

# Cisco Expressway サービスアビリティ ガイド

メンテナンスおよび操作ガイド

最終更新日：2016年12月

X8.9

目次

## 目次

目次 .....	2
はじめに.....	3
変更履歴 .....	3
システム メトリックの収集の概要 .....	4
<b>Expressway</b> でのシステム メトリックの収集の設定 .....	5
リモート サーバでのシステム メトリックの設定.....	5
システム メトリックのトラブルシューティング.....	6
コール詳細レコード (CDR) .....	7
CDR の設定方法 .....	7
CDR にアクセスする API .....	7
<b>Smart Call Home (SCH)</b> .....	9
<b>Smart Call Home</b> の概要 .....	9
参照先 .....	10
コール詳細レコードの例 .....	10
システム メトリックの参考資料 .....	12
シスコの法的情報 .....	15
シスコの商標または登録商標.....	16

はじめに

## はじめに

### 変更履歴

表 1：メンテナンスおよび操作ガイドの変更履歴

日付	変更内容	理由
2016 年 12 月	Smart Call Home に関する情報を追加。	X8.9 をリリース。
2015 年 11 月	コール詳細レコード情報を追加。	X8.7 をリリース。
2015 年 7 月	システム メトリック機能を掲載した初版。	X8.6 をリリース。

システム メトリックの収集の概要

## システム メトリックの収集の概要

システム メトリックの収集と、Expressway で機能する仕組み

システム メトリックの収集は、システム パフォーマンスに関する統計情報をパブリッシュする機能で、パフォーマンスをリモートからモニタできるようにします。

Expressway は、ハードウェア、OS、およびアプリケーションのパフォーマンスに関する統計情報を収集し、データを集約するリモート ホスト（通常はデータ分析サーバ）にそれらの統計情報をパブリッシュします。

システム メトリックの収集の設定場所

この機能は Web インターフェイスまたはコマンド ラインで設定できます。1 つのピアの設定がクラスタ全体に適用されるので、クラスタをモニタする場合は、プライマリ ピアで設定を行うことを推奨します。

さらに、リモート サーバではいくつかの設定も必要になります。つまり、サーバで `collectd` デーモンを実行する必要があります。また、クライアントが可視できるアドレス上でリッスンするように `collectd` ネットワーク プラグインを設定する必要があります。詳細については、モニタする環境によって異なるためここでは割愛します。

このデータの使用方法

データを使用してグラフを生成したり、統計情報を集約したり、パフォーマンスを分析したりできます。これには、Circonus や Graphite などのツールを使用します。

## Expressway でのシステム メトリックの収集の設定

## Expressway でのシステム メトリックの収集の設定

以下の手順では、統計情報を収集して特定のサーバにパブリッシュするように、Web インターフェイスを使用して Expressway を設定します。オプションの詳細については、「[システム メトリック参考資料](#)」（12 ページ）を参照してください。

1. Expressway にログインし、[Maintenance] > [Logging] に移動します。
2. [System Metrics Collection] を [On] に切り替えます。
3. [Collection server address] に入力します。  
IP アドレス、ホスト名、または FQDN を使用してリモート サーバを特定できます。
4. 必要に応じて、[Collection Interval] および [Collection server port] を変更します。  
収集サーバがデフォルト（25826）とは異なるポートでリッスンしている場合は、ポートを変更する必要があります。ポ  
リシーでデフォルトの間隔（60 秒）よりも詳細なメトリックが必要な場合は、収集間隔を変更する必要があります。
5. [Save] をクリックします。

## リモート サーバでのシステム メトリックの設定

各自の環境でデータ分析に使用するサーバの選択および設定は、このマニュアルでは扱いません。Circonus および Graphite は、collectd の情報を処理できるアプリケーションです。

使用する分析ツールが、collectd デーモンからのデータの受信をサポートしている必要があります。このデーモンは Expressway で実行され、collectd ネットワーク プラグインを使用してメトリックを分析サーバにプッシュします。

このネットワーク プラグインは、データのカプセル化のための collectd バイナリ プロトコルを実装します。分析サーバは、このデータを解析して提示できなければなりません。分析サーバには、collectd または代替のソフトウェアに基づいてデータの収集方法や表示方法を設定する独自の UI が備わっていることがあります。

分析サーバで collectd を使用する場合は、サーバが次のように動作するように collectd.conf ファイルを変更する必要があります。

- collectd クライアント（Expresswa など）からのデータをリッスンする：ネットワーク プラグインを有効にし、サーバの IP アドレスを使ってリッスン ブロックを設定する必要があります。次に例を示します。

```
<Plugin "network">
  Listen "198.51.100.15"
</Plugin>
```

- 受信データを人間が判読できる形式（CSV ファイルなど）で保存する：csv プラグインを有効にして、ファイルを書き込む場所を指定する必要があります。次に例を示します。

```
<Plugin "csv">
  DataDir "/var/lib/collectd/csv"
  StoreRates true
</Plugin>
```

## Expressway でのシステム メトリックの収集の設定

### 参照

- [https://collectd.org/wiki/index.php/Networking\\_introduction](https://collectd.org/wiki/index.php/Networking_introduction)
- [https://collectd.org/documentation/manpages/collectd.conf.5.shtml#plugin\\_network](https://collectd.org/documentation/manpages/collectd.conf.5.shtml#plugin_network)
- [https://collectd.org/wiki/index.php/Binary\\_protocol](https://collectd.org/wiki/index.php/Binary_protocol)
- <https://collectd.org/wiki/index.php/Plugin:CSV>
- [https://collectd.org/documentation/manpages/collectd.conf.5.shtml#plugin\\_csv](https://collectd.org/documentation/manpages/collectd.conf.5.shtml#plugin_csv)

## システム メトリックのトラブルシューティング

**Expresswa がデータを送信しているか。**

Expresswa から TCP ダンプを取得し、データ分析サーバのアドレスにパケットが送信されたことを確認します。

[Maintenance] > [Diagnostics] > [Diagnostics logging] に移動し、[Take tcpdump while logging] というラベルの付いたボックスをオンにして、ロギングを開始します。

## コール詳細レコード (CDR)

# コール詳細レコード (CDR)

サービス (デフォルトはオフ) を有効にすると、システムで CDR をキャプチャできます。さらに、リモート ロギングを使用している場合は、それらを syslog メッセージとしてパブリッシュできます。

[Service only] を選択すると、CDR が 7 日間保管されます。それらの CDR は、Representational State Transfer (REST) API を介してのみ、Expressway に読み込むことができます。[Service and logging] を選択した場合、ローカル データはイベント ログで公開され、CDR も INFO メッセージとして syslog ホストに送信されます。

## CDR の設定方法

Expressway で CDR を設定するには、次の手順を実行します。

1. [Maintenance] > [Logging] に移動します。
2. [Logging Options] セクションで、次の説明に従って [Call Detail Records] フィールドを設定します。

CDR モード	説明
消灯	CDR はローカルにログ記録されません (デフォルト)。
サービスのみ	CDR はローカルに 7 日間保存され、その後削除されます。Web GUI を通じてレコードにアクセスできません。
サービスとロギング	CDR はローカルに 7 日間保存され、その後削除されます。レコードにローカル イベント ログからアクセスできます。外部ロギングが有効になっている場合は外部 syslog サーバからもアクセスできます。

## CDR にアクセスする API

次の安全な REST API を使用して必要な情報を収集できます。

- `get_all_records` (最長 7 日前までのすべてのレコードを返します)。
- `get_records_for_interval` (指定された時間内のレコードを返します)。
- `get_records_for_filter` (任意の組み合わせを使用して結果をフィルタリングします)。
- `get_all_csv_records` (最長 7 日前までのすべてのレコードを csv 形式で返します)。

目的の API にアクセスするには、`https://<Expressway_IP>/api/external/callusage/<API>` を使用します。

### 例

- `http://<Expressway_IP>/callusage/get_all_records`
- `http://<Expressway_IP>/callusage/get_records_for_interval?fromtime=<fromtime>&totime=<to_time>` (for example `http://10.50.157.80:8000/callusage/get_records_for_interval?fromtime=2014-05-09%2000:00:00&totime=2014-05-10%2000:00:00`)

## コール詳細レコード (CDR)

## 入力パラメータ

パラメータ	説明
fromtime	CDR レコードが必要な期間の開始時刻。「YYYY-MM-DD HH:MI:SS」の形式で入力する必要があります (必須パラメータ)。
totime	CDR レコードが必要な期間の終了時刻。「YYYY-MM-DD HH:MI:SS」の形式で入力する必要があります (必須パラメータ)。

- [http://<Expressway\\_IP>/callusage/get\\_records\\_for\\_interval?fromtime=<fromtime>&totime=<to\\_time>](http://<Expressway_IP>/callusage/get_records_for_interval?fromtime=<fromtime>&totime=<to_time>) (for example [http://10.50.157.80:8000/callusage/get\\_records\\_for\\_interval?fromtime=2014-05-09%2000:00:00&totime=2014-05-10%2000:00:00](http://10.50.157.80:8000/callusage/get_records_for_interval?fromtime=2014-05-09%2000:00:00&totime=2014-05-10%2000:00:00))
- [http://<Expressway\\_IP>/callusage/get\\_records\\_for\\_filter?uid=<uid>&src\\_alias=<src\\_alias>&dest\\_alias=<dest\\_alias>&protocol=<protocol>](http://<Expressway_IP>/callusage/get_records_for_filter?uid=<uid>&src_alias=<src_alias>&dest_alias=<dest_alias>&protocol=<protocol>) (for example [http://10.50.157.80:8000/callusage/get\\_records\\_for\\_filter?uid=6e3b5a8a-346c-421b-aa2e-f4409c43a81a&src\\_alias=TC149-057-h323@domain.com&dest\\_alias=TC149-065-h323@domain.com&protocol=H323%20%3C-%3E%20H323](http://10.50.157.80:8000/callusage/get_records_for_filter?uid=6e3b5a8a-346c-421b-aa2e-f4409c43a81a&src_alias=TC149-057-h323@domain.com&dest_alias=TC149-065-h323@domain.com&protocol=H323%20%3C-%3E%20H323))

## 入力パラメータ

パラメータ	説明
uuid	レコードの固有識別子。
src_alias	コールの発信ポイント。
dest_alias	コールの宛先ポイント。
protocol	コールに使用されたプロトコル (SIP、H323 など)。

- [http://<Expressway\\_IP>/callusage/get\\_all\\_csv\\_records](http://<Expressway_IP>/callusage/get_all_csv_records)

## 制限事項

- 通話履歴は 7 日間だけローカルに保存され、その後自動的に削除されます。

CDR の例およびプロパティの定義については、「追加情報」の項の「[コール詳細レコードの例](#) (10 ページ) を参照してください。



Smart Call Home (SCH)

## Smart Call Home (SCH)

### Smart Call Home の概要

Smart Call Home は、無償で利用できる Expressway の組み込みサポート機能です。プロアクティブな診断とリアルタイムのアラートを提供し、高いネットワーク可用性と運用効率の向上を実現します。

Smart Call Home は、スケジュールベースの通知とイベントベースの通知をユーザに送信します。

- スケジュールベース：インベントリ、テレメトリー、および設定に関するメッセージです。これらのメッセージを使用してデバイス レポートを生成し、障害の傾向を特定することでハードウェアとソフトウェアの品質を向上させます。これらの通知は、毎月 1 日に送信されます。
- イベントベース：Expressway ですでにサポートされている非同期イベントです（アラームや ACR など）。これらの通知は、イベントが発生すると Smart Call Home サーバにポストされます。

### Smart Call Home の設定方法

Smart Call Home をオンにするには、次の手順を実行します。

1. [Maintenance] > [Serviceability] > [Smart Call Home] に移動します
2. [Configuration] セクションの [Smart Call Home] フィールドで、モードを選択します。

モード	定義
Off (デフォルト)	Expressway は Smart Call Home サーバに情報を送信しません。
点灯	Smart Call Home サービスを起動します。[Replace Smart Call Home server certificate] と [Your Contact Details] オプションが表示されます。
On (匿名)	匿名モードで Smart Call Home サービスを起動します。[Replace Smart Call Home server certificate] オプションが表示されます。Expressway は Smart Call Home にレポートを送信しますが、Smart Call Home サーバが通知を送信できないように顧客の詳細は匿名のままになります。

**注：** Expressway には Smart Call Home サーバ証明書がインストールされているので、Smart Call Home サーバと安全に通信できます。Smart Call Home サーバで証明書を更新しない限り、証明書を置き換える必要はありません。

3. Smart Call Home を [On] にすることを選択した場合は、最小要件として、電子メールアドレスを [Your Contact Details] セクションに入力する必要があります。
4. [Save] をクリックします。

参照先

## 参照先

### コール詳細レコードの例

#### CDR の例

```
[{"initial_call": "false", "protocol": "SIP <-> SIP", "protocol_summary": "", "disconnect_reason": "200 OK",
"licensed": "false", "tag": "b8d52a60-16a1-4bdb-be93-f5a675408811", "aside_request_uri": "",
"box_call_serial_number": "22cd0e7d-c498-4068-9239-624038fe5130", "source_alias":
"sip:10000005@10.196.4.82", "uuid": "800fe013-83f4-4094-a5e6-e2f9489912e2", "last_updated_timestamp":
1444725389, "details": {"Call":{"SerialNumber":"800fe013-83f4-4094-a5e6-
e2f9489912e2"},"BoxSerialNumber":"22cd0e7d-c498-4068-9239-624038fe5130"},"Tag":"b8d52a60-16a1-
4bdb-be93-f5a675408811"},"State":"Disconnected"},"StartTime":"2015-10-13
01:36:26.485636"},"InitialCall":{"False"},"Licensed":{"False"},"LicensedAsTraversal":
{"False"},"SourceAlias":{"sip:10000005@10.196.4.82"},"DestinationAlias":{"sip:10000010@cucm-
82"},"ToLocalB2BUA":{"False"},"Audio":{"False"},"License":{"Traversal":{"0"},"NonTraversal":
{"0"},"DemotedTraversal":{"0"},"CollaborationEdge":{"0"},"Cloud":{"0"},"Duration":
{"3"},"Legs":[{"Leg":{"Protocol":"SIP"},"SIP":{"Address":{"10.196.4.61:5073"},"Transport":
{"TLS"},"Aliases":{"Alias":{"Type":"Url"},"Origin":{"Unknown"},"Value":
{"sip:10000005@10.196.4.82"}}}]}]},"Targets":[{"Target":{"Type":"Url"},"Origin":{"Unknown"},"Value":
{"sip:10000010@10.196.4.116}}]}]},"BandwidthNode":{"DefaultZone"},"EncryptionType":{"AES"},"Cause":
{"200"},"Reason":{"OK}}]},"Leg":{"Protocol":"SIP"},"SIP":{"Address":{"10.196.4.71:7001"},"Transport":
{"TLS"},"Aliases":{"Alias":{"Type":"Url"},"Origin":{"Unknown"},"Value":{"sip:10000010@cucm-
82"}}}]},"Source":{"Aliases":{"Alias":{"Type":"Url"},"Origin":{"Unknown"},"Value":
{"10000005@10.196.4.82"}}}]},"BandwidthNode":{"Traversal-zone"},"EncryptionType":{"AES"},"Cause":
{"200"},"Reason":{"OK}}]},"Sessions":[{"Session":{"Status":{"Completed"},"MediaRouted":
{"False"},"CallRouted":{"True"},"Participants":{"Leg":{"1"},"Leg":{"2"},"Incoming":{"Leg":
{"1"},"Outgoing":{"Leg":{"2}}}}]}]},"EndTime":"2015-10-13 01:36:29.745651"},"status": "Disconnected",
"destination_alias": "sip:10000010@cucm-82", "licensed_as_traversal": "false", "service_uuid": "e6723fd0-5ca2-
11e1-b86c-0800200c9a66", "start_time": "2015-10-13 01:36:26.485636", "traversal_license_tokens": 0,
"bside_destination_alias": "", "active": "false", "media_routed": "false", "aside_destination_alias": "",
"non_traversal_license_tokens": 0, "bside_request_uri": "", "end_time": "2015-10-13 01:36:29.745651", "audio":
"false"}]
```

注：上記の CDR の例は、csv を除いてすべての API に適用されます。

#### csv CDR の例

uuid,service\_uuid,active,initial\_call,licensed,licensed\_as\_traversal,status,tag,box\_call\_serial\_number,start\_time,end\_time,s  
ource\_alias,destination\_alias,aside\_destination\_alias,bside\_destination\_alias,aside\_request\_uri,bside\_request\_uri,protocol  
\_summary,protocol,media\_routed,audio,traversal\_license\_tokens,non\_traversal\_license\_tokens,disconnect\_reason,details,l  
ast\_updated\_timestamp

```
800fe013-83f4-4094-a5e6-e2f9489912e2,e6723fd0-5ca2-11e1-b86c-
0800200c9a66,false,false,false,false,Disconnected,b8d52a60-16a1-4bdb-be93-f5a675408811,22cd0e7d-c498-4068-
9239-624038fe5130,2015-10-13 01:36:26.485636,2015-10-13
01:36:29.745651,sip:10000005@10.196.4.82,sip:10000010@cucm-82,,,,,SIP <-> SIP,false,false,0,0,200
OK,"{"Call":{"SerialNumber":"800fe013-83f4-4094-a5e6-e2f9489912e2"},"BoxSerialNumber":"22cd0e7d-
c498-4068-9239-624038fe5130"},"Tag":"b8d52a60-16a1-4bdb-be93-f5a675408811"},"State":
"Disconnected"},"StartTime":"2015-10-13 01:36:26.485636"},"InitialCall":{"False"},"Licensed":
{"False"},"LicensedAsTraversal":{"False"},"SourceAlias":{"sip:10000005@10.196.4.82"},"DestinationAlias":
{"sip:10000010@cucm-82"},"ToLocalB2BUA":{"False"},"Audio":{"False"},"License":{"Traversal":
{"0"},"NonTraversal":{"0"},"DemotedTraversal":{"0"},"CollaborationEdge":{"0"},"Cloud":
{"0"},"Duration":{"3"},"Legs":[{"Leg":{"Protocol":"SIP"},"SIP":{"Address":
"10.196.4.61:5073"},"Transport":{"TLS"},"Aliases":{"Alias":{"Type":"Url"},"Origin":
```

## 参照先

```

""Unknown"", ""Value"": ""sip:10000005@10.196.4.82"" }}, ""Targets"": [{"Target": {"Type": "Url", "Origin":
""Unknown"", ""Value"": ""sip:10000010@10.196.4.116"" }}, ""BandwidthNode"":
""DefaultZone"", ""EncryptionType"": ""AES"", ""Cause"": ""200"", ""Reason"": ""OK""}, {"Leg": {"Protocol":
""SIP"", ""SIP"": {"Address": ""10.196.4.71:7001"", ""Transport"": ""TLS"", ""Aliases"": [{"Alias": {"Type":
""Url"", ""Origin"": ""Unknown"", ""Value"": ""sip:10000010@cucm-
82"" }}, {"Source": {"Aliases": [{"Alias": {"Type": "Url", "Origin": ""Unknown"", ""Value"":
""10000005@10.196.4.82"" }}, {"BandwidthNode": ""Traversal-zone"", ""EncryptionType"": ""AES"", ""Cause"":
""200"", ""Reason"": ""OK"" }}, {"Sessions": [{"Session": {"Status": ""Completed"", ""MediaRouted"":
""False"", ""CallRouted"": ""True"", ""Participants"": {"Leg": ""1"", ""Leg": ""2"", ""Incoming"": {"Leg":
""1""}, ""Outgoing"": {"Leg": ""2"" }}}} ], ""EndTime"": ""2015-10-13 01:36:29.745651"" }}, 1444725389

```

## 定義

次の表は、CDR に表示されるプロパティの定義を示しています。

フィールド	定義
uuid	CDR エントリの ID です。
service_uuid	レコードの収集元（プロキシ、Lync 2BUA、または暗号化 B2BUA）を識別するために使用される ID。
active	コールがライブであるか履歴コールであるかに関する詳細。
initial_call	コールが複数コンポーネントである場合（B2BUA ホップが含まれている場合）、B2BUA コールに関連付けるために内部で使用されます。
licensed	このフィールドには、コールでライセンスが使用されたかどうかが表示されます。
licensed_as_traversal	このフィールドには、コールでトラバーサルライセンスが使用されたかどうかが表示されます。
status	200 OK メッセージはコールが正常に行われたことを通知します。コールが失敗した場合は、このフィールドにエラーメッセージが表示されます。
tag	コール ID。
box_call_serial_number	複数のコールを（たとえば、B2BUA を通じて）まとめるために追加される ID です。
start_time	このフィールドには、コールの日付と時刻が表示されます。タイムゾーンは、[System] > [Times] > [Time Zone] で設定でき、日付形式は YYYY-MM-DD です。
end_time	このフィールドには、コールの終了時刻が表示されます。
source_alias	このフィールドには、発信者のエイリアスが表示されます。
destination_alias	このフィールドには、呼び出し先のエイリアスが表示されます。
aside_destination_alias	発信者（Lync との相互運用の場合は MS Lync クライアント）のエイリアス。
bside_destination_alias	呼び出し先（または Lync 以外のクライアント）のエイリアス。
aside_request_uri	発信者（Lync との相互運用の場合は MS Lync クライアント）の要求 URI。
bside_request_uri	呼び出し先（または Lync 以外のクライアント）の要求 URI。
protocol	このフィールドは、コールが SIP <-> SIP、SIP <-> H323、H323 <-> SIP、または H323 <-> H323 であった場合に表示されます。
protocol_summary	このフィールドは上記と同じですが、コールが複数コンポーネントであったか DVO であったかなど、追加情報が表示されることがあります。
media_routed	このフィールドには、コール時にメディア（NAT/IWF/B2BUA など）が送信されたかどうかが表示されます。
audio	このフィールドには、コールが音声のみであったかどうかが表示されます。

## 参照先

フィールド	定義
traversal_license_tokens	このフィールドには、コール フォーク/ブランチがメディア（音声はトークン 1 つ、ビデオは 2 つ）を取得したかどうかが表示されます。*
non_traversal_license_tokens	このフィールドには、コール フォーク/ブランチがメディア（音声はトークン 1 つ、ビデオは 2 つ）を取得する必要がなかったかどうかが表示されます。*
disconnect_reason	このフィールドには、通常のコール ティアダウンやその他のエラーなど、コール終了の理由（最終ステータス）が表示されます。
details	このフィールドには、メディアの統計情報など、コールの詳細が表示されます。
last_updated_timestamp	上記のフィールドのいずれかが更新された最終時刻が表示されます。

\* コールがセットアップされると、これらのエントリの 1 つだけ（つまり、応答したフォーク/ブランチのエントリのみ）がゼロ以外の値になります。

## システム メトリックの参考資料

## Expressway の設定オプション

表 2 : Expressway の collectd の設定コマンド

コマンドの機能	Web UI の場所	CLI コマンドの例
メトリック収集のオンとオフの切り替え	<b>[Maintenance] &gt; [Logging] &gt; [System Metrics Collection]</b>	<code>xconfig log SystemMetrics mode: on</code>
サーバアドレスの指定	<b>[Maintenance] &gt; [Logging] &gt; [Collection server address]</b>	<code>xconfig log SystemMetrics network address: address</code>
リスニング ポートの指定	<b>[Maintenance] &gt; [Logging] &gt; [Collection server port]</b>	<code>xconfig log SystemMetrics network port: 25826</code>
収集間隔の指定	<b>[Maintenance] &gt; [Logging] &gt; [Collection Interval]</b>	<code>xconfig log SystemMetrics interval: 60</code>
システム メトリック設定の読み取り	<b>[Maintenance] &gt; [Logging]</b>	<code>xstatus SystemMetrics</code>

## Expressway から収集されるメトリック

次のハードウェア統計情報がモニタされます。

- aggregation-cpu-sum
- aggregation-cpu-average
- システム内の各コアのコア単位の CPU 使用状況
- df
- disk
- 負荷
- プロトコル - Tcp (protocols-Tcp)

## 参照先

- プロトコル - Udp (protocols-Udp)
- swap
- ユーザ
- メモリ
- Uptime
- プロセス

次のアプリケーション データが collectd のカスタム `exec-app` プラグインによってモニタされます。

- `gauge-active_alarms` は、この Expressway 上のアクティブなアラームの数です。
- `gauge-active_calls` は、この Expressway によって処理中のコールの数です。
- `gauge-<service name>` は、各システム サービスのステータスです。
- `gauge-<zone name>_ActiveCalls` は、指定したゾーン内のアクティブなコールの数です。
- `gauge-<zone name>_BandwidthAllocated` は、指定したゾーンに割り当てられている合計帯域幅を測定します。
- `gauge-<zone name>_BandwidthLimit`

これらのメトリックのそれぞれが自由形式のデータを許可する collectd GAUGE データ ソース タイプを使用します。たとえば、収集サーバでは、`collectdHostnamecollectd.exec-app.gauge-active_calls` など、完全な collectd 値の名前が表示されます。

ユーザによる設定が可能なゾーン名の場合は、[collectd メトリックの命名スキーマ](#)と一致しないことがあります。収集サーバがスキーマを適用している場合、一部のゾーンからのメトリックが受け付けられない可能性があります。

## 収集サーバに送信されるデータ

ネットワーク プラグインは [collectd バイナリ プロトコル](#)を使用して、モニタ対象のハードウェア リソースやソフトウェア プロセスを表す数字、文字列、値データをカプセル化します。

ネットワーク プラグインは、デフォルトで UDP 25826 を使用して、メトリックのデータ パケットを一定の間隔で分析サーバにプッシュします。分析サーバはデータを解析し、人間が判読できる形式で表示します。

分析サーバが collectd ネットワーク プラグインと csv プラグインを使用している場合は、小さい CSV ファイルとしてメトリックを保存し、メトリックの名前とタイムスタンプを使用してファイル名 (`gauge-H323-2015-05-21` など) を作成します。

## Expressway に実装される collectd プラグイン

表 3 : Expressway アプリケーションに実装されている collectd プラグイン

プラグイン名	説明/詳細情報
[集約]	CPU 値をカウンタの <code>aggregation_cpu_sum</code> と <code>aggregation_cpu_average</code> に集約します。
CPU	プロセッサ情報。raw 情報が <code>aggregation_cpu_average</code> と <code>aggregation_cpu_sum</code> に集約されます。
DF	ファイル システム情報。 <a href="#">collectd Wiki の DF の説明</a> を参照してください。
ディスク	ハード ディスクのパフォーマンス。 <a href="#">collectd Wiki のディスクの説明</a> を参照してください。

## 参照先

プラグイン名	説明/詳細情報
Exec-app	コール、アラーム、ゾーン、およびサービスに関する特定の Expressway 情報を返すカスタマイズされた <b>exec</b> バージョン
負荷	タスク キューに基づくシステム負荷
メモリ	メモリの統計情報
ネットワーク	リモート アドレスへのパブリッシュを可能にします。このプラグインは、データの 캡セル化のための <a href="#">collectd バイナリ プロトコル</a> を実装します。リモート サーバには適切な解析ツールが必要です。
プロトコル	Expressway で使用されるプロトコルの設定可能なサブセット
プロセス	<p>システム プロセスをカウントし、ステータス別（実行中、スリープ中、ゾンビなど）にグループ化します。</p> <p>また、特定のプロセスの詳細な統計情報も収集します。プラグインは次のプロセスを詳しくモニタします。</p> <ul style="list-style-type: none"> <li>• app</li> <li>• bramble</li> <li>• credentialmanagerservermain</li> <li>• cvs_main</li> <li>• erlang-beam</li> <li>• erlang-epmd</li> <li>• httpd</li> <li>• httpserver</li> <li>• ivy</li> <li>• licensemanagerservermain</li> <li>• managementconnectormain</li> <li>• managementframework</li> <li>• openssl2nss</li> <li>• policyservermain</li> <li>• syslog-ng</li> <li>• XCP</li> </ul>
Swap	ディスクに書き込まれたシステム メモリの量
Uptime	システムの稼働時間を追跡して、平均実行時間や特定期間の最大稼働時間など、カウンタを提供します。 <a href="#">collectd Wiki のアップタイムの説明</a> を参照してください。
ユーザ	現在ログインしているユーザの数

シスコの法的情報

## シスコの法的情報

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

シスコによる TCP ヘッダー圧縮の実装には、カリフォルニア大学バークレー校 (UCB) が UNIX オペレーティング システムの UCB パブリック ドメイン バージョンの一部として開発したプログラムを使用しています。

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

ハード コピーおよびソフト コピーの複製は公式版とみなされません。最新版はオンライン版を参照してください。

シスコは世界各国 200 箇所にオフィスを開設しています。各オフィスの住所、電話番号、FAX 番号は当社の Web サイト ([www.cisco.com/go/offices](http://www.cisco.com/go/offices)) をご覧ください。

© 2016 Cisco Systems, Inc. All rights reserved.

シスコの商標または登録商標

## シスコの商標または登録商標

シスコおよびシスコのロゴは、米国およびその他の国におけるシスコおよびその関連会社の商標または登録商標です。シスコの商標の一覧については、[www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks) をご覧ください。掲載されている第三者の商標はそれぞれの権利者の財産です。「パートナー」または「partner」という用語は、シスコと他社との間のパートナーシップ関係を意味するものではありません。(1110R)