ
cpunx.cd	copy (コピー)
rtunx.cd	run task (タスクの実行)

ファイル名	Process タイプ
rjunx.cd	run job (ジョブの実行)
sbunx.cd	submit (実行依頼)

以下の表は、サンプル・シェル・スクリプトのファイル名を記載しています。必要に応じて、シェル・スクリプトを修正してください。

ファイル名	シェル・スクリプト・タイプ
selstat.sh	統計の選択
send.sh	送信
recv.sh	受信
wildcard	複数ファイルの PDS への送信
statarch.sh	統計ファイルのアーカイブ
statrestore.sh	アーカイブされた統計ファイルのリストア
lcu.sh	ローカル接続ユーティリティ (LCU) ツールの起動
spadmin.sh	Secure+ 管理ツールの起動
spcli.sh	Secure+ CLI (SPCLI) の起動
spcust_sample1.sh	STS プロトコル用の Secure+ の構成
spcust_sample2.sh	STS プロトコル用の Secure+ の構成
spcust_sample3.sh	SSL または TLS プロトコルを使用するための Secure+ の構成

以下の表は、サンプル・プログラムの名前を記載しています。

プログラム名	説明
apicheck.c	このプログラムは、ファイルをリモート・システムにコピーする Process の実行依頼を行います。この例では、MAXDELAY が使用されます。これは、プログラムが、ファイルの転送が終わるまで実行を終了しないことを意味します。このモジュールのコンパイルには、標準の c コンパイラーが使用されます。
apicheck.C	<i>「Sterling Connect:Direct for UNIX ユーザー・ガイド (Sterling Connect:Direct for UNIX User Guide)」に記載される C++ コンパイラーの 1 つを使用してコンパイルされることを除き、apicheck.c と同じ。</i>
exit_skeleton.c	このプログラムは、Sterling Connect:Direct for UNIX と連携して機能するユーザー出口プログラムのスケルトンです。これは、全 3 種のユーザー出口の使用について示します。
exit_skeleton.C	<i>「Sterling Connect:Direct for UNIX ユーザー・ガイド (Sterling Connect:Direct for UNIX User Guide)」に記載される C++ コンパイラーの 1 つを使用してコン</i>

プログラム名	説明
exit_sample.c	これは、セキュリティー出口がパスチケット機能に近いコードを使用して示されることを除き、スケルトン・ユーザー出口プログラムと同じプログラムです。
sdksample.C	このプログラムは、Sterling Connect:Direct for UNIX に対する SDK インターフェイスを使用して、さまざまなコマンドを実行します。

## Sterling Connect:Direct for UNIX ファイル

このセクションでは、Sterling Connect:Direct for UNIX の構成ファイルを説明するとともに、ディレクトリー構成を例示し、インストールされる個別ファイルの一覧を記載します。

### Sterling Connect:Direct for UNIX の構成ファイル

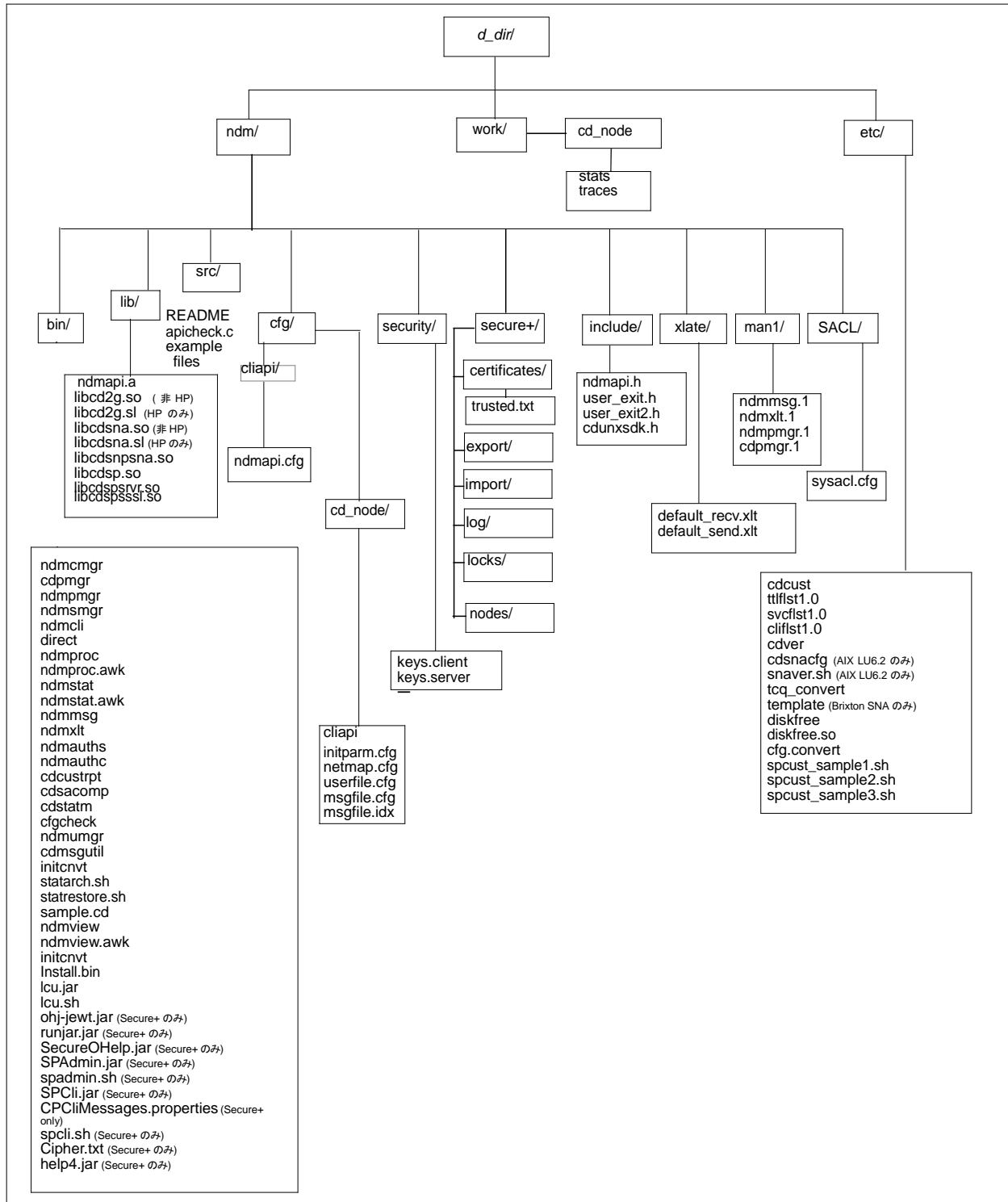
Sterling Connect:Direct for UNIX は、インストールおよびカスタマイズの実施時に、以下の構成ファイルを作成します。これらのファイルは、Sterling Connect:Direct for UNIX サーバーが正確に稼働するために必要となります。

構成ファイル	説明
初期化パラメーター・ファイル	起動時に使用する情報をサーバーに提供します。インストール時に、初期化パラメーター・ファイルに必要な設定を特定します。
ユーザー認証情報ファイル	ローカル・ユーザー情報およびリモート・ユーザー情報のレコード・タイプを格納しています。インストール時にこのファイルをカスタマイズして、リモート・ユーザー ID をローカル・ユーザー ID に対してマッピングし、ユーザー認証情報ファイルにリモート・ユーザー情報レコードを作成します。
強固なアクセス制御ファイル	Sterling Connect:Direct for UNIX のセキュリティーを向上させて、root アクセス権における許可、拒否、または制限を行います。このファイルは、Sterling Connect:Direct for UNIX のインストール時に作成されます。このファイルが削除されている、または破損している場合には、全ユーザーに対する Sterling Connect:Direct for UNIX へのアクセスが拒否されます。
ネットワーク・マップ・ファイル	ローカル・ノードおよびネットワーク内のその他の Sterling Connect:Direct for UNIX について記載します。Sterling Connect:Direct for UNIX が通信する各ノードのリモート・ノード・レコードを定義することができます。
サーバー認証キー・ファイル	クライアントの API 接続要求の妥当性を検証します。妥当性が確認されたクライアントのみ、接続が許可されます。
クライアント構成ファイル	クライアントが Sterling Connect:Direct for UNIX への接続に使用するポートおよびホスト名を特定します。
クライアント認証キー・ファイル	Sterling Connect:Direct for UNIX クライアントが接続する先の Sterling Connect:Direct for UNIX サーバーを特定します。複数のサーバー・エントリーを保持することが可能です。

## Sterling Connect:Direct for UNIX のディレクトリー構成

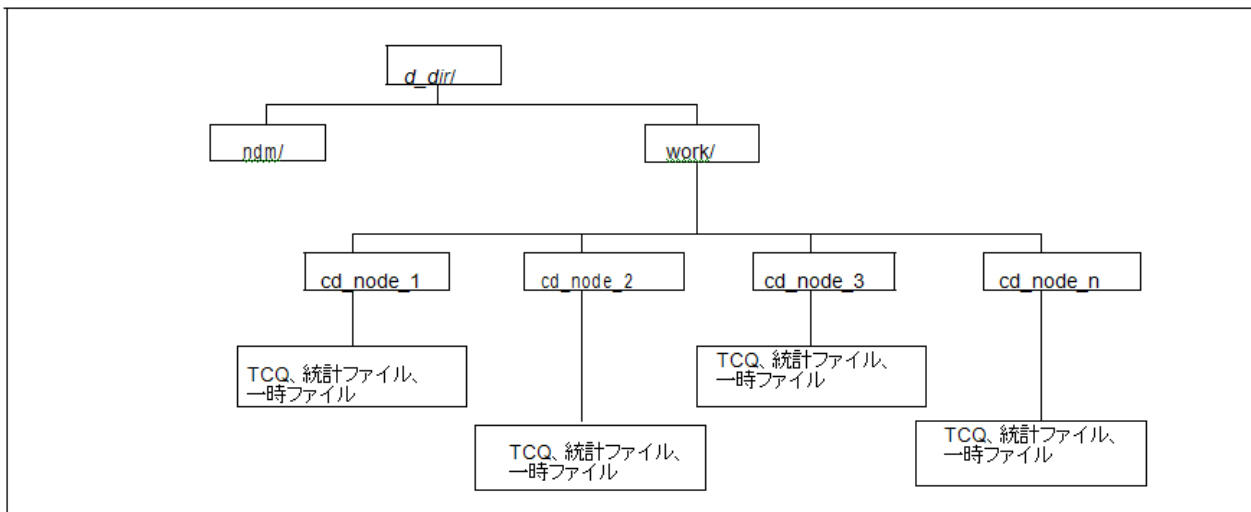
以下の図は、Sterling Connect:Direct for UNIX のディレクトリー構成を示しています。このディレクトリー・ツリーは、Sterling Connect:Direct for UNIX がインストールされる宛先ディレクトリーである `d_dir/` で始まっています。このディレクトリー構成は、同一のネットワーク上に存在、また、同一のコンピューター上に存在する可能性のある複数のノードについて表示しています。このディレクトリー構成の体系化によって、`cdpmgr` や `ndmcmgr` といった Sterling Connect:Direct for UNIX のプログラムを共有することができるようになります。複数のノードが存在する場合には、各ノードに対し、構成ファイル用に独自の `d_dir/ndm/cfg/cd_node/` ディレクトリー構成が存在する必要があります。`cd_node` は、Sterling Connect:Direct for UNIX のノード名になります。

注: Secure+ ディレクトリーは、Sterling Connect:Direct for UNIX Secure Plus がインストールされる場合にのみ利用可能です。



注: 以下の図を参照して、ノードの作業ディレクトリーを確認してください。

各ノードに対して、`d_dir/work/cd_node` が作成されます。以下の図は、複数のノードにおける作業ディレクトリー、および各ノードに対して作成された作業ファイル (例: TCQ ファイル) を示しています。



## Serling Connect:Direct for UNIX 文書

### 本書について

『IBM Sterling Connect:Direct for UNIX 管理ガイド (IBM Sterling Connect:Direct for UNIX Administration Guide)』は、Sterling Connect:Direct for UNIX をインストールするネットワーク運用スタッフおよびプログラマーを対象としています。本書によって、Sterling Connect:Direct for UNIX のインストールに必要な知識を得ることができます。

本書は、アプリケーション、ネットワーク、環境を含めた、UNIX オペレーティング・システムの知識があることを前提としています。LU6.2 接続においては、独立 LU6.2 接続をサポートするための SNA パッケージの構成における高度な能力が必要となります。

### タスクの概要

以下の表は、Sterling Connect:Direct for UNIX のインストールにおけるタスクの実行に必要な情報の参照先について案内しています。

タスク	参照先
Sterling Connect:Direct for UNIX の理解	第 1 章: <i>Sterling Connect:Direct for UNIX</i>
Sterling Connect:Direct for UNIX のインストール	第 2 章: <i>Installing Sterling Connect:Direct for UNIX のインストール</i>



タスク	参照先
Sterling Connect:Direct for UNIX のカスタマイズ	第 2 章: <i>Sterling Connect:Direct for UNIX</i> のインストール
Sterling Connect:Direct for UNIX サーバーのセットアップ	第 2 章: <i>Sterling Connect:Direct for UNIX</i> のインストール
Sterling Connect:Direct for UNIX クライアントのセットアップ	第 2 章: <i>Sterling Connect:Direct for UNIX</i> のインストール
強固なアクセス制御ファイルの作成、セッション・マネージャー、Process マネージャー、コマンド・マネージャー、ユーザー・マネージャー・プログラム上のファイル許可の変更	第 2 章: <i>Sterling Connect:Direct for UNIX</i> のインストール
Sterling Connect:Direct for UNIX File Agent のインストール	第 2 章: <i>Sterling Connect:Direct for UNIX</i> のインストール
Sterling Connect:Direct for UNIX Secure Plus のインストール	第 2 章: <i>Sterling Connect:Direct for UNIX</i> のインストール
Sterling Connect:Direct for UNIX File Agent を使用したファイルの管理	第 3 章: <i>Sterling Connect:Direct File Agent</i> を使用したファイルの管理
Sterling Connect:Direct for UNIX のインストールおよびカスタマイズに必要な情報の記入	付録 A: インストール用ワークシート
LU6.2 接続の構成	付録 B: SNA LU6.2 接続
マニュアル・ページへのアクセスおよび表示	付録 C: <i>Sterling Connect:Direct</i> マニュアル・ページ

# 構成ファイルの保守

Sterling Connect:Direct の運用環境は、構成ファイルによって定義されます。以下の構成ファイルがカスタマイズ手順実行中に作成されます。

- ◆ 初期化パラメーター・ファイル
- ◆ クライアント構成パラメーター・ファイル
- ◆ ネットワーク・マップ・ファイル
- ◆ アクセス・ファイル 2 つ: userfile.cfg および sysacl.cfg

最初のカスタマイズの後、これらのファイルを必要に応じて修正することが可能となります。この章では、構成ファイルを修正するための情報を記載しています。

## 構成ファイルについて

構成ファイルは、レコードから成るテキスト・ファイルです。レコードは、1 行の論理行です。論理行とは、バックスラッシュ (¥) 記号によってつながる 1 行以上の物理行のことです。19 ページの表では、第 4 および第 5 物理行が論理行の例を表しています。第 4 行は、その行が次の物理行に続くことを示すために、バックスラッシュ (¥) 記号で終了しています。サンプルの第 1 行は、シャープ (#) 記号で開始しています。シャープ記号は、この行にコメントが含まれていることを示しています。

レコードには、レコード名と、1 つ以上のパラメーター・ペアが含まれます。パラメーター・ペアとは、パラメーター名とパラメーター値のことです。第 2 行にはレコード名 ndm.path が含まれます。第 2 行には、パラメーター名が path、パラメーター値が /ndm/users/c のパラメーター・ペア path、/ndm/users/c も含まれています。パラメーター・ペアは、以下のフォーマットのように、コロン (:) で結び付けられ、等号 (=) で区切られています。以下は、レコード名 ndm.path、パラメーター名 path、パラメーター値 /ndm/users/c の 1 つの完全なレコードの例です。

```
ndm.path:path=/ndm/users/c:
```

レコード名とパラメーター名は大/小文字を区別しません。パラメーター値は大/小文字を区別します。

第 7 行から第 23 行は、より長い論理レコードを表しています。第 7 行はレコード名 local.node を含み、その後にオプションのコロン (:) とバックスラッシュ (¥) 記号が続いています。第 7 行と第 23 行の間の行はすべてバックスラッシュ (¥) 記号で終了しています。第 23 行はレコードの終わ

りなので、バックスラッシュ (\) 記号を含みません。

以下の表は初期化パラメーター・ファイルの一部を表し、Sterling Connect:Direct 構成ファイルのフォーマットを説明しています。

行	内容	注
1	#Miscellaneous Parameters	# 記号はコメントを示します。
2	ndm.path:path=/ndm/users/c:	レコード名 = ndm.path、 パラメーター = path、 値 = /ndm/users/c
3	proc.prio:default=8:	レコード名 = proc.prio、 パラメーター = default、 値 = 8
6	#Local Sterling Connect:Direct connection information	# 記号はコメントを示します。
7	local.node:¥	レコード名 = local.node
13	.	
...	.	
21	.	
22	:tcp.api=rusty;3191:¥	パラメーター = tcp.api、 値 = rusty;3191
23	:tcp.api.bufsize= <b>32768</b> :	パラメーター = tcp.api.bufsize、 値 = 32768

構成ファイルは重複が可能です。まったく同一のレコードに関しては、重複不可の場合もあります。例えば、初期化パラメーター・ファイル内に、複数のリモート・ノード情報 (**rnnode.listen**) レコードを定義することができます。

## 構成ファイルの修正

Sterling Connect:Direct の構成ファイルは、任意のテキスト・エディターで修正することができます。また、Sterling Connect:Direct for UNIX で提供される **cdcust** コマンドを使用して、新しい構成ファイルの作成が可能です。

- ◆ テキスト・エディターを使用した構成ファイルの修正: vi エディターなど、あらゆるテキスト・エディターを使用して、Sterling Connect:Direct for UNIX 構成ファイルを修正することができます。
- ◆ **cdcust** を使用した構成ファイルの作成: カスタマイズ手順を始動するには、以下のコマンドを入力してください。*d\_dir* は、Sterling Connect:Direct for UNIX のパス名です。

```
$ d_dir/etc/cdcust
```

---

## 初期化パラメーター・ファイルの保守

初期化パラメーターによって、システム・オペレーションを制御する、さまざまな Sterling Connect:Direct の設定が決定されます。初期化パラメーター・ファイルは、Sterling Connect:Direct for UNIX のインストール時に作成され、必要に応じて更新が可能です。

任意のテキスト・エディターを用いて、Sterling Connect:Direct 初期化パラメーター・ファイルを修正することができます。ファイル内の値を変更する前に、Sterling Connect:Direct サーバーをシャットダウンしてください。値を変更し、ファイルを保存した後、サーバーを再起動してください。再起動することにより、更新された値が検証され、無効の場合には、エラーメッセージが生成されます。この章では、すべての使用可能なパラメーターについて解説されています。

Sterling Connect:Direct Browser User Interface を使用して、ローカル・ノード接続レコードを更新する場合には、サーバーの終了および再起動は必要ありません。

---

**注:** この章に記載されている手順の中には、Sterling Connect:Direct Browser User Interface を用いて実行可能なものもあります。Sterling Connect:Direct Browser User Interface に関する詳細については、Sterling Connect:Direct Browser User Interface の CD-ROM またはオンラインで利用可能な IBM 資料ライブラリーからの文書を参照してください。

---

---

### 初期化パラメーター・ファイル

初期化パラメーター・ファイルは `d_dir/ndm/cfg/cd_node/initparm.cfg` にあります。`d_dir` は、Sterling Connect:Direct for UNIX がインストールされる宛先ディレクトリー、`cd_node` はノード名です。

初期化パラメーター・ファイルには、レコードが含まれます。各レコードには、そのレコードの属性を定義するパラメーターが含まれます。レコードの概要は以下の通りです。

- ◆ 各種パラメーター: Sterling Connect:Direct for UNIX のノード名、Sterling Connect:Direct for UNIX のロケーション、プラグ可能認証モジュール (PAM) サービス構成ファイルのロケーション、SNODE 作業ファイルの共有ワークエリアのロケーション、Process のデフォルト優先順位、実行ディレクトリー内で特殊文字を含むコマンドの制限の有無を含む、各種の情報を提供します。
- ◆ リモート・ノード接続情報: `rnode.listen` レコードには、インバウンド接続の監視のためのパラメーターが含まれます。

- ◆ 伝送制御キュー (TCQ) 情報: TCQ レコードによって、Process を削除前に、Held in Error で保留する時間を定義します。
- ◆ グローバルなコピー (copy) パラメーター: **copy.parms** レコードは、チェックポイント・パラメーター、ファイル・サイズ制限、変換テーブル情報、例外処理、CRC 検査、ファイル割り当て再試行パラメーター、および圧縮オプションといった、コピー操作で使用されるデフォルト・パラメーターを定義します。
- ◆ グローバルなタスク実行 (run task) パラメーター: **runtask.parms** レコードは、再試行オプションを定義するパラメーターを定義します。
- ◆ 統計ファイル情報: **stats** レコードには、ファイル・サイズ制限、統計ファイルに書き込まれる情報の種類、統計ファイルをアーカイブ前に保持する時間といった、デフォルト統計ファイル情報を定義するパラメーターが含まれます。
- ◆ サーバー認証情報: サーバーを認証するための、**authentication** レコード・パラメーターです。
- ◆ ユーザー出口パラメーター: **user.exits** レコードは、ユーザー出口手順で使用されるプログラムを定義します。
- ◆ ファイアウォール・ナビゲーション情報: **firewall.parms** レコードは、サーバーがファイアウォール越しに移動する際、アウトバウンド・セッションに使用されるポートまたはポート範囲を定義します。

以下の初期化パラメーター・ファイルのサンプルは、これらのパラメーターの一部が指定される様子を示しています。

```
# Miscellaneous Parameters
ndm.path:path=/sci/users/mscarbro/cd4000:¥
      :snode.work.path=/sci/users/mscarbro/cd4000/shared:

ndm.node:name=mws_joshua_4000:
ndm.pam:service=cdlogin:
ndm.quiesce:quiesce.resume=n

proc.prio:default=10: restrict:cmd=y

# TCQ information
tcq:¥
  :max.age=8:

# Global copy parameters.
copy.parms:¥
  :ckpt.interval=2M:¥
  :ulimit=N:¥
  :xlate.dir=/sci/users/mscarbro/cd4000/ndm/xlate:¥
  :xlate.send=def_send.xlt:¥
  :xlate.recv=def_recv.xlt:¥
  :continue.on.exception=y:
```

```
# Global runtask parameters.
runtask.parms:¥
:restart=y:

# Stat file info.
stats:¥
:file.size=1048576:¥
:log.commands=n:¥
:log.select=n:¥
:syslog.logd=daemon:

# Authenticator
authentication:¥
:server.program=/sci/users/mscarbro/cd4000/ndm/bin/ndmauths:¥
:server.keyfile=/sci/users/mscarbro/cd4000/ndm/security/keys.server:

# user exit information
user.exits:¥
:security.exit.program=:¥
:file.open.exit.program=:¥
:stats.exit.program=:

# Remote CDU nodes
rnode.listen:¥
:recid=rt.sles96440:¥
:comm.info=0.0.0.0;9974:¥
:comm.transport=udt33:

# Secure+ parameters
secure+:¥
:certificate.directory=/home/nis02/jlyon/certs:¥
:s+cmd.enforce.secure.connection=n:
```

## 各種パラメーターの更新

このセクションでは、各種レコードを指定し、使用可能なパラメーターを定義します。必須パラメーターは太字で表示されています。

### パス・レコードの更新

**ndm.path** レコードは、Sterling Connect:Direct のファイルのパスを特定します。以下の表は、このレコードで使用可能なパラメーターを示しています。

パラメーター	説明	値
path	すべての Sterling Connect:Direct のサブディレクトリーおよびファイルへのパス	パスの指定

## SNODE 作業パスの更新

**snode.work.path** パラメーターは、**ndm.path** レコードの一部であり、クラスター・ファイル・システム (NFS ではない) 上の SNODE 作業ファイルの共有ワークエリアを特定します。このオプションのパラメーターによって、SNODE 作業ファイルが負荷分散されるようノード間で共有される方法が与えられます。**snode.work.path** パラメーターが指定される場合、この領域内で SNODE 戻りコード・ファイル (steprc ファイル) および **copy** チェックポイント情報が作成されます。以下の表は、**snode.work.path** パラメーターを説明しています。

パラメーター	説明	値
snode.work.path	SNODE 作業ファイルの共有ワークエリアへのパス。 注: 同一クラスター内のすべてのノードにおいて、同一パスを指定してください。	パスの指定

## ノード名レコードの更新

**ndm.node** レコードは、Sterling Connect:Direct のノード名を特定します。以下の表は、このレコードで使用可能なパラメーターを示しています。

パラメーター	説明	値
name	ノード名。	最長 16 バイト。ノード名がそれより長い場合は切り捨てられます。

## PAM サービス・レコードの更新

**ndm.pam** レコードは、Sterling Connect:Direct の Process のユーザー認証に使用される PAM サービス構成ファイルを特定します。サービス初期化パラメーターが定義されており、PAM が Sterling Connect:Direct サーバーにインストールされている場合には、PAM がサービス提供アプリケーションに対するユーザー認証を行います。以下の表は、このレコードで使用可能なパラメーターを示しています。

パラメーター	説明	値
service	PAM サービス構成ファイル名。	ファイル名

## 停止/再開レコードの更新

**ndm.quiesce** レコードは、Sterling Connect:Direct が「テスト」モードで動作しているかどうかを特定します。テスト・モードを有効にするには、NDMPXTBL テーブルと併せて、このレコードを使用してください。**quiesce.resume** パラメーターを有効にする場合、インストール・ディレクトリー ndm/cfg/<nodename> 内に、使用環境に合わせて更新された NDMPXTBL パラメーター・テーブルが存在する必要があります。テスト・モードおよび NDMPXTBL テーブルに関する詳細については、『付録 C: テスト・モードでの Sterling Connect:Direct の使用』を参照してください。

以下の表は、このレコードで使用可能なパラメーターを説明しています。

パラメーター	説明	値
quiesce.resume	Sterling Connect:Direct のテスト・モードを有効化または無効化します。	y   n Y - テスト・モードを有効化。 n - テスト・モードを無効化。 デフォルトは n です。

## 優先順位レコードの更新

**proc.prio** レコードは Process の優先順位のデフォルト値を特定します。以下の表は、このレコードで使用可能なパラメーターを示しています。

パラメーター	説明	値
default	Process の優先順位のデフォルト値。	1 ~ 15。デフォルト値は <b>10</b> です。15 が最高優先度。

## 実行ディレクトリー内の特殊文字の使用制限

ユーザー構成ファイル (userfile.cfg) で実行ディレクトリー制限が定義されている場合には、**restrict** レコードによって、特定の特殊文字を含むコマンドの使用が許可されるかどうかを決定します。userfile.cfg ファイルに関する詳細については、58 ページの『ローカル・ユーザー情報レコード・フォーマットの更新』および62 ページの『リモート・ユーザー情報レコードの更新』を参照してください。このレコードでは、以下のパラメーターが使用可能です。

パラメーター	説明	値
cmd	特定の特殊文字を含むコマンドが許されているかどうかを決定します。	y   n y - 以下の特殊文字を含むコマンドの使用を制限します。 ; & '   n - 許可されるコマンドを制限しません。



## リモート・ノード接続レコードの更新

**rnode.listen** レコードには、インバウンド接続要求を監視するために、ローカル・ノードによって使用されるパラメーターが含まれています。サーバー稼働中に **rnode.listen** レコード内の IP アドレスおよびポート番号の変更を行うことができます。変更が有効になる前に、サーバーをリサイクルする必要があることに注意してください。以下の表は、リモート・ノード接続パラメーターを説明しています。

パラメーター	説明	値
recid	<b>rnode.listen</b> レコードの固有 ID。	テキスト・ストリング
comm.info	<p>リモート・ノードからの、TCP/IP または LU6.2 を使用した接続要求を監視するために必要な情報。このパラメーターでは、以下が必要とされます。</p> <ul style="list-style-type: none"> <li>• TCP/IP 接続の場合、ホスト名または IP アドレス、およびポート番号を指定します。IP アドレスとポートを指定する場合には、パラメーターをセミコロン (;) で区切ります。複数のアドレス/ホスト名は、コンマで区切ります。例: 10.23.107.5;1364, fe00:0:0:2014::7;1364, msdallas-dt;1364</li> <li>• LU6.2 接続の場合、リモート接続の SNA 構成プロファイルの名前として、プロファイル名を指定してください。Sterling Connect:Direct は、カスタマイズ手順の実行中にデフォルト名 hostl1 を生成します。AIX SNA では、hostl1 はサイド情報プロファイル名を参照します。HP SNA、SunLink SNA および Brixton SNA では、hostl1 は構成ファイルと同じディレクトリにある SNA プロファイル・ファイルを参照します。</li> </ul> <p>IP アドレスおよびホスト名の指定に関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』を参照してください。</p>	<p>TCP/IP 接続の場合、ホスト名または IP アドレス、およびポート番号を指定してください。</p> <p>10.23.107.5;1364</p> <p>複数の IP アドレス/ホスト名は、コンマで区切ります。</p> <p>fe00:0:0:2014::7;1364, msdallas-dt;1364</p> <p>読みやすくするため、コンマの後にスペースを加えることができます。</p> <p>特定のアダプターを監視するためにはその IP アドレスを設定し、すべてのアダプターを監視するためには <b>0.0.0.0</b> を指定します。</p> <p>デフォルト・ポートは <b>1364</b> です。LU6.2 接続の場合は、最長 8 字までのプロファイル名を指定します。</p>
comm.transport	リモート・ノードに対するトランスポート・プロトコル	<p>tcp   lu62   blklu62   udt33</p> <p>tcp - TCP/IP 接続</p> <p>lu62 - AIX SNA LU6.2 接続</p> <p>blklu62 - 他の LU6.2 接続</p> <p>udt33 - UDT 接続</p>

## 伝送制御キュー (TCQ) レコードの更新

tcq レコードは、伝送制御キューに関連する情報を提供します。このレコードでは、以下のパラメーターが使用可能です。

パラメーター	説明	値
max.age	Held in Error ステータスにある Process が、自動的に削除される前に、TCQ 内に留まる最大日数。	3 桁の 10 進数。max.age=0 の場合には、Sterling Connect:Direct は Process の自動削除を行いません。 デフォルトは 8 日です。

## Sterling Connect:Direct Secure Plus レコード (Secure+ レコード) の追加と更新

Sterling Connect:Direct Secure Plus レコード (Secure+ レコード) は、Sterling Connect:Direct クライアント API からの Sterling Connect:Direct Secure Plus のリモート構成に関連する情報を提供します。このレコードは、デフォルトでは initparm.cfg ファイルには含まれていません。initparm.cfg ファイルに Secure+ レコードを手動で追加する必要があります。このレコードでは、以下のパラメーターが使用可能です。

パラメーター	説明	値
certificate.directory	Sterling Connect:Direct クライアント API から Secure+ コマンド用に実行されたデフォルト証明書ディレクトリーを指定します。証明書ディレクトリーが構成されない場合には、インストール時に作成されたデフォルトのディレクトリーが使用されます。	ディレクトリー・パス名
s+cmd.enforce.secure.connection	非セキュア接続の Sterling Connect:Direct クライアント API からの Secure+ コマンドの受け入れの可否を指定します。	y n y - 非セキュア接続からのコマンドは拒否されます。デフォルトは y です。 n - 非セキュア接続からのコマンドを受け入れません。

## グローバルなコピー (copy) レコードの更新

**copy.parms** というグローバルな copy レコードは、Sterling Connect:Direct の copy 操作におけるデフォルト情報を提供します。ecz パラメーターは、Process で拡張圧縮が定義された場合にのみ使用されます。このレコードでは、以下のパラメーターが使用可能です。

レコード	説明	値
ckpt.interval	チェックポイントの前に、copy 操作で送信されるデフォルト・バイト数。 以下は、各バイト区分における、最大桁数のリストです。 no - チェックポイントなし nnnnnnnn - 8 桁までの 10 進数 nnnnnnnnK - 8 桁までの 10 進数で、K は 1024 バイトを表します。 nnnnnnnnM - 7 桁までの 10 進数で、M は 1048576 バイトを表します。 nnnnG - 4 桁までの 10 進数で、G は 1073741824 バイトを表します。	最大可能値は 1 テラバイト (TB)。通常値 は 64KB です。
ulimit	copy 操作時に、ユーザー出力ファイル・サイズ制限を超過した場合に取られるアクション。	n - 制限を無視します。n がデフォルト値です。 y - ユーザー・ファイルのサイズ制限を認識します。copy 操作時に、この制限を超えると、操作が失敗します。
xlate.dir	変換テーブルを含むディレクトリーの名前。	任意の有効ディレクトリー。 デフォルト・パスは <b>d_dir/ndm/xlate</b> です。
xlate.send	リモート・ノードへのデータの送信時に使用されるデフォルトの変換テーブル。	有効なディレクトリー。 デフォルトのファイル名は <b>def_send.xlt</b> です。
xlate.recv	リモート・ノードからのデータのコピー時に使用されるデフォルトの変換テーブル名。	デフォルトのファイル名は <b>def_recv.xlt</b> で、 <b>xlate.dir</b> パラメーターで定義されるディレクトリーに存在します。
continue.on.exception	Process での例外の処理に使用されるメソッド。リモート・ノードで実行された STOP IMMEDIATE または FLUSH 例外に起因してステップが失敗した場合は、このパラメーター値に関係なく、その Process は HE で保留キューに配置されます。	y - 次のステップから Process を続行します。 n - Process を HE の値を使用して保留キューに配置します。デフォルトは n です。

レコード	説明	値
ecz.compression.level	圧縮レベルを指定します。	1 ~ 9。デフォルトは 1 です。 1 - 最高速ですが、最も低い圧縮レベルを提供します。 9 - 最も高い圧縮レベルを提供しますが、最低速です。
ecz.memory.level	内部圧縮状態を維持するのにどれくらいの仮想メモリーを割り当てる必要があるのかを示します。	1 - 9。デフォルトは 4 です。 1 - 使用メモリーが最小です。 9 - 使用メモリーが最大です。
ecz.window.size	圧縮ウィンドウおよびヒストリー・バッファのサイズ。ウィンドウが大きいくほど、圧縮レベルも高くなりますが、仮想メモリーの使用量も増加します。	有効値は 9 ~ 15。 デフォルトは 13 です。
retry.codes	<p>ファイル割り当ての再試行を認識するコード。ファイル割り当ての再試行によって、ローカルまたはリモート・ノードにおいて、ファイル割り当てエラーまたはファイル・オープン・エラーの生じた Process を、エラーの発生した copy ステップを起点として再実行することができます。この機能により、ファイルがすでに使用されている場合でも、障害の発生した Process の再試行が可能になります。</p> <p>ローカル・ノードまたはリモート・ノードにおいてファイル割り当てエラーまたはファイル・オープン・エラーが発生すると、PNODE は retry.codes パラメーターおよび retry.msgids パラメーター内で、エラーまたはメッセージ ID を検索します。エラー・コードまたはメッセージ ID が検出されると、その Process が再試行されます。</p> <p>エラー・コードはオペレーティング・システムごとに異なり、同一のエラー・コードが異なる意味を持つ可能性があるため、2 つの異なるプラットフォーム間で通信する場合には、メッセージ ID を使用して、再試行状態を識別してください。</p> <p>再試行は、コードのみ、IDのみ、またはその両方の組み合わせに基づいて行うことができます。</p> <p>再試行状態が検出されると、そのセッションは正常に終了され、Process は タイマー・キューに配置されます。</p>	任意の有効エラー・コード。

レコード	説明	値
retry.msgids	<p>ファイル割り当て再試行をサポートするため、使用するメッセージ ID を識別します。</p> <p>エラー・コードはオペレーティング・システムごとに異なり、同一のエラー・コードが異なる意味を持つ可能性があるため、2 つの異なるプラットフォーム間で通信する場合には、メッセージ ID を使用して、再試行状態を識別してください。</p> <p>ローカルまたはリモート・ノードで、ファイル割り当てまたは読み込みエラーが発生すると、PNODE は retry.msgids パラメーター内で、メッセージ ID を検索します。メッセージ ID が見つかったと、その Process は再試行されます。</p> <p>再試行は、コードのみ、メッセージ ID のみ、またはその両方の組み合わせに基づいて行うことができます。</p> <p>再試行状態が検出されると、そのセッションは正常に終了され、Process はタイマー・キューに配置されます。</p>	<p>任意の有効なファイル割り当て再試行メッセージ。</p>
tcp.crc	<p>TCP/IP Process での CRC 機能をグローバルにオンまたはオフにします。</p>	<p>y   n</p> <p>y - CRC 機能をグローバルにオンにします。</p> <p>n - CRC 機能をグローバルにオフにします。デフォルトは n です。</p>
tcp.crc.override	<p>CRC 検査に対する、ネットマップ・リモート・ノードおよび Process ステートメントのオーバーライドが許可されるかどうかを決定します。この値が n に設定されると、CRC 検査に対する設定のオーバーライドは無効となります。</p>	<p>y   n</p> <p>y - CRC 検査に対する、ネットマップ・リモート・ノードおよび Process ステートメントのオーバーライドを許可します。</p> <p>n - CRC 検査に対する、ネットマップ・リモート・ノードおよび Process ステートメントのオーバーライドを許可しません。デフォルトは n です。</p>
strip.blanks	<p>末尾の空白文字がレコード末から削除されるかどうかを決定します。strip.blanks が初期化パラメーター内で定義されていない場合には、デフォルト値 i が使用されます。</p>	<p>y   n   i</p> <p>y - 空白文字をレコード末から削除します。</p> <p>n - 空白文字をレコード末から削除しません。</p> <p>i - strip.blanks の設定は、リモート・ノード・タイプのデフォルト値によって、以下のように決定されます。</p> <ul style="list-style-type: none"> <li>● z/OS、VM、VSE、および i5OS - y</li> <li>● 他のすべてのプラットフォーム - n</li> </ul>

レコード	説明	値
insert.newline	datatype=text のファイルの受信時に、各レコード末に適宜改行文字 (LF) を付加します。デフォルトでは、レコード末に既に改行文字 (LF) が存在する場合には付加されません。	y   n y - 適宜、改行文字を付加 n - 必要な場合には改行文字を付加

## グローバルなタスク実行 (run task) レコードの更新

**run task.parms** と呼ばれるグローバルなタスク実行 (run task) レコードは、再始動時に PNODE と SNODE が再同期できない場合に使用されます。SNODE で run task ステップ実行中に Process が中断されると、Sterling Connect:Direct は、SNODE での直前の run task ステップを、現在の run task ステップに同期させるよう試みます。同期に失敗した場合には、Sterling Connect:Direct は **restart** パラメーターを読み取り、**run task** ステップを再実行するかどうかを決定します。このレコードでは、以下のパラメーターが使用可能です。

パラメーター	説明	値
restart	SNODE のステップで <b>run task</b> 実行中に処理が中断された場合および再始動後に同期に失敗した場合、Sterling Connect:Direct は <b>restart</b> パラメーターを読み取り、 <b>run task</b> ステップを再実行するかどうかを決定します。SNODE の初期化パラメーター・ファイル内でこのパラメーターを設定してください。  注: ロード・バランシング・クラスターが使用され、かつ、snode.work.path が指定されると、再同期に失敗した場合に限って <b>restart</b> パラメーターが有効となります。	y   n y - run task プログラムが再実行されます。デフォルトは y です。 n - Process は <b>run task</b> ステップをスキップします。

## 統計ファイル情報レコードの更新

**stats** と呼ばれる統計ファイル情報レコードは、統計に関する機能を定義します。このレコードでは、以下のパラメーターが使用可能です。

パラメーター	説明	値
file.size	個々の統計データ・ファイルの最大サイズ (バイト)。統計ファイル名は <b>Syyyymmdd.ext</b> のフォーマットで記述されます。 <b>yyyy</b> は年を表し、 <b>mm</b> は月を表し、 <b>dd</b> は日を表します。拡張子 ( <b>ext</b> ) は 001 から始まります。24 時間以内に統計ファイルが定義されたサイズに達した場合には、同一ファイル名の新ファイルが作成され、拡張子の値が 1 増加します。	nnnnnnnn、nnnnnnnnK、nnnnnnnnM または nnnnG - 統計ファイルのデフォルト出力ファイルのサイズ制限を設定します。K は 1024 バイトを表します。M は 1048576 バイトを表します。G は 1073741824 バイトを表します。指定できる最大値は 1 TB です。
log.commands	コマンドが統計ファイルに書き込まれるかどうかを決定します。select statistics コマンドおよび select process コマンドを除くすべてのコマンドを記録したい場合には、このパラメーターを y に、log.select パラメーターを n に設定してください。	y   n y - コマンドは統計ファイルに書き込まれます。 n - コマンド統計ファイルに書き込まれません。デフォルトは n です。
log.select	select process コマンドまたは select statistics コマンドの実行時に、Sterling Connect:Direct によって統計レコードが作成されるかどうかを指定します。	y   n y - 統計レコードが作成されます。 n - 統計レコードは作成されません。デフォルトは n です。
syslog.logd	Sterling Connect:Direct が UNIX syslogd デーモンに対してエラー・メッセージを送信できるようにします。syslog 構成手順に関しては、マニュアル・ページを参照してください。	使用可能な値: kern - カーネル user - ユーザー・レベル mail - メール・サブシステム daemon - システム・デーモン auth - セキュリティーまたは認証  syslog - syslogd デーモン lpr - ライン・プリンター・サブシステム news - ニュース・サブシステム uucp - uucp サブシステム。 デフォルト値は <b>daemon</b> です。

パラメーター	説明	値
max.age	アーカイブの対象となる前に、統計ファイルが満たす必要のある経過時間を指定します。毎日 1 回、max.age の経過時間と一致する統計ファイルを識別し、tar コマンドおよび compress コマンドを実行して圧縮アーカイブを作成した後、アーカイブされた統計ファイルを削除するシェル・スクリプトが実行されます。	3 桁の 10 進数。 デフォルトは 8 日です。 0 -アーカイブなし。

Process の実行によって、複数の統計レコードが生成されます。大量に生成される統計レコードに対応するため、Sterling Connect:Direct は、毎日午前零時に、現行の統計ファイルを閉じ、新しい統計ファイルを作成します。ファイル・サイズが **file.size** 初期化パラメーターに設定された値を超える場合には、午前零時より前に現行のファイルを閉じることもできます。デフォルトのファイル・サイズは1 メガバイトです。

統計ファイルは、`d_dir/work/cd_node` ディレクトリーに保存されます。統計ファイル名のフォーマットは **Syyyymmdd.ext** で、yyyy は年、mm は月、dd は日を示します。拡張子 (**ext**) は 001 から開始します。同日中に新しい統計ファイルが作成される度に、拡張子は 1 増加します。

Sterling Connect:Direct for UNIX は、統計ファイルをアーカイブおよびページするユーティリティを提供しています。統計ファイルをアーカイブするタイミングは、max.age パラメーターの設定によって指定してください。max.age パラメーターは、アーカイブされる前に、統計ファイルが満たす必要のある経過時間を指定します。毎日 1 回、statach.sh と呼ばれるスクリプトが開始されます。このスクリプトは、max.age 以上の統計ファイルを識別した後、max.age パラメーターと一致する全統計レコードの圧縮アーカイブ・ファイルを作成するために、tar コマンドおよび compress コマンドを実行します。統計ファイルがアーカイブされると、これらのファイルはページされます。

アーカイブ・ファイルは、統計ファイルおよび TCQ (伝送制御キュー) が保存されているディレクトリーに保存されます。シェル・スクリプト `tatach.sh` は、`ndm/bin` ディレクトリーに置かれています。使用環境の必要に応じて、スクリプトを適宜修正してください。

アーカイブ済みの統計ファイルをリストアップしたい場合には、`statstore.sh` スクリプトを実行します。このスクリプトは、tar コマンドを使用して、アーカイブにあるすべての統計ファイルをリストアップします。ファイルがリストアップされると、`select statistics` コマンドを使用して、統計レコードを表示することができます。

## サーバー認証レコードの更新

**authentication** と呼ばれるサーバー認証レコードは、認証手順の実行中に使用されます。このレコードでは、以下のパラメーターが使用可能です。



パラメーター	説明	値
server.program	認証手順の実行中に使用されるサーバー・プログラム名およびそのロケーション。	デフォルトは <b>ndmauths</b> です。
server.keyfile	認証手順の実行中に使用されるキー・ファイル名およびそのロケーション。	デフォルトは <b>keys.server</b> です。

## ユーザー出口レコードの更新

**user.exits** と呼ばれるユーザー出口レコードは、指定されたプログラムへのインターフェースを提供します。使用可能なユーザー出口には、統計出口、ファイル・オープン出口、およびセキュリティ出口があります。このレコードでは、以下のパラメーターが使用可能です。

パラメーター	説明	値
stats.exit.program	ユーザー出口の手順実行中に使用されるゲートウェイ制御プログラム。統計レコードが書き出される度に、この出口に制御が渡されます。	ゲートウェイ制御プログラム名
file.open.exit.program	ユーザー出口の手順実行中に使用されるファイル・オープン出口プログラム。これにより、送信および受信ノード上のファイル名の制御が可能になります。この出口は、ファイルが開かれる前に受信 (リモート) ノードを制御するために配置されています。この出口は <b>copy</b> ステートメントのみに適用され、すべてのファイル制御パラメーター (データ・セット名、ファイル名、sysopt パラメーター、および処理を含む) に対するアクセスを提供します。	ファイル・オープン出口プログラム名
security.exit.program	ユーザー出口の手順実行中に使用されるセキュリティ出口プログラム。この出口は、パスチケットを生成および検証するとともに、MVS ホスト上で使用可能、かつ、NETSP 製品を使用する UNIX AIX および OS/2 コンピューター上で IBM によってサポートもされている、RACF セキュリティ・システムの一部である PASSTICKET といった、他のパスワード対応プログラムもサポートします。	セキュリティ出口プログラム名
security.exit.flag	security.exit.program のデフォルト動作を変更します。このパラメーターはオプションです。	snode_sec_exit_only   sec_exit_only  snode_sec_exit_only - Sterling Connect:Direct が SNODE として機能している時にセキュリティ出口を使用するようにします。Sterling Connect:Direct が有効な メッセージを受信した後、ローカル・ユーザーを確立するために、プロキシおよびセキュア・ポイント・オブ・エン트리 (SPOE) を評価します。セキュリティ出口は、Sterling Connect:Direct が PNODE の場合は使用されません。  sec_exit_only - Sterling Connect:Direct が常にセキュリティ出口を使用するようにします。Sterling Connect:Direct が有効なメッセージを受信した後、ローカル・ユーザーを確立するために、プロキシおよびセキュア・ポイント・オブ・エン트리 (SPOE) を評価します。

## ファイアウォール・ナビゲーション・レコードの更新

firewall.parms と呼ばれるファイアウォール・ナビゲーション・レコードによって、アウトバウンド Sterling Connect:Direct セッションに対して、特定のTCP/IP および UDT アドレスに対応する、特定の TCP/IP および UDT ソース・ポート番号またはポート番号範囲の割り当てが可能となります。これらのポートは、インバウンド Sterling Connect:Direct セッションを許可するために、取引先のファイアウォールでも開いている必要があります。この機能は、セキュリティ・ポリシーに違反することなく、パケット・フィルタリング・ファイアウォール環境内にある Sterling Connect:Direct サーバーへの制御されたアクセスを可能にします。

**注:** UDT で使用する際のファイアウォールの構成を行う前に、『付録 A: ファイアウォール・ナビゲーションの構成』を参照して、UDT および TCP 間での、セッション確立およびファイアウォール・ナビゲーションにおける違いについて確認してください。

このレコードでは、以下のパラメーターが使用可能です。

パラメーター	説明	値
tcp.src.ports	TCP/IP 接続の場合、パケット・フィルタリング・ファイアウォール使用時の、リモート IP アドレスおよびそのアドレスに許可されるポート。このパラメーターは、ローカル・ノードが PNODE として機能する時にのみ必要とされます。 1 つのアドレスに対するすべての値を括弧の中に入れ、各値をコンマで区切ってください。	有効 IP アドレス (オプションとして IP アドレス範囲の上限を示すマスクも可能)、および指定された IP アドレスに関連する出力ポート番号またはポート番号の範囲。例えば、 (199.2.4.*, 1000)、(fd00:0:0:2015:*:* 2000-3000)、(199.2.4.0/255.255.255.0, 4000-5000)、(fd00:0:0:2015::0/48, 6000, 7000) IP アドレスのパターンを定義するワイルドカード文字 (*) がサポートされています。ワイルドカード文字が使用された場合には、オプションのマスクは無効となります。 IP アドレス、マスクおよびポートに関する詳細については、『付録 B: アドレス、ホスト名、およびポートの指定』を参照してください。
tcp.src.ports.list.iterations	Sterling Connect:Direct が再試行状態に移行する前に、接続の試行に使用可能なポートのリストをスキャンする回数。	1 ~ 255 の任意の数値。デフォルト値は 2 です。

パラメーター	説明	値
udp.src.ports	<p>UDT 接続の場合、パケット・フィルタリング・ファイアウォール使用時の、リモート IP アドレスおよびそのアドレスに許可されるポート。このパラメーターは、ローカル・ノードが PNODE として機能するか SNODE として機能するかに関わらず、ファイアウォール使用時に推奨されます。</p> <p>1 つのアドレスに対するすべての値を括弧の中に入れ、各値をコンマで区切ってください。</p>	<p>有効 IP アドレス(オプションとして IP アドレスの範囲の上限を示すマスクも可能)、および指定された IP アドレスに関連する出力ポート番号またはポート番号の範囲。例えば、</p> <p>(199.2.4.*, 1000)、(fd00:0:0:2015::*, 2000-3000)、(199.2.4.0/255.255.255.0, 4000-5000)、(fd00:0:0:2015::0/48, 6000, 7000)</p> <p>IP アドレスのパターンを定義するワイルドカード文字 (*) がサポートされています。ワイルドカード文字が使用された場合には、オプションのマスクは無効となります。</p> <p>IP アドレス、マスクおよびポートに関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』参照してください。</p>
udp.src.ports.list.iterations	<p>Sterling Connect:Direct が再試行状態に移行する前に、接続の試行に使用可能なポートのリストをスキャンする回数。</p>	<p>1 ~ 255 の任意の数値。デフォルト値は 2 です。</p>

---

## クライアント構成ファイルの保守

クライアント構成ファイルは、エンド・ユーザー・アプリケーション (End User Applications: EUA) とのインターフェースをとるパラメーター・レコードから構成されます。クライアント・ファイルには以下のパラメーターが含まれます。

- ◆ Sterling Connect:Direct API 構成パラメーター
- ◆ Sterling Connect:Direct CLI 構成パラメーター
- ◆ クライアント認証パラメーター

Sterling Connect:Direct 構成ファイルは、任意のテキスト・エディターで編集することが可能です。新しい構成ファイルを作成したい場合には、**cdcust** コマンドを使用してください。

---

### クライアント構成ファイル

クライアント構成ファイルはカスタマイズ手順の実行中に作成され、`d_dir/ndm/cfg/cliapi/ndmapi.cfg` にあります。`d_dir` は、Sterling Connect:Direct がインストールされているディレクトリーです。

以下の例は、クライアント構成ファイルのサンプルを表示しています。

```
# Connect:Direct for UNIX Client configuration file

cli.parms:¥
:script.dir=/home/qatest/jsmith/cdunix/hp/ndm/bin/:¥
:prompt.string="Test CD on Medea":

api.parms:¥
:tcp.hostname=alicia:¥
:tcp.port=1393:¥
:wait.time=50:

# Authenticator
authentication:¥
:client.program=/home/qatest/jsmith/cdunix/hp/ndm/bin/ndmauthc:¥
:client.keyfile=/home/qatest/jsmith/cdunix/hp/ndm/sc/keys.client:
```

## API 構成レコードの更新

Sterling Connect:Direct の API 構成レコード **api.parms** は、API 通信のために使用されます。API 構成レコードのパラメーターは、以下の表で説明されています。

パラメーター	説明	値
tcp.hostname	API が通常接続するホスト名または IP アドレス。	ホスト名または IP アドレス。 IP アドレスおよびホスト名の指定に関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』を参照してください。
tcp.port	API が通常接続する先の TCP/IP ポート番号。	ポート番号。デフォルトは <b>1363</b> です。
wait.time	サーバーからの応答を待つ待機時間 (秒)。この制限を超えると、メッセージ ID XCMG000I が表示されます。	待機時間 (秒)。デフォルトは <b>50</b> 秒です。

## CLI 構成レコードの更新

CLI 構成レコード **cli.parms** は、**select statistics** コマンドおよび **select process** コマンドの出力フォーマットを設定するスクリプト・ファイルのロケーションを指定し、CLI プロンプトのカスタマイズを可能とします。**select statistics** コマンドおよび **select process** コマンドの出力フォーマットを設定するために、このスクリプトをカスタマイズする場合には、スクリプトのロケーションを指定する **script.dir** パラメーターを更新してください。CLI コマンドラインに、デフォルトの「Direct」プロンプトの代わりにカスタマイズされたプロンプトを表示したい場合は、**prompt.string** パラメーターで使用するプロンプトを指定してください。**cli.parms** パラメーターは、以下の表で説明されています。

パラメーター	説明	値
script.dir	カスタマイズされたスクリプト・ファイルの保存されているディレクトリー。 <b>select statistics</b> コマンドおよび <b>select process</b> コマンドの出力フォーマットを設定するためにカスタム・スクリプトを作成済みの場合には、このパラメーターを指定してください。ファイル名は必ず <b>ndmstat</b> および <b>ndmproc</b> でなければなりません。	ディレクトリー名。 デフォルト・ディレクトリーは <b>ndm/bin/</b> です。
prompt.string	クライアントが開始された時に、コマンドラインに表示する CLI プロンプトを指定します。 スペースもしくは特殊文字がプロンプト・ストリングに含まれる場合は、それを単一引用符または二重引用符で囲ってください。 このパラメーター内および コマンドライン (-P パラメーターを使用) で、カスタマイズされたプロンプトの設定が可能です。プロンプト・ストリングが両方に指定された場合、コマンドラインの -P パラメーターが優先されます。 デフォルトのプロンプトがオーバーライドされると、Welcome バナーおよびコマンド・プロンプトに新しいプロンプト・ストリングが表示されます。	最大 32 文字のプロンプト・ストリング。 デフォルトは「 <b>Direct</b> 」です。

## クライアント認証レコードの更新

クライアント認証レコード **authentication** は、認証手順の実行中に使用されます。クライアント認証パラメーターは以下の表で説明されています。

パラメーター	説明	値
client.program	認証手順の実行中に使用されるクライアント・プログラム。	クライアント・プログラム名。 デフォルトは <b>ndmauthc</b> です。
client.keyfile	認証手順の実行中に使用されるキー・ファイル。	クライアント・キー・ファイル。 デフォルトは <b>keys.client</b> です。





---

# ネットワーク・マップ・ファイルの保守

この章では、ネットワーク・マップ・ファイル中のパラメーターについて説明します。このファイルは、Sterling Connect:Direct のインストール時に作成されます。必要に応じて、テキスト・エディターを使用して、ネットワーク・マップ・ファイルにあるリモート・ノード・レコードを追加または修正してください。サーバー稼働中に、ネットワーク・マップ・ファイルを動的に変更することも可能です。

---

**注** この章に記載されている手順の中には、Sterling Connect:Direct Browser User Interface を用いて実行可能なものもあります。Sterling Connect:Direct Browser User Interface に関する詳細については、Sterling Connect:Direct Browser User Interface の CD-ROM またはオンラインで利用可能な IBM 資料ライブラリーからの文書を参照してください

---

---

## ネットワーク・マップ・ファイル

ネットワーク・マップには、ネットワーク内のローカル・ノードおよびリモート・ノードについて説明する接続情報が含まれます。ローカル・ノードが通信を行う各ノードに対して、1つのリモート・ノード情報レコードが作成されます。

ネットワーク・マップ・ファイルは、`d_dir/ndm/cfg/cd_node/netmap.cfg` にあります。`d_dir` は Sterling Connect:Direct がインストールされているディレクトリー、`cd_node` はノード名です。

---

**注:** TCP/IP を使用している場合、ローカル・ノードは、リモート・ノード情報レコードを必要とすることなく、リモート・ノードと通信することが可能です。submit コマンドまたは Process ステートメントで必要な接続情報を指定してください。

---

## リモート・ノードのネットワーク・マップ・エントリーのサンプル

以下のサンプルは、リモート・ノードへの TCP/IP 接続および Sun LU6.2 接続に対するネットワーク・マップ上のリモート・ノード・エントリーを示しています。

```
# Sample Network Map remote node entry for a TCP/IP connection
remote.customer.node:¥
:conn.retry.stwait=00.00.30:¥
:conn.retry.stattempts=3:¥
:conn.retry.ltwait=00.10.00:¥
:conn.retry.ltattempts=6:¥
:tcp.max.time.to.wait=180:¥
:runstep.max.time.to.wait=0:¥
:contact.name=:¥
:contact.phone=:¥
:descrip=:¥
:sess.total=255:¥
:sess.pnode.max=255:¥
:sess.snode.max=255:¥
:sess.default=1:¥
:comm.info=10.20.246.49;9974:¥
:comm.transport=tcp:¥
:comm.bufsize=65536:¥
:pacing.send.delay=0:¥
:pacing.send.count=0:
# Sample Network Map remote node entry for a Sun LU6.2 connection
# hostl1 is the profile name
MVS.SAM1.NODE:¥
:conn.retry.stwait=00.00.30:¥
:conn.retry.stattempts=3:¥
:conn.retry.ltwait=00.10.00:¥
:conn.retry.ltattempts6:¥
:contact.name=:¥
:contact.phone=:¥
:descrip=:¥
:sess.total=255:¥
:sess.pnode.max=128:¥
:sess.snode.max=127:¥
:sess.default=1:¥
:comm.info=hostl1:¥
:comm.transport=blkl62:¥
:comm.bufsize=16000:
```

**注:** ネットワーク・マップ内でフィールドに関するコメントを挿入するには、必ず最初の列に # を入力してください。最初の列に # がない場合、コメントとは認識されずに、フィールドが読み込まれてしまいます。

## ローカル・ノード接続レコードの更新

local.node レコードには、2 つの異なる目的があります。ローカル・ノード設定を構成すること、そして、リモート・ノード・エントリーでオーバーライドされることが可能なデフォルト構成値を提供する

ことです。short-term および long-term の、2 セットの接続再試行パラメーターが作成されます。short-term パラメーターは、short-term 接続失敗時の再試行を定義します。Sterling Connect:Direct は、short-term 試行をすべて終了した後、long-term パラメーターを使用します。long-term 試行では、接続の問題がすぐには解決されないことを想定しているため、より低頻度の試行が行われるよう設定されています。

以下は、local.node レコードのパラメーターです。太字は必須パラメーターです。

パラメーター	説明	値
api.max.connects	ローカル・ノードに対して許可される、同時 API 接続の最大数。 api.max.connects の値および sess.total の値は、使用可能なファイル記述子の数を超えることはできません。この値はシステムに依存します。  正常に確立した各 API 接続に対して、コマンド・マネージャー (CMGR) が作成されます。PMGR が作成できるコマンド・マネージャーの数はシステムに依存し、各 UNIX Process で使用可能なファイル記述子の数によって制限されます。UNIX オペレーティング・システムで設定されるファイル記述子の数は、Sterling Connect:Direct の稼動に影響を与える可能性があります。	1 ~ 256。 デフォルトは <b>16</b> です。
comm.bufsize	TCP/IP 接続での、リモート・ノードとのデータ送受信におけるバッファー・サイズです。	TCP/IP では、値の制限はありません (2,147,483,623 まで)。LU6.2 では、最大 32000 です。デフォルトは <b>65536</b> バイトです。
conn.retry.stwait	接続障害発生直後からの各接続再試行間の待機時間。フォーマットは <b>hh.mm.ss</b> で、 <b>hh</b> は時間、 <b>mm</b> は分、 <b>ss</b> は秒をそれぞれ指定します。	最大値はクロック・フォーマットの最高値 23.59.59 に制限されます。デフォルトは <b>00.00.30</b> 、すなわち 30 秒です。
conn.retry.stattempts	接続障害が発生した後に、接続再試行を行う回数。	0 ~ 9999。 デフォルトは <b>6</b> です。
conn.retry.ltwait	すべての short-term 再試行が終了した後の、各 long-time 再試行間の待機時間。フォーマットは <b>hh.mm.ss</b> で、 <b>hh</b> は時間、 <b>mm</b> は分、 <b>ss</b> は秒をそれぞれ指定します。	00.00.00 ~ 23.59.59。 デフォルトは <b>00.10.00</b> 、すなわち 10 分です。
conn.retry.ltattempts	すべての short-term 接続再試行の終了後に、接続再試行を行う回数。	0 ~ 9999。 デフォルトは <b>6</b> です。

パラメーター	説明	値
conn.retry.exhaust.action	指定された short-term および long-term の接続再試行がすべて終了した後に取られるアクション。	hold   delete hold - すべての接続再試行終了後に Process を「Held in Error」ステータスで、保留キューに配置します。これは、デフォルト値です。 delete - TCQ (伝送制御キュー) から Process を削除します。
contact.name	Sterling Connect:Direct 管理者またはオペレーターの名前。	名前
contact.phone	Sterling Connect:Direct 管理者またはオペレーターの電話番号。	電話番号
descrip	レコードの一部として含まれるコメント。	制限なしのテキスト・ストリング
lu62.writex.wait	IBM AIX オペレーティング・システムで SNA を使用している場合には、このパラメーターを使用して、接続再試行するまでの待機時間を指定します。	0:00:00 ~ 59:59:59。 値のフォーマットは hh:mm:ss です。デフォルト値は 1 です。
lu62.writex.retry.attempts	IBM AIX オペレーティング・システムで SNA を使用している場合には、このパラメーターを使用して、接続を試みる回数を指定します。	0 ~ 2,147,483,647。 デフォルト値は 0 です。
netmap.check	SNODE で、拡張セキュリティ・テストが実行されます。TCP/IP 接続では、着信ソケット接続のリモート IP アドレスが、netmap.cfg ファイルの comm.info レコードと比較されます。これらの値は、確立される Sterling Connect:Direct セッションと一致する必要があります。comm.info レコードは、正式なネットワーク名でも、適切なファイルにリストされたエイリアス名でも (例えば、システムが NIS または DNS を導入していない場合には /etc/hosts)、IP アドレスでもかまいません。すべての接続において、リモート・ノード名が netmap.cfg に存在する必要があります。	y   n y - セキュリティ・チェックにより、リモート・ノード名が netmap.cfg ファイルにあることを検証するように指定します。 n - これらのセキュリティ・チェックが一切行われないうように指定します。 デフォルト値は n です。

パラメーター	説明	値
outgoing.address	高可用性環境で実行されている場合、このパラメーターを使用することで、リモート・ノードがネットワーク・マップ・チェックに使用する仮想 IP アドレスの指定が可能となり、高可用性環境内から開始する際に、Process がエラーとなることを防止します。この値に IP アドレスが指定されると、ネットワーク・マップ・チェックは SNODE のネットワーク・マップ レコードの comm.info 内に設定された値ではなく、このアドレスを検証します。	nnn.nnn.nnn.nnn (IPv4) または nnnn:nnnn:nnnn:nnnn:nn nn:nnnn:nnnn:nnnn (IPv6)  IP アドレスおよびホスト名の指定に関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』を参照してください。
pacing.send.delay	リモート・ノードへの send 操作間での待機時間です。1つのパケットの最後と、それに続くパケットの最初との間の時間をミリ秒単位の 10 進数で表します。時間ベースのペーシングは、ネットワーク・トラフィックの要因とはなりません。 このパラメーターの値は、LU6.2 接続に対しては無効です。	フォーマットは nnn です。 この値のサイズ制限はありません。 デフォルトは 0 です。このタイプのペーシングがないことを示しています。
pacing.send.count	リモート・ノードからのペーシング応答を自動的に待機する状態に移行する前に実行される、送信オペレーションの数。 このパラメーターの値は、LU6.2 接続に対しては無効です。	フォーマットは nnn です。 この値のサイズ制限はありません。 デフォルトは 0 です。このタイプのペーシングがないことを示しています。

パラメーター	説明	値
proxy.attempt	<p>snodeid の ID サブパラメーターが、リモート・システム上のローカル・ユーザー ID への変換に使用されるプロキシまたはダミー・ユーザー ID を含むことを可能にします。ダミー・ユーザー ID の使用によって、ローカル・システムおよびリモート・システムのどちらも、相手サイドからの有効なユーザー ID を必要としないため、セキュリティが強化されます。</p> <p>以下のコードは、ユーザー ID のセキュリティ・チェックの実行に使用するロジックを示しています。</p> <pre> if proxy.attempt = yes   if snodeid (ID only or ID &amp;   password) is specified    if ID@PNODE found in userfile.cfg     security checking OK   else     if (ID, PSWD) is valid UNIX ID &amp; password       security OK     else       security checking failed   else /*snodeid is NOT specified */     if submitter@PNODE found in userfile.cfg       security OK     else       security checking failed   else /*i.e. proxy.attempt = no, the default value */     if snodeid (ID &amp; password) is specified       if (ID, PSWD) is valid UNIX ID &amp; password         security OK       else         security checking failed     else /*snodeid is NOT specified */       if submitter@PNODE found in userfile.cfg         security OK       else         security checking failed </pre>	<p>y   n</p> <p>y - リモート・ユーザーが snodeid パラメーター内にダミー・ユーザー ID を指定できるように指定します。</p> <p>n - リモート・ユーザーが snodeid パラメーター内にダミー・ユーザー ID を指定できないように指定します。</p> <p>デフォルトは n です。</p>

パラメーター	説明	値
runstep.max.time.to.wait	リモートの実行ステップの完了を待機する最大時間。リモートの実行ステップには、リモートでの run task、run job、または submit ステートメントが含まれます。この待機時間は、tcp.max.time.to.wait パラメーターで指定される待機時間とは異なります。 runstep.max.time.to.wait を使用することにより、リモートのステップの完了に tcp.max.time.to.wait での指定値よりも長い時間を要する場合に、Process がエラーとなることを防止します。	0 ~ 10000。 秒単位の値。 デフォルト値は <b>0</b> です。
sess.total	全ノードとローカル・ノード間の最大の同時接続数。 api.max.connects および sess.total の合計が、使用可能なファイル記述子の数を超えることはできません。この値はシステムに依存します。 許可される Sterling Connect:Direct 同時セッションの数 (最大 999 個) を処理するのに十分な数のファイル記述子を定義する必要があります。	0 ~ 999。 1 ~ 3 桁の数字。 デフォルトは <b>255</b> です。
sess.pnode.max	ローカル・ノードがセッションのイニシエーターである場合の最大の同時接続数。 PNODE セッション数は、セッション総数を超えることはできません。 sess.pnode.max が sess.total よりも大きい場合、sess.pnode.max の値は自動的に sess.total の値まで切り捨てられます。	0 ~ 999。 デフォルトは <b>255</b> です。
sess.snode.max	ローカル・ノードがセッションの 2 次ノードである場合の最大の同時接続数。 SNODE セッション数は、セッション総数を超えることはできません。sess.snode.max が sess.total よりも大きい場合、自動的に sess.total.0 の値に変更されます。	0 ~ 999。 デフォルトは <b>255</b> です。
sess.default	セッション・マネージャーを開始するデフォルト・セッション・クラス。Process は、指定されたクラス、またはより上位の任意のセッション・クラスで実行されます。	1 ~ 50。 デフォルトは <b>1</b> です。
tcp.api.bufsize	Sterling Connect:Direct CLI/API とのデータ送受信におけるバッファー・サイズ。	この値に制限はありません。 デフォルトは <b>32768</b> バイトです。

パラメーター	説明	値
tcp.api	CLI または API からの、TCP/IP を使った接続要求の監視に必要な情報。ホストは Sterling Connect:Direct が実行されているホスト名または IP アドレスです。ポートは Sterling Connect:Direct の通信ポートを指定します。コマンドで区切るにより、複数のホスト名/IP アドレスとポートの組み合わせの指定が可能です。このパラメーターは必須です。	<p>ホスト名/IPアドレス;nnnn            ホスト名 - Sterling Connect:Direct ホスト・コンピューター名。            IP アドレス - Sterling Connect:Direct を実行するマシンの IP アドレス:            nnn.nnn.nnn.nnn (IPv4)            または            nnnn:nnnn:nnnn:nnnn:nnnn:n            nnn:nnnn:nnnn (IPv6)            ポート - Sterling Connect:Direct の通信ポートを指定。フォーマットは nnnn (10 進数) です。ホスト名/IP アドレスとポートは、セミコロンで区切られます:            msdallas-dt;1364            複数の アドレス/ホスト名とポートの組み合わせの指定が可能です。(コマンドで区切ってください。):            10.23.107.5;1364,            fe00:0:0:2014::7;1364,            msdallas-dt;1364            IP アドレスおよびホスト名の指定に関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』を参照してください。</p>
tcp.api.inactivity.timeout	これは、クライアント・プログラムからのコマンドが未受信の時に、CMGR が終了する前に待機する最大時間です。	0 ~ 86399 (23 時間 59 分 59 秒) 値は秒単位です。デフォルトは <b>0</b> です。これはタイムアウトが発生しないことを意味します。
tcp.max.time.to.wait	TCP/IP 使用時に、ローカル・ノードがリモート・ノードからのメッセージを待機する最大時間。この待機時間が終了すると、その Process はタイマー・キューに移動され、Sterling Connect:Direct はリモート・ノードとのセッションの再確立を試みます。0 に設定されると、待機時間はオペレーティング・システムによって制限されない限り無制限となります。	0 ~ 10000。 値は秒単位です。デフォルト値は <b>180</b> です。



## ローカル・ノードの TCP/IP 設定の更新

**tcp.ip.default** レコードは、リモート・ノードが IP アドレスによって指定された時に使用するデフォルト情報を定義します。**tcp.ip.default** レコード・パラメーター は、以下の表で説明されています。

パラメーター	説明	値
conn.retry.stwait	接続障害発生直後からの各接続再試行間の待機時間。フォーマットは <b>hh.mm.ss</b> で、 <b>hh</b> は時間、 <b>mm</b> は分、 <b>ss</b> は秒をそれぞれ指定します。	最大値はクロック・フォーマットの最高値 23.59.59 に制限されます。 デフォルトは <b>00.00.30</b> 、すなわち 30 秒です。
conn.retry.stattempts	接続障害が発生した後に、接続試行を行う回数。	0 ~ 9999。 デフォルトは <b>6</b> です。
conn.retry.ltwait	すべての short-term 再試行が終了した後の、各 long-time 再試行間の待機時間。フォーマットは <b>hh.mm.ss</b> で、 <b>hh</b> は時間、 <b>mm</b> は分、 <b>ss</b> は秒をそれぞれ指定します。	0 ~ 23.59.59 デフォルトは <b>00.10.00</b> 、すなわち 10 分です。
conn.retry.ltattempts	すべての short-term 再試行の終了後に、接続試行を行う回数。	0 ~ 9999。 デフォルトは <b>6</b> です。
comm.bufsize	リモート・ノードとのデータ送受信におけるバッファ・サイズ。	TCP/IP での値には制限はありません (2,147,483,623 まで)。 LU6.2 では、最大 32000 です。 デフォルトは <b>16000</b> バイトです。
conn.retry.exhaust.action	指定された short-term および long-term の再試行がすべて終了した後に取られるアクション。	<u>hold</u>   delete hold - すべての再試行終了後に Process を「Held in Error」ステータスで、保留キューに配置します。これはデフォルト値です。 delete - TCQ (伝送制御キュー) から Process を削除します。
tcp.max.time.to.wait	TCP/IP 使用時にローカル・ノードがリモート・ノードからのメッセージを待機する最大時間。この待機時間が終了すると、その Process はタイマー・キューに移動され、Sterling Connect:Direct はリモート・ノードとのセッションの再確立を試みます。	0 ~ 10,000。 値は秒単位です。 デフォルト値は <b>180</b> です。 0 に設定されると、待機時間はオペレーティング・システムによって制限されない限り無制限となります。

パラメーター	説明	値
runstep.max.time.to.wait	リモートの実行ステップの完了を待機する最大時間。リモートの実行ステップには、リモートでの run task、run job、または submit ステートメントが含まれます。この待機時間は、tcp.max.time.to.wait パラメーターで指定される待機時間とは異なります。runstep.max.time.to.wait を使用することにより、リモート・ステップの完了に tcp.max.time.to.wait での指定値よりも長い時間を要する場合に、Process がエラーとなることを防止します。	0 ~ 10000。 値は秒単位です。 デフォルト値は <b>0</b> です。
contact.name	管理者またはオペレーターの名前。	名前
contact.phone	Sterling Connect:Direct 管理者またはオペレーターの電話番号。	電話番号
descrip	レコードの一部として含まれるコメント。	制限なしのSTRING
sess.total	全ノードとローカル・ノード間の最大の同時接続数。 api.max.connects および sess.total の合計が、使用可能なファイル記述子の数を超えることはできません。この値はシステムに依存します。	0 ~ 999。 1 ~ 3 桁の数字。 デフォルトは <b>255</b> です。
sess.pnode.max	ローカル・ノードがセッションのイニシエーターである場合の最大の同時接続数。	0 ~ 999。 デフォルトは <b>255</b> です。
sess.snode.max	ローカル・ノードがセッションの 2 次ノードである場合の最大の同時接続数。	0 ~ 999。 デフォルトは <b>255</b> です。
sess.default	セッション・マネージャーを開始するデフォルト・セッション・クラス。Process は、指定されたクラス、またはより上位の任意のセッション・クラスで実行されます。このパラメーターの値は、local.node レコード内の対応する値をオーバーライドします。	1 ~ 50。 デフォルトは <b>1</b> です。
pacing.send.delay	リモート・ノードへの send 操作間での待機時間です。1 つのパケットの最後と、それに続くパケットの最初との間の時間をミリ秒単位の 10 進数で表します。時間ベースのペーシングは、ネットワーク・トラフィックの要因とはなりません。  このパラメーターの値は、LU6.2 接続に対しては無効です。	<i>nnn</i> この数値のサイズ制限はありません。 デフォルトは <b>0</b> です。これはこのタイプのペーシングがないことを示しています。

パラメーター	説明	値
pacing.send.count	<p>リモート・ノードからのペーシング応答を自動的に待機する状態に移行する前に実行される send 操作の数。</p> <p>このパラメーターの値は、LU6.2 接続に対しては無効です。</p>	<p>この値のサイズ制限はありません。デフォルトは <b>0</b> です。これはこのタイプのペーシングがないことを示しています。</p>

## リモート・ノード接続情報の更新

リモート・ノード接続情報レコードには、リモート・ノード情報が含まれます。必要に応じて、これらのパラメーターを更新して、リモート・ノード接続に使用するデフォルト値を定義してください。新しく定義するリモート・ノードに対しては、それぞれ新しいパラメーター・セットを追加してください。以下は、リモート・ノード接続パラメーターを表しています。

パラメーター	説明	値
alternate.comminfo	<p>Sterling Connect:Direct/Plex z/OS と いった、IP アドレスが複数存在するシステムでの、ネットマップがチェックされるセッション確立のサポートを提供します。このパラメーターを使用して、複数 IP アドレス環境の一部である IP アドレスまたは ホスト名をすべてリストします。</p> <p>Sterling Connect:Direct/Plex では、Sterling Connect:Direct/Plex マネージャーとは異なる IP アドレスを持つ各 Sterling Connect:Direct サーバーのアドレスが、このリストに含まれる必要があります。開始ノードの IP アドレスが <b>comm.info</b> パラメーターで指定された IP アドレスと一致しない場合には、<b>alternate.comminfo</b> パラメーター内の他の有効な IP アドレスが照合されます。</p> <p>IP アドレスおよびホスト名の指定に関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』を参照してください。</p>	<p>ホスト名/IP アドレスまたは *</p> <p>ホスト名 - IP アドレスに関連付けられているホスト名。例えば、</p> <pre>:alternate.comminfo=hops (hops はローカル・ドメインでのマシン)</pre> <pre>:alternate.comminfo=hops.csg.sterco mm.com (完全修飾ホスト名)</pre> <p>IP アドレス - Sterling Connect:Direct を実行しているマシンの IP アドレス (IPv4 または IPv6 フォーマット)</p> <pre>nnn.nnn.nnn.nnn (IPv4) または</pre> <pre>nnnn:nnnn:nnnn:nnnn:nnnn:nnnn:nn</pre> <pre>nn:nnnn (IPv6)</pre> <p>例えば、</p> <pre>:alternate.comminfo=10.23.107.5</pre> <pre>:alternate.comminfo=fe00:0:0:2014::7</pre> <p>複数の アドレス/ホスト名を指定するには、コンマ (,) で区切ってください。読みやすくするため、コンマの後にスペースを付加することができます。例えば、</p> <pre>10.23.107.5, fe00:0:0:2014::7, msdallas-dt</pre> <p>* - 任意の IP アドレスを受け入れません。これによって IP アドレス 検証が オフになります。</p> <p><b>注:</b> *.mydomain.com や、 myplex??.mydomain.com のような部分的パターン・マッチはサポートされていません。、</p>

パラメーター	説明	値
alt.comm.outbound	<p>アウトバウンド Process に使用される代替通信アドレス (通信パス)。このパラメーターは、複数の NIC カードを持つリモート・ノードの代替アドレスを提供します。ローカル・ノードが PNODE の時は、(comm.info パラメーターで指定される) プライマリー・アドレスに対する、最初の試行が失敗した場合には、代替アドレスが試されます。接続確立後に接続が失われた場合には、再試行メカニズムを通じた接続再確立試行は最初の接続と同じアドレスを使用をします。</p> <p>ローカル・ノード が SNODE の時は、代替アドレスはネットマップ・チェックで使用されます。</p>	<p>完全修飾ホスト名/IP アドレス ;nnnn</p> <p>ホスト名/IP アドレスおよびポートはセミコロン (;) で区切られます。代替通信パスのリストはコンマで区切られ、リストはトップダウンで処理されます。例えば、salmon;9400, 10.20.40.65;9500</p> <p>IP アドレスおよびホスト名の指定に関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』を参照してください</p>
comm.bufsize	<p>TCP/IP 接続での、リモート・ノードとのデータの送受信におけるバッファー・サイズ。</p>	<p>TCP/IP では、値の制限はありません ( 2,147,483,623 まで)。</p> <p>LU6.2 では、最大 32000。デフォルトは <b>65536</b> バイトです。</p>

パラメーター	説明	値
comm.info	<p>TCP/IP または LU6.2を使用した、リモート・ノードに対する接続要求を開始するために必要な情報。この情報は、ローカル Sterling Connect:Direct ノードがアウトバウンド要求を開始するのに使用するネットワーク・カードを参照します。</p> <ul style="list-style-type: none"> <li>• TCP/IP 接続の場合、ホスト名または IP アドレス、およびポート番号を指定してください。IP アドレスとポートを指定する場合には、パラメーターをセミコロン (;) で区切ってください。</li> <li>• LU6.2 接続の場合、リモート接続の SNA 構成プロファイルの名前として、プロファイル名を指定してください。Sterling Connect:Direct は、カスタマイズ手順の実行中に、デフォルト名 host11 を生成します。AIX SNA では、host11 はサイド情報プロファイル名を参照します。HP SNA、SunLink SNA および Brixton SNA では、host11 は構成ファイルと同じディレクトリーにある SNA プロファイル・ファイルを参照します。</li> </ul>	<p>TCP/IP 接続の場合、ホスト名または IP アドレス、およびポート番号を指定してください。</p> <p>デフォルト・ポートは <b>1364</b> です。</p> <p>LU6.2 接続の場合、最長8字までのプロファイル名を指定してください。</p> <p>IP アドレスおよびホスト名の指定に関する詳細については、『付録 B: IP アドレス、ホスト名、およびポートの指定』を参照してください。</p>
comm.transport	<p>リモート・ノードのトランスポート・プロトコル。</p>	<p>tcp   lu62   blklu62   udt33</p> <p>tcp - TCP/IP 接続</p> <p>lu62 - AIX SNA LU6.2 接続</p> <p>blklu62 - 他の LU6.2 接続</p> <p>udt33 - UDT 接続</p>
conn.retry.stwait	<p>接続障害発生直後からの各再試行間の待機時間。フォーマットは <b>hh.mm.ss</b> で、<b>hh</b> は時間、<b>mm</b> は分、<b>ss</b> は秒をそれぞれ指定します。</p>	<p>最大値はクロック・フォーマットの最高値 23.59.59 に制限されます。</p> <p>デフォルトは <b>00.00.30</b>、すなわち 30 秒です。</p>
conn.retry.stattempts	<p>接続障害が発生した後に、接続再試行を行う回数。</p>	<p>0 ~ 9999</p> <p>デフォルトは <b>6</b> です。</p>

パラメーター	説明	値
conn.retry.ltwait	すべての short-term 再試行が終了した後の、各 long-time 再試行間の待機時間。フォーマットは <b>hh.mm.ss</b> で、 <b>hh</b> は時間を、 <b>mm</b> は分を、 <b>ss</b> は秒をそれぞれ指定します。	0 ~ 23.59.59 デフォルトは <b>00.10.00</b> 、すなわち 10 分です。
conn.retry.ltempts	すべての short-term 再試行が終了した後に、接続再試行を行う回数。	0 ~ 9999。 デフォルトは <b>6</b> です。
conn.retry.exhaust.action	指定された short-term および long-term の接続再試行がすべて終了した後に取られるアクション。	hold   delete hold - すべての再試行終了後に Process を「Held in Error」ステータスで、保留キューに配置します。これはデフォルト値です。 delete - TCQ (伝送制御キュー) から Process を削除します。
contact.name	Sterling Connect:Direct 管理者またはオペレーターの名前。	名前
contact.phone	Sterling Connect:Direct 管理者またはオペレーターの電話番号。	電話番号
descrip	レコードの一部として含まれるコメント。	制限なしのSTRING
pacing.send.count	リモート・ノードからのペーシング応答を自動的に待機する状態に移行する前に実行される、send 操作の数。このパラメーターの値は、LU6.2 接続に対しては無効です。	この値のサイズ制限はありません。デフォルトは <b>0</b> です。これはこのタイプのペーシングがないことを示しています。
pacing.send.delay	リモート・ノードへの send 送信間での待機時間です。1つのパケットの最後と、それに続くパケットの最初との間の時間をミリ秒単位の 10 進数で表します。時間ベースのペーシングは、ネットワーク・トラフィックの要因とはなりません。このパラメーターの値は、LU6.2 接続に対しては無効です。	<i>nnn</i> この数値のサイズ制限はありません。デフォルトは <b>0</b> です。これはこのタイプのペーシングがないことを示しています。

パラメーター	説明	値
runstep.max.time.to.wait	リモートの実行ステップの完了を待機する最大時間。リモートの実行ステップには、リモートでの run task、run job、または submit ステートメントが含まれます。この待機時間は、tcp.max.time.to.wait パラメーターで指定される待機時間とは異なります。runstep.max.time.to.wait を使用することにより、リモートのステップの完了に tcp.max.time.to.wait での指定値よりも長い時間を要する場合に、Process がエラーとなることを防止します。値は秒単位です。	0 ~ 10000 デフォルト値は <b>0</b> です。
sess.total	全ノードとローカル・ノード間の最大の同時接続数です。 api.max.connects および sess.total の合計が、使用可能なファイル記述子の数を超えることはできません。この値はシステムに依存します。	0 ~ 999 1 ~ 3 桁の数字。 デフォルトは <b>255</b> です。
sess.pnode.max	ローカル・ノードがセッションのイニシエーターである場合の最大の同時接続数です。PNODE セッション数は、セッション総数を超えることはできません。 sess.pnode.max が sess.total よりも大きい場合、sess.pnode.max の値は自動的に sess.total の値まで切り捨てられます。	0 ~ 999。 デフォルトは <b>255</b> です。
sess.snode.max	ローカル・ノードがセッションの 2 次ノードである場合の最大の同時接続数です。 SNODE セッション数は、セッション総数を超えることはできません。 sess.snode.max が sess.total よりも大きい場合、自動的に sess.total の値に変更されます。	0 ~ 999。 デフォルトは <b>255</b> です。
tcp.crc	リモート・ノードでの TCP/IP Process の CRC 機能をオンまたはオフにします。	y   n デフォルトは <b>n</b> です。

---

## アクセス情報ファイルの保守

ユーザー認証情報ファイル、強固なアクセス制御ファイル、プログラム・ディレクトリー、ローカルおよびリモート・ユーザー情報レコード、およびセキュリティー出口を通して、Sterling Connect:Direct へのアクセスを制御することができます。

---

### ユーザー認証情報ファイル

ユーザーが Sterling Connect:Direct にアクセスし、Sterling Connect:Direct コマンドおよびステートメントを使用するためには、**userfile.cfg** と呼ばれるユーザー認証情報ファイル中の各ユーザー ID に対するレコードを定義する必要があります。ユーザー ID が、ローカル・ユーザー情報レコードのキーになります。これは、ローカル・システムでの有効なユーザー ID であるとともに、一意的な値である必要があります。

---

**注:** ローカル・ユーザーのソフトウェアへのアクセスを無効にするには、ローカル・ユーザー情報レコードを削除またはコメントアウトしてください。

---

ユーザー ID としてアスタリスク (\*) 指定することによって、汎用ユーザー ID の作成が可能です。ユーザーが特定のローカル・ユーザー情報レコードを持たない場合には、デフォルトでは、ユーザー認証はこの汎用レコード中に指定された値で行われます。汎用ローカル・ユーザー情報レコードが定義されておらず、ユーザーに対する特定のローカル・ユーザー情報レコードも定義されていない場合、そのユーザーは、Sterling Connect:Direct を使用できません。

Sterling Connect:Direct は、リモート・ユーザー ID を Sterling Connect:Direct がインストールされているロケーションでの有効なローカル・ユーザー ID へと変換するために、リモート・ユーザー情報レコードを任意に使用する可能性があります。受信 Process に `snodeid` パラメーターがコーディングされていない場合には、Sterling Connect:Direct は、このプロキシ関係を使用して、リモート・ユーザーの Sterling Connect:Direct コマンドおよびステートメント実行権限を決定します。

Sterling Connect:Direct for UNIX では、アスタリスク (\*) 文字を使用して、ローカル・ユーザー ID に対するリモート・ユーザー ID のマッピングを容易にする汎用マッピングを設定します。アスタリスクはノード名またはホスト名に相当します。例えば、`*@ノード名` と指定することで、リモート・ユーザー ID を、あるノード名上にある全ユーザー ID にマッピング、`id@*` と指定することで、全ノード上の特定のユーザー ID にマッピング、または、`*@*` と指定することで、全ノード上の全ユーザーにマッチさせることができます。



以下の表は、リモート・ユーザー ID からローカル・ユーザー ID への特殊文字を使用したマッピングのサンプルを表示しています。

リモート・ユーザー ID	@	リモート・ノード名	マッピング先	ローカル・ユーザー ID	マッピング結果
user	@	*	=	test02	全リモート・ノード上のリモート・ユーザー ID 「user」が、ローカル・ユーザー ID test02 にマッピングされます。
*	@	mvs.node3	=	labs3	リモート・ノード mvs.node3 上の全リモート・ユーザー ID がローカル・ユーザー ID labs3 にマッピングされます。
*	@	*	=	vip01	全リモート・ノード上の全リモート・ユーザー ID がローカル・ユーザー ID vip01 にマッピングされます。

スクリプト・ベースのカスタマイズ手順で、全レコードを生成することができます。もしくは、1 つか 2 つのレコードのみを生成して、その後テキスト・エディターを使って追加レコードを生成することも可能です。カスタマイズ後に、パラメーターを変更したい場合も考えられます。必要に応じて、**cdcust** を使用して、新しいユーザー・ファイルの作成、もしくは、テキスト・エディターを使用して、ファイルの修正を行ってください。

### ユーザー認証ファイルのサンプル

以下のサンプルは、ユーザー認証ファイルを表示しています。このサンプルでは、SAM1 がリモート・ユーザー ID、MVS.SAM1.NODE がリモート・ノード名、sam がローカル UNIX ユーザー ID となっています。

```

SAM1@MVS.SAM1.NODE:¥
:local.id=sam:¥
:pstmt.upload=y:¥
:pstmt.upload_dir=/home/qatest/username/ndm/uploaddir:¥
:pstmt.download=y:¥
:pstmt.download_dir=/home/qatest/username/ndm/downloaddir:¥
:pstmt.run_dir=/home/qatest/username/ndm/rundir:¥
:pstmt.submit_dir=/home/qatest/username/ndm/submitdir:¥
:descrip=: sam:¥
:admin.auth=y:¥
:pstmt.copy.ulimit=y:¥
:pstmt.upload=y:¥
:pstmt.upload_dir=/home/qatest/username/ndm/uploaddir:¥
:pstmt.download=y:¥
:pstmt.download_dir=/home/qatest/username/ndm/downloaddir:¥
:pstmt.run_dir=/home/qatest/username/ndm/rundir:¥
:pstmt.submit_dir=/home/qatest/username/ndm/submitdir:¥
:name=:¥
:phone=:¥
:descrip=:
:cmd.s+conf=n:

```

## ローカル・ユーザー情報レコード・フォーマットの更新

ローカル・ユーザー・レコード `userid` は、各ユーザー ID のデフォルト値を定義します。ローカル・ユーザー情報レコード内の大半のパラメーターは、以下の値を取ります。

- ◆ `y` - ユーザーがその機能を実行できることを表します。`process` コマンドおよび `select statistics` コマンドの場合には、ユーザーは Process に影響を与えると同時に、そのユーザー ID によって所有される統計を表示することができます。
- ◆ `n` - ユーザーがその機能を実行できないことを表します。
- ◆ `a` - ユーザーが、全ユーザーによって所有される Process に対してコマンドを実行することができ、全ユーザーの統計レコードを生成することができることを表します。

以下の表は、ローカル・ユーザー情報パラメーターを定義しています。デフォルト値には下線が引かれています。

パラメーター	説明	値
<code>admin.auth</code>	ユーザーにおける管理権限の有無を決定します。 <code>y</code> に設定されると、ユーザーはすべてのコマンドをデフォルトで実行することができますが、デフォルトをオーバーライドする特定のコマンド・パラメーターも存在します。 <code>n</code> に設定された場合、特定のコマンド・パラメーターは個別に付与される必要があります。	<code>y   <u>n</u></code> <code>y</code> - ユーザーに管理権限があります。 <code>n</code> - ユーザーに管理権限はありません。 デフォルトは <code>n</code> です。
<code>cmd.chgproc</code>	ユーザーによる <b>change process</b> コマンドの実行の可否を決定します。 「 <code>y</code> 」の値が指定されると、ユーザーは、そのユーザーが所有するターゲットに対するコマンドの実行が可能となります。一方、「 <code>a</code> 」が指定されると、ユーザーは、全ユーザーが所有するターゲットに対するコマンドの実行が可能となります。	<code>y   <u>n</u>   a</code> <code>y</code> - ユーザーによるコマンドの実行を許可します。 <code>n</code> - ユーザーによるコマンドの実行を禁止します。デフォルトは <code>n</code> です。 <code>a</code> - ユーザーによる、全ユーザーによって所有されるターゲットに対するコマンドの実行を許可します。
<code>cmd.delproc</code>	ユーザーによる <b>delete process</b> コマンドの実行の可否を決定します。 「 <code>y</code> 」の値が指定されると、ユーザーは、そのユーザーが所有するターゲットに対するコマンドの実行が可能となります。一方、「 <code>a</code> 」が指定されると、ユーザーは、全ユーザーが所有するターゲットに対するコマンドの実行が可能となります。	<code>y   <u>n</u>   a</code> <code>y</code> - ユーザーによるコマンドの実行を許可します。 <code>n</code> - ユーザーによるコマンドの実行を禁止します。デフォルトは <code>n</code> です。 <code>a</code> - ユーザーによる、全ユーザーによって所有されるターゲットに対するコマンドの実行を許可します。

パラメーター	説明	値
cmd.flsproc	ユーザーによる <b>flush process</b> コマンドの実行の可否を決定します。 「y」の値が指定されると、ユーザーは、そのユーザーが所有するターゲットに対するコマンドの実行が可能となります。一方、「a」が指定されると、ユーザーは、全ユーザーが所有するターゲットに対するコマンドの実行が可能となります。	y   n   a y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。 a - ユーザーによる、全ユーザーによって所有されるターゲットに対するコマンドの実行を許可します。
cmd.selproc	ユーザーによる <b>select process</b> コマンドの実行の可否を決定します。 「y」の値が指定されると、ユーザーは、そのユーザーが所有するターゲットに対するコマンドの実行が可能となります。一方、「a」が指定されると、ユーザーは、全ユーザーが所有するターゲットに対するコマンドの実行が可能となります。	y   n   a y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。 a - ユーザーによる、全ユーザーによって所有されるターゲットに対するコマンドの実行を許可します。
cmd.viewproc	ユーザーによる <b>view process</b> コマンドの実行の可否を決定します。 「y」の値が指定されると、ユーザーは、そのユーザーが所有するターゲットに対するコマンドの実行が可能となります。一方、「a」が指定されると、ユーザーは、全ユーザーが所有するターゲットに対するコマンドの実行が可能となります。	y   n   a y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。 a - ユーザーによる、全ユーザーによって所有されるターゲットに対するコマンドの実行を許可します。
cmd.selstats	ユーザーによる <b>select statistics</b> コマンドの実行の可否を決定します。 「y」の値が指定されると、ユーザーは、そのユーザーが所有するターゲットに対するコマンドの実行が可能となります。一方、「a」が指定されると、ユーザーは、全ユーザーが所有するターゲットに対するコマンドの実行が可能となります。	y   n   a y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。 a - ユーザーによる、全ユーザーによって所有されるターゲットに対するコマンドの実行を許可します。
cmd.stopndm	ユーザーによる <b>stop</b> コマンドの実行の可否を決定します。	y   n y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。
cmd.s+conf	ユーザーによる IBM Sterling Control Center や Java API といったネットワーク・クライアントからの Sterling Connect:Direct Secure Plus を構成するためのコマンドの実行の可否を決定します。 注: このパラメーターは、spadmin.sh や spcli.sh といったローカル・ツールに対しては影響を与えません。	y   n y - ユーザーによるコマンドの実行を許可します。デフォルトは y です。 n - ユーザーによるコマンドの実行を禁止します。

パラメーター	説明	値
cmd.submit	ユーザーによる <b>submit process</b> コマンドの実行の可否を決定します。	y   n y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。
cmd.trace	ユーザーによる <b>trace</b> コマンドの実行の可否を決定します。	y   n y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。
pstmt.crc	ユーザーに、Process ステートメントでキーワード「CRC」を使用するための初期設定のオーバーライドを許可します。	y   n y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。
descrip	管理者によるレコードへの説明用のコメントの追加を許可します。	制限なしのテキスト・ストリング
name	ユーザーの名前。	ユーザー名
phone	ユーザーの電話番号。	ユーザーの電話番号
pstmt.copy	ユーザーによる <b>copy</b> ステートメントの実行の可否を決定します。	y   n y - ユーザーによるコマンドの実行を許可します。 n - ユーザーによるコマンドの実行を禁止します。デフォルトは n です。
pstmt.copy.ulimit	copy 操作時に、ユーザー出力ファイルのサイズ制限を超えた場合に取られるアクション。このパラメーターの値は、初期化パラメーター・ファイルの ulimit パラメーターに相当する値をオーバーライドします。	y   n   nnnnnnnn   nnnnnnnnK   nnnnnnnnM   nnnnG y - ユーザー・ファイルのサイズ制限を有効とします。copy 操作時に、この制限を超えると、その操作は失敗します。 n - 制限を無視します。デフォルトは n です。 nnnnnnnn、nnnnnnnnK、nnnnnnnnM、または nnnnG - K は 1024 バイトを表します。 M は 1048576 バイトを表します。 G は 1073741824 バイトを表します。 指定できる最大値は 1 TB です。

パラメーター	説明	値
pstmt.upload	ユーザーによる、このローカル・ノードからのファイルの送信の可否を決定します。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。	Y n y - ユーザーによるファイル送信を許可します。デフォルトは y です。 n - ユーザーによるファイル送信を禁止します。
pstmt.upload_dir	ユーザーがファイルを送信できる元のディレクトリー。このパラメーターに対して値が設定されると、このディレクトリーもしくはそのサブディレクトリーのみからのファイル送信が可能となります。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。このパラメーターが定義される場合、Copy ステートメント内のファイル名は、このディレクトリーに関連付けられている必要があります。絶対パス名の使用が可能ですが、パスはこの指定と一致している必要があります。	ディレクトリー・パス名
pstmt.download	ユーザーによる、このローカル・ノードへのファイルの受信の可否を決定します。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。	Y n y - ユーザーによるファイル受信を許可します。デフォルトは y です。 n - ユーザーによるファイル受信を禁止します。
pstmt.download_dir	ユーザーがファイルを受信できる先のディレクトリー。このパラメーターに対して値が設定されると、このディレクトリーもしくはそのサブディレクトリーのみへのファイル受信が可能となります。設定されない場合は、任意のディレクトリーへのファイル受信が可能です。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。	ディレクトリー・パス名
pstmt.run_dir	Sterling Connect:Direct がインストールされ、ユーザーが <b>run job</b> ステートメントや <b>run task</b> ステートメントで実行するプログラムおよびスクリプトが含まれるディレクトリー。指定ディレクトリー外でのプログラム実行またはスクリプト実行の試みはいずれも失敗します。  UNIX 制限付きシェルは、pstmt.run_dir に含まれるコマンドのみにユーザーを制限することによる、強化されたセキュリティーを提供します。ユーザーが pstmt.run_dir を指定しない場合、コマンドは Bourne シェルで開始されます。  実行ディレクトリーでの特殊文字の使用を制限するには、必ず <b>restrict:cmd</b> 初期化パラメーターを Y に設定してください。 <b>restrict:cmd</b> 初期化パラメーターの指定に関する詳細については、24 ページの『 <b>実行ディレクトリー内の特殊文字の使用制限</b> 』を参照してください。	ディレクトリー・パス名

パラメーター	説明	値
pstmt.runjob	ユーザーによる <b>run job</b> ステートメントの実行の可否を指定します。	y   n y - ユーザーによるステートメントの実行を許可します。 n - ユーザーによるステートメントの実行を禁止します。デフォルトは <b>n</b> です。
pstmt.runtask	ユーザーによる <b>run task</b> ステートメントの実行の可否を指定します。	y   n y - ユーザーによるステートメントの実行を許可します。 n - ユーザーによるステートメントの実行を禁止します。デフォルトは <b>n</b> です。
pstmt.submit	ユーザーによる <b>submit</b> ステートメントの実行の可否を指定します。	y   n y - ユーザーによるステートメントの実行を許可します。 n - ユーザーによるステートメントの実行を禁止します。デフォルトは <b>n</b> です。
pstmt.submit_dir	ユーザーが Process の実行依頼を行うことができる元のディレクトリー。これは Process 内の <b>submit</b> に対して有効です。	ディレクトリー・パス名
snode.ovrd	ユーザーによる、 <b>submit</b> コマンド、 <b>process</b> ステートメントおよび <b>submit</b> ステートメントでの、 <b>snodeid</b> パラメーターのコーディングの可否を指定します。	y   n y - ユーザーによる <b>snodeid</b> パラメーターのコーディングを許可します。 n - ユーザーによる <b>snodeid</b> パラメーターのコーディングを禁止します。デフォルトは <b>n</b> です。
pstmt.crc	ユーザーに Process ステートメントで CRC 検査の使用を指定する権限を与えます。このパラメーターを y に設定することで、ユーザーは初期化パラメーターまたはネットワーク・マップ構成ファイル内の初期設定をオーバーライドすることが可能となります。	y   n y - ユーザーによる Process ステートメントでの CRC 検査の指定を許可します。 n - ユーザーによる Process ステートメントでの CRC 検査の指定を禁止します。デフォルトは <b>n</b> です。

リモート・ユーザー情報レコードとローカル・ユーザー情報レコードで同一のパラメーターが指定された場合、それが NULL 値である場合を除いて、リモート・ユーザー情報レコードのパラメーターが優先されます。リモート・レコードで NULL 値が指定されている場合は、ローカル・ユーザー・レコードが優先されます。

## リモート・ユーザー情報レコードの更新

リモート・ユーザー情報レコードには、レコードのキーとなる、リモート・ユーザー ID およびリモート・ノード名が含まれます。local.id パラメーターは、このユーザーに対するローカル・ユーザー情報レコードを指定します。リモート・ユーザーのローカル・ユーザー情報レコードを作成する必要

があります。

**注:** リモート・ユーザーによる Sterling Connect:Direct 使用を禁止するには、リモート・ユーザーが Process で SNODEID パラメーターを指定している場合を除き、リモート・ユーザー情報を削除もしくはコメントアウトしてください。

リモート・ユーザー情報レコードは、リモート・ユーザー ID@リモート・ノード名のフォーマットとなっています。これは、リモート・ユーザーとして定義された、ユーザーおよびリモート・ノード名のペアを指定します。この値はレコードのキーとなるので、一意的な値である必要があります。このローカル・ノードと通信するリモート・ノード上の各ユーザーに対して、リモート・ユーザー情報レコードを作成してください。

以下は、リモート・ユーザー情報レコードのパラメーターです。

パラメーター	説明	値
local.id	セキュリティ・チェックでリモート・ユーザーに代わって使用するローカル・ユーザー ID。local.id パラメーターはローカル・ユーザー情報レコードを識別できる必要があります。	ローカル・ユーザー ID
pstmt.copy	ユーザーによる <b>copy</b> ステートメントの実行の可否を決定します。	y   n y - ユーザーによるステートメントの実行を許可します。 n - ユーザーによるステートメントの実行を禁止します。デフォルトは n です。
pstmt.upload	ユーザーによるこのローカル・ノードからのファイル送信の可否を決定します。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。	y   n y - ユーザーによるファイル送信を許可します。デフォルトは y です。 n - ユーザーによるファイル送信を禁止します。
pstmt.upload_dir	ユーザーがファイルを送信できる元のディレクトリー。このパラメーターに対して値が設定されると、このディレクトリーもしくはそのサブディレクトリーのみからのファイル送信が可能となります。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。このパラメーターが定義される場合、Copy ステートメント内のファイル名は、このディレクトリーに関連付けられている必要があります。絶対パス名の使用が可能ですが、パスはこの指定と一致している必要があります。	ディレクトリー・パス名
pstmt.download	ユーザーによるこのローカル・ノードへのファイル受信の可否を決定します。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。	y   n y - ユーザーによるファイル受信を許可します。デフォルトは y です。 n - ユーザーによるファイル受信を禁止します。

パラメーター	説明	値
pstmt.download_dir	ユーザーがファイルを受信できる先のディレクトリー。このパラメーターに対して値が設定されると、このディレクトリーもしくはそのサブディレクトリーのみへのファイル受信が可能となります。設定されない場合は、任意のディレクトリーへのファイル受信が可能です。ファイル・オープン出口が使用されている場合、このパラメーターはこの出口へ渡されますが、強制的ではありません。	ディレクトリー・パス名
pstmt.run_dir	ユーザーが run job ステートメントや run task ステートメントで実行できるプログラムおよびスクリプトが含まれるディレクトリー。指定ディレクトリー外でのプログラム実行またはスクリプト実行の試みはいずれも失敗します。  実行ディレクトリーでの特殊文字の使用を制限するには、必ず <b>restrict:cmd</b> 初期化パラメーターを Y に設定してください。 <b>restrict:cmd</b> 初期化パラメーターの指定に関する詳細については、24 ページの『 <b>実行ディレクトリー内の特殊文字の使用制限</b> 』を参照してください。	ディレクトリー・パス名
pstmt.submit_dir	ユーザーが Process の実行依頼を行うことができる元のディレクトリー。これは Process 内の <b>submit</b> に対して有効です。	ディレクトリー・パス名
pstmt.runjob	ユーザーによる <b>run job</b> ステートメントの実行の可否を指定します。	y   n y - ユーザーによるステートメントの実行を許可します。 n - ユーザーによるステートメントの実行を禁止します。デフォルトは n です。
pstmt.runtask	ユーザーによる <b>run task</b> ステートメントの実行の可否を指定します。	y   n y - ユーザーによるステートメントの実行を許可します。 n - ユーザーによるステートメントの実行を禁止します。デフォルトは n です。
pstmt.submit	ユーザーによる <b>submit</b> ステートメントの実行の可否を指定します。	y   n y - ユーザーによるステートメントの実行を許可します。 n - ユーザーによるステートメントの実行を禁止します。デフォルトは n です。
descrip	レコードへの説明用のコメントの追加を許可します。	テキスト・ストリング



## 強固なアクセス制御ファイルの更新

一般ユーザーが、Sterling Connect:Direct for UNIX を通して root アクセス権を得ることを防止する対策として、インストール時に **sysacl.cfg** と呼ばれる強固なアクセス制御ファイルがディレクトリー `d_dir/ndm/SACL/` に作成されます。デフォルトでは、一般ユーザーは Sterling Connect:Direct for UNIX を通しての root アクセスはできません。Sterling Connect:Direct for UNIX を通して一般的な root ユーザー・アクセスを与えたい場合には、**sysacl.cfg** ファイルにアクセスし、これを更新する必要があります。

注: Sterling Connect:Direct for UNIX を通して root アクセスを制限したくない場合であっても、**sysacl.cfg** ファイルは必須です。このファイルが削除されている、または破損している場合には、全ユーザーに対する Sterling Connect:Direct for UNIX へのアクセスが拒否されます。

**sysacl.cfg** ファイルのレイアウトは、**userfile.cfg** ファイルのユーザー部分と同一になります。あるユーザーにおける **sysacl.cfg** ファイルでの値の設定は、**userfile.cfg** ファイルでのそのユーザーの値をオーバーライドします。

**root:deny.access** パラメーターは、**sysacl.cfg** ファイルで指定され、Sterling Connect:Direct への root アクセスを許可、拒否、もしくは制限します。このパラメーターは必須です。

**root:deny.access** パラメーターに対して、以下の値を指定することができます。

パラメーター	説明	値
deny.access	IBM Sterling Connect:Direct へのルート・アクセスを許可、拒否、もしくは制限します。	<p>y   n   d</p> <p>y - いかなる Process も root 権限を獲得することはできません。</p> <p>n - PNODE の Process は root 権限を獲得できますが、SNODE の Process は root 権限を獲得できません。これはデフォルト値です。</p> <p>d - すべての Process による root 権限の獲得が可能です。</p>

**sysacl.cfg** ファイル内で、ユーザーに対し **root:deny.access** パラメーターが定義されていたために、そのユーザーがアクセスを拒否された場合、メッセージが記録され、セッションが終了されます。ユーザーが制限付き ID を持っていた場合、情報メッセージが記録されます。

## シャドー・パスワードの自動検出

UNIX オペレーティング・システムの一部のバージョンではシャドー・パスワード・ファイルが使用可能となっているため、これが使用できる場合には、Sterling Connect:Direct for UNIX はシャドー・パスワードの使用を自動的に検出します。

## プログラム・ディレクトリーへのアクセス制限

プログラム・ディレクトリーは、指定されたスクリプトおよびコマンドへのアクセス制限を行うことで、run task および run job プロセス・ステートメントに対する拡張セキュリティーを提供します。指定ディレクトリー外でのプログラムの実行またはスクリプトの実行の試みはいずれも失敗します。プログラム・ディレクトリーは、`pstmt.run_dir` パラメーターによって指定されます。プログラム・ディレクトリーが指定されると、UNIX 制限付きシェルが呼び出され、拡張セキュリティーが提供されます。プログラム・ディレクトリーが指定されない場合には、制限無しのコマンドの実行を行うため、通常の Bourne シェルが起動されます。

制限付きシェルは通常の Bourne シェルと非常に似ていますが、ユーザーによる以下の機能の実行を制限します。

- ◆ ディレクトリーの変更 (cd)
- ◆ PATH または SHELL 環境変数の変更
- ◆ スラッシュ (/) 文字を含むコマンド名の使用
- ◆ 出力のリダイレクト (> および >>)

制限付きシェルに関する詳細については、適切な UNIX マニュアルまたは UNIX セキュリティー・テキストブックを参照してください。

制限付きシェルは、以下のように定義される環境変数 HOME、IFS、PATH、および LOGNAME のみを使用して開始されます。

```
HOME=run_dir
IFS=空白文字(タブ、スペース、および改行)
PATH=/usr/rbin および run_dir
LOGNAME=ユーザーの UNIX ID
```

環境変数は親 Process から継承するものではないので、シェル環境変数を通じてスクリプトまたはコマンドに対してデータが渡されることはありません。制限付きのシェルは指定されたスクリプトおよびコマンドへのアクセスは制限しますが、スクリプトおよびコマンドの実行内容を制限するものではありません。例えば、*run\_dir* ディレクトリー内で実行されるシェル・スクリプトは、PATH の値を変更し、スラッシュ (/) 文字を含むコマンド名の実行が可能です。このため、システム管理者は、どのスクリプトおよびコマンドにユーザーがアクセスできるかを制御するとともに、ユーザーに *run\_dir* ディレクトリーまたは *run\_dir* ディレクトリー内のファイルに対する書き込み権限を与えないようにすることが重要となります。

## セキュリティー出口

初期化パラメーター・ファイル `initparm.cfg` のセキュリティー出口は、パスワード対応プログラムへのインターフェースを提供します。

この出口は、パスチケットを生成および検証します。また、他のパスワード対応プログラムもサポートしています。他のプログラムの例として、MVS ホスト上で使用可能、かつ、NETSP 製品を使用する UNIX AIX および OS/2 コンピューター上で IBM によってサポートもされている、RACF セキュリティー・システムの一部である PASSTICKET が挙げられます。

セキュリティー出口に関する詳細については、『第 3 章: 初期化パラメーター・ファイルの保守』を参照してください。



---

## クライアントおよびサーバー認証 キー・ファイルの保守

この章には、クライアントおよびサーバー認証キー・ファイルについての情報が含まれています。これらのキー・ファイルは、システムにインストールされている任意のテキスト・エディターでの編集が可能です。

---

### クライアントおよびサーバー認証キー・ファイル

Sterling Connect:Direct クライアント/サーバー・セキュリティは、Sterling Connect:Direct サーバーに存在する、パスワードに似たキー、および、そのサーバーと通信する各 API に存在する同一のキーに依存します。これらのキーはシステム管理者によって定義および調整されます。

keys.client と呼ばれるクライアント・キー・ファイルは API が常駐するノード上に存在します。サーバー・キー・ファイル keys.server はサーバーが常駐するノード上に存在します。キー・ファイルは、ディレクトリー `d_dir/security` に置かれています。

#### キー・ファイルのフォーマット

キー・ファイル内のレコードには、他の API またはサーバー・キー・ファイルのエントリーにマッチする、最大 4 つのキーを含めることができます。キー・ファイルは、必要に応じた数だけキー・ファイル・レコードを含むことが可能です。以下のサンプルは、キー・ファイル・エントリーのフォーマットを示しています。

hostname	MRLN SIMP key [key [key [key]
----------	-------------------------------

## キー・ファイル・パラメーターの更新

以下の表は、使用可能なキー・ファイル・パラメーターを説明しています。:

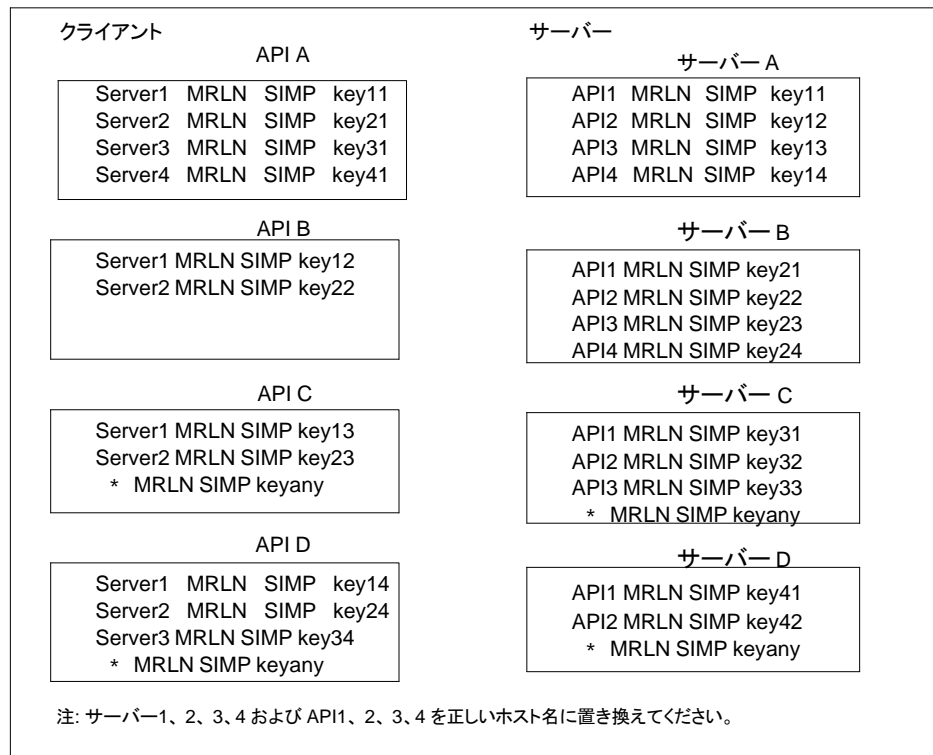
パラメーター	説明	値
hostname	通信を行いたいサーバーのホスト名、または自身のサーバーとの通信を許可する API のホスト名。ホスト名の後には 1 つ以上の空白文字が続きます。サーバー構成ファイル内のホスト名をアスタリスク (*) 文字で置き換えると、サーバーはキーがマッチする任意の API からの接続を受け入れます。1 つのファイルにつき、1 つのアスタリスクのみ使用が可能です。アスタリスクを使用したエントリーは、常に特定のホスト名の記載されたエントリーの後に置いてください。	1 ~ 16 文字。そのキー・ファイル内で一意的な値である必要があります。
MRLN SIMP	他のフィールドと 1 つ以上のスペースによって区切られた、必須文字ストリング。	文字ストリング
key	セキュリティー・キー。キーと SIMP は、1 つ以上のスペースで区切ります。	最大 22 文字長。A ~ Z、a ~ z、0 ~ 9、ピリオド (.)、スラッシュ (/) を使用できます。

## クライアント認証キー・ファイルのサンプル

以下の図では、クライアントの列に API キーのリスト、サーバーの列にサーバー・キーのリストを例示しています。

- ◆ API A には、key11、key21、key31、key41 が含まれています。サーバー A にも key11 エントリーがあるため、API A は key11 によって、サーバー A との通信が可能となります。API1 は API A が常駐するホスト名であり、Server1 はサーバー A が常駐するホスト名であることを必ず確認してください。
- ◆ API D には、key14、key24、key34 が含まれています。サーバー A にも key14 エントリーがあるため、API D は key14 によって、サーバー A との通信が可能となります。API4 は API D が常駐するホスト名であり、Server1 はサーバー A が常駐するホスト名であることを必ず確認してください。

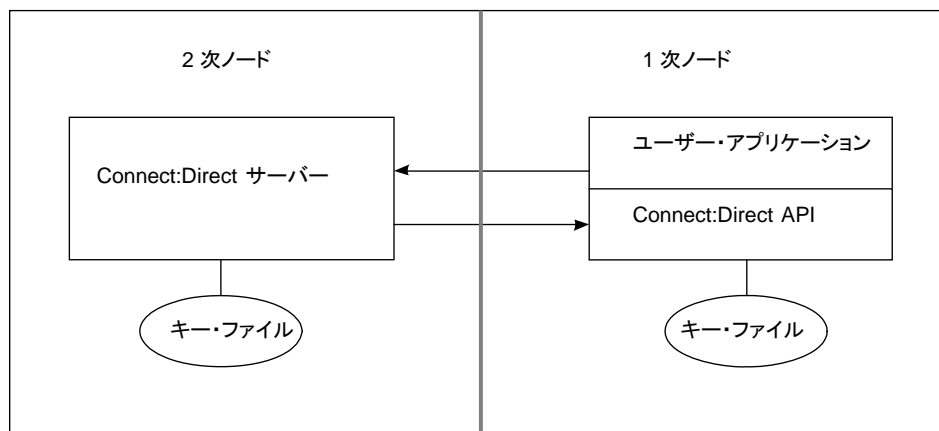
- ◆ API C はマッチするキーを通して、サーバー A およびサーバー B との通信が可能です。API C はまた、\* **MRLN SIMP keyany** 行を通してのみ、サーバー C およびサーバー D との通信が可能となります。



## 認証手順

Sterling Connect:Direct 認証手順では、ユーザーにシステムへのアクセス権があるかどうかを決定します。

Sterling Connect:Direct セキュリティーは、繰り返しログオンを要求することなく、各ユーザーIDを確実に識別することを目標としています。加えて、セキュリティは、無許可のユーザーによって認証手順がバイパスされることがないようにするため、すべての要求が必ず Sterling Connect:Direct API から発信されるよう設計されています。以下の図は、認証を実行するコンポーネントを表示しています。



### サーバー認証パラメーター

サーバー認証パラメーターは、**initparm.cfg** で指定されます。これらのファイルを修正するには、所有権および書き込み権限が必要となります。所有権はインストール手順の実行中に設定されます。

さらに、keys.server ファイルを含むディレクトリーは UNIX 権限が 0700 に、keys.server ファイルは UNIX 権限が 0600 に、それぞれ指定されている必要があります。これらのファイルを root が所有することはできません。

以下のサーバー認証パラメーターが、認証手順の実行中に CMGR によって使用されます。

パラメーター	説明
server.program	認証手順の実行中に使用されるサーバー・プログラム
server.keyfile	認証手順の実行中に使用されるキー・ファイル

### クライアント認証パラメーター

クライアント認証パラメーターは **ndmapi.cfg** で指定されます。これらのファイルを修正するには、所有権および書き込み権限が必要となります。所有権はインストール手順の実行中に設定されます。

さらに、keys.client ファイルを含むディレクトリーは UNIX 権限が 0700 に、keys.client ファイルは UNIX 権限が 0600 にそれぞれ指定されている必要があります。

以下のクライアント認証パラメーターが、認証手順の実行中に CLI/API によって使用されます。

パラメーター	説明
client.program	認証手順の実行中に使用されるクライアント・プログラム
client.keyfile	認証手順の実行中に使用されるキー・ファイル

---

## ファイアウォール・ナビゲーションの構成

ファイアウォール・ナビゲーションによって、自社または取引先のセキュリティー・ポリシーに違反することなく、パケット・フィルタリング・ファイアウォール環境内で実行される Sterling Connect:Direct システムへの制御されたアクセスが可能となります。Sterling Connect:Direct セッションにおいて、特定の宛先アドレス (場合によっては複数アドレス) に特定の TCP または UDT ソース・ポート番号、あるいはソース・ポート番号の範囲を割り当てることによって、このアクセス制御を行います。

Sterling Connect:Direct 初期化パラメーターでソース・ポートの構成を行う前に、特に UDT 用にファイアウォールを実装する場合には、この章の内容をよく確認してください。

---

### ファイアウォール・ナビゲーションの実装

ファイアウォール・ナビゲーションを Sterling Connect:Direct に実装するには:

1. 初期化パラメーター・ファイルのファイアウォール・ナビゲーション レコードを更新する前に、ローカル・ファイアウォール管理者と、IP アドレスおよび関連付けられるソース・ポートの割り当てを調整します。
2. TCP あるいは UDT を使用しているのかに基づき、必要に応じて、Sterling Connect:Direct 初期化パラメーター・ファイルに以下のパラメーターを追加します。
  - tcp.src.ports
  - tcp.src.ports.list.iterations
  - udp.src.ports
  - udp.src.ports.list.iterations
3. リモート・サイトのファイアウォール管理者と、指定されたポート番号の調整を行います。

---

### ファイアウォール・ナビゲーション

ローカル Sterling Connect:Direct ノードとリモート Sterling Connect:Direct ノードとの通信を可能にするには、ローカル・ファイアウォールでファイアウォール・ルールを作成する必要があります。典型的なパケット・フィルター型のファイアウォール・ルールでは、ローカル・ファイアウォールが、ローカル・アドレス、ローカル・ポート、リモート・アドレス、およびリモート・ポートを伴う、特定のプロトコルからのパケットに対して一方 (インバウンドまたはアウトバウンド) に関与しているよう指定します。ファイアウォール・ナビゲーションは、TCP と UDT で異なるため、フ



ファイアウォール・ルールの構成も TCP と UDT で異なります。

## TCP ファイアウォール・ナビゲーション・ルール

以下の表では、TCP ルールが 2 つのセクションで示されています。1 つ目のセクションは、ローカル・ノードが PNODE として機能している時に必要となるルールに適用されます。2 つ目のセクションは、ローカル・ノードが SNODE として機能している時に必要となるルールに適用されます。典型的なノードは、PNODE として機能する場合も、SNODE として機能する場合もあるため、そのファイアウォールにおいて両方のルール・セットが存在する必要があります。

TCP PNODE ルール			
ルール名	ルールの方向	ローカル・ポート	リモート・ポート
PNODE session	アウトバウンド	ローカル C:D のソース・ポート	リモート C:D の listen ポート
TCP SNODE ルール			
ルール名	ルールの方向	ローカル・ポート	リモート・ポート
SNODE session	インバウンド	ローカル C:D の listen ポート	リモート C:D のソース・ポート

## UDT ファイアウォール・ナビゲーション・ルール

UDT ファイアウォール・ルールは UDP プロトコルに対して適用されます。UDP パケットのデフォルトのファイアウォール・ルールとしては、UDT セッションの確立におけるコールバック機能に起因する混乱を回避するため、ローカル・システムへのインバウンド・パケットおよびローカル・システムからのアウトバウンド・パケットをブロックするルールが推奨されます。

以下の表では、UDT ルールが 2 つのセクションで示されています。1 つ目のセクションは、ローカル・ノードが PNODE として機能している時に必要となるルールに適用されます。2 つ目のセクションは、ローカル・ノードが SNODE として機能している時に必要となるルールに適用されます。典型的なノードは、PNODE として機能する場合も、SNODE として機能する場合もあるため、そのファイアウォールには両方のルール・セットが存在する必要があります。

UDT PNODE ルール			
ルール名	ルールの方向	ローカル・ポート	リモート・ポート
PNODE Session Request	アウトバウンド	ローカル C:D のソース・ポート	リモート C:D の listen ポート
PNODE Session	アウトバウンド	ローカル C:D のソース・ポート	リモート C:D のソース・ポート
UDT SNODE ルール			
ルール名	ルールの方向	ローカル・ポート	リモート・ポート
SNODE listen	インバウンド	ローカル C:D の listen ポート	リモート C:D のソース・ポート
SNODE session	インバウンド	ローカル C:D のソース・ポート	リモート C:D のソース・ポート

## ファイアウォールの構成例

TCP および UDT のファイアウォールの構成例では、以下の IP アドレスおよびソース・ポートが使用されます。

注: 以下の例で使用する IP アドレスは、区別しやすくするために選択されたもので、有効な IP アドレスを示すものではありません。

- ◆ ローカル・ノードは IP アドレス 222.222.222.222 と listen ポート 2264 を持っています。リモート・ノードと通信するためのソース・ポートは 2000 ~ 2200 です。
- ◆ リモート・ノードは IP アドレス 333.333.333.333 と listen ポート 3364 を持っています。ローカル・ノードと通信するためのソース・ポートは 3000 ~ 3300 です。

注: UDT と TCP のセッション確立における違いに関しては、75 ページの『セッションの確立』を参照してください。

### TCP ファイアウォールの構成例

Sterling Connect:Direct 管理者が、ポート 2264 で listen するよう、ローカル・ノードの設定を行います。ローカル・ノードのソース・ポートの構成には、以下の初期化パラメーター設定が使用されます。

- ◆ tcp.src.ports = (333.333.333.333, 2000-2200)
- ◆ tcp.src.ports.list.iterations = 1

この構成は、リモート・ノード・アドレス 333.333.333.333 と通信する際に、2000 ~ 2200 の範囲のソース・ポートを使用し、このポート範囲において使用可能なポートの検索を 1 度実行するように指定しています。ローカル・ノードは、リモート・ノードと通信する際に、PNODE および SNODE の両方として機能します。

このシナリオでは、ローカル・ノードのファイアウォール・ルールは以下のようになります。

ルール名	ルールの方向	ローカル・ポート	リモート・ポート
PNODE session request	アウトバウンド	2000 ~ 2200	3364
SNODE session	インバウンド	2264	3000 ~ 3300

### UDT ファイアウォールの構成例

Sterling Connect:Direct 管理者が、ポート 2264 で listen するよう、ローカル・ノードの構成を行います。ローカル・ノードのソース・ポートの構成には、以下の初期化パラメーター設定が使用されます。

- ◆ udp.src.ports = (333.333.333.333, 2000-2200)
- ◆ udp.src.ports.list.iterations = 1

この構成は、リモート・ノード・アドレス 333.333.333.333 と通信する際に、2000 ~ 2200 の範囲のソース・ポートを使用し、このポート範囲において使用可能なポートの検索を 1 度実行するように指定しています。ローカル・ノードは、リモート・ノードと通信する際に、PNODE および SNODE の両方として機能します。

このシナリオでは、ローカル・ノードのファイアウォール・ルールは以下のようになります。

ルール名	ルールの方向	ローカル・ポート	リモート・ポート
PNODE session request	アウトバウンド	2000 ~ 2200	3364
PNODE session	アウトバウンド	2000 ~ 2200	3000 ~ 3300
SNODE listen	インバウンド	2264	3000 ~ 3300
SNODE session	インバウンド	2000 ~ 2200	3000 ~ 3300

## アウトバウンド・パケットのブロック

ローカル・システムからのアウトバウンド UDP パケットのデフォルトのルールとしては、パケットをブロックすることが推奨されます。この推奨に従わない場合、ポートの使用は、一見してファイアウォールのインバウンド・ルールに違反するように見える可能性があります。

例を使用して、この状況を示すことができます。前セクションでの例で以下を想定します。

- ◆ ローカル・ノードは SNODE。
- ◆ デフォルトのアウトバウンド・ルールでは、ローカル・システムからのすべてのアウトバウンド UDP パケットが許可される。
- ◆ 「SNODE session」ルールが、誤って適用されなかった。

UDT セッションの確立におけるコールバック機能によって、SNODE セッションが、ポート 2000 ~ 2200 で確立できる可能性があります。ポート 2000 ~ 2200 は、インバウンド UDP パケットに対してブロックされているため、これにより混乱が生じる可能性があります。

推奨されるデフォルトのアウトバウンド・ルールを使用し、前セクションでの記載どおりに PNODE および SNODE ルールを適用した場合、使用するポートでの混乱を避けることができ、UDT のコールバック機能が設計通りに機能するため、信頼性がサポートされます。

## セッションの確立

セッションの確立は、TCP と UDT とで異なります。これらの違いは、ファイアウォール・ルールの設定方法、およびファイアウォール・ナビゲーション初期化パラメーターの構成方法に影響を与えます。

### TCP セッションの確立

Sterling Connect:Direct TCP クライアントは、Sterling Connect:Direct TCP サーバーに対して、その listen ポートを通して接触します。Sterling Connect:Direct クライアントはポートのリスト (**tcp.src.ports** 初期化パラメーターを使用して指定) をスキャンし、接続可能なポートを探します。Sterling Connect:Direct がリストをスキャンする回数は、**tcp.src.ports.list.iterations** 初期化パラメーターを使用して指定されています。Sterling Connect:Direct が使用可能なポートを検出すると、リモート・ノードとの通信が開始されます。

## UDT セッションの確立

Sterling Connect:Direct UDT クライアントが Sterling Connect:Direct UDT サーバーに対して、その listen ポートを通してセッション要求のための接触を図ると、UDT サーバーは、そのセッションで使用するための異なるサーバー・ポートで応答します。クライアントは、そのセッション・ポートを通して、サーバーへの接触を試みます。Sterling Connect:Direct クライアントは、ポートのリスト (**udp.src.ports** 初期化パラメーター内で指定) をスキャンし、接続可能なポートを探します。Sterling Connect:Direct がリストをスキャンする回数は、**udp.src.ports.list.iterations** 初期化パラメーターを使用して指定されています。Sterling Connect:Direct が使用可能なポートを検出すると、リモート Sterling Connect:Direct サーバーとの通信が開始されます。特定の時間が経ってもセッションが確立出来ない場合は、サーバーがクライアントへの接触を試みます。

---

## IP アドレス、ホスト名、およびポートの指定

Sterling Connect:Direct は、インターネット・プロトコル・バージョン 4 (IPv4) およびインターネット・プロトコル・バージョン 6 (IPv6) の両バージョン、また、ホスト名も受け入れます。指定するフィールドによって、複数の方法で IP アドレス/ホスト名、およびポートを入力することができます。

- ◆ アドレスまたはホスト名のみ
- ◆ ポート番号のみ
- ◆ アドレス/ホスト名、およびポート番号
- ◆ 複数のアドレス/ホスト名、およびポートの組み合わせ

IP アドレス/ホスト名、およびポートを指定する場合は、以下のガイドラインに従ってください。

---

### IP アドレス

Sterling Connect:Direct は、IPv4 アドレスおよび IPv6 アドレスの両方を受け入れます。IBM Sterling Connect:Direct で IP アドレスを指定する際には常に、IPv4 または IPv6 のいずれかのアドレスを使用することができます。

#### IPv4 アドレス

IPv4 は、 $2^{32}$  個のアドレスをサポートし、10.23.107.5 のように、3 桁の 10 進数 (各桁 0 ~ 9) がドットで区切られた 4 つのグループで表記されます。

#### IPv6 アドレス

IPv6 は、 $2^{128}$  個のアドレスをサポートし、1001:0dc8:0:0:ff10:143e:57ab のように、4 桁の 16 進数がコロンで区切られた 8 つのグループで表記されます。IPv6 アドレスでは、以下のガイドラインが適用されます。

- ◆ 4 桁の数字がゼロ (0000) のグループがある場合、そのゼロを省略して 2 つのコロン (::) で置き換えることができます。例えば、

2001:0db8:85a3:0000:1319:8a2e:0370:1337 を以下のように短縮できます。 2001:0db8:85a3::1319:8a2e:0370:1337
--------------------------------------------------------------------------------------------------

- ◆ 0000 からなるグループが連続している場合、グループの数に関わらず、2 つのコロン (::) で置き換えることができます。ただし、1 つのアドレスにつき、1 組のコロンのペア (::) のみ使用可能です。例えば、

```
001:0db8:0000:0000:0000:0000:1319:58ab
を以下のように短縮できます。
2001:0db8:0000:0000::1319:58ab
```

- ◆ 0000 からなるグループ内の先行ゼロを省略することができます (0000 は 0 に短縮可能)。例えば、

```
2001:0db8:0000:0000:0000:0000:1319:58ab
を下のように短縮できます。
2001:0db8:0:0:0:0:1319:58ab
```

- ◆ IPv6 アドレスの末尾に表れる連続する 4 バイトを、ドットを区切り記号として 10 進数で表記することができます。例えば、

```
::ffff:102:304
を以下のように書き換えられます。
::ffff:1.2.3.4
```

この表記は互換性アドレスにおいて有効です。

---

## ホスト名

IP アドレスではなく、ホスト名を指定する場合には、Sterling Connect:Direct は IP アドレスをオペレーティング・システムから取得します。それが IPv4 フォーマットであるか IPv6 フォーマットであるかに関わらず、オペレーティング・システムから返された最初の IP アドレスが使用されます。

ホスト名 (ネット名、ホスト名、ゲートウェイ名、またはドメイン名) は、アルファベット (A ~ Z)、数字 (0 ~ 9)、負符号 (-)、およびピリオド (.) からなる、最大 24 文字のテキスト・ストリングです。

以下のガイドラインも適用されます。

- ◆ ホスト名の一部にブランク文字やスペース文字を使用することは許可されません。
- ◆ ピリオドはドメイン形式のホスト名の構成要素を区切る場合にのみ、使用が許可されます。
- ◆ ホスト名は大/小文字の区別をしません。
- ◆ ホスト名の最初と最後の文字は、アルファベットまたは数字でなければなりません。
- ◆ 1 文字のみのホスト名またはニックネームは許可されません。

## ポート番号

10.23.107.5;1364 といったように、ポート番号の直前にセミコロン (;) を使用することで、IP アドレス/ホスト名の末尾にポート番号を付加することができます。この表記方法は、Sterling Connect:Direct 特有のものであり、業界標準ではありません。

ポート番号は、0 から 65535 の範囲である必要があります。1024 より小さい番号は、予約済みポート番号として指定されているため、使用すべきではありません。以下の例は、これらの規則を使用した、IP アドレス/ホスト名へのポート番号の付加を示しています。

```
10.23.107.5;1364
fe00:0:0:2014::7;1364
msdallas-dt;1364
```

## 複数のアドレス、ホスト名、およびポート

コンマ (,) で区切ることによって、複数の IPv4/IPv6 アドレスおよびホスト名の指定が可能です。読みやすくするため、コンマの後にスペースを付加することができます。以下はその例です。

```
10.23.107.5, fe00:0:0:2014::7, msdallas-dt
```

各アドレスまたはホスト名に対するポート番号の指定も可能です。ポートは、セミコロン (;) を使用して、対応するアドレス/ホスト名と区切られ、各アドレス/ホスト名およびポートの組み合わせは、コンマ (,) で区切られます。読みやすくするため、コンマの後にスペースを付加することができます。以下の例は、複数のアドレス/ホスト名およびポートの組み合わせを示しています。

```
10.23.107.5;1364, fe00:0:0:2014::7;1364, msdallas-dt;1364
```

複数のアドレス/ホスト名 (ポート番号との組み合わせも含む) は、1024 文字に制限されています。

## IP アドレス範囲へのマスクの使用

初期化パラメーター・ファイルで **tcp.src.ports** パラメーターの値を指定する際に、特定のポート、複数のポート、またはポート範囲を使用する IP アドレス範囲の上限を指定するために、マスクを使用することができます。Sterling Connect:Direct は、以下の **initparms.cfg** ファイルからのエントリーのサンプルに示されるように、IPv4 および IPv6 のアドレスの両方に対するマスクをサポートしています。

```
tcp.src.ports=(199.2.4.*, 1000), (fd00:0:0:2015::*, 2000-3000), (199.2.4.0/255.255.255.0, 4000-5000),
(fd00:0:0:2015::0/48, 6000, 7000)
```

以下のアドレスのサンプルは、それに続く内容を指定します。

(199.2.4.\*, 1000) - 199.2.4.0 から 199.2.4.255 の範囲内のすべての IPv4 アドレスは、ポート 1000 のみを使用します。

(fd00:0:0:2015::\*, 2000-3000) - fd00:0:0:2015:0:0:0:0 から fd00:0:0:2015:ffff:ffff:ffff:ffff の範囲内のすべての IPv6 アドレスは、2000 から 3000 の範囲のポートを使用します。

(199.2.4.0/255.255.255.0, 4000-5000) - 199.2.4.0 から 199.2.255.255 の範囲内のすべての IPv4 アドレスは、4000 から 5000 の範囲のポートを使用します。

(fd00:0:0:2015::0/48, 6000, 7000) - fd00:0:0:2015:0:0:0:0 から fd00:0:0:ffff:ffff:ffff:ffff:ffff の範囲内のすべての IPv6 アドレスは、ポート 6000 またはポート 7000 を使用します。

上記のエントリーのサンプルが示すように、IP アドレスのパターンを定義するために、ワイルドカード文字 (\*) がサポートされています。それぞれが有効なソース・ポートのリストを伴う、255 までの固有の IP アドレス・パターン、もしくは1024 文字長までの指定が可能です。ワイルドカード文字が使用されると、マスク・オプションは無効となります。



---

## テスト・モードでの Sterling Connect:Direct の使用

Sterling Connect:Direct の実働インスタンスが以下の機能を実行するために、テスト・モードを有効にすることができます。

- ◆ 新規のアプリケーションおよび顧客との接続のテスト
- ◆ Flush Process コマンドを使用してすべてのアクティブな実働作業を終了した後、テストが終了するまでの、新たな実働作業の実行の防止
- ◆ テストが終了した後の通常の実働作業の再開
- ◆ アプリケーションによる個々のファイル転送の制御
- ◆ 個々のノードおよびアプリケーションの有効化または無効化

テストが行われている間は、テスト作業に関連する Process のみ (特にファイル転送) が実行されます。テスト実施中のアプリケーションに実働データが転送されることはなく、実働アプリケーションにテスト・データが転送されることもありません。

---

### テスト・モードの処理フロー

**quiesce.resume** 初期化パラメーターを使用して、テスト・モードを有効とし、NDMPXTBL という名前のパラメーター・テーブル中にテキスト・レコードの形で設定を保存することによって、各 Sterling Connect:Direct Process の実行/不実行を指定してください。パラメーター・ファイルのサンプル NDMPXTBL.sample は、/ndm/src ディレクトリーに置かれています。このファイルをテスト環境に合わせて更新した後、ndm/cfg/<nodename> インストール・ディレクトリーに配置してください。quiesce.resume パラメーターを有効にした場合、Sterling Connect:Direct のテスト・モードでの稼動に NDMPXTBL テーブルが必要となります。

実行に包含 (「I」コマンド・コードを使用)、または実行から除外する(「X」コマンド・コード) 1 つ以上の Process とのマッチの検索用に、以下の基準を指定することができます。

- ◆ Process 名の全部または一部
- ◆ リモート・ノード名の全部または一部
- ◆ Sterling Connect:Direct 実行依頼者 ID および実行依頼者ノードの組み合わせの全部または一部

Sterling Connect:Direct に実行する Process を伝えるだけでなく、実行されない Process をどうすべきかもシステムに伝えてください。実行が禁止されている Process に対しては、以下の処理を指定することができます。

- ◆ Process を保留キューに配置する。
- ◆ Process をセッション再試行のため、タイマー・キューに配置する。
- ◆ キューから Process をフラッシュする。

テスト・モードの使用法に関する詳細については、84 ページの『テスト・シナリオのサンプル』を参照してください。

テスト・モードが有効にされると、Sterling Connect:Direct はパラメーター・テーブルの構文チェックを行い、テーブルが無効の場合には、初期化は失敗します。テーブルが有効な場合、Sterling Connect:Direct はそのスキャンを行い、実行しようとしている Process と一致するパターンを検索します。一致するパターンが見つかった場合、「I」(包含) コマンド・コードが有効な Process の実行が許可されます。コマンド・コード「X」(除外) が有効な Process の実行は許可されません。テーブル中に一致するパターンが見つからなかった場合、一致するパターンが見つかった場合とは反対の処理が行われます。すなわち、一致するパターンが見つからなかった場合は、コマンド・コード「I」が有効となっている Process の実行は許可されず、コマンド・コード「X」が有効となっている Process の実行が許可されます。

Process の実行が許可されない場合には、Process を保留、再試行、あるいはフラッシュするように、NDMPXTBL パラメーター・テーブルに指定された処理が実施され、ゼロでない戻りコードが返されます。テスト・モードで Process の実行が阻止された場合、適切なメッセージが発行され、統計ログで確認できます。

---

**注:** リモート・ノードで開始された Process に対して、テスト・モードは、リモート・ノードがその Process の PNODE (Process 所有者) であり、ローカル・ノードが SNODE (2 次ノード) である場合を除き、ローカル Sterling Connect:Direct ノード上で実行依頼された Process に対してと同様に機能します。NDMPXTBL パラメーター・テーブル内で一致するエントリーの検索が実行され、リモートで開始された Process は実行が許可されるか、もしくは実行から除外されます。このタイプの転送では、ローカル・ノードが SNODE であるため、NDMPXTBL パラメーター・テーブル内の Process の処理設定を強制できません。Process の処理は、リモート PNODE によって決定されます。通常、リモート・ノードは Process を HE (Held in Error) ステータスで保留キューに配置します。

---

## NDMPXTBL パラメーター・テーブルの準備

任意のテキスト・エディターを使用して、Sterling Connect:Direct で提供される NDMPXTBL パラメーター・テーブルのサンプルを修正することができます。パラメーター・テーブルを更新する際には、NDMPXTBL と名前を付け、サーバーのインストール・ディレクトリーに保存します。パラメーター・テーブル・ファイルは、サーバーがアクティブな時に作成または更新することが可能であり、ファイルに対するいかなる変更も、その変更後に開始されるセッションから有効となります。同様に、**quiesce.resume** 初期化パラメーターも、サーバーがアクティブである時に修正することが可能です。**quiesce.resume** 初期化パラメーターに関する詳細については、24 ページの『停止/再開レコードの更新』を参照してください。

---

**注:** **quiesce.resume** 初期化パラメーターを有効にする場合、NDMPXTBL パラメーター・テーブルが必要となります。

---

各テーブル・エントリーまたはレコードは、1 列目の 1 文字のコマンド・コードから成ります。大半のコマンド・コードには 2 列目から始まるパラメーターが存在し、これはコマンド・コードの機能によって異なります。

コマンド・コード	説明	サブパラメーター/例
*	コメント行。	* 以下の Process のみ実行してください。
E	テーブル・エントリーに基づいて、Process の実行を有効化します。テーブル内の最初の非コメント・エントリーは、「E」または「D」のいずれかである必要があります。	このエントリーの 2 列目には、PNODE Process の実行が許可されていない場合の Process の処理を示す以下の値のうちの一つが含まれる必要があります。 H - Process を保留キューに配置します。 R - Process をセッション再試行でタイマー・キューに配置します。 F - Process をキューからフラッシュします。
D	パラメーター・テーブルの内容に関わらず、すべての Process の実行が無効化され、Process の実行は、ゼロでない (エラー) 戻りコードおよびメッセージ LPRX003E で失敗します。テーブル内の最初の非コメント・エントリーは、「E」または「D」のいずれかである必要があります。	コマンド・コード「E」に対するパラメーターを 2 列目に記載することができます。これは、コマンド・コード「D」の 2 列目を空白に変更する必要性をなくすことで、「E」から「D」への変更、またその反対への変更を容易にする便宜を図るものです。
P	Process 名の全部または一部に基づいて Process のマッチングを行います。末尾のワイルドカードとしてのアスタリスク (*) の使用がサポートされています。命名規則を用いたマッチングで一致する特定のアプリケーションへの Process 実行を有効または無効にすることができます。	PCOPY - 単一 Process とのマッチングを行います。 PACH* - 「ACH」で始まる全 Process とのマッチングを行います。 P* - 全 Process とのマッチングを行います。
N	リモート・ノード名の全部または一部に基づいて Process のマッチングを行います。末尾のワイルドカードとしてのアスタリスク (*) の使用がサポートされています。	NCD.NODE1 - 単一リモート・ノード名とのマッチングを行います。 NCD.NODEA* - 「CD.NODEA」で始まる全リモート・ノード名とのマッチングを行います。 N* - 全リモート・ノード名とのマッチングを行います。
S	全表記またはワイルドカード文字を使用した表記の Sterling Connect:Direct 実行依頼者 ID および実行依頼者ノードの組み合わせに基づいて Process のマッチングを行います。フォーマットは <id>@<ノード> です。	SACTQ0ACD@TPM002 - 特定の ID およびノードの組み合わせとのマッチングを行います。 S*@TPM002 - ノード TPM002 の全 ID とのマッチングを行います。 SACTQ0ACD@* - 全ノードの ID とのマッチングを行います。 S*@* - 全ノードの全 ID とのマッチングを行います。これは、すべての Process とのマッチングを行うもう一つの方法です。

コマンド・コード	説明	サブパラメーター/例
I	<p>このコマンド・コードに従い、テーブル内のパターンとマッチする Process を実行に含めます。テーブル内の 2 番目の非コメント・エントリーは、「I」もしくは「X」のいずれかである必要があります。テーブル内のパターンとマッチしない Process は<b>実行されません</b>。</p> <p><b>注:</b> Process の選択に使用するコマンド・コードを選択する際に、小さい方のグループを特定し、それに対応するコマンド・コードを使用してください。例えば、実行される Process の数が実行されない Process の数より小さい場合には、コマンド・コードに「I」を指定し、その Process のグループにマッチするパターンを追加してください。</p>	<p>ER</p> <p>I</p> <p>NCD.BOSTON</p> <p>CD.BOSTON ノードの Process のみを実行に含めます。他のすべてのリモート・ノードで予定されている Process は、セッション再試行でタイマー・キューに置かれます。</p>
X	<p>このコマンド・コードに従い、テーブル内のパターンとマッチする Process を実行から除外します。テーブル内の 2 番目の非コメント・エントリーは、「X」もしくは「I」のいずれかである必要があります。テーブル内のパターンとマッチしない Process は<b>実行されません</b>。</p>	<p>EH</p> <p>X</p> <p>SDALLASOPS@*</p> <p>任意のノードの ID DALLASOPS から実行依頼された Process を実行から除外します。</p>
L	<p>テーブルの最終エントリー。</p>	

## テスト・シナリオのサンプル

以下は、Sterling Connect:Direct Process の実行/不実行を定義する NDMPXTBL パラメーター・テーブルを使用した、テスト・モードの異なる適用例を示しています。

### 実行する Process の指定

この例では、Sterling Connect:Direct は、ACH で始まる Process、もしくは、DITEST01 または DITEST02 という名前の Process のすべてを実行します。他の Process はすべて保留キーに配置されます。

<p>* Enable processing. Only permit processes matching one of the patterns</p> <p>* to execute. Hold processes that don't execute.</p> <p>EH</p> <p>I</p> <p>PACH</p> <p>*</p> <p>PDITEST01</p> <p>PDITEST02</p> <p>L</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 除外する Process の指定

この例では、Sterling Connect:Direct は、ACH で始まる Process、もしくは、DITEST01 または DITEST02 という名前の Process のすべてを実行しません。他の Process はすべて実行されます。

```
* Exclude matching processes. Permit all others to execute.
EH
X
PACH*
PDITEST01
PDITEST02
L
```

## 2 次ノードおよび実行依頼者ユーザー ID/ノードによる Process 実行の許可

この例では、Sterling Connect:Direct は、以下の基準の 1 つとマッチするすべての Process を実行します。

- ◆ 名前が DI.NODE1 である、特定の 2 次ノード (SNODE)
- ◆ 名前が DI0017 で始まる SNODE
- ◆ ノード DI0049 からの、任意の Sterling Connect:Direct 実行依頼者 ID
- ◆ 任意のノードからの特定の Sterling Connect:Direct 実行依頼者 ID ACHAPP

上記の基準を 1 つも満たさない Process はすべてキューからフラッシュされます。

```
* Only permit matching processes to execute. Flush those that do not.
EF
I
NDI.NODE1
NDI0017*
S*@DI0049
SACHAPP@*
L
```

## テスト・モードの停止

この例では、実行される Process はなく、エラーを示すゼロでない戻りコードと、メッセージ ID LPRX003E が表示されます。テーブルの残りの部分は無視され (Process をキューからフラッシュする「F」コードも含む)、すべての Process は保留キューに置かれます。

テストを再開するには、「D」コマンド・コードを「E」に変更してください。

```
* Execute no processes at all. Put them in the hold queue and return.
DF
I
PACH*
PDITEST01
PDITEST02
L
```

## A

### アプリケーション・プログラミング・インターフェース (Application Program Interface: API)

エンド・ユーザー・アプリケーション(End User Application: EUA) が Sterling Connect:Direct ソフトウェアと通信できるようにするために呼び出される関数のライブラリー。

## C

### クライアント (Client)

Sterling Connect:Direct サーバーへの要求を作成し、サーバーによる応答を受け入れるプログラム。

### コマンドライン・インターフェース (Command Line Interface: CLI)

コマンドライン環境から Sterling Connect:Direct の Process およびコマンドを実行依頼することが可能なプログラム。

### コマンド・マネージャー (Command Manager: CMGR)

API から送信されるコマンドを実行し、結果を API に返すプログラム。CMGR は API と連動して、Sterling Connect:Direct の認証手順を実行します。これにより、システムへのアクセスが承認されているユーザー名およびパスワードかどうかを判断します。CMGR は、コマンドの実行の際に要求があった場合には PMGR と通信します。

## D

### デーモン (daemon)

クライアントにサービスを提供する長期実行プロセス。PMGR は、Sterling Connect:Direct for UNIX のデーモンです。

## 診断コマンド (Diagnostic Commands)

Sterling Connect:Direct ソフトウェアにおける問題の診断を支援する Sterling Connect:Direct コマンド。

## E

### エンド・ユーザー・アプリケーション (End User Application: EUA)

特定のタスクを遂行するために、エンド・ユーザーによって開発されるアプリケーション・プログラム。

### 実行キュー (Execution Queue)

TCQ (伝送制御キュー) 内の論理キュー。実行キュー内の Process は、Sterling Connect:Direct ノードから、あるいは Sterling Connect:Direct ノードへのデータ転送を行っているか、またはタスク実行が可能となる前に、リモート Sterling Connect:Direct ノードへの接続を待機しています。

## H

### 保留キュー (Hold Queue)

TCQ (伝送制御キュー内) の論理キュー。保留キュー内の Process は、オペレーターによる介入を待った後、スケジューリングのため待機キューに移動します。

## I

### IBM Sterling Connect:Direct

Sterling Connect:Direct を参照。

### IBM Sterling Connect:Direct Browser User Interface

Sterling Connect:Direct Browser User Interface を参照。

### IBM Sterling Connect:Direct for UNIX

Sterling Connect:Direct for UNIX を参照。

### IBM Sterling Connect:Direct Node

Sterling Connect:Direct ノードを参照。

## IBM Sterling Connect:Direct Process

Sterling Connect:Direct Process を参照。

## IBM Sterling Control Center

Sterling Control Center を参照。

# M

## 監視コマンド (Monitoring Commands)

統計ファイルおよび TCQ (伝送制御キュー) からの Sterling Connect:Direct Process 実行結果に関する情報の表示を可能とする Sterling Connect:Direct コマンド。

# O

## 操作制御コマンド (Operational Control Commands)

Process の実行依頼、TCQ (伝送制御キュー) 内の特定の Process の特性の変更、TCQ からの実行中の Process や非実行中の Process の排除、および Sterling Connect:Direct の停止を可能とする、Sterling Connect:Direct コマンド。

# P

## Process マネージャー (Process Manager: PMGR)

Sterling Connect:Direct ソフトウェアの初期化、Sterling Connect:Direct API およびリモート Sterling Connect:Direct ノードからの接続要求の受け入れ、コマンド・マネージャーおよびセッション・マネージャーの作成、一元化された Sterling Connect:Direct 機能が必要な場合のコマンド・マネージャーおよびセッション・マネージャーからの要求の受け入れ、および Sterling Connection:Direct ソフトウェアの実行の停止を行う、長期実行型の Sterling Connect:Direct サーバー。

## 1 次ノード (Primary Node: PNODE)

Process が実行される Sterling Connect:Direct ノード。1 次ノードは、制御ノードまたはソース・ノードとも呼ばれますが、PNODE は受信側にもなり得るため、必ずしも送信ノードとはなりません。各 Process において、1 つの PNODE と 1 つの SNODE が存在します。Process の実行依頼者は、常に PNODE になります。PNODE 名は、1 から 16 文字の長さで指定できます。



## S

### セッション (Session)

2 つの Sterling Connect:Direct ノード間における接続。

### セッション・マネージャー (Session Manager: SMGR)

リモート Sterling Connect:Direct ノードとの接続を作成および終了し、必要な Sterling Connect:Direct 作業を実行するサーバー・コンポーネント。

### 2 次ノード (Secondary Node: SNODE)

Process 実行時に 1 次ノード (PNODE) と通信する Sterling Connect:Direct ノード。2 次ノード (SNODE) は、参加ノード、ターゲット・ノード、宛先ノードと称される場合もあります。各 Process には、1 つの PNODE と 1 つの SNODE が存在します。2 次ノードは、別のノード (PNODE) によって開始された Process の実行に参加するノードです。SNODE 名は、1 から 16 文字の長さで指定できます。

### Sterling Connect:Direct

複数のデータ・センター全体にわたって情報の配信および実働アクティビティの管理を行う、データ転送ソフトウェア製品ファミリー。

### Sterling Connect:Direct Browser User Interface

コマンドライン・インターフェースを通じた Sterling Connect:Direct コマンドの実行依頼に代わって、Sterling Connect:Direct Browser User Interface を使用することで、Microsoft Internet Explorer などのインターネット・ブラウザから Process を作成、実行依頼、監視することができます。適切な Sterling Connect:Direct 権限を持っている場合には、Sterling Connect:Direct Browser User Interface を使用して、ネットワーク・マップや初期化パラメーターの表示や変更といった、Sterling Connect:Direct システム管理タスクを実行することもできます。

### Sterling Connect:Direct File Agent

Sterling Connect:Direct のアプリケーション・プログラムおよびコンポーネント。指定されたディレクトリーをスキャンして、ファイルの存在を検索します。監視対象ディレクトリーにファイルが見つかり、Sterling Connect:Direct は、Process を実行依頼するか、またはそのファイルに対するルールに指定されているアクションを実行します。

### Sterling Connect:Direct for UNIX

Sterling Connect:Direct 製品の UNIX での実装。

### Sterling Connect:Direct ノード (Sterling Connect:Direct Node)

Sterling Connect:Direct を稼働するすべてのコンピューター/ワークステーション。

## Sterling Connect:Direct Process

事前定義してディレクトリー内に保存することができ、API を通して実行依頼することで Sterling Connect:Direct for UNIX アクティビティーを開始することが可能な一連のステートメント。Process 機能の例には、ファイルのコピーやジョブの実行などがあります。

## Sterling Control Center

Sterling Connect:Direct z/OS、UNIX、Microsoft Windows サーバーにおいて、オペレーション担当者に対して継続的かつ全社的なビジネス・アクティビティーの監視機能を提供する一元化された管理システム。複数の Sterling Connect:Direct サーバーを管理して Process の中断、解放、および削除を行い、Sterling Connect:Direct サーバーの停止や、実行中の Process または完了した Process に関する詳細な統計情報の表示を行います。また、ネットワーク内の Sterling Connect:Direct for z/OS、UNIX、Microsoft Windows サーバー全体における Sterling Connect:Direct 処理を確認し、アクティブな Process および完了した Process に関する情報を検索する際のサービス・レベルを監視します。スケジュール通りに発生した、または発生しないデータ配信イベントの通知を受け取り、処理基準に基づいて、アラートの生成、Eメール通知の送信、エンタープライズ管理システム (EMS) に対する Simple Network Management Protocol (SNMP) トラップの生成、またはシステム・コマンドの実行を行うことができるルールを定義します。サーバー障害または Process が時間どおりに開始しないといった、アラートを監視します。

## T

### TCQ ステータス値 (TCQ Status Value)

Process が TCQ (伝送制御キュー) に置かれた際に、Sterling Connect:Direct によって Process に割り当てられる 2 文字のコード。Process のステータスは、**select process** コマンドを使用して調査することができます。

### 伝送制御キュー (Transmission Control Queue: TCQ)

Sterling Connect:Direct for UNIX に実行依頼されたすべての Process を保持するキュー。TCQ には、以下の 4 つの論理キューが含まれます

- ◆ 実行
- ◆ 待機
- ◆ タイマー
- ◆ 保留

### タイマー・キュー (Timer Queue)

TCQ (伝送制御キュー) 内の論理キュー。タイマー・キュー上の Process は、開始時刻を待った後、スケジューリングのため待機キューに移動します。

## W

### 待機キュー (Wait Queue)

TCQ (伝送制御キュー) 内の論理キュー。待機キュー上の Process は、リモート Sterling Connect:Direct ノードから、またはリモート Sterling Connect:Direct ノードへの接続を待機します。

## A

初期化パラメーター・ファイルについて 20  
admin.auth、ローカル・ユーザー情報レコード 58  
alt.comm.outbound、リモート接続パラメーター 52  
API 構成パラメーター、リスト 38  
api.max.connects、ローカル・ノード接続パラメーター 43  
api.parms レコード 38  
認証パラメーター、説明 32  
認証レコード 39

## C

cdcust スクリプト、構成ファイルの修正 19  
ckpt.interval、copy パラメーター 27  
CLI 構成パラメーター、リスト 38  
CLI/API 構成ファイル  
  client.keyfile 39  
  client.program 39  
  ロケーション 37  
  tcp.hostname 38  
  tcp.port 38  
  wait.time 38  
クライアントおよびサーバー認証キー・ファイル、概要 68  
クライアント認証キー・ファイル  
  認証パラメーター 71  
  概要 68  
  権限 71  
クライアント認証パラメーターのリスト 39  
クライアント構成ファイルの定義 38  
client.keyfile、CLI/API 構成パラメーター 39  
client.program、CLI/API 構成パラメーター 39  
cmd.chgproc、ローカル・ユーザー情報レコード 58  
cmd.delproc、ローカル・ユーザー情報レコード 58  
cmd.flsproc、ローカル・ユーザー情報レコード 59  
cmd.selproc、ローカル・ユーザー情報レコード 59

cmd.selstats、ローカル・ユーザー情報レコード 59  
cmd.stopndm、ローカル・ユーザー情報レコード 59  
cmd.submit、ローカル・ユーザー情報レコード 60  
cmd.trace、ローカル・ユーザー情報レコード 60  
CMGR、説明 8  
comm.bufsize リモート・ノード・パラメーター 52  
comm.info、リモート・ノード接続パラメーター 25、53  
comm.transport  
  LU 6.2 パラメーター 25、53  
  リモート・ノード接続情報 53  
  リモート・ノード接続パラメーター 25  
  リモート・ノード・パラメーター 25  
コマンド・マネージャー、概要 8  
構成  
  ファイル、修正 19、37  
  初期化パラメーター・ファイル 20  
  ネットワーク・マップ・パラメーター 41  
  ユーザー認証パラメーター 56  
conn.retry.l attempts  
  ローカル・ノード・パラメーター 43、49  
  リモート・ノード・パラメーター 54  
conn.retry.l twait  
  ローカル・ノード・パラメーター 43、49  
  リモート・ノード・パラメーター 54  
conn.retry.s attempts  
  ローカル・ノード・パラメーター 43、49  
  リモート・ノード・パラメーター 53  
conn.retry.s twait  
  ローカル・ノード・パラメーター 43、49  
  リモート・ノード・パラメーター 53  
contact.name  
  ローカル・ノード・パラメーター 44、50  
  リモート・ノード・パラメーター 54  
contact.phone  
  ローカル・ノード・パラメーター 44、50  
  リモート・ノード・パラメーター 54

continue.on.exception、copy パラメーター 27

copy パラメーター、説明 27

copy.parms レコード 27

CRC 検査 29、55、60、62

## D

default、優先順位パラメーター 24

descrip

ローカル・ノード・パラメーター 44、50

ローカル・ユーザー情報 レコード 60

リモート・ノード・パラメーター 54

リモート・ユーザー情報 レコード 64

説明

CMGR 8

PMGR 7

SMGR 8

ユーザー認証 9

## E

ecz.compression.level、copy パラメーター 28

ecz.memory.level、copy パラメーター 28

ecz.window.size、copy パラメーター 28

## F

file.open.exit.program、ユーザー出口パラメーター 34

file.size、ファイル情報パラメーター 31

ファイル

クライアント認証キー・ファイル 68

初期化パラメーター、修正 20

サーバー認証キー・ファイル 68

強固なアクセス制御ファイル 65

ユーザー認証情報ファイル、修正 56

ファイアウォール・ナビゲーション・パラメーター、説明 35

firewall.parms レコード 35

## G

一般、サーバー構成ファイルのホスト名 69

## H

ホスト名

複数 79

指定 78

## I

初期化パラメーター・ファイル

概要 20

authentication 32

ckpt.interval 27

comm.info 25

comm.transport 25、53

copy.parms 27

定義 20

ecz.compression.level 28

ecz.memory.level 28

ecz.window.size 28

file.open.exit.program 34

file.size 31

local.node 42

ロケーション 20

log.commands 31

log.select 31

max.age 26

修正 20

ndm.pam 23

path パラメーター 22

優先順位レコード 24

recid 25

restrict:cmd 61、64

retry.codes 28

retry.msgids 29

rnode.listen 25

runtask.parms 30

security.exit.program 34

server.keyfile 33

server.program 33

stats.exit.program 34

syslog.logd 31

tcp.crc 29

tcp.crc.override 29

tcp.src.ports 35、36、72

tcp.src.ports.list.iterations 35、36、72

TCQ 26

ulimit 27

xlate.dir 27

xlate.recv 27

xlate.send2 27

insert.newline パラメーター 30

IP アドレス

マスク 79

IP アドレス範囲、マスクの使用 79

IP アドレス 77

IPv4 77

IPv6 77

複数 79

IPv4 77

IPv4 アドレス 77

IPv6 77

IPv6 アドレス 77  
ガイドライン 77

## K

キー・ファイル

概要 68

クライアントに必要な権限 71

サーバーに必要な権限 71

## L

ローカル・ユーザー情報レコード

概要 58

admin.auth 58

cmd.chgproc 58

cmd.delproc 58

cmd.flsproc 59

cmd.selproc 59

cmd.selstats 59

cmd.stopndm 59

cmd.submit 60

cmd.trace 60

descrip 60

name 60

phone 60

pstmt.copy 60、63

pstmt.copy.ulimit 60

pstmt.crc 60、62

pstmt.download\_dir 61、64

pstmt.runjob 62、64

pstmt.runtask 59、61

pstmt.submit 63、64

pstmt.submit\_dir 62

pstmt.upload 61、63

pstmt.upload\_dir 61、63

snode.ovrd 62

local.id、リモート・ユーザー情報レコード 63

ローカル・ノード、初期化パラメーター・レコード 42

log.commands、ファイル情報パラメーター 31

log.select、ファイル情報パラメーター 31

## M

max.age、パラメーター 11

max.age、TCQ パラメーター 26

構成ファイルの修正 19

## N

name パラメーター、ndm.node レコード内 23

name、ローカル・ユーザー情報レコード内 60

ndm.node レコード 23

ndm.path レコード 22

ndm.pam レコード 23

snode.work.path パラメーター 23

NDMPXTBL パラメーター・テーブル 82

NDMPXTBL テーブル 81

netmap.check、ローカル・ノード・パラメーター 44

ネットワーク・マップ・ファイル

comm.bufsize 52

comm.info パラメーター 53

conn.retry.ltattempts 43、49、54

conn.retry.ltwait 43、49、54

conn.retry.stattempts 43、49

conn.retry.stwait 43、49、53

contact.name 44、50、54

contact.phone 44、50、54

descrip 44、50、54

ロケーション 41

修正 41

netmap.check 44

pacing.send.count 45、51、54

pacing.send.delay 45、51、54

proxy.attempt 46

runstep.max.time.to.wait 47、50、55

sess.default 47、50

sess.pnode.max 47、50、55

sess.snode.max 47、50、55

sess.total 47、50、55

tcp.api 48

tcp.api.bufsize 47

tcp.crc 55

tcp.ip.default 49

tcp.max.time.to.wait 48、49

## P

pacing.send.count  
   ローカル・ノード・パラメーター 45  
   リモート・ノード・パラメーター 54  
   ローカル・ノード・パラメーターの TCP/IP 設定 51  
 pacing.send.delay  
   ローカル・ノード・パラメーター 43、51  
   リモート・ノード・パラメーター 54  
 path パラメーター、ndm.path レコード内 22  
 権限  
   クライアントに必要な権限 71  
   サーバーに必要な権限 71  
 phone、ローカル・ユーザー情報レコード 60  
 PMGR、説明 7  
 ポート番号  
   指定 79  
 ポート  
   複数 79  
 proc.prio レコード 24  
 Process 再始動、概要 9  
 Process、サンプル 11  
 プロファイル名、LU 6.2 パラメーター 25、53  
 プログラム・ディレクトリー、概要 66  
 proxy.attempt、ローカル・ノード・パラメーター 46  
 pstmt.copy、ローカル・ユーザー情報レコード 60、63  
 pstmt.copy.ulimit、ローカル・ユーザー情報レコード 60  
 pstmt.crc、ローカル・ユーザー情報レコード 60、62  
 pstmt.download  
   ローカル・ユーザー情報 61、63  
 pstmt.download\_dir  
   ローカル・ユーザー情報レコード 61、64  
 pstmt.run\_dir  
   ローカル・ユーザー情報レコード 61  
   リモート・ユーザー情報レコード 64  
 pstmt.runjob、ローカル・ユーザー情報レコード 62、64  
 pstmt.runtask、ローカル・ユーザー情報レコード 62、64  
 pstmt.submit、ローカル・ユーザー情報レコード 62、64  
 pstmt.submit\_dir  
   ローカル・ユーザー情報レコード 62  
   リモート・ユーザー情報レコード 64  
 pstmt.upload  
   ローカル・ユーザー情報レコード 61、63

pstmt.upload\_dir  
   ローカル・ユーザー情報レコード 61、63

## Q

停止と再開、テスト・モード 81  
 quiesce.resume  
   テスト・モード 81

## R

recid、リモート・ノード接続パラメーター 25  
 レコード  
   api.parms 38  
   authentication 32  
   copy.parms 27  
   firewall.parms 35  
   ローカル・ユーザー情報 58  
   ndm.node 23  
   ndm.path 22  
   リモート・ユーザー情報 62  
   リモート・ユーザー ID@リモート・ノード名 63  
   runtask.parms 30  
   stats 31  
   tcp.ip.default 49  
   tcq 26  
   user.exits 34  
 レコード、authentication 39  
 リモート・ノード情報レコード  
   作成 13  
   修正 41  
 リモート・ユーザー情報レコード  
   概要 62  
   descrip 64  
   local.id 63  
   pstmt.run\_dir 64  
   pstmt.submit\_dir 64  
 リモート・ユーザー ID@リモート・ノード名、ユーザー認証  
   情報レコード 63  
 restart、run task パラメーター 30  
 restrict  
   cmd、初期化パラメーター 61  
   cmd 初期化パラメーター 64  
 制限付きシェル、概要 66  
 retry.codes、copy パラメーター 28  
 retry.msgids、copy パラメーター 29  
 mode.listen レコード 25

Run task、パラメーター 30

runstep.max.time.to.wait

ローカル・ノード・パラメーター 47、50

リモート・ノード・パラメーター 52

## S

サンプル

Process 11

シェル・スクリプト 12

セキュリティ

認証手順 70

クライアント認証パラメーター 71

キー・ファイルのフォーマット 68

プログラム・ディレクトリー 66

サーバー認証パラメーター 71

セキュリティ出口、初期化パラメーター・ファイル内 66

security.exit.program、ユーザー出口パラメーター 34

サーバー認証キー・ファイル、認証パラメーター 71

サーバー認証パラメーター

説明 32

概要 68

権限 71

server.keyfile、サーバー認証パラメーター 33

server.program、サーバー認証パラメーター 33

sess.default、ローカル・ノード・パラメーター 47、50

sess.pnode.max

ローカル・ノード・パラメーター 47、50

リモート・ノード・パラメーター 55

sess.snode.max

ローカル・ノード・パラメーター 47、50

リモート・ノード・パラメーター 55

sess.total

ローカル・ノード・パラメーター 47、50

リモート・ノード・パラメーター 55

セッション・マネージャー、概要 8

シャドー・パスワード検出 65

シェル・スクリプト、サンプル 12

SMGR、説明 8

snode.ovrd、ローカル・ユーザー情報レコード 62

snode.work.path パラメーター 23

統計ファイル情報、パラメーター 31

stats レコード 31

stats.exit.program、ユーザー出口パラメーター 34

Sterling Connect:Direct

クライアント認証パラメーター 71

構成、概要 18

セキュリティ 68

セキュリティ、認証手順 70

サーバー認証パラメーター 71

strip.blanks パラメーター 29

強固なアクセス制御ファイル 65

sysacl.cfg、強固なアクセス制御 65

syslog.logd、ファイル情報パラメーター 31

## T

tcp.api、ローカル・ノード・パラメーター 48

tcp.api.bufsize、ローカル・ノード・パラメーター 47

tcp.api.inactivity.timeout、ローカル・ノード・パラメーター 48

tcp.crc

copy パラメーター 29

リモート・ノード・パラメーター 55

tcp.crc.override、copy パラメーター 29

tcp.hostname CLI/API 構成パラメーター 38

tcp.ip.default、初期化パラメーター・レコード 49

tcp.max.time.to.wait、ローカル・ノード・パラメーター 48、49

tcp.port、CLI/API 構成パラメーター 38

tcp.src.ports、ファイアウォール・ナビゲーション・パラメーター 35

tcp.src.ports.list.iterations、ファイアウォール・ナビゲーション  
パラメーター 35

TCP/IP パラメーター、説明 25

TCQ パラメーター、説明 26

テスト・モード

NDMPXTBL テーブル 81

処理フロー 81

シナリオのサンプル 84

テスト・モード 81

テスト・モード、シナリオ 84

テキスト・エディター、構成ファイルの修正 19



## U

udp.src.ports、ファイアウォール・ナビゲーション・パラメーター 36

udp.src.ports.list.iterations、ファイアウォール・ナビゲーション・パラメーター 36

ulimit、copy パラメーター 27

UNIX、制限付きシェル 66

ユーザー認証情報ファイル

説明 9

ローカル・ユーザー情報 58

修正 56

プログラム・ディレクトリー 66

リモート・ユーザー情報 62

リモート・ユーザー ID@リモート・ノード名 63

ユーザー ID 56

ユーザー出口パラメーター、説明 34

user.exits レコード 34

userfile.cfg、内容と使用 56

## W

wait.time、CLI/API 構成パラメーター 38

## X

xlate.dir、copy パラメーター 27

xlate.recv、copy パラメーター 27

xlate.send、copy パラメーター 27

---

## 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。

〒103-8510

東京都中央区日本橋箱崎町 19 番 21 号

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  
で提供し、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*

*Dept. 600A*

*Mail Drop 1329*

*Somers, NY 10589*

*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問い合わせください。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります。単に目標を示しているものです。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。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。これらのサンプル・プログラムは特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

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的創作物にも、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。

© IBM 2011. 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. 2011.

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

## 商標

IBM、IBM ロゴおよび [ibm.com](http://ibm.com) は、世界の多くの国で登録された International Business Machines Corp. の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、

[www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml) の『[Copyright and trademark information](#)』を参照してください。

Adobe、Adobe ロゴ、PostScript、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

IT Infrastructure Library は英国 Office of Government Commerce の一部である the Central Computer and Telecommunications Agency の登録商標です。

インテル、Intel、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Centrino、Intel Centrino ロゴ、Celeron、Xeon、Intel SpeedStep、Itanium、および Pentium 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。

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 NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

ITIL は英国 Office of Government Commerce の登録商標および共同体登録商標であって、米国特許商標庁にて登録されています。

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。

Java およびすべての Java 関連の商標およびロゴは Oracle やその関連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。

Cell Broadband Engine, Cell/B.E は、米国およびその他の国における Sony Computer Entertainment, Inc. の商標であり、同社の許諾を受けて使用しています。

Linear Tape-Open, LTO、LTO ロゴ、Ultrium、および Ultrium ロゴは、米国およびその他の国における HP、IBM Corp.、および Quantum の商標です。

Connect Control Center<sup>®</sup>、Connect:Direct<sup>®</sup>、Connect:Enterprise<sup>®</sup>、Gentran<sup>®</sup>、Gentran<sup>®</sup>:Basic<sup>®</sup>、Gentran:Control<sup>®</sup>、Gentran:Director<sup>®</sup>、Gentran:Plus<sup>®</sup>、Gentran:Realtime<sup>®</sup>、Gentran:Server<sup>®</sup>、Gentran:Viewpoint<sup>®</sup>、Sterling Commerce<sup>™</sup>、Sterling Information Broker<sup>®</sup>、および Sterling Integrator<sup>®</sup> は、Sterling Commerce<sup>™</sup>, Inc. および IBM Company の商標または登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。