

SmartMesh IP VManager CLI ガイド

目次

1	本書について	4
1.1	関連資料	4
1.1.1	スタータ・キットのクイック・ガイド	4
1.1.2	ユーザ・ガイド	4
1.1.3	デバイスの対話操作インターフェース	4
1.1.4	アクセス・ポイント・モード	5
1.1.5	ソフトウェア開発ツール	5
1.1.6	アプリケーション・ノート	5
1.1.7	新規設計の開始時に役立つ資料	5
1.1.8	ソフトウェア	5
1.1.9	その他の役立つ資料	5
1.2	表記規則	6
1.3	改訂履歴	7
2	はじめに	8
2.1	マネージャの接続	8
2.1.1	ユーザ・アカウントの管理	9
2.1.2	役立つヒント	10
2.1.3	設定済構成とアクティブ構成	10
2.1.4	同等のAPIコマンド	10
3	コマンド	11
3.1	clear	11
3.2	config	12
3.2.1	config delete	12
3.2.2	config deletei	13
3.2.3	config get	14
3.2.4	config geti	16
3.2.5	config set	17
3.2.7	config reload	21
3.2.8	config restore	22
3.3	exec	23
3.3.1	exec clearStats	23
3.3.2	exec decommission	24
3.3.3	exec deleteMote	25

3.3.4	exec deleteUnjoined	26
3.3.5	exec exchJoinKey	27
3.3.6	exec exchNetId	28
3.3.7	exec motelog	29
3.3.8	exec setAdv	30
3.3.9	exec sendData	31
3.3.10	exec sendIP	32
3.4	exit/logout/quit	34
3.5	help	35
3.6	ping	36
3.7	reset	37
3.7.1	reset mote	37
3.7.2	reset network	38
3.8	show	39
3.8.1	show acl	39
3.8.2	show alarms	40
3.8.3	show time	41
3.8.4	show apcs	42
3.8.5	show aps	43
3.8.6	show dcl	44
3.8.7	show ini	45
3.8.8	show mote/ap	46
3.8.9	show motes	48
3.8.10	show network	49
3.8.11	show paths	51
3.8.12	show route	52
3.8.13	show services	53
3.8.14	show sessions	54
3.8.15	show system	55
3.8.16	show time	56
3.8.17	show unjoined	57
3.8.18	show user	58
3.8.19	show users	59
3.8.20	show ver	60
3.9	sm	61
3.10	su	62
3.11	trace	63

1 本書について

1.1 関連資料

SmartMesh IPネットワーク向けに以下の資料が提供されています。

1.1.1 スタータ・キットのクイック・ガイド

- [SmartMesh VManager Easy Start Guide](#) - VManagerの基本的なインストール方法とネットワークの動作確認テストについて説明しています。
- [SmartMesh IP Embedded Manager Easy Start Guide](#) - Embedded Managerの基本的なインストール方法とネットワークの動作確認テストについて説明しています。
- [SmartMesh IP Embedded Manager Tools Guide](#) - インストールのセクションではシリアル・ドライバのインストール手順について説明しており、Easy Start Guideやその他のチュートリアルで使用されるサンプル・プログラムも含まれています。

1.1.2 ユーザ・ガイド

- [SmartMesh IPユーザ・ガイド](#)- ネットワーク概念についての説明と、モートおよびマネージャのAPIを使用して特定のタスク(データ送信や統計情報の収集など)を実行する方法について説明しています。この資料は、APIガイドを使用するための予備知識を提供します。また、SmartMeshの用語集も含まれています。

1.1.3 デバイスの対話操作インターフェース

マネージャと対話操作するためのインターフェースには、プログラムからやり取りするアプリケーション・プログラミング・インターフェース(API)と、ユーザがやり取りするコマンド・ライン・インターフェース(CLI)の2種類があります。

- [SmartMesh IP Embedded Manager CLI Guide](#)- ユーザがEmbedded Managerとやり取りするために使用します(クライアントの開発中やトラブルシューティングなど)。このガイドは、CLIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP Embedded Manager API Guide](#)- プログラムを使用してEmbedded Managerとやり取りするために使用します。このガイドは、APIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP VManager CLI Guide](#)- ユーザがVManagerとやり取りするために使用します(クライアントの開発中やトラブルシューティングなど)。このガイドは、CLIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP VManager API Guide](#)- プログラムを使用してVManagerとやり取りするために使用します。このガイドは、APIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP Mote CLI Guide](#)- ユーザがモートとやり取りするために使用します(センサー・アプリケーションの開発中やトラブルシューティングなど)。このガイドは、CLIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP Mote API Guide](#)- プログラムを使用してモートとやり取りするために使用します。このガイドは、APIの接続とそのコマンド・セットについて説明しています。

1.1.4 アクセス・ポイント・モード

- [SmartMesh IPユーザ・ガイド](#) - DC2274をアクセス・ポイント・モードとして使用するための再プログラミングについて説明しています。
- [VManager AP Bridge User's Guide](#) - アクセス・ポイント・ブリッジ参照ソフトウェアのユーザ・ガイドです。

1.1.5 ソフトウェア開発ツール

- [Dustcloud.org](#) - モードおよびマネージャAPIを使用し、ネットワークを視覚化するための各種オープン・ソース・ソフトウェア・ツールに関する技術文書とリンクが含まれています。

1.1.6 アプリケーション・ノート

- [SmartMesh IPアプリケーション・ノート](#) - SmartMesh IPネットワーク固有の各種トピックと、SmartMeshネットワーク全般に当てはまるトピックが含まれています。

1.1.7 新規設計の開始時に役立つ資料

- 使用されるモード(LTC5800-IPM SoCなど)、またはこれに基づく[モジュール](#)のデータシート。
- 使用される組込みマネージャ(LTC5800-IPR SoCなど)、またはこれに基づく[組込みマネージャ](#)のデータシート。
- モード／マネージャSoC用または[モジュール用のハードウェア統合ガイド](#) - 設計にSoCまたはモジュールの統合を盛り込むためのベスト・プラクティスを提供しています。
- 組込みマネージャ用の[ハードウェア統合ガイド](#) - 設計する際に組込みマネージャを統合させるためのベスト・プラクティスを提供しています。
- [Board Specific Integration Guide](#) - SoCモードおよびマネージャに対するデフォルトのI/O設定方法と、「ヒューズ表」を使用した水晶発振器のキャリブレーション情報について説明しています。
- [Hardware Integration Application Notes](#) - SoC設計チェックリスト、アンテナ選定ガイドなどを含みます。
- [ESP Programmer Guide](#) - DC9010 Programmer Boardと、デバイスへのファームウェアのロードに使用するESPソフトウェアのガイドです。

1.1.8 ソフトウェア

- ESPソフトウェア - モードまたはモジュールにファームウェア・イメージをプログラミングするために使用します。[ESP Programmer Guide](#)を参照してください。
- Fuse Tableソフトウェア - [Board Specific Configuration Guide](#)で説明されているヒューズ表を作成するために使用します。

1.1.9 その他の役立つ資料

- [よくある質問の一覧](#)。


1.2 表記規則


本書では、以下の表記規則を使用します。


コンピュータ・タイプ (Computer type) は、URLの指定など、ユーザが入力する情報を示します。


太字は、ボタン、フィールド、メニュー・コマンド、デバイス・ステート、モードを示します。

*斜体*は、新しい用語やAPIとそのパラメータを示します。

 ヒントは、製品に関して役立つ情報を提供します。

 情報テキストは、背景や前後関係の理解に役立つ追加情報を提供します。

 注記は、概念についてより詳しい説明を提供します。

 **警告！** 警告は、データ損失やハードウェアまたはユーザへの物理的な損害を引き起こす可能性のある動作をユーザに知らせます。

コード・ブロックは、コード例を示します。

CLIコマンドの記述には、以下の表記と用語を使用します。

	フィールドに対する選択肢を表します。例えば、 <code><macAddr> <moteld></code> は、モートの指定にモートIDまたはMACアドレスを使用できることを意味します。
<>	必須フィールドを表します。
{ }	フィールドのグループを表します。
[]	省略可能フィールドを表します。
MAC アドレス	MACアドレスの指定では、スペースを使用しないでください。ハイフンは省略可能です(本書では、読みやすくするためにハイフンを付けています)。 以下の例はどちらも有効です。 <pre> \$> show mote 00000000000022CA \$> show mote 00-00-00-00-00-00-22-CA </pre>

1.3 改訂履歴

リビジョン	日付	説明
1	2015/12/15	初期リリース
2	2016/08/19	フェーズIの製作
3	2016/11/08	show apcsコマンドの追加、show mote/apコマンドの更新、多数の軽微な改善
4	2016/11/18	config set networkコマンドの更新
5	2017/03/15	show unjoinedコマンドの明確化、いくつかのサンプル出力の更新

2 はじめに

本書では、コマンド・ライン・インターフェース(CLI)にログオンすることでSmartMesh IP VManagerに送信できるコマンドについて説明します。CLIは、マネージャサーバへのssh接続を介して使用できます。CLIは、開発中やインタラクティブなトラブルシューティングなどで、ユーザがマネージャとやり取りするためのものです。ほとんどのコマンドはアトミックであるため、1つのコマンドとその引数がCLIに入力されると、1つのレスポンスが返されます。例を挙げると、helpコマンドは使用可能なコマンドのリストを返します。ネットワーク操作を実行するコマンドはアトミックではなく、レスポンスの受信後に非同期で出力が生成されます。トレースはアトミックではないため、コマンドを取り消すまで出力が非同期で生成されます。

マシン間通信(ホストプログラムとマネージャの通信など)には、アプリケーション・プログラミング・インターフェース(API)を使用します。詳細については、[SmartMesh IP VManager API Guide](#)を参照してください。

2.1 マネージャの接続

CLIにアクセスするには、はじめにマネージャのLinuxシェルに接続する必要があります。これには、sshをサポートするWindows上の任意のターミナル・プログラム(PuTTYやTeraTermなど)、またはLinux/OS X上の任意のコンソール／ターミナル・ウィンドウを使用できます。sshの実行には、マネージャサーバのIPアドレスが必要です。サーバのIPアドレスが192.168.1.100である場合、以下のコマンドを使用します。

```
$ ssh dust@192.168.1.100
```

パスワードを要求するプロンプトが表示されます。**dust**ユーザのデフォルト・パスワードは、**dust**です。SSH接続では、システムのユーザ管理コマンドを介してユーザが管理されます。

Linuxシェルからconsoleコマンドを実行し、CLIを開始します。デフォルトのユーザ名は**dust**で、パスワードは**dust**です。コンソール接続とAPI接続では、VManagerの構成内でユーザが管理されます。

```
dust@voyager-vm:~$ console
Welcome to the Voyager CLI Console on Linux
Version 0.1.0.10

Enter your username: dust
dust's password:
$>
```

(コンソールにログインするための)デフォルトのユーザ名およびパスワードが、Linuxのログインと同じものになっていますが、これらは異なるログインであるため、別々に再設定することができます。

2.1.1 ユーザ・アカウントの管理

CLIにログインした後、CLIコマンド`config set user`を使用して、デフォルト・ユーザ・アカウントのパスワード(またはその他のユーザ・フィールド)を変更できます。

```
$> show user dust
User information:
  Id:          dust
  Privilege:   USER
  Description: duster

$> config set user dust description="a dust user"
Done

$> config reload users Done

$> show user dust
User information:
  Id:          dust
  Privilege:   USER
  Description: a dust user
```

永続ユーザ・データベースを変更した場合、変更を有効にするために再ロードする必要があります。新規ユーザの作成や不要なユーザの削除についても、同じことが当てはまります。

```
$> show users
User configuration:
  Id:          dust

$> config set user bobo description="bobo" password="free4willy" privilege=user
Done

$> config reload users
Done

$> show users
User configuration:
  Id:          dust
  Id:          bobo

$> config delete user bobo
Done

$> config reload users
Done

$> show users
User configuration:
  Id:          dust
```

2.1.2 役立つヒント

本書では、SmartMesh IP VManagerで使用できるコンソール・コマンドについて説明します。このコンソールでは、大文字と小文字を区別する必要があります。

コマンド入力時に、いくつかのコンソール・ショートカットを使用できます。

- 上矢印キーは、過去のコマンド履歴を循環表示します。
- タブキーは、部分的に入力されたコマンド名を自動的に補完します。
- プレフィックスとして#を使用すると、MACアドレスを簡略化できます。例: `show mote #38-01-02`は、`show mote 00-17-0D-00-00-38-01-02`に展開されます。

出力が複数ページにわたるコマンドの場合、`-p`(または`--page`)スイッチを使用して起動することで、ページ分割した出力を生成できます。リターンキーを押すと次のページに進みます。

2.1.3 設定済構成とアクティブ構成

VManagerでは、アクティブな構成パラメータと、次回のマネージャ起動時に使用するために設定されたパラメータが区別されます。

- `show`コマンドは、アクティブ構成を表示します。
- `config set/get`コマンドは、設定済のパラメータを保存または表示します。
- `config reload`コマンドは、特定のモジュールのアクティブ構成を設定済に上書きします。
- `config restore`コマンドは、設定済構成をデフォルト値で上書きします(別名「ファクトリ・リストア」)。

2.1.4 同等のAPIコマンド

CLIコマンドと同等のAPIコマンドは、Swagger形式で記載します。例:

```
POST /path/to/a/resource
```

同等のAPIコールに対するリクエストおよびレスポンスのJSON形式については、Swaggerの技術文書を参照してください。

3 コマンド

3.1 clear

説明

このコマンドは、コンソール画面をクリアします。

構文

```
clear
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> clear
```

3.2 config

configコマンドは、設定済の非アクティブな構成設定と相互作用します。現在使用されているアクティブな設定を表示するには、[show](#)コマンドを使用します。

3.2.1 config delete

説明

設定済構成のリストからアイテムを1つ削除します。

構文

```
config delete acl <macAddr> |
              dcl <macAddr> |
              user <userId>
```

パラメータ

パラメータ	説明
(以下のいずれか1つ)	
acl	アクセス制御リスト(access control list)からモートを1つ削除、 <i>macAddr</i> 、 <i>joinkey</i>
dcl	拒否制御リスト(deny control list)からモートを1つ削除、 <i>macAddr</i>
user	システムからユーザを削除


例

```
$> config delete dcl 00-17-0D-00-00-12-34-56
Done
$> config delete user mrdusty
Done
```

3.2.2 config deletei

説明

ユーザが変更したINI値の内容を削除し、デフォルト値にリセットします。パラメータ自体は削除されず、値がシステムのデフォルトにリセットされます。存在しないパラメータを指定すると、エラーメッセージが返されます。

 このコマンドにはスーパーユーザ権限が必要です。詳細については、[suコマンド](#)を参照してください。特定の性能目標を達成するためにアナログ・デバイスから助言を受けた場合を除き、INIパラメータは変更しないでください。

構文

```
config deletei <param>
```

パラメータ

パラメータ	説明
param	リセットするパラメータ

例

```
#> config deletei BWMULT  
Done
```

3.2.3 config get

説明

指定されたアイテムの設定済構成を表示します。-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。

構文

```
config get [-p|--page] acl <macAddr> |
          dcl <macAddr> |
          network |
          system |
          user <userId> |
          users
```

パラメータ

パラメータ	説明
(以下のいずれか1つ)	
acl	設定済のアクセス制御リスト(acl)の構成を取得 - acl全体またはモートmacAddrのエントリ
dcl	設定済の拒否制御リスト(dcl)の構成を取得 - dcl全体またはモートmacAddrのエントリ
network	設定済のネットワーク構成を取得
system	設定済のシステム構成を取得
user	ユーザのuserIdの設定済ユーザ構成を取得
users	設定済ユーザ構成を取得

例

```
$> config get -p acl
00-17-0D-00-00-12-34-AA
00-17-0D-00-00-12-34-AB

$> config get dcl
00-17-0D-00-00-12-34-56

$> config get network
Network configuration:
  networkId:          294
  topologyType:       MESH
  dsFrameMultiplierDelay: 3600000
  ccaMode:             0
  ipAddrPrefix:       FE80::
  basePkPeriod:       15000
  dsFrameMultiplier:  1
  joinSecurityType:   COMMON_SKEY
  minServicePkPeriod: 100
  dsFrameSize:        512
  numParents:         2
  gpsMode:            False
  channelList:        32767
  usFrameSize:        1024

$> config get system
System configuration:
  Name:
  Location:
  CLI timeout: 0 minutes


$> config get users
User configuration:
Id:    dust

$> config get user dust
User information:
Id:    dust
Password: ****
Privilege: USER
Description: duster
```

3.2.4 config geti

説明

内部のINI設定を表示します。パラメータ`param`を指定しない場合、すべてのパラメータが表示されます。`param`フィールドのワイルドカードとして、アスタリスク(*)を使用できます。

 このコマンドにはスーパーユーザ権限が必要です。詳細については、[suコマンド](#)を参照してください。一般的な用途に合わせて多数の調整パラメータ(本書には未掲載)が最適化されていますが、ほとんどのユースケースではデフォルト・パラメータを使用します。特定の性能目標を達成するためにアナログ・デバイスズから助言を受けた場合を除き、INIパラメータは変更しないでください。

構文

```
config geti [param]
```

パラメータ

パラメータ	説明
param	INIパラメータの名前

例

```
#> config geti *
MNGRINI Module configuration :
ADV_SEND_TO:180000
ADV_TIME_DELAY:3600000
ADV_TIME_OFF:1000
ADV_TIME_ON:1000
.....
#> config geti NUMBCAST
MNGRINI Module configuration :
NUMBCAST:2
```


3.2.5 config set

説明

非アクティブな設定済構成の設定を変更します。アイテムが存在しない場合、適宜作成されます。フィールド引数名にスペースが含まれる場合、二重引用符で囲む必要があります。例: `sysName = "Foo Bar"`

非アクティブなメモリ内で変更したすべての設定は、アクティブ・メモリに読み込まれるまでシステムに影響を与えません。アクティブ・メモリに読み込むには、`config reload <module> | all`コマンドを使用します。また、アスタリスク(*)の付いた設定を有効にするには、システム・リセットが必要です。`reset --reload`コマンドを使用してください。

構文

```
config set acl <macAddr> joinKey=<joinKey> |
    dcl <macAddr> |
    network {<field=val> [field2=val2] ...} |
    system {<field=val> [field2=val2] ...} |
    user <userId> {<field=val> [field2=val2] ...}
```

パラメータ

パラメータ (以下のいずれか1つ)	説明
acl	アクセス制御リスト(access control list)エントリ(<i>macAddr</i> , <i>joinkey</i> で構成されるタプル)を保存。 <i>joinkey</i> は16バイトキー
dcl	拒否制御リスト(deny control list)エントリをモート <i>macAddr</i> 用に保存
network	ネットワーク構成フィールドを保存
system	システム構成フィールドを保存
user	ユーザ構成を保存(注: UserIDに使用できるのは小文字と数字のみ)

例

```
$> config set acl 00-17-0D-00-00-60-04-B0 joinkey=445553544E4554574F524B53524F434B
Done

$> config set dcl 00-17-0D-00-00-12-34-56
Done

$> config set system sysName="Foo" location="Bar Baz"
Done

$> config set user mrdusty privilege=user
Done

$> config reload all
Done
```

config set network parameters

以下のネットワークパラメータを設定できます。アスタリスク(*)の付いたアイテムを有効にするには、`reset network -reload`コマンドを使用し、ネットワーク全体をリセットする必要があります。

パラメータ	説明	デフォルト値
networkId *	ネットワークID	1229
topologyType *	ルーティングおよびカスケード接続(MESH)、ルーティングおよび非カスケード接続(EVENT)、非ルーティング(STAR)	MESH
downFrameMultiplierDelay *	最初のモートが参加してから、下りフレーム乗数の使用に切り替えるまでの時間(ミリ秒)	3600000
ccaMode *	チャンネル・アセスメントのクリアに使用されるモード。 0 = False = オフ、1 = True = オン	False
ipAddrPrefix *	WSN向けのIPアドレスのプレフィックス	FE80::
basePkPeriod	各デバイスに割り当てる最小帯域幅(ミリ秒)	15000
downFrameMultiplier *	安定状態での下りフレームの乗数。ネットワークの形成後、下りフレームのサイズを、 <code>downFrameSize</code> から <code>downFrameSize * downFrameMultiplier</code> に拡張	1
joinSecurityType *	参加デバイスのセキュリティ・モード - COMMON_SKEY、COMMON_SKEY_QUARANTINE、UNIQUE_SKEYのいずれか	COMMON_SKEY
commonJoinKey	共通参加鍵(joinSecurityModeにCOMMON_SKEYまたはCOMMON_SKEY_QUARANTINEが設定されている場合のみ有効)。長さ128ビットのbase64形式でなければなりません。セキュリティ上の理由から、GETレスポンスで鍵は返されません	44 55 53 54 4E 45 54 57 4F 52 4B 53 52 4F 43 4B
minServicePkPeriod	可能な最小(最速)サービス要求時間(ミリ秒)	100
downFrameSize *	下りフレームサイズ(スロット)。upFrameSizeの1/4から4倍の範囲内にする必要があります。	512
numParents	モートあたりの親の数	2
channelList *	使用されるチャンネルのビットマップ(b0 = IEEEチャンネル11)	32767
upFrameSize *	上りフレームサイズ(スロット)。downFrameSizeの1/4から4倍の範囲内にする必要があります。	512
gpsMode *	ネットワークのクロック源、GPSまたはフリー・ランニング。 0 = False = オフ、1 = True = オン	False

例

```
$> config set network networkId=1229
Done
$> reset network --reload
```

config set system parameters

以下のシステムパラメータを設定できます。

パラメータ	説明
sysName	ユーザが割り当てる名前の文字列
location	ユーザが割り当てるロケーションの文字列
cliTimeout	非アクティブなセッションのCLIタイムアウト

スペースを含む文字列は二重引用符で囲む必要があります。例: "Mr Dusty"

config set user parameters

新しいユーザを作成します。以下のユーザパラメータを設定できます。

パラメータ	説明
userId	ユーザID(文字列)。ユーザIDの制約については下記を参照
description	ユーザの説明
password	ユーザのパスワード
privilege	ユーザの権限レベル、 <i>user</i> または <i>viewer</i>

権限設定には2つの種類があります。*user*権限はすべての(非su)コマンドにアクセス可能で、システム設定を変更できます。*viewer*権限は、設定の表示のみに限定されます。

userIdに使用できる文字は、小文字、数字、アンダースコア(_)、ハイフン(-)のみで、文字で始まる必要があります。大文字、スペース、その他の文字は使用できません。

例

```
$> config set user mrdusty privilege=user password=mypw
Done
```

3.2.6 config seti

説明

内部のINIパラメータを変更します。

⚠ このコマンドにはスーパーユーザ権限が必要です。詳細については、[suコマンド](#)を参照してください。一般的な用途に合わせて多数の調整パラメータ(本書には未掲載)が最適化されていますが、ほとんどのユースケースではデフォルト・パラメータを使用します。特定の性能目標を達成するためにアナログ・デバイスから助言を受けた場合を除き、INIパラメータは変更しないでください。

構文

```
config seti <param>=<value>
```

パラメータ

パラメータ	説明
param	INIパラメータの名前
value	INIパラメータの新しい値

例

```
$> config seti numbcast=3  
OK
```

3.2.7 config reload

説明

「設定済」メモリに格納された構成情報をアクティブ・メモリに読み込みます。新しく読み込まれたパラメータを使用して、直ちにシステムが起動されます。

すべてのモジュールを再ロードするには、モジュールの代わりに「all」を指定します。

構文

```
config reload <module>|all
```

パラメータ

パラメータ	説明
module	指定されたモジュールの再ロード

例

```
$> config reload all
Done

Creating a new user and making it active would be done as follows...
$> config set user mrdusty privilege=user password=mypw
Done

$> show user mrdusty
Error: Entry not found
$> config reload users
Done
$> show user mrdusty
User information:
  userId:      fred
  privilege:   USER
  description:
```

3.2.8 config restore

説明

すべての構成を工場出荷時のデフォルト設定に戻します。

⊖ このコマンドを使用すると、すべての設定が工場出荷時のデフォルト設定に戻されます。ユーザ設定はすべて失われ、回復することはできません。

⊖ このコマンドの実行後、明示的にVManagerを再起動する必要があります。

構文

```
config restore
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> config restore
```

```
OK
```

3.3 exec

execコマンドは、ネットワークまたはトポロジ・データベースと相互作用します。

3.3.1 exec clearStats

説明

蓄積された統計情報をクリアします。

構文

```
exec clearStats
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> exec clearStats
```

```
Done
```

3.3.2 exec decommission

説明

このコマンドは、モートをネットワークから問題なく削除するための準備として、子モートを別の親に移動します。戻り値はコールバックIDです。後でモートの削除準備ができたときのコンソール通知に、このコールバックIDが含まれます。子モートが接続可能な親モートを見つけられない場合があります（推奨される実装ガイドラインに従った場合、そのようなことは起こりません）。その場合、モートがリセットされると子モートが孤立します。

構文

```
exec decommission <macAddr|motelId>
```

パラメータ

パラメータ	説明
macAddrまたはmotelId	macAddrまたはmotelIdで指定されたモートの関連付けの削除

例

```
$> exec decommission 00-17-0D-00-00-12-34-56

Done. Callback id: 1
```


3.3.3 exec deleteMote

説明

ネットワークからモートを削除します。もしモートがロストしていない状況であれば、最初に関連付け削除 (decommission) を実行し、潜在的なデータロス回避する必要があります。

このコマンドはモートの参加カウンタをクリアするために使われます。またこれによってモートの参加カウンタは完全に再起動されます。同様に、ブリンクモードにおけるblinkパケットにおいても、参加カウンタがマッチするまでパケット送出はされません。同じことがblinkモードを使用したパケットにも当てはまり、参加カウンタが一致するまで何も受信できなくなります。

構文

```
exec deleteMote <macAddr|moteId>
```

パラメータ

パラメータ	説明
macAddrまたはmoteId	macAddrまたはmoteIdで指定されたモートの削除

例

```
$> exec deleteMote 00-17-0D-00-00-12-34-56
```

```
Done
```

3.3.4 exec deleteUnjoined

説明

現在トポロジに含まれていないモートのconfigdbエントリを削除します。このコマンドは再プログラミング等によって (configdbにおける参加カウンタが不一致となるような) 未参加モートとなったモートを削除する場合などに有効です。

構文

```
exec deleteUnjoined
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> exec deleteUnjoined
```

```
Done
```

3.3.5 exec exchJoinKey

説明

このコマンドは、指定されたモートの参加鍵を変更します。このコマンドに対してモートが肯定的な応答を返すと、対応するACLエントリも更新されます。

構文

```
exec exchJoinKey <macAddr|moteld> <key>
```

パラメータ

パラメータ	説明
macAddrまたはmoteld	macAddrまたはmoteldで指定されたモートのjoinKeyの変更
key	新しい16バイトの参加鍵

例

```
$> exec exchJoinKey 00-17-0D-00-00-12-34-56 0001020304050607080A0B0C0D0E0F  
  
Done. Callback id: 1
```

3.3.6 exec exchNetId

説明

このコマンドは、ネットワーク内のすべてのモートに対して、新しいネットワークIDを配布します。戻り値はコールバックIDです。後でnetidが変更されるときのコソール通知に、このコールバックIDが含まれます。

構文

```
exec exchNetId <newNetId>
```

パラメータ

パラメータ	説明
newNetId	ネットワークのIDをnewNetIdに変更

例

```
$> exec exchNetId 294  
  
Done. Callback id: 2
```

3.3.7 exec motelog

説明

このコマンドは、モートのリセット・ログを取得します。

構文

```
exec motelog <macAddr|moteId>
```

パラメータ

パラメータ	説明
macAddrまたはmoteId	macAddrまたはmoteIdで指定されたモートからログ・ファイルを取得

例


```
$> exec motelog 00-17-0D-00-00-12-34-56
$> TRACE MOTELOG
2015-12-09 15:10:27.890 from mote 00-17-0D-00-00-12-34-56 : (x100)
```

この例では、通常の(ウォッチドッグ)リセットを示すリセット・コード0x100が、モートから返されています。

3.3.8 exec setAdv

説明

このコマンドはネットワーク内のアドバタイズメントを制御します。offに設定するとすべてのアドバタイズメントが無効になるため、モートはネットワークに参加できません。onに設定するとモートのアドバタイズメントが有効になり、モートがネットワークに参加できます。VManagerは、アドバタイズメントのオンとオフを自動的に切り替えません。

 ネットワーク内のアドバタイズメントを無効にすることは危険です。アドバタイズメントを無効にした場合、新しいモートがネットワークに参加できず、リセット後に既存のモートが再参加することもできません。非常に珍しいケースで、例えばネットワークにモートを参加させないようにしたり、消費電力を節約したりするために、アドバタイズメントの無効化が役立つ場合もありますが、たいていの場合、アドバタイズメントはVManagerの制御下に置いたままにすることを推奨します。

構文

```
exec setAdv <on|off>
```

パラメータ

パラメータ	説明
onまたはoff	on = アドバタイズメントの有効化、off = アドバタイズメントの無効化

例

```
$> exec setAdv on
Done
```

3.3.9 exec sendData

説明

モートにパケットを送信します。このコマンドは、APIコマンドのPOST /motes/m/{mac}/dataPacketを実行した場合に相当します。

構文

```
exec sendData <macAddr|moteId> <srcPort> <dstPort> <priority> <payload>
```

パラメータ

パラメータ	説明
macAddrまたはmoteId	macAddrまたはmoteIdで指定されたモートへのパケットの送信
srcPort	パケットのUDP送信元ポート
dstPort	パケットのUDP宛先ポート
priority	パケットの優先順位。LOW、MEDIUM、HIGHのいずれか
payload	パケットのペイロード・サイズ(バイト、16進数)。ペイロードの最大サイズは80バイト

例

送信元ポート = 0xF0B8、宛先ポート = 0xF0B8、優先順位 = MEDIUM、ペイロード・サイズ = 0x11 0x 22 0x 33 0x44 0x55で、モートにパケットを送信します。

```
$> exec sendData 00-17-0D-00-00-12-34-56 61624 61624 MEDIUM 1122334455
Done. Callback id: 2
```

3.3.10 exec sendIP

説明

このコマンドは、指定したペイロードを含むパケットをモートに送信します。ユーザが有効な6LoWPANヘッダを構築し、データの最初に付加する必要があります。このコマンドは、APIコマンドのPOST /motes/m/{mac}/ipPacketを実行した場合に相当します。

構文

```
exec sendIP <macAddr> <priority> <encryptionOffset> <payload>
```

パラメータ

パラメータ	説明
macAddr	パケットの宛先となるモートのMACアドレス
priority	パケットの優先順位。LOW、MEDIUM、HIGHのいずれか
encryptionOffset	パケットの暗号化部分の開始位置までのオフセット(単位はバイト、通常はユーザ・データの開始位置)
payload	パケットのペイロード・サイズ(バイト、16進数)。最初に付加された6LoWPANヘッダとその後のユーザ・データで構成されます。ペイロードの最大サイズは80バイト

例

送信元ポート = 0xF0B8、宛先ポート = 0xF0B8、優先順位 = MEDIUM、ペイロード・サイズ = 0x11 0x 22 0x 33 0x44 0x55で、マネージャからモート00-17-0D-00-00-12-34-56にパケットを送信します。

- 暗号化オフセットは0です。
- IPヘッダ・フィールドは以下のとおりです。

フィールド	バイト	コンテンツ
LOWPAN_IPHC	2	011.11.1.10:C.S.ss.M.D.dd <ul style="list-style-type: none"> ● 011 - IPHCディスパッチ ● 11 - トラフィック・クラスおよびフロー・ラベルの省略 ● 1 - 次のヘッダ・フィールドの圧縮 ● 10 - ホップ・リミットの圧縮 ● C(コンテキスト識別マーク) = 0(省略) ● S(送信元の圧縮) = 1 ● ss(送信元モード) = 11(省略) ● M(マルチキャスト) = 0(なし) ● D(宛先の圧縮) = 1 ● dd(宛先モード) = 11(省略) 0111.1110.0111.0111 = 0x7E77
Context ID Extension	0	未使用
Source Address	0	省略
Destination Address	0	省略
UDP Header	0/1	111101.S.D <ul style="list-style-type: none"> ● S/D(送信元/宛先ポートの圧縮) = 11 1111.0111 = 0xF7
UDP Ports	1	s.d <ul style="list-style-type: none"> ● s(送信元ポート = F0Bs) = b1000 ● d(宛先ポート = F0Bd) = b1000 1000.1000 = 0x88

6LoWPANヘッダは、0x7D77F788です。

```
$> exec sendIP 00-17-0D-00-00-12-34-56 MEDIUM 0 7E77F77D77F7881122334455

Done. Callback id: 20
```

3.4 exit/logout/quit

説明

このコマンドはコンソール・アプリケーションを終了し、ユーザをLinuxシェルに戻します。exit、logout、quitのいずれも使用できます。

構文

```
exit | logout | quit
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> exit
```

3.5 help

説明

使用できるコマンドのリスト、またはコマンドの詳細を表示します。

構文

```
help [command] [subCommand]
```

パラメータ

パラメータ	説明
command	任意のCLIコマンド

例

```
$> help
For more information on a specific command, type HELP command-name
clear          Clear screen
...

$> help clear
Usage: clear

Clear screen.
```

3.6 ping

説明

このコマンドは、指定したモートまたはAPに対して、応答時間、温度、電圧を含むレスポンスを送信するように要求します。これはアプリケーション層のコマンドで、ICMPエコーは使用しません。

構文

```
ping <macAddr|moteId>
```

パラメータ

パラメータ	説明
macAddrまたはmoteId	macAddrまたはmoteIdで指定されたモートへのpingを実行

例

```
$> ping 1
Done. Callback id: 2
$> PING 2015-10-12 14:12:40.598 Reply from Mote #1, mac: 00-17-0d-00-00-12-34-56
    CallbackId: 2, Latency: 8ms (0 hops), Data: 3276mV, 34C
```

3.7 reset

このコマンドは、特定のデバイスまたはネットワークをリセットします。

3.7.1 reset mote

説明

モートまたはAPに対するリセットを実行します。

構文

```
reset <mote|ap> <macAddr|moteId>
```

パラメータ

パラメータ	説明
(以下のいずれか1つ)	
moteまたはap	macAddrまたはmoteIdで指定されたデバイス・タイプのリセット
macAddrまたはmoteId	リセットするデバイス

例

```
$> reset mote 25

OK

$> reset ap 00-01-02-03-04-05-06-07

OK
```

3.7.2 reset network

説明

ネットワークをリセットします。オプションのreload引数は、ネットワークを再起動する前に構成を再ロードするかどうかを指定します。

構文

```
reset network [--reload]
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> reset network --reload  
Done
```

3.8 show

このコマンドは、現在の(アクティブな)構成、統計情報、揮発性情報を表示します。永続パラメータを表示するには、[config](#)を使用します。

出力が複数ページにわたるコマンドの場合、`-p`(または`--page`)スイッチを使用して起動することで、ページ分割した出力を生成できます。リターンキーを押すとページが進みます。

3.8.1 show acl

説明

マネージャのアクセス制御リスト(access control list)にホワイトリストとして現在含まれているモートの一覧を表示します。`-p`または`--page`オプションを使用すると、ページごとに出力が一時停止します。MACアドレスを指定すると、そのデバイスのエントリのみが出力されます。

構文

```
show acl [-p|--page] [macAddr]
```

パラメータ

パラメータ	説明
macAddr	MACアドレス <code>macAddr</code> を持つデバイスのみを出力(ACLに含まれる場合)

例

```
$> > show acl
ACL:

MAC: 00-17-0D-00-00-38-FF-FF

MAC: 00-17-0D-00-00-37-6E-A1
```

3.8.2 show alarms

説明

すべてのオープン・アラームのリストを返します。

構文

```
show alarms
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show alarms

Alarm info list:
 2015-03-17 16:34:38 Maximum number of motes reached
```


3.8.3 show time

説明

このコマンドは、指定されたAPブリッジに関連付けられたAPブリッジ・コネクタに関する情報を表示します。

構文

```
show apc <macAddr|moteld>
```

パラメータ

パラメータ	説明
<macAddr moteld>	macAddrまたはmoteldで指定されたモートの情報を返します

例

```
$> show apc 1
APC apc-603528, Interface ID: 1
  AP #1, MAC: 00-17-0D-00-00-60-35-28 (Oper)
  Version: 1.0.1.16 (built 2016/08/04 11:43:34)
  State: Working
  Connection: 127.0.0.1:41444

APC statistics:
RX pkts: 119998
TX pkts: 37149
TX delays: <5ms: 37149, <7ms: 0, <10ms: 0, <50ms: 0, >50ms: 0; Max delay: 4.743ms
Pauses: 0
Disconnects: 0
```

3.8.4 show apcs

説明

ネットワーク内にあるすべてのアクセス・ポイント・コントローラ (APC) のリストを表示します。それぞれのAPCは、APブリッジからVManagerへの接続を表します。-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。

構文

```
show apcs [-p|--page]
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show apcs
Name           ID   APC St   AP MAC                               AP ID   AP St
-----
apc-603528     1   Working 00-17-0D-00-00-60-35-28             1   Oper
apc-603772     3   Working 00-17-0D-00-00-60-37-72             3   Oper
```

このコマンドは、ネットワーク内に現在ある (または過去にあった) APCをすべて表示します。

- Name: APC識別マーク
- ID: APのインデックス
- APC St: APC接続の現在の状態 (**Working**、**Busy**、**Offline**)
- AP MAC: APモートのEUI-64
- AP ID: マネージャによって割り当てられたAPモートの短縮アドレス
- AP St: APモートの現在の状態 (**Nego**、**Conn**、**Oper**、**Lost**)

3.8.5 show aps

説明

ネットワーク内にあるすべてのアクセス・ポイント(AP)のリストを表示します。このコマンドは、モートのリストを表示する `show notes` コマンドと似ています。-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。

構文

```
show aps [-p|--page]
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show aps
AP MAC                               Id Clk State  State time  Jn  Nbrs Links
-----
00-17-0D-00-00-60-35-28             1 Int  Oper   0-20:06:36  1   4    60
00-17-0D-00-00-60-37-72             3 Net  Oper   0-07:26:08  2   4    28

APs: 2. Live: 2, joining: 0, lost: 0
```

このコマンドは、ネットワーク内に現在ある(または過去にあった)APをすべて表示します。

- AP MAC: デバイスのEUI-64
- ID: マネージャによってこのデバイスに割り当てられた短縮アドレス
- Clk: APのクロック源 (**Int**、**Ext**、**Net**)
- State: 各デバイスの現在の状態 (**Nego**、**Conn**、**Oper**、**Lost**)
- State time: 現在までのデバイスの実行時間 (**d-hh:mm:ss**)。デバイスの状態が **Operational** の場合、このデバイスがどれだけ長くネットワークに参加しているかが State time に表示されます
- Age: マネージャがこのデバイスから最後のパケットを受信してから経過した秒数
- Jn: デバイスがネットワークに参加して **Operational** 状態まで進んだ回数
- Nbrs: このデバイスがアクティブ・リンクを持つ近隣モートの数
- Links: このデバイスに対するアクティブ・リンクの総数

3.8.6 show dcl

説明

マネージャの拒否制御リスト(deny control list)にブラックリストとして現在含まれている(参加が許可されない)モートの一覧を表示します。-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。MACアドレスを指定すると、そのデバイスのエントリのみが出力されます。

構文

```
show dcl [-p|--page] [macAddr]
```

パラメータ

パラメータ	説明
macAddr	MACアドレス <code>macAddr</code> を持つデバイスのみを出力(DCLに含まれる場合)

例


```
$> > show dcl
DCL:

MAC: 00-17-0D-00-0B-AD-0B-AD
```

3.8.7 show ini

説明

現在使用されている「アクティブな」INI設定を表示します。パラメータ`param`が指定されないか、アスタリスク(*)が使用された場合、すべてのパラメータを表示します。

 このコマンドにはスーパーユーザ権限が必要です。詳細については、[suコマンド](#)を参照してください。

構文

```
show ini [param | *]
```

パラメータ

パラメータ	説明
param	表示するINIパラメータ。「*」を指定すると、すべてのINIパラメータを表示します

例

```
#> show ini NUMBCAST
MNGRINI Module configuration :
NUMBCAST:2
```

3.8.8 show mote/ap

説明

このコマンドは、指定されたモートまたはアクセス・ポイント(AP)について、ネットワークおよび近隣モートの情報を返します。
-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。通常は、リンクされた近隣モートだけが表示されますが、-aまたは--all-neighborsオプションを使用すると、すべての近隣モート(リンク済みと検出済み)が表示されます。また、通常はパスあたりの総リンク数が表示されますが、-lまたは--linksオプションを使用すると、各リンクの情報が表示されます。

構文

```
show mote [-p|--page] [-a|--all-neighbors] [-l|--links] <macAddr|moteId>
show ap [-p|--page] [-a|--all-neighbors] [-l|--links] <macAddr|APIId>
```

パラメータ

パラメータ	説明
macAddrまたはmoteld	macAddrまたはmoteldで指定されたモートの情報を返します

例

```
$> show mote 00-17-0D-00-00-DD-EE-FF
MOTE #2, MAC: 00-17-0D-00-00-DD-EE-FF
  Version: 1.3.2.4 (stack 1.2.3.6)
  State: Oper, Hops: 1.0, State time: 0-00:02:55, Age: 30
  Power: 65534 (Routing)
  Power Cost: Max 65534, FullTx 65, FullRx 65, Used 401
  Capacity: 200 links, 31 neighbors
  Number of neighbors (parents, children) : 1 (1, 0)
  Bandwidth total / descendants (requested) : 14849 / 0 (27840)
  Number of links total, TX / RX / requested: 12, 3 / 0 / 1

Statistics:
  Reliability: 100.000% (0 lost, 15 total)
  Avg Latency: 1052 ms, 3712 ms est. to mote
  Voltage: 3306 mV
  Charge consumed: 29335 mC

Neighbors:
# 1 parent Q:75 links: 3 rssi:-41/-41 Ready

$> show ap 1
AP #1, MAC: 00-17-0D-00-00-60-39-7F
  Version: 1.4.0.76 (stack 1.4.0.5)
  Identity: apc-60397f
  State: Oper, Hops: 0.0, State time: 1-21:13:27
  Clock Source: Int
  Capacity: 1000 links, 499 neighbors
  Number of neighbors (parents, children) : 7 (0, 7)
  Descendant bandwidth : 758
  Number of links total, TX / RX : 50, NA / 30

Neighbors:
# 2 child Q: 94 links: 3 rssi:-47/-53
# 3 child Q: 95 links: 3 rssi:-58/-59
# 4 child Q: 96 links: 2 rssi:-42/-53
# 5 child Q: 92 links: 10 rssi:-54/-55
# 6 child Q: 96 links: 3 rssi:-45/-56
# 7 child Q: 91 links: 5 rssi:-50/-62
# 8 child Q: 93 links: 4 rssi: 0/-53
```

応答に含まれるフィールドは以下のとおりです。

一般:

- Mote #: モートのmoteld
- MAC: モートのEUI-64
- State: 現在の状態 - Idle、Negot1-2、Conn1-5、Oper、Lostのいずれか
- Hops: 上りデータの実証的な平均ホップ数
- State time: 最後の状態が変わってから経過した時間
- Age: マネージャがこのデバイスから最後にパケットを受信してから経過した時間(秒)
- Power: 電力(モートから返されるpowerSrcInfoパラメータのmaxStCurrent)、RoutingまたはLow Power (ルーティングに必要な電力よりmaxStCurrentが低い場合)。「ルーティング・タイプ」は、モートまたはマネージャに対して設定できます。モートまたはマネージャで、モートを非ルーティングとして宣言している場合、このモートには子モートやアダプタイズメント・リンクは割り当てられません。
- Power Cost: モートから返されるpowerSrcInfoパラメータ
- Capacity: モートでサポートされるリンクと近隣モートの最大数
- Number of neighbors: 親の近隣モート数 + 子の近隣モート数 = 近隣モート数。カッコ内の最初のエントリが親の近隣モート数、2番目のエントリが子の近隣モート数
- Bandwidth(ミリ秒/パケット): 合計(モート + 子孫) / 子孫。カッコ内はこのモートだけに要求された帯域幅。この値が低いほど、帯域幅は高くなります。上の例では、モートが要求した帯域幅を上回る帯域幅が割り当てられています
- Number of links: すべてのスロットフレームにわたる総リンク数、上りTxリンク数、上りRXリンク数、このモートが要求した帯域幅に相当するリンク数

Statistics:

- Reliability: 割合(%)、カッコ内は紛失数と総生成数
- Avg Latency: 上りの平均遅延、下りの(推定)平均遅延
- Voltage: 電源電圧(mV)
- Charge consumed: 最後にモートをリセットしてから消費した電気量(mC)

Neighbors:

- 近隣モートのmoteld、関係(親または子、検出のみの場合は「-」)、パス品質(%、パス安定性が測定されるまでは30%または75%)、近隣モートへの上りリンク数、他モートとのRSSI(From/To)、パス使用中(Ready)または検出のみ(Not Ready)

3.8.9 show motes

説明

ネットワーク内にあるすべてのモートのリストを表示します。-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。

構文

```
show motes [-p|--page]
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show motes
Mote MAC                Id  State  State time  Age  Jn  Nbrs  Links
-----
00-17-0D-00-00-30-4B-1D  2  Oper   0-20:10:32  12   1   4     20
00-17-0D-00-00-30-44-A4  4  Oper   0-20:09:56  14   1   4     20
00-17-0D-00-00-30-5A-BD  5  Oper   0-20:09:53   0   1   4     45

Motes: 3. Live: 3, joining: 0, lost: 0

$> show aps
AP MAC                Id  Clk  State  State time  Jn  Nbrs  Links
-----
00-17-0D-00-00-60-35-28  1  Int  Oper   0-20:11:28   1   4     60
00-17-0D-00-00-60-37-72  3  Net  Oper   0-07:31:00   2   4     28

APs: 2. Live: 2, joining: 0, lost: 0
```

このコマンドは、ネットワーク内に現在ある(または過去にあった)モートをすべて表示します。

- MAC: デバイスのEUI-64
- MotelID: マネージャによってこのデバイスに割り当てられた短縮アドレス
- State: 各デバイスの現在の状態(**Nego**、**Conn**、**Oper**、**Lost**)
- State time: デバイスが現在の状態まで進んでからの時間(**d-hh:mm:ss**)。デバイスの状態が**Operational**の場合、このモートがどれだけ長くネットワークに参加しているかがState timeに表示されます
- Age: マネージャがこのモートから最後のパケットを受信してから経過した秒数
- Jn: デバイスがネットワークに参加して**Operational**状態まで進んだ回数
- Nbrs: このデバイスがアクティブ・リンクを持つ近隣モートの数
- Links: このデバイスに対するアクティブ・リンクの総数

3.8.10 show network

説明

ネットワーク構成および統計に関する情報を表示します。

構文

```
show network
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show network
Network configuration:
  networkId:          1229
  topologyType:      MESH
  downFrameMultiplierDelay: 3600000
  ccaMode:           False
  ipAddrPrefix:      FE80::
  basePkPeriod:      15000
  downFrameMultiplier: 1
  joinSecurityType:  COMMON_SKEY
  minServicePkPeriod: 100
  downFrameSize:     512
  numParents:        2
  channelList:       32767
  upFrameSize:       1024

Network statistics:
  Network start time: 2015-12-04 15:13:42.408, uptime 4-01:32:49
  Live notes:        12
  Reliability:       100.000% (0 lost, 477682 total)
  Avg Latency:       1168 ms
  Path stability:    81.164%
  Advertising:       On
  Queue (net/user): 0/0
  Current frame size: 512
```

Network configuration

詳細については、`config set network parameters`を参照してください。

Network statistics

- Network start time: 開始時刻およびアップタイム
- Live nodes: **Operational**状態にあるモートの数(APではない)
- Reliability: ネットワーク信頼性は、モートが生成したパケットのうち、マネージャに正しく受信されたものの割合(%)として計算されます。健全なネットワークでは、この数値が99.99%以上になります。この統計情報は、受信するパケットのセキュリティ・カウンタを監視することで、マネージャに対して直接カウントされます
- Avg Latency: マネージャが受信したパケットの平均上り遅延。マネージャはパケット・ヘッダの生成タイムスタンプ(ASN)をチェックし、現在のASNと比較することで、個々のパケットの遅延を計算します
- Path stability: ネットワーク・パスの安定性は、総MAC伝送数のうち成功した伝送の割合(%)です。この数値は、それぞれのモート配置によって異なります。このネットワークは、たとえ安定性が50%でも100%の信頼性を達成するように設計されています。安定性が低い場合、再試行が必要なパケット数が多くなるので、モートはより多くの電力と帯域幅を使用します。この統計情報は、モートから返される健全性レポートに基づいて計算されます。マネージャは、新しい健全性レポートを受け取るたびに、伝送と失敗の総数を増やします。
- Advertising: 現在のアドバタイズメントの状態
- Queue: マネージャ・キューに含まれるパケット数(マネージャ生成パケットとユーザ生成パケットの両方)
- Current frame size: 下りフレーム・サイズ(スロット)。`downFrameMultiplier > 1`の場合、安定状態で下りフレームを長くすることができます

3.8.11 show paths

説明

このコマンドは、指定されたモートに対するすべてのパスの情報を表示します。-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。-aまたは--allオプションを使用すると、未使用の潜在パスのリストが追加で表示されます。

構文

```
show paths [-p|--page] [-a|--all] <macAddr|motelId>
```

パラメータ

パラメータ	説明
macAddrまたはmotelId	デバイスのmacAddrまたはmotelId

例

```
$> show paths 00-17-0D-00-00-AA-BB-CC
Neighbors:
# 2 child Q:82 links: 3 rssi:-41/-41 Ready
# 3 child Q:75 links: 4 rssi: 0/ 0 Not Ready
```

このコマンドは、指定されたデバイスとそのアクティブ近隣モート間のパスをすべて表示します。各行に、近隣モートのモートID、関係（親または子）、パスに割り当てられたリンクの数、パス品質の統計情報、RSSIが表示されます。

この例に表示されたモート2は、このモートの子で、成功／失敗の統計情報に基づいて計算された品質が82%（パス安定性）で、3つのリンクが割り当てられており、RSSIは-41dBです。

3.8.12 show route

説明

マネージャから指定されたモートまでのソース・ルートを表示します。

構文

```
show route <macAddr|moteId>
```

パラメータ

パラメータ	説明
macAddrまたはmotelId	macAddrまたはmotelIdで指定されたモートまでのソース・ルートの表示

例

```
S> show route 20
Route for Mote #20, 00-17-0D-00-00-38-00-D9
-> AP # 1, 00-17-0D-00-00-60-36-D7
-> MOTE # 7, 00-17-0D-00-00-38-00-55
-> MOTE # 11, 00-17-0D-00-00-38-00-B6
-> MOTE # 14, 00-17-0D-00-00-38-00-C0
-> MOTE # 17, 00-17-0D-00-00-38-00-C9
```

それぞれのホップに対して1行出力されます。この例では、motelId = 20までのルートは、1-7-11-14-17-20になります。

3.8.13 show services

説明

指定されたモートのサービス情報を表示します。

構文

```
show services <macAddr|moteld>
```

パラメータ

パラメータ	説明
macAddrまたはmoteld	macAddrまたはmoteldで指定されたモートのサービス情報を表示

例

```
$> show services 00-17-0D-00-00-DD-EE-FF
Services:
# 0 MAC: 00-17-0D-00-00-00-FF-FE
  Allocated BW: 27840
  Latency:      1737 (1 hops)
```

3.8.14 show sessions

説明

現在のユーザ・セッションのリストを表示します。ユーザがコンソールにログインするか、外部APIで認証されるたびに、セッションが1つ作成されます。

i 外部APIは、APIリクエストに認証ヘッダが含まれる場合は常に新規ユーザ・セッションを生成します。このため、*show sessions*の出力には多数のセッションが表示される場合があります。

内部では、ユーザ・セッションの数が活動に基づく最大数に制限されているため、セッションが多くなりすぎると古いセッションが削除されます。

構文

```
show sessions
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show sessions
User          Last activity          Login          Client Info
-----
dust          2016-05-24 11:10:44.542 2016-05-24 11:10:44.542
```

3.8.15 show system

説明

システム開始時刻およびアップタイム、ユーザ指定のシステム名文字列、ユーザ指定のロケーション文字列、CLIタイムアウトなどのシステム情報を表示します。表示されるフィールドは、`config set system`コマンドを使用して設定できます。

構文

```
show system
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show system System
information:
  System start: 2015-12-04 15:13:42.433, Uptime:    3-23:15:12
  sysName:      Thermall
  location:     Unit5
  cliTimeout:   0 minutes
```

3.8.16 show time

説明

サーバ・システム時間とネットワーク(SP)時間の2種類の時間を表示します。

構文

```
show time
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show time  
System start : 2015-12-01 15:10:22.627, System uptime : 0-19:43:07  
Network start: 2015-12-01 16:50:40.717, Network uptime: 0-18:02:49
```


3.8.17 show unjoined

説明

参加カウンタがconfigdbに保存されているが、現在のトポロジに含まれていないモートを表示します。通常の使用ではこのような状態にはなりません。このコマンドは開発作業などで有効なコマンドです。(例、モートソフトウェアアップデートやネットワークリセット時)

構文

```
show unjoined
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show unjoined  
No entries found
```

3.8.18 show user

説明

指定されたユーザの情報を表示します。

構文

```
show user <userId>
```

パラメータ

パラメータ	説明
userId	userIdで指定されたユーザの情報を表示

例

```
$> show user dust
User information:
  Id:      dust
  Password: ****
  Privilege: USER
  Description: duster
```

3.8.19 show users

説明

すべてのユーザの情報を表示します。-pまたは--pageオプションを使用すると、ページごとに出力が一時停止します。

構文

```
show users [-p|--page]
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> show users
User configuration:
Id:      dust
```

3.8.20 show ver

説明

VManagerの個別ソフトウェア・コンポーネントとVManagerパッケージのバージョンを表示します。

構文

```
show ver [component]
```

パラメータ

パラメータ	説明
component	空白にしてすべてを表示するか、コンポーネント名を指定します。ほとんどの場合、コンポーネント名は出力で使用される名前と一致します。外部APIコンポーネントを指定する場合、 <i>apiserver</i> を使用します。

例

```
$> show ver
Package version:      1.1.0.8
External API version: 1.1.0.8 (built 2017/03/02 11:16:10)
AuthManager version: 1.1.0.8 (built 2017/03/02 11:16:13)
ConfigDB version:    1.1.0.8 (built 2017/03/02 11:16:17)
Console version:      1.1.0.8 (built 2017/03/02 11:16:21)
Manager version:      1.1.0.8 (built 2017/03/02 11:15:44)
Watchdog version:     1.1.0.8 (built 2017/03/02 11:16:23)

$> show ver console
Console version:      1.1.0.8 (built 2017/03/02 11:16:21)
```

3.9 sm

説明

このコマンドは、ネットワークに含まれるすべてのモートとアクセス・ポイント(AP)のリストを表示します。[show notes](#)コマンドや[show aps](#)コマンドと似ています。

構文

```
sm [-p|--page]
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません

例

```
$> sm

AP MAC                Id Clk State  State time  Age Jn  Nbrs Links
-----
00-17-0D-00-00-60-37-9C  1 Int  Oper   0-00:01:29 112  1   3   15

Mote MAC              Id State  State time  Age Jn  Nbrs Links
-----
00-17-0D-00-00-30-08-2D  2 Oper   0-00:03:02  11  1   1   10
00-17-0D-00-00-30-08-05  3 Nego   0-00:00:02  23  1   0   0

APs: 1, Motes: 2, 1 live, 1 joining
```

このコマンドは、ネットワーク内に現在ある(または過去にあった)デバイスをすべて表示します。

- MAC: デバイスのEUI-64
- MotelD: マネージャによってこのデバイスに割り当てられた短縮アドレス
- Clk: クロック源 (APのみ)
- State: 各デバイスの現在の状態 (**Nego**、**Conn**、**Oper**、**Lost**)
- State time: デバイスが現在の状態まで進んでからの時間(**d-hh:mm:ss**) デバイスの状態が**Operational**の場合、このモートがどれだけ長くネットワークに参加しているかがState timeに表示されます
- Age: マネージャがこのモートから最後のパケットを受信してから経過した秒数APの場合、この値が大きくなる場合があります
- Jn: デバイスがネットワークに参加して**Operational**状態まで進んだ回数
- Nbrs: このデバイスがアクティブ・リンクを持つ近隣モートの数
- Links: このデバイスに対するアクティブ・リンクの総数

3.10 su

説明

スーパーユーザ・モードに切り替えます。スーパーユーザ・モードでは、主にデバッグやテストに使用するいくつかのコマンドを使用できます。スーパーユーザ・モードに切り替わったことを示すため、コンソールのプロンプトが\$>から#>に変わります。

構文

```
su becareful
```

パラメータ

パラメータ	説明
	このコマンドにパラメータはありません。

例

```
$> su becareful  
#>
```

3.11 trace

説明

コンソールへのトレース出力を有効化／無効化します。この設定を有効にするには、先にset loglevelコマンドを使用してトレースをアクティブ化する必要があります。subscribeコマンドを入力すると非常に多くのトレースを使用できますが、これらの大半は内部のデバッグを目的としたものです。一般的な用途に適したトレースは以下のとおりです。

TraceId	説明
mngr.net.io.data	ユーザ・データ・パケット
mngr.tplgdb.mote	モートの状態変化(参加プロセスにおけるモートの進行状況など)
mngr.tplgstat	上り遅延およびホップ情報

構文

```
set loglevel <traceId> TRACE
trace [<traceId> <on|off>]
```

パラメータ

パラメータ	説明
traceId	traceId/トレースの有効化
onまたはoff	on = トレースの有効化、off = トレースの無効化


例

```
$> set loglevel mngr.tplgstat TRACE
Done
$> trace mngr.tplgstat on
Trace enabled for mngr.tplgstat

2015-12-15 15:59:06.698 mngr.tplgstat: L_TRACE Stat. Mote #12 ASN packet: 58559826934 current:
58559827083
2015-12-15 15:59:06.702 mngr.tplgstat: L_TRACE Stat. Mote #12 AP#1 Hops. New hops: 50 new average:
64
2015-12-15 15:59:06.705 mngr.tplgstat: L_TRACE Stat. Mote #12 Latency. New latency: 1080250 usec.
New average: 871756 usec.
```

この例では、マネージャがmoteId12からパケットを受信しています。マネージャは現在のASNをパケット・ヘッダ内のASNと比較し、上り遅延を μs で計算します。次に、ヘッダのTTLフィールドを使用して、APに到達するまでのホップ数を計算し、この値を01ホップ単位で出力します。上記例では、パケット・ホップが50と出力されており、パケットがAPに到達するまでに5つのホップを経由したことが示されています。このトレースには、IIRフィルタを使用した平均の上り遅延とホップ数も表示します。

商標

Eterna、Mote-on-Chip、SmartMesh IPは、Dust Networks, Incの商標です。Dust Networksロゴ、Dust、Dust Networks、SmartMeshは、Dust Networks, Incの登録商標です。LT、LTC、LTM、は、アナログ・デバイセズの登録商標です。第三者のブランド名および製品名は各社の商標であり、情報提供のみを目的として使用されています。

著作権

本書は、米国著作権法、国際著作権法、その他の知的財産法および産業財産法によって保護されています。本書はアナログ・デバイセズおよびその実施許諾者によって専有されており、制限付きライセンスに従って配布されます。アナログ・デバイセズの書面による事前の認可なく、本書の全部または一部を使用、複製、変更、逆アSEMBル、逆コンパイル、リバース・エンジニアリング、配布、再配布することは、その形式、手段にかかわらず禁じられています。

制限付き権利: 米国政府による使用、複製、開示は、FAR 52.227-14(g)(2)(6/87)およびFAR 52.227-19(6/87)、またはDFAR 252.227-7015(b)(6/95)およびDFAR 227.7202-3(a)ならびにこれに準ずる法律および規制と後継の法律および規制に規定された制限の対象となります。

免責事項

本書は現状のまま提供され、明示、暗示を問わず一切の保証を行なわないものとします。かかる保証には、特定目的に対する商品性または適合性の黙示的保証が含まれますが、これに限定されません。

本書には技術的な誤りやその他の間違いが含まれる場合があります。訂正と改善は、新しいバージョンの文書に取り入れられる可能性があります。

アナログ・デバイセズは、製品やサービスの適用または使用により発生する責任を負いかねます。また、間接的あるいは偶発的損害を含むがそれに限定されない、いかなる責任も負わないものとします。

アナログ・デバイセズの製品は、誤動作がユーザの深刻な人身傷害につながると合理的に予想できる生命維持装置、デバイス、またはその他のシステムでの使用、またはその機能不全により生命維持装置またはシステムの故障あるいはその安全性や有効性に影響すると合理的に予想できる生命維持装置またはシステムの重要な部品としての使用を目的として設計されていません。このような用途での使用を目的としてこれらの製品を使用または販売しているアナログ・デバイセズの顧客は、顧客自身の責任でそれを行い、このような意図しないまたは不正な使用に関連する人身傷害または死亡に直接または間接的に起因するすべての主張、費用、損害、支出、および妥当な額の弁護士費用、また、かかるクレームでアナログ・デバイセズに該当製品の設計または製造に関わる過失があったと主張される場合でも、これを完全に補償し、アナログ・デバイセズとその役員、従業員、子会社、関連会社、および販売代理店に何ら損害を与えないことに同意するものとします。

アナログ・デバイセズは、いつでも製品またはサービスに対する修正、変更、拡張、改良、その他の変更を行う権利を保有し、製品またはサービスを予告なく中止する権利を有します。顧客は、発注の前に最新の関連情報を入手し、その情報が最新で完全であることを確認する必要があります。すべての製品は、注文承諾時または販売時に提供される、販売に関するDust Networkの契約条件に従い販売されます。

アナログ・デバイセズは、アナログ・デバイセズの製品またはサービスが使用される組み合わせ、マシン、またはプロセスに関連するアナログ・デバイセズの特許、著作権、マスクワーク権、その他のアナログ・デバイセズの知的所有権に従って、明示か黙示かにかかわらず、ライセンスが付与されることを保証または主張するものではありません。第三者の製品またはサービスに関してアナログ・デバイセズが公開した情報は、その製品またはサービスを使用するためのアナログ・デバイセズからのライセンス提供、あるいはその保証または推奨を意味するものではありません。このような情報を使用する場合、第三者の特許または他の知的所有権に従って第三者からのライセンスが必要になるか、またはアナログ・デバイセズの特許または他の知的所有権に従ってアナログ・デバイセズからのライセンスが必要になります。

Dust Networks, Incは、アナログ・デバイセズの完全所有子会社です。

© Analog Devices, Inc. 2012-2016 All Rights Reserved.