Firepower システム データベース アクセス ガイド

バージョン 6.0
2016 年 6 月 1 日

Cisco Systems, Inc.
www.cisco.com
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.

【注意】シスコ製品をご使用になる前に、安全上の注意（www.cisco.com/jp/go/safety_warning/）をご確認ください。

本書は、米国シスコシステムズ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動 / 変更されている場合がありますことをご了承ください。
あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。

また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

CONTENTS

概要

Firepower システム® データベース アクセス機能により、シスコ Firepower Management Center では、JDBC SSL 接続をサポートしているサードパーティのクライアントを使用して、侵入、検出、ユーザ アクティビティ、相関、接続、脆弱性、アプリケーション、および URL 統計のデータベース テーブルを照会できます。

Crystal Reports、Actuate BIRT、JasperSoft iReport などの業界標準のレポート作成ツールを使用してクエリを作成し、送信することができます。また、独自のカスタム アプリケーションを設定してプログラム制御下にある シスコ データを照会することもできます。たとえば、侵入およびディスカバリ イベント データについて定期的にレポートしたり、アラート ダッシュボードをリフレッシュしたりするサーブレットを構築することが可能です。

1 つのクライアントで複数の Firepower Management Center に接続できますが、それぞれへのアクセスを個別に設定する必要があることに注意してください。

接続するアプライアンスを決定する際には、シスコ アプライアンス上のデータベースを照会すると、使用可能なアプライアンス リソースが減少する点に注意してください。クエリを慎重に設計し、組織の優先度に対応した適切な時点でクエリを送信する必要があります。

詳細については、次の項を参照してください。

- バージョン 6.0 でのデータベース アクセスの重要な変更（1-1 ページ）
- 前提条件（1-5 ページ）
- 最初に行う作業（1-7 ページ）

バージョン 6.0 でのデータベース アクセスの重要な変更

Firepower システム 展開をバージョン 5.4.x から バージョン 6.0 にアップグレードする場合、次に示す変更に注意してください。これらの変更の一部では、クエリを更新する必要があります。

バージョン 6.0 の新規テーブルおよび変更されたテーブル

次の表に、バージョン 6.0 のデータベース アクセス テーブルに対する変更を示します。

表 1-1　バージョン 6.0 でのテーブルの変更の概要

テーブル	変更の説明
app_ids_stats_current_timeframe（5-4 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
app_stats_current_timeframe（5-6 ページ）	次のフィールドが追加されました。 • bypass • domain_name • domain_uuid • netmap_num • would_bypass
application_info（6-7 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid
application_tag_map（6-9 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid
audit_log（3-1 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid
compliance_events_stats_current_timeframe（5-8 ページ）	このテーブルは、コンプライアンス イベントおよびホワイト リスト イベントの数に関する統計を追跡するために追加されました。
connection_log（7-1 ページ）	次のフィールドが追加されました。 • dns_ttl • dns_response • domain_name • domain_uuid • endpoint_profile • http_response_code • hostname_in_query • location_ip • security_group • sinkhole
connection_summary（7-13 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
dns_query_stats_current_timeframe（5-9 ページ）	このテーブルは、DNS クエリの統計情報を追跡するために追加されました。

表 1-1　バージョン 6.0 でのテーブルの変更の概要（続き）

テーブル	変更の説明
domain_control_information（6-11 ページ）	このテーブルは、ドメインとその親ドメインに関する情報を追跡するために追加されました。
fireamp_event（3-2 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • http_response_code
geolocation_stats_current_timeframe（5-11 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
ids_impact_stats_current_timeframe（5-12 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
intrusion_event（4-1 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
intrusion_event_packet（4-7 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
ip_reputation_stats_current_timeframe（5-14 ページ）	このテーブルは、指定されたセキュリティ インテリジェンス カテゴリの IP アドレス、URL、および DNS ドメインに対する要求に関連付けられている接続と帯域幅使用に関する統計情報を追跡するために追加されました。
network_discovery_event（6-12 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid
rna_host（6-13 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid
session_stats_current_timeframe（5-15 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid

表 1-1　バージョン 6.0 でのテーブルの変更の概要（続き）

テーブル	変更の説明
ssl_stats_current_timeframe（5-17 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
storage_stats_by_disposition_current_timeframe（5-19 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
storage_stats_by_file_type_current_timeframe（5-21 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
tag_info（6-57 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid
transmission_stats_by_file_type_current_timeframe（5-22 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
url_category_stats_current_timeframe（5-23 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
url_reputation_stats_current_timeframe（5-25 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
user_ids_stats_current_timeframe（5-26 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num
user_stats_current_timeframe（5-28 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • netmap_num

表 1-1　バージョン 6.0 でのテーブルの変更の概要（続き）

テーブル	変更の説明
user_ipaddr_history（6-60 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • endpoint_profile • location_ip • security_group
si_connection_log（7-17 ページ）	次のフィールドが追加されました。 • dns_ttl • dns_response • domain_name • domain_uuid • endpoint_profile • http_response_code • hostname_in_query • location_ip • security_group • sinkhole
user_discovery_event（8-2 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • endpoint_profile • location_ip • security_group
compliance_event（9-1 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid
file_event（10-1 ページ）	次のフィールドが追加されました。 • domain_name • domain_uuid • http_response_code • netmap_num

前提条件

データベースアクセス機能を使用する前に、以降の項で説明する前提条件を満たしている必要があります。

- ライセンス（1-6 ページ）
- Firepower システム 機能と用語（1-6 ページ）
- 通信ポート（1-6 ページ）

ライセンス

いずれかの シスコ ライセンスがインストールされている場合には、外部データベースを照会できます。ただし、一部のテーブルは、ライセンスされている機能に関連付けられています。これらのテーブルにデータが取り込まれるのは、関連付けられている機能を使用できるようにライセンスが設定されており、そのデータが生成されるように展開が適切に設定されている場合だけです。ライセンスされていない機能に関連付けられているテーブルを照会することはできません。ライセンスの詳細については、『*Firepower Management Center Configuration Guide*』の「Understanding Licensing」を参照してください。

Firepower システム 機能と用語

このマニュアルの情報を理解するには、Firepower システム の機能と名称、そのコンポーネントの機能について理解しておく必要があります。これらのコンポーネントにより生成される各種イベント データについて理解しておく必要があります。『*Firepower Management Center Configuration Guide*』では、聞き慣れない用語や製品固有の用語の定義を頻繁に確認できます。この構成ガイドには、このマニュアルで説明するフィールドのデータに関する追加情報も収録されています。

通信ポート

Firepower システム では、内外のアプライアンス間で通信を行い、ネットワーク展開内で特定の機能を有効にするために、特定のポートを使用する必要があります。

Firepower Management Center でデータベース アクセスを有効にすると、クライアントとアプライアンスの間で JDBC トラフィックを伝送する接続にポート 1500 と 2000 が使用されます。

クライアント システム

Firepower システム データベースへの接続に使用するコンピュータに、Java ソフトウェア（Java Runtime Environment（JRE）とも呼ばれます）または Java 仮想マシン（JVM）をインストールする必要があります。最新バージョンの Java を `http://java.com/` からダウンロードできます。

Firepower Management Center から、データベースへの接続に使用する JDBC ドライバ ファイルが含まれているパッケージをダウンロードして解凍する必要があります。このパッケージには、Firepower Management Center との暗号化通信用の SSL 証明書のインストールに使用する実行可能ファイルや、これらのユーティリティのその他のソース ファイルも含まれています。

また、ご使用のコンピュータで該当するシステム設定（環境変数など）を変更する方法も理解しておく必要があります。

クエリ アプリケーション

Firepower システム データベースを照会するには、商用のレポート ツール（Actuate BIRT、JasperSoft iReport、または Crystal Reports など）や、JDBC SSL 接続をサポートするその他の任意のアプリケーション（カスタム アプリケーションを含む）を使用できます。このマニュアルでは、データベースに接続するために必要な情報（JDBC URL、ドライバ JAR ファイル、ドライバ クラスなど）について説明します。ただし、JDBC SSL 接続の詳しい設定手順については、ご使用のレポート ツールのドキュメントを参照してください。

シスコ には、RunQuery と呼ばれるサンプル コマンドライン Java アプリケーションもあります。このアプリケーションを使用して、データベース接続をテストし、スキーマを表示し、基本クエリまたはアドホック クエリを手動で実行することができます。RunQuery のソース コードは、カスタム Java アプリケーションでデータベース接続を設定する際のリファレンスでもあります。RunQuery ソース コードは、Firepower Management Center からダウンロードする ZIP パッケージに含まれています。

RunQuery はサンプル クライアントであり、フル機能のレポート ツール**ではありません**。シスコでは、データベース照会の主な手段としてこのツールを使用しないことを**強く**推奨します。RunQuery の使用法については、ZIP パッケージに含まれている README ファイルを参照してください。

データベース アクセス機能が使用する JDBC 機能は次に示すものだけである点に注意してください。

- データベース メタデータ（スキーマ、バージョン、およびサポートされている機能などの情報を含みます）。
- SQL クエリの実行

データベース アクセスではその他の JDBC 機能（ストアド プロシージャ、トランザクション、バッチ コマンド、複数の結果セット、挿入/更新/削除機能など）は使用されません。

データベース クエリ

データベースを照会するには、結合条件を使用した 1 つまたは複数のテーブルに対する `SELECT` ステートメントの記述方法と実行方法を理解している必要があります。

ユーザを支援するため、このマニュアルでは、サポートされている MySQL クエリの構文、Firepower システム データベース スキーマ、許可されている結合、およびクエリに関連するその他の重要な要件と制約事項について説明します。

最初に行う作業

前提条件（1-5 ページ）で説明する前提条件を満たしていることを確認したら、最初に Firepower Management Center に接続するようにクライアント システムを設定できます。

データベース アクセスのセットアップ（2-1 ページ）では、アクセスを許可するようにアプライアンスを設定する方法、アプライアンスに接続するようにクライアント システムを設定する方法、およびアプライアンスに接続するようにレポート アプリケーションを設定する方法について説明します。また、いくつかの基本的なクエリの手順と、サポートされている MySQL 構文について説明します。

このマニュアルの残りの部分には、データベースのスキーマと結合に関する情報とサンプル クエリを収録しており、次の章で構成されています。

- スキーマ：システムレベル テーブル（3-1 ページ）では、システムレベル テーブル（監査ログやヘルス イベントなど）のスキーマと結合について説明します。
- スキーマ：侵入テーブル（4-1 ページ）では、侵入関連テーブルのスキーマと結合について説明します。
- スキーマ：統計情報追跡テーブル（5-1 ページ）では、アプリケーション、URL、およびユーザ統計情報のテーブルのスキーマと結合について説明します。
- スキーマ：検出イベントおよびネットワーク マップのテーブル（6-1 ページ）では、検出イベントとネットワーク マップ情報（ネットワーク資産に関する情報）が格納されているテーブルのスキーマと結合について説明します。
- スキーマ：接続ログ テーブル（7-1 ページ）では、接続イベントと接続サマリ イベントの情報が格納されているテーブルのスキーマと結合について説明します。
- スキーマ：ユーザ アクティビティ テーブル（8-1 ページ）では、ユーザ検出およびアイデンティティ データが格納されているテーブルのスキーマと結合について説明します。
- スキーマ：相関テーブル（9-1 ページ）では、相関関連テーブル（ホワイト リスト イベントおよび違反、修復ステータス データなど）のスキーマと結合について説明します。
- スキーマ：ファイル イベント テーブル（10-1 ページ）では、ファイル イベントが格納されているテーブルのスキーマと結合について説明します。

データベース アクセスのセットアップ

データベースへの読み取り専用アクセスを取得するには、最初にアクセスを許可するようにアプライアンスを設定する必要があります。次に、JDBC ドライバをダウンロードし、アクセスするアプライアンスからの SSL 証明書を受け入れることで、アプライアンスに接続するクライアント システムを設定する必要があります。最後に、アプライアンスに接続するようにレポート アプリケーションを設定します。

注 データベース アクセスをセットアップする前に、前提条件（1-5 ページ）で説明されている前提条件を満たす必要があります。

詳細については、次の項を参照してください。

- データベース ユーザ アカウントの作成（2-1 ページ）
- Firepower Management Center でのデータベースアクセスの有効化（2-2 ページ）
- JDBC ドライバのダウンロード（2-3 ページ）
- クライアント SSL 証明書のインストール（2-4 ページ）
- サードパーティ アプリケーションを使用したデータベースへの接続（2-5 ページ）
- カスタム プログラムを使用したデータベースへの接続（2-7 ページ）
- データベースの照会（2-9 ページ）
- クエリのトラブルシューティング（2-15 ページ）
- サンプル クエリ（2-16 ページ）

データベース ユーザ アカウントの作成

ライセンス: すべて

Firepower システム データベースへのアクセスを設定するには、最初にユーザ アカウントを作成し、Firepower システム データベースにアクセスするための権限をそのアカウントに割り当てる必要があります。この権限を付与するには、システム提供の External Database User ユーザ ロール、または組織が作成し、External Database User 権限が含まれているカスタム ユーザ ロールを、このアカウントに割り当てます。ユーザ アカウントの作成と、特定のユーザ ロールの権限の確認については、『*Firepower Management Center Configuration Guide*』を参照してください。

警告　**External Database Access はグローバル権限です。External Database Access が付与されているユーザは、すべてのドメインに対して情報を照会できます。**

ヒント　システム提供の Administrator ロールが割り当てられているユーザには、デフォルトで External Database User 権限が付与されています。

マルチドメイン型展開では、Admin アクセス権限があるドメインでユーザ アカウントを作成できます。ただし、External Database User ロールはグローバル ドメイン レベルでのみ使用可能です。External Database User は、ドメインに関係なくすべてのイベントにアクセスできます。

Firepower Management Center でのデータベースアクセスの有効化

ライセンス: すべて

外部データベース ユーザの作成後に、アプライアンスのデータベースへのアクセスを許可するように Firepower Management Center を設定する必要があります。また、アプライアンスでデータベース アクセス リストを設定し、外部データベースを照会するすべてのホスト IP アドレスを追加する必要があります。

データベース アクセスを有効にするには、次の手順を実行します。

アクセス権限: Admin

ステップ 1　Firepower Management Center で [System] > [Configuration] を選択します。

ステップ 2　左側の [External Database Access] をクリックします。

[Database Settings] メニューが表示されます。

ステップ 3　[Allow External Database Access] チェックボックスをオンにします。

[Access List] フィールドが表示されます。

ステップ 4　サードパーティ アプリケーションの要件に応じて、[Server Hostname] フィールドにFirepower Management Centerの完全修飾ドメイン名（FQDN）または IPv4 アドレスを入力します。証明書をインストールするときに IPv6 アドレスを使用できないため、IPv6 アドレスは使用できません。

FQDN を入力する場合は、クライアントが Firepower Management Center の FQDN を解決できることを確認する必要があります。IP アドレスを入力する場合は、クライアントがその IP アドレスを使用して Firepower Management Center に接続できることを確認する必要があります。

ステップ 5　1 つ以上の IP アドレスからのデータベース アクセスを追加するため、[Add Hosts] をクリックします。

[Access List] フィールドに [IP Address] フィールドが表示されます。

ステップ 6　[IP Address] フィールドでは、追加する IP アドレスに応じて次のいずれかを入力できます。

- 正確な IPv4 アドレス（`192.168.1.101` など）
- 正確な IPv6 アドレス（`2001:DB8::4` など）
- IP アドレス範囲。

  - Firepower システムで IP アドレス範囲を使用する方法については、『*Firepower Management Center Configuration Guide*』の IP アドレス規則を参照してください。
- 任意の IP アドレスを示す `any`

ステップ 7　[Add] をクリックします。

IP アドレスがデータベース アクセス リストに追加されます。

ステップ 8　必要に応じてデータベース アクセス リストのエントリを削除するには、削除アイコン（ ）をクリックします。

ステップ 9　[Save] をクリックします。

データベース アクセス設定が保存されます。

ステップ 10　次の項（JDBC ドライバのダウンロード）の手順に進みます。

JDBC ドライバのダウンロード

ライセンス: すべて

外部データベース ユーザを作成し、Firepower Management Center でデータベース アクセスを許可するように設定したら、JDBC ドライバをクライアント システムにダウンロードします。データベースに接続するときに、この JDBC ドライバを使用する必要があります。

JDBC ドライバをダウンロードするには、次の手順を実行します。

アクセス権限: Admin

ステップ 1　Firepower Management Center で [System] > [Configuration] を選択します。

ステップ 2　左側の [External Database Access] をクリックします。

[Database Settings] メニューが表示されます。

ステップ 3　[Client JDBC Driver] の横にある [Download] をクリックし、ブラウザのプロンプトに従って `client.zip` パッケージをダウンロードします。

ステップ 4　ZIP パッケージを解凍します。場所を書きとめておきます。
パッケージのファイル構造を保持してください。
ドライバはその他のファイルと共に ZIP ファイル（`client.zip`）に含まれています。パッケージのディレクトリ構造は次のとおりです。

- `bin`: RunQuery と呼ばれるサンプル クライアントと、クライアントと Firepower Management Center の間での暗号化通信用証明書をインストールするときに使用する実行ファイルが含まれています。
- `lib`: JDBC ドライバ JAR ファイルが含まれています。
- `src:bin` ディレクトリ内の実行ファイルのソース コードが含まれています。

ステップ 5　次の項（クライアント SSL 証明書のインストール）の手順に進みます。

クライアント SSL 証明書のインストール

JDBC ドライバのダウンロード後に、システム提供プログラム InstallCert を使用して Firepower Management Center からの SSL 証明書を受け入れ、インストールします。クライアント システムと Firepower Management Center は SSL 証明書認証を使用して安全に通信します。証明書を受け入れる場合、現在実行されている JRE の **security** ディレクトリ内のキーストア（**jssecacerts**）に証明書が追加されます。

```
$JAVA_HOME/jre[version]/lib/security
```

次に、Microsoft Windows と UNIX を実行しているコンピュータでのキーストアの一般的な場所を示します。

- `C:\Program Files\Java\jre[version]\lib\security\jssecacerts`
- `/var/jre[version]/lib/security/jssecacerts`

注　データベース アクセス機能にアクセスするために使用する予定の Java クエリ アプリケーションが異なる JRE を使用している場合は、その別の JRE の **security** ディレクトリにキーストアをコピーする必要があります。

InstallCert を使用して SSL をインストールするには、次の手順を実行します。

ステップ 1　コンピュータでコマンドライン インターフェイスを開きます。

ステップ 2　コマンド プロンプトから、ZIP パッケージの解凍時に作成された **bin** にディレクトリを変更します。

ステップ 3　Firepower Management Center の SSL 証明書をインストールするには、次のように入力して Enter キーを押します。

```
java InstallCert defense_center
```

defense_center は Firepower Management Center の FQDN または IP アドレスです。InstallCert は IPv6 アドレスをサポートしていません。IPv6 ネットワークでは、解決可能なホスト名を使用する必要があります。

Microsoft Windows を実行しているコンピュータでは次の例に示すような出力が表示されます。

```
Loading KeyStore C:\Program Files\Java\jre6\lib\security...
Opening connection to defensecenter.example.com:2000...
Starting SSL handshake...
Subject GENERATION=server, T=vjdbc, O="シスコ, Inc.",
...
...
```

証明書を表示するように求められます。

ステップ 4　オプションで、証明書を表示します。

証明書を受け入れるように求められます。

ステップ 5　証明書を受け入れます。

証明書が受け入れられると、Microsoft Windows を実行しているコンピュータから次の例のような出力が表示されます。

```
Added certificate to keystore 'C:\Program Files\Java\jre6\lib\security\jssecacerts'
using alias 'defensecenter.example.com-1'
```

Crystal Reports を使用する予定の場合は、キーストア（`jssecacerts`）の場所を書きとめておきます。この情報は、後で必要になります。

ステップ 6　次の選択肢があります。

- サードパーティ アプリケーションを使用する場合は、次の項（サードパーティ アプリケーションを使用したデータベースへの接続（2-5 ページ））の手順に進みます。
- カスタム アプリケーションを使用する場合は、カスタム プログラムを使用したデータベースへの接続（2-7 ページ） の手順に進みます。

サードパーティ アプリケーションを使用したデータベースへの接続

証明書のインストール後、JDBC SSL 接続をサポートするサードパーティ クライアントを使用して、Firepower Management Center でデータベースを照会できます。次の表に、クライアントと Firepower Management Center 間の接続を設定するために必要な情報を示します。

表 2-1　データベース アクセス クライアントの接続情報

情報	説明
JDBC URL	次に示す JDBC URL によって Firepower システム データベースが識別されるため、クライアントの JDBC ドライバがそのデータベースとの接続を確立できます。 `jdbc:vjdbc:rmi://defense_center:2000/VJdbc,eqe` *defense_center* は Firepower Management Center の FQDN または IP アドレスです。
JDBC ドライバ JAR ファイル	Firepower システム データベースへの接続を設定するときには次に示す JAR ファイルを使用する必要があります。 • `vjdbc.jar` • `commons-logging-1.1.jar` これらのファイルは、JDBC ドライバのダウンロード（2-3 ページ）の説明に従い **`client.zip`** ファイルをダウンロードして解凍した **`lib`** サブディレクトリにあります。
JDBC ドライバ クラス	Firepower システム データベースへの接続を設定するときには次に示すドライバ クラスを使用する必要があります。 `com.sourcefire.vjdbc.VirtualDriver`
ユーザ名とパスワード	アプライアンスのデータベースに接続するには、External Database User 権限が付与されているユーザ アカウントを使用します。詳細については、データベース ユーザ アカウントの作成（2-1 ページ）を参照してください。

以降の項では、広く使用されている 3 つの業界標準レポート ツールを使用して Firepower システム データベースに接続する際のヒントを説明します。これらのツールまたは他の Java ベース アプリケーションのいずれを使用するかに関係なく、ご使用のレポート ツールのドキュメントで、JDBC SSL 接続の詳しい作成手順を参照してください。

Crystal Reports

以下の情報は 32 ビット Windows 環境への Crystal Reports 2011 のインストールに適用されます。64 ビット Windows 環境を実行している場合はファイルパスが異なる可能性があります。

Crystal Reports 2011 が Firepower システム データベースに接続できるようにするには、次の操作が必要です。

- Firepower Management Center からダウンロードした JDBC ドライバ JAR ファイルを Crystal Reports クラスパスに追加します。Crystal Reports のデフォルト インストールの場合、次に示すファイルでクラスパス セクションを編集できます。
  ```
  C:\Program Files\SAP BusinessObjects\SAP Business Objects
  Enterprise XI 4.0\Java\CRConfig.xml
  ```
- クライアント SSL 証明書のインストール時に作成したキーストアを、適切な Crystal Reports セキュリティ ディレクトリにコピーします。Crystal Reports のデフォルト インストールの場合、ディレクトリは次のとおりです。
  ```
  C:\Program Files\SAP BusinessObjects\SAP Business Objects
  Enterprise XI 4.0\win32_x86\jdk\jre\lib\security
  ```
- **シスコ** をデータベース名として使用し、Database Expert との新しい JDBC（JNDI）の接続を作成します。

JasperSoft iReport

iReport が Firepower システム データベースに接続できるようにするには、次の操作が必要です。

- Firepower Management Center からダウンロードした JDBC ドライバ JAR ファイルを iReport クラスパスに追加します。
- Firepower Management Center からダウンロードした JDBC ドライバ JAR ファイルを使用して 新しい JDBC ドライバを追加します。ドライバ ファイルの追加後は、iReport は正しいドライバ クラスを検出するはずです。
- 作成したドライバを使用して、新しいデータベース接続を作成します。

Actuate BIRT

BIRT が Firepower システム データベースに接続できるようにするには、次の操作が必要です。

- [Generic JDBC Driver] テンプレートを使用してドライバ定義を追加します。
- [Generic JDBC] プロファイル タイプを使用して新しいデータベース接続を作成します。
- [JDBC Data Source] データ ソース タイプを使用してレポートのデータ ソースを作成します。

ヒント　新しい JDBC データ ソース プロファイルの作成時に シスコ ドライバ クラスを選択できない場合は、Firepower Management Center からダウンロードした JDBC ドライバ JAR ファイルを使用してドライバを追加します。

カスタム プログラムを使用したデータベースへの接続

証明書をインストールしたら、カスタム Java レポート ツールが Firepower システム データベースを照会できるように設定できます。シスコ には、RunQuery という名前のサンプル Java コマンド ライン アプリケーションがあります。このアプリケーションは、Firepower Management Center に含まれている JDBC ドライバを使用して必要な SSL 接続を確立します。RunQuery は、テーブル レコードとテーブル メタデータの両方を取得します。ソース コードは、Firepower Management Center からダウンロードした ZIP パッケージの `src` ディレクトリに含まれています。JDBC ドライバのダウンロード (2-3 ページ) を参照してください。

注　RunQuery はサンプル クライアントであり、フル機能のレポート ツール**ではありません**。シスコでは、データベース照会の主な手段としてこのツールを使用しないことを**強く**推奨します。RunQuery の使用法については、ZIP パッケージに含まれている README ファイルを参照してください。

カスタム プログラムを使用したデータベースへの接続に関する詳細については、次の項を参照してください。

- カスタム Java プログラムのサンプル コード (2-7 ページ)：RunQuery アプリケーションがデータベース接続のセットアップとクエリの送信に使用する Java クラスとメソッドについて説明します。
- アプリケーションの実行 (2-8 ページ)：Java アプリケーションを実行するための環境要件について説明します。

カスタム Java プログラムのサンプル コード

RunQuery ソース コードでは、後述する関数が使用されます。次のコード例に、可能な実装方法の 1 つを示します。

SSL プロバイダー接続の動的な設定

クライアントに SSL セキュリティ証明書をインストールしたら (クライアント SSL 証明書のインストール (2-4 ページ) を参照)、プログラムで次の行を使用して JSSE プロバイダーを動的に登録できます。

```
Security.addProvider(new com.sun.net.ssl.internal.ssl.(Provider());
```

プログラムの JDBC ドライバの初期設定

`Class.forName()` メソッドを使用して Java アプリケーションに JDBC ドライバ クラスを次のようにロードできます。

```
Class.forName("com.sourcefire.vjdbc.VirtualDriver").newInstance();
```

コマンド ラインから起動するプログラムの場合は、ユーザが次のように JDBC クラスを指定できます。

```
java -Djdbc.drivers="com.sourcefire.vjdbc.VirtualDriver" program_name ...
```

program_name はプログラムの名前です。

データベースへのプログラムの接続

プログラムがクエリを送信する前に、プログラムで JDBC 接続オブジェクトを取得する必要があります。接続を確立して接続オブジェクトを取得するには、次のように DriverManager.getConnection メソッドを使用します。

```
Connection conn = DriverManager.getConnection("jdbc:vjdbc:rmi://my_dc:2000/VJdbc,eqe",
"user", "password");
```

my_dc は Firepower Management Center の FQDN または IP アドレス、user はデータベース アクセス ユーザ アカウント名、*password* はアカウント パスワードです。

シスコ テーブルのデータの照会

クエリを送信し、取得したレコードを結果セットに割り当てるため、SQL クエリ オブジェクトを次のように作成します。

```
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("sql");
```

sql は SQL クエリです。サポートされている SQL 関数については、データベースの照会（2-9 ページ）を参照してください。

テーブル クエリの結果の作成

上記のクエリにより生成される結果セット（rs）から、次のようにフィールドを出力できます。

```
while(rs.next())
{
    for(int i=1; i<= md.getColumnCount(); i++)
    {
        System.out.print(rs.getString(i) + " ");
    }
    System.out.print("\n");
}
```

スキーマ情報の取得

プログラムは、データベースのテーブルを次のようにリストできます。

```
DatabaseMetaData metaData = conn.getMetaData();
      ResultSet tables = meta.getTables(null, null, null, null);
      while (tables.next())
      {
System.out.println(tables.getString("TABLE_NAME"));
      }
```

プログラムは、テーブルの列を次のようにリストできます。

```
ResultSet columns = metaData.getColumns(null, null, "table_name", null);
```

table_name は、データベース テーブルの名前です。

アプリケーションの実行

アプリケーションを実行する前に、クライアント コンピュータの CLASSPATH を設定し、アプリケーションの JAR ファイルの場所と現在のディレクトリを組み込む必要があります。

JDBC ドライバのダウンロード（2-3 ページ）の説明に従ってデータベース アクセスの ZIP パッケージをダウンロードして解凍した場合は、CLASSPATH を次のように更新します。

UNIX 環境でアプリケーションを実行するには:

ステップ 1　次のコマンドを使用します。

```
export CLASSPATH=$CLASSPATH:.:path/lib/vjdbc.jar:path/lib/commons-logging-1.1.jar
```

path は、Firepower Management Center からダウンロードした ZIP パッケージを解凍したディレクトリ パスです。

Windows 7 環境でアプリケーションを実行するには:

ステップ 1　[Computer] アイコンを右クリックして [Properties] を選択します。

[System] ウィンドウが表示されます。

ステップ 2　[Advanced System Settings] をクリックします。

[System Properties] ウィンドウが表示されます。

ステップ 3　[Advanced] タブを選択します。

ステップ 4　[Environment Variables...] をクリックします。

[Environment Variables] ウィンドウが表示されます。

ステップ 5　**CLASSPATH** システム変数を選択して [Edit...] をクリックします。

[Edit System Variable] ウィンドウが表示されます。

ステップ 6　[Variable value:] フィールドに次を追加します。

```
.;path\bin;.;path\lib\vjdbc.jar;.;path\lib\commons-logging-1.1.jar;.;path\lib
```

path は、Firepower Management Center からダウンロードした ZIP パッケージを解凍したディレクトリ パスです。

ステップ 7　[OK] をクリックして、値を保存します。

[Environment Variables] ウィンドウが表示されます。

ステップ 8　[OK] をクリックして、値を保存します。これでアプリケーションを実行できます。

データベースの照会

External Database Access はグローバル権限です。クエリで制限していない限り、結果はデータベース内のすべてのドメインを対象としています。

以降の項では、サポートされているクエリの構文と、その他の重要なクエリ関連の要件および制約について説明します。

- サポートされている SHOW ステートメント構文（2-10 ページ）では、Firepower システム データベース照会のためにサポートされている MySQL `SHOW` ステートメント構文について説明します。
- サポートされている DESCRIBE または DESC ステートメント構文（2-10 ページ）では、Firepower システム データベース照会のためにサポートされている MySQL `DESCRIBE` ステートメント構文について説明します。

- サポートされている SELECT ステートメント構文（2-11 ページ）では、Firepower システム データベース照会のためにサポートされている MySQL **SELECT** ステートメント構文について説明します。
- 結合の制約（2-12 ページ）では、Firepower システム データベース照会のためにサポートされている結合と、任意のテーブルに対して許可されている結合に関する情報の確認方法を説明します。
- 使い慣れていない形式で保存されているデータの照会（2-13 ページ）では、使い慣れていない形式で保存されているデータ（UNIX タイムスタンプおよび IP アドレスを含む）に対してクエリを実行し、クエリが正常に完了し、予期しているとおりの結果が表示されるようにする方法について説明します。
- パフォーマンス上の理由によるクエリの制限（2-15 ページ）では、クエリに対する制約を適用する場合に、Firepower システム のパフォーマンスを低下させないための推奨事項について説明します。
- クエリのヒント（2-15 ページ）では、さまざまなアプライアンスで侵入イベントを照会する場合のヒントについて説明します。

スキーマの情報と許可されている結合については、以降の章を参照してください。

- スキーマ：システムレベル テーブル（3-1 ページ）
- スキーマ：侵入テーブル（4-1 ページ）
- スキーマ：統計情報追跡テーブル（5-1 ページ）
- スキーマ：検出イベントおよびネットワーク マップのテーブル（6-1 ページ）
- スキーマ：接続ログ テーブル（7-1 ページ）
- スキーマ：ユーザ アクティビティ テーブル（8-1 ページ）
- スキーマ：相関テーブル（9-1 ページ）

サポートされている SHOW ステートメント構文

SHOW ステートメントは、Firepower システム データベースのすべてのテーブルをリストします。Firepower システム データベースの照会に使用できる、サポートされている MySQL **SHOW** ステートメント構文を次に示します。

```
SHOW TABLES;
```

上記にない **SHOW** ステートメント構文はサポートされていません。

サポートされている DESCRIBE または DESC ステートメント構文

Firepower システム データベースでは、**DESCRIBE** ステートメントを限定的に使用できます。Firepower システム データベースでは、DESCRIBE ステートメントの出力には列の名前と各列のデータのタイプだけがリストされます。Firepower システム データベースの照会に使用できる、サポートされている MySQL **DESCRIBE** ステートメント構文を次に示します。

```
DESCRIBE table_name;
```

Firepower システム データベースでは、これと同一のコマンド DESC もサポートされています。

```
DESC table_name;
```

表 2-2　サポートされている DESCRIBE ステートメントの構文

項目	説明
table_name	照会するテーブルの名前

上記にない **DESCRIBE** ステートメント構文はサポートされていません。特に、Firepower システム データベース アクセス機能では次の要素はサポートされていません。

- INDEX FOR 句
- TABLE 句
- PROCEDURE 句

サポートされている SELECT ステートメント構文

Firepower システム データベースの照会に使用できる、サポートされている MySQL SELECT ステートメント構文を次に示します。

```
SELECT
[ALL | DISTINCT]
[COUNT ( field ) | COUNT (*) ]

select_expr [,select_expr ...]
FROM table_references
[WHERE where_condition]
[GROUP BY { column_name | position } [ ASC | DESC ], …]
[HAVING where_condition]
[ORDER BY { column_name | position } [ ASC | DESC ], …]
[LIMIT { [offset,] row_count | row_count OFFSET offset}]
```

次の表に、上記の SELECT ステートメントの句と引数の必須構文について詳しく説明します。

表 2-3　サポートされている SELECT ステートメント構文

項目	説明
select_expr	{*column_name* [[AS] *alias*] | *function*(...)[[AS] *alias*] | *aggregate_function*(...)[[AS] *alias*]}
column_name	照会するフィールドの名前
機能	{ABS | CAST | CEILING | CHAR_LENGTH | COALESCE | CONV | CHARACTER_LENGTH | CONCAT | CONVERT | COUNT | CURRENT_DATE | CURRENT_TIME | CURRENT_TIMESTAMP | EXTRACT | FLOOR | HEX | INET_ATON | INET_NTOA | INET6_ATON | INET6_NTOA | LEFT | LOWER | LPAD | MID | MOD | NULLIF | OCTET_LENGTH | POSITION | RIGHT | ROUND | SUBSTRING | SYSDATE | TIME | TIMESTAMP | TRIM | UPPER}
aggregate_function	{AVG | COUNT | COUNT(DISTINCT) | MAX | MIN | SUM}
field	関数の実行対象フィールドの名前

表 2-3　サポートされている SELECT ステートメント構文（続き）

項目	説明
table_references	次のいずれかを入力します。 • *table_reference* INNER JOIN *table_reference join_condition* • *table_reference* LEFT [OUTER] JOIN *table_reference join_condition*
table_reference	*table_name* [[AS] alias]
table_name	照会するテーブルの名前
join_condition	ON *conditional_expr*
conditional_expr	結合に対応したフィールド値の等価比較。詳細については、結合の制約（2-12 ページ）を参照してください。
where_condition	次のいずれかを入力します。 • IS NULL or IS NOT NULL • NOT, ! • BETWEEN ...AND ... • LIKE • =, !=, <>, >, >=, <, <=

サポートされている MySQL 構文の記述方法に慣れていない場合は、次の表でヒントを参照してください。

表 2-4　MySQL 構文の形式

記号	表記	意味
角カッコ	[]	オプションの句または引数
波カッコ	{}	必要な句または引数
パイプ	|	句または引数の選択

上記にない SELECT ステートメント構文はサポートされていません。特に、Firepower システム データベース アクセス機能では次の要素はサポートされていません。

- SELECT *：フィールドを明示的に指定する必要があります。
- ユニオン
- サブクエリ
- GROUP BY 句の WITH ROLLUP 修飾子
- INTO 句
- FOR UPDATE 句

結合の制約

パフォーマンスおよびその他の実際的な理由から、Firepower システム データベースのテーブルに対して実行できる結合は限定されています。シスコ では、その結果がイベント分析に役立つ可能性のない結合は実行できません。

内部結合または左（外部）結合だけを実行できます。入れ子になった結合、クロス結合、自然結合、右（外部）結合、フル（外部）結合、および USING 句を使用した結合はサポートされていません。

スキーマに関するドキュメントでは、各テーブルでサポートされている結合が示されています。リストされていない結合はサポートされていません。たとえば、**compliance_event** テーブルと **intrusion_event** テーブルを IP アドレス フィールドで結合することはできません。これは、両方のテーブルに IP アドレス情報が含まれている場合でも同様です。また、廃止されたテーブルと廃止されたフィールドでの結合はリストされていません。

使い慣れていない形式で保存されているデータの照会

Firepower システム データベースでは、表示に適していない形式で一部のデータが保存されています。以降の項では、さまざまなフィールドに対してクエリを実行し、クエリが正常に完了し、予期しているとおりの結果が表示されるようにする方法について詳しく説明します。

IPv6 形式のアドレス

Firepower システム データベースには、IPv6 アドレスがバイナリ形式で保存されます。結果を 16 進数形式で表示するには、**HEX()** 関数を使用します。特定の IPv6 アドレスのデータベースを照会するには、**UNHEX()** 関数を使用します。

たとえば、モニタ対象セッションに関する情報を含む **connection_log** テーブルを照会し、その照会を特定の IPv6 アドレスで制限するステートメントを次に示します。

```
SELECT HEX(initiator_ip), HEX(responder_ip), packets_sent, bytes_sent

FROM connection_log

WHERE initiator_ip = UNHEX('20010db80000000000000000000004321');
```

IPv4 アドレス

Firepower システム データベースでは、IPv6 アドレスと同じフィールドに、IPv4 アドレスがバイナリ形式で保存されます。IPv6 アドレスと同様に、16 進数形式で表示する場合は HEX() 関数を使用します。データベースは RFC に準拠するためにビット 80 ～ 95 に 1 を取り込みますが、これによって無効な IPv6 アドレスが生成されます。たとえば IPv4 アドレス 10.5.15.1 は 00000000000000000000FFFF0A050F01 として保存されます。

MAC アドレス

Firepower システム データベースには、MAC アドレスがバイナリ形式で保存されます。結果を 16 進数形式で表示するには、HEX() 関数を使用します。

たとえば、次のステートメントは **rna_host_mac_map** テーブルを照会します。このテーブルには、IP アドレスでは識別されなかったホストと その MAC アドレスに関する情報が含まれており、照会で取得できるホストが最初の 5 つのホストに限定されます。

```
SELECT HEX(host_id), HEX(mac_address)
FROM rna_host_mac_map
LIMIT 5;
```

パケット データ

Firepower システム データベースは、侵入イベントのパケット データをバイナリ形式で保存します。結果を 16 進数形式で表示するには、HEX() 関数を使用します。

たとえば、次のステートメントは **intrusion_event_packet** テーブルを照会し、特定イベントのパケット データを取得します。

```
SELECT HEX(packet_data)
FROM intrusion_event_packet
WHERE event_id = 1234;
```

UNIX タイムスタンプ

Firepower システム データベースではほとんどのタイムスタンプが UNIX タイムスタンプとして保存されます。UNIX タイムスタンプは、1970 年 1 月 1 日 00:00:00（UTC）以降の経過秒数を表します。現地時間で結果を表示するには、**FROM_UNIXTIME()** 関数を使用します。

たとえば、次のステートメントは **audit_log** テーブルを照会し、最大 25 件の結果を返します。このテーブルには、アプライアンスの Web インターフェイスでのユーザ アクションがすべて記録されています。

```
SELECT FROM_UNIXTIME(action_time_sec), user, message
FROM audit_log
LIMIT 0, 25;
```

データベースのすべての時刻は UTC の時刻であることに注意してください。**CONVERT_TZ()** 関数を使用できますが、この関数は結果を UTC で返します。

一部のイベントには、マイクロ秒の精度が関連付けられている点に注意してください。UNIX タイムスタンプとマイクロ秒の増分を連結するには、**CONCAT()** 関数と **LPAD()** 関数を使用します。たとえば、次のステートメントは **intrusion_event** テーブルを照会します。

```
SELECT CONCAT(FROM_UNIXTIME(event_time_sec)), '.', LPAD (event_time_usec, 6, '0')),
HEX(host_id),
rule_message
FROM intrusion_event
LIMIT 0, 25;
```

データベースに対し、特定の UNIX タイムスタンプのイベントを照会するには、**UNIX_TIMESTAMP()** 関数を使用します。

パフォーマンス上の理由によるクエリの制限

Firepower システム データベース テーブルに対して実行できる結合はシステムにより制限されていますが、一部のコストがかかるクエリ（Firepower Management Center のパフォーマンスに悪影響を及ぼす可能性のあるクエリ）が許可されます。

したがって、大規模なテーブルの場合は結果セットを制限するようにします。戦略としては次のものがあります。

- 特定のリーフ ドメインにクエリを制限する
- 特定の時間範囲にクエリを制限する
- IP アドレスによりクエリを制限する
- **LIMIT** 句を使用する

展開によっては、多数のテーブルを照会する際に結果セットを制限する必要があることがあります。特に次のテーブルには、DC3000 での最大 1 億件のイベントを格納できます。

- **fireamp_event**
- **intrusion_event**
- **intrusion_event_packet**
- **connection_log**（5.0 より古いバージョンでの名前：**rna_flow**）
- **connection_summary**（5.0 より古いバージョンでの名前：**rna_flow_summary**）

モニタ対象ネットワークでシステムが検出したホストの数によっては、ネットワーク マップ テーブルに対するクエリはコストが高くなることがあります。

クエリのヒント

以降の項では、検出エンジンまたは侵入イベントを含むクエリを作成する場合に、固有の結果が得られるようにするためのヒントを示します。

デバイス名

デバイス名は複数の Firepower Management Center 間で必ずしも一意である必要はありません。一意であるようにするには、特定のデバイス UUID をクエリに含めます。

侵入イベント

複数の管理対象デバイス上の侵入イベントに一意に一致するためには、**intrusion_event** テーブルの次のフィールドをクエリに指定できます。

- intrusion_event.event_id
- intrusion_event.event_time_sec
- intrusion_event.sensor_uuid

クエリのトラブルシューティング

1 つのクライアントへのアクセスを許可するように複数の Firepower Management Center を設定できますが、各システムを個別に設定する必要があります。各システムで使用可能な情報は複数の要因によって決まります。照会するデータがない場合、クエリから予期している結果が返されません。

以下に、クエリから結果が返されない原因のいくつかを示します。

- クエリの条件が細かすぎます。たとえば、クエリの時間範囲または IP アドレス範囲を調整する必要があります。
- イベント発生の原因となったネットワーク トラフィックによっては、イベントの一部のフィールドにデータが取り込まれないことがあります。たとえば、すべての接続イベントにペイロード情報が含まれているわけではありません。
- 照会するイベント タイプのログ記録を設定していませんでした。
- イベント保存を無効にしていました。
- ユーザがアクセスできないドメインを照会しようとしています。

イベントの生成方法とログ記録方法の詳細については、『*Firepower Management Center Configuration Guide*』を参照してください。

サンプル クエリ

以降の項に収録されているサンプル クエリでは、データベース アクセス機能の使用方法を示します。

- ユーザの監査レコード (2-16 ページ)
- プライオリティおよび分類別の侵入イベント (2-17 ページ)
- 侵入イベントおよびその関連ポリシー (2-17 ページ)
- 検出されたホストのリスト (2-17 ページ)
- 検出されたサーバのリスト (2-17 ページ)
- ネットワークのサーバの脆弱性 (2-18 ページ)
- オペレーティング システムの概要 (2-18 ページ)
- ホストのオペレーティング システムの脆弱性 (2-18 ページ)
- ホストの違反カウント (2-19 ページ)

注意　展開によっては、一部のサンプル クエリの実行には非常にコストがかかることがあります。詳細については、「パフォーマンス上の理由によるクエリの制限 (2-15 ページ)」を参照してください。

ユーザの監査レコード

次のクエリは、特定ユーザの監査ログのレコードをすべて返します。タイムスタンプは UTC で表示されます。

```
SELECT FROM_UNIXTIME(action_time_sec), user, message
FROM audit_log
WHERE user = 'eventanalyst';
```

プライオリティおよび分類別の侵入イベント

次のクエリは、[Events By Priority and Classification] ワークフローの [Drilldown of Event]、[Priority]、[Classification] ビューを複製します。ユーザ設定でデフォルトの [Intrusion Events] ワークフローを変更していない場合、これが Firepower Management Center Web インターフェイスで [Analysis] > [Intrusion Events] を選択すると最初に表示されるページです。

```
SELECT rule_message, priority, rule_classification, count(*) as Count
FROM intrusion_event
WHERE reviewed="0" GROUP BY rule_message, priority, rule_classification
ORDER BY Count
DESCLIMIT 0, 25;
```

侵入イベントおよびその関連ポリシー

次のクエリは、指定した週の侵入イベントをリストします。イベントごとに、関連する侵害ポリシー違反とルール分類が表示されます。

```
SELECT FROM_UNIXTIME(event_time_sec) AS event_time, event_id AS intrusion_event,
intrusion_event_policy_name AS policy, rule_classification AS classification
FROM intrusion_event
WHERE event_time_sec BETWEEN UNIX_TIMESTAMP('2011-10-01 00:00:00') AND
UNIX_TIMESTAMP('2011-10-07 23:59:59')
ORDER BY policy ASC;
```

検出されたホストのリスト

次のクエリは、ネットワークで検出されたすべての MAC ホスト（IP アドレスのないホスト）のホスト ネットワーク マップでの基本情報と、各 NIC のハードウェア ベンダーを返します。

```
SELECT HEX(mac_address), mac_vendor, host_type, FROM_UNIXTIME(last_seen_sec)
FROM rna_mac_host;
```

次のクエリは、IP アドレスを MAC アドレスにマップします。

```
SELECT HEX(ipaddr), HEX(mac_address), HEX(host_id)
FROM rna_host_ip_map LEFT JOIN rna_host_mac_map on
rna_host_ip_map.host_id=rna_host_mac_map.host_id;
```

検出されたサーバのリスト

次のクエリは、2 つの関連するテーブルを結合し、ネットワークで検出されたサーバを、その属性の多くとともにリストとして返します。これは、Firepower Management Center の Web インターフェイスでサーバのテーブル ビューに表示される内容に似ています。

```
SELECT FROM_UNIXTIME(s.last_used_sec), HEX(s.host_id), s.port, s.protocol, s.hits,
i.service_name, i.vendor, i.version, i.source_type, s.confidence
FROM AS s
```

```
LEFT JOIN rna_ip_host_service_info AS i ON (s.host_id = i.host_id AND s.port = i.port AND
s.protocol =

i.protocol);
```

このクエリは、データベース アクセスで必要なため、host_id、port、および protocol のセットでテーブルを左結合する点に注意してください。rna_host_service の結合（6-35 ページ）および rna_host_service_info の結合（6-40 ページ）を参照してください。

ネットワークのサーバの脆弱性

次のクエリは、2 つの脆弱性関連テーブルを結合し、特定のホストで検出された有効なサーバ関連の脆弱性と、各脆弱性がネットワーク上で悪用可能であるかどうかのリストを示します。

```
SELECT h.rna_vuln_id, v.title, v.remote

FROM rna_host_service_vulns AS h

LEFT JOIN rna_vuln AS v ON (h.rna_vuln_id = v.rna_vuln_id)

WHERE h.ip_address = INET_ATON('10.10.10.4')

AND h.invalid = 0;
```

このクエリは、rna_host_service_vulns（6-46 ページ）と rna_vuln の結合（6-56 ページ）で必要なため、**rna_vuln_id** でテーブルを左結合する点に注意してください。

オペレーティング システムの概要

次のクエリは、[Operating System Summary] ワークフローで [Summary of OS Names] ページを複製します。ユーザ設定でデフォルトのワークフローを変更していない場合、これが Firepower Management Center Web インターフェイスで [Analysis] > [Hosts] を選択し、次に [Hosts] を選択すると最初に表示されるページです。

```
SELECT vendor, product, count(*) AS total

FROM rna_host_os

GROUP BY vendor, product

ORDER BY total DESC;
```

ホストのオペレーティング システムの脆弱性

次のクエリは、2 つの脆弱性関連テーブルを結合し、特定のホストで検出された有効なオペレーティング システム関連の脆弱性と、各脆弱性がネットワーク上で悪用可能であるかどうかのリストを示します。

```
SELECT h.rna_vuln_id, v.title, v.remote

FROM rna_host_os_vulns AS h

LEFT JOIN rna_vuln AS v ON (h.rna_vuln_id = v.rna_vuln_id)

WHERE h.host_id = UNHEX('9610B6E6F1784DA4B39BEA7A210AAD68')

AND h.invalid = 0;
```

このクエリは、データベース アクセスで必要なため、**rna_vuln_id** でテーブルを左結合する点に注意してください。rna_host_os_vulns（6-30 ページ）および rna_vuln の結合（6-56 ページ）を参照してください。

ホストの違反カウント

次のクエリは、[Host Violation Count] ワークフローで [Host Violation Count] ページを複製します。ユーザ設定でデフォルトの [Compliance White List Violations] ワークフローを変更していない場合、これが Firepower Management Center Web インターフェイスで [Analysis] > [Correlation] > [White List Violations] を選択すると最初に表示されるページです。

```
SELECT host_id, HEX(host_id), white_list_name, count(*) AS total

FROM white_list_violation

GROUP BY host_id, white_list_name

ORDER BY total DESC;
```

スキーマ：システムレベル テーブル

この章では、システムレベルの機能（監査、アプライアンス ヘルス モニタ、マルウェア検出、およびセキュリティ更新のログ記録など）のスキーマとサポートされている結合について説明します。

詳細については、次の表に示す項を参照してください。

表 3-1　システムレベル テーブルのスキーマ

参照先	次の内容が格納されるテーブル	バージョン
audit_log（3-1 ページ）	アプライアンスの Web インターフェイスとのユーザ インタラクション。	4.10.x+
fireamp_event（3-2 ページ）	AMP for Endpointsマルウェア検出および検疫イベント。	5.1+
health_event（3-9 ページ）	モニタ対象アプライアンスのヘルス ステータス イベント。	4.10.x+
sru_import_log（3-10 ページ）	アプライアンスにインポートされたルールの更新。	5.0+

audit_log

audit_log テーブルには、Firepower システム ユーザと Web インターフェイスのインタラクションに関する情報が格納されます。監査ログにはローカル アプライアンスのレコードだけが格納され、管理対象アプライアンスのレコードは格納されない点に注意してください。

詳細については、次の項を参照してください。

- audit_log のフィールド（3-1 ページ）
- audit_log の結合（3-2 ページ）
- audit_log のサンプル クエリ（3-2 ページ）

audit_log のフィールド

次の表に、**audit_log** テーブルでアクセスできるデータベース フィールドについて説明します。

表 3-2　audit_log のフィールド

フィールド	説明
action_time_sec	アプライアンスにより監査レコードが生成された日時を示す UNIX タイムスタンプ。
domain_name	ユーザがログインしたドメインの名前。

表 3-2　audit_log のフィールド(続き)

フィールド	説明
domain_uuid	ユーザがログインしたドメインの UUID。これはバイナリで示されます。
message	ユーザが実行した操作。
source	Web インターフェイス ユーザのホストの IP アドレス(ドット形式 10 進表記法)。
subsystem	監査レコードが生成されたときにユーザがたどったメニュー パス。
user	監査イベントをトリガーしたユーザのユーザ名。

audit_log の結合

audit_log テーブルに対して結合を実行することはできません。

audit_log のサンプル クエリ

次のクエリは、最大 25 件の最新監査ログ エントリを、時刻でソートし、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT from_unixtime(action_time_sec)
AS Time, user, subsystem, message, source, count(*)
AS Total
FROM audit_log
GROUP BY source, subsystem, user, message
WHERE domain_name= "Global \ Company B \ Edge"
ORDER BY source DESC;
```

fireamp_event

fireamp_event テーブルには、AMP for Endpoints により検出されたマルウェア イベントと、AMP for Firepower により検出されたネットワーク ベースのイベントに関する情報が格納されます。これらのイベントには、クラウド内で検出または検疫されたマルウェア、検出方法、マルウェアの影響を受けるホストとユーザに関する情報が含まれています。個々のマルウェア イベントに関するその他の情報は、イベントの生成方法と生成理由に応じて異なります。

AMP for Firepower はネットワーク トラフィックでマルウェア ファイルを検出することから、ネットワーク ベースのマルウェア イベントには、ファイルの送信に使用された接続に関する、ポート、アプリケーション プロトコル、および送信元 IP アドレスの情報が含まれます。

AMP for Endpoints 展開からインポートされたマルウェア イベントと IOC には、コンテキスト接続情報は含まれていませんが、ダウンロード時または実行時に取得された情報(ファイル パス、呼び出し元クライアント アプリケーションなど)が含まれています。

詳細については、次の項を参照してください。

- fireamp_event のフィールド (3-3 ページ)
- fireamp_event の結合 (3-8 ページ)
- fireamp_event のサンプル クエリ (3-8 ページ)

fireamp_event のフィールド

次の表に、**fireamp_event** テーブルでアクセスできるデータベース フィールドについて説明します。

表 3-3　fireamp_event のフィールド

フィールド	説明
application_id	ファイル転送を実行するアプリケーションにマップされている ID 番号。
application_name	転送を実行するアプリケーションの名前。
cert_valid_end_date	接続で使用された SSL 証明書が有効ではなくなった時点を示す UNIX タイムスタンプ。
cert_valid_start_date	接続で使用された SSL 証明書の発行時点を示す UNIX タイムスタンプ。
client_application_id	クライアント アプリケーションの内部識別番号(該当する場合)。
client_application_name	クライアント アプリケーションの名前(該当する場合)。
cloud_name	マルウェア イベントの発生元クラウド サービスの名前。各 cloud_name 値には cloud_uuid 値が関連付けられています。
cloud_uuid	マルウェア イベントの発生元クラウド サービスの内部の固有 ID。各 cloud_uuid 値には cloud_name 値が関連付けられています。
connection_sec	マルウェア イベントに関連付けられている接続イベントの UNIX タイムスタンプ(00:00:00 01/01/1970 からの経過秒数)。
counter	同じ秒数で発生した複数のイベントを区別するために使用されるイベント固有のカウンタ。
detection_name	検出されたマルウェアまたは検疫されたマルウェアの名前。
detector_type	マルウェアを検出したディテクタ。各 detector_type 値には detector_type_id 値が関連付けられています。有効な表示値とその関連 ID を次に示します。 • ClamAV:128 • ETHOS:8 • SPERO:32 • SHA:4 • Tetra:64
detector_type_id	マルウェアを検出した検出テクノロジーの内部 ID。各 detector_type_id 値には detector_type 値が関連付けられています。有効な表示値とその関連タイプを次に示します。 • 4:SHA • 8:ETHOS • 32:SPERO • 64:Tetra • 128:ClamAV

表 3-3　fireamp_event のフィールド(続き)

フィールド	説明
direction	ファイルのアップロードとダウンロードのいずれが行われたかを示す値。次のいずれかの値になります。 • Download • Upload 現時点では、この値はプロトコルに依存しています(例えば接続が HTTP の場合はダウンロード)。
disposition	ファイルのマルウェア ステータス。有効な値は次のとおりです。 • CLEAN:ファイルはクリーンであり、マルウェアが含まれていない。 • UNKNOWN:ファイルにマルウェアが含まれているかどうかが不明である。 • MALWARE:ファイルにマルウェアが含まれている。 • UNAVAILABLE:ソフトウェアから シスコ クラウドに対して、特性を確認する要求を送信できなかったか、または シスコ クラウド サービスが要求に応答しなかった。 • CUSTOM SIGNATURE:ファイルがユーザ定義のハッシュと一致するため、ユーザが指定した方法で処理された。
domain_name	イベントが検出されたドメインの名前。
domain_uuid	イベントが検出されたドメインの UUID。これはバイナリで示されます。
dst_continent_name	宛先ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
dst_country_id	宛先ホストの国のコード。
dst_country_name	宛先ホストの国の名前。
dst_ip_address_v6	このフィールドは廃止されており、null を返します。
dst_ipaddr	接続の宛先の IPv4 または IPv6 アドレスのバイナリ表現。
dst_port	接続の宛先のポート番号。
endpoint_user	シスコ クラウドによりイベントが検出された場合に シスコ AMP for Endpoints エージェントにより判別されるユーザ。このユーザは LDAP に関連付けられておらず、discovered_users テーブルに表示されません。
event_description	イベント タイプに関連付けられている追加イベント情報。
event_id	マルウェア イベントの内部固有 ID。

表 3-3　fireamp_event のフィールド（続き）

フィールド	説明
event_subtype	マルウェア検出につながったアクション。各 event_subtype 値には event_subtype_id 値が関連付けられています。有効な表示値とその関連 ID を次に示します。 • Create:1 • Execute:2 • Move:22 • Scan:4
event_subtype_id	マルウェア検出につながったアクションの内部 ID。各 event_subtype_id 値には event_subtype 値が関連付けられています。有効な表示値とその関連サブタイプを次に示します。 • 1:作成 • 2:実行 • 4:スキャン • 22:移動
event_type	マルウェア イベントのタイプ。各 event_type 値には event_type_id 値が関連付けられています。有効な表示値とその関連 ID を次に示します。 • Blocked Execution:553648168 • Cloud Recall Quarantine:553648155 • Cloud Recall Quarantine Attempt Failed:2164260893 • Cloud Recall Quarantine Started:553648147 • Cloud Recall Restore from Quarantine:553648154 • Cloud Recall Restore from Quarantine Failed:2164260892 • Cloud Recall Restore from Quarantine Started:553648146 • FireAMP IOC:1107296256 • Quarantine Failure:2164260880 • Quarantined Item Restored:553648149 • Quarantine Restore Failed:2164260884 • Quarantine Restore Started:553648150 • Scan Completed, No Detections:554696715 • Scan Completed With Detections:1091567628 • Scan Failed:2165309453 • Scan Started:554696714 • Threat Detected:1090519054 • Threat Detected in Exclusion:553648145 • Threat Detected in Network File Transfer:1 • Threat Detected in Network File Transfer (Retrospective):2 • Threat Quarantined:553648143

表 3-3　fireamp_event のフィールド(続き)

フィールド	説明
event_type_id	マルウェア イベント タイプの内部 ID。各 event_type_id 値には event_type 値が関連付けられています。有効な表示値とその関連タイプを次に示します。 • 553648143:Threat Quarantined • 553648145:Threat Detected in Exclusion • 553648146:Cloud Recall Restore from Quarantine Started • 553648147:Cloud Recall Quarantine Started • 553648149:Quarantined Item Restored • 553648150:Quarantine Restore Started • 553648154:Cloud Recall Restore from Quarantine • 553648155:Cloud Recall Quarantine • 553648168:Blocked Execution • 554696714:Scan Started • 554696715:Scan Completed, No Detections • 1090519054:Threat Detected • 1091567628:Scan Completed With Detections • 1107296256:FireAMP IOC • 2164260880:Quarantine Failure • 2164260893:Cloud Recall Quarantine Attempt Failed • 2164260884:Quarantine Restore Failed • 2164260892:Cloud Recall Restore from Quarantine Failed • 2165309453:Scan Failed
file_name	検出または検疫されたファイルの名前。この名前には、UTF-8 文字を使用できます。
file_path	検出または検疫されたファイルのファイル パス。ファイル名は含まれません。このパスには、UTF-8 文字を使用できます。
file_sha	検出または検疫されたファイルの SHA-256 ハッシュ値。
file_size	検出または検疫されたファイルのサイズ(バイト単位)。
file_timestamp	検出または検疫されたファイルの作成タイムスタンプ。
file_type	検出または検疫されたファイルのファイル タイプ。
file_type_id	検出または検疫されたファイルのファイル タイプの内部 ID。
http_response_code	イベントで HTTP 要求に対して返された応答コード。
instance_id	イベントを生成した管理対象デバイスの Snort インスタンスの数値 ID。
ioc_count	イベントで検出された侵害の痕跡の数。
parent_file_name	検出が行われたときに、検出または検疫されたファイルにアクセスしたファイルの名前。
parent_file_sha	検出が行われたときに、検出または検疫されたファイルにアクセスした親ファイルの SHA-256 のハッシュ値。

表 3-3　fireamp_event のフィールド(続き)

フィールド	説明
policy_uuid	イベントをトリガーしたアクセス コントロール ポリシーの固有識別子として機能する識別番号。
retroactive_disposition	特性が更新されている場合のファイルの特性。特性が更新されていない場合、このフィールドには disposition フィールドと同じ値が格納されます。有効な値は disposition フィールドと同じです。
得点	動的分析中に観測された、悪意のある可能性がある振る舞いに基づく数値(0 ～ 100)。
security_context	トラフィックが通過したセキュリティ コンテキスト(仮想ファイアウォール)の説明。システムがこのフィールドにデータを設定するのは、マルチ コンテキスト モードの ASA FirePOWER デバイスの場合だけです。
sensor_address	イベントを生成したデバイスの IP アドレス。
sensor_id	イベントを生成したデバイスの ID。
sensor_name	イベント レコードを生成した管理対象デバイスのテキスト名。接続デバイスではなくレポート デバイス自体を参照するイベントの場合、このフィールドは null です。
sensor_uuid	管理対象デバイスの固有識別子(fireamp_event.sensor_name が null の場合は 0)。
src_continent_name	送信元ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
src_country_id	送信元ホストの国のコード。
src_country_name	送信元ホストの国の名前。
src_ip_address_v6	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
src_ipaddr	接続元の IPv4 または IPv6 アドレスのバイナリ表現。
src_port	接続元のポート番号。
ssl_issuer_common_name	SSL 証明書の発行元の共通名。これは一般に証明書発行元のホストとドメイン名ですが、その他の情報が含まれていることもあります。
ssl_issuer_country	SSL 証明書の発行元の国。
ssl_issuer_organization	SSL 証明書の発行元の組織。
ssl_issuer_organization_unit	SSL 証明書の発行元の組織単位。
ssl_serial_number	発行元 CA によって割り当てられた SSL 証明書のシリアル番号。
ssl_subject_common_name	SSL 証明書の件名共通名。これは通常、証明書件名のホストとドメイン名ですが、他の情報が含まれていることもあります。
ssl_subject_country	SSL 証明書の件名の国。

表 3-3　fireamp_event のフィールド(続き)

フィールド	説明
ssl_subject_organization	SSL 証明書の件名の組織。
ssl_subject_organization_unit	SSL 証明書の件名の組織単位。
threat_name	脅威の名前。
timestamp	マルウェア イベント生成時のタイムスタンプ。
url	接続元の URL。
user_id	ファイルを送信または受信したホストの最終ログイン ユーザの内部識別番号。このユーザは **discovered_users** テーブルに含まれています。
username	ファイルを送信または受信したホストの最終ログイン ユーザの名前。
web_application_id	Web アプリケーションの内部識別番号(該当する場合)。
web_application_name	Web アプリケーションの名前(該当する場合)。

fireamp_event の結合

次の表に、**fireamp_event** テーブルで実行できる結合について説明します。

表 3-4　fireamp_event の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
dst_ipaddr または src_ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

fireamp_event のサンプル クエリ

次のクエリは、指定されたユーザに関連付けられている 25 件のマルウェア イベントを、timestamp の昇順にソートして返します。

```
SELECT event_id, timestamp, src_ipaddr, dst_ipaddr, username, cloud_name, event_type,
event_subtype, event_description, detection_name, detector_type, file_name,
parent_file_name

FROM fireamp_event

WHERE username="username" ORDER BY timestamp ASC

LIMIT 25;
```

health_event

health_event テーブルには、Firepower システム によって生成されたヘルス イベントに関する情報が格納されます。

詳細については、次の項を参照してください。

- health_event のフィールド（3-9 ページ）
- health_event の結合（3-10 ページ）
- health_event のサンプル クエリ（3-10 ページ）

health_event のフィールド

次の表に、**health_event** テーブルでアクセスできるデータベース フィールドについて説明します。

表 3-5　health_event のフィールド

フィールド	説明
description	関連するヘルス モジュールがヘルス イベントを生成した条件の説明。たとえば、プロセスが実行できない場合に生成されるヘルス イベントには [Unable to Execute] というラベルが付けられます。
domain_name	イベントが検出されたドメインの名前。
domain_uuid	イベントが検出されたドメインの UUID。これはバイナリで示されます。
event_time_sec	Firepower Management Center によりヘルス イベントが生成された日時を示す UNIX タイムスタンプ。
id	イベントの内部識別番号。
module_name	イベントを生成したヘルス モジュールの名前。
sensor_name	イベント レコードを生成した管理対象デバイスのテキスト名。接続デバイスではなくレポート デバイス自体を参照するヘルス イベントの場合、このフィールドは **null** です。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
status	sensor_uuid で識別されるアプライアンスについて報告されたヘルス モニタ ステータス。値は次のとおりです。 • red：重大ステータス。アプライアンス上の 1 つ以上のヘルス モジュールが制限を超え、問題が解決されていません。 • yellow：警告ステータス。アプライアンス上の 1 つ以上のヘルス モジュールが制限を超え、問題が解決されていません。 • green：通常ステータス。アプライアンス上のすべてのヘルス モジュールがアプライアンスに適用された正常性ポリシーで設定された制限内で動作しています。 • recovered：アプライアンス上のすべてのヘルス モジュールがアプライアンスに適用された正常性ポリシーで設定された制限内で動作しています。これには、前に重大または警告状態だったモジュールも含まれます。 • disabled：アプライアンスが無効またはブラックリストに追加されているか、現在到達不能であるか、または正常性ポリシーがアプライアンスに適用されていません。 • error：アプライアンス上の 1 つ以上のヘルス モニタリング モジュールで障害が発生し、それ以降、正常に再実行されていません。

表 3-5　health_event のフィールド（続き）

フィールド	説明
units	ヘルス テストの結果の測定単位。たとえば %（ディスク使用量のパーセンテージ）。
value	ヘルス テストの結果の単位数。例えば 80% の場合 value は 80 です。

health_event の結合

health_eventg テーブルに対して結合を実行することはできません。

health_event のサンプル クエリ

次のクエリは、定義された時間枠内にログに記録された最大 25 件の最新のヘルス イベントを、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT module_name, FROM_UNIXTIME(event_time_sec)
AS event_time, description, value, units, status, sensor_name
FROM health_event
WHERE event_time_sec AND domain_name= "Global \ Company B \ Edge"
BETWEEN UNIX_TIMESTAMP("2011-10-01 00:00:00")
AND UNIX_TIMESTAMP("2011-10-07 23:59:59")
ORDER BY event_time DESC
LIMIT 0, 25;
```

sru_import_log

sru_import_log テーブルには、アプライアンスで実行されたルール更新プロセスについて説明します。**sru_import_log** テーブルは、Firepower システム バージョン 5.0 以降で廃止されたテーブル **seu_import_log** を置き換えます。

詳細については、次の項を参照してください。

- sru_import_log のフィールド（3-11 ページ）
- sru_import_log の結合（3-12 ページ）
- sru_import_log のサンプル クエリ（3-12 ページ）

sru_import_log のフィールド

次の表に、**sru_import_log** テーブルでアクセスできるデータベース フィールドについて説明します。

表 3-6　sru_import_log のフィールド

フィールド	説明
action	インポートされたルール更新オブジェクト タイプに対して行われたアクションを示します。 • apply:[Reapply intrusion policies after the Rule Update import completes] オプションがインポートで有効にされた。 • changed:ルール更新コンポーネントまたはルールで、ルール更新コンポーネントが変更されたか、ルールのリビジョン番号が大きく、GID と SID が同じだった。 • collision:ルール更新コンポーネントまたはルールで、アプライアンス上の既存のコンポーネントまたはルールとリビジョンが競合していることから、インポートがスキップされた。 • deleted:ルールで、ルール更新からルールが削除された。 • disabled:ルールで、システム提供のデフォルト ポリシーでルールが無効になっていた。 • drop:ルールで、システム提供のデフォルト ポリシーで、ルールが [Drop and Generate Events] に設定されていた。 • enabled:ルール更新で、編集、プリプロセッサ、ルール、またはルール更新により提供されるその他の機能が、システム提供のデフォルト ポリシーで有効になっていた。 • error:ルール更新またはローカル ルール ファイルで、インポートが失敗した。 • new:ルールで、このアプライアンスにオブジェクトが初めて格納された。
detail	インポートされたルール更新によってコンポーネントまたはルールに適用される変更に固有のコメント文字列。ルールが変更されていない場合は空白。
domain_name	ルールが使用されたドメインの名前。
domain_uuid	ルールが使用されたドメインの UUID。これはバイナリで示されます。
generator_id	ルールのジェネレータの GID。
import_time_sec	ルール更新インポートがログに記録された日時を示す UNIX タイムスタンプ。
name	インポート オブジェクトの名前。ルールでは、これはルール メッセージに対応します。ルール更新コンポーネントでは、これはコンポーネント名(オンライン ヘルプ、Snort など)です。
policy	All:ルールがすべてのデフォルト ポリシーに含まれていることを示します。
revision	ルールのリビジョン番号。
signature_id	ルール、ルール セット、デコーダ、またはプリプロセッサの SID。
sru_name	ルール更新の記述名。
sru_uuid	ルール更新の固有識別子。
type	ルール更新でインポートしたオブジェクトのタイプ:update、rule、variable などです。

sru_import_log の結合

sru_import_log テーブルに対して結合を実行することはできません。

sru_import_log のサンプル クエリ

次のクエリは、最大 25 件の結果をタイムスタンプの降順にソートし、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT FROM_UNIXTIME(import_time_sec)
AS time, name, type, action, generator_id, signature_id, revision, policy
FROM sru_import_log
WHERE domain_name= "Global \ Company B \ Edge"
ORDER BY time DESC
LIMIT 0, 25;
```

スキーマ：侵入テーブル

この章では、侵入イベント、侵入イベントをトリガーしたパケット、および関連するルール メッセージのスキーマとサポートされている結合について説明します。

詳細については、次の表に示す項を参照してください。

表 4-1　　侵入テーブルのスキーマ

参照先	次の内容が格納されるテーブル	バージョン
intrusion_event（4-1 ページ）	侵入イベント（エクスプロイトの日時とタイプ、および攻撃の攻撃元と標的に関するコンテキスト情報など）。	4.10.x+
intrusion_event_packet（4-7 ページ）	侵入イベントをトリガーした 1 つ以上のパケットの内容。	4.10.x+
rule_message（4-8 ページ）	侵入イベントのルール メッセージ（関連付けられているジェネレータ ID（GID）、シグニチャ ID（SID）、およびバージョン データなどを含む）。	4.10.x+
rule_documentation（4-9 ページ）	ルールに関する情報（攻撃シナリオ、影響を受けるシステム、およびルールの作成時点と作成者に関する情報など）。	5.2+

intrusion_event

`intrusion_event` テーブルには、Firepower システム により特定された侵入の可能性に関する情報が格納されます。可能性のある侵入ごとに、イベントと関連レコードがデータベースに生成されます。このレコードには、エクスプロイトの日時とタイプ、アクセス コントロール ポリシーとルール、侵入ポリシーとルール、および攻撃の攻撃元と標的に関するその他のコンテキスト情報が含まれています。

ヒント　パケットベースのイベントの場合、イベントをトリガーした 1 つ以上のパケットのコピーも使用可能になります。intrusion_event_packet のサンプル クエリ（4-8 ページ）を参照してください。

詳細については、次の項を参照してください。

- intrusion_event のフィールド（4-2 ページ）
- intrusion_event の結合（4-6 ページ）
- intrusion_event のサンプル クエリ（4-7 ページ）

intrusion_event のフィールド

次の表に、**intrusion_event** テーブルでアクセスできるデータベース フィールドについて説明します。

表 4-2　intrusion_event のフィールド

フィールド	説明
access_control_policy_name	侵入イベントを生成した侵入ポリシーに関連付けられているアクセス コントロール ポリシー。アクセス コントロール ポリシー名とアクセス コントロール ルール名の組み合わせは、Firepower Management Center で一意である点に注意してください。
access_control_policy_UUID	侵入イベントを生成した侵入ポリシーに関連付けられているアクセス コントロール ポリシーの UUID。
access_control_rule_id	侵入イベントを生成した侵入ポリシーに関連付けられているアクセス コントロール ルールの内部識別番号。
access_control_rule_name	侵入イベントを生成した侵入ポリシーに関連付けられているアクセス コントロール ルールの名前。アクセス コントロール ルールの名前は、1 つのポリシー内では固有であるが、異なるポリシー間では固有ではない点に注意してください。
application_protocol_id	アプリケーション プロトコルの内部識別番号。
application_protocol_name	次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合) • pending(システムがさらにデータを必要としている場合) • 空白(接続にアプリケーション情報がない場合)
blocked	侵入イベントをトリガーしたパケットの処理を示す値。 • 0:パケットはドロップされなかった • 1:パケットはドロップされた(インライン型、スイッチ型、またはルーティング型展開) • 2:侵入ポリシーが、インライン型、スイッチ型、またはルーティング型展開で設定されているデバイスに適用されている場合は、イベントをトリガーしたパケットがドロップされている可能性がある。
client_application_id	侵入イベントで使用されたクライアント アプリケーションの内部識別番号。
client_application_name	侵入イベントで使用されたクライアント アプリケーション(使用可能な場合)。次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合) • 汎用クライアント名(システムがクライアント アプリケーションを検出したが、特定のアプリケーションであることを識別できない場合)。 • null(接続にアプリケーション情報がない場合)
connection_sec	侵入イベントに関連付けられている接続イベントの UNIX タイムスタンプ(00:00:00 01/01/1970 からの経過秒数)。
counter	特定の秒で発生した接続イベントごとに増加する番号。これは、同じ秒で発生した複数の接続イベントを区別するために使用されます。

表 4-2　intrusion_event のフィールド(続き)

フィールド	説明
detection_engine_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
detection_engine_uuid	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
domain_name	イベントのために指定されたドメインの名前。
domain_uuid	イベントのために指定されたドメインの UUID。これはバイナリで示されます。
dst_continent_name	宛先ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
dst_country_id	宛先ホストの国のコード。
dst_country_name	宛先ホストの国の名前。
dst_ip_address	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
dst_ip_address_v6	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
dst_ipaddr	トリガー イベントに関連する宛先ホストの IPv4 または IP v 6 アドレスのバイナリ表現。
dst_port	次のいずれかを行います。 • イベント プロトコル タイプが TCP または UDP の場合は宛先ポート番号 • イベント プロトコルタイプが ICMP の場合は ICMP コード。
dst_user_dept	宛先ユーザの所属部門。
dst_user_email	宛先ユーザの電子メール アドレス。
dst_user_first_name	宛先ユーザの名前。
dst_user_id	宛先ユーザの内部識別番号。宛先ユーザとは、侵入イベント発生前に宛先ホストにログインしていた最終ユーザです。
dst_user_last_name	宛先ユーザの姓。
dst_user_last_seen_sec	システムが宛先ユーザのログインを最後に報告した日時を示す UNIX タイムスタンプ。
dst_user_last_updated_sec	システムが宛先ユーザのレコードを最後に更新した日時を示す UNIX タイムスタンプ。
dst_user_name	宛先ユーザのユーザ名。

表 4-2　intrusion_event のフィールド(続き)

フィールド	説明
dst_user_phone	宛先ユーザの電話番号。
event_id	イベントの内部識別番号。Firepower Management Center でイベントを一意に識別します。
event_time_sec	イベント パケットがキャプチャされた日時を示す UNIX タイムスタンプ。
event_time_usec	イベントのタイムスタンプのマイクロ秒単位の増分。マイクロ秒単位の精度を使用できない場合、この値は 0 です。
http_response_code	イベントで HTTP 要求に対して返された応答コード。
icmp_code	イベントが ICMP トラフィックの場合は ICMP コード。イベントが ICMP トラフィックから生成されたものではない場合は null。
icmp_type	イベントが ICMP トラフィックの場合は ICMP タイプ。イベントが ICMP トラフィックから生成されたものではない場合は null。
impact	イベントの影響フラグ値。整数値は次のとおりです。 • 1:レッド(脆弱) • 2:オレンジ(脆弱の可能性あり) • 3:イエロー(現在は脆弱でない) • 4:ブルー(不明なターゲット) • 5:グレー(不明な影響)
instance_id	イベントを生成した管理対象デバイスの Snort インスタンスの数値 ID。
interface_egress_name	発信トラフィックのインターフェイスの名前。
interface_ingress_name	着信トラフィックのインターフェイスの名前。
intrusion_event_policy_uuid	侵入イベントをトリガーした侵入ポリシーの固有識別子。
intrusion_event_policy_name	侵入イベントを生成した侵入ポリシー。
ioc_count	イベントで検出された侵害の痕跡の数。
network_analysis_policy_name	侵入イベントを生成した侵入ポリシーに関連付けられているネットワーク分析ポリシー。
network_analysis_policy_UUID	侵入イベントを生成した侵入ポリシーに関連付けられているネットワーク分析ポリシーの UUID。
priority	イベントに関連付けられているルール分類のプライオリティ。ルールのプライオリティはユーザ インターフェイスで設定されます。
protocol_name	侵入イベントに関連付けられているトラフィック プロトコルのテキスト名。
protocol_num	プロトコルの IANA 番号。IANA 番号のリストは http://www.iana.org/assignments/protocol-numbers にあります。
reviewed	侵入イベントが確認済みとしてマークされているかどうか。 • 1:確認済み • 0:未確認
rule_classification	侵入イベントに関連付けられているルール分類の説明。通常、イベントをトリガーしたルールによって検出された攻撃を説明します。例:A Network Trojan was Detected.
rule_classification_id	侵入イベントに関連付けられているルール分類の識別番号。

表 4-2　　intrusion_event のフィールド(続き)

フィールド	説明
rule_generator	侵入イベントを生成したコンポーネント。生成コンポーネントは、ルール エンジン、デコーダ、またはプリプロセッサです。
rule_generator_id	侵入イベントを生成した rule_generator で指定されているコンポーネントのジェネレータ ID(GID)。
rule_message	イベントを説明するテキスト。ルールベースの侵入イベントの場合、メッセージはルールから生成されます。デコーダベースおよびプリプロセッサ ベースのイベントの場合、メッセージはハード コーディングされています。
rule_revision	侵入イベントに関連付けられているルールのリビジョン番号。
rule_signature_id	侵入イベントのシグニチャ ID(SID)。侵入イベントが生成される原因である特定のルール、デコーダ メッセージ、またはプリプロセッサ メッセージを識別します。
security_context	トラフィックが通過したセキュリティ コンテキスト(仮想ファイアウォール)の説明。マルチコンテキスト モードの ASA FirePOWER デバイスでは、システムはこのフィールドにのみ入力することに注意してください。
security_zone_egress_name	ポリシー違反をトリガーした侵入イベントの出力セキュリティ ゾーン。
security_zone_ingress_name	ポリシー違反をトリガーした侵入イベントの入力セキュリティ ゾーン。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。
src_continent_name	宛先ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
src_country_id	宛先ホストの国のコード。
src_country_name	宛先ホストの国の名前。
src_ip_address	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
src_ip_address_v6	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
src_ipaddr	トリガー イベントに関連する送信元ホストの IPv4 または IPv6 アドレスのバイナリ表現。

表 4-2　intrusion_event のフィールド（続き）

フィールド	説明
src_port	次のいずれかを行います。 • イベント プロトコル タイプが TCP または UDP の場合は送信元ポート番号 • イベント プロトコルタイプが ICMP の場合は ICMP タイプ。
src_user_dept	送信元ユーザの所属部門。
src_user_email	送信元ユーザの電子メール アドレス。
src_user_first_name	送信元ユーザの名前。
src_user_id	送信元ユーザの内部識別番号。送信元ユーザとは、侵入イベント発生前に送信元ホストにログインしていた最終ユーザです。
src_user_last_name	送信元ユーザの姓。
src_user_last_seen_sec	システムが送信元ユーザのログインを最後に報告した日時を示す UNIX タイムスタンプ。
src_user_last_updated_sec	送信元ユーザのレコードの最終更新日時を示す UNIX タイムスタンプ。
src_user_name	送信元ユーザのユーザ名。
src_user_phone	送信元ユーザの電話番号。
vlan_id	侵入イベントをトリガーしたパケットに関連付けられている最も内側の VLAN の識別番号。
web_application_id	侵入イベントで使用された Web アプリケーションの内部識別番号（該当する場合）。
web_application_name	侵入イベントで使用された Web アプリケーション（該当する場合）。次のいずれかになります。 • アプリケーションの名前（確実な識別が可能な場合） • web browsing（システムがアプリケーション プロトコル HTTP を検出したが、特定の Web アプリケーションを検出できない場合）。 • 空白（接続に HTTP トラフィックがない場合）。

intrusion_event の結合

次の表に、**intrusion_event** テーブルで実行できる結合について説明します。

表 4-3　intrusion_event の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_protocol_id または client_application_id または web_application_id	application_info.application_id application_host_map.application_id application_tag_map.application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id

表 4-3　intrusion_event の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
dst_ipaddr または src_ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

intrusion_event のサンプル クエリ

次のクエリは、25 件の最も頻繁に発生した未確認の侵入イベント結果を、Count に基づいて降順にソートして返します。

```
SELECT rule_message, priority, rule_classification, count(*) as Count
FROM intrusion_event
WHERE reviewed="0"
GROUP BY rule_message, priority, rule_classification
ORDER BY Count DESCLIMIT 0, 25;
```

intrusion_event_packet

intrusion_event_packet テーブルには、侵入イベントをトリガーした 1 つ以上のパケットの内容に関する情報が格納されます。管理対象デバイスから Firepower Management Center へのパケット転送を禁止している場合は、**intrusion_event_packet** テーブルにはデータが格納されません。

詳細については、次の項を参照してください。

- intrusion_event_packet のフィールド（4-7 ページ）
- intrusion_event_packet の結合（4-8 ページ）
- intrusion_event_packet のサンプル クエリ（4-8 ページ）

intrusion_event_packet のフィールド

次の表に、**intrusion_event_packet** テーブルでアクセスできるデータベース フィールドについて説明します。

表 4-4　intrusion_event_packet のフィールド

フィールド	説明
detection_engine_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
detection_engine_uuid	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
domain_name	イベントのために指定されたドメインの名前。
domain_uuid	イベントのために指定されたドメインの UUID。これはバイナリで示されます。
event_id	イベントの識別番号。この ID は、特定の管理対象デバイスにおいて固有です。

表 4-4　intrusion_event_packet のフィールド（続き）

フィールド	説明
linktype	パケットの外部レイヤの形式を示す内部キー。管理対象デバイスがパケットを正しく復号化するためにこのキーを使用します。リンク タイプ 1 だけがサポートされています。
netmap_num	イベントが検出されたドメインの Netmap ID。
packet_data	イベントをトリガーしたパケットの内容。
packet_time_sec	イベント パケットがキャプチャされた日時を示す UNIX タイムスタンプ。
packet_time_usec	イベントのタイムスタンプのマイクロ秒単位の増分。マイクロ秒単位の精度を使用できない場合、この値は 0 です。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。

intrusion_event_packet の結合

intrusion_event_packet テーブルに対して結合を実行することはできません。

intrusion_event_packet のサンプル クエリ

次のクエリは、選択されたイベント ID に一致するすべてのパケットに関するパケット情報を返します。

```
SELECT event_id, packet_time_sec, sensor_address, packet_data
FROM intrusion_event_packet
WHERE event_id="1";
```

rule_message

rule_message テーブルは、侵入ルールのルール メッセージのマスタ リストです。各ルール メッセージにはその識別情報が付いています。

詳細については、次の項を参照してください。

- rule_message のフィールド (4-9 ページ)
- rule_message の結合 (4-9 ページ)
- rule_message のサンプル クエリ (4-9 ページ)

rule_message のフィールド

次の表に、**rule_message** テーブルでアクセスできるデータベース フィールドについて説明します。

表 4-5　rule_message のフィールド

フィールド	説明
generator_id	ルールをトリガーするコンポーネントの GID。
message	トリガーされたルールに関連付けられているメッセージ。
rev_uuid	ルール リビジョンの固有識別子。
revision	ルールのリビジョン番号。
signature_id	アプライアンスのユーザ インターフェイスでレンダリングされるルールの識別番号。
uuid	ルールの固有識別子。

rule_message の結合

rule_message テーブルに対して結合を実行することはできません。

rule_message のサンプル クエリ

次のクエリは、GID が 1、SID が 1200 の侵入ルールの侵入ルール メッセージを返します。

```
SELECT generator_id, signature_id, revision, message
FROM rule_message
WHERE generator_id="1"
AND signature_id="1200";
```

rule_documentation

rule_documentation 表には、アラートの生成に使用されたルールに関する情報が格納されます。詳細については、次の項を参照してください。

- rule_documentation のフィールド（4-10 ページ）
- rule_documentation の結合（4-10 ページ）
- rule_documentation のサンプル クエリ（4-10 ページ）

rule_documentation のフィールド

次の表に、**rule_documentation** テーブルでアクセスできるデータベース フィールドについて説明します。

表 4-6　　rule_documentation のフィールド

フィールド	説明
additional_references	その他の情報およびリファレンス。
affected_systems	脆弱性の影響を受けるシステム。
attack_scenarios	潜在的な攻撃の例。
contributors	ルールおよびその他の関連ドキュメントの作成者の連絡先情報。
corrective_action	脆弱性を排除または緩和するためのパッチ、更新、およびその他の手段に関する情報。
detailed_information	基礎となる脆弱性、ルールが実際に検索する内容、および影響を受けるシステムに関する情報。
ease_of_attack	攻撃の難易度（simple、medium、hard、または difficult）と、その攻撃がスクリプトを使用して実行できるものであるかどうか。
false_negatives	検出漏れとなる可能性のある例。デフォルト値は None Known です。
false_positives	誤検出となる可能性のある例。デフォルト値は None Known です。
impact	この脆弱性を利用した侵害がさまざまなシステムに与える可能性のある影響。
rule_revision	ルール リビジョン番号。
rule_signature_id	イベントに対応するルールの識別番号。
summary	脅威または脆弱性の説明。
updated	ルールの最終更新日時を示す UNIX タイムスタンプ。

rule_documentation の結合

rule_documentation テーブルに対して結合を実行することはできません。

rule_documentation のサンプル クエリ

次のクエリは、ID が 1 の侵入ルールの攻撃シナリオ、修正措置、影響、およびサマリを返します。

```
SELECT attack_scenarios, corrective_action, impact, summary
FROM rule_documentation
WHERE rule_signature_id="1";
```

CHAPTER 5

スキーマ：統計情報追跡テーブル

この章では、アプリケーションと URL の統計情報追跡テーブルのスキーマとサポートされている結合について説明します。これらのテーブルには、次の統計情報が収集されます。

- アクセス コントロールと侵入イベント（アプリケーション別およびユーザ別）
- 帯域幅の使用状況と接続に関する決定（アプリケーション別およびユーザ別）
- 帯域幅の使用状況と接続に関する決定（URL レピュテーション（リスク）別および URL のビジネス関連度別）

各テーブルの詳細情報へのリンクについては、次の表を参照してください。

表 5-1　アプリケーションと URL の統計情報のテーブル

参照先	次の統計情報が格納されるテーブル	バージョン
app_ids_stats_current_timeframe（5-4 ページ）	アクセス コントロール アクティビティおよび侵入からの保護アクティビティ（アプリケーション別およびアプリケーション属性の範囲別）	5.0+
app_stats_current_timeframe（5-6 ページ）	トラフィック ボリュームとシステム アクセス コントロール アクティビティ（接続の許可また拒否）（アプリケーション別およびアプリケーション属性の範囲別）	5.0+
compliance_events_stats_current_timeframe（5-8 ページ）	コンプライアンスおよびホワイトリスト イベント	6.0+
dns_query_stats_current_timeframe（5-9 ページ）	DNS クエリ	6.0+
geolocation_stats_current_timeframe（5-11 ページ）	アクセス コントロール アクティビティ（ロケーション別）。	5.2+
ids_impact_stats_current_timeframe（5-12 ページ）	影響レベル別の侵入イベントの統計情報（ブロックされた接続とドロップされた可能性のある接続）。	5.1.1+
ip_reputation_stats_current_timeframe（5-14 ページ）	指定されたセキュリティ インテリジェンス カテゴリの IP アドレス、URL、および DNS ドメインに対する要求に関連付けられている帯域幅使用状況と接続に関する統計情報が格納されます。	6.0+
session_stats_current_timeframe（5-15 ページ）	すべての接続の統計情報が格納されます。統計情報は、バイト数、接続、センサー、および時刻に基づいて抽出できます。	5.2+

表 5-1　アプリケーションと URL の統計情報のテーブル（続き）

参照先	次の統計情報が格納されるテーブル	バージョン
ssl_stats_current_timeframe（5-17 ページ）	SSL 接続の統計情報が格納されます。統計情報は、バイト数、接続、センサー、および時刻に基づいて抽出できます。	5.4+
storage_stats_by_disposition_current_timeframe（5-19 ページ）	特性に基づくファイルの統計情報が格納されます。統計情報は、バイト数、特性、センサー、および時刻に基づいて抽出できます。	5.3+
storage_stats_by_file_type_current_timeframe（5-21 ページ）	ファイル タイプに基づくファイルの統計情報が格納されます。統計情報は、バイト数、ファイル タイプ、センサー、および時刻に基づいて抽出できます。	5.3+
transmission_stats_by_file_type_current_timeframe（5-22 ページ）	ファイル タイプに基づく接続の統計情報が格納されます。統計情報は、バイト数、接続、ファイル タイプ、センサー、および時刻に基づいて抽出できます。	5.3+
url_category_stats_current_timeframe（5-23 ページ）	要求された Web サイトのカテゴリ別のトラフィック ボリュームとシステム アクセス コントロール アクティビティ（接続の許可また拒否）	5.0+
url_reputation_stats_current_timeframe（5-25 ページ）	要求された Web サイトのレピュテーション別のトラフィック ボリュームとシステム アクセス コントロール アクティビティ（接続の許可また拒否）	5.0+
user_ids_stats_current_timeframe（5-26 ページ）	ユーザ別のアクセス コントロール アクティビティと侵入からの保護アクティビティ。	5.0+
user_stats_current_timeframe（5-28 ページ）	ユーザ別のトラフィック ボリュームとシステム アクセス コントロール アクティビティ（接続の許可また拒否）	5.0+

統計情報追跡テーブルについて

テーブルの名前の末尾に、データの時間枠を示す **current_day**、**current_month**、または **current_year** が付きます。たとえば、**app_ids_stats_current_***timeframe* は、**app_stats_current_day**、**app_stats_current_month**、および **app_stats_current_year** となります。**app_stats_current_year** テーブルには 360 日分の統計情報が格納され、**current_month** テーブルには 30 日分の統計情報が格納されます。

Firepower Management Center は、ネットワーク内の管理対象デバイスから未加工のカウントを受信するたびに、3 種類のテーブルをすべて更新しますが、更新間隔は徐々に広くなります。**current_day** テーブルの間隔が最も狭く（テーブルに応じて 15 秒または 5 分）、**current_year** テーブルでは間隔が最も広くなります（24 時間）。詳細については、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

統計情報追跡テーブルの保存特性

重要な詳細について、次の表を参照してください。

表 5-2　　統計情報テーブルの保存特性

テーブル タイプ	間隔(精度)	保存期間
current_day	次のテーブルでは 15 秒: **app_ids_stats_current_***timeframe* および **user_ids_stats_current_***timeframe*	過去 24 時間のすべての間隔に、現在の間隔を加算した期間
	次のテーブルでは 5 分: **app_stats_current_***timeframe*、 **user_stats_current_***timeframe*、 **url_category_stats_current_***timeframe*、および **url_reputation_stats_current_***timeframe*	過去 24 時間のすべての間隔に、現在の間隔を加算した期間
current_month	1 時間	過去 30 日分の時間に現在の時間を加算した期間
current_year	24 時間	過去 360 日に当日を加算した期間

保存間隔はその開始時刻により定義されます。たとえば、**current_month** テーブルには 10:00:00 ～ 10:59:59 までの間のカウントが、タイムスタンプが 10:00:00 の 1 つのレコードとして格納されます。1 日の開始時刻は 00:00:00、終了時刻は 23:59:59 であることに注意してください。間隔の開始時刻は UNIX タイムスタンプ(GMT)として保存されます。

統計情報テーブルの照会時の時間間隔の指定

クエリで有効な時間間隔は、テーブルと、クエリの time_start_sec フィールドの両方によって定義されます。

たとえば SQL ステートメントに time_start_sec = 6:00:00 と指定されている場合、間隔はテーブルのタイプに応じて次のように異なります。

- **current_day** テーブル:6:00:00 ～ 6:00:14(15 秒のテーブルの場合)または 6:00:00 ～ 6:04:59(5 分のテーブルの場合)。
- **current_month** テーブル:6:00:00 ～ 6:59:59。
- **current_year** テーブル:0:00:00 ～ 翌日の 23:59:59。

データを取得する最も簡単な方法は、間隔の開始時刻を指定する方法です。たとえば **app_ids_stats_current_day** テーブルから取得する場合は、次のいずれかを指定します。

```
00:00:00
00:00:15
00:00:30
23:59:45
```

クエリに含まれているタイムスタンプが、間隔の開始時刻と異なる場合、システムでは要求が次のように変更されます。

- 開始時刻を最も近い間隔の時刻に繰り上げます。
- 終了時刻を最も近い間隔の時刻に繰り下げます。

たとえば次のクエリでは開始時刻が繰り上げられます。

```
SELECT application_id
FROM app_ids_stats_current_month
WHERE start_time_sec = UNIX_TIMESTAMP("2011-12-01 12:30:00");
```

これは以下と同じです

```
SELECT application_id
FROM app_ids_stats_current_month
WHERE start_time_sec = UNIX_TIMESTAMP("2011-12-01 1:00:00");
```

複数の間隔からなる範囲でクエリを実行する場合は、間隔の開始時刻が繰り上げられ、終了時刻が繰り下げられます。次に例を示します。

```
SELECT application_id
FROM app_ids_stats_current_month
WHERE start_time_sec BETWEEN UNIX_TIMESTAMP("2011-12-10 12:59:00") and
UNIX_TIMESTAMP("2011-12-10 16:28:00");
```

これは次のように変更されます。

```
SELECT application_id
FROM app_ids_stats_current_month
WHERE start_time_sec BETWEEN UNIX_TIMESTAMP("2011-12-10 13:00:00") and
UNIX_TIMESTAMP("2011/12/12 16:00:00");
```

クエリ間隔がテーブルの時間枠を超えて延長される場合、通常は別のテーブルから追加データを取得できますが、その別のテーブルのデータの精度がより粗いことがあります。たとえば、過去 2 日間の帯域幅使用状況を取得するには、**current_day** テーブル（精度は 5 分）から昨日分の結果を取得できますが、**current_month**（時間単位）または **current_year**（日単位）から昨日のみの統計情報を取得することもできます。

app_ids_stats_current_timeframe

app_ids_stats_current_*timeframe* テーブルには、モニタ対象ネットワークのアプリケーション アクティビティと侵入イベントに関する統計情報が格納されます。統計情報は、検出されるアプリケーション、アプリケーション タイプ（アプリケーション プロトコル、クライアント アプリケーション、または Web アプリケーション）、およびアプリケーションのリスクとビジネスとの関連度に基づいて抽出できます。これらのテーブルでは、侵入ポリシー違反が原因でブロックされた接続と、推定される侵入の影響も追跡されます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

app_ids_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- app_ids_stats_current_timeframe のフィールド（5-5 ページ）
- app_ids_stats_current_timeframe の結合（5-6 ページ）
- app_ids_stats_current_timeframe のサンプル クエリ（5-6 ページ）

app_ids_stats_current_timeframe のフィールド

次の表に、**app_ids_stats_current_timeframe***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-3　　app_ids_stats_current_timeframe のフィールド

フィールド	説明
application_id	アプリケーションの内部識別番号。
application_name	ユーザ インターフェイスに表示されるアプリケーション名。
blocked	侵入ポリシーの違反が原因でブロックされた接続の数。
business_relevance	ビジネスの生産性に対するアプリケーションの関連度のインデックス（1 ～ 5）。1 は非常に低く、5 は非常に高いことを示します。
business_relevance_description	ビジネスとの関連度の説明（very low、low、medium、high、very high）。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
impact_level_1	アプリケーションについて記録された影響レベル 1（脆弱）の侵入イベントの数。
impact_level_2	影響レベル 2（脆弱な可能性あり）の侵入イベントの数。
impact_level_3	影響レベル 3（ホストは現在脆弱ではない）の侵入イベントの数
impact_level_4	影響レベル 4（ターゲットが不明）の侵入イベントの数。
impact_level_5	影響レベル 5（不明な脆弱性）の侵入イベントの数。
is_client_application	検出されたアプリケーションがクライアント アプリケーションであるかどうかを示す true/false フラグ。
is_server_application	検出されたアプリケーションがアプリケーション プロトコルであるかどうかを示す true/false フラグ。
is_web_application	検出されたアプリケーションが Web アプリケーションであるかどうかを示す true/false フラグ。
netmap_num	統計情報が収集されたドメインの Netmap ID。
risk	アプリケーションの推定リスクのインデックス（1 ～ 5）。1 は非常に低いリスク、5 は重大なリスクを示します。
risk_description	推定リスクの説明（very low、low、medium、high、critical）。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	イベントを提供したデバイスの ID。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。
would_have_dropped	侵入ポリシーがインライン型展開でパケットをドロップするように設定されている場合にドロップされるパケットの数。

app_ids_stats_current_timeframe の結合

次の表に、**app_ids_stats_current_timeframe***timeframe* テーブルで実行できる結合について説明します。

表 5-4　　app_ids_stats_current_timeframe の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_id	application_info.application_id application_host_map.application_id application_tag_map.application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id

app_ids_stats_current_timeframe のサンプル クエリ

次のクエリは、**app_ids_stats_current_month** テーブルから最大 25 件のアプリケーション レコードを返します。各レコードには、特定の時間間隔におけるアプリケーションのブロックされた接続数と侵入イベント数が含まれます。

```
SELECT from_unixtime(start_time_sec), sum(blocked)

FROM app_ids_stats_current_day

WHERE start_time_sec = unix_timestamp("2013-12-15");
```

app_stats_current_timeframe

app_stats_current_*timeframe* テーブルには、アプリケーション別、およびトラフィックをモニタするデバイス別の帯域幅使用状況とアクセス コントロール アクション(接続の許可または拒否)に関する統計情報が格納されます。これらの統計情報は、アプリケーションのビジネスとの関連度、推定リスク、およびタイプに基づいてフィルタリングできます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性(5-3 ページ)を参照してください。

app_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

app_stats_current_timeframe のフィールド

次の表に、**app_stats_current_timeframe***timeframe* テーブルでアクセスできるフィールドについて説明します。

表 5-5　app_stats_current_timeframe のフィールド

フィールド	説明
application_id	アプリケーションの内部識別番号。
application_name	ユーザ インターフェイスに表示されるアプリケーション名。
business_relevance	ビジネスの生産性に対するアプリケーションの関連度のインデックス（1 ～ 5）。1 は非常に低く、5 は非常に高いことを示します。
business_relevance_description	ビジネスとの関連度の説明（very low、low、medium、high、very high）。
bypass	遅延が原因でのバイパスが可能なパケットの数。
bytes_in	指定された期間中のアプリケーションの着信トラフィックのバイト数。
bytes_out	指定された期間中のアプリケーションの発信トラフィックのバイト数。
connections_allowed	許可された接続の数。
connections_denied	アクセス コントロール ポリシーの違反が原因で拒否された接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
is_client_application	検出されたアプリケーションがクライアント アプリケーションであるかどうかを示す true/false フラグ。
is_server_application	検出されたアプリケーションがアプリケーション プロトコルであるかどうかを示す true/false フラグ。
netmap_num	統計情報が収集されたドメインの Netmap ID。
is_web_application	検出されたアプリケーションが Web アプリケーションであるかどうかを示す true/false フラグ。
risk	アプリケーションの推定リスクのインデックス（1 ～ 5）。1 は非常に低いリスク、5 は重大なリスクを示します。
risk_description	推定リスクの説明（very low、low、medium、high、critical）。
sensor_address	トラフィックをモニタしていた管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
sensor_id	トラフィックを検出した管理対象デバイスの内部識別番号。
sensor_name	トラフィックを検出した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔の開始時刻を示す UNIX タイムスタンプ。開始時刻の指定については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。
would_bypass	バイパス対象であったが検査されたパケットの数。

app_stats_current_timeframe の結合

次の表に、**app_stats_current_***timeframe* テーブルで実行できる結合について説明します。

表 5-6　app_stats_current_timeframe の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_id	application_info.application_id application_host_map.application_id application_tag_map.application_id rna_host_service.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id

app_stats_current_timeframe のサンプル クエリ

次のクエリは、Firepower Management Center に接続しているすべての管理対象デバイスで、1 日間においてビジネスの関連度が低くリスクが高いアプリケーションに関連付けられている着信トラフィック ロードおよび発信トラフィック ロードを返します。

```
SELECT start_time_sec, sum(bytes_in), sum(bytes_out)

FROM app_stats_current_day

WHERE business_relevance <= 2

AND risk >= 4 AND start_time_sec = unix_timestamp("2013-12-15");
```

compliance_events_stats_current_timeframe

compliance_stats_events_current_*timeframe* テーブルには、特定の時間枠におけるコンプライアンスおよびホワイト リスト イベントの数に関する統計情報が格納されます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性 (5-3 ページ) を参照してください。

compliance_events_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- compliance_events_stats_current_timeframe のフィールド (5-9 ページ)
- compliance_event_stats_current_timeframe の結合 (5-9 ページ)
- compliance_event_stats_current_timeframe のサンプル クエリ (5-9 ページ)

compliance_events_stats_current_timeframe のフィールド

次の表に、**compliance_events_stats_current_*timeframe*** テーブルでアクセスできるフィールドについて説明します。

表 5-7　compliance_events_stats_current_timeframe のフィールド

フィールド	説明
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
netmap_num	統計情報が収集されたドメインの Netmap ID。
priority_0_events	特定の時間枠内に検出されたプライオリティ 0 のイベント。
priority_1_events	特定の時間枠内に検出されたプライオリティ 1 のイベント。
priority_2_events	特定の時間枠内に検出されたプライオリティ 2 のイベント。
priority_3_events	特定の時間枠内に検出されたプライオリティ 3 のイベント。
priority_4_events	特定の時間枠内に検出されたプライオリティ 4 のイベント。
priority_5_events	特定の時間枠内に検出されたプライオリティ 5 のイベント。
rule	イベントをトリガーしたホワイト リストのルール。このルールが空の場合、イベントはコンプライアンス イベントです。
start_time_sec	測定間隔の開始時刻を示す UNIX タイムスタンプ。開始時刻の指定については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。

compliance_event_stats_current_timeframe の結合

compliance_event_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

compliance_event_stats_current_timeframe のサンプル クエリ

次のクエリは、1 日間におけるプライオリティ 0、1、および 2 のイベントと、関連するホワイト リスト ルールを、ドメインを基準に並べ替えて返します。

```
SELECT domain_name, priority_0_events, priority_1_events, priority_2_events, rule

FROM compliance_event_stats_current_day

ORDER BY domain_name DESC;
```

dns_query_stats_current_timeframe

dns_query_stats_current_*timeframe* テーブルには、DNS クエリの統計情報が格納されます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

dns_query_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- dns_query_stats_current_timeframe のフィールド (5-10 ページ)
- dns_query_stats_current_timeframe の結合 (5-10 ページ)
- dns_query_stats_current_timeframe のサンプル クエリ (5-10 ページ)

dns_query_stats_current_timeframe のフィールド

次の表に、**dns_quaery_stats_current_*timeframe*** テーブルでアクセスできるフィールドについて説明します。

表 5-8　　dns_query_stats_current_timeframe のフィールド

フィールド	説明
bytes_in	指定された期間中の着信トラフィックのバイト数。
bytes_out	指定された期間中の発信トラフィックのバイト数。
connections_allowed	指定された DNS クエリで許可された接続の数。
connections_denied	アクセス コントロール ポリシー違反が原因で、指定された DNS クエリで拒否された接続の数。
dns_record_type	DNS クエリで使用される DNS ルックアップのタイプ。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
sensor_addres	トラフィックをモニタしていた管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
sensor_id	トラフィックを検出した管理対象デバイスの内部識別番号。
sensor_name	トラフィックを検出した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。
start_time_sec	測定間隔の開始時刻を示す UNIX タイムスタンプ。開始時刻の指定については、統計情報テーブルの照会時の時間間隔の指定 (5-3 ページ) を参照してください。

dns_query_stats_current_timeframe の結合

dns_query_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

dns_query_stats_current_timeframe のサンプル クエリ

次のクエリは、1 日間における各センサーの DNS レコード タイプに関連付けられている接続の数を、センサー名でソートし、ドメインを Global \ Company B \ Edge に限定して返します。

```
SELECT sensor_name, dns_record_type, sum(connections_allowed), sum(connections_denied)
FROM dns_query_stats_current_day
ORDER BY sensor_name DESC
WHERE domain_name= "Global \ Company B \ Edge";
```

geolocation_stats_current_timeframe

geolocation_stats_*timeframe* テーブルには、ロケーション レベルに基づく侵入イベントに関する統計情報が格納されます。統計情報は、影響レベル、デバイス、およびパケットの処理方法に基づいて抽出できます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

geolocation_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- geolocation_stats_current_timeframe のフィールド（5-11 ページ）
- geolocation_stats_current_timeframe の結合（5-12 ページ）
- geolocation_stats_current_timeframe のサンプル クエリ（5-12 ページ）

geolocation_stats_current_timeframe のフィールド

次の表に、**geolocation_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-9　　geolocation_stats_current_timeframe のフィールド

フィールド	説明
bytes_from	セッション レスポンダが送信した合計バイト数。
bytes_to	セッション イニシエータが送信した合計バイト数。
destination_continent	宛先ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
destination_country	宛先ホストの国のコード。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
flows_allowed	許可されたフローの数。
flows_denied	アクセス コントロール ポリシーの違反が原因で拒否されたフローの数。
netmap_num	統計情報が収集されたドメインの Netmap ID。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。

表 5-9　geolocation_stats_current_timeframe のフィールド(続き)

フィールド	説明
sensor_id	イベントを提供したデバイスの ID。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。
source_continent	送信元ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
source_country	送信元ホストの国のコード。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定(5-3 ページ)を参照してください。

geolocation_stats_current_timeframe の結合

geolocation_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

geolocation_stats_current_timeframe のサンプル クエリ

次のクエリは、当日における最初の 25 件のアジアからの接続イベントの送信元の国とセンサー名を、ドメインを Global \ Company B \ Edge に限定して返します。

```
SELECT sensor_name, source_continent
FROM geolocation_stats_current_year
WHERE destination_continent='as' and domain_name= "Global \ Company B \ Edge"
LIMIT 20;
```

ids_impact_stats_current_timeframe

ids_impact_stats_*timeframe* テーブルには、影響レベルに基づく侵入イベントに関する統計情報が格納されます。統計情報は、影響レベル、デバイス、およびパケットの処理方法に基づいて抽出できます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性(5-3 ページ)を参照してください。

ids_impact_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- ids_impact_stats_current_timeframe のフィールド（5-13 ページ）
- ids_impact_stats_current_timeframe の結合（5-13 ページ）
- ids_impact_stats_current_timeframe のサンプル クエリ（5-14 ページ）

ids_impact_stats_current_timeframe のフィールド

次の表に、**ids_impact_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-10　ids_impact_stats_current_timeframe のフィールド

フィールド	説明
blocked	侵入ポリシーの違反が原因でブロックされた接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
impact_level_1	アプリケーションについて記録された影響レベル 1（脆弱）の侵入イベントの数。
impact_level_2	影響レベル 2（脆弱な可能性あり）の侵入イベントの数。
impact_level_3	影響レベル 3（ホストは現在脆弱ではない）の侵入イベントの数
impact_level_4	影響レベル 4（ターゲットが不明）の侵入イベントの数。
impact_level_5	影響レベル 5（不明な脆弱性）の侵入イベントの数。
netmap_num	統計情報が収集されたドメインの Netmap ID。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	イベントを提供したデバイスの ID。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。
would_have_dropped	侵入ポリシーがインライン型展開でパケットをドロップするように設定されている場合にドロップされるパケットの数。

ids_impact_stats_current_timeframe の結合

ids_impact_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

ids_impact_stats_current_timeframe のサンプル クエリ

次のクエリは、当日における最初の 25 件の `blocked` イベントと `would_have_dropped` イベントを、ドメインを `Global \ Company B \ Edge` に限定して返します。

```
SELECT blocked, would_have_dropped

FROM ids_impact_stats_current_year

WHERE domain_name= "Global \ Company B \ Edge"

LIMIT 25;
```

ip_reputation_stats_current_timeframe

ip_category_stats_current_*timeframe* テーブルには、指定されたセキュリティ インテリジェンス カテゴリの IP アドレス、URL、および DNS ドメインへの要求に関連付けられている帯域幅使用状況と接続に関する統計情報が格納されます。クエリの対象を、トラフィックをモニタしていた管理対象デバイスに制限することもできます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

ids_impact_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- ip_reputation_stats_current_timeframe のフィールド（5-14 ページ）
- ip_reputation_stats_current_timeframe の結合（5-15 ページ）
- ip_reputation_stats_current_timeframe のサンプル クエリ（5-15 ページ）

ip_reputation_stats_current_timeframe のフィールド

次の表に、**ip_reputation_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-11　ip_reputation_stats_current_timeframe のフィールド

フィールド	説明
bytes_in	指定された期間中の着信トラフィックのバイト数。
bytes_out	指定された期間中の発信トラフィックのバイト数。
connections_allowed	指定された IP で許可された接続の数。
connections_denied	アクセス コントロール ポリシー違反が原因で、指定された IP で拒否された接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
name	セキュリティ インテリジェンスの名前（例:「URL Malware」）。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	イベントを提供したデバイスの ID。

表 5-11　ip_reputation_stats_current_timeframe のフィールド（続き）

フィールド	説明
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。
type	エントリの情報のタイプ。有効な値は次のとおりです。 0：ネットワーク インテリジェンス統計情報。 1：DNS セキュリティ インテリジェンス統計情報。 2：URL セキュリティ インテリジェンス統計情報。

ip_reputation_stats_current_timeframe の結合

ip_reputation_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

ip_reputation_stats_current_timeframe のサンプル クエリ

次のクエリは、当日における最初の 25 件の接続（受信バイト数と送信バイト数、接続数、接続のタイプ、およびセンサーを表示）を、ドメインを Global \ Company B \ Edge に限定し、ドメインを基準に並べ替えて返します。

```
SELECT uuid_btoa(domain_uuid), domain_name, type, name, bytes_in, bytes_out,
connections_allowed, connections_denied, sensor_name
FROM ip_reputation_stats_current_day
ORDER BY domain_name DESC
WHERE domain_name= "Global \ Company B \ Edge";
LIMIT 25;
```

session_stats_current_timeframe

session_stats_*timeframe* テーブルには、すべての接続の統計情報が格納されます。統計情報は、バイト数、接続、センサー、および時刻に基づいて抽出できます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

session_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- session_stats_current_timeframe のフィールド（5-16 ページ）
- session_stats_current_timeframe の結合（5-16 ページ）
- session_stats_current_timeframe のサンプル クエリ（5-16 ページ）

session_stats_current_timeframe のフィールド

次の表に、**session_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-12　session_stats_current_timeframe のフィールド

フィールド	説明
bytes_in	指定された期間中の着信トラフィックのバイト数。
bytes_out	指定された期間中の発信トラフィックのバイト数。
connections_allowed	指定された URL カテゴリで許可された接続の数。
connections_denied	アクセス コントロール ポリシー違反が原因で、指定された URL カテゴリで拒否された接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
id	このフィールドは使用されず、常に 0 を返します。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	イベントを提供したデバイスの ID。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。

session_stats_current_timeframe の結合

session_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

session_stats_current_timeframe のサンプル クエリ

次のクエリは、当日における各センサーで拒否された接続と許可された接続の数を、sensor_name の降順で、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT sensor_name, sensor_id connections_denied, connections_allowed
FROM session_stats_current_day
ORDER BY sensor_name DESC
WHERE domain_name= "Global \ Company B \ Edge";
```

ssl_stats_current_timeframe

ssl_stats_current_*timeframe* テーブルには、SSL 接続の統計情報が格納されます。統計情報は、バイト数、接続、センサー、および時刻に基づいて抽出できます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

ssl_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- ssl_stats_current_timeframe のフィールド（5-17 ページ）
- ssl_stats_current_timeframe の結合（5-19 ページ）
- ssl_stats_current_timeframe のサンプル クエリ（5-19 ページ）

ssl_stats_current_timeframe のフィールド

次の表に、**ssl_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-13　ssl_stats_current_timeframe のフィールド

フィールド	説明
block	リセットなしでドロップされた SSL セッションの数。
block_with_reset	リセットありでドロップされた SSL セッションの数。
cached_session	セッション キャッシュ内で見つかった SSL セッションの数。
cannot_determine_verdict	SSL ルールの評価中に発生したハンドシェイク エラーの数。
cert_expired	証明書が期限切れであった SSL セッションの数。
cert_invalid_issuer	証明書発行元が無効であったか、または Trusted CA リストで見つからなかった SSL セッションの数。
cert_invalid_signature	証明書に無効なシグニチャが含まれていた SSL セッションの数。
cert_not_checked	証明書が検査されなかった SSL セッションの数。
cert_not_yet_valid	証明書が有効ではなかった SSL セッションの数。
cert_revoked	証明書が失効していた SSL セッションの数。
cert_self_signed	証明書が自己署名されていた SSL セッションの数。
cert_unknown	証明書ステータスが不明だった SSL セッションの数。
cert_valid	証明書が有効だった SSL セッションの数。
cert_validation_cache_hit	証明書が検証キャッシュ内で見つかった回数。
cert_validation_cache_miss	証明書が検証キャッシュ内で見つからなかった回数。
decrypt_resign_self_signed	自己署名証明書を使用する SSL セッションが復号化/再署名方式を使用して復号化された回数。
decrypt_resign_self_signed_replace_key_only	自己署名証明書を使用する SSL セッションが、キー置換のみの複合化/再署名方式を使用して復号化された回数。
decrypt_resign_signed_cert	自己署名証明書を使用する SSL セッションが復号化/再署名方式を使用して復号化された回数。
decrypt_with_known_key	SSL セッションが既知のキー方式を使用して復号化された回数。

表 5-13　ssl_stats_current_timeframe のフィールド（続き）

フィールド	説明
decryption_error	復号化中にエラーが発生した SSL セッションの数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
do_not_decrypt	SSL セッションが検出されたが復号化されなかった回数。
handshake_error	SSL ルールの評価前に発生したハンドシェイク エラーの数。
netmap_num	統計情報が収集されたドメインの Netmap ID。
orig_cert_cache_hit	元の証明書がキャッシュ内で見つかった回数。
orig_cert_cache_miss	元の証明書がキャッシュ内で見つからなかった回数。
resigned_cert_cache_hit	再署名証明書がキャッシュ内で見つかった回数。
resigned_cert_cache_miss	再署名証明書がキャッシュ内で見つからなかった回数。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
sensor_id	イベントを提供したデバイスの ID。
sensor_name	イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
session_cache_hit	SSL セッション ID またはチケットがキャッシュで見つかった回数。
session_cache_miss	SSL セッション ID またはチケットがキャッシュで見つからなかった回数。
session_incorrectly_identified_as_ssl	SSL を使用するものとして誤って指定されていたセッションの数。
ssl_compression	SSL 圧縮を使用したセッションの数。
ssl_sessions_decrypted	正常に復号化された SSL セッションの数。
ssl_sessions_not_decrypted	正常に復号化されなかった SSL セッションの数。
ssl_sessions_reused_by_id	SSL セッションが ID を再利用した回数。
ssl_sessions_reused_by_ticket	SSL セッションがチケットを再利用した回数。
ssl_sessions_with_errors	エラーが発生した SSL セッションの数。
ssl_v20	SSL バージョン 2.0 を使用する SSL セッションの数。
ssl_v30	SSL バージョン 3.0 を使用する SSL セッションの数。
ssl_version_unknown	不明なバージョンの SSL を使用する SSL セッションの数。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。
tls_v10	TLS バージョン 1.0 を使用する SSL セッションの数。
tls_v11	TLS バージョン 1.1 を使用する SSL セッションの数。
tls_v12	TLS バージョン 1.2 を使用する SSL セッションの数。
total_ssl_sessions	検出された SSL セッションの総数。
uncached_session	ID またはチケットのキャッシュ ミスにより復号化できなかった回数。

表 5-13　ssl_stats_current_timeframe のフィールド（続き）

フィールド	説明
undecryptable_in_passive_mode	デバイスがパッシブ モードであるために復号化できなかった SSL セッションの数。
unknown_cipher_suite	不明な暗号スイートを使用する SSL セッションの数。
unsupported_cipher_suite	既知であるがサポートされていない暗号スイートを使用する SSL セッションの数。

ssl_stats_current_timeframe の結合

ssl_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

ssl_stats_current_timeframe のサンプル クエリ

次のクエリは、当日における各センサーの SSL セッションの数、復号化されたセッションの数、復号化されなかったセッションの数、パッシブ モードで復号化できなかったセッションの数を、sensor_name の降順で、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT sensor_name, total_ssl_sessions, ssl_sessions_decrypted,
ssl_sessions_not_decrypted, undecryptable_in_passive_mode
FROM ssl_stats_current_day
WHERE domain_name= "Global \ Company B \ Edge"
ORDER BY sensor_name DESC;
```

storage_stats_by_disposition_current_timeframe

storage_stats_by_disposition_*timeframe* テーブルには、保存ファイルの統計情報が格納されます。統計情報は、バイト数、接続、センサー、および時刻に基づいて抽出できます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

storage_stats_by_disposition_*timeframe* テーブルの詳細については、次に示す項を参照してください。

storage_stats_by_disposition_current_timeframe のフィールド

次の表に、**storage_stats_by_disposition_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-14　storage_stats_by_disposition_current_timeframe のフィールド

フィールド	説明
bytes_written	ファイルのサイズ(バイト単位)。
disposition	ファイルのマルウェア ステータス。有効な値は次のとおりです。 • CLEAN:ファイルはクリーンであり、マルウェアが含まれていない。 • UNKNOWN:ファイルにマルウェアが含まれているかどうかが不明である。 • MALWARE:ファイルにマルウェアが含まれている。 • UNAVAILABLE:ソフトウェアから シスコ クラウドに対して、特性を確認する要求を送信できなかったか、または シスコ クラウド サービスが要求に応答しなかった。 • CUSTOM SIGNATURE:ファイルがユーザ定義のハッシュと一致するため、ユーザが指定した方法で処理された。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
netmap_num	統計情報が収集されたドメインの Netmap ID。
number_dropped	ドロップされたこの特性のファイルの数。
number_stored	保存されたこの特性のファイルの数。
sensor	ファイルを検出したデバイスの ID。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定(5-3 ページ)を参照してください。

storage_stats_by_disposition_current_timeframe の結合

session_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

storage_stats_by_disposition_current_timeframe のサンプル クエリ

次のクエリは、当日における各センサーでドロップされたファイルの数と保存されたファイルの数を、sensor_name の降順で、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT sensor_name, number_dropped, number_stored
FROM storage_stats_by_disposition_current_day
WHERE domain_name= "Global \ Company B \ Edge"
ORDER BY sensor_name DESC;
```

storage_stats_by_file_type_current_timeframe

storage_stats_by_file_type_current_*timeframe* テーブルには、ファイル タイプ別の保存ファイルの統計情報が格納されます。統計情報は、バイト数、接続、センサー、および時刻に基づいて抽出できます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

storage_stats_by_file_type_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

storage_stats_by_file_type_current_timeframe のフィールド

次の表に、**storage_stats_by_file_type_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-15　storage_stats_by_file_type_current_timeframe のフィールド

フィールド	説明
bytes_written	ファイルのサイズ（バイト単位）。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
file_type	検出または検疫されたファイルのファイル タイプ。
file_type_id	ファイル タイプにマップされている ID 番号。
netmap_num	統計情報が収集されたドメインの Netmap ID。
number_dropped	ドロップされたこのタイプのファイルの数。
number_stored	保存されたこのタイプのファイルの数。
sensor	ファイルを検出したデバイスの ID。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。

storage_stats_by_file_type_current_timeframe の結合

session_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

storage_stats_by_file_type_current_timeframe のサンプル クエリ

次のクエリは、当日における各センサーでドロップされたファイルの数と保存されたファイルの数を、file_type の降順で、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT sensor_name, number_dropped, number_stored, file_type
FROM storage_stats_by_file_type_current_day
WHERE domain_name= "Global \ Company B \ Edge"
ORDER BY file_type DESC;
```

transmission_stats_by_file_type_current_timeframe

transmission_stats_by_file_type_current_*timeframe* テーブルには、ファイル タイプ別の保存ファイルの統計情報が格納されます。統計情報は、バイト数、接続、センサー、および時刻に基づいて抽出できます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

transmission_stats_by_file_type_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

- transmission_stats_by_file_type_current_timeframe のフィールド（5-22 ページ）
- transmission_stats_by_file_type_current_timeframe の結合（5-23 ページ）
- transmission_stats_by_file_type_current_timeframe のサンプル クエリ（5-23 ページ）

transmission_stats_by_file_type_current_timeframe のフィールド

次の表に、**transmission_stats_by_file_type_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。このタイプのすべてのテーブルには同じフィールドが含まれています。

表 5-16　transmission_stats_by_file_type_current_timeframe のフィールド

フィールド	説明
bytes_sent	送信バイト数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
file_type	検出または検疫されたファイルのファイル タイプ。
file_type_id	ファイル タイプにマップされている ID 番号。
netmap_num	統計情報が収集されたドメインの Netmap ID。
number_dropped	ドロップされたこのタイプのファイルの数。
number_sent	送信されたこのタイプのファイルの数。
sensor	ファイルを検出したデバイスの ID。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。

表 5-16　transmission_stats_by_file_type_current_timeframe のフィールド（続き）

フィールド	説明
sensor_name	侵入イベントを生成した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔開始日時を示す UNIX タイムスタンプ。詳細については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。

transmission_stats_by_file_type_current_timeframe の結合

transmission_stats_by_file_type_current_timeframe_ テーブルに対して結合を実行することはできません。

transmission_stats_by_file_type_current_timeframe のサンプル クエリ

次のクエリは、当日における各センサーでドロップされた接続の数と送信された接続の数を、file_type の降順で、Global \ Company B \ Edge ドメインに限定して返します。

```
SELECT sensor_name, number_dropped, number_sent, file_type
FROM transmission_stats_by_file_type_current_day
WHERE domain_name= "Global \ Company B \ Edge"
ORDER BY file_type DESC;
```

url_category_stats_current_timeframe

url_category_stats_current_timeframe_ テーブルには、指定された URL カテゴリの URL に対する要求に関連付けられている帯域幅使用状況と接続に関する統計情報が格納されます。クエリの対象を、トラフィックをモニタしていた管理対象デバイスに制限することもできます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

url_category_stats_current_timeframe_ テーブルの詳細については、次に示す項を参照してください。

url_category_stats_current_timeframe のフィールド

次の表に、**url_category_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。

表 5-17　url_category_stats_current_timeframe のフィールド

フィールド	説明
bytes_in	指定された期間中の着信トラフィックのバイト数。
bytes_out	指定された期間中の発信トラフィックのバイト数。
category	URL のカテゴリ。
connections_allowed	指定された URL カテゴリで許可された接続の数。
connections_denied	アクセス コントロール ポリシー違反が原因で、指定された URL カテゴリで拒否された接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
netmap_num	統計情報が収集されたドメインの Netmap ID。
sensor_address	トラフィックをモニタしていた管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	トラフィックを検出した管理対象デバイスの内部識別番号。
sensor_name	トラフィックをモニタする管理対象デバイス。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。
start_time_sec	測定間隔の開始時刻を示す UNIX タイムスタンプ。開始時刻の指定については、統計情報テーブルの照会時の時間間隔の指定(5-3 ページ)を参照してください。

url_category_stats_current_timeframe の結合

url_category_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

url_category_stats_current_timeframe のサンプル クエリ

次のクエリは最大 25 件の URL カテゴリ レコードを返します。各レコードには、指定された間隔における関連する着信トラフィックと発信トラフィックのバイト数、および許可された接続と拒否された接続が含まれています。このクエリは Games カテゴリと Global \ Company B \ Edge ドメインに限定されます。

```
SELECT category, sensor_name, sensor_address, start_time_sec, bytes_in, bytes_out,
connections_allowed, connections_denied

FROM url_category_stats_current_year

WHERE category="Games" AND domain_name= "Global \ Company B \ Edge"

LIMIT 0, 25;
```

url_reputation_stats_current_timeframe

url_reputation_stats_current_*timeframe* テーブルには、指定されたレピュテーションの URL に対する要求に関連付けられている帯域幅使用状況と接続に関する統計情報が格納されます。クエリ結果を、トラフィックをモニタしていた管理対象デバイスに制限することもできます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

url_reputation_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

url_reputation_stats_current_timeframe のフィールド

次の表に、**url_reputation_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。

表 5-18　url_reputation_stats_current_timeframe のフィールド

フィールド	説明
bytes_in	指定された期間中の着信トラフィックのバイト数。
bytes_out	指定された期間中の発信トラフィックのバイト数。
connections_allowed	許可された接続の数。
connections_denied	アクセス コントロール ポリシーの違反が原因で拒否された接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
netmap_num	統計情報が収集されたドメインの Netmap ID。
reputation	要求された URL に関連付けられているリスク。次のいずれかが必要です。 • High risk • Suspicious site • Benign site with security risks • Benign site • Well known • Risk unknown
sensor_address	トラフィックをモニタしていた管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	トラフィックをモニタしていた管理対象デバイスの内部識別番号。
sensor_name	トラフィックをモニタしていた管理対象デバイスの名前。

表 5-18　url_reputation_stats_current_timeframe のフィールド（続き）

フィールド	説明
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔の開始時刻を示す UNIX タイムスタンプ。開始時刻の指定については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。

url_reputation_stats_current_timeframe の結合

url_reputation_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

url_reputation_stats_current_timeframe のサンプル クエリ

次のクエリは、**url_reputation_stats_current_month** テーブルから最大 25 件の URL レピュテーション レコードを返します。各レコードには、測定間隔中の着信トラフィックと発信トラフィックのバイト数、および許可された接続と拒否された接続が含まれています。このクエリは High risk レピュテーションと Global \ Company B \ Edge ドメインに限定されます。

```
SELECT sensor_name, sensor_address, reputation, start_time_sec, bytes_in, bytes_out,
connections_allowed, connections_denied
FROM url_reputation_stats_current_year
WHERE reputation="High risk" AND domain_name= "Global \ Company B \ Edge"
LIMIT 0, 25;
```

user_ids_stats_current_timeframe

user_ids_stats_current_*timeframe* テーブルは、ユーザ別のアクセス フィルタリングの統計情報と影響の統計情報が格納されるラウンドロビン テーブルです。

このタイプの **current_day**、**current_month**、および **current_year** テーブルについては、統計情報追跡テーブルの保存特性（5-3 ページ）を参照してください。

ラウンド ロビン統計情報テーブルの使用法については、統計情報追跡テーブルについて（5-2 ページ）を参照してください。

user_ids_stats_current_*timeframe* テーブルの詳細については、次に示す項を参照してください。

user_ids_stats_current_timeframe のフィールド

次の表に、**user_ids_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。

表 5-19　　user_ids_stats_current_timeframe のフィールド

フィールド	説明
blocked	侵入ポリシーの違反が原因でブロックされた接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
impact_level_1	ユーザについて記録された影響レベル 1（脆弱）の侵入イベントの数。
impact_level_2	ユーザについて記録された影響レベル 2（脆弱な可能性あり）の侵入イベントの数。
impact_level_3	ユーザについて記録された影響レベル 3（ホストは現在脆弱ではない）の侵入イベントの数。
impact_level_4	ユーザについて記録された影響レベル 4（不明なターゲット）の侵入イベントの数。
impact_level_5	ユーザについて記録された影響レベル 5（不明な脆弱性）の侵入イベントの数。
netmap_num	統計情報が収集されたドメインの Netmap ID。
sensor_address	トラフィックをモニタしていた管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	トラフィックを検出した管理対象デバイスの内部識別番号。
sensor_name	トラフィックを検出した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
start_time_sec	測定間隔の開始時刻を示す UNIX タイムスタンプ。開始時刻の指定については、統計情報テーブルの照会時の時間間隔の指定（5-3 ページ）を参照してください。
user_id	ホストの最終ログイン ユーザの内部識別番号。
username	ホストの最終ログイン ユーザのユーザ名。
would_have_dropped	侵入ポリシーがインライン型展開でパケットをドロップするように設定されている場合にドロップされるパケットの数。

user_ids_stats_current_timeframe の結合

user_ids_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

user_ids_stats_current_timeframe のサンプル クエリ

次のクエリは、**user_ids_stats_current_month** テーブルから最大 25 件のユーザ レコードを返します。各レコードには、Global \ Company B \ Edge ドメインでの選択された username のブロックされた接続の数と侵入イベントの数が含まれます。

```
SELECT username, start_time_sec, blocked, impact_level_1, impact_level_2,
impact_level_3, impact_level_4, impact_level_5 FROM user_ids_stats_current_year
WHERE username="username" AND domain_name= "Global \ Company B \ Edge"
LIMIT 0, 25;
```

user_stats_current_timeframe

user_stats_current_*timeframe* テーブルには、ユーザ別の帯域幅使用状況とアクセス コントロール アクション(接続の許可または拒否)に関する統計情報が格納されます。クエリの対象を、トラフィックをモニタしていた管理対象デバイスに制限することもできます。

current_day、**current_month**、および **current_year** 統計情報テーブルについては、統計情報追跡テーブルの保存特性(5-3 ページ)を参照してください。

詳細については、次の項を参照してください。

- user_stats_current_timeframe のフィールド(5-28 ページ)
- user_stats_current_timeframe の結合(5-29 ページ)
- user_stats_current_timeframe のサンプル クエリ(5-29 ページ)

user_stats_current_timeframe のフィールド

次の表に、**user_stats_current_***timeframe* テーブルでアクセスできるフィールドについて説明します。

表 5-20　user_stats_current_timeframe のフィールド

フィールド	説明
bytes_in	測定間隔におけるユーザの着信トラフィックのバイト数。
bytes_out	測定間隔におけるユーザの発信トラフィックのバイト数。
connections_allowed	測定間隔においてこのユーザに対して許可された接続数。
connections_denied	アクセス コントロール ポリシー違反が原因で、このユーザに対して拒否された接続の数。
domain_name	統計情報のために指定されたドメインの名前。
domain_uuid	統計情報のために指定されたドメインの UUID。これはバイナリで示されます。
netmap_num	統計情報が収集されたドメインの Netmap ID。
sensor_address	トラフィックをモニタしていた管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_id	トラフィックを検出した管理対象デバイスの内部識別番号。
sensor_name	トラフィックを検出した管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。
start_time_sec	測定間隔の開始時刻を示す UNIX タイムスタンプ。開始時刻の指定については、統計情報テーブルの照会時の時間間隔の指定(5-3 ページ)を参照してください。

表 5-20　user_stats_current_timeframe のフィールド（続き）

フィールド	説明
user_id	トラフィックを生成したホストの最終ログイン ユーザの内部識別番号。
username	トラフィックを生成したホストの最終ログイン ユーザのユーザ名。

user_stats_current_timeframe の結合

user_stats_current_*timeframe* テーブルに対して結合を実行することはできません。

user_stats_current_timeframe のサンプル クエリ

次のクエリは最大 25 件のユーザ レコードを返します。各レコードには、domain_name= "Global \ Company B \ Edge ドメイン内での測定間隔中の着信トラフィックと発信トラフィックのバイト数、および許可された接続と拒否された接続が含まれています。

```
SELECT sensor_name, sensor_address, username, start_time_sec, bytes_in, bytes_out,
connections_allowed, connections_denied
FROM user_stats_current_year
WHERE username="username" AND domain_name= "Global \ Company B \ Edge"
LIMIT 0, 25;
```

CHAPTER 6

スキーマ：検出イベントおよびネットワーク マップのテーブル

この章では、検出イベントと シスコ ネットワーク マップに関連するテーブルのスキーマとサポートされている結合について説明します。

Firepower システム は、ホストとネットワーク デバイスにより生成されるトラフィックをモニタし、継続的に検出イベントを生成します。

ネットワーク マップとは、検出イベントで報告されるネットワーク資産に関する情報のリポジトリです。ネットワーク マップには、検出された各ホストとネットワーク デバイスに関する情報（オペレーティング システム、サーバ、クライアント アプリケーション、ホスト属性、脆弱性など）が含まれています。

脆弱性は、ホストに被害を及ぼす可能性がある特定の侵害やエクスプロイトの記述です。シスコには独自の脆弱性データベース（VBD）があります。この VBD は Bugtraq データベースと MITRE の CVE データベースを相互参照します。また、ホスト入力機能を使用してサードパーティの脆弱性データをインポートできます。

ネットワーク マップでの特定のホストに関する情報は、ホストのタイプと、モニタ対象トラフィックで使用可能な情報によって異なる可能性があることに注意してください。

詳細については、次の表に示す項を参照してください。「バージョン」欄は、示されている各テーブルをサポートしている Firepower システム のバージョンを示します。廃止されたテーブルは製品の現行バージョンでも引き続きサポートされていますが、今後もサポートを引き続き受けるために、廃止されたテーブルとフィールドを使用しないでおくことをシスコ**強く**推奨します。

表 6-1　検出イベントおよびネットワーク マップのテーブルのスキーマ

参照先	次の内容が格納されるテーブル	バージョン
application_host_map（6-5 ページ）	モニタ対象ネットワークのホストで検出されたアプリケーション。	5.0+
application_ip_map（A-1 ページ）	モニタ対象ネットワークで検出されたアプリケーションに関連付けられているカテゴリ、タグ、生産性、およびリスク。	5.2+
application_ip_map（A-1 ページ）	モニタ対象ネットワークで検出されたアプリケーションに関連付けられているカテゴリ、タグ、生産性、およびリスク。 バージョン 5.2 で廃止置き換わるテーブル：application_ip_map（A-1 ページ）。	5.0-5.1.x

表 6-1　検出イベントおよびネットワーク マップのテーブルのスキーマ（続き）

参照先	次の内容が格納されるテーブル	バージョン
application_tag_map（6-9 ページ）	モニタ対象ネットワークで検出されたアプリケーションに関連付けられているタグ	5.0+
domain_control_information（6-11 ページ）	ドメイン階層情報	6.0+
network_discovery_event（6-12 ページ）	検出イベントおよびホスト入力イベント。	5.0+
rna_host（6-13 ページ）	モニタ対象ネットワークのホストの基本情報。	5.2+
rna_host_attribute（6-15 ページ）	モニタ対象ネットワークの各ホストに関連付けられているホスト属性。	5.2+
rna_host_client_app（6-17 ページ）	モニタ対象ネットワークのホストで検出されたクライアント アプリケーション。	5.2+
rna_host_client_app（6-17 ページ）	モニタ対象ネットワークのホストで検出された HTTP（Web ブラウザ）クライアント アプリケーションに関連付けられているペイロード。	5.2+
rna_host_ioc_state（6-22 ページ）	ホストの侵害状態が格納されます。	5.3+
rna_host_ip_map（6-26 ページ）	モニタ対象ネットワークのホストの MAC アドレスにホスト ID を相関付けます。	5.2+
rna_host_os（6-29 ページ）	モニタ対象ネットワークのホストで検出されたオペレーティング システム。	5.2+
rna_host_os_vulns（6-30 ページ）	モニタ対象ネットワークのホストに関連付けられている脆弱性。	5.2+
rna_host_protocol（6-32 ページ）	モニタ対象ネットワークのホストで検出されたプロトコル。	4.10.x+
rna_host_protocol（6-32 ページ）	プロトコルを検出した管理対象デバイスに関連するモニタ対象ネットワーク上のホスト。	5.2+
rna_host_service（6-35 ページ）	モニタ対象ネットワークのホストで検出されたサービス。	5.2+
rna_host_service_banner（6-37 ページ）	モニタ対象ネットワークのホストで検出されたサービスのサービス ベンダーとバージョン（「バナー」）をアドバタイズするネットワーク トラフィックのヘッダー。	5.2+
rna_host_service_info（6-38 ページ）	モニタ対象ネットワークのホストで検出されたサービスの詳細。	5.2+
rna_host_service_payload（6-42 ページ）	モニタ対象ネットワークのホストで検出されたサービスに関連付けられているペイロード。	5.2+
rna_host_service_subtype（6-44 ページ）	モニタ対象ネットワークのホストで検出されたサービスのサブ サービス。	5.2+
rna_host_service_vulns（6-46 ページ）	モニタ対象ネットワークのホストで検出されたサービスに関連付けられている脆弱性。	5.2+
rna_host_third_party_vuln（6-47 ページ）	モニタ対象ネットワークのホストに関連付けられているサードパーティの脆弱性。	5.2+

表 6-1　検出イベントおよびネットワーク マップのテーブルのスキーマ（続き）

参照先	次の内容が格納されるテーブル	バージョン
rna_host_third_party_vuln_bugtraq_id（6-49 ページ）	モニタ対象ネットワークのホストに関連付けられており、かつ Bugtraq データベース（http://www.securityfocus.com/bid/）の脆弱性にも関連付けられているサードパーティの脆弱性。	5.2+
rna_host_third_party_vuln_cve_id（6-50 ページ）	モニタ対象ネットワークのホストに関連付けられており、かつ MITRE の CVE データベースの脆弱性にも関連付けられているサードパーティの脆弱性。（http://www.cve.mitre.org/）。	5.2+
rna_host_third_party_vuln_rna_id（6-52 ページ）	モニタ対象ネットワークのホストに関連付けられており、かつ VDB データベースの脆弱性にも関連付けられているサードパーティの脆弱性。	5.2+
rna_ip_host（A-1 ページ）	モニタ対象ネットワークの IP ホストの基本情報。 バージョン 5.2 で廃止置き換わるテーブル rna_host（6-13 ページ）。	4.10.x-5.1.x
rna_ip_host_client_app（A-1 ページ）	モニタ対象ネットワークのホストで検出されたクライアント アプリケーション。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_client_app（6-17 ページ）。	4.10.x-5.1.x
rna_ip_host_client_app_payload（A-1 ページ）	モニタ対象ネットワークの IP ホストで検出された HTTP（Web ブラウザ）クライアント アプリケーションに関連付けられているペイロード。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_client_app（6-17 ページ）。	4.10.x-5.1.x
rna_ip_host_os（A-1 ページ）	モニタ対象ネットワークの IP ホストで検出されたオペレーティング システム。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_os（6-29 ページ）。	4.10.x-5.1.x
rna_ip_host_os_vulns（A-1 ページ）	モニタ対象ネットワークの IP ホストに関連付けられている脆弱性。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_os_vulns（6-30 ページ）。	4.10.x--5.1.x
rna_ip_host_sensor（A-1 ページ）	ホストを検出した管理対象デバイスに関連するモニタ対象ネットワーク上の IP ホスト。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_protocol（6-32 ページ）。	5.0-5.1.x
rna_ip_host_service（A-1 ページ）	モニタ対象ネットワークの IP ホストで検出されたサービス。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_service（6-35 ページ）。	4.10.x-5.1.x

表 6-1　検出イベントおよびネットワーク マップのテーブルのスキーマ（続き）

参照先	次の内容が格納されるテーブル	バージョン
rna_ip_host_service_banner（A-1 ページ）	モニタ対象ネットワークのホストで検出されたサービスのサービス ベンダーとバージョン（「バナー」）をアドバタイズするネットワーク トラフィックのヘッダー。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_service_banner（6-37 ページ）。	4.10.x-5.1.x
rna_ip_host_service_info（A-1 ページ）	モニタ対象ネットワークの IP ホストで検出されたサービスの詳細。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_service_info（6-38 ページ）。	4.10.x-5.1.x
rna_ip_host_service_payload（A-1 ページ）	モニタ対象ネットワークの IP ホストで検出されたサービスに関連付けられているペイロード。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_service_payload（6-42 ページ）。	4.10.x-5.1.x
rna_ip_host_service_subtype（A-1 ページ）	モニタ対象ネットワークの IP ホストで検出されたサービスのサブサービス。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_service_subtype（6-44 ページ）。	4.10.x-5.1.x
rna_ip_host_service_vulns（A-1 ページ）	モニタ対象ネットワークの IP ホストで検出されたサービスに関連付けられている脆弱性。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_service_vulns（6-46 ページ）。	4.10.x-5.1.x
rna_ip_host_third_party_vuln（A-1 ページ）	モニタ対象ネットワークの IP ホストに関連付けられているサードパーティの脆弱性。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_third_party_vuln（6-47 ページ）。	4.10.x-5.1.x
rna_ip_host_third_party_vuln_bugtraq_id（A-1 ページ）	モニタ対象ネットワークの IP ホストに関連付けられており、かつ Bugtraq データベース（`http://www.securityfocus.com/bid/`）の脆弱性にも関連付けられているサードパーティの脆弱性。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_third_party_vuln_bugtraq_id（6-49 ページ）。	4.10.x-5.1.x
rna_ip_host_third_party_vuln_cve_id（A-1 ページ）	モニタ対象ネットワークの IP ホストに関連付けられており、かつ MITRE の CVE データベースの脆弱性にも関連付けられているサードパーティの脆弱性。（`http://www.cve.mitre.org/`）。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_third_party_vuln_cve_id（6-50 ページ）。	4.10.x-5.1.x
rna_ip_host_third_party_vuln_rna_id（A-1 ページ）	モニタ対象ネットワークの IP ホストに関連付けられており、かつ VDB の脆弱性にも関連付けられているサードパーティの脆弱性。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_third_party_vuln_rna_id（6-52 ページ）。	4.10.x-5.1.x

表 6-1　検出イベントおよびネットワーク マップのテーブルのスキーマ（続き）

参照先	次の内容が格納されるテーブル	バージョン
rna_ip_host_user_history（A-1 ページ）	モニタ対象ネットワークの特定の IP ホストに対するユーザ アクティビティ。 バージョン 5.2 で廃止置き換わるテーブル：user_ipaddr_history（6-60 ページ）。	4.10.x-5.1.x
rna_mac_host（A-1 ページ）	モニタ対象ネットワークの MAC ホスト（IP アドレスを持たないホスト）。	4.10.x-5.1.x
rna_mac_host_sensor（A-2 ページ）	ホストを検出した管理対象デバイスに関連するモニタ対象ネットワーク上の IP ホスト。	5.0-5.1.x
rna_mac_ip_map（A-2 ページ）	モニタ対象ネットワークの IP ホストの MAC アドレス。 バージョン 5.2 で廃止置き換わるテーブル：rna_host_ip_map（6-26 ページ）および rna_host_mac_map（6-27 ページ）。	4.10.x-5.1.x
rna_vuln（6-54 ページ）	シスコ VDB の脆弱性。	4.10.x+
tag_info（6-57 ページ）	検出されたアプリケーションを特徴付けるタグ。	5.0+
url_categories（6-58 ページ）	モニタ対象ネットワークのホストからアクセスされた URL を特徴付けるカテゴリ。	5.0+
url_reputations（6-59 ページ）	モニタ対象ネットワークのホストからアクセスされた URL を特徴付けるレピュテーション。	5.0+
user_ipaddr_history（6-60 ページ）	モニタ対象ネットワークの特定のホストに対するユーザ アクティビティ。	5.2+

application_host_map

`application_host_map` テーブルには、ネットワークで検出された各アプリケーションに関連付けられているカテゴリとタグに関する情報が格納されます。

詳細については、次の項を参照してください。

- application_host_map のフィールド（6-6 ページ）
- application_host_map の結合（6-6 ページ）
- application_host_map のサンプル クエリ（6-7 ページ）

application_host_map のフィールド

次の表に、**application_host_map** テーブルでアクセスできるフィールドについて説明します。

表 6-2　application_host_map のフィールド

フィールド	説明
application_id	アプリケーションの内部識別番号。
application_name	ユーザ インターフェイスに表示されるアプリケーション名。
application_tag_id	このフィールドは廃止されており、null を返します。
business_relevance	ビジネスの生産性へのアプリケーションの関連度のインデックス(1 ～ 5)。1 は非常に低く、5 は非常に高いことを示します。
business_relevance_description	ビジネスとの関連度の説明(very low、low、medium、high、very high)。
host_id	ホストの ID 番号。
risk	アプリケーション リスクのインデックス(1 ～ 5)。1 は非常に低いリスク、5 は重大なリスクを示します。
risk_description	リスクの説明(very low、low、medium、high、critical)。

application_host_map の結合

次の表に、**application_host_map** テーブルで実行できる結合について説明します。

表 6-3　application_host_map の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_id	app_ids_stats_current_timeframe.application_id application_info.application_id application_tag_map.application_id app_stats_current_timeframe.application_id connection_log.application_protocol_id connection_log.client_application_id connection_log.web_application_id connection_summary.application_protocol_id file_event.application_id intrusion_event.application_protocol_id intrusion_event.client_application_id intrusion_event.web_application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id si_connection_log.application_protocol_name si_connection_log.client_application_id si_connection_log.web_application_id

表 6-3　application_host_map の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

application_host_map のサンプル クエリ

次のクエリは、host_id が 8 のホストで検出されたアプリケーションに関する情報を返します。

```
SELECT host_id, application_id, application_name, business_relevance, risk
FROM application_host_map
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

application_info

application_info テーブルには、モニタ対象ネットワークのホストで検出可能なアプリケーションに関する情報が含まれています。

アプリケーションに関連付けられているタグのリストを **application_tag_map** テーブルから取得するには **application_id** で結合します。同様に、アプリケーションの関連カテゴリのリストを **application_host_map** から取得するには **application_id** で結合します。

詳細については、次の項を参照してください。

- application_info のフィールド（6-8 ページ）
- application_info の結合（6-8 ページ）
- application_info のサンプル クエリ（6-9 ページ）

application_info のフィールド

次の表に、**application_info** テーブルでアクセスできるフィールドについて説明します。

表 6-4　application_info のフィールド

フィールド	説明
application_description	アプリケーションの説明。
application_id	アプリケーションの内部識別番号。
application_name	ユーザ インターフェイスに表示されるアプリケーション名。
business_relevance	ビジネスの生産性に対するアプリケーションの関連度のインデックス(1 ～ 5)。1 は非常に低く、5 は非常に高いことを示します。
business_relevance_description	ビジネスとの関連度の説明(very low、low、medium、high、very high)。
domain_name	アプリケーションが検出されたドメインの名前。
domain_uuid	アプリケーションが検出されたドメインの UUID。これはバイナリで示されます。
is_client_application	検出されたアプリケーションがクライアントであるかどうかを示す true/false フラグ。
is_server_application	検出されたアプリケーションがサーバ アプリケーションであるかどうかを示す true/false フラグ。
is_web_application	検出されたアプリケーションが Web アプリケーションであるかどうかを示す true/false フラグ。
risk	アプリケーションの推定リスクのインデックス(1 ～ 5)。1 は非常に低いリスク、5 は重大なリスクを示します。
risk_description	リスクの説明(very low、low、medium、high、critical)。

application_info の結合

次の表に、**application_info** テーブルで実行できる結合について説明します。

表 6-5　application_info の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_id	application_host_map.application_id app_ids_stats_current_timeframe.application_id application_tag_map.application_id app_stats_current_timeframe.application_id connection_log.application_protocol_id connection_log.client_application_id connection_log.web_application_id si_connection_log.application_protocol_id si_connection_log.application_protocol_name si_connection_log.client_application_id si_connection_log.web_application_id connection_summary.application_protocol_id file_event.application_id intrusion_event.application_protocol_id intrusion_event.client_application_id intrusion_event.web_application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id

application_info のサンプル クエリ

次のクエリは、Global \ Company B \ Edge ドメイン内で検出された、host_id が 8 のアプリケーションのレコードを返します。

```
SELECT application_id, application_name, application_description, business_relevance,
risk

FROM application_info

WHERE application_id="8" AND domain_name= "Global \ Company B \ Edge";
```

application_tag_map

application_tag_map テーブルには、ネットワークで検出された各アプリケーションに関連付けられているタグに関する情報が格納されます。

詳細については、次の項を参照してください。

- application_tag_map のフィールド（6-10 ページ）
- application_tag_map の結合（6-10 ページ）
- application_tag_map のサンプル クエリ（6-11 ページ）

application_tag_map のフィールド

次の表に、application_tag_map テーブルでアクセスできるフィールドについて説明します。

表 6-6　application_tag_map のフィールド

フィールド	説明
application_id	アプリケーションの内部識別番号。
application_name	ユーザ インターフェイスに表示されるアプリケーション。
domain_name	アプリケーションが検出されたドメインの名前。
domain_uuid	アプリケーションが検出されたドメインの UUID。これはバイナリで示されます。
tag_id	タグの内部識別番号。
tag_name	ユーザ インターフェイスに表示されるタグのテキスト。
tag_type	category または type のいずれか。

application_tag_map の結合

次の表に、application_tag_map テーブルで実行できる結合について説明します。

表 6-7　application_tag_map の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_id	app_ids_stats_current_timeframe.application_id application_info.application_id application_host_map.application_id app_stats_current_timeframe.application_id connection_log.application_protocol_id connection_log.client_application_id connection_log.web_application_id connection_summary.application_protocol_id file_event.application_id intrusion_event.application_protocol_id intrusion_event.client_application_id intrusion_event.web_application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id si_connection_log.application_protocol_name si_connection_log.application_protocol_id si_connection_log.client_application_id si_connection_log.web_application_id
tag_id	tag_info.tag_id

application_tag_map のサンプル クエリ

次のクエリは、指定されたアプリケーションに関連付けられているタグ レコードをすべて返します。

```
SELECT application_id, application_name, tag_id, tag_name
FROM application_tag_map
WHERE application_name="Active Directory";
```

domain_control_information

domain_control_information テーブルは、ドメインをその UUID にマップし、各ドメインの親ドメインの名前と UUID を示します。

詳細については、次の項を参照してください。

- domain_control_information のフィールド（6-11 ページ）
- domain_control_information の結合（6-11 ページ）
- domain_control_information のサンプル クエリ（6-11 ページ）

domain_control_information のフィールド

次の表に、**domain_control_information** テーブルでアクセスできるフィールドについて説明します。

表 6-8　domain_control_information のフィールド

フィールド	説明
domain_name	ドメインの名前
domain_uuid	ドメインの UUID。これはバイナリで示されます。
parent_domain_name	親ドメインの名前（該当する場合）。
parent_domain_uuid	親ドメインの UUID（該当する場合）。これはバイナリで示されます。

domain_control_information の結合

domain_control_information テーブルに対して結合を実行することはできません。

domain_control_information のサンプル クエリ

次のクエリは、すべてのドメイン名、ASCII 形式のドメイン UUID、およびその親ドメインを返します。

```
SELECT domain_name, uuid_btoa(domain_uuid), parent_domain_name
FROM domain_control_information;
```

network_discovery_event

network_discovery_event テーブルには、検出イベントとホスト入力イベントに関する情報が格納されます。Firepower システム は、（新しいネットワーク機能を検出したか、または以前に識別されたネットワーク資産の変更を検出することで）モニタ対象ネットワークで変更を検出すると、検出イベントを生成します。Firepower システム は、ユーザがネットワーク資産を追加、変更、または削除することでネットワーク マップを手動で変更すると、ホスト入力イベントを生成します。

network_discovery_event テーブルは、Firepower システム バージョン 5.0 以降で廃止されたテーブル **rna_events** を置き換えます。

詳細については、次の項を参照してください。

- network_discovery_event のフィールド（6-12 ページ）
- network_discovery_event の結合（6-13 ページ）
- network_discovery_event のサンプル クエリ（6-13 ページ）

network_discovery_event のフィールド

次の表に、**network_discovery_event** テーブルでアクセスできるフィールドについて説明します。

表 6-9　network_discovery_event のフィールド

フィールド	説明
confidence	Firepower システム によりサービスを識別するために割り当てられた信頼度（0 ～ 100）。
description	イベントの説明。
domain_name	イベントが検出されたドメインの名前。
domain_uuid	イベントが検出されたドメインの UUID。これはバイナリで示されます。
event_id	イベントの内部識別番号。
event_time_sec	イベントが生成された日時を示す UNIX タイムスタンプ。
event_time_usec	イベントのタイムスタンプのマイクロ秒単位の増分。
event_type	イベントのタイプ。New Host や Identity Conflict など。
ip_address	このフィールドは廃止されており、null を返します。
ipaddr	イベントに関連するホストの IPv4 または IP v 6 アドレスのバイナリ表現。
mac_address	イベントに関連するホストの MAC アドレス。
mac_vendor	イベントに関連するホストの NIC ハードウェアのベンダー。
port	イベントをトリガーしたネットワーク トラフィックが使用していたポート。
sensor_address	検出イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
sensor_name	検出イベントを生成した管理対象デバイス。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
user_dept	ホストの最終ログイン ユーザが所属する部門。
user_email	ホストの最終ログイン ユーザの電子メールアドレス。
user_first_name	ホストの最終ログイン ユーザの名前。

表 6-9　network_discovery_event のフィールド（続き）

フィールド	説明
user_id	ホストの最終ログイン ユーザの内部識別番号。
user_last_name	ホストの最終ログイン ユーザの姓。
user_last_seen_sec	ホストの最終ログイン ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
user_last_updated_sec	ホストの最終ログイン ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。
user_name	ホストの最終ログイン ユーザのユーザ名。
user_phone	ホストの最終ログイン ユーザの電話番号。

network_discovery_event の結合

次の表に、**network_discovery_event** テーブルを使用して実行できる結合について説明します。

表 6-10　network_discovery_event の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

network_discovery_event のサンプル クエリ

次のクエリは、特定の期間内の検出イベント レコードを返します。このレコードには、ユーザ、検出デバイス名、タイムスタンプ、ホスト IP アドレスなどが含まれます。

```
SELECT sensor_name, event_time_sec, event_time_usec, event_type, ipaddr, user_id,
hex(mac_address), mac_vendor, port, confidence FROM network_discovery_event
WHERE event_time_sec
BETWEEN UNIX_TIMESTAMP("2013-01-01 00:00:00") AND UNIX_TIMESTAMP("2013-01-01 23:59:59")
ORDER BY event_time_sec DESC, event_time_usec DESC;
```

rna_host

rna_host テーブルには、管理対象ネットワークのホストの基本情報が格納されます。

バージョン 5.2 以降、このテーブルは **rna_ip_host** を置き換えます。

詳細については、次の項を参照してください。

- rna_host のフィールド（6-14 ページ）
- rna_host の結合（6-14 ページ）
- rna_host のサンプル クエリ（6-15 ページ）

rna_host のフィールド

次の表に、**rna_host** テーブルでアクセスできるフィールドについて説明します。

表 6-11　rna_host のフィールド

フィールド	説明
criticality	ホストの重要度(None、Low、Medium、または High)。
domain_name	ホストが検出されたドメインの名前。
domain_uuid	ホストが検出されたドメインの UUID。これはバイナリで示されます。
hops	ホストから、そのホストを検出した管理対象デバイスまでのネットワーク ホップ数。
host_id	ホストの ID 番号。
host_name	ホストの名前。
host_type	ホストのタイプ:Host、Router、Bridge、NAT Device、または Load Balancer。
jailbroken	モバイル デバイスのオペレーティング システムがジェイルブレイクされているかどうかを示す true/false フラグ。
last_seen_sec	システムがホスト アクティビティを最後に検出した日時を示す UNIX タイムスタンプ。
mobile	検出されたホストがモバイル デバイスであるかどうかを示す true/false フラグ。
netbios_name	ホストの NetBIOS 名の文字列。
notes	ホストの Notes ホスト属性の内容。
vlan_id	VLAN 識別番号(該当する場合)。
vlan_priority	VLAN タグに含まれるプライオリティ値。
vlan_type	VLAN タグを含むカプセル化パケットのタイプ。 • 0:イーサネット • 1:トークン リング

rna_host の結合

次の表に、**rna_host** テーブルで実行できる結合について説明します。

表 6-12　rna_host の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	application_host_map.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ip_map.host_id rna_host_ioc_state.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host のサンプル クエリ

次のクエリは、Global \ Company B \ Edge ドメイン内の 25 件の **rna_host** レコードを、ホストのタイプに基づいて並べ替えて返します。このレコードには、ホスト ID、VLAN ID、ホストが最後に認識された時点、およびホストのタイプなどが含まれています。

```
SELECT host_id, vlan_id, last_seen_sec, host_type

FROM rna_host

WHERE domain_name= "Global \ Company B \ Edge"

ORDER BY host_type

LIMIT 0, 25;
```

rna_host_attribute

rna_host_attribute テーブルには、モニタ対象ネットワーク内の各ホストに関連付けられているホスト属性に関する情報が格納されます。これは、廃止された **rna_ip_host_attribute** テーブルを置き換えるテーブルです。

詳細については、次の項を参照してください。

- rna_host_attribute のフィールド（6-16 ページ）
- rna_host_attribute の結合（6-16 ページ）
- rna_host_attribute のサンプル クエリ（6-16 ページ）

rna_host_attribute のフィールド

次の表に、**rna_host_attribute** テーブルでアクセスできるフィールドについて説明します。

表 6-13　rna_host_attribute のフィールド

フィールド	説明
attribute_name	ホスト属性。Host Criticality や Default White List など。
attribute_value	ホスト属性の値。
host_id	ホストの ID 番号。

rna_host_attribute の結合

次の表に、**rna_host_attribute** テーブルで実行できる結合について説明します。

表 6-14　rna_host_attribute の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	application_host_map.host_id rna_host.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_attribute のサンプル クエリ

次のクエリは、選択されたホスト ID に関連付けられているすべてのホスト属性と値を返します。

```
SELECT attribute_name, attribute_value
FROM rna_host_attribute
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_client_app

rna_host_client_app テーブルには、モニタ対象ネットワークのホストで検出されたクライアント アプリケーションに関する情報が格納されます。これは、廃止された **rna_ip_host_client_app** テーブルを置き換えるテーブルです。

詳細については、次の項を参照してください。

- rna_host_client_app のフィールド（6-17 ページ）
- rna_host_client_app の結合（6-18 ページ）
- rna_host_client_app のサンプル クエリ（6-19 ページ）

rna_host_client_app のフィールド

次の表に、**rna_ip_host_client_app** テーブルでアクセスできるフィールドについて説明します。

表 6-15　rna_host_client_app のフィールド

フィールド	説明
application	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
application_protocol_id	検出されたアプリケーション プロトコルの内部 ID。
application_protocol_name	次のいずれかになります。 • アプリケーションの名前（確実な識別が可能な場合） • pending（システムがさらにデータを必要としている場合） • 空白（接続にアプリケーション情報がない場合）
application_type	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
client_application_id	アプリケーションが識別可能な場合、アプリケーションの内部識別番号。
client_application_name	次のいずれかになります。 • アプリケーションの名前（確実な識別が可能な場合）。 • 汎用クライアント名（システムがクライアント アプリケーションを検出したが、特定のアプリケーションであることを識別できない場合）。 • 空白（接続にクライアント アプリケーション情報がない場合）。
hits	クライアント アプリケーションが検出された回数。
host_id	ホストの ID 番号。
last_used_sec	システムがアプリケーション アクティビティを最後に検出した日時を示す UNIX タイムスタンプ。
version	ホストで検出されたアプリケーションのバージョン。

rna_host_client_app の結合

次の表に、rna_host_client_app テーブルで実行できる結合について説明します。

表 6-16　rna_host_client_app の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id
host_id および application_protocol_id および client_application_id および version	次のセット： rna_host_client_app_payload.host_id rna_host_client_app_payload.application_protocol_id rna_host_client_app_payload.client_application_id rna_host_client_app_payload.version

表 6-16　rna_host_client_app の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
application_protocol_id または client_application_id	app_ids_stats_current_timeframe.application_id application_info.application_id application_host_map.application_id application_tag_map.application_id app_stats_current_timeframe.application_id connection_log.application_protocol_id connection_log.client_application_id connection_log.web_application_id connection_summary.application_protocol_id si_connection_log.application_protocol_name si_connection_log.client_application_id si_connection_log.web_application_id file_event.application_id intrusion_event.application_protocol_id intrusion_event.client_application_id intrusion_event.web_application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_service_info.application_protocol_id rna_host_service_payload.web_application_id

rna_host_client_app のサンプル クエリ

次のクエリは、host_id が 8 のホストで検出されたクライアント アプリケーションに関する情報を返します。

```
SELECT host_id, client_application_id, client_application_name, version, hits,
application_protocol_id, application_protocol_name, last_used_sec
FROM rna_host_client_app
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_client_app_payload

rna_host_client_app_payload テーブルには、モニタ対象ネットワークで検出されたホストの Web アプリケーションに関連付けられている HTTP トラフィックのペイロードに関する情報が含まれています。

詳細については、次の項を参照してください。

- rna_host_client_app_payload のフィールド（6-20 ページ）
- rna_host_client_app_payload の結合（6-21 ページ）
- rna_host_client_app_payload のサンプル クエリ（6-22 ページ）

rna_host_client_app_payload のフィールド

次の表に、**rna_host_client_app_payload** テーブルでアクセスできるフィールドについて説明します。

表 6-17　rna_host_client_app_payload のフィールド

フィールド	説明
application	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
application_protocol_id	検出されたアプリケーション プロトコルの内部 ID(使用可能な場合)。クライアント アプリケーションと Web アプリケーションの両方の特性を持つトラフィックの場合、client_application_id フィールドと web_application_id フィールドの両方に同じ値が入っています。
application_protocol_name	次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合) • pending(システムがさらにデータを必要としている場合) • 空白(接続にアプリケーション情報がない場合)
application_type	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
client_application_id	クライアント アプリケーションの内部識別番号。
client_application_name	次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合)。 • 汎用クライアント名(システムがクライアント アプリケーションを検出したが、特定のアプリケーションであることを識別できない場合)。 • 空白(接続にクライアント アプリケーション情報がない場合)。
host_id	ホストの ID 番号。
payload_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
payload_type	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
version	ホストで検出された Web アプリケーションのバージョン。
web_application_id	Web アプリケーションの内部識別番号(使用可能な場合)。クライアント アプリケーションと Web アプリケーションの両方の特性を持つトラフィックの場合、client_application_id フィールドと web_application_id フィールドの両方に同じ値が入っています。
web_application_name	次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合)。 • web browsing(システムがアプリケーション プロトコル HTTP を検出したが、特定の Web アプリケーションを検出できない場合)。 • 空白(接続に HTTP トラフィックがない場合)。

rna_host_client_app_payload の結合

次の表に、`rna_host_client_app_payload` テーブルで実行できる結合について説明します。

表 6-18　rna_host_client_app_payload の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
`host_id`	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id
次のセット： `host_id`, `application_protocol_id`, `client_application_id`, `version`	次のセット： rna_host_client_app.host_id rna_host_client_app.application_protocol_id rna_host_client_app.client_application_id rna_host_client_app.version

表 6-18　rna_host_client_app_payload の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
client_application_id または web_application_id	app_ids_stats_current_timeframe.application_id application_info.application_id application_host_map.application_id application_tag_map.application_id app_stats_current_timeframe.application_id connection_log.application_protocol_id connection_log.client_application_id connection_log.web_application_id connection_summary.application_protocol_id si_connection_log.application_protocol_name si_connection_log.client_application_id si_connection_log.web_application_id file_event.application_id intrusion_event.application_protocol_id intrusion_event.client_application_id intrusion_event.web_application_id rna_host_service_info.application_protocol_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id

rna_host_client_app_payload のサンプル クエリ

次のクエリは、host_id が 8 のホストで検出された Web アプリケーションに関する情報を返します。

```
SELECT host_id, web_application_id, web_application_name, version,
client_application_id, client_application_name

FROM rna_host_client_app_payload

WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_ioc_state

rna_host_ioc_state テーブルには、モニタ対象ネットワークのホストの IOC 状態が格納されます。

詳細については、次の項を参照してください。

- rna_host_ioc_state のフィールド（6-23 ページ）
- rna_host_ioc_state の結合（6-25 ページ）
- rna_host_ioc_state のサンプル クエリ（6-25 ページ）

rna_host_ioc_state のフィールド

次の表に、**rna_host_ioc_state** テーブルでアクセスできるフィールドについて説明します。

表 6-19　rna_host_ioc_state のフィールド

フィールド	説明
first_seen	侵害が最初に検出された時点を示す UNIX タイムスタンプ。
first_seen_sensor_address	侵害を最初に検出した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
first_seen_sensor_name	侵害を最初に検出した管理対象デバイス。
host_id	ホストの ID 番号。
ioc_category	侵害のカテゴリ。有効な値は次のとおりです。 • CnC Connected • Exploit Kit • High Impact Attack • Low Impact Attack • Malware Detected • Malware Executed • Dropper Infection • Java Compromise • Word Compromise • Adobe Reader Compromise • Excel Compromise • PowerPoint Compromise • QuickTime Compromise
ioc_description	侵害の説明

表 6-19　rna_host_ioc_state のフィールド（続き）

フィールド	説明
ioc_event_type	侵害のイベント タイプ。有効な値は次のとおりです。 • Adobe Reader launched shell • Dropper Infection Detected by AMP for Endpoints • Excel Compromise Detected by AMP for Endpoints • Excel launched shell • Impact 1 Intrusion Event – attempted-admin • Impact 1 Intrusion Event – attempted-user • Impact 1 Intrusion Event – successful-admin • Impact 1 Intrusion Event – successful-user • Impact 1 Intrusion Event – web-application-attack • Impact 2 Intrusion Event – attempted-admin • Impact 2 Intrusion Event – attempted-user • Impact 2 Intrusion Event – successful-admin • Impact 2 Intrusion Event – successful-user • Impact 2 Intrusion Event – web-application-attack • Intrusion Event – exploit-kit • Intrusion Event – malware-backdoor • Intrusion Event – malware-CnC • Java Compromise Detected by AMP for Endpoints • Java launched shell • PDF Compromise Detected by AMP for Endpoints • PowerPoint Compromise Detected by AMP for Endpoints • PowerPoint launched shell • QuickTime Compromise Detected by AMP for Endpoints • QuickTime launched shell • Security Intelligence Event – CnC • Suspected Botnet Detected by AMP for Endpoints • Threat Detected by AMP for Endpoints – Subtype is 'executed' • Threat Detected by AMP for Endpoints – Subtype is not 'executed' • Threat Detected in File Transfer – Action is not 'block' • Word Compromise Detected by AMP for Endpoints • Word launched shell
ioc_id	侵害の一意の ID 番号。
is_disabled	この侵害が無効にされていたかどうか。
last_seen	この侵害が最後に検出された時点を示す UNIX タイムスタンプ。

表 6-19　rna_host_ioc_state のフィールド（続き）

フィールド	説明
last_seen_sensor_address	侵害を最後に検出した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
last_seen_sensor_name	侵害を最後に検出した管理対象デバイス。

rna_host_ioc_state の結合

次の表に、**rna_host_ioc_state** テーブルで実行できる結合について説明します。

表 6-20　rna_host_ioc_state の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_ioc_state のサンプル クエリ

次のクエリは、指定された期間内の最大 25 件のホストとその ioc を返します。

```
SELECT host_id, ioc_id
FROM rna_host_ioc_state
WHERE first_seen
BETWEEN UNIX_TIMESTAMP("2011-10-01 00:00:00")
AND UNIX_TIMESTAMP("2011-10-07 23:59:59")
ORDER BY ioc_id DESC
LIMIT 0, 25;
```

rna_host_ip_map

rna_host_ip_map テーブルは、モニタ対象ネットワーク内のホストの IP アドレスにホスト ID を相関付けます。

詳細については、次の項を参照してください。

- rna_host_ip_map のフィールド（6-26 ページ）
- rna_host_ip_map の結合（6-26 ページ）
- rna_host_ip_map のサンプル クエリ（6-27 ページ）

rna_host_ip_map のフィールド

次の表に、rna_host_ip_map テーブルでアクセスできるフィールドについて説明します。

表 6-21　rna_host_ip_map のフィールド

フィールド	説明
host_id	ホストの ID 番号。
ipaddr	ホストの IP アドレスのバイナリ表現。

rna_host_ip_map の結合

次の表に、rna_host_ip_map テーブルで実行できる結合について説明します。

表 6-22　rna_host_ip_map の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

表 6-22　rna_host_ip_map の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
ipaddr	compliance_event.dst_ipaddr compliance_event.src_ipaddr connection_log.initiator_ipaddr connection_log.responder_ipaddr connection_summary.initiator_ipaddr connection_summary.responder_ipaddr fireamp_event.dst_ipaddr fireamp_event.src_ipaddr intrusion_event.dst_ipaddr intrusion_event.src_ipaddr network_discovery_event.ipaddr si_connection_log.initiator_ipaddr si_connection_log.responder_ipaddr user_discovery_event.ipaddr user_ipaddr_history.ipaddr white_list_event.ipaddr

rna_host_ip_map のサンプル クエリ

次のクエリは、選択されたホストの MAC 情報を返します。

```
SELECT host_id

FROM rna_host_ip_map

WHERE HEX(ipaddr) = "00000000000000000000FFFF0A0A0A04";
```

rna_host_mac_map

rna_host_mac_map テーブルは、モニタ対象ネットワーク内のホストの MAC アドレスにホスト ID を相関付けます。

詳細については、次の項を参照してください。

- rna_host_mac_map のフィールド（6-27 ページ）
- rna_host_mac_map の結合（6-28 ページ）
- rna_host_mac_map のサンプル クエリ（6-28 ページ）

rna_host_mac_map のフィールド

次の表に、**rna_host_mac_map** テーブルでアクセスできるフィールドについて説明します。

表 6-23　rna_host_mac_map のフィールド

フィールド	説明
host_id	ホストの ID 番号。
mac_address	ホストの MAC アドレス。
mac_vendor	検出されたホストのネットワーク インターフェイスのベンダー。

rna_host_mac_map の結合

次の表に、**rna_host_mac_map** テーブルで実行できる結合について説明します。

表 6-24　rna_host_mac_map の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_mac_map のサンプル クエリ

次のクエリは、host_id が 8 のホストの MAC 情報を返します。

```
SELECT HEX(mac_address)
FROM rna_host_mac_map
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_os

rna_host_os テーブルには、モニタ対象ネットワークのホストで検出されたオペレーティング システムに関する情報が格納されます。

詳細については、次の項を参照してください。

- rna_host_os のフィールド（6-29 ページ）
- rna_host_os の結合（6-30 ページ）
- rna_host_os のサンプル クエリ（6-30 ページ）

rna_host_os のフィールド

次の表に、**rna_host_os** テーブルでアクセスできるフィールドについて説明します。

表 6-25　rna_host_os のフィールド

フィールド	説明
confidence	Firepower システム によりオペレーティング システムを識別するために割り当てられた信頼度（0 ～ 100）。
created_sec	システムがホスト アクティビティを最初に検出した日時を示す UNIX タイムスタンプ。
host_id	ホストの ID 番号。
last_seen_sec	システムがホスト アクティビティを最後に検出した日時を示す UNIX タイムスタンプ。
os_uuid	ホストで検出されたオペレーティング システムの固有識別子。UUID は、Firepower システム データベース内のオペレーティング システムの名前、ベンダー、およびバージョンにマップされます。
product	ホストで検出されたオペレーティング システム。
source_type	ホストのオペレーティング システムのアイデンティティ ソース。 • User：Web ユーザ インターフェイスからデータを入力したユーザの名前 • Application：ホスト入力機能を使用して別のアプリケーションからインポートされた • Scanner：Nmap、またはシステム ポリシーによって追加された別のスキャナ • rna：Firepower システム により検出（検出イベント、ポート一致、またはパターン一致のいずれかにより検出）。 • NetFlow：NetFlow 対応デバイスによってデータがエクスポートされた
vendor	ホストで検出されたオペレーティング システムのベンダー。
version	ホストで検出されたオペレーティング システムのバージョン。

rna_host_os の結合

次の表に、rna_host_os テーブルで実行できる結合について説明します。

表 6-26　rna_host_os の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_os のサンプル クエリ

次のクエリは、host_id が 8 のホストのオペレーティング システムに関する情報を返します。

```
SELECT vendor, product, version, source_type, confidence
FROM rna_host_os
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_os_vulns

rna_host_os_vulns テーブルには、モニタ対象ネットワーク内のホストに関連付けられている脆弱性に関する情報が格納されます。

詳細については、次の項を参照してください。

- rna_host_os_vulns のフィールド（6-31 ページ）
- rna_host_os_vulns の結合（6-31 ページ）
- rna_host_os_vulns のサンプル クエリ（6-32 ページ）

rna_host_os_vulns のフィールド

次の表に、**rna_host_os_vulns** テーブルでアクセスできるフィールドについて説明します。

表 6-27　rna_host_os_vulns のフィールド

フィールド	説明
host_id	ホストの ID 番号。
invalid	脆弱性がホストで有効であるかどうかを示す値。 • 0:脆弱性が有効 • 1:脆弱性が無効
rna_vuln_id	脆弱性の内部識別番号。

rna_host_os_vulns の結合

次の表に、**rna_host_os_vulns** テーブルで実行できる結合について説明します。

表 6-28　rna_host_os_vulns の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
rna_vuln_id	rna_vuln.bugtraq_id rna_vuln.rna_vuln_id rna_host_third_party_vuln_rna_id.rna_vuln_id rna_host_third_party_vuln_cve_id.cve_id rna_host_third_party_vuln_bugtraq_id.bugtraq_id
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_os_vulns のサンプル クエリ

次のクエリは、host_id が 8 のホストのオペレーティング システムの脆弱性を返します。

```
SELECT rna_vuln_id, invalid
FROM rna_host_os_vulns
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_protocol

rna_host_protocol テーブルには、モニタ対象ネットワークのホストで検出されたプロトコルに関する情報が格納されます。

詳細については、次の項を参照してください。

- rna_host_protocol のフィールド（6-32 ページ）
- rna_host_protocol の結合（6-33 ページ）
- rna_host_protocol のサンプル クエリ（6-33 ページ）

rna_host_protocol のフィールド

次の表に、**rna_host_protocol** テーブルでアクセスできるフィールドについて説明します。

表 6-29　rna_host_protocol のフィールド

フィールド	説明
host_id	ホストの ID 番号。
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
layer	プロトコルを実行しているネットワーク層（Network または Transport）。
mac_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
mac_vendor	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
protocol_name	ホストが使用するトラフィック プロトコル。
protocol_num	プロトコルの IANA 指定のプロトコル番号。

rna_host_protocol の結合

次の表に、**rna_host_protocol** テーブルで実行できる結合について説明します。

表 6-30　rna_host_protocol の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_protocol のサンプル クエリ

次のクエリは、host_id が 8 のホストのプロトコル レコードをすべて返します。

```
SELECT protocol_num, protocol_name
FROM rna_host_protocol
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_sensor

rna_host_sensor テーブルには、モニタ対象ネットワーク内のホスト IP アドレスが格納され、各アドレスを検出した管理対象デバイスが示されています。

rna_host_sensor テーブルは、Firepower システム バージョン 5.2 以降で廃止されたテーブル **rna_ip_host_sensor** を置き換えます。

詳細については、次の項を参照してください。

- rna_host_sensor のフィールド（6-34 ページ）
- rna_host_sensor の結合（6-34 ページ）
- rna_host_sensor のサンプル クエリ（6-34 ページ）

rna_host_sensor のフィールド

次の表に、**rna_host_sensor** テーブルでアクセスできるフィールドについて説明します。

表 6-31　　rna_host_sensor のフィールド

フィールド	説明
host_id	ホストの ID 番号。
sensor_address	検出イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_name	管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。

rna_host_sensor の結合

次の表に、**rna_host_sensor** テーブルで実行できる結合について説明します。

表 6-32　　rna_host_sensor の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_sensor のサンプル クエリ

次のクエリは、最大 25 件のホストと、それらのホストを検出したセンサーを **rna_host_sensor** テーブルから返します。

```
SELECT host_id, sensor_address, sensor_name
FROM rna_host_sensor
LIMIT 0, 25;
```

rna_host_service

rna_host_service テーブルには、ネットワーク ポートとトラフィック プロトコルの組み合わせによりモニタ対象ネットワークのホストで検出されたサービスに関する一般情報が含まれています。

詳細については、次の項を参照してください。

- rna_host_service のフィールド（6-35 ページ）
- rna_host_service の結合（6-35 ページ）
- rna_host_service のサンプル クエリ（6-36 ページ）

rna_host_service のフィールド

次の表に、**rna_host_service** テーブルでアクセスできるフィールドについて説明します。

表 6-33　rna_host_service のフィールド

フィールド	説明
confidence	Firepower システム によりサーバを識別するために割り当てられた信頼度（0 ～ 100）。
hits	サーバが検出された回数。
host_id	ホストの ID 番号。
last_used_sec	システムが最後にサーバ アクティビティを検出した日時を示す UNIX タイムスタンプ。
port	サーバで使用されるポート。
protocol	トラフィック プロトコル:TCP または UDP。

rna_host_service の結合

次の表に、**rna_host_service** テーブルで実行できる結合について説明します。

表 6-34　rna_host_service の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id
次のセット： host_id port protocol	次のセット： rna_host_service_banner.host_id rna_host_service_banner.port rna_host_service_banner.protocol 次のセット： rna_host_service_info.host_id rna_host_service_info.port rna_host_service_info.protocol 次のセット： rna_host_service_payload.host_id rna_host_service_payload.port rna_host_service_payload.protocol

rna_host_service のサンプル クエリ

次のクエリは、host_id が 8 のホストについて検出された、最初の 25 件のサーバ レコードを返します。

```
SELECT hits, protocol, port, confidence

FROM rna_host_service

WHERE HEX(host_id) = "00000000000000000000000000000008"

LIMIT 0, 25;
```

rna_host_service_banner

rna_ip_host_service_banner テーブルには、ネットワーク トラフィックからのヘッダー情報が格納されます。このヘッダー情報は、モニタ対象ネットワークのホストのサーバのベンダーとバージョン(「バナー」)をアドバタイズします。ネットワーク検出ポリシーの [Capture Banners] オプションを有効にしていない場合は、Firepower システム はサーバ バナーを保存しない点に注意してください。

詳細については、次の項を参照してください。

- rna_host_service_banner のフィールド (6-37 ページ)
- rna_host_service_banner の結合 (6-37 ページ)
- rna_host_service_banner のサンプル クエリ (6-38 ページ)

rna_host_service_banner のフィールド

次の表に、rna_host_service_banner テーブルでアクセスできるフィールドについて説明します。

表 6-35　rna_host_service_banner のフィールド

フィールド	説明
バナー	サーバのバナー(そのサーバについて検出された最初のパケットの最初の 256 バイト)。
host_id	ホストの ID 番号。
port	サーバで使用されるポート。
protocol	トラフィック プロトコル:TCP または UDP。

rna_host_service_banner の結合

次の表に、rna_host_service_banner テーブルで実行できる結合について説明します。

表 6-36　rna_host_service_banner の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
次のセット: host_id port protocol	次のセット: rna_host_service.host_id rna_host_service.port rna_host_service.protocol 次のセット: rna_host_service_info.host_id rna_host_service_info.port rna_host_service_info.protocol 次のセット: rna_host_service_payload.host_id rna_host_service_payload.port rna_host_service_payload.protocol

表 6-36　rna_host_service_banner の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_service_banner のサンプル クエリ

次のクエリは、host_id が 8 のホストのサーバ バナーを返します。

```
SELECT port, protocol, banner
FROM rna_host_service_banner
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_service_info

rna_host_service_info テーブルには、モニタ対象ネットワークのホストで検出されたサーバに関する詳細情報が格納されます。

詳細については、次の項を参照してください。

- rna_host_service_info のフィールド（6-39 ページ）
- rna_host_service_info の結合（6-40 ページ）
- rna_host_service_info のサンプル クエリ（6-41 ページ）

rna_host_service_info のフィールド

次の表に、**rna_host_service_info** テーブルでアクセスできるフィールドについて説明します。

表 6-37　rna_host_service_info のフィールド

フィールド	説明
application_id	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して空白を返します。
application_protocol_id	検出されたアプリケーション プロトコルの内部 ID(使用可能な場合)。
application_protocol_name	次のいずれか: • アプリケーション プロトコルの名前(確実な識別が可能な場合) • pending(システムがさらにデータを必要としている場合) • 空白(接続にアプリケーション情報がない場合)
business_relevance	ビジネスの生産性に対するアプリケーションの関連度のインデックス(1 ～ 5)。1 は非常に低く、5 は非常に高いことを示します。
business_relevance_ description	ビジネスとの関連度の説明(very low、low、medium、high、very high)。
created_sec	システムが最初にアプリケーション プロトコルを検出した日時を示す UNIX タイムスタンプ。
host_id	ホストの ID 番号。
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
last_used_sec	システムが最後にサーバ アクティビティを検出した日時を示す UNIX タイムスタンプ。
port	サーバで使用されるポート。
protocol	トラフィック プロトコル:TCP または UDP。
risk	アプリケーション リスクのインデックス(1 ～ 5)。1 は非常に低いリスク、5 は非常に高いリスクを示します。
risk_description	リスクの説明(very low、low、medium、high、very high)。
service_info_id	サーバの内部識別番号。
service_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
source_type	サーバのアイデンティティ ソース: • User:Web ユーザ インターフェイスからデータを入力したユーザの名前 • Application:ホスト入力機能を使用して別のアプリケーションからインポートされた • Scanner:NMAP により追加されたか、または送信元タイプが Scanner でホスト入力機能からインポートされた • rna:Firepower システム により検出(検出イベント、ポート一致、またはパターン一致のいずれかにより検出)。 • NetFlow:NetFlow 対応デバイスによってデータがエクスポートされた

表 6-37　rna_host_service_info のフィールド（続き）

フィールド	説明
vendor	ホストのサーバのベンダー。
version	ホストで検出されたサーバのバージョン。

rna_host_service_info の結合

次の表に、rna_host_service_info テーブルで実行できる結合について説明します。

表 6-38　rna_host_service_info の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_protocol_id	app_ids_stats_current_timeframe.application_id application_info.application_id application_host_map.application_id application_tag_map.application_id app_stats_current_timeframe.application_id connection_log.application_protocol_id connection_log.client_application_id connection_log.web_application_id connection_summary.application_protocol_id si_connection_log.application_protocol_name si_connection_log.client_application_id si_connection_log.web_application_id file_event.application_id intrusion_event.application_protocol_id intrusion_event.client_application_id intrusion_event.web_application_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id

表 6-38　rna_host_service_info の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_payload.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id
次のセット： host_id および port および protocol	次のセット： rna_host_service.host_id rna_host_service.port rna_host_service.protocol 次のセット： rna_host_service_banner.host_id rna_host_service_banner.port rna_host_service_banner.protocol 次のセット： rna_host_service_payload.host_id rna_host_service_payload.port rna_host_service_payload.protocol

rna_host_service_info のサンプル クエリ

次のクエリは、host_id が 8 のホストで検出されたアプリケーション プロトコルに関する情報を返します。

```
SELECT host_id, application_protocol_name, version, vendor, created_sec, last_used_sec,
business_relevance, risk

FROM rna_host_service_info

WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_service_payload

rna_host_service_payload テーブルには、モニタ対象ネットワーク内のホストにより関連付けられている Web アプリケーションに関する情報が含まれています。

詳細については、次の項を参照してください。

- rna_host_service_payload のフィールド (6-42 ページ)
- rna_host_service_payload の結合 (6-43 ページ)
- rna_host_service_payload のサンプル クエリ (6-44 ページ)

rna_host_service_payload のフィールド

次の表に、**rna_host_service_payload** テーブルでアクセスできるフィールドについて説明します。

表 6-39　rna_host_service_payload のフィールド

フィールド	説明
application_id	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
application_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
host_id	ホストの ID 番号。
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
payload_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
payload_type	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
port	サーバで使用されるポート。
protocol	トラフィック プロトコル:TCP または UDP。
web_application_id	Web アプリケーションの内部識別番号。
web_application_name	次のいずれかになります。 • Web アプリケーションの名前(確実な識別が可能な場合) • web browsing(システムがアプリケーション プロトコル HTTP を検出したが、特定の Web アプリケーションを検出できない場合)。 • 空白(接続に HTTP トラフィックがない場合)。

rna_host_service_payload の結合

次の表に、rna_host_service_payload テーブルで実行できる結合について説明します。

表 6-40　rna_host_service_payload の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
web_application_id	app_ids_stats_current_timeframe.application_id application_info.application_id application_host_map.application_id application_tag_map.application_id app_stats_current_timeframe.application_id connection_log.application_protocol_id connection_log.client_application_id connection_log.web_application_id connection_summary.application_protocol_id si_connection_log.application_protocol_name si_connection_log.client_application_id si_connection_log.web_application_id file_event.application_id intrusion_event.application_protocol_id intrusion_event.client_application_id intrusion_event.web_application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id
次のセット： host_id port protocol	次のセット： rna_host_service.host_id rna_host_service.port rna_host_service.protocol 次のセット： rna_host_service_banner.host_id rna_host_service_banner.port rna_host_service_banner.protocol 次のセット： rna_host_service_info.host_id rna_host_service_info.port rna_host_service_info.protocol

表 6-40　rna_host_service_payload の結合(続き)

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_vulns.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_service_payload のサンプル クエリ

次のクエリは、host_id が 8 のホストで検出された Web アプリケーションに関する情報を返します。

```
SELECT host_id, web_application_id, web_application_name, port, protocol

FROM rna_host_service_payload

WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_service_subtype

rna_host_service_subtype テーブルには、モニタ対象ネットワークのホストで検出されたサーバのサブサーバに関する情報が格納されています。

詳細については、次の項を参照してください。

- rna_host_service_subtype のフィールド (6-45 ページ)
- rna_host_service_subtype の結合 (6-45 ページ)
- rna_host_service_subtype のサンプル クエリ (6-45 ページ)

rna_host_service_subtype のフィールド

次の表に、**rna_host_service_subtype** テーブルでアクセスできるフィールドについて説明します。

表 6-41　rna_host_service_subtype のフィールド

フィールド	説明
host_id	ホストの ID 番号。
port	サーバで使用されるポート。
protocol	トラフィック プロトコル:TCP または UDP。
service_name	次のいずれかになります。 • トリガー イベントに関連付けられているホストのサーバ。 • none または空白(識別のためのデータが使用できない場合) • pending(追加データが必要な場合) • unknown(システムが既知のサーバ フィンガープリントに基づいてサーバを識別できない場合)
source_type	サーバのアイデンティティ ソース: • User:Web ユーザ インターフェイスからデータを入力したユーザの名前 • Application:ホスト入力機能を使用して別のアプリケーションからインポートされた • Scanner:NMAP により追加されたか、または送信元タイプが Scanner でホスト入力機能からインポートされた • rna:Firepower システム により検出(検出イベント、ポート一致、またはパターン一致のいずれかにより検出)。 • NetFlow:NetFlow 対応デバイスによってデータがエクスポートされた
sub_service_name	ホストで検出されたサブサーバ。
sub_service_vendor	ホストで検出されたサブサーバのベンダー。
sub_service_version	ホストで検出されたサブサーバのバージョン。
vendor	ホストで検出されたサーバのベンダー。
version	ホストで検出されたサーバのバージョン。

rna_host_service_subtype の結合

rna_host_service_subtype テーブルに対して結合を実行することはできません。

rna_host_service_subtype のサンプル クエリ

次のクエリは、host_id が 8 のホストで検出されたすべてのサブサーバ レコードを返します。

```
SELECT host_id, service_name, version, sub_service_name, sub_service_version,
sub_service_vendor
FROM rna_host_service_subtype
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_service_vulns

rna_host_service_vulns テーブルには、モニタ対象ネットワーク内のホストで検出されたサーバにマップされている脆弱性に関する情報が格納されています。

詳細については、次の項を参照してください。

- rna_host_service_vulns のフィールド (6-46 ページ)
- rna_host_service_vulns の結合 (6-46 ページ)
- rna_host_service_vulns のサンプル クエリ (6-47 ページ)

rna_host_service_vulns のフィールド

次の表に、**rna_host_service_vulns** テーブルでアクセスできるフィールドについて説明します。

表 6-42　rna_host_service_vulns のフィールド

フィールド	説明
application_id	ホストで実行されているアプリケーション プロトコルの内部識別番号。
application_name	ユーザ インターフェイスに表示されるアプリケーション プロトコル名。
host_id	ホストの ID 番号。
invalid	アプリケーション プロトコルを実行しているホストで脆弱性が有効であるかどうかを示す値。 • 0:脆弱性が有効 • 1:脆弱性が無効
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
port	サーバで使用されるポート。
protocol	トラフィック プロトコル:TCP または UDP。
rna_vuln_id	脆弱性の内部識別番号。
service_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
vendor	ホストで検出されたサーバのベンダー。
version	ホストで検出されたサーバのバージョン。

rna_host_service_vulns の結合

次の表に、**rna_host_service_vulns** テーブルで実行できる結合について説明します。

表 6-43　rna_host_service_vulns の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
rna_vuln_id	rna_vuln.bugtraq_id rna_vuln.rna_vuln_id rna_host_third_party_vuln_rna_id.rna_vuln_id rna_host_third_party_vuln_cve_id.cve_id rna_host_third_party_vuln_bugtraq_id.bugtraq_id
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_payload.host_id rna_host_third_party_vuln_bugtraq_id.host_id rna_host_third_party_vuln_cve_id.host_id rna_host_third_party_vuln_rna_id.host_id rna_host_third_party_vuln.host_id

rna_host_service_vulns のサンプル クエリ

次のクエリは、host_id が 8 のホストで検出されたすべてのサーバ脆弱性に関する情報を返します。

```
SELECT host_id, rna_vuln_id, vendor, service_name, version, invalid FROM
rna_host_service_vulns
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_third_party_vuln

rna_host_third_party_vuln テーブルには、モニタ対象ネットワーク内のホストに関連付けられているサードパーティの脆弱性に関する情報が含まれています。このテーブルの情報は、ホスト入力機能によってインポートされるサードパーティの脆弱性データによって決定されることに注意してください。

詳細については、次の項を参照してください。

- rna_host_third_party_vuln のフィールド（6-48 ページ）
- rna_host_third_party_vuln の結合（6-48 ページ）
- rna_host_third_party_vuln のサンプル クエリ（6-49 ページ）

rna_host_third_party_vuln のフィールド

次の表に、**rna_host_third_party_vuln** テーブルでアクセスできるフィールドについて説明します。

表 6-44　rna_host_third_party_vuln のフィールド

フィールド	説明
description	脆弱性に関する説明。
host_id	ホストの ID 番号。
invalid	脆弱性がホストで有効であるかどうかを示す値。 • 0:脆弱性が有効 • 1:脆弱性が無効
name	脆弱性のタイトル。
port	ポート番号(脆弱性が、特定のポートで検出された関連アプリケーションまたはサーバに関連付けられている場合)。
protocol	トラフィック プロトコル(TCP または UDP)(そのプロトコルを使用するアプリケーションに脆弱性が関連付けられている場合)。
source	脆弱性のソース。
third_party_vuln_id	脆弱性に関連付けられた識別番号。

rna_host_third_party_vuln の結合

次の表に、**rna_host_third_party_vuln** テーブルで実行できる結合について説明します。

表 6-45　rna_host_third_party_vuln の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id

rna_host_third_party_vuln のサンプル クエリ

次のクエリは、host_id が 8 のホストのサードパーティの脆弱性に関する情報を返します。

```
SELECT host_id, third_party_vuln_id, name, description, source, invalid
FROM rna_host_third_party_vuln
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_third_party_vuln_bugtraq_id

rna_host_third_party_vuln_bugtraq_id テーブルには、Bugtraq データベースの脆弱性にマップされており、またモニタ対象ネットワーク内のホストに関連付けられているサードパーティの脆弱性に関する情報が格納されます。このテーブルのサード パーティ脆弱性データは、ホスト入力機能によってインポートされることに注意してください。

詳細については、次の項を参照してください。

- rna_host_third_party_vuln_bugtraq_id のフィールド（6-49 ページ）
- rna_host_third_party_vuln_bugtraq_id の結合（6-50 ページ）
- rna_host_third_party_vuln_bugtraq_id のサンプル クエリ（6-50 ページ）

rna_host_third_party_vuln_bugtraq_id のフィールド

次の表に、**rna_host_third_party_vuln_bugtraq_id** テーブルでアクセスできるフィールドについて説明します。

表 6-46　rna_host_third_party_vuln_bugtraq_id のフィールド

フィールド	説明
bugtraq_id	脆弱性に関連付けられている Bugtraq データベースの識別番号。
description	脆弱性に関する説明。
host_id	ホストの ID 番号。
invalid	脆弱性がホストで有効であるかどうかを示す値。 • 0:脆弱性が有効 • 1:脆弱性が無効
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
name	脆弱性の名前またはタイトル。
port	ポート番号（脆弱性が、特定のポートで検出された関連アプリケーションまたはサーバに関連付けられている場合）。
protocol	トラフィック プロトコル（TCP または UDP）（そのプロトコルを使用するアプリケーションに脆弱性が関連付けられている場合）。
source	脆弱性のソース。
third_party_vuln_id	脆弱性に関連付けられているサードパーティの識別番号。

rna_host_third_party_vuln_bugtraq_id の結合

次の表に、**rna_host_third_party_vuln_bugtraq_id** テーブルで実行できる結合について説明します。

表 6-47　rna_host_third_party_vuln_bugtraq_id の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
bugtraq_id	rna_vuln.bugtraq_id rna_vuln.rna_vuln_id rna_host_os_vulns.rna_vuln_id rna_host_service_vulns.rna_vuln_id
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id

rna_host_third_party_vuln_bugtraq_id のサンプル クエリ

次のクエリは、host_id が 8 のホストの BugTraq 脆弱性を返します。

```
SELECT host_id, third_party_vuln_id, bugtraq_id, name, description, source, invalid
FROM rna_host_third_party_vuln_bugtraq_id
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_third_party_vuln_cve_id

rna_host_third_party_vuln_cve_id テーブルには、MITRE の CVE データベースの脆弱性にマップされており、またモニタ対象ネットワーク内のホストに関連付けられているサードパーティの脆弱性に関する情報が格納されます。このテーブルには、ホスト入力機能によってインポートされるサードパーティの脆弱性データが格納されていることに注意してください。

詳細については、次の項を参照してください。

- rna_host_third_party_vuln_cve_id のフィールド（6-51 ページ）
- rna_host_third_party_vuln_cve_id の結合（6-51 ページ）
- rna_host_third_party_vuln_cve_id のサンプル クエリ（6-52 ページ）

rna_host_third_party_vuln_cve_id のフィールド

次の表に、**rna_host_third_party_vuln_cve_id** テーブルでアクセスできるフィールドについて説明します。

表 6-48　　rna_host_third_party_vuln_cve_id のフィールド

フィールド	説明
cve_id	MITRE の CVE データベースにおいて脆弱性に関連付けられている識別番号。
description	脆弱性に関する説明。
host_id	ホストの ID 番号。
invalid	脆弱性がホストで有効であるかどうかを示す値。 • 0:脆弱性が有効 • 1:脆弱性が無効
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
name	脆弱性の名前またはタイトル。
port	ポート番号(脆弱性が、特定のポートで検出された関連アプリケーションまたはサーバに関連付けられている場合)。
protocol	トラフィック プロトコル(TCP または UDP)(そのプロトコルを使用するアプリケーションに脆弱性が関連付けられている場合)。
source	脆弱性のソース。
third_party_vuln_id	脆弱性に関連付けられた識別番号。

rna_host_third_party_vuln_cve_id の結合

次の表に、**rna_host_third_party_vuln_cve_id** テーブルで実行できる結合について説明します。

表 6-49　　rna_host_third_party_vuln_cve_id の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
cve_id	rna_vuln.bugtraq_id rna_vuln.rna_vuln_id rna_host_os_vulns.rna_vuln_id rna_host_service_vulns.rna_vuln_id

表 6-49　rna_host_third_party_vuln_cve_id の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id

rna_host_third_party_vuln_cve_id のサンプル クエリ

次のクエリは、host_id が 8 のホストの CVE 脆弱性を返します。

```
SELECT host_id, third_party_vuln_id, cve_id, name, description, source, invalid
FROM rna_host_third_party_vuln_cve_id
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_host_third_party_vuln_rna_id

rna_host_third_party_vuln_bugtraq_id テーブルには、Firepower システム 脆弱性データベース（VDB）の脆弱性にマップされており、またモニタ対象ネットワーク内のホストに関連付けられているサードパーティの脆弱性に関する情報が格納されています。このテーブルのサード パーティ脆弱性データは、ホスト入力機能によってインポートされることに注意してください。

詳細については、次の項を参照してください。

- rna_host_third_party_vuln_rna_id のフィールド（6-53 ページ）
- rna_host_third_party_vuln_rna_id の結合（6-53 ページ）
- rna_host_third_party_vuln_rna_id のサンプル クエリ（6-54 ページ）

rna_host_third_party_vuln_rna_id のフィールド

次の表に、**rna_host_third_party_vuln_rna_id** テーブルでアクセスできるフィールドについて説明します。

表 6-50　rna_host_third_party_vuln_rna_id のフィールド

フィールド	説明
description	脆弱性に関する説明。
host_id	ホストの ID 番号。
invalid	脆弱性がホストで有効であるかどうかを示す値。 • 0:脆弱性が有効 • 1:脆弱性が無効
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
name	脆弱性の名前またはタイトル。
port	ポート番号(脆弱性が、特定のポートで検出された関連アプリケーションまたはサーバに関連付けられている場合)。
protocol	トラフィック プロトコル(TCP または UDP)(そのプロトコルを使用するアプリケーションに脆弱性が関連付けられている場合)。
rna_vuln_id	脆弱性を追跡するために シスコ が使用する脆弱性の識別番号。
source	脆弱性のソース。
third_party_vuln_id	脆弱性に関連付けられた識別番号。

rna_host_third_party_vuln_rna_id の結合

次の表に、**rna_host_third_party_vuln_rna_id** テーブルで実行できる結合について説明します。

表 6-51　rna_host_third_party_vuln_rna_id の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
rna_vuln_id	rna_vuln.bugtraq_id rna_vuln.rna_vuln_id rna_host_os.rna_vuln_id rna_host_service_vulns.rna_vuln_id

表 6-51　rna_host_third_party_vuln_rna_id の結合（続き）

このテーブルで結合に使用するフィールド	結合できるフィールド
host_id	rna_host.host_id rna_host_attribute.host_id rna_host_protocol.host_id rna_host_os_vulns.host_id application_host_map.host_id rna_host_client_app.host_id rna_host_client_app_payload.host_id rna_host_ioc_state.host_id rna_host_ip_map.host_id rna_host_mac_map.host_id rna_host_os.host_id rna_host_sensor.host_id rna_host_service.host_id rna_host_service_banner.host_id rna_host_service_info.host_id rna_host_service_payload.host_id rna_host_service_vulns.host_id

rna_host_third_party_vuln_rna_id のサンプル クエリ

次のクエリは、host_id が 8 のホストのすべてのサードパーティ脆弱性と VDB ID を返します。

```
SELECT host_id, third_party_vuln_id, rna_vuln_id, name, description, source, invalid
FROM rna_host_third_party_vuln_rna_id
WHERE HEX(host_id) = "00000000000000000000000000000008";
```

rna_vuln

rna_vuln テーブルには、シスコ VDB の脆弱性に関する情報が格納されます。

詳細については、次の項を参照してください。

- rna_vuln のフィールド（6-55 ページ）
- rna_vuln の結合（6-56 ページ）
- rna_vuln のサンプル クエリ（6-57 ページ）

rna_vuln のフィールド

次の表に、**rna_vuln** テーブルでアクセスできるフィールドについて説明します。

表 6-52　rna_vuln のフィールド

フィールド	説明
authentication	この脆弱性のエクスプロイトに認証が必要であるかどうか • Required • Not Required • Unknown
availability	脆弱性のエクスプロイトが可能な状況: • Always • User Initiated • Time Dependent • 不明(Unknown)
available_exploits	脆弱性に対して既知のエクスプロイトがあるかどうか • TRUE • FALSE
bugtraq_id	Bugtraq データベースにおいて脆弱性に関連付けられている識別番号。
class	脆弱性のクラス: • Configuration Error • Boundary Condition Error • Design Error
credibility	脆弱性の信頼度: • Conflicting Reports • Conflicting Details • Single Source • Reliable Source • Multiple Sources • Vendor Confirmed
credit	脆弱性を報告したユーザまたは組織
ease	脆弱性のエクスプロイトの容易さ。 • No Exploit Required • Exploit Available • No Exploit Available
effect	脆弱性がエクスプロイトされた場合に発生する可能性のある影響の詳細。
entry_date	脆弱性がデータベースに登録された日付。
exploit	脆弱性のエクスプロイトを確認できる情報。

表 6-52　rna_vuln のフィールド（続き）

フィールド	説明
impact	脆弱性の影響。これは侵入データ、検出イベント、および脆弱性アセスメントの間の相関に基づいて決定した影響レベルに対応しています。指定できる値は 1 ～ 10 です。10 が最も高い重大度です。脆弱性の影響度の値は、Bugtraq エントリの作成者が決定します。
local	脆弱性がローカルでエクスプロイトされる必要があるかどうかを示します。 • TRUE • FALSE
long_description	脆弱性についての一般的な説明。
mitigation	脆弱性を緩和する方法を説明します。
modified_date	脆弱性の最終変更日（該当する場合）。
publish_date	脆弱性が公開された日付。
remote	脆弱性がネットワーク上でエクスプロイト可能であるかどうかを示します。 • TRUE • FALSE
rna_vuln_id	脆弱性を追跡するためにシステムで使用する シスコ の脆弱性 ID 番号。
scenario	攻撃者が脆弱性をエクスプロイトするシナリオの説明。
short_description	脆弱性の概要。
snort_id	Snort ID（SID）データベースにおいて脆弱性に関連付けられている識別番号。つまり、侵入ルールで特定の脆弱性を悪用するネットワーク トラフィックを検出できると、その脆弱性は、侵入ルールの SID に関連付けられます。
solution	脆弱性に対する解決策。
technical_description	脆弱性に関する技術的な説明。
title	脆弱性のタイトル。

rna_vuln の結合

次の表に、**rna_vuln** テーブルで実行できる結合について説明します。

表 6-53　rna_vuln の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
rna_vuln_id または bugtraq_id	rna_host_os_vulns.rna_vuln_id rna_host_service_vulns.rna_vuln_id rna_host_third_party_vuln_rna_id.rna_vuln_id rna_host_third_party_vuln_cve_id.cve_id rna_host_third_party_vuln_bugtraq_id.bugtraq_id

rna_vuln のサンプル クエリ

次のクエリは、最大 25 件の脆弱性に関する情報を返します。これらのレコードは、当該脆弱性に基づいて生成されたイベントの数の順にソートされます。

```
SELECT rna_vuln_id, bugtraq_id, snort_id, title, publish_date, impact, remote, exploit,
long_description, technical_description, solution, count(*) as count
FROM rna_vuln
GROUP BY rna_vuln_id
ORDER BY rna_vuln_id DESC LIMIT 0, 25;
```

tag_info

tag_info テーブルには、ネットワークで検出されたアプリケーションに関連付けられているタグに関する情報が含まれています。アプリケーションには複数のタグが関連付けられていることがある点に注意してください。

詳細については、次の項を参照してください。

- tag_info のフィールド（6-57 ページ）
- tag_info の結合（6-58 ページ）
- tag_info のサンプル クエリ（6-58 ページ）

tag_info のフィールド

次の表に、**tag_info** テーブルでアクセスできるフィールドについて説明します。

表 6-54　tag_info のフィールド

フィールド	説明
domain_name	アプリケーションが検出されたドメインの名前。
domain_uuid	アプリケーションが検出されたドメインの UUID。これはバイナリで示されます。
tag_description	タグの説明。
tag_id	タグの内部 ID。
tag_name	ユーザ インターフェイスに表示されるタグのテキスト。
tag_type	次のいずれかが必要です。 • category • tag

tag_info の結合

次の表に、**tag_info** テーブルで実行できる結合について説明します。

表 6-55　tag_info の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
tag_id	application_tag_map.tag_id

tag_info のサンプル クエリ

次のクエリは、Global \ Company B \ Edge ドメイン内で選択されているタグ ID のアプリケーション タグ レコードを返します。

```
SELECT tag_id, tag_name, tag_type, tag_description

FROM tag_info

WHERE tag_id="100" AND domain_name= "Global \ Company B \ Edge";
```

url_categories

url_categories テーブルには、モニタ対象ネットワークのホストから要求される URL を特徴付けるカテゴリがリストされます。

詳細については、次の項を参照してください。

- url_categories のフィールド (6-58 ページ)
- url_categories の結合 (6-58 ページ)
- url_categories のサンプル クエリ (6-59 ページ)

url_categories のフィールド

次の表では、**url_categories** テーブルのフィールドについて説明します。

表 6-56　url_categories のフィールド

フィールド	説明
category_description	URL カテゴリの説明。
category_id	URL カテゴリの内部識別番号。

url_categories の結合

url_categories テーブルに対して結合を実行することはできません。

url_categories のサンプル クエリ

次のクエリは、選択したカテゴリ ID のカテゴリ レコードを返します。

```
SELECT category_id, category_description
FROM url_categories
WHERE category_id="1";
```

url_reputations

url_reputations テーブルには、モニタ対象要求でホストから要求される URL を特徴付けるレピュテーションがリストされます。

詳細については、次の項を参照してください。

- url_reputations のフィールド（6-59 ページ）
- url_reputations の結合（6-59 ページ）
- url_reputations のサンプル クエリ（6-59 ページ）

url_reputations のフィールド

次の表では、**url_reputations** テーブルのフィールドについて説明します。

表 6-57　url_reputations のフィールド

フィールド	説明
reputation_description	レピュテーションの説明。
reputation_id	URL のレピュテーションの内部識別番号。

url_reputations の結合

url_reputations テーブルに対して結合を実行することはできません。

url_reputations のサンプル クエリ

次のクエリは、特定のレピュテーション ID の URL レピュテーション情報を返します。

```
SELECT reputation_id, reputation_description
FROM url_reputations
WHERE reputation_id="1";
```

user_ipaddr_history

user_ipaddr_history テーブルには、モニタ対象ネットワーク内の特定のホストに対するユーザ アクティビティに関する情報が含まれます。

詳細については、次の項を参照してください。

- user_ipaddr_history のフィールド（6-60 ページ）
- user_ipaddr_history の結合（6-61 ページ）
- user_ipaddr_history のサンプル クエリ（6-61 ページ）

user_ipaddr_history のフィールド

次の表に、**user_ipaddr_history** テーブルでアクセスできるフィールドについて説明します。

表 6-58　user_ipaddr_history のフィールド

フィールド	説明
authentication_type	ユーザが使用する認証のタイプ。値は次のとおりです。 • 0:認証は不要 • 1:パッシブ認証、AD エージェント、または ISE セッション • 2:キャプティブ ポータルの正常な認証 • 3:キャプティブ ポータル ゲスト認証 • 4:キャプティブ ポータルの失敗認証
domain_name	ユーザが検出されたドメインの名前。
domain_uuid	ユーザが検出されたドメインの UUID。これはバイナリで示されます。
endpoint_profile	接続エンドポイントで使用されるデバイスのタイプの名前。
end_time_sec	ホストに異なるユーザがログインしていることを Firepower システム が検出した日時を示す UNIX タイムスタンプ。これにより、以前のユーザのセッションの推定される終了がマークされます。Firepower システム ではログオフが検出されないことに注意してください。
id	ユーザ履歴レコードの内部識別番号。
ipaddr	ホストの IP アドレスのバイナリ表現。
location_ip	ISE と通信するインターフェイスの IP アドレス。IPv4 または IPv6 のアドレスを使用できます。
start_time_sec	ユーザがホストにログインしたことを Firepower システム が検出した日時を示す UNIX タイムスタンプ。
security_group	ネットワーク トラフィック グループの ID 番号。
user_dept	ユーザの所属部門。
user_email	ユーザの電子メール アドレス。
user_first_name	ユーザの名。
user_id	ユーザの内部識別番号。
user_last_name	ユーザの姓。

表 6-58　user_ipaddr_history のフィールド（続き）

フィールド	説明
user_last_seen_sec	ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
user_last_updated_sec	ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。
user_name	ユーザのユーザ名。
user_phone	ユーザの電話番号です。
user_rna_service	ユーザの検出時に使用されていたアプリケーション プロトコルの名前(使用可能な場合)。

user_ipaddr_history の結合

次の表に、**user_ipaddr_history** テーブルで実行できる結合について説明します。

表 6-59　user_ipaddr_history の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
ipaddr	compliance_event.dst_ipaddr compliance_event.src_ipaddr connection_log.initiator_ipaddr connection_log.responder_ipaddr connection_summary.initiator_ipaddr connection_summary.responder_ipaddr fireamp_event.dst_ipaddr fireamp_event.src_ipaddr intrusion_event.dst_ipaddr intrusion_event.src_ipaddr network_discovery_event.ipaddr rna_host_ip_map.ipaddr si_connection_log.initiator_ipaddr si_connection_log.responder_ipaddr user_discovery_event.ipaddr white_list_event.ipaddr
user_id	discovered_users.user_id user_discovery_event.user_id

user_ipaddr_history のサンプル クエリ

次のクエリは、指定された開始タイムスタンプ以降の、選択された IP アドレスのユーザ アクティビティ レコードをすべて返します。

```
SELECT ipaddr, start_time_sec, end_time_sec, user_name, user_rna_service,
user_last_seen_sec, user_last_updated_sec

FROM user_ipaddr_history

WHERE HEX(ipaddr) = "00000000000000000000FFFF0A0A0A04" AND start_time_sec >=
UNIX_TIMESTAMP("2011-10-01 00:00:00");
```

スキーマ：接続ログ テーブル

この章では、接続データのスキーマとサポートされている結合について説明します。

詳細については、次の表に示す項を参照してください。「バージョン」欄は、示されている各テーブルでサポートされているデータベース アクセスのバージョンを示します。

表 7-1　　接続ログ テーブルのスキーマ

参照先	次の内容が格納されるテーブル	バージョン
connection_log（7-1 ページ）	個別の接続。廃止されたテーブル `rna_flow` を置き換えます。	5.0+
connection_summary（7-13 ページ）	接続ログのサマリ。廃止されたテーブル `rna_flow_summary` を置き換えます。	5.0+
si_connection_log（7-17 ページ）	個別の接続。セキュリティ インテリジェンスに使用されます。	5.3+

connection_log

`connection_log` テーブルには、接続イベントに関する情報が格納されます。Firepower システムは、モニタ対象ホストとその他のホストの間で接続が確立されると接続イベントを生成します。このイベントには、モニタ対象トラフィックに関する詳細情報が含まれています。

`connection_log` テーブルは、Firepower システム バージョン 5.0 以降で廃止されたテーブル `rna_flow` を置き換えます。

詳細については、次の項を参照してください。

- connection_log のフィールド（7-2 ページ）
- connection_log の結合（7-13 ページ）
- connection_log のサンプル クエリ（7-13 ページ）

connection_log のフィールド

次の表に、**connection_log** テーブルでアクセスできるデータベース フィールドについて説明します。

表 7-2　connection_log のフィールド

フィールド	説明
access_control_policy_name	接続をログに記録したアクセス コントロール ルール(またはデフォルト アクション)を含むアクセス コントロール ポリシー。
access_control_policy_UUID	接続をログに記録したアクセス コントロール ルール(またはデフォルト アクション)を含むアクセス コントロール ポリシーの UUID。
access_control_reason	アクセス コントロール ルールによって接続がログに記録された理由。次の 1 つまたは複数が表示されます。 • IP Block • IP Monitor • User Bypass • File Monitor • File Block • Intrusion Monitor • Intrusion Block • File Resume Block • File Resume Allow • File Custom Detection • SSL Block • DNS Block • DNS Monitor • URL Block • URL Monitor • HTTP Injection • Intelligent App Bypass • ログに記録された接続がない場合は空白
access_control_rule_action	アクセス コントロール ルールに関連付けられているアクション(またはデフォルト アクション):allow、block など。
access_control_rule_id	ルールの内部識別番号。
access_control_rule_name	接続をログに記録したアクセス コントロール ルール(またはデフォルト アクション)。
application_protocol_id	アプリケーション プロトコルの内部識別番号。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
application_protocol_name	次のいずれかになります。 • アプリケーションの名前（確実な識別が可能な場合） • unknown（システムが既知のサーバ フィンガープリントに基づいてサーバを識別できない場合） • pending（システムがさらにデータを必要としている場合） • 空白（接続にアプリケーション情報がない場合）
bytes_recv	セッション レスポンダが送信した合計バイト数。
bytes_sent	セッション イニシエータが送信した合計バイト数。
cert_valid_end_date	接続に使用された SSL 証明書の有効期限を示す UNIX タイムスタンプ。
cert_valid_start_date	接続で使用された SSL 証明書の発行時点を示す UNIX タイムスタンプ。
client_application_id	侵入イベントで使用されたクライアント アプリケーションの内部識別番号。
client_application_name	侵入イベントで使用されたクライアント アプリケーション（使用可能な場合）。次のいずれかになります。 • アプリケーションの名前（確実な識別が可能な場合）。 • 汎用クライアント名（システムがクライアント アプリケーションを検出したが、特定のアプリケーションであることを識別できない場合）。 • 空白（接続にクライアント アプリケーション情報がない場合）。
client_application_version	クライアント アプリケーションのバージョン。
connection_type	接続情報の検出ソース。次のいずれかを行います。 • rna:シスコ デバイスにより検出される場合 • netflow:NetFlow 対応デバイスによりエクスポートされる場合
counter	接続イベントに関連する侵入イベントのカウンタ。
dns_ttl	DNS レスポンスの存続期間（秒単位）。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
dns_response	DNS 応答。有効な値は次のとおりです。 • 0:NoError — エラーなし • 1:FormErr — フォーマット エラー • 2:ServFail — サーバ障害 • 3:NXDomain — 存在していないドメイン • 4:NotImp — 未実装 • 5:Refused — クエリ拒否 • 6:YXDomain — 名前が存在してはならない状況で存在している • 7:YXRRSet — RR セットが存在してはならない状況で存在している • 8:NXRRSet — 存在しているべき RR セットが存在していない • 9:NotAuth — 未承認 • 10:NotZone — 名前がゾーンに含まれていない • 16:BADSIG — TSIG 署名失敗 • 17:BADKEY — キーが認識されない • 18:BADTIME — 時間範囲外の署名 • 19:BADMODE — 不適切な TKEY モード • 20:BADNAME — 重複するキー名 • 21:BADALG — サポートされていないアルゴリズム • 22:BADTRUNC — 不適切な切り捨て • 3841:NXDOMAIN — ファイアウォールからの NXDOMAIN 応答 • 3842:SINKHOLE — ファイアウォールからのシンクホール応答
domain_name	セッションのドメインの名前。
domain_uuid	セッションのドメインの UUID。これはバイナリで示されます。
endpoint_profile	接続エンドポイントで使用されるデバイスのタイプの名前。
file_count	セッションで Snort によって識別されたファイルの数。セッションで識別されるファイルごと 1 件のレコードが生成されます。
first_packet_sec	セッションの最初のパケットが検出された日時を示す UNIX タイムスタンプ。
flow_id	接続の内部識別番号。
http_response_code	接続での HTTP 要求に対する応答コード。
hostname_in_query	接続が DNS クエリの場合に使用されるホスト名。
icmp_code	イベントが ICMP トラフィックの場合は ICMP コード。イベントが ICMP トラフィックから生成されたものではない場合は null。
icmp_type	イベントが ICMP トラフィックの場合は ICMP タイプ。イベントが ICMP トラフィックから生成されたものではない場合は null。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
initiator_continent_name	セッションを開始したホストが位置する地域の名前。 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
initiator_country_id	セッションを開始したホストの国のコード。
initiator_country_name	セッションを開始したホストの国の名前。
initiator_ip	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
initiator_ip_address	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
initiator_ipaddr	セッションを開始したホストの IP アドレスのバイナリ表現。
initiator_ipv4	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
initiator_port	セッション イニシエータが使用するポート。
initiator_user_dept	イニシエータ ホストの最終ログイン ユーザが所属する部門。
initiator_user_email	イニシエータ ホストの最終ログイン ユーザの電子メールアドレス。
initiator_user_first_name	イニシエータ ホストの最終ログイン ユーザの名前。
initiator_user_id	イニシエータ ホストの最終ログイン ユーザの内部識別番号。
initiator_user_last_name	イニシエータ ホストの最終ログイン ユーザの姓。
initiator_user_last_seen_sec	イニシエータ ホストの最終ログイン ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
initiator_user_last_updated_sec	イニシエータ ホストの最終ログイン ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。
initiator_user_name	イニシエータ ホストの最終ログイン ユーザのユーザ名。
initiator_user_phone	イニシエータ ホストの最終ログイン ユーザの電話番号。
instance_id	イベントを生成した管理対象デバイスの Snort インスタンスの数値 ID。
interface_egress_name	接続に関連付けられた入力インターフェイス。
interface_ingress_name	接続に関連付けられた出力インターフェイス。
ioc_count	接続で検出された侵害の痕跡の数。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
ips_event_count	侵入イベントのしきい値に達するまでに、接続で生成された侵入イベントの数。
last_packet_sec	セッションの最後のパケットが検出された日時を示す UNIX タイムスタンプ。
location_ip	ISE と通信するインターフェイスの IP アドレス。IPv4 または IPv6 のアドレスを使用できます。
monitor_rule_id_1	接続に関連付けられている 1 番目のモニタ ルールの ID。この ID は、monitor_rule_name_1 に格納されている名前に関連付けられています。
monitor_rule_id_2	接続に関連付けられている 2 番目のモニタ ルールの ID。この ID は、monitor_rule_name_2 に格納されている名前に関連付けられています。
monitor_rule_id_3	接続に関連付けられている 3 番目のモニタ ルールの ID。この ID は、monitor_rule_name_3 に格納されている名前に関連付けられています。
monitor_rule_id_4	接続に関連付けられている 4 番目のモニタ ルールの ID。この ID は、monitor_rule_name_4 に格納されている名前に関連付けられています。
monitor_rule_id_5	接続に関連付けられている 5 番目のモニタ ルールの ID。この ID は、monitor_rule_name_5 に格納されている名前に関連付けられています。
monitor_rule_id_6	接続に関連付けられている 6 番目のモニタ ルールの ID。この ID は、monitor_rule_name_6 に格納されている名前に関連付けられています。
monitor_rule_id_7	接続に関連付けられている 7 番目のモニタ ルールの ID。この ID は、monitor_rule_name_7 に格納されている名前に関連付けられています。
monitor_rule_id_8	接続に関連付けられている 8 番目のモニタ ルールの ID。この ID は、monitor_rule_name_8 に格納されている名前に関連付けられています。
monitor_rule_name_1	接続に関連付けられている 1 番目のモニタ ルールの名前。この名前は、monitor_rule_id_1 に格納されている ID に関連付けられています。
monitor_rule_name_2	接続に関連付けられている 2 番目のモニタ ルールの名前。この名前は、monitor_rule_id_2 に格納されている ID に関連付けられています。
monitor_rule_name_3	接続に関連付けられている 3 番目のモニタ ルールの名前。この名前は、monitor_rule_id_3 に格納されている ID に関連付けられています。
monitor_rule_name_4	接続に関連付けられている 4 番目のモニタ ルールの名前。この名前は、monitor_rule_id_4 に格納されている ID に関連付けられています。
monitor_rule_name_5	接続に関連付けられている 5 番目のモニタ ルールの名前。この名前は、monitor_rule_id_5 に格納されている ID に関連付けられています。
monitor_rule_name_6	接続に関連付けられている 6 番目のモニタ ルールの名前。この名前は、monitor_rule_id_6 に格納されている ID に関連付けられています。
monitor_rule_name_7	接続に関連付けられている 7 番目のモニタ ルールの名前。この名前は、monitor_rule_id_7 に格納されている ID に関連付けられています。
monitor_rule_name_8	接続に関連付けられている 8 番目のモニタ ルールの名前。この名前は、monitor_rule_id_8 に格納されている ID に関連付けられています。
netbios_domain	接続で使用された NetBIOS ドメイン。
netflow_dst_as	宛先の NetFlow 自律システム番号、起点またはピア
netflow_dst_mask	Netflow 宛先アドレス プレフィックス マスク。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
netflow_dst_tos	パケットが宛先から送信元に流れる場合の IP ヘッダーのタイプ オブ サービス（ToS）。
netflow_snmp_in	送信元から宛先へ流れるパケットが使用するインターフェイスの ID。
netflow_snmp_out	宛先から送信元へ流れるパケットが使用するインターフェイスの ID。
netflow_src_as	送信元の NetFlow 自律システム番号、起点またはピア
netflow_src_mask	Netflow 送信元アドレス プレフィックス マスク。
netflow_src_tos	パケットが送信元から宛先に流れる場合の IP ヘッダーのタイプ オブ サービス（ToS）。
network_analysis_policy_name	侵入イベントを生成した侵入ポリシーに関連付けられているネットワーク分析ポリシー
network_analysis_policy_UUID	侵入イベントを生成した侵入ポリシーに関連付けられているネットワーク分析ポリシーの UUID
packets_recv	セッションを開始したホストが受信したパケットの総数。
packets_sent	セッションを開始したホストが送信したパケットの総数。
protocol_name	接続で使用されたプロトコルの名前。
protocol_num	プロトコルの IANA 番号。IANA 番号のリストは http://www.iana.org/assignments/protocol-numbers にあります。
responder_continent_name	セッション イニシエータに対して応答したホストの所在地の地域の名前。 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
responder_country_id	セッション イニシエータに対して応答したホストの国のコード。
responder_country_name	セッション イニシエータに対して応答したホストの国の名前。
responder_ip	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
responder_ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
responder_ipaddr	セッション イニシエータに対して応答したホストの IPv4 または IP v 6 アドレスのバイナリ表現。
responder_ipv4	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
responder_port	セッション レスポンダが使用するポート。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
responder_user_dept	セッション イニシエータに対して応答したホストの最終ログイン ユーザの所属部門。
responder_user_email	セッション イニシエータに対して応答したホストの最終ログイン ユーザの電子メールアドレス。
responder_user_first_name	セッション イニシエータに対して応答したホストの最終ログイン ユーザの名前。
responder_user_id	セッション イニシエータに対して応答したホストの最終ログイン ユーザの内部識別番号。
responder_user_last_name	セッション イニシエータに対して応答したホストの最終ログイン ユーザの姓。
responder_user_last_seen_sec	セッション イニシエータに対して応答したホストの最終ログイン ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
responder_user_last_updated_sec	セッション イニシエータに対して応答したホストの最終ログイン ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。
responder_user_name	セッション イニシエータに対して応答したホストの最終ログイン ユーザのユーザ名。
responder_user_phone	セッション イニシエータに対して応答したホストの最終ログイン ユーザの電話番号。
security_context	トラフィックが通過したセキュリティ コンテキスト（仮想ファイアウォール）の説明。システムがこのフィールドにデータを設定するのは、マルチ コンテキスト モードの ASA FirePOWER デバイスの場合だけです。
security_group	ネットワーク トラフィック グループの ID 番号。
security_intelligence_category	接続に関連付けられているセキュリティ インテリジェンス カテゴリ。
security_intelligence_ip	セキュリティ インテリジェンスによりモニタされる、接続に関連付けられている IP アドレスが、送信元 IP（src）と宛先 IP（dst）のいずれであるか。
security_zone_egress_name	接続イベントの出力セキュリティ ゾーン。
security_zone_ingress_name	接続イベントの入力セキュリティ ゾーン。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は ipv4 address,ipv6 address です。
sensor_name	セッションをモニタした管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
sinkhole	シンクホール オブジェクトに関連付けられているリビジョン UUID。
source_device	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
src_device_ip	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
src_device_ipaddr	次のいずれかを行います。 • 接続データをエクスポートした NetFlow 対応デバイスの IP アドレスのバイナリ表現 • 0（シスコ 管理対象デバイスにより検出される接続の場合）。
src_device_ipv4	• バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
ssl_actual_action	SSL ルールに基づいて接続に対して実行されたアクション。ルールに指定されているアクションが不可能なことがあるため、これは予期していたアクションとは異なることがあります。有効な値は次のとおりです。 • Unknown • Do Not Decrypt • block • Block With Reset • Decrypt (Known Key) • Decrypt (Replace Key) • Decrypt (Resign)
ssl_cipher_suite	SSL 接続で使用される暗号スイート。値は 10 進形式で格納されます。値によって示される暗号スイートについては、 www.iana.org/assignments/tls-parameters/tls-parameters.xhtml を参照してください。
ssl_expected_action	SSL ルールに基づいて接続に対して実行する必要があるアクション。有効な値は次のとおりです。 • *Unknown* • *Do Not Decrypt* • *block* • *Block With Reset* • *Decrypt (Known Key)* • *Decrypt (Replace Key)* • *Decrypt (Resign)*
ssl_flow_flags	暗号化接続のデバッグ レベル フラグ。有効な値は次のとおりです。 • 0x00000001：NSE_FLOW__VALID — 他のフィールドを有効にするために設定する必要があります • 0x00000002：NSE_FLOW__INITIALIZED — 内部構造が処理可能です • 0x00000004：NSE_FLOW__INTERCEPT — SSL セッションが代行受信されました

表 7-2　connection_log のフィールド（続き）

フィールド	説明
ssl_flow_messages	SSL ハンドシェイク時にクライアントとサーバ間で交換されたメッセージ。詳細については、http://tools.ietf.org/html/rfc5246 を参照してください。 • 0x00000001:NSE_MT__HELLO_REQUEST • 0x00000002:NSE_MT__CLIENT_ALERT • 0x00000004:NSE_MT__SERVER_ALERT • 0x00000008:NSE_MT__CLIENT_HELLO • 0x00000010:NSE_MT__SERVER_HELLO • 0x00000020:NSE_MT__SERVER_CERTIFICATE • 0x00000040:NSE_MT__SERVER_KEY_EXCHANGE • 0x00000080:NSE_MT__CERTIFICATE_REQUEST • 0x00000100:NSE_MT__SERVER_HELLO_DONE • 0x00000200:NSE_MT__CLIENT_CERTIFICATE • 0x00000400:NSE_MT__CLIENT_KEY_EXCHANGE • 0x00000800:NSE_MT__CERTIFICATE_VERIFY • 0x00001000:NSE_MT__CLIENT_CHANGE_CIPHER_SPEC • 0x00002000:NSE_MT__CLIENT_FINISHED • 0x00004000:NSE_MT__SERVER_CHANGE_CIPHER_SPEC • 0x00008000:NSE_MT__SERVER_FINISHED • 0x00010000:NSE_MT__NEW_SESSION_TICKET • 0x00020000:NSE_MT__HANDSHAKE_OTHER • 0x00040000:NSE_MT__APP_DATA_FROM_CLIENT • 0x00080000:NSE_MT__APP_DATA_FROM_SERVER

表 7-2　connection_log のフィールド（続き）

フィールド	説明
ssl_flow_status	SSL フローのステータス。アクションが実行された理由、またはエラー メッセージが出された理由を示す値です。有効な値は次のとおりです。 • 'Unknown' • 'No Match' • 'Success' • 'Uncached Session' • 'Unknown Cipher Suite' • 'Unsupported Cipher Suite' • 'Unsupported SSL Version' • 'SSL Compression Used' • 'Session Undecryptable in Passive Mode' • 'Handshake Error' • 'Decryption Error' • 'Pending Server Name Category Lookup' • 'Pending Common Name Category Lookup' • 'Internal Error' • 'Network Parameters Unavailable' • 'Invalid Server Certificate Handle' • 'Server Certificate Fingerprint Unavailable' • 'Cannot Cache Subject DN' • 'Cannot Cache Issuer DN' • 'Unknown SSL Version' • 'External Certificate List Unavailable' • 'External Certificate Fingerprint Unavailable' • 'Internal Certificate List Invalid' • 'Internal Certificate List Unavailable' • 'Internal Certificate Unavailable' • 'Internal Certificate Fingerprint Unavailable' • 'Server Certificate Validation Unavailable' • 'Server Certificate Validation Failure' • 'Invalid Action'
ssl_issuer_common_name	SSL 証明書の発行元の共通名。これは一般に証明書発行元のホストとドメイン名ですが、その他の情報が含まれていることもあります。
ssl_issuer_country	SSL 証明書の発行元の国。
ssl_issuer_organization	SSL 証明書の発行元の組織。
ssl_issuer_organization_unit	SSL 証明書の発行元の組織単位。
ssl_policy_action	ルールが一致しない場合のためにポリシーで設定されているデフォルト アクション。

表 7-2　connection_log のフィールド（続き）

フィールド	説明
ssl_policy_name	接続を処理した SSL ポリシーの ID 番号。
ssl_policy_reason	SSL ポリシーが SSL セッションをログに記録した理由。
ssl_rule_action	SSL ルールに対しユーザ インターフェイスで選択されたアクション（allow、block など）。
ssl_rule_name	接続を処理した SSL ルールまたはデフォルト アクションの ID 番号。
ssl_serial_number	発行元 CA によって割り当てられた SSL 証明書のシリアル番号。
ssl_server_name	SSL Client Hello でサーバ名に指定された名前。
ssl_subject_common_name	SSL 証明書の件名共通名。これは一般に証明書の件名のホストとドメイン名ですが、その他の情報が含まれていることもあります。
ssl_subject_country	SSL 証明書の件名の国。
ssl_subject_organization	SSL 証明書の件名の組織。
ssl_subject_organization_unit	SSL 証明書の件名の組織単位。
ssl_url_category	サーバ名と証明書の共通名から識別されるフローのカテゴリ。
ssl_version	接続の暗号化に使用された SSL または TLS プロトコル バージョン。
tcp_flags	セッションで検出された TCP フラグ。
url	セッション中にモニタ対象ホストによって要求された URL（使用可能な場合）。
url_category	モニタ対象ホストによって要求された URL のカテゴリ。
url_reputation	モニタ対象ホストによって要求された URL のレピュテーション。次のいずれかが必要です。 • 1:高リスク • 2:疑わしいサイト • 3:セキュリティ リスクのある無害なサイト • 4:無害なサイト • 5:既知
web_application_id	Web アプリケーションの内部識別番号。
web_application_name	次のいずれかになります。 • アプリケーションの名前（確実な識別が可能な場合）。 • web browsing（システムがアプリケーション プロトコル HTTP を検出したが、特定の Web アプリケーションを検出できない場合）。 • 空白（接続に HTTP トラフィックがない場合）。

connection_log の結合

次の表に、connection_log テーブルを使用して実行できる結合について説明します。

表 7-3　connection_log の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_protocol_id または client_application_id または web_application_id	application_info.application_id application_host_map.application_id application_tag_map.application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id
initiator_ipaddr または responder_ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

connection_log のサンプル クエリ

次のクエリは、connection_log テーブルから最大 25 件の接続イベント レコードを、パケットのタイムスタンプに基づいて降順にソートして返します。

```
SELECT first_packet_sec, last_packet_sec, initiator_ipaddr, responder_ipaddr,
security_zone_ingress_name, security_zone_egress_name, initiator_port, protocol_name,
responder_port, application_protocol_id, client_application_id, web_application_id, url,
url_category, url_reputation

FROM connection_log

WHERE first_packet_sec <= UNIX_TIMESTAMP("2011-10-01 00:00:00" ) ORDER BY
first_packet_sec

DESC, last_packet_sec DESC LIMIT 0, 25;
```

connection_summary

connection_summary テーブルには、接続サマリまたは集約された接続に関する情報が格納されます。Firepower システム は、5 分間隔で接続を集約します。接続を集約対象にするには、接続が次の条件に対応している必要があります。

- 同一の送信元の IP アドレスと宛先 IP アドレスである
- 同じプロトコルを使用している
- 同じアプリケーションを使用している
- 同じ管理対象デバイスにより検出されているか（Firepower での管理対象デバイスにより検出されたセッションの場合）、または同じ NetFlow 対応デバイスによりエクスポートされ、同じ管理対象デバイスにより処理されている

接続サマリの集約データには、イニシエータ ホストとレスポンダ ホストにより送信されたパケットとバイトの総数と、サマリに含まれる接続の数などがあります。

connection_summary テーブルは、Firepower システム バージョン 5.0 以降で廃止されたテーブル **rna_flow_summary** を置き換えます。

詳細については、次の項を参照してください。

- connection_summary のフィールド（7-14 ページ）
- connection_summary の結合（7-16 ページ）
- connection_summary のサンプル クエリ（7-17 ページ）

connection_summary のフィールド

次の表に、**connection_summary** テーブルでアクセスできるデータベース フィールドについて説明します。

表 7-4　connection_summary のフィールド

フィールド	説明
application_protocol_id	アプリケーション プロトコルの内部識別番号。
application_protocol_name	次のいずれかになります。 • アプリケーションの名前（確実な識別が可能な場合） • unknown（システムが既知のサーバ フィンガープリントに基づいてサーバを識別できない場合） • pending（システムがさらにデータを必要としている場合） • 空白（接続にアプリケーション情報がない場合）
bytes_recv	セッション レスポンダが送信した合計バイト数。
bytes_sent	セッション イニシエータが送信した合計バイト数。
connection_type	接続情報の検出ソース。次のいずれかを行います。 • rna：シスコ デバイスにより検出される場合 • netflow：NetFlow 対応デバイスによりエクスポートされる場合
domain_name	セッションのドメインの名前。
domain_uuid	セッションのドメインの UUID。これはバイナリで示されます。
flow_type	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
id	接続サマリの内部識別番号。
initiator_ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
initiator_ipaddr	セッションを開始したホストの IP アドレスのバイナリ表現。
initiator_user_dept	イニシエータ ホストの最終ログイン ユーザが所属する部門。
initiator_user_email	イニシエータ ホストの最終ログイン ユーザの電子メールアドレス。
initiator_user_first_name	イニシエータ ホストの最終ログイン ユーザの名前。
initiator_user_id	イニシエータ ホストの最終ログイン ユーザの内部識別番号。

表 7-4　connection_summary のフィールド（続き）

フィールド	説明
initiator_user_last_name	イニシエータ ホストの最終ログイン ユーザの姓。
initiator_user_last_seen_sec	イニシエータ ホストの最終ログイン ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
initiator_user_last_updated_sec	イニシエータ ホストの最終ログイン ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。
initiator_user_name	イニシエータ ホストの最終ログイン ユーザのユーザ名。
initiator_user_phone	イニシエータ ホストの最終ログイン ユーザの電話番号。
interface_egress_name	接続に関連付けられた入力インターフェイス。
interface_ingress_name	接続に関連付けられた出力インターフェイス。
netmap_num	接続が検出されたドメインの Netmap ID。
num_connections	サマリに含まれる接続の数。長時間接続（複数の接続サマリ間隔にわたる接続）の場合、最初の接続サマリ間隔の分だけ増加します。
packets_recv	セッション レスポンダが送信した合計パケット数。
packets_sent	セッション イニシエータが送信した合計パケット数。
protocol_name	集約セッションで使用されたプロトコルの名前。
protocol_num	プロトコルの IANA 番号。IANA 番号のリストは http://www.iana.org/assignments/protocol-numbers にあります。
responder_ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
responder_ipaddr	集約されたセッションのイニシエータに応答したホストの IP アドレスのバイナリ表現。
responder_port	集約セッションでレスポンダが使用したポート。
responder_user_dept	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザの所属部門。
responder_user_email	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザの電子メールアドレス。
responder_user_first_name	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザの名前。
responder_user_id	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザの内部識別番号
responder_user_last_name	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザの姓。
responder_user_last_seen_sec	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
responder_user_last_updated_sec	セッション イニシエータに対して応答したホストの最終ログイン ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。

表 7-4　connection_summary のフィールド（続き）

フィールド	説明
responder_user_name	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザのユーザ名。
responder_user_phone	集約セッションのイニシエータに対して応答したホストの最終ログイン ユーザの電話番号。
security_zone_egress_name	接続イベントの出力セキュリティ ゾーン。
security_zone_ingress_name	接続イベントの入力セキュリティ ゾーン。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。
sensor_name	集約セッションをモニタした管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
source_device	送信元デバイスの ID。これは次のいずれかです。 • 接続のデータをエクスポートした NetFlow 対応デバイスの IP アドレス。 • Firepower（接続が シスコ 管理対象デバイスにより検出された場合）
start_time_sec	サマリのセッション集約に使用される 5 分間の間隔の開始時点の日時を示す UNIX タイムスタンプ。

connection_summary の結合

次の表に、**connection_summary** テーブルを使用して実行できる結合について説明します。

表 7-5　connection_summary の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_protocol_id	application_info.application_id application_host_map.application_id application_tag_map.application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id
initiator_ipaddr または responder_ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

connection_summary のサンプル クエリ

次のクエリは、選択されたデバイスにより検出されたイベント サマリ レコードを最大 5 件返します。

```
SELECT initiator_ipaddr, responder_ipaddr, protocol_name, application_protocol_id,
source_device, sensor_name, sensor_address, packets_recv, packets_sent, bytes_recv,
bytes_sent, connection_type, num_connections

FROM connection_summary

WHERE sensor_name='linden' limit 5;
```

si_connection_log

si_connection_log テーブルには、セキュリティ インテリジェンス イベントに関する情報が格納されます。Firepower システム は、セキュリティ インテリジェンスにより接続がブラックリストに追加またはモニタされるときに、セキュリティ インテリジェンス イベントを生成します。このイベントには、モニタ対象トラフィックに関する詳細情報が含まれています。

詳細については、次の項を参照してください。

- si_connection_log のフィールド（7-17 ページ）
- si_connection_log の結合（7-28 ページ）
- si_connection_log のサンプル クエリ（7-28 ページ）

si_connection_log のフィールド

次の表に、**si_connection_log** テーブルでアクセスできるデータベース フィールドについて説明します。

表 7-6　si_connection_log のフィールド

フィールド	説明
access_control_policy_name	接続をログに記録したアクセス コントロール ルール（またはデフォルト アクション）を含むアクセス コントロール ポリシー。
access_control_policy_UUID	接続をログに記録したアクセス コントロール ルール（またはデフォルト アクション）を含むアクセス コントロール ポリシーの UUID。

表 7-6　si_connection_log のフィールド(続き)

フィールド	説明
access_control_reason	アクセス コントロール ルールによって接続がログに記録された理由。次の 1 つまたは複数が表示されます。 • IP Block • IP Monitor • User Bypass • File Monitor • File Block • Intrusion Monitor • Intrusion Block • File Resume Block • File Resume Allow • File Custom Detection • SSL Block • DNS Block • DNS Monitor • URL Block • URL Monitor • HTTP Injection • Intelligent App Bypass • ログに記録された接続がない場合は空白
access_control_rule_action	アクセス コントロール ルールに関連付けられているアクション(またはデフォルト アクション):allow、block など。
access_control_rule_id	ルールの内部識別番号。
access_control_rule_name	接続をログに記録したアクセス コントロール ルール(またはデフォルト アクション)。
application_protocol_id	アプリケーション プロトコルの内部識別番号。
application_protocol_name	次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合) • unknown(システムが既知のサーバ フィンガープリントに基づいてサーバを識別できない場合) • pending(システムがさらにデータを必要としている場合) • 空白(接続にアプリケーション情報がない場合)
bytes_recv	セッション レスポンダが送信した合計バイト数。
bytes_sent	セッション イニシエータが送信した合計バイト数。
cert_valid_end_date	接続に使用された SSL 証明書の有効期限を示す UNIX タイムスタンプ。
cert_valid_start_date	接続で使用された SSL 証明書の発行時点を示す UNIX タイムスタンプ。
client_application_id	侵入イベントで使用されたクライアント アプリケーションの内部識別番号。

表 7-6　si_connection_log のフィールド(続き)

フィールド	説明
client_application_name	侵入イベントで使用されたクライアント アプリケーション(使用可能な場合)。次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合) • 汎用クライアント名(システムがクライアント アプリケーションを検出したが、特定のアプリケーションであることを識別できない場合)。 • 空白(接続にクライアント アプリケーション情報がない場合)。
client_application_version	クライアント アプリケーションのバージョン。
connection_type	接続情報の検出ソース。次のいずれかを行います。 • rna:シスコ デバイスにより検出される場合 • netflow:NetFlow 対応デバイスによりエクスポートされる場合
counter	接続イベントに関連する侵入イベントのカウンタ。
dns_ttl	DNS レスポンスの存続期間(秒単位)。
dns_response	DNS 応答。有効な値は次のとおりです。 • 0:NoError — エラーなし • 1:FormErr — フォーマット エラー • 2:ServFail — サーバ障害 • 3:NXDomain — 存在していないドメイン • 4:NotImp — 未実装 • 5:Refused — クエリ拒否 • 6:YXDomain — 名前が存在してはならない状況で存在している • 7:YXRRSet — RR セットが存在してはならない状況で存在している • 8:NXRRSet — 存在しているべき RR セットが存在していない • 9:NotAuth — 未承認 • 10:NotZone — 名前がゾーンに含まれていない • 16:BADSIG — TSIG 署名失敗 • 17:BADKEY — キーが認識されない • 18:BADTIME — 時間範囲外の署名 • 19:BADMODE — 不適切な TKEY モード • 20:BADNAME — 重複するキー名 • 21:BADALG — サポートされていないアルゴリズム • 22:BADTRUNC — 不適切な切り捨て • 3841:NXDOMAIN — ファイアウォールからの NXDOMAIN 応答 • 3842:SINKHOLE — ファイアウォールからのシンクホール応答
domain_name	セッションのドメインの名前。

表 7-6　si_connection_log のフィールド(続き)

フィールド	説明
domain_uuid	セッションのドメインの UUID。これはバイナリで示されます。
endpoint_profile	接続エンドポイントで使用されるデバイスのタイプの名前。
file_count	セッションで Snort によって識別されたファイルの数。セッションで識別されるファイルごと 1 件のレコードが生成されます。
first_packet_sec	セッションの最初のパケットが検出された日時を示す UNIX タイムスタンプ。
http_response_code	接続での HTTP 要求に対する応答コード。
hostname_in_query	接続が DNS クエリの場合に使用されるホスト名。
icmp_code	イベントが ICMP トラフィックの場合は ICMP コード。イベントが ICMP トラフィックから生成されたものではない場合は null。
icmp_type	イベントが ICMP トラフィックの場合は ICMP タイプ。イベントが ICMP トラフィックから生成されたものではない場合は null。
initiator_continent_name	セッションを開始したホストの所在地の地域の名前。 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
initiator_country_id	セッションを開始したホストの国のコード。
initiator_country_name	セッションを開始したホストの国の名前。
initiator_ipaddr	セッションを開始したホストの IP アドレスのバイナリ表現。
initiator_port	セッション イニシエータが使用するポート。
initiator_user_dept	イニシエータ ホストの最終ログイン ユーザが所属する部門。
initiator_user_email	イニシエータ ホストの最終ログイン ユーザの電子メールアドレス。
initiator_user_first_name	イニシエータ ホストの最終ログイン ユーザの名前。
initiator_user_id	イニシエータ ホストの最終ログイン ユーザの内部識別番号。
initiator_user_last_name	イニシエータ ホストの最終ログイン ユーザの姓。
initiator_user_last_seen_sec	イニシエータ ホストの最終ログイン ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
initiator_user_last_updated_sec	イニシエータ ホストの最終ログイン ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。
initiator_user_name	イニシエータ ホストの最終ログイン ユーザのユーザ名。
initiator_user_phone	イニシエータ ホストの最終ログイン ユーザの電話番号。

表 7-6　si_connection_log のフィールド（続き）

フィールド	説明
instance_id	イベントを生成した管理対象デバイスの Snort インスタンスの数値 ID。
interface_egress_name	接続に関連付けられた入力インターフェイス。
interface_ingress_name	接続に関連付けられた出力インターフェイス。
ioc_count	接続で検出された侵害の痕跡の数。
ips_event_count	侵入イベントのしきい値に達するまでに、接続で生成された侵入イベントの数。
last_packet_sec	セッションの最後のパケットが検出された日時を示す UNIX タイムスタンプ。
location_ip	ISE と通信するインターフェイスの IP アドレス。IPv4 または IPv6 のアドレスを使用できます。
monitor_rule_id_1	接続に関連付けられている 1 番目のモニタ ルールの ID。この ID は、monitor_rule_name_1 に格納されている名前に関連付けられています。
monitor_rule_id_2	接続に関連付けられている 2 番目のモニタ ルールの ID。この ID は、monitor_rule_name_2 に格納されている名前に関連付けられています。
monitor_rule_id_3	接続に関連付けられている 3 番目のモニタ ルールの ID。この ID は、monitor_rule_name_3 に格納されている名前に関連付けられています。
monitor_rule_id_4	接続に関連付けられている 4 番目のモニタ ルールの ID。この ID は、monitor_rule_name_4 に格納されている名前に関連付けられています。
monitor_rule_id_5	接続に関連付けられている 5 番目のモニタ ルールの ID。この ID は、monitor_rule_name_5 に格納されている名前に関連付けられています。
monitor_rule_id_6	接続に関連付けられている 6 番目のモニタ ルールの ID。この ID は、monitor_rule_name_6 に格納されている名前に関連付けられています。
monitor_rule_id_7	接続に関連付けられている 7 番目のモニタ ルールの ID。この ID は、monitor_rule_name_7 に格納されている名前に関連付けられています。
monitor_rule_id_8	接続に関連付けられている 8 番目のモニタ ルールの ID。この ID は、monitor_rule_name_8 に格納されている名前に関連付けられています。
monitor_rule_name_1	接続に関連付けられている 1 番目のモニタ ルールの名前。この名前は、monitor_rule_id_1 に格納されている ID に関連付けられています。
monitor_rule_name_2	接続に関連付けられている 2 番目のモニタ ルールの名前。この名前は、monitor_rule_id_2 に格納されている ID に関連付けられています。
monitor_rule_name_3	接続に関連付けられている 3 番目のモニタ ルールの名前。この名前は、monitor_rule_id_3 に格納されている ID に関連付けられています。
monitor_rule_name_4	接続に関連付けられている 4 番目のモニタ ルールの名前。この名前は、monitor_rule_id_4 に格納されている ID に関連付けられています。
monitor_rule_name_5	接続に関連付けられている 5 番目のモニタ ルールの名前。この名前は、monitor_rule_id_5 に格納されている ID に関連付けられています。
monitor_rule_name_6	接続に関連付けられている 6 番目のモニタ ルールの名前。この名前は、monitor_rule_id_6 に格納されている ID に関連付けられています。
monitor_rule_name_7	接続に関連付けられている 7 番目のモニタ ルールの名前。この名前は、monitor_rule_id_7 に格納されている ID に関連付けられています。

表 7-6　si_connection_log のフィールド(続き)

フィールド	説明
monitor_rule_name_8	接続に関連付けられている 8 番目のモニタ ルールの名前。この名前は、monitor_rule_id_8 に格納されている ID に関連付けられています。
netbios_domain	接続で使用された NetBIOS ドメイン。
netflow_dst_as	宛先の NetFlow 自律システム番号、起点またはピア
netflow_dst_mask	Netflow 宛先アドレス プレフィックス マスク。
netflow_dst_tos	パケットが宛先から送信元に流れる場合の IP ヘッダーのタイプ オブ サービス(ToS)。
netflow_snmp_in	送信元から宛先へ流れるパケットが使用するインターフェイスの ID。
netflow_snmp_out	宛先から送信元へ流れるパケットが使用するインターフェイスの ID。
netflow_src_as	送信元の NetFlow 自律システム番号、起点またはピア
netflow_src_mask	Netflow 送信元アドレス プレフィックス マスク。
netflow_src_tos	パケットが送信元から宛先に流れる場合の IP ヘッダーのタイプ オブ サービス(ToS)。
network_analysis_policy_name	侵入イベントを生成した侵入ポリシーに関連付けられているネットワーク分析ポリシー
network_analysis_policy_UUID	侵入イベントを生成した侵入ポリシーに関連付けられているネットワーク分析ポリシーの UUID
packets_recv	セッションを開始したホストが受信したパケットの総数。
packets_sent	セッションを開始したホストが送信したパケットの総数。
protocol_name	接続で使用されたプロトコルの名前。
protocol_num	プロトコルの IANA 番号。IANA 番号のリストは http://www.iana.org/assignments/protocol-numbers にあります。
responder_continent_name	セッション イニシエータに対して応答したホストの所在地の地域の名前。 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
responder_country_id	セッション イニシエータに対して応答したホストの国のコード。
responder_country_name	セッション イニシエータに対して応答したホストの国の名前。
responder_ipaddr	セッション イニシエータに対して応答したホストの IPv4 または IP v 6 アドレスのバイナリ表現。
responder_port	セッション レスポンダが使用するポート。

表 7-6　si_connection_log のフィールド（続き）

フィールド	説明
responder_user_dept	セッション イニシエータに対して応答したホストの最終ログイン ユーザの所属部門。
responder_user_email	セッション イニシエータに対して応答したホストの最終ログイン ユーザの電子メールアドレス。
responder_user_first_name	セッション イニシエータに対して応答したホストの最終ログイン ユーザの名前。
responder_user_id	セッション イニシエータに対して応答したホストの最終ログイン ユーザの内部識別番号。
responder_user_last_name	セッション イニシエータに対して応答したホストの最終ログイン ユーザの姓。
responder_user_last_seen_sec	セッション イニシエータに対して応答したホストの最終ログイン ユーザのユーザ アクティビティを Firepower システム が最後に検出した日時を示す UNIX タイムスタンプ。
responder_user_last_updated_sec	セッション イニシエータに対して応答したホストの最終ログイン ユーザのユーザ レコードを Firepower システム が最後に更新した日時を示す UNIX タイムスタンプ。
responder_user_name	セッション イニシエータに対して応答したホストの最終ログイン ユーザのユーザ名。
responder_user_phone	セッション イニシエータに対して応答したホストの最終ログイン ユーザの電話番号。
security_context	トラフィックが通過したセキュリティ コンテキスト（仮想ファイアウォール）の説明。マルチコンテキスト モードの ASA FirePOWER デバイスでは、システムはこのフィールドにのみ入力することに注意してください。
security_group	ネットワーク トラフィック グループの ID 番号。
security_intelligence_category	接続に関連付けられているセキュリティ インテリジェンス カテゴリ。
security_intelligence_ip	セキュリティ インテリジェンスによりモニタされる、接続に関連付けられている IP アドレスが、送信元 IP（src）と宛先 IP（dst）のいずれであるか。
security_zone_egress_name	接続イベントの出力セキュリティ ゾーン。
security_zone_ingress_name	接続イベントの入力セキュリティ ゾーン。
sensor_address	イベントを生成した管理対象デバイスの IP アドレス。形式は ipv4 address,ipv6 address です。
sensor_name	セッションをモニタした管理対象デバイスの名前。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
sinkhole	シンクホール オブジェクトに関連付けられているリビジョン UUID。
src_device_ipaddr	次のいずれかを行います。 • 接続データをエクスポートした NetFlow 対応デバイスの IP アドレスのバイナリ表現 • 0（シスコ 管理対象デバイスにより検出される接続の場合）。

表 7-6　si_connection_log のフィールド(続き)

フィールド	説明
ssl_actual_action	SSL ルールに基づいて接続に対して実行されたアクション。 ルールに指定されているアクションが不可能なことがあるため、これは予期していたアクションとは 異なることがあります。有効な値は次のとおりです。 • 'Unknown' • 'Do Not Decrypt' • 'Block' • 'Block With Reset' • 'Decrypt (Known Key)' • 'Decrypt (Replace Key)' • 'Decrypt (Resign)'
ssl_cipher_suite	SSL 接続で使用される暗号スイート。値は 10 進形式で格納されます。参照先 www.iana.org/assignments/tls-parameters/tls-parameters. xhtml を参照してください。
ssl_expected_action	SSL ルールに基づいて接続に対して実行される必要がある アクション。有効な値は次のとおりです。 • 'Unknown' • 'Do Not Decrypt' • 'Block' • 'Block With Reset' • 'Decrypt (Known Key)' • 'Decrypt (Replace Key)' • 'Decrypt (Resign)'
ssl_flow_flags	暗号化接続のデバッグ レベル フラグ。可能性あり 値は次のとおりです。 • 0x00000001:NSE_FLOW__VALID — 他のフィールドを有効にするために設定する必要があります • 0x00000002:NSE_FLOW__INITIALIZED — 内部構造が処理可能です • 0x00000004:NSE_FLOW__INTERCEPT — SSL セッションが代行受信されました

表 7-6　si_connection_log のフィールド（続き）

フィールド	説明
ssl_flow_messages	SSL ハンドシェイク時にクライアントとサーバ間で交換されたメッセージ。詳細については、http://tools.ietf.org/html/rfc5246 を参照してください。 • 0x00000001：NSE_MT__HELLO_REQUEST • 0x00000002：NSE_MT__CLIENT_ALERT • 0x00000004：NSE_MT__SERVER_ALERT • 0x00000008：NSE_MT__CLIENT_HELLO • 0x00000010：NSE_MT__SERVER_HELLO • 0x00000020：NSE_MT__SERVER_CERTIFICATE • 0x00000040：NSE_MT__SERVER_KEY_EXCHANGE • 0x00000080：NSE_MT__CERTIFICATE_REQUEST • 0x00000100：NSE_MT__SERVER_HELLO_DONE • 0x00000200：NSE_MT__CLIENT_CERTIFICATE • 0x00000400：NSE_MT__CLIENT_KEY_EXCHANGE • 0x00000800：NSE_MT__CERTIFICATE_VERIFY • 0x00001000：NSE_MT__CLIENT_CHANGE_CIPHER_SPEC • 0x00002000：NSE_MT__CLIENT_FINISHED • 0x00004000：NSE_MT__SERVER_CHANGE_CIPHER_SPEC • 0x00008000：NSE_MT__SERVER_FINISHED • 0x00010000：NSE_MT__NEW_SESSION_TICKET • 0x00020000：NSE_MT__HANDSHAKE_OTHER • 0x00040000：NSE_MT__APP_DATA_FROM_CLIENT • 0x00080000：NSE_MT__APP_DATA_FROM_SERVER

表 7-6　si_connection_log のフィールド(続き)

フィールド	説明
ssl_flow_status	SSL フローのステータス。アクションが実行された理由、またはエラーメッセージが出された理由を示す値です。有効な値は次のとおりです。 • 'Unknown' • 'No Match' • 'Success' • 'Uncached Session' • 'Unknown Cipher Suite' • 'Unsupported Cipher Suite' • 'Unsupported SSL Version' • 'SSL Compression Used' • 'Session Undecryptable in Passive Mode' • 'Handshake Error' • 'Decryption Error' • 'Pending Server Name Category Lookup' • 'Pending Common Name Category Lookup' • 'Internal Error' • 'Network Parameters Unavailable' • 'Invalid Server Certificate Handle' • 'Server Certificate Fingerprint Unavailable' • 'Cannot Cache Subject DN' • 'Cannot Cache Issuer DN' • 'Unknown SSL Version' • 'External Certificate List Unavailable' • 'External Certificate Fingerprint Unavailable' • 'Internal Certificate List Invalid' • 'Internal Certificate List Unavailable' • 'Internal Certificate Unavailable' • 'Internal Certificate Fingerprint Unavailable' • 'Server Certificate Validation Unavailable' • 'Server Certificate Validation Failure' • 'Invalid Action'
ssl_issuer_common_name	SSL 証明書の発行元の共通名。これは一般に証明書発行元のホストとドメイン名ですが、その他の情報が含まれていることもあります。
ssl_issuer_country	SSL 証明書の発行元の国。
ssl_issuer_organization	SSL 証明書の発行元の組織。
ssl_issuer_organization_unit	SSL 証明書の発行元の組織単位。
ssl_policy_action	ルールが一致しない場合のためにポリシーで設定されているデフォルト アクション。

表 7-6　si_connection_log のフィールド(続き)

フィールド	説明
ssl_policy_name	接続を処理した SSL ポリシーの ID 番号。
ssl_policy_reason	SSL ポリシーが SSL セッションをログに記録した理由。
ssl_rule_action	SSL ルールに対しユーザ インターフェイスで選択されたアクション(allow、block など)。
ssl_rule_name	接続を処理した SSL ルールまたはデフォルト アクションの ID 番号。
ssl_serial_number	発行元 CA によって割り当てられた SSL 証明書のシリアル番号。
ssl_server_name	SSL Client Hello でサーバ名に指定された名前。
ssl_subject_common_name	SSL 証明書の件名共通名。これは一般に証明書の件名のホストとドメイン名ですが、その他の情報が含まれていることもあります。
ssl_subject_country	SSL 証明書の件名の国。
ssl_subject_organization	SSL 証明書の件名の組織。
ssl_subject_organization_unit	SSL 証明書の件名の組織単位。
ssl_url_category	サーバ名と証明書の共通名から識別されるフローのカテゴリ。
ssl_version	接続の暗号化に使用された SSL または TLS プロトコル バージョン。
tcp_flags	セッションで検出された TCP フラグ。
url	セッション中にモニタ対象ホストによって要求された URL(使用可能な場合)。
url_category	モニタ対象ホストによって要求された URL のカテゴリ。
url_reputation	モニタ対象ホストによって要求された URL のレピュテーション。次のいずれかが必要です。 • 1:高リスク • 2:疑わしいサイト • 3:セキュリティ リスクのある無害なサイト • 4:無害なサイト • 5:既知
web_application_id	Web アプリケーションの内部識別番号。
web_application_name	次のいずれかになります。 • アプリケーションの名前(確実な識別が可能な場合)。 • web browsing(システムがアプリケーション プロトコル HTTP を検出したが、特定の Web アプリケーションを検出できない場合)。 • 空白(接続に HTTP トラフィックがない場合)。

si_connection_log の結合

次の表に、**si_connection_log** テーブルを使用して実行できる結合について説明します。

表 7-7　si_connection_log の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_protocol_name または application_id または client_application_id または web_application_id	application_info.application_id application_host_map.application_id application_tag_map.application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id
initiator_ipaddr または responder_ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

si_connection_log のサンプル クエリ

次のクエリは、**si_connection_log** テーブルから最大 25 件の接続イベント レコードを、パケットのタイムスタンプに基づいて降順にソートして返します。

```
SELECT first_packet_sec, last_packet_sec, initiator_ipaddr, responder_ipaddr,
security_zone_ingress_name, security_zone_egress_name, initiator_port, protocol_name,
responder_port, application_protocol_id, client_application_id, web_application_id, url,
url_category, url_reputation

FROM si_connection_log

WHERE first_packet_sec <= UNIX_TIMESTAMP("2011-10-01 00:00:00") ORDER BY
first_packet_sec

DESC, last_packet_sec DESC LIMIT 0, 25;
```

スキーマ:ユーザ アクティビティ テーブル

この章では、ユーザ アクティビティおよびアイデンティティ イベントのスキーマとサポートされている結合について説明します。Firepower システム は、さまざまなユーザ ログイン(LDAP、POP3、IMAP、SMTP、AIM、SIP など)を追跡することでネットワークでのユーザ アクティビティを検出できます。

詳細については、次の表に示す項を参照してください。

表 8-1　ユーザ アイデンティティ テーブルのスキーマ

参照先	次の内容が格納されるテーブル	バージョン
discovered_users(8-1 ページ)	システムにより検出されたユーザに関する情報。	5.0+
user_discovery_event(8-2 ページ)	ネットワーク上のユーザ アクティビティの詳細を示すユーザ検出イベント。	5.0+

discovered_users

discovered_users テーブルには、システムにより検出された各ユーザの詳細情報が格納されます。

discovered_users テーブルは、Firepower システム バージョン 5.0 以降で廃止されたテーブル **rua_users** を置き換えます。

詳細については、次の項を参照してください。

- discovered_users のフィールド(8-1 ページ)
- discovered_users の結合(8-2 ページ)
- discovered_users のサンプル クエリ(8-2 ページ)

discovered_users のフィールド

次の表に、**discovered_users** テーブルでアクセスできるフィールドについて説明します。

表 8-2　discovered_users のフィールド

フィールド	説明
dept	ユーザの所属部門。
email	ユーザの電子メール アドレス。

表 8-2　discovered_users のフィールド（続き）

フィールド	説明
first_name	ユーザの名前。
ip_address	このフィールドは廃止されており、すべてのクエリに対して null が返されます。
ipaddr	ユーザ ログインが検出されたホストの IPv4 または IPv6 アドレスのバイナリ表現。
last_name	ユーザの姓。
last_seen_sec	システムがユーザのログインを最後に報告した日時を示す UNIX タイムスタンプ。
last_updated_sec	ユーザの情報の最終更新日時を示す UNIX タイムスタンプ。
name	ユーザの名前。
phone	ユーザの電話番号。
rna_service	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
user_id	ホストの最終ログイン ユーザの内部識別番号。

discovered_users の結合

次の表に、**rua_user** テーブルで実行できる結合について説明します。

表 8-3　discovered_users の結合

左結合できるフィールド	結合できる他のテーブルの結合タイプ
user_id	user_discovery_event.user_id user_ipaddr_history.user_id

discovered_users のサンプル クエリ

次のクエリは、指定された日時以降に生成された検出ユーザ レコードを最大 25 件まで返します。

```
SELECT user_id, ip_address, email, name, last_seen_sec, last_updated_sec
FROM discovered_users
WHERE last_seen_sec >= UNIX_TIMESTAMP("2011-10-01 00:00:00")
LIMIT 0, 25;
```

user_discovery_event

user_discovery_event テーブルには、各ユーザ検出イベントのレコードが格納されます。

バージョン 5.0 以降、Firepower システム は検出エンジンではなく、管理対象デバイス レベルでのユーザ アクティビティの検出を記録することに注意してください。このテーブルの detection_engine_name フィールドと detection_engine_uuid フィールドはそれぞれ、sensor_name フィールドと sensor_uuid フィールドに置き換えられました。これらのフィールドに対するクエリは、ユーザ検出イベントを生成した管理対象デバイスに関する情報を返します。

詳細については、次の項を参照してください。

- user_discovery_event のフィールド（8-3 ページ）
- user_discovery_event の結合（8-4 ページ）
- user_discovery_event のサンプル クエリ（8-4 ページ）

user_discovery_event のフィールド

次の表に、`user_discovery_event` テーブルでアクセスできるフィールドについて説明します。

表 8-4　　user_discovery_event のフィールド

フィールド	説明
application_protocol_id	検出されたアプリケーション プロトコルの内部 ID。
application_protocol_name	次のいずれかになります。 • 接続で使用されたアプリケーションの名前（LDAP、POP3 など） • pending（システムで、何らかの理由でアプリケーションを識別できない場合） • 空白（接続にアプリケーション情報がない場合）
説明	検出イベントのタイプが [Delete User Identity] または [User Identity Dropped] の場合は、ユーザ名。それ以外の場合は空白。
domain_name	ユーザが検出されたのドメインの名前。
domain_uuid	ユーザが検出されたドメインの UUID。これはバイナリで示されます。
endpoint_profile	接続エンドポイントで使用されるデバイスのタイプの名前。
event_id	検出イベントの内部識別番号。
event_time_sec	検出イベントの日時を示す UNIX タイムスタンプ。
event_type	検出イベントのタイプ。New User Identity や User Login など。
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
ipaddr	ユーザ アクティビティが検出されたホストの IP アドレスのバイナリ表現。
location_ip	ISE と通信するインターフェイスの IP アドレス。IPv4 または IPv6 のアドレスを使用できます。
reported_by	ユーザ ログインを報告する Active Directory サーバの IPv4 アドレス、IPv6 アドレス、または NetBIOS 名。
security_group	ネットワーク トラフィック グループの ID 番号。
sensor_address	ユーザ検出イベントが検出された管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_name	ユーザ検出イベントが検出された管理対象デバイスのテキスト名。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
user_dept	ホストの最終ログイン ユーザが所属する部門。
user_email	ホストの最終ログイン ユーザの電子メール アドレス。
user_first_name	ユーザの名。

表 8-4　user_discovery_event のフィールド（続き）

フィールド	説明
user_id	ホストの最終ログイン ユーザの内部識別番号。
user_last_name	ユーザの姓。
user_last_seen_sec	システムがユーザのログインを最後に報告した日時を示す UNIX タイムスタンプ。
user_last_updated_sec	ユーザの情報の最終更新日時を示す UNIX タイムスタンプ。
user_name	ホストの最終ログイン ユーザのユーザ名。
user_phone	ホストの最終ログイン ユーザの電話番号。

user_discovery_event の結合

次の表に、**user_discovery_event** テーブルで実行できる結合について説明します。

表 8-5　user_discovery_event の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr
user_id	discovered_users.user_id user_ipaddr_history.user_id

user_discovery_event のサンプル クエリ

次のクエリは、特定の日時以降に、選択された管理対象デバイスにより生成されたユーザ イベント レコードを最大 25 件まで返します。

```
SELECT event_time_sec, ipaddr, sensor_name, event_type, user_name, user_last_seen_sec,
user_last_updated_sec

FROM user_discovery_event

WHERE sensor_name = sensor_name

AND user_last_seen_sec >= UNIX_TIMESTAMP("2011-10-01 00:00:00") ORDER BY event_type ASC

LIMIT 0, 25;
```

スキーマ：相関テーブル

この章では、相関関連イベント（修復ステータスやホワイト リスト イベントなど）のスキーマとサポートされている結合について説明します。詳細については、次の表に示す項を参照してください。

表 9-1　　相関テーブルのスキーマ

参照先	次の内容が格納されるテーブル	バージョン
compliance_event（9-1 ページ）	相関イベント。このイベントは、アクティブな相関ポリシー内の相関ルールがトリガーされると生成されます。	4.10.x+
remediation_status（9-6 ページ）	修復ステータス イベント。このイベントは、アクティブな相関ポリシーによって応答として修復がトリガーされると生成されます。	4.10.x+
white_list_event（9-8 ページ）	ホワイト リスト イベント。このイベントは、アクティブなホワイト リスト コンプライアンス ポリシーでホワイト リストのコンプライアンスに準拠していないホストが検出されると、生成されます。	4.10.x+
white_list_violation（9-10 ページ）	ホワイト リスト違反。この違反は、ネットワーク上のホストが、アクティブなコンプライアンス ポリシーのコンプライアンス ホワイト リストにどのように違反しているかを追跡します。	4.10.x+

compliance_event

`compliance_event` テーブルには、Firepower Management Center により生成される相関イベントに関する情報が格納されます。

詳細については、次の項を参照してください。

- compliance_event のフィールド（9-2 ページ）
- compliance_event の結合（9-6 ページ）
- compliance_event のサンプル クエリ（9-6 ページ）

compliance_event のフィールド

このテーブルのフィールドの多くは、相関ルールをトリガーしたイベントのタイプに応じて空白になることがある点に注意してください。たとえば、システムで特定のアプリケーション プロトコルまたは特定のポートで稼働している Web アプリケーションが検出されたために、Firepower Management Center により相関イベントが生成される場合、その相関イベントには、侵入関連の情報は含まれません。また、このテーブルのフィールドは、Firepower システム の設定に基づいて空白になることもあります。たとえば Control ライセンスを所有していない場合、相関イベントにはユーザ アイデンティティ情報が含まれません。

バージョン 5.0 以降、Firepower システム は検出エンジンではなく、管理対象デバイス レベルでのネットワークおよびユーザ アクティビティの検出を記録することに注意してください。現在、**compliance_event** テーブルの detection_engine_name フィールドと detection_engine_uuid フィールドは空白だけを返し、これらのフィールドを結合するクエリはレコードを返しません。イベントが検出された場所に関する情報については、detection_engine_uuid フィールドではなく sensor_uuid フィールドを照会する必要があります。

次の表に、**compliance_event** テーブルでアクセスできるデータベース フィールドについて説明します。

表 9-2　compliance_event のフィールド

フィールド	説明
blocked	侵入イベントをトリガーしたパケットの処理を示す値。 • 0:パケットはドロップされなかった • 1:パケットはドロップされた（インライン型、スイッチ型、またはルーティング型展開） • 2:侵入ポリシーが、インライン型、スイッチ型、またはルーティング型展開のデバイスに適用されている場合は、イベントをトリガーしたパケットがドロップされている可能性がある。
description	相関イベントと、このイベントがどのように引き起こされたかに関する情報。
detection_engine_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
detection_engine_uuid	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
domain_name	イベントが検出されたドメインの名前。
domain_uuid	イベントが検出されたドメインの UUID。これはバイナリで示されます。
dst_host_criticality	相関イベントに関連する宛先ホストにユーザが割り当てたホスト重要度: None、Low、Medium、または High
dst_host_type	宛先ホストのタイプ:Host、Router、Bridge、NAT Device、または Load Balancer
dst_ip_address	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
dst_ip_address_v6	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
dst_ipaddr	トリガー イベントに関連する宛先ホストの IPv4 または IP v 6 アドレスのバイナリ表現。
dst_os_product	宛先ホストのオペレーティング システムの名前。

表 9-2　compliance_event のフィールド(続き)

フィールド	説明
dst_os_vendor	宛先ホストのオペレーティング システムのベンダー。
dst_os_version	宛先ホストのオペレーティング システムのバージョン番号。
dst_port	イベント プロトコル タイプが TCP または UDP の場合にトラフィックを受信するホストのポート番号。プロトコル タイプが ICMP の場合は ICMP コード。
dst_rna_service	トリガー イベントに関連付けられている送信元ホストのアプリケーション プロトコル(判明している場合)。判明していない場合は、次のいずれかになります。 • none または空白:アプリケーション プロトコル トラフィックがありません。 • unknown:既知のサーバ フィンガープリントに基づいてサーバを識別できませんでした。 • pending:システムにさらに情報が必要です。
dst_user_dept	宛先ユーザの所属部門。
dst_user_email	宛先ユーザの電子メール アドレス。
dst_user_first_name	宛先ユーザの名前。
dst_user_id	宛先ユーザの内部識別番号。宛先ユーザとは、イベント発生前に宛先ホストにログインした最終ユーザです。
dst_user_last_name	宛先ユーザの姓。
dst_user_last_seen_sec	システムが宛先ユーザのログインを最後に報告した日時を示す UNIX タイムスタンプ。
dst_user_last_updated_sec	宛先ユーザの情報の最終更新日時を示す UNIX タイムスタンプ。
dst_user_name	宛先ユーザのユーザ名。
dst_user_phone	宛先ユーザの電話番号。
dst_vlan_id	宛先ホストの VLAN ID 番号(該当する場合)。
event_id	デバイスによって生成されたトリガー侵入イベントの識別番号。
event_time_sec	トリガー イベントの日時を示す UNIX タイムスタンプ。
event_time_usec	トリガー イベントのタイムスタンプのマイクロ秒単位の増分。
event_type	相関ルールをトリガーした基礎となるイベントのタイプ、または Firepower Management Center が相関イベントを生成する原因となった基礎となるイベントのタイプ値は次のとおりです。 • ids:侵入イベント トリガー • rna:検出イベント、ホスト入力イベント、接続イベント、またはトラフィック プロファイル変更トリガー • rua:ユーザ検出イベント トリガー • whitelist:コンプライアンス ホワイト リスト違反トリガー
host_event_type	イベント タイプ(New Host、Identity Conflict など)。

表 9-2　compliance_event のフィールド(続き)

フィールド	説明
id	相関イベントの内部識別番号。
impact	イベントの影響フラグ値。値は次のとおりです。 • 1:レッド(脆弱) • 2:オレンジ(脆弱の可能性あり) • 3:イエロー(現在は脆弱でない) • 4:ブルー(不明なターゲット) • 5:グレー(不明な影響) 相関ルールが侵入イベントによってトリガーされた場合にのみ設定します。
interface_egress_name	接続に関連付けられた入力インターフェイス。
interface_ingress_name	接続に関連付けられた出力インターフェイス。
policy_name	違反が発生した相関ポリシー。
policy_rule_name	ポリシー違反をトリガーした相関ルール。
policy_rule_uuid	相関ルールの固有識別子。
policy_time_sec	相関イベントが生成された日時を示す UNIX タイムスタンプ。
policy_uuid	相関ポリシーの固有識別子。
priority	相関イベントのプライオリティ。ユーザ インターフェイスで設定されます。このイベント プライオリティは、トリガーされたルールのプライオリティまたは違反が発生した相関ポリシーのプライオリティによって決まります。
protocol_name	イベントに関連付けられているプロトコル(使用可能な場合)。
protocol_num	IANA 指定のプロトコル番号(使用可能な場合)。
rna_event_type	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
rua_event_type	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
rule_generator_id	トリガー侵入イベントを生成したコンポーネントのジェネレータ ID 番号(GID)。
rule_message	相関ルールをトリガーした侵入イベントを説明するテキスト。ルールベースのイベントの場合、イベント メッセージはルールから生成されます。デコーダベースおよびプリプロセッサベースのイベントの場合、メッセージはハード コーディングされています。
rule_signature_id	イベントのシグニチャ ID(SID)。トリガー侵入イベントが生成される原因となった特定のルール、デコーダ メッセージ、またはプリプロセッサ メッセージを識別します。
security_zone_egress_name	相関イベントの出力セキュリティ ゾーン。
security_zone_ingress_name	相関イベントの入力セキュリティ ゾーン。
sensor_address	コンプライアンス イベントをトリガーした基礎となるイベントを生成した管理対象デバイスの IP アドレス。形式は *ipv4_address,ipv6_address* です。

表 9-2　compliance_event のフィールド(続き)

フィールド	説明
sensor_name	コンプライアンス イベントをトリガーした基礎となるイベントを生成した管理対象デバイス。
sensor_uuid	管理対象デバイスの固有識別子(sensor_name が null の場合は 0)。
src_host_criticality	コンプライアンス イベントに関連する送信元ホストにユーザが割り当てたホスト重要度:None、Low、Medium、または High。
src_host_type	送信元ホストのタイプ:Host、Router、Bridge、NAT Device、または Load Balancer。
src_ip_address	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
src_ip_address_v6	バージョン 5.2 で廃止されたフィールド。後方互換性を維持するため、このフィールドの値は null には設定されませんが、信頼できません。
src_ipaddr	トリガー イベントに関連する送信元ホストの IPv4 または IPv6 アドレスのバイナリ表現。
src_os_product	送信元ホストのオペレーティング システムの名前。
src_os_vendor	送信元ホストのオペレーティング システムのベンダー。
src_os_version	送信元ホストのオペレーティング システムのバージョン番号。
src_port	送信元ホストのポート番号。ICMP トラフィックの場合は ICMP タイプが表示されます。
src_rna_service	トリガー イベントに関連付けられている送信元ホストのアプリケーション プロトコル(判明している場合)。判明していない場合は、次のいずれかになります。 • none または空白:アプリケーション プロトコル トラフィックがありません。 • unknown:既知のサーバ フィンガープリントに基づいてサーバとアプリケーション プロトコルを識別できませんでした。 • pending:システムにさらに情報が必要です。
src_user_dept	送信元ユーザの所属部門。
src_user_email	送信元ユーザの電子メール アドレス。
src_user_first_name	送信元ユーザの名前。
src_user_id	送信元ユーザの内部識別番号。これは、イベント発生前に送信元ホストにログインしていた最終ユーザです。
src_user_last_name	送信元ユーザの姓。
src_user_last_seen_sec	システムが送信元ユーザのログインを最後に報告した日時を示す UNIX タイムスタンプ。
src_user_last_updated_sec	送信元ユーザの情報の最終更新日時を示す UNIX タイムスタンプ。
src_user_name	送信元ユーザのログイン ユーザ名。
src_user_phone	送信元ユーザの電話番号。

表 9-2　compliance_event のフィールド（続き）

フィールド	説明
src_vlan_id	送信元ホストの VLAN 識別番号（該当する場合）。
user_event_type	トリガー ユーザ イベントのタイプ（New User Identity または User Login など）。

compliance_event の結合

次の表に、**compliance_event** テーブルで実行できる結合について説明します。

表 9-3　compliance_event の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
dst_ipaddr または src_ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

compliance_event のサンプル クエリ

次のクエリは、1 週間の相関イベント レコードを最大 25 件返します。これらのレコードには、イベント時刻、送信元と宛先の IP アドレス、送信元と宛先のポート、ポリシー情報などのイベント情報が含まれています。

```
SELECT event_id, policy_time_sec, impact, blocked, src_ipaddr, dst_ipaddr, src_port,
dst_port, description, policy_name, policy_rule_name, priority, src_host_criticality,
dst_host_criticality, security_zone_egress_name, security_zone_ingress_name,
sensor_name, interface_egress_name, interface_ingress_name

FROM compliance_event WHERE event_type!="whitelist"

AND policy_time_sec

BETWEEN UNIX_TIMESTAMP("2011-10-01 00:00:00")

AND UNIX_TIMESTAMP("2011-10-07 23:59:59")

domain_name= "Global \ Company B \ Edge"

ORDER BY policy_time_sec

DESC LIMIT 0, 25;
```

remediation_status

remediation_status テーブルには、修復イベントに関する情報が格納されます。修復イベントは、Firepower Management Center が相関ポリシー違反に対応して修復を開始すると生成されます。

詳細については、次の項を参照してください。

- remediation_status のフィールド (9-7 ページ)

remediation_status のフィールド

次の表に、**remediation_status** テーブルでアクセスできるデータベース フィールドについて説明します。

表 9-4　　remediation_status のフィールド

フィールド	説明
id	違反が発生し、修復をトリガーした相関ポリシーの識別番号。
policy_name	違反が発生し、修復をトリガーした相関ポリシー。
policy_rule_name	修復をトリガーした特定の相関ルール。
policy_rule_uuid	相関ルールの固有識別子。
policy_time_sec	修復をトリガーした相関イベントの生成日時を示す UNIX タイムスタンプ。
policy_uuid	相関イベントをトリガーした相関ポリシーの固有識別子。
remediation_name	開始された修復。
remediation_time_sec	Firepower Management Center により修復が開始された日時を示す UNIX タイムスタンプ。
status_text	修復の開始時に発生した事象を説明するメッセージ（「successful completion of remediation」など）。

remediation_status の結合

remediation_status テーブルに対して結合を実行することはできません。

remediation_status のサンプル クエリ

次のクエリは、特定の日付より前に生成された最大 25 件のレコードを返します。これらのレコードには修復のステータスに関する情報（修復のタイムスタンプ、ステータス メッセージなど）が含まれます。

```
SELECT policy_time_sec, remediation_time_sec, remediation_name, policy_name,
policy_rule_name, status_text
FROM remediation_status WHERE remediation_time_sec <= UNIX_TIMESTAMP("2011-10-01
00:00:00")
ORDER BY policy_time_sec
DESC LIMIT 0, 25;
```

white_list_event

white_list_event テーブルにはホワイト リスト イベントが格納されます。ホワイト リスト イベントは、ホストがアクティブなホワイト リスト コンプライアンス ポリシーのホワイト リストに準拠していないことがシステムにより検出されると生成されます。

バージョン 5.0 以降、Firepower システム は検出エンジンではなく、管理対象デバイス レベルでのネットワークおよびユーザ アクティビティの検出を記録することに注意してください。**white_list_event** テーブルの detection_engine_name フィールドと detection_engine_uuid フィールドは null だけを返し、またこれらのフィールドを結合するクエリはレコードを返しません。detection_engine_uuid フィールドの代わりに sensor_uuid フィールドを照会すると、同等の情報が返されます。

詳細については、次の項を参照してください。

- white_list_event のフィールド (9-8 ページ)
- white_list_event の結合 (9-9 ページ)
- white_list_event のサンプル クエリ (9-10 ページ)

white_list_event のフィールド

次の表に、**white_list_event** テーブルでアクセスできるデータベース フィールドについて説明します。

表 9-5　white_list_event のフィールド

フィールド	説明
description	ホワイト リスト違反の説明。
detection_engine_name	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
detection_engine_uuid	バージョン 5.0 で廃止されたフィールド。すべてのクエリに対して null を返します。
host_criticality	ホワイト リストに準拠していないホストに対してユーザが割り当てた重要度 ([None]、[Low]、[Medium]、または [High])。
host_type	ホストのタイプ：Host、Router、Bridge、NAT Device、または Load Balancer。
id	ホワイト リスト イベントの内部固有識別子。
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
ip_address_v6	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
ipaddr	準拠していないホストの IP アドレスのバイナリ表現。
os_product	オペレーティング システムの製品名。
os_vendor	オペレーティング システムのベンダー。
os_version	オペレーティング システムのバージョン番号。
policy_name	ホワイト リストを含む違反コンプライアンス ポリシー。
policy_time_sec	イベントが生成された日時を示す UNIX タイムスタンプ。
policy_uuid	ホワイトリスト イベントを含むコンプライアンス ポリシーの固有識別子。

表 9-5　white_list_event のフィールド（続き）

フィールド	説明
port	サービス ホワイト リスト違反をトリガーしたイベント（非準拠サービスの結果として違反が発生した場合）に関連付けられているポート（存在する場合）。他のタイプのホワイト リスト違反の場合、このフィールドは空白です。
priority	ホワイト リスト イベントのプライオリティ。ユーザ インターフェイスで設定されます。
protocol_name	イベントに関連付けられているプロトコル（使用可能な場合）。
protocol_num	IANA 指定のプロトコル番号（使用可能な場合）。
rna_service	ホワイト リスト違反をトリガーしたサービス（使用可能な場合）。
sensor_address	トラフィックを検出した管理対象デバイスの IP アドレス。形式は *ipv4_address*,*ipv6_address* です。
sensor_name	ホワイト リスト イベントを生成したデバイス。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
user_dept	ユーザの所属部門。
user_email	ユーザの電子メール アドレス。
user_first_name	ユーザの名前。
user_id	イベント発生前にホストにログインしていた最終ユーザの内部 ID 番号。
user_last_name	ユーザの姓。
user_last_seen_sec	システムがユーザのログインを最後に報告した日時を示す UNIX タイムスタンプ。
user_last_updated_sec	ユーザの情報の最終更新日時を示す UNIX タイムスタンプ。
user_name	ユーザのログイン ユーザ名。
user_phone	ユーザの電話番号。
vlan_id	VLAN 識別番号（該当する場合）。
white_list_name	違反が発生したホワイト リスト。
white_list_uuid	ホワイト リストの固有識別子。

white_list_event の結合

次の表に、**white_list_event** テーブルで実行できる結合について説明します。

表 9-6　white_list_event の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
ipaddr	rna_host_ip_map.ipaddr user_ipaddr_history.ipaddr

white_list_event のサンプル クエリ

次のクエリは、指定された時点よりも前に生成されたレコードを最大 25 件返します。これらのレコードには、ホワイト リスト イベントに関する情報（コンプライアンス ポリシー名、イベント生成時点のタイムスタンプ、ホワイト リストの名前など）が含まれます。

```
SELECT policy_name, policy_time_sec, ipaddr, user_name, port, description,
white_list_name, priority, host_criticality, sensor_name

FROM white_list_event WHERE policy_time_sec <= UNIX_TIMESTAMP("2011-10-01 00:00:00")

ORDER BY policy_time_sec DESC LIMIT 0, 25;
```

white_list_violation

white_list_violation テーブルは、コンプライアンス ホワイト リスト違反を追跡します。この違反は、ネットワーク上のホストが、アクティブなコンプライアンス ポリシーのコンプライアンス ホワイト リストにどのように違反しているかを追跡します。

詳細については、次の項を参照してください。

- white_list_violation のフィールド（9-10 ページ）
- white_list_violation の結合（9-11 ページ）
- white_list_violation のサンプル クエリ（9-11 ページ）

white_list_violation のフィールド

次の表に、**white_list_violation** テーブルでアクセスできるデータベース フィールドについて説明します。

表 9-7　white_list_violation のフィールド

フィールド	説明
host_id	ホワイト リストに違反するホストの ID 番号。
info	ホワイト リスト違反に関連付けられたすべての利用可能なベンダー、製品、またはバージョン情報。 ホワイト リストに違反するプロトコルの場合、このフィールドには、違反の原因がネットワーク プロトコルとトランスポート プロトコルのどちらなのかも示されます。
ip_address	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
port	サービス ホワイト リスト違反をトリガーしたイベント（非準拠サービスの結果として違反が発生した場合）に関連付けられているポート（存在する場合）。他のタイプのホワイト リスト違反の場合、このフィールドは空白です。
protocol_name	イベントに関連付けられているプロトコル。

表 9-7　white_list_violation のフィールド（続き）

フィールド	説明
type	ホワイト リスト違反のタイプ。非準拠が原因で違反が発生したかどうかを示します。 • オペレーティング システム（os） • サービス（service） • クライアント アプリケーション（client app） • プロトコル（protocol）
violation_time_sec	違反がログに記録された日時を示す UNIX タイムスタンプ。
white_list_name	違反が発生したホワイト リスト。
white_list_uuid	ホワイト リストの固有識別子。

white_list_violation の結合

white_list_violation テーブルに対して結合を実行することはできません。

white_list_violation のサンプル クエリ

次のクエリは、ホワイト リスト違反に関する情報（ホワイト リストに違反するホストの IP アドレス、違反が発生したホワイト リストの名前、違反の数など）を含むレコードを最大 25 件返します。

```
SELECT host_id, white_list_name, count(*)

FROM white_list_violation

GROUP BY white_list_name, host_id

ORDER BY white_list_name

DESC LIMIT 0, 25;
```

スキーマ：ファイル イベント テーブル

この章では、ファイル イベントのスキーマとサポートされている結合について説明します。詳細については、次の表に示す項を参照してください。

表 10-1　　ファイル イベント テーブルのスキーマ

参照先	次の内容が格納されるテーブル	バージョン
file_event（10-1 ページ）	モニタ対象ネットワーク内でファイル転送が検出されると生成されるファイル イベント。	5.1.1+

次に示すテーブルは使用可能ですが、シスコ では現在、これらのテーブルでの検索がサポートされていません。

- `file_categories`
- `file_rules`
- `file_types`
- `file_type_rule_map`
- `file_type_category_map`

file_event

`file_event` テーブルには、Firepower Management Center により生成されるファイル イベントに関する情報が格納されます。モニタ対象ネットワークでファイル転送が検出されるたびに、新しいファイル イベントが生成されます。AMP for Firepower によってマルウェアとして識別されたファイルは、ファイル イベントとマルウェア イベントの両方を生成します。エンドポイント ベースのマルウェア イベントには、対応するファイル イベントはありません。また、ファイル イベントには AMP for Endpoints 関連のフィールドはありません。

詳細については、次の項を参照してください。

- file_event のフィールド（10-2 ページ）
- file_event の結合（10-6 ページ）
- file_event のサンプル クエリ（10-6 ページ）

file_event のフィールド

file_event テーブルには、モニタ対象ネットワークを通過する際に検出されたファイルに関する情報が格納されます。各ファイル イベントを接続イベントに相関付けることができます。ファイルとファイル転送の詳細（ファイルの名前、サイズ、送信元、宛先、および方向、ファイルの SHA256 ハッシュ、ファイルを検出したデバイス、ファイルがマルウェアとして見なされるかどうか、など）が記録されます。

表 10-2　file_event のフィールド

フィールド	説明
action	ファイル タイプに基づいてファイルに対して実行されたアクション。次のいずれかの値になります。 • 1:検出 • 2:ブロック • 3:マルウェア クラウド検索 • 4:マルウェア ブロック • 5:マルウェア ホワイトリスト • 6:クラウド ルックアップ タイムアウト
application_id	ファイル転送を使用するアプリケーションにマップされている ID 番号。
application_name	次のいずれか。 • 接続で使用されたアプリケーションの名前。 • pending または unknown（システムがアプリケーションを識別できない場合）。 • 空白（接続にアプリケーション情報がない場合）
archived	ファイルがアーカイブされているかどうかを示します。
cert_valid_end_date	接続で使用された SSL 証明書が有効ではなくなった時点を示す UNIX タイムスタンプ。
cert_valid_start_date	接続で使用された SSL 証明書の発行時点を示す UNIX タイムスタンプ。
client_application_id	クライアント アプリケーションの内部識別番号（該当する場合）。
client_application_name	クライアント アプリケーションの名前（該当する場合）。
connection_sec	ファイル イベントに関連付けられている接続イベントの UNIX タイムスタンプ（00:00:00 01/01/1970 からの経過秒数）。
counter	同じ秒数で発生した複数のイベントを区別するために使用されるイベント固有のカウンタ。
direction	ファイルのアップロードとダウンロードのいずれが行われたか。現時点では、この値はプロトコルに完全に依存しています（例えば接続が HTTP の場合はダウンロード）。

表 10-2　file_event のフィールド(続き)

フィールド	説明
disposition	ファイルのマルウェア ステータス。有効な値は次のとおりです。 • CLEAN:ファイルはクリーンであり、マルウェアが含まれていない。 • UNKNOWN:ファイルにマルウェアが含まれているかどうかが不明である。 • MALWARE:ファイルにマルウェアが含まれている。 • UNAVAILABLE:ソフトウェアから シスコ クラウドに対して、特性を確認する要求を送信できなかったか、または シスコ クラウド サービスが要求に応答しなかった。 • CUSTOM SIGNATURE:ファイルがユーザ定義のハッシュと一致するため、ユーザが指定した方法で処理された。
domain_name	イベントが検出されたドメインの名前。
domain_uuid	イベントが検出されたドメインの UUID。これはバイナリで示されます。
dst_continent_name	宛先ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
dst_country_id	宛先ホストの国のコード。
dst_country_name	宛先ホストの国の名前。
dst_ip_address_v6	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
dst_ipaddr	トリガー イベントに関連する宛先ホストの IP アドレスのバイナリ表現。
dst_port	接続の宛先のポート番号。
event_description	イベント タイプに関連付けられている追加イベント情報。
event_id	イベント識別番号。
file_name	検出されたファイルの名前。この名前には、UTF-8 文字を使用できます。
file_sha	ファイルの SHA256 ハッシュ
file_size	検出されたファイルのサイズ(バイト単位)。
file_type	検出または検疫されたファイルのファイル タイプ。
file_type_category	ファイル カテゴリの説明
file_type_category_id	ファイル カテゴリの数値 ID。
file_type_id	ファイル タイプにマップされている ID 番号。
http_response_code	イベントで HTTP 要求に対して返された応答コード。
instance_id	イベントを生成した管理対象デバイスの Snort インスタンスの数値 ID。

表 10-2　file_event のフィールド（続き）

フィールド	説明
netmap_num	イベントが検出されたドメインの Netmap ID。
policy_uuid	イベントをトリガーしたアクセス コントロール ポリシーの固有識別子として機能する識別番号。
sandboxed	ファイルが動的分析のために送信されているかどうかを示します。値は次のとおりです。 • Sent for Analysis • Failed to Send • File Size is Too Small • File Size is Too Large • Sent for Analysis • Analysis Complete • Failure (Network Issue) • Failure (Rate Limit) • Failure (File Too Large) • Failure (File Read Error) • Failure (Internal Library Error) • File Not Sent, Disposition Unavailable • Failure (Cannot Run File) • Failure (Analysis Timeout) • File Not Supported
得点	動的分析中に観測された、悪意のある可能性がある振る舞いに基づく数値（0 ～ 100）。
security_context	トラフィックが通過したセキュリティ コンテキスト（仮想ファイアウォール）の説明。システムは、マルチコンテキスト モードの ASA FirePOWER デバイスの場合のみ、このフィールドに値を入力することに注意してください。
sensor_address	イベントを提供したデバイスの IP アドレスのバイナリ表現。
sensor_id	イベントを提供したデバイスの ID。
sensor_name	イベント レコードを生成した管理対象デバイスのテキスト名。接続デバイスではなくレポート デバイス自体を参照するイベントの場合、このフィールドは null です。
sensor_uuid	管理対象デバイスの固有識別子（sensor_name が null の場合は 0）。
signature_processed	ファイルの署名が処理されたかどうかを示します。

表 10-2　file_event のフィールド（続き）

フィールド	説明
src_continent_name	送信元ホストが位置する地域の名前 **:不明 na:北米 as:アジア af:アフリカ eu:欧州 sa:南米 au:オーストラリア an:南極
src_country_id	送信元ホストの国のコード。
src_country_name	送信元ホストの国の名前。
src_ip_address_v6	バージョン 5.2 で廃止されたフィールド。すべてのクエリに対して null を返します。
src_ipaddr	トリガー イベントに関連する送信元ホストの IPv4 または IPv6 アドレスのバイナリ表現。
src_port	接続元のポート番号。
ssl_issuer_common_name	SSL 証明書の発行元の共通名。これは一般に証明書発行元のホストとドメイン名ですが、その他の情報が含まれていることもあります。
ssl_issuer_country	SSL 証明書の発行元の国。
ssl_issuer_organization	SSL 証明書の発行元の組織。
ssl_issuer_organization_unit	SSL 証明書の発行元の組織単位。
ssl_serial_number	発行元 CA によって割り当てられた SSL 証明書のシリアル番号。
ssl_subject_common_name	SSL 証明書の件名共通名。これは一般に証明書の件名のホストとドメイン名ですが、その他の情報が含まれていることもあります。
ssl_subject_country	SSL 証明書の件名の国。
ssl_subject_organization	SSL 証明書の件名の組織。
ssl_subject_organization_unit	SSL 証明書の件名の組織単位。
storage	ファイルの保存ステータス。値は次のとおりです。 • File Stored • Unable to Store File • File Size is Too Large • File Size is Too Small • Unable to Store File • File Not Stored, Disposition Unavailable
threat_name	脅威の名前。
timestamp	ファイル タイプを識別するために十分なファイルが送信された時点を示す UNIX タイムスタンプ。

表 10-2　file_event のフィールド(続き)

フィールド	説明
url	ファイル送信元の URL。
user_id	宛先ユーザの内部識別番号。宛先ユーザとは、イベント発生前に宛先ホストにログインした最終ユーザです。
username	user_id に関連付けられている名前。
web_application_id	Web アプリケーションの内部識別番号(該当する場合)。
web_application_name	Web アプリケーションの名前(該当する場合)。

file_event の結合

次の表に、**file_event** テーブルで実行できる結合について説明します。

表 10-3　file_event の結合

このテーブルで結合に使用するフィールド	結合できるフィールド
application_id	application_info.application_id application_host_map.application_id application_tag_map.application_id rna_host_service_info.application_protocol_id rna_host_client_app_payload.web_application_id rna_host_client_app_payload.client_application_id rna_host_client_app.client_application_id rna_host_client_app.application_protocol_id rna_host_service_payload.web_application_id

file_event のサンプル クエリ

次のクエリは、特性が CLEAN ではない最大 10 件のファイル イベントと、アプリケーション名、接続情報、およびファイルの名前を返します。

```
SELECT file_event.application_name, file_event.connection_sec, file_event.counter,

file_event.file_name

FROM file_event

WHERE file_event.disposition != "CLEAN" limit 10;
```

廃止テーブル

この付録では、以前のリリースで使用され、現在は廃止となったテーブルについて説明します。これらのテーブルを照会できますが、フィールドの値が正しくない可能性があります（ほとんどの場合フィールド値は `null` です）。これらのテーブルでは結合はサポートされていません。

表 A-1　廃止テーブル

テーブル	置き換わるテーブル	最終使用バージョン：
application_ip_map	application_host_map（6-5 ページ）	5.1.1
rna_ip_host	rna_host（6-13 ページ）	5.1.1
rna_ip_host_attribute	rna_host_attribute（6-15 ページ）	5.1.1
rna_ip_host_client_app	rna_host_client_app（6-17 ページ）	5.1.1
rna_ip_host_client_app_payload	rna_host_client_app_payload（6-19 ページ）	5.1.1
rna_ip_host_os	rna_host_os（6-29 ページ）	5.1.1
rna_ip_host_os_vulns	rna_host_os_vulns（6-30 ページ）	5.1.1
rna_ip_host_sensor	rna_host_sensor（6-33 ページ）	5.1.1
rna_ip_host_service	rna_host_service（6-35 ページ）	5.1.1
rna_ip_host_service_banner	rna_host_service_banner（6-37 ページ）	5.1.1
rna_ip_host_service_info	rna_host_service_info（6-38 ページ）	5.1.1
rna_ip_host_service_payload	rna_host_service_payload（6-42 ページ）	5.1.1
rna_ip_host_service_subtype	rna_host_service_subtype（6-44 ページ）	5.1.1
rna_ip_host_service_vulns	rna_host_service_vulns（6-46 ページ）	5.1.1
rna_ip_host_third_party_vuln	rna_host_third_party_vuln（6-47 ページ）	5.1.1
rna_ip_host_third_party_vuln_bugtraq_id	rna_host_third_party_vuln_bugtraq_id（6-49 ページ）	5.1.1
rna_ip_host_third_party_vuln_cve_id	rna_host_third_party_vuln_cve_id（6-50 ページ）	5.1.1
rna_ip_host_third_party_vuln_rna_id	rna_host_third_party_vuln_rna_id（6-52 ページ）	5.1.1
rna_ip_host_user_history	user_ipaddr_history（6-60 ページ）	5.1.1
rna_mac_host	rna_host_mac_map（6-27 ページ）	5.1.1

表 A-1　廃止テーブル

テーブル	置き換わるテーブル	最終使用バージョン:
rna_mac_host_sensor	rna_host_mac_map（6-27 ページ）	5.1.1
rna_mac_ip_map	rna_host_ip_map（6-26 ページ） rna_host_mac_map（6-27 ページ）	5.1.1

A

C

D

F

H

I

N

R

S

T

U

W