システム設定

この章では、次の StarOS オプションを設定する手順について説明します。

「使用する前に」で説明したように、システムを最初に設定する手順が完了していることを前提としています。

重要 VPC-DI の展開の場合にこの章の手順を実行するには、アクティブな制御機能（CF）VM にログインする必要があります。

重要 この項の設定例で使用されているコマンドは、最もよく使用されるコマンドやキーワードのオプションです。多くの場合は、他のオプションのコマンドやキーワードオプションを使用できます。詳細については、『*Command Line Interface Reference*』を参照してください。

インターフェイスとポートの設定の確認と保存

次の StarOS CLI コマンドを入力して、インターフェイスの設定が正しいことを確認します。

show ip interface

このコマンドの出力は、次に示すようになります。この例では、*mangement1* という名前のインターフェイスがローカルコンテキストで設定されています。

```
Intf Name:       LOCAL1
Intf Type:       Broadcast
Description:     management1
VRF:             None
IP State:        UP (Bound to 1/1 untagged, ifIndex 16842753)
IP Address:      192.168.100.3        Subnet Mask:     255.255.255.0
Bcast Address:   192.168.100.255      MTU:             1500
Resoln Type:     ARP                  ARP timeout:     60 secs
L3 monitor LC-port switchover: Disabled
Number of Secondary Addresses: 0
```

次のコマンドを入力して、ポートの設定が正しいことを確認します。

show configuration port *slot/port*

VPC-DI の場合、スロットは CF または SF VM に対応します。ハイパーバイザは VPC-DI インスタンスの初期設定時に、各 VM に一意のスロット番号を割り当てます。スロット番号 1 と 2 は CF VM に割り当てられ、スロット番号 3 ～ 32 は SF VM に割り当てられます。各 SF は 1 ～ 12 の番号が付いた 4 個の vNIC と、10 ～ 21 の番号が付いた対応する仮想イーサネットポートをサポートします。SF ポート番号 10 を設定する必要があります。

この前のコマンドでは、次に示すような出力が生成されます。スロット 1 （アクティブ CF）のポート 1 の設定が表示されます。

```
config
  port ethernet 1/1
    no shutdown
    bind interface LOCAL1 local
```

「設定の確認と保存」の章の説明に従って、設定を保存します。

システムタイミングの設定

システムには、統計カウンタ、アカウンティングレコード、ロギング、およびイベント通知のタイムスタンプを提供するクロックが備わっています。システムクロックの初期設定後に、1 つ または複数の Network Time Protocol（NTP）サーバと通信するようにシステムを設定し、クロックが常に正確であることを確認できます。

すべての VPC インスタンスは、ホストが配置されている IaaS データセンターで使用されているタイミング標準に合わせる必要があります。

タイミングソースの設定に加えて、システムのタイムゾーンを設定する必要があります。

システムクロックとタイムゾーンの設定

システムクロックとタイムゾーンを設定するには、次のコマンド例を使用します。

clock set *date:time*
configure

```
clock timezone timezone [ local ]
end
```

注：

- YYYY:MM:DD:HH:mm 形式 または YYYY:MM:DD:HH:mm:ss に日付、および時刻を入力します。
- サポートされているタイムゾーンの完全なリストについては、**clock timezone** コマンドのオンラインヘルプを参照してください。
- オプションの **local** キーワードは、指定されたタイムゾーンがローカルタイムゾーンであることを示します。
- サマータイムは、そのベースになるタイムゾーンに合わせて自動的に調整されます。

「設定の確認と保存」の章の説明に従って、設定を保存します。

クロックとタイムゾーンの設定の確認と保存

時間とタイムゾーンが正しく設定されていることを確認するには、次のコマンドを入力します。

```
show clock
```

出力には、設定した日付、時刻、およびタイムゾーンが表示されます。

ネットワーク時間プロトコルのサポートの設定

この項では、Network Time Protocol（NTP）を使用できるようにするためのシステムの設定に関する情報を提供し、手順を示します。

重要 NTP サポートを実装する前に、システムクロックとタイムゾーンを設定します。これにより、NTP サーバで修正する必要がある期間が大幅に短縮されます。

（注） また、NTP は VPC VM を実行している市販（COTS）の すべてのサーバでも設定する必要があります。StarOS の NTP 設定は、COTS サーバの NTP 設定と一致している必要があります。

StarOS が提供するサービスの多くでは、NTP を介して取得する正確な時刻を保持する必要があります。StarOS で使用されている時間基準が正確でないと、サービスの信頼性が失われることがあります。このため、通常のシステム動作には NTP の設定が必要であることを前提とする必要があります。

システムは、NTP を使用して内部クロックを外部の時刻ソース（通常は GPS NTP ソース、あるいはその他の Stratum 2 または 3 サーバ、スイッチ、またはルータ）と同期します。

システムには次が必要です。

- NTP が有効になっていること。
- NTP が、ローカルコンテキスト内のみで使用されるように設定されていること。他のコンテキストを使用すると（enable は設定可能であるため指定できる）、問題が発生します。
- 少なくとも 3 台の外部 NTP サーバ用に NTP が設定されていること。サーバが 3 台以上ある場合は、サーバの異常や破損または誤設定が検出され、除外することができます。一般的に、サーバの数が多いほど（理にかなった範囲で）良好な結果が得られます。

重要 **prefer** キーワードを使用して外部 NTP サーバを設定しないでください。NTP クロック選択アルゴリズムには、最適なサーバを選択する機能がすでに組み込まれています。通常、**prefer** を使用すると、NTP 自体が決定するよりも適切な選択が行われません。

重要 Cisco TAC から指示された場合を除き、**maxpoll**、**minpoll**、または **version** キーワードの設定は変更しないでください。

必要な NTP 関連付けパラメータを設定するには、次の例を使用します。

```
configure
  ntp
    enable
    server ip_address1
    server ip_address2
    server ip_address3
    end
```

注：

- デフォルトでは *context_name* は *local* に設定されます。これが推奨されている設定です。
- **server** コマンドには多数のオプションがあります。詳細については、『*Command Line Interface Reference*』の「*NTP Configuration Mode Commands*」の章を参照してください。
- IPv4 のドット付き 10 進表記または IPv6 のコロンで区切られた 16 進表記を使用して、NTP サーバの IP アドレスを入力します。

重要 少なくとも 3 台（できれば 4 台）の NTP サーバを使用してシステムを設定します。

「設定の確認と保存」の章の説明に従って、設定を保存します。

ローカルソース使用した NTP サーバの設定

NTPは、ネットワークピア、ローカルの外部クロック（GPSデバイスなど）、または外部ソースなしのローカルクロックを使用できます。

通常、外部ソースのないローカルクロックは、より適切なクロックが使用できない場合の最終手段です。ほとんどの場合はサイトの中間 NTP サーバ上に設定されます。そのため、WAN ネットワークが停止したときに、サイト内のホストは互いに同期し続けることができます。

これは、ntpd または多くの市販の NTP デバイスに設定できます。このクロックのストラタム番号は常に上位（8+）であるため、通常の状態（実際のソースが使用可能である場合）は、このローカルクロックは使用されません。

ロード バランサの使用

NTP デーモンと NTP プロトコルは、設定された各サーバが NTP を実行していることを前提としています。NTP クライアントが、一連の実際の NTP サーバにパケットをリレーして配布するロードバランサと同期するように設定されている場合、ロードバランサはそれらのパケットを動的に配布し、NTP クライアントを混乱させる場合がありますNTP パケットは遅延とジッターに敏感です。ロードバランサを介してそれらをリレーすることは、NTPクライアントが混乱する可能性があり、サポートされている操作ではありません。

NTP の設定確認

NTP設定が正しいことを確認します。Execモードのプロンプトで次のコマンドを入力します。

show ntp associations

出力には、すべての NTP サーバに関する情報が表示されます。2 つの NTP サーバを展開する例については、次の出力を参照してください。

```
+----Peer Selection: ( ) - Rejected / No Response
|                    (x) - False Tick
|                    (.) - Excess
|                    (-) - Outlyer
|                    (+) - Candidate
|                    (#) - Selected
|                    (*) - System Peer
|                    (o) - PPS Peer
v
     remote            refid      st t when poll reach   delay   offset  jitter
==============================================================================
*10.81.254.202   .GPS.            1 u  160 1024  377   21.516    0.019   0.009
```

次の表では、**show ntp association** コマンドによって出力されるパラメータについて記述しています。

表 *1 : NTP* パラメータ

列のタイトル	説明
リモート	現在の NTP サーバのリスト。これらの文字の 1 つは、各 IP アドレスの前にあり、サーバの現在の状態を示します。 • ( ) Rejected/No response • X False tick • . Excess • - Outlyer • + Candidate • # Selected • * System peer • (o) PPS peer
refid	サーバが同期している最後に報告された NTP の参照。
st	NTP サーバのストラタムレベル。
t	通信タイプ：ブロードキャスト、マルチキャストなど
when	最終接続からの経過秒数。
poll	システムと NTP サーバ間の電話機の設定ユーティリティ。
reach	この NTP サーバへの過去 8 回のポーリングで受信された応答を示す、到達可能性シフトレジスタの 8 進値。
delay	システムと NTP サーバ間で交換されるメッセージのラウンドトリップ遅延（ミリ秒単位）。
offset	システムクロックを NTP サーバと同期させるために調整する必要があるミリ秒数。
ジッター	システムと NTP サーバ間のジッター（ミリ秒単位）。

ソフトウェア RSS の設定

Cisco Unified Computing System（USC）の NIC は、ハードウェアベースの受信側スケーリング（RSS）をサポートしています。ただし、RSS は IP トラフィックでのみサポートされています。MPLS、GTP、L2TP、GRE などの他のネットワークプロトコルでは、すべてのトラフィックが単一のキューにルーティングされます。

VPC-DI は、処理のために使用可能な vCPU コアに MPLS トラフィックを配信するソフトウェア RSS 機能を提供します。これにより、リソース使用率が増加し、スループットが向上します。

ソフトウェア RSS 機能は、Cisco UCS NIC ハードウェア RSS サポートを補足することができます。これは、ハードウェア NIC でサポートされていない一部のトラフィック（このリリースでのみ MPLS トラフィック）を配信することを意味します。VPC-DI は、包括的な RSS カバレッジを提供することもできます。つまり、すべてのトラフィックを配信します。このオプションは、RSS をサポートしていないハードウェアが使用されている場合に適用されます。

iftask sw-rss コマンドで RSS を使用するように設定します。

```
config
   iftask sw-rss {comprehensive | supplemental}
```

すべての着信トラフィックに対して RSS を設定するには、**comprehensive** キーワードを使用します。**supplemental** キーワードを使用して、ハードウェア RSS 機能でサポートされていないプロトコル（このリリースでのみ MPLS トラフィック）で RSS を設定します。

DI ネットワーク RSS の暗号化

機能の概要と変更履歴

要約データ

該当製品または機能エリア	すべて
該当プラットフォーム	VPC-DI
機能のデフォルト	無効：設定が必要
このリリースでの関連する変更点	N/A
関連資料	*VPC-DI* システム管理ガイド

マニュアルの変更履歴

重要　リリース 21.2 および N5.1 よりも前に導入された機能の改訂履歴の詳細は示していません。

改訂の詳細	リリース
Distributed Instance Network（DI ネットワーク）の RSS トラフィックのデフォルト設定は現在は無効になっていますが、新しい CLI コマンドを使用して有効にすることができます。以前のリリースでは、この機能は自動的に有効になっており、設定できませんでした。	21.8
最初の導入。	21.2 よりも前

変更された機能

以前の動作：21.8 よりも前のリリースでは、仮想化 StarOS インスタンスの内部分散インスタンスネットワーク（DI ネットワーク）上のすべてのトラフィックに対して Receive Side Scaling（RSS）がデフォルトで有効になっていました。

新しい動作：リリース 21.8 以降では、RSS はデフォルトで無効になっており、新しい CLI を介して有効にすることができます。

コマンドの変更

iftask di-net-encrypt-rss

この新しい CLI コマンドは、DI ネットワーク上の暗号化トラフィックでの RSS の有効化を制御するために追加されました。

```
configure
   [no] iftask di-net-encrypt-rss
   end
```

（注）　デフォルト設定では無効になっています。

SF ブート設定ポーズの設定

特定の状況下では、VPC-DI 展開内で、すべての SF が起動プロセスを完了する前に CF がブート設定を適用します。

次のコンフィギュレーションモードコマンドを実行すると、**wait cards active** 指定されたすべてのカードが動作可能になるか、タイムアウト期間が経過するまで（いずれかの基準が最初に満たされるまで）設定が一時停止されます。一時停止は、ローカル管理コンテキストの作成と ntp/snmp 設定の直後に行われます。

このコマンドは、シャーシのロードまたはリロード後に SF がオンラインになっており、それらの SF に関する設定が適用されない（その結果、失われる）シナリオを修正します。

```
configure
   [ no ] wait cards active { all | number } [ standby number ] timeout
seconds
   end
```

注：

- **all**：すべてのアクティブモードカードが動作ステータスを達成するまで一時停止します。
- *number*：指定した数のアクティブモードカードが動作ステータスになるまで一時停止します。*number* は、アクティブモードカードの数によって 0 になります。
- **standby** *number*：（オプション）指定した数の非アクティブモードカードが動作ステータスになるまで待機します。

  number は、0 またはアクティブモード SF 用に設定されていないサービススロットの数です。
- **timeout** *seconds*：指定されたカードセットが動作ステータスになるまで、1 ～ 3600 秒待ちます。この状態が満たされた場合に、待機が終了します。それ以外の場合は、タイムアウト期間が経過すると待機が終了します。

次のコマンド例では、すべてのアクティブカードと 1 つのスタンバイカードがアクティブになるまで、最大 120 秒待機するようにシステムに指示します。

wait cards active all standby 1 timeout 120

CLI タイムスタンプの有効化

CLI で実行されるすべてのコマンドのタイムスタンプ（日付と時刻）を表示するには、Exec モードのルートプロンプトで次のコマンドを入力します。

```
timestamps
```

日付と時刻は、このコマンドを実行するとすぐに表示されます。

「設定の確認と保存」の 章の説明に従って、設定を保存します。

CLI 確認プロンプトの設定

Exec モードとグローバル コンフィギュレーション モードのさまざまなコマンドによって、コマンドを実行する前にユーザに確認のプロンプト（[Are you sure? [Yes|No]:]）が表示されます。

この項では、次の設定について説明します。

- 現在の CLI セッション（Exec モード）またはすべての CLI セッションとユーザ（グローバル コンフィギュレーション モード）のコマンドを自動的に確認する。

- Exec モードの **configure** コマンドと **autoconfirm** コマンドに対してのみ確認を要求する。
- Exec モード設定コマンドの確認を選択的に要求する。

自動確認の有効化

autoconfirm コマンドを使用して、設定コマンドの確認プロンプトを無効にできます。**autoconfirm** コマンドは、Exec モードおよびグローバル コンフィギュレーション モードで使用できます。自動確認機能を有効にすると、設定コマンドプロンプトに「Yes」の応答が自動的に提供されます。これには、**リロード**や**シャットダウン**などの重要なコマンドが含まれます。デフォルトでは自動確認機能は無効になっています。

Exec モードでは、自動確認は現在のインタラクティブ CLI セッションにのみ適用されます。

グローバル コンフィギュレーション モードでは、すべての CLI ユーザのすべての CLI セッションに自動確認が適用されます。

```
configure
   autoconfirm
   end
```

自動確認を有効にした後に無効にするには、**no autoconfirm** コマンドを使用します。

重要　コマンドガードが有効になっている場合、自動確認はコマンドガードを無効にします。

自動確認は「ease-of-use」機能として意図されています。「Are you sure?」という答えがあることを前提としています。[Y/N]" プロンプトは「Yes」になり、プロンプトをスキップします。この使用は、ユーザがこれらの「safety-net」プロンプトを必要としないエキスパートであることを意味します。

autoconfirm コマンドと configure コマンドに対する確認の要求

autoconfirm（Exec モードとグローバル コンフィギュレーション モード）コマンドと、グローバル コンフィギュレーション モードの **commandguard** コマンドを介した **configure**（Exec モード）コマンドに対する確認プロンプトが必要な場合があります。

重要　autoconfirm が有効になっている場合、Exec モードとグローバル コンフィギュレーション モードの両方で autoconfirm が無効になるまで commandguard は有効になりません。

次のコマンドシーケンスは、commandguard 機能を有効にします。

```
configure
   commandguard
   end
```

commandguard を有効にすると、次の例に示すように確認プロンプトが表示されます。

```
[local]host_name# configure
Are you sure? [Yes|No]: yes
[local]host_name(config)#
```

commandguard がいったん有効になった後に無効にするには、**no commandguard** コマンドを使用します。

commandguard のステータスは、**show configuration** コマンドで出力されます。

特定の Exec モードのコマンドに対する確認の要求

commandguard コマンドのキーワードを使用すると、自動確認が有効になっている場合でも、Exec モード コンフィギュレーション コマンドの特定のカテゴリに対して必須のプロンプトを適用できます。

コマンド構文は次のとおりです。

```
configure
   commandguard exec-command exec_mode_category
   end
```

注：

- **exec-command** *exec_mode_category* は、Exec モード コンフィギュレーション コマンドの次のいずれかのカテゴリを指定します。
  - card
  - clear
  - copy
  - debug
  - delete
  - filesystem
  - hd
  - reload
  - rename
  - shutdown
  - task
  - upgrade
- 複数の **commandguard exec-command** *exec_mode_category* コマンドを入力できます。
- 自動確認が有効になっているかどうかに関係なく、指定したカテゴリワードで始まる Exec モードのすべてのコマンドに確認のプロンプトを表示します。
- **no commandguard exec-command** *exec_mode_category* コマンドを使用して、特定のカテゴリの確認プロンプトをオフにすることができます。
- Exec モードコマンドに対する **commandguard exec-command** で自動確認をオーバーライドした場合、StarOS はコマンドを実行しようとしたときに自動確認がオーバーライドされた理由を示す情報メッセージを表示します。
- **-noconfirm** キーワードをサポートする Exec モード コンフィギュレーション コマンドに対する確認プロンプトのオーバーライドを選択できます。

たとえば、**commandguard exec-command card** が有効になっている場合、確認プロンプトは次のように表示されます。

```
[local]host_name# card busy-out 1
Info: commandguard prevents autoconfirm of this command
Are you sure? [Yes|No]: yes
[local]host_name#
```

システム管理ユーザの設定

はじめに、システムのコンテキストレベルのセキュリティ管理者を設定する方法について説明します。

この項では、次の権限を持つ追加の管理ユーザを設定する手順について説明します。

- **セキュリティ管理者**：読み取り/書き込み権限を持ち、管理者、演算子、およびインスペクターで使用できるものを含め、すべての CLI コマンドを実行できます。
- **管理者**：読み取り/書き込み権限を持ち、CLI で任意のコマンドを実行できます。ただし、セキュリティ管理者だけが設定できるセキュリティ関連のコマンドは実行できません。管理者は、システム設定を構成または変更したり、オペレータやインスペクタで使用可能なコマンドを含め、すべてのシステムコマンドを実行したりできます。
- **オペレータ**：Execモードコマンドのより大きなサブセットに対する読み取り専用権限を持ちます。これらは、インスペクタモードの一部であるすべてのコマンドと、システムモニタリング、統計情報、および障害管理機能の一部を実行できます。オペレータには、コンフィギュレーションモードを開始する機能がありません。
- **インスペクタ**：いくつかの読み取り専用 Exec モードコマンドに制限されます。これらのほとんどは、さまざまな統計情報や条件を表示するための **show** コマンドです。インスペクタは、**show configuration** コマンドを実行できず、設定モードを開始する権限もありません。

設定手順は、管理ユーザのタイプ（コンテキストレベルまたはローカルユーザ）に従って分類されます。

重要 これらのユーザ権限とタイプの違いについては、「はじめに」を参照してください。

ユーザ名の文字制限

ユーザ名には、英数字（a-z、A-Z、0-9）、ハイフン、下線、およびピリオドだけを含めることができます。ハイフン文字を最初の文字にすることはできません。これは、AAA ユーザ名とローカルユーザ名に適用されます。

これらの標準規格に準拠していないユーザ名を作成しようとすると、「Invalid character」というメッセージが表示されます。有効な文字「0123456789.-_abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ」です。

コンテキストレベルの管理ユーザの設定

このユーザタイプはコンテキストレベルで設定され、ログイン時にユーザ名とパスワードを検証するために AAA サブシステムに依存します。これは、設定ファイルを介してローカルに設定された管理ユーザアカウント、または外部 RADIUS サーバまたは TACACS+ サーバの両方に当てはまります。これらのユーザタイプのパスワードは一度割り当てられ、設定ファイルでアクセスできます。

この項では、コンテキストレベルの管理ユーザタイプを設定するための情報と手順について説明します。

アカウントごとまたは認証方式ごとにシミュレーションの CLI セッションの最大数を設定することができます。セキュリティ設定と属性に影響を与えたり、システムのサービス、安定性、およびパフォーマンスに悪影響を及ぼす可能性がある特定のアカウントを保護します。同時 CLI セッションの最大数は、新しいローカルユーザログインと新しい AAA コンテキストベースのログインを試行するときに設定できます。セッションの最大数が 0 に設定されている場合は、ログインタイプに関係なくユーザが認証されます。CLI タスクが開始されると、カウントを識別するためのチェックが完了します。この場合、CLI は、そのユーザのセッション数が 0 よりも大きい 1 であると判断し、出力にエラーメッセージを表示し、starCLIActiveCount オブジェクトおよび starCLIMaxCount SNMP MIB オブジェクトと、starGlobalCLISessionsLimit アラームおよび starUserCLISessionsLimit SNMP MIB アラームを生成します。

ローカルユーザのユーザ名におけるグローバル コンフィギュレーション モードのコマンドの **max sessions** キーワードは、ローカルユーザに使用可能な同時セッションの最大数を設定します。

最大セッションにおけるコンテキスト コンフィギュレーション モードのコマンドを使用すると、管理ユーザは、対応するユーザに許可される最大同時セッション数を設定できます。

これらのコマンドの詳細については、『*Command Line Interface Reference* 』を参照してください。

コンテキストレベルのセキュリティ管理者の設定

追加のセキュリティ管理者を設定するには、次の例を使用します。

```
configure
  context local
    administrator user_name { [ encrypted ] [ nopassword ] password password
 }
     end
```

注：

- 追加のキーワードオプションを使用して、アクティブな管理者を識別したり、管理者に時間しきい値を設定したりできます。**administrator** コマンドの詳細については、『*Command Line Interface Reference*』を参照してください。
- **nopassword** オプションを使用すると、関連付けられたパスワードを使用せずに管理者を作成できます。認証の唯一の手段として ssh 公開キー（SSH コンフィギュレーションモードの **authorized key** コマンド）を使用する場合は、このオプションを有効にします。このオプションを有効にすると、管理者パスワードを使用してユーザアカウントにアクセスできなくなります。

「設定の確認と保存」の章の説明に従って、設定を保存します。

コンテキストレベルの管理者の設定

次の例を使用して、コンテキストレベルの設定管理者を設定します。

```
configure
  context local
    config-administrator user_name { [ encrypted ] [ nopassword ] password
 password }
     end
```

注：

- 追加のキーワードオプションを使用して、アクティブな管理者を識別したり、管理者に時間しきい値を設定したりできます。**config-administrator** コマンドの詳細については、『*Command Line Interface Reference*』を参照してください。
- **nopassword** オプションを使用すると、関連付けられたパスワードを使用せずに設定管理者を作成できます。認証の唯一の手段として ssh 公開キー（SSH コンフィギュレーションモードの **authorized key** コマンド）を使用する場合は、このオプションを有効にします。このオプションを有効にすると、設定管理者パスワードを使用してユーザアカウントにアクセスできなくなります。

「設定の確認と保存」の章の説明に従って、設定を保存します。

コンテキストレベルのオペレータの設定

次の例を使用して、コンテキストレベルのオペレータを設定します。

```
configure
  context local
    operator user_name { [ encrypted ] [ nopassword ] password password }
    end
```

注：

- 追加のキーワードオプションを使用して、アクティブな管理者を識別したり、管理者に時間しきい値を設定したりできます。オペレータコマンドの詳細については、『**Command Line Interface Reference**』を参照してください。
- **nopassword** オプションを使用すると、関連付けられたパスワードなしでオペレータを作成できます。認証の唯一の手段として ssh 公開キー（SSH コンフィギュレーションモードの **authorized key** コマンド）を使用する場合は、このオプションを有効にします。このオ

プションを有効にすると、オペレータパスワードを使用してユーザアカウントにアクセスできなくなります。

「設定の確認と保存」の章の説明に従って、設定を保存します。

コンテキストレベルのインスペクタの設定

次の例を使用して、コンテキストレベルのインスペクタを設定します。

```
configure
   context local
     inspector user_name { [ encrypted ] [ nopassword ] password password
}
     end
```

注：

- 追加のキーワードオプションを使用して、アクティブな管理者を識別したり、管理者に時間しきい値を設定したりできます。**Inspector** コマンドの詳細については、『*Command Line Interface Reference*』を参照してください。
- **nopassword** オプションを使用すると、関連付けられたパスワードを使用することなくインスペクタを作成できます。認証の唯一の手段として ssh 公開キー（SSH コンフィギュレーション モードの **authorized key** コマンド）を使用する場合は、このオプションを有効にします。このオプションを有効にすると、インスペクタパスワードを使用してユーザアカウントにアクセスできなくなります。

「設定の確認と保存」の章の説明に従って、設定を保存します。

システムと LI 設定の分離

合法的傍受（LI）の設定には、機密情報が含まれています。通常のビルドのデフォルトでは、li-administration 権限のない管理者が LI 設定コマンドを表示できます。ただし、LI 設定コマンドの表示は、システム設定の残りの部分から制限したり、分離したりすることができます。

グローバル コンフィギュレーション モードの **require segregated li-configuration** コマンドによって、システムおよび合法的傍受の CLI の表示は永続的に分離されます。合法的傍受のキーワードを指定した CLI コマンドは暗号化され、li-administration 権限を持つ管理者によってのみ表示できます。

重要 信頼できるビルドでは、LI の分離はオンになっており、無効にすることはできません。**require segregated li-configuration** コマンドは非表示です。

システム設定から LI 設定を分離すると、StarOS に次の影響があります。

- li-administration 権限を持つ管理者のみが、**show configuration** コマンドの出力で合法的傍受の CLI コマンドを確認できます。
- **save configuration** コマンドを実行すると、合法的傍受の CLI 設定コマンドが自動的に暗号化されます。

- CLI コマンドを介して保存されたコンフィギュレーションファイルをロードすると（たとえば、**configure** <*url*>）、暗号化済み合法的傍受の CLI コマンドは、LI 権限を持つ管理者に対してのみ復号および実行されます。LI 権限のない管理者の場合は、暗号化済み合法的傍受の CLI コマンドは復号および実行されません。
- 起動設定がロードされているシステム起動中に、暗号化済み合法的傍受の設定が復号され、サイレントにロードされます。つまり、合法的傍受の CLI 設定はコンソールポートに表示されません。
- Exec モードの **configure** コマンドは、LI 管理者が保存されたコンフィギュレーション ファイルから、暗号化された合法的傍受の設定のみをロードできるようにするキーワードをサポートするようになりました（たとえば、**configure encrypted** <*url*> など）。**暗号化された**キーワードは、LI 管理者のみが実行できます。
- 暗号化済み合法的傍受の設定（分離された LI）を使用してシステムを実行している場合、**show boot initial config** コマンドの出力には、最初の起動時に 2 番目のパスを実行する必要があるかどうかを示す行が含まれています。この行には、暗号化済み合法的傍受の設定が処理された場合は「encrypted li」と表示されます。その行に「encrypted li errors」と表示されている場合は、2 番目のパスが正しく動作しなかったか、予期されていないまたは本来の情報ではない出力が提供されています。
- li-administration 権限を持つユーザは、**show logs encrypted-li** コマンドを使用して、暗号化済み合法的傍受の設定におけるブート設定の出力を表示できます。

グローバル コンフィギュレーション モードの **require segregated li-configuration** コマンドと関連コマンドの詳細については、『*Lawful Intercept Configuration Guide*』の付録「*Lawful Intercept CLI Commands*」を参照してください。

（注） 『*Lawful Intercept Configuration Guide*』は、www.cisco.com では利用できません。このガイドのコピーを入手するには、シスコのアカウント担当者にお問い合わせください。

リリース 21.4 以降（信頼できるビルドのみ）：

- ユーザは、それぞれのコンテキストインターフェイスを使用してのみシステムにアクセスできます。
- ユーザが別のコンテキストインターフェイスを使用してそれぞれのコンテキストにログインしようとすると、そのユーザは拒否されます。
- ユーザが「authorized-keys」または「allowusers」のいずれかのコンテキストに設定されているかどうかに関係なく、この機能では、独自のコンテキストインターフェイス以外のコンテキストインターフェイスを介してログインを試行すると、これらのユーザは拒否されます。
- ローカル以外のコンテキストに設定されているユーザは、ログインを試行するコンテキストを指定する必要があります。次に例を示します。

  ssh username@ctx_name@ctx_ip_addrs

コンテキストレベルの管理ユーザ設定の確認

設定が正常に行われたことを確認するには、次のコマンドを入力します。

show configuration context local

このコマンドは、このセッション時にローカルコンテキスト内で変更したすべての設定パラメータを表示します。次に、このコマンドの出力例を示します。この例では、*testadmin* という名前のセキュリティ管理者が設定されています。

```
config
  context local
    interface mgmt1
      ip address 192.168.1.10 255.255.255.0
    #exit
    subscriber default
    #exit
    administrator testadmin encrypted password fd01268373c5da85
    inspector testinspector encrypted password 148661a0bb12cd59
exit
  port ethernet 5/1
    bind interface mgmt1 local
  #exit
```

ローカルユーザの管理ユーザの設定

ローカルユーザタイプは、ANSI T1.276-2003 のパスワードセキュリティ保護をサポートしています。パスワード、パスワード履歴、ロックアウト状態などのローカルユーザアカウント情報は、/flash で維持されます。この情報は、AAA ベースの認証の対象となる個別のローカルユーザデータベースにただちに保存され、システムの残りの部分では使用されません。そのため、設定されているローカルユーザアカウントは、システム設定の残りの部分では表示されません。

重要 リリース 20.0 以降の信頼できる StarOS ビルドでは、ローカルユーザデータベースは無効になっています。グローバル コンフィギュレーション モードの **local-user** コマンドと Exec モードの **show local-user** コマンド、および **update local-user** コマンドは使用できません。信頼できるビルドの詳細については、「システム操作と設定」の章を参照してください。

次の例を使用して、ローカルユーザの管理ユーザを設定します。

```
configure
   local-user username name
   end
```

注：

- 追加のキーワードオプションを使用して、アクティブな管理者を特定したり、管理者に時間のしきい値を設定したりできます。*local-user username* コマンドの詳細については、『**Command Line Interface Reference**』を参照してください。

ローカルユーザデータベースの詳細については、ローカルユーザデータベースの更新とダウングレード （18 ページ）を参照してください。

ローカルユーザ設定の確認

設定が正常に行われたことを確認するには、次のコマンドを入力します。

show local-user verbose

このコマンドは、設定されたローカルユーザの管理ユーザに関する情報を表示します。次に、このコマンドの出力例を示します。この例では、*SAUser* という名前のローカルユーザが設定されています。

```
Username:                SAUser
  Auth Level:            secadmin
  Last Login:            Never
  Login Failures:        0
  Password Expired:      Yes
  Locked:                No
  Suspended:             No
  Lockout on Pw Aging:   Yes
  Lockout on Login Fail: Yes
```

ローカルユーザデータベースの更新

次の Exec モードコマンドを実行して、ローカルユーザ（管理）設定を更新します。このコマンドは、管理ユーザを作成、削除、または編集した直後に実行する必要があります。

update local-user database

ローカルユーザデータベースの更新とダウングレード

20.0 よりも前のリリースでは、ローカルユーザのパスワードは MD5 メッセージ ダイジェスト アルゴリズムでハッシュされ、ローカルユーザデータベースに保存されていました。リリース 20. 0 では、PBKDF2（パスワードベースのキー派生関数バージョン 2）を使用し、入力されたデータ、ソルト、および反復回数に基づいて、特定の長さのキーを取得するようになりました。ローカルユーザアカウントのパスワードは、ランダムに生成されたソルトと多くの反復回数を備えた PBKDF2 方式を使用してハッシュされ、パスワードストレージの保護が強化されます。

リリース 20.0 にアップグレードする場合、ローカルユーザデータベース内の既存のユーザパスワードは MD5 から PBKDF2 ハッシングに自動的にアップグレードされません（ハッシュされたパスワード値のみが保存されます）。ハッシュ関数は一方向であるため、保存されたハッシュ値からユーザパスワードを取得することはできません。したがって、既存のハッシュパスワードを強力なハッシュパスワードに自動的に変換することはできません。

データベースを更新するには、セキュリティ管理者が Exec モードの **update local-user database** CLI コマンドを実行する必要があります。このコマンドを実行すると、StarOS は /flash ディレクトリからデータベースを読み取り、データベースを新しい形式で再構築してディスクに書き込みます。

データベースのアップグレードプロセスでは、MD5 ハッシュパスワードが PBKDF2 形式に自動的に変換されることはありません。StarOS は、古い暗号化アルゴリズムを使用してユーザを

認証し続けます。これは、「弱いハッシュ」フラグの古い暗号化アルゴリズムを使用してユーザにフラグを付けます。このフラグは、 EXEC モードの **show local-user [verbose]** CLI コマンドの出力に表示されます。ユーザがクレデンシャルを使用して再ログインすると、StarOS は MD5 アルゴリズムを使用して入力したパスワードが確認され、PBKDF2 アルゴリズムを使用して新しいハッシュが作成されて、その結果がデータベースに保存されます。その後、StarOS はそのユーザの「弱いハッシュ」フラグをクリアします。

重要 ハッシュ関数は一方向であるため、PBKDF2 でハッシュされたパスワードを MD5 形式に変換することはできません。ローカルユーザデータベースは、20.0 よりも前の StarOS リリースに戻す前にダウングレードする必要があります。

MD5 ハッシュアルゴリズムを使用するようにローカルユーザデータベースをダウングレードするには、セキュリティ管理者が Exec モードの **downgrade local-user database** コマンドを実行する必要があります。StarOS は確認のプロンプトを表示し、パスワードを再入力するようセキュリティ管理者に要求します。ダウングレードコマンドを実行する前に、ユーザは入力したパスワードで再認証されます。確認後、パスワードは適切な古いまたは弱い暗号化アルゴリズムを使用してハッシュされ、データベースに保存されて、以前のバージョンの StarOS がセキュリティ管理者を認証できるようになります。

ダウングレードプロセスでは、PBKDF2 でハッシュされたパスワードは MD5 形式に変換されません。ダウングレードプロセスでは、（/flash ディレクトリから）データベースが再度読み込まれ、古い形式でデータベースが再構築されてからディスクに書き込まれます。PBKDF2 でハッシュされたパスワードは MD5 ハッシュアルゴリズムに変換できず、以前の StarOS リリースでは PBKDF2 暗号化アルゴリズムを解析できないため、StarOS は PBKDF2 アルゴリズムを介して暗号化されたすべてのユーザを一時停止します。MD5 アルゴリズム（「弱いハッシュ」フラグ）を介して暗号化されたユーザは、ログイン情報を使用してログインを続行できます。システムが以前の StarOS リリースで起動した後、一時停止されたユーザは **show local-user [verbose]** コマンドの出力で確認できます。

一時停止されたユーザを再アクティブ化するには、セキュリティ管理者が次の手順を実行します。

- Exec モードの **password change local-user** *username* コマンドを使用して、一時停止されたユーザの一時パスワードを設定します。
- コンフィギュレーション モードの **no suspend local-user** *username* コマンドを使用して、ユーザの一時停止フラグをリセットします。

指定されたルートディレクトリへのユーザアクセスの制限

デフォルトでは、FTP/SFTP アクセス権を持つ管理者ユーザは、/mnt/user/ ディレクトリの下にある任意のファイルにアクセスし、変更を加えることができます。次のディレクトリに「オールオアナッシング」ベースでアクセスが許可されます。/flash、/cdrom、/hdraid、/records、/usb1、/usb2

管理者または設定管理者は、ファイルディレクトリとアクセス権限を使用して SFTP サブシステムのリストを作成できます。ローカルユーザが作成されると、管理者は SFTP サブシステムを割り当てます。ユーザの承認レベルがセキュリティ管理者または管理者でない場合、ユーザは読み取り専用権限を持つサブシステムにのみアクセスできます。このディレクトリは、ユーザのルートディレクトリとして使用されます。この情報は、openssh sftp サーバに渡される環境変数として設定されます。

SFTP ルートディレクトリは、ローカルユーザ、管理者、および設定管理者と関連付ける前に作成する必要があります。複数の SFTP ディレクトリを作成できます。各ディレクトリは、1 人以上のユーザに割り当てることができます。

SFTP ルートディレクトリの設定

subsystem sftp コマンドを使用すると、SFTP ルートディレクトリおよび関連付けられているアクセス権限レベルを割り当てることができます。

```
configure
   context local
      server sshd
        subsystem sftp  [ name sftp_name root-dir pathname mode { read-only
 | readwrite } ]
```

注：

- *sftp_name* は、このサブシステムを一意に識別する英数字の文字列です。
- *pathname* は、SFTP ファイルの転送先となるルートディレクトリを指定します。次のオプションがあります。
  - /hd-raid/records/cdr
  - /flash

SFTP ルートディレクトリとローカルユーザの関連付け

local-user username コマンドを使用すると、管理者は SFTP ルートディレクトリを指定されたユーザ名に関連付けることができます。

```
configure
  local-user username user_name authorization-level level ftp sftp-server
 sftp_name password password
  exit
```

SFTP ルートディレクトリと管理者の関連付け

administrator コマンドを使用すると、管理者は、指定された管理者に SFTP ルートディレクトリを関連付けることができます。

```
configure
   context local
     administrator user_name password password ftp sftp-server sftp_name
     exit
```

SFTP ルートディレクトリと設定管理者の関連付け

config-administrator コマンドを使用すると、管理者は、指定された設定管理者に SFTP ルートディレクトリを関連付けることができます。

```
configure
   context local
    config-administrator user_name password password ftp sftp-server sftp_name
     exit
```

システム管理ユーザ用の TACACS+ の設定

ここでは、VPC-DI の TACACS+（Terminal Access Controller Access Control System+）AAA（認証、許可、およびアカウンティング）サービスの機能と設定について説明します。

動作

TACACS+は、セキュアな暗号化されたプロトコルです。管理ユーザアカウントデータベースを使用してプロビジョニングされたTACACS+サーバにリモートでアクセスすることで、VPC-DI システムはシステム管理ユーザに TACACS+ AAA サービスを提供できます。TACACS+ は、UDP ではなく TCP を使用する TACACS プロトコルの拡張バージョンです。

システムは、TACACS+ ネットワークアクセスサーバ（NAS）として機能します。NAS として、承認されたシステム管理ユーザに代わって、システムが TACACS+ AAA サービスを要求します。正常に認証されるには、システムがアクセスするのと同じローカルコンテキストとネットワーク内にTACACS+ サーバが存在する必要があります。

システムは、TACACS+マルチ接続モードをサポートしています。マルチ接続モードでは、セッションごとに TACACS+ サーバへの個別のプライベート TCP 接続が開き、維持されます。TACACS+ セッションが終了すると、サーバへの接続が終了します。

TACACS+ は、VPC-DI 上のシステム全体を網羅する機能です。TACACS+ AAA サービスの設定は、TACACS コンフィギュレーション モードで実行されます。TACACS+ 機能の有効化は、グローバル コンフィギュレーション モードで実行されます。システムでは、最大 3 台の TACACS+ サーバの設定がサポートされます。

システムで設定して有効にすると、TACACS+ 認証が最初に試行されます。デフォルトでは、TACACS+ 認証が失敗すると、システムは RADIUS などの TACACS+ 以外の AAA サービスを使用してユーザの認証を試行します。

アカウントごとまたは認証方式ごとにシミュレーションのCLIセッションの最大数を設定することができます。セキュリティ設定と属性に影響を与えたり、システムのサービス、安定性、およびパフォーマンスに悪影響を及ぼす可能性がある特定のアカウントを保護します。同時 CLI セッションの最大数は、新しい TACACS+ ユーザログインを試行するときに設定できます。最大セッション機能を使用するには、TACACS＋サーバ属性オプションのオプションを使用することを**推奨します**。2 番目の方法は、**user-id** コマンドに **maxsess** キーワードを指定して StarOS CLI コンフィギュレーションモードの TACACS+ モードを使用することです。セッションの最大数が 0 に設定されている場合は、ログインタイプに関係なくユーザが認証されます。

CLI タスクが開始されると、カウントを識別するためのチェックが完了します。この場合、CLI は、そのユーザのセッション数が 0 よりも大きい 1 であると判断し、出力にエラーメッセージを表示し、starCLIActiveCount オブジェクトおよび starCLIMaxCount SNMP MIB オブジェクトと、starGlobalCLISessionsLimit アラームおよび starUserCLISessionsLimit SNMP MIB アラームを生成します。

max-sessions TACACS+ コンフィギュレーション モード コマンドは、TACACS+ で使用可能なセッションの最大数を設定します。また、 **user-id** TACACS+ コンフィギュレーション モード コマンドの **default** オプションは、特定の TACACS+ ユーザ識別子にデフォルトの属性を設定します。 これらのコマンドの詳細については、『*Command Line Interface Reference* 』を参照してください。

重要 ユーザは、StarOS と TACACS+ サーバの両方の設定で使用可能なシミュレーション CLI セッションの最大数を定義できます。ただし、このオプションを使用しないことを強く推奨します。

重要 *15.0 MR4* よりも後のリリースの場合、TACACS+ アカウンティング（CLI イベントロギング）は、権限レベルが 15 と 13 に設定されている合法的傍受ユーザに対しては生成されません。

ユーザアカウントの要件

TACACS+ AAA サービスを設定する前に、次の TACACS+ サーバと StarOS ユーザアカウントのプロビジョニング要件に注意してください。

TACACS+ ユーザアカウントの要件

TACACS+ サーバは、次の TACACS+ ユーザアカウント情報を使用してプロビジョニングする必要があります。

- 既知の管理ユーザのリスト。
- 各ユーザのプレーンテキストまたは暗号化されたパスワード。
- 各ユーザが属するグループの名前。
- ユーザグループのリスト。
- 各グループで許可/拒否される TACACS+ の特権レベルとコマンド。

重要 TACACS+ の権限レベルは、ネットワークの TACACS+ サーバデータベースに属性値ペア（AVP）として保存されます。ユーザは、権限レベルに関連付けられた一連のコマンドに制限されます。次の表で、TACACS+ の権限レベルを StarOS CLI の管理ロールと責任にマッピングする方法を説明します。

CLI 管理ロールへの TACACS+ 権限レベルのデフォルトマッピングを表示するには、Exec モードの **show tacacs priv-lvl** コマンドを実行します。デフォルトのマッピングは、StarOS リリースとビルドタイプによって異なります。

Tacacs+ の権限レベルをデフォルトの StarOS 認証値から再設定するには、TACACS+ コンフィギュレーションモードの **priv-lvl** コマンドと **user-id** コマンドを使用します。詳細については、『*Command Line Interface Reference*』の「*TACACS+ Configuration Mode Commands*」の章を参照してください。

重要 リリース 20.0 以降の信頼できる StarOS のビルドでは、FTP はサポートされていません。

StarOS ユーザアカウントの要件

システムへの管理アクセスが許可されている TACACS+ ユーザには、StarOS で次のユーザアカウント情報が定義されている必要があります。

- ユーザ名
- パスワード
- 管理ロールと権限

重要 システムでユーザと管理者の権限を定義する手順については、「システム管理ユーザの設定」を参照してください。

TACACS+ AAA サービスの設定

この項では、システム上の管理ユーザに TACACS+ AAA サービスを設定する方法について説明します。

注意 TACACS+ AAA サービスを初めて設定する場合は、管理者ユーザが StarOS にログインするために TACACS+ 以外のサービスを使用する必要があります。そうしないと、TACACS+ ユーザはシステムへのアクセスが拒否されます。

TACACS+ 以外のサービスを使用してシステムにログインします。

システムで TACACS+ AAA サービスを設定するには、次の例を使用します。

```
configure
   tacacs mode
      server priority priority_number ip-address tacacs+srvr_ip_address
      end
```

（注）

- **server priority** *priority_number*：1 ～ 3 の整数（*18.2* よりも前のリリース）または 1 ～ 4 の整数（*18.2*+ のリリース）である必要があります。これは、この TACACS+サーバが TACACS+ 認証を使用する順序を指定します。優先順位は 1 が最も高く、3 または 4 が最も低くなります。優先順位の数値は、設定されている TACACS+ サーバに対応します。
- **ip-address**：有効な TACACS+ サーバの IPv4 アドレスである必要があります。これを使用し、TACACS+AAA サービスを介してこのシステムにアクセスする管理ユーザを認証します。
- デフォルトでは、TACACS+設定によって、認証、許可、およびアカウンティングの各サービスが提供されます。

StarOS で TACACS+ を有効にします。

```
configure
   aaa tacacs+
   end
```

詳細については、コンソールの TACACS+ 認証の無効化 （26 ページ）を参照してください。

「設定の確認と保存」の章の説明に従って、設定を保存します。

重要 すべての TACACS+ コンフィギュレーション モードのコマンドとオプションの詳細については、『*Command Line Reference*』の「*TACACS Configuration Mode Commands*」の章を参照してください。

非ローカル VPN 認証のための TACACS+ の設定

デフォルトでは、TACACS+ 認証はローカルコンテキストへのログインに関連付けられています。TACACS+認証は、非ローカルコンテキストVPNログイン用に設定することもできます。TACACS+ は、以下で説明するオプションを使用して設定し、有効にする必要があります。

stop キーワードオプションは、TACACS+ コンフィギュレーション モードの **on-unknown-user** コマンドで使用できます。TACACS+ がコマンドキーワードオプションで有効になっている場合、ユーザがログインを試行する VPN コンテキスト名は、ユーザ名文字列で指定された VPN 名と一致している必要があります。コンテキスト名が一致しない場合、ログインは失敗して終了します。

このオプションを指定しないと、ログインシーケンスは代替ログイン方式を介して別のコンテキストで認証を試行します。たとえば、**on-unknown-user stop** の設定がない場合、管理者アカウントは、非ローカルの VPN コンテキストを介してローカルコンテキストにログインできま

す。ただし、**on-unknown-user stop** の設定では、ローカルコンテキストのログインは試行されず、管理者アカウントのログイン認証は失敗します。

```
configure
   tacacs mode
      on-unkown-user stop &quest;
      end
```

TACACS+ の設定の確認

この項では、TACACS+ 設定の確認方法について説明します。

システム CLI からログアウトし、TACACS+ サービスを使用して再度ログインします。

重要 StarOS で TACACS+ AAA サービスを設定して有効にすると、システムは最初に TACACS+ AAA サービスを介して管理ユーザを認証しようとします。デフォルトでは、TACACS+ 認証が失敗した場合、システムは TACACS+ AAA 以外のサービスを使用して認証を続行します。

Exec モードのプロンプトで、次のコマンドを入力します。

```
show tacacs [ client | priv-lvl | session | summary ]
```

show tacacs コマンドの出力には、ユーザ名、ログイン時刻、ログインステータス、現在のセッション状態、権限レベルなどのサマリー情報がアクティブな TACACS+ セッションごとに表示されます。 オプションのフィルタキーワードによって、追加情報が表示されます。

次に、このコマンドの出力例を示します。この例では、*asradmin* という名前のシステム管理ユーザが、TACACS+ AAA サービスを介してシステムに正常にログインしています。

```
active session #1:
   login username               : asradmin
   login tty                    : /dev/pts/1
   time of login                : Fri Oct 22 13:19:11 2011
   login server priority        : 1
   current login status         : pass
   current session state        : user login complete
   current privilege level      : 15
   remote client application    : ssh
   remote client ip address     : 111.11.11.11
   last server reply status     : -1
total TACACS+ sessions          : 1
```

重要 すべての TACACS+ メンテナンスコマンドの詳細については、『*Command Line Interface Reference*』を参照してください。

TACACS+ サーバでの IPv6 アドレスサポート

認証方式の分離

コンソールポートにアクセスし、SSH/telnet セッション（vty 回線）を確立するための個別の認証方式を設定できます。

TACACS+ をグローバルに設定すると、コンソールおよび vty 回線へのアクセスは両方ともその方式を使用して認証されます。

コンソールポートは StarOS への最後のリゾートアクセスであるため、コンソールのローカル認証を設定し、vty 回線に TACACS+ を使用することができます。

重要　この機能は、AAA（認証、許可、およびアカウンティング）サービス、およびローカルユーザに対して拡張されます。たとえば、ローカルユーザは、vty 回線を介してのみアクセス権を持つコンソールアクセスおよび AAA（VPN コンテキスト）ユーザのみを持つことができます。

認証方式（コンソールと vty 回線）を分離するには、認証のタイプに基づいてユーザのコンソールアクセスを無効にする必要があります。

コンソールの TACACS+ 認証の無効化

グローバル コンフィギュレーション モードの **aaa tacacs+** コマンドの **noconsole** キーワードは、コンソール回線での TACACS+ 認証を無効にします。

```
configure
  aaa tacacs+ noconsole
  exit
```

デフォルトでは、TACACS+ サーバ認証は、コンソールまたは vty 回線からのログインに対して実行されます。**noconsole** を有効にすると、コンソール回線のローカルデータベース認証を優先して TACACS+ 認証がバイパスされます。vty 回線では、TACACS+ は有効のままになります。

重要　**aaa tacacs+ noconsole** が設定されている場合は、有効なログイン情報を持つローカルユーザがコンソールポートにログインできます。**on-authen-fail stop** と **on-unknown-user stop** が TACACS+ コンフィギュレーションモードでイネーブルになっている場合でも、コンソールポートにログイン可能です。ユーザが TACACS+ ユーザでない場合、vty 回線でログインできません。

コンソールでの AAA ベースの認証の無効化

グローバル コンフィギュレーション モードの **local-user allow-aaa-authentication** コマンドの **noconsole** キーワードは、コンソール回線での AAA ベースの認証を無効にします。

```
configure
  local-user allow-aaa-authentication noconsole
  exit
```

ローカルユーザ認証は、AAA ベースの認証と **local-user allow-aaa-authentication noconsole** が有効になる前に常に実行されるため、**no local-user allow-aaa-authentication** が設定されている場合と同じに動作します。VTY 回線には影響しません。

重要 このコマンドは、ローカルで使用されているデータベースが使用できないため、信頼できるビルドには適用されません。

コンテキストレベルで TACACS+ 認証を無効化

グローバル コンフィギュレーション モードで **aaa tacacs+**を有効にすると、TACACS+ 認証はすべてのコンテキスト（ローカルおよび非ローカル）に自動的に適用されます。一部のネットワーク展開では、特定のコンテキストに対して TACACS+ サービスを無効にすることができます。

コンテキスト内で TACACS+ サービスを無効にするには、**no aaa tacacs+** コンテキスト設定コマンドを使用します。

```
configure
  context ctx_name
    no aaa tacacs+
```

以前に無効にされていたコンテキスト内で TACACS+ サービスを有効にするには、**aaa tacacs+** コンテキスト設定コマンドを使用します。

重要 コンテキストレベルでサービスを選択的に無効にするには、その前に、グローバル コンフィギュレーション モード（すべてのコンテキスト）で AAA TACACS+ サービスを有効にする必要があります。グローバルに有効になっていない場合は、コンテキストレベルで TACACS+ サービスを選択的に有効にすることはできません。

コンソール/VTY 回線でのローカルユーザログインの制限

StarOS ユーザを作成するときのセキュリティ管理者は、そのユーザがコンソールを使用してログインするか、または VTY 回線を使用してログインするかを指定できます。グローバル コンフィギュレーション モードの **local-user username** コマンドの **[noconsole | novty]** キーワードはこれらのオプションをサポートしています。

```
configure
  local-user username <username> [ noconsole | novty ]
  exit
```

noconsole キーワードを指定すると、ユーザはコンソールポートにログインできなくなります。**novty** キーワードを指定すると、ユーザは SSH または telnet セッションを介してログインできなくなります。どちらのキーワードも指定されていない場合は、コンソールと VTY 回線の両方へのアクセスが許可されます。

重要 **noconsole** キーワードまたは **novty** キーワードの使用は、新しいローカルユーザデータベース形式でのみサポートされています。**update local-user database** を実行していない場合は、これらのキーワードを有効にする前に実行する必要があります。そうしないと、**noconsole** キーワードと **novty** キーワードはローカルユーザデータベースに保存されません。システムの再起動後も、すべてのユーザがコンソールと VTY 回線にアクセスできます。詳細については、ローカルユーザデータベースの更新とダウングレード（18 ページ）を参照してください。

重要 このコマンドは、ローカルで使用されているデータベースが使用できないため、信頼できるビルドには適用されません。

AAA ベースのユーザのコンソールアクセスの制限

AAA ベースのユーザは、通常、VTY 回線を介してログインします。ただし、一部のユーザはコンソール回線にだけアクセスするように制限することもできます。ローカルユーザデータベースを使用しない場合（または信頼できるビルドを実行している場合）は、他の AAA ベースのユーザについてはコンソール回線へのアクセスを制限することでこれを行う必要があります。コンソール回線にアクセスできないすべてのレベルの管理者ユーザに対して **noconsole** キーワードを有効にします。

noconsole キーワードは、次に示すコンテキスト コンフィギュレーションモード コマンドで使用できます。

```
configure
  context <ctx_name>
   administrator <username> { encrypted | nopassword | password } noconsole

   config-administrator <username> { encrypted | nopassword | password }
 noconsole
   inspector <username> { encrypted | nopassword | password } noconsole
   operator <username> { encrypted | nopassword | password } noconsole
   exit
```

noconsole キーワードを指定すると、コンソール回線へのユーザアクセスが無効になります。デフォルトでは、**noconsole** は有効になっていないため、すべての AAA ベースのユーザがコンソール回線にアクセスできます。

重要 **local-user allow-aaa-authentication noconsole** コマンドが優先されます。この場合、すべての AAA ベースのユーザがコンソール回線にアクセスできません。

設定変更の確認

Exec モードの **show configuration** コマンドを使用して、認証方式の分離に関連した変更を確認できます。設定変更を保存した後、**show configuration | grep noconsole** および **show configuration | grep novty** を実行します。これらのコマンドの出力には、実行したすべての変更が示されます。

シャーシキーの設定

シャーシキーは、システムごとに設定する必要があります。このキーは、コンフィギュレーション ファイルにある暗号化されたパスワードを復号するために使用されます。

概要

シャーシキーは、設定ファイル内の暗号化されたパスワードを暗号化および復号するために使用されます。2 つ以上のシャーシが同じシャーシキー値で設定されている場合、暗号化されたパスワードは、同じシャーシキー値を共有するシャーシのいずれかで復号できます。そのため、特定のシャーシキー値は、異なるシャーシキー値で暗号化されたパスワードは復号できません。

シャーシキーは、ファイルに保存され、設定ファイル内の機密データ（パスワードや秘密など）を保護するためのマスターキーとして使用されるシャーシ ID を生成するために使用されます。

リリース 15.0 以降では、シャーシ ID はシャーシキーの SHA256 ハッシュです。シャーシキーは、CLI コマンドまたはクイック セットアップウィザードを使用してユーザが設定できます。シャーシ ID が存在しない場合、シャーシ ID を生成するにはローカル MAC アドレスを使用します。

リリース 19.2 以降では、ユーザはクイック セットアップ ウィザードまたは CLI コマンドを使用して、シャーシキーを明示的に設定する必要があります。設定されていない場合、ローカル MAC アドレスを使用してもデフォルトのシャーシ ID は生成されません。シャーシキー（つまり、シャーシ ID）がない場合、機密データは保存された設定ファイルには表示されません。シャーシ ID は、ユーザが入力したシャーシキーおよび 32 バイトのセキュアな乱数による SHA256 ハッシュ（base36 形式で暗号化）です。これにより、キーセキュリティを得るためにシャーシキーとシャーシ ID には 32 バイトのエントロピーが確保されます。

シャーシ ID を使用できない場合、設定ファイル内の機密データの暗号化と復号は機能しません。

新しいシャーシキーの値の設定

CLI コマンド

重要 **chassis key value** コマンドと **chassis keycheck** コマンドが実行できるのは、セキュリティ管理者権限を持つユーザのみです。

新しいシャーシキーを入力するには、Exec モードの **chassis key value** *key_string* コマンドを使用します。

key_string は、1 ～ 16 文字の英数字文字列です。シャーシキーは、パスワードのように一方向の暗号化された値として保存されます。このため、シャーシキーの値がプレーンテキスト形式で表示されることはありません。

Exec モードの **chassis keycheck** *key_string* コマンドは、入力した *key_string* に基づいて一方向の暗号化されたキー値を生成します。生成された暗号化キーの値は、以前に入力したシャーシキー値の暗号化されたキー値と比較されます。暗号化された値が一致すると、コマンドは成功し、キーチェックに合格します。比較に失敗した場合は、キーチェックが失敗したことを示すメッセージが表示されます。デフォルトのシャーシキー（MAC アドレス）が現在使用されている場合、比較するシャーシキー値がないため、このキーチェックは常に失敗します。

複数のシャーシが同じシャーシキー値を共有しているかどうかを確認するには、**chassis keycheck** コマンドを使用します。

重要 リリース 19.2 以降では、既存のシャーシ ID ファイルが存在しない場合、**chassis keycheck** コマンドは非表示になります。

詳細については、『*Command Line Interface Reference*』の「*Exec Mode Commands*」の章を参照してください。

リリース 15.0 以降では、より安全なアルゴリズムを使用してシャーシキーからシャーシ ID が生成されます。結果として得られる 44 文字のシャーシ ID は、同じファイルに保存されます。

リリース 14 とリリース 15 のシャーシ ID の形式は異なります。リリース 15 では、リリース 14 のシャーシ ID が認識され、有効と見なされます。14.x から 15.0 へのアップグレードでは、シャーシ ID または設定ファイルを変更する必要はありません。

ただし、クイックセットアップウィザードまたはCLIコマンドを使用してリリース15でシャーシキーをリセットすると、新しいシャーシ ID がリリース 15 形式（16 文字ではなく 44）で生成されます。リリース 14 のビルドでは、44 文字のシャーシ ID は認識されません。その後でシャーシをリリース 14 にダウングレードすると、新たに 16 文字のシャーシ ID が生成されます。古いキー形式に対応するには、ダウングレード前に設定ファイルを v12.2 以前の形式で保存する必要があります。ダウングレードされたシャーシにv15設定ファイルをロードしようとすると、StarOS は設定ファイルに保存されているパスワード/秘密を復号できなくなります。

リリース 19.2 以降では、シャーシ ID ファイルがすでに存在しているシャーシでは何も変更されません。ただし、両方の管理カードにシャーシ ID ファイルがなくなった場合、既存のすべての設定ファイルが無効になります。元の値と同じ新しいシャーシキーを入力しても、シャーシ ID の生成には新しい方法が使用されているため、問題は解決しません。

⚠

注意 新しいシャーシキーを設定した後、リロードを開始する前に設定を保存する必要があります。「設定の確認と保存」の章を参照してください。

クイック セットアップ ウィザード

クイック セットアップ ウィザードでは、ユーザにシャーシキーの値を入力するように求めるプロンプトが表示されます。シャーシキーの値が入力されていない場合は、シャーシの MAC アドレス（20.0 より前のリリース）を使用してデフォルトのシャーシが生成されます。

リリース 20.0 以降では、シャーシ ID ファイルが存在しない場合、クイック セットアップ ウィザードにより、ユーザにシャーシキーを入力するように求めるプロンプトが表示されます。シャーシキーが入力されていない場合、デフォルトのシャーシ ID は生成されません。

クイック セットアップ ウィザードを実行するには、Exec モードの **setup** コマンドを実行します。

```
[local]host_name# setup
1. Do you wish to continue with the Quick Setup Wizard[yes/no]: y
2. Enable basic configuration[yes/no]: y
3. Change chassis key value[yes/no]: y
4. New chassis key value: key_string
```

FSC ファブリックの自動リセットの有効化

デフォルトでは、スイッチファブリック内で過剰な数の破棄されたファブリック出力パケットが発生した場合、ファブリック ストレージ カードを手動でリセットしてファブリックをリカバリする必要があります。

過剰な数の破棄されたファブリック出力パケットが検出された場合に、必要に応じて FSC の自動リセットを有効にすることができます。

グローバル コンフィギュレーション モードの **fabric fsc-auto-recover** コマンドは、過剰な数の破棄されたファブリック出力パケットを検出したときに、自動 FSC リセットを有効または無効にします。

次のコマンドシーケンスは、この機能を有効にします。

```
configure
   fabric fsc-auto-recovery { disable | enable } [ max-attempts [
number_attempts | unlimited ] ]
   end
```

max-attempts [*number_attempts* **| unlimited]** は、StarOS が各 FSC のリセットを試行する回数を 1 ～ 99 の整数または無制限で指定します（FSC がリセットされるまで停止しません）。デフォルトの設定は 1 秒です。

重要　この機能を有効にするには、最初にグローバル コンフィギュレーション モードの **fabric egress drop-threshold** コマンドを介して、ファブリック出力ドロップしきい値を設定する必要があります。