HP NV Analytics

ソフトウェアバージョン: 12.50

ユーザーズ・ガイド

ドキュメントリリース日 : 2015 年 8 月 (英語版)
ソフトウェアリリース日 : 2015 年 8 月

ご注意

保証

HP製品、またはサービスの保証は、当該製品、およびサービスに付随する明示的な保証文によってのみ規定されるものとします。ここでの記載は、追加保証を提供するものではありません。ここに含まれる技術的、編集上の誤り、または欠如について、HPはいかなる責任も負いません。

ここに記載する情報は、予告なしに変更されることがあります。

権利の制限

機密性のあるコンピューターソフトウェアです。これらを所有、使用、または複製するには、HPからの有効な使用許諾が必要です。商用コンピューターソフトウェア、コンピューターソフトウェアに関する文書類、および商用アイテムの技術データは、FAR12.211および12.212の規定に従い、ベンダーの標準商用ライセンスに基づいて米国政府に使用許諾が付与されます。

著作権について

商標について

Adobe ® は、Adobe Systems Incorporated (アドビシステムズ社) の登録商標です。

Microsoft ® およびWindows ® は、米国におけるMicrosoft Corporationの登録商標です。

UNIX ® は、The Open Groupの登録商標です。

ドキュメントの更新情報

このマニュアルの表紙には、以下の識別情報が記載されています。

- ソフトウェアバージョンの番号は、ソフトウェアのバージョンを示します。
- ドキュメントリリース日は、ドキュメントが更新されるたびに変更されます。
- ソフトウェアリリース日は、このバージョンのソフトウェアのリリース期日を表します。

更新状況、およびご使用のドキュメントが最新版かどうかは、次のサイトで確認できます。
https://softwaresupport.hp.com

このサイトを利用するには、HP Passportへの登録とサインインが必要です。HP Passport IDの登録は、次のWebサイトから行なうことができます。https://softwaresupport.hp.com にアクセスして、[**Register**] をクリックしてください。

サポート

HPソフトウェアサポートオンラインWebサイトを参照してください。https://softwaresupport.hp.com

このサイトでは、HPのお客様窓口のほか、HPソフトウェアが提供する製品、サービス、およびサポートに関する詳細情報をご覧いただけます。

HPソフトウェアオンラインではセルフソルブ機能を提供しています。お客様のビジネスを管理するのに必要な対話型の技術サポートツールに、素早く効率的にアクセスできます。HPソフトウェアサポートのWebサイトでは、次のようなことができます。

- 関心のあるナレッジドキュメントの検索
- サポートケースの登録とエンハンスメント要求のトラッキング
- ソフトウェアパッチのダウンロード
- サポート契約の管理
- HPサポート窓口の検索
- 利用可能なサービスに関する情報の閲覧
- 他のソフトウェアカスタマーとの意見交換
- ソフトウェアトレーニングの検索と登録

一部のサポートを除き、サポートのご利用には、HP Passportユーザーとしてご登録の上、サインインしていただく必要があります。また、多くのサポートのご利用には、サポート契約が必要です。HP Passport IDを登録するには、次のWebサイトにアクセスしてください。https://softwaresupport.hp.com にアクセスして、[**Register**] をクリックしてください。

アクセスレベルの詳細については、次のWebサイトをご覧ください。
https://softwaresupport.hp.com/web/softwaresupport/access-levels

HP Software Solutions統合とベストプラクティス

HP Software Solutions Now（https://h20230.www2.hp.com/sc/solutions/index.jsp）では、HPソフトウェアのカタログ記載製品がどのような仕組みで連携、情報の交換、ビジネスニーズの解決に対応するのかご確認いただけます。

Cross Portfolio Best Practices Library（https://hpln.hp.com/group/best-practices-hpsw）では、ベストプラクティスに関するさまざまなドキュメントや資料をご覧頂けます。

このPDF版オンラインヘルプについて

本ドキュメントはPDF版のオンラインヘルプです。このPDFは、ヘルプ情報から複数のトピックを簡単に印刷したり、オンラインヘルプをPDF形式で閲覧できるようにするために提供されています。このコンテンツは本来、オンラインヘルプとしてWebブラウザーで閲覧することを想定して作成されているため、トピックによっては正しいフォーマットで表示されない場合があります。また、インタラクティブトピックの一部はこのPDF版では提供されません。これらのトピックは、オンラインヘルプから正しく印刷することができます。

目次

第1章: NV Analytics

NV Analytics は，ネットワーク上でのアプリケーションのパフォーマンスに悪影響を与える要因を特定するために役立ちます。NV Analytics は，パケット・リスト・データに基づいて分析を実行し，結果のデータを，アプリケーションの動作に関する有用な情報を記載したレポートとして表示します。

HTTP，HTTPS およびその他のプロトコルのウォーターフォール図による分析から，個別リソースのサイズとロード時間を視覚的に把握して，トランザクションの応答時間をすばやく分析し，最適化可能な分野を簡単に特定できます。

ここでは次の項目について説明します。

- **システム要件：**NV Analytics の最小ホスト要件を示します。詳細については「システム要件」(7ページ)を参照してください。
- **ソフトウェアのインストール：**NV Analytics のインストール手順を説明します。詳細については「NV Analytics のインストール」(8ページ)を参照してください。
- **ログ・ファイル：**NV Analytics ログ・ファイルの場所を示します。詳細については「ログ・ファイル」(11ページ)を参照してください。

システム要件

NV Analytics の最小要件は次のとおりです。

プロセッサ	クワッド・コア 2.5 GHz 以上
メモリ	4 GB RAM
ハード・ディスク	1 GB の空きディスク容量
デスクトップ・オペレーティング・システム	• Windows Server 2008 R2 SP1（64 ビット版） • Windows 7 SP1（32/64 ビット版） • Windows Server 2012 R2 • Windows 8.1（64 ビット版）
ブラウザ	• Internet Explorer 9.0 以上 • FireFox • Chrome • Safari
Microsoft Office（レポートのエクスポート用）	• Office 2007

	• Office 2010 • Office 2013

NV Analytics のインストール

注: NV Analytics レポートは，Network Virtualization の一部です。VuGen マシンに NV をインストールした場合，スタンドアロンの NV Analytics バージョンをインストールしなくても，VuGen で直接 NV Analytics レポートを表示できます。

スタンドアロン・バージョンの NV Analytics をすでに使用している場合，更新された 12.50 バージョンをインストールできます。これは NV のインストール・ファイルに含まれています。

ソフトウェアの前提条件

NV Analytics をインストールする前に，Wireshark バージョン 1.10.8 以降がインストールされている必要があります。

次のソフトウェアは，存在しない場合はインストール中にインストールされます。

- 次のどれも存在しない場合は，Java Runtime Environment 8.0 アップデート 45（32 ビット）がインストールされます。
  - JRE 6.0 アップデート 24 以降（32 ビット）
  - JRE 7.0（32 ビット）
  - JRE 8.0（32 ビット）
- Microsoft .NET Framework 4.5.2 Full
- Microsoft Silverlight 5.1.30514

ソフトウェアのインストール

NV Analytics をインストールするには，Analytics セットアップ・ファイル **Analytics_setup.exe** を（管理者として）実行し，［**Install（インストール）**］をクリックし，画面に表示される指示に従います。

NV Analytics のアップグレード：インストールを実行すると，インストーラは NV Analytics の前のバージョンがインストールされているかどうかを確認します。インストールされているバージョンに応じて，インストーラは次のいずれかを実行します。

- 前のバージョンをアンインストールし，インストールを続行します。
- 前のバージョンを先にアンインストールしてからインストールを再実行するように指示を表示します。

NV Analytics をインストールした後で，コンピュータを再起動する必要があります。

インストール後には，Windows の［スタート］メニューから NV Analytics にアクセスします。［**すべてのプログラム**］ > ［**HP Software**］ > ［**HP Network Virtualization**］ > ［**NV Analytics**］ > ［**NV Analytics**］。

Windows 8.x 以降では，［**スタート**］または［**アプリ**］画面から NV Analytics に直接アクセスできます。

サイレント・インストール

注: NV Analytics のインストールの前に Wireshark がインストールされていない場合，インストールは中止されます。

NV Analytics のサイレント・インストールを行うには：

1. NV Analytics セットアップ・ファイルをインストール・パッケージから適当な場所にコピーします。
2. Windows の［スタート］メニューから［**すべてのプログラム**］ > ［**アクセサリ**］で［**コマンドプロンプト**］を右クリックして，［**管理者として実行**］をクリックします。
3. コマンド・ウィンドウで，ステップ 1 でファイルをコピーした場所に移動し，次のコマンドと必要なコマンド・ライン・オプションを入力します。

   Analytics_setup.exe /s /v"/qn コマンド・ライン・オプション"

   コマンド・ライン・オプション（* は必須コマンド・ライン・オプションを示します）：

   - PORT= < ポート番号 >

     NV Analytics に接続するためのポート。

     別の Network Virtualization コンポーネントがすでにインストールされている場合，NV Analytics は同じポートを使用します。

   - ENABLE_REMOTE=TRUE | FALSE

     ファイアウォールでポートを開きます。

     標準設定は TRUE です。

   - DATA_FOLDER=\" < データ・フォルダのパス > \"

     内部アプリケーション・データおよびユーザ・データが保存される場所。

     標準設定は < 共通アプリケーション・データ・フォルダ > \HP\NV（Windows 7 では C:\ProgramData\HP\NV）です。

   - INSTALLDIR=\" < インストール・フォルダのパス > \"

     アプリケーション・ファイルがインストールされる場所。

     標準設定は C:\Program Files (x86)\HP\NV\ です。

   - REBOOT_IF_NEED=TRUE | FALSE

再起動が必要な場合，インストール完了後にコンピュータを自動的に再起動します。

標準設定は TRUE です。

サイレント・アンインストール

NV Analytics のサイレント・アンインストールを行うには：

1. Analytics_setup.exe セットアップ・ファイルをインストール・パッケージから適当な場所にコピーします。
2. Windows の［スタート］メニューから［**すべてのプログラム**］>［**アクセサリ**］で［**コマンド プロンプト**］を右クリックして，［**管理者として実行**］をクリックします。
3. コマンド・プロンプト・ウィンドウで，ステップ 1 でファイルをコピーした場所に移動し，次のコマンドと必要なコマンド・ライン・オプションを入力します。

Analytics_setup.exe /s /removeonly /v"/qn コマンド・ライン・オプション"

コマンド・ライン・オプション（コマンド・ライン・オプションはすべて省略可能です）：

- REBOOT_IF_NEED=TRUE | FALSE

  再起動が必要な場合，アンインストール完了後にコンピュータを自動的に再起動します。

  標準設定は TRUE です。

- FORCE_REBOOT=TRUE | FALSE

  再起動が必要かどうかにかかわらず，アンインストール完了後にコンピュータを自動的に再起動します。

  標準設定は FALSE です。

- DELETE_DATA=TRUE | FALSE

  保存されている NV Analytics データをすべて削除します。

  標準設定は FALSE です。

インストール・ログ・ファイル

インストール・ログは C:\HP Log の下にあります。ログ・ファイルの名前は次のとおりです。

< 製品名 > _ < 日付 > _ < 時刻 > .log

例：

HP NV for Load Generator_6-4-2015_15-29-27.log

HP NV for Controller_6-4-2015_15-37-38.log

ログ・ファイル

HP Network Virtualization 製品のログ・ファイルは，< インストール・ディレクトリ > \logs にあります。標準設定では \Program Files\HP\NV\logs or \Program Files (x86)\HP\NV\logs です。

第2章: ライセンスの設定

HP Network Virtualization コンポーネントをインストールしたら，NV Analytics ライセンスをインストールする必要があります。NV Analytics ライセンスは，VuGen と統合されている NV Analytics 分析機能と，NV Analytics のスタンドアロン・バージョンを使用するために必要です。

NV Analytics ライセンスは NV License Manager から管理されます。

注: NV のエミュレーション機能全般と NV グローバル・ライブラリへのアクセスは，Network Virtualization 仮想ユーザ・ライセンスに基づいて，LoadRunner/Performance Center ライセンスによって管理されます。

関連作業：

- 「ライセンス方法」(12ページ)について知る
- 「NV License Manager へのアクセス」(13ページ)
- 「シート・ライセンスのインストール」(13ページ)
- 「ライセンス・サーバのセットアップ」(14ページ)
- 「フローティング・ライセンスの使用」(14ページ)

ライセンス方法

Network Virtualization 製品に対して利用できるライセンス方法を以下に示します。

シート・ライセンス

シート・ライセンスは，特定のコンピュータ上の特定の Network Virtualization 製品に対して作成され，別のコンピュータに移転することはできません。

フローティング・ライセンス

フローティング・ライセンスを使用する場合，ライセンスはライセンス・サーバに保持され，必要に応じてチェックアウトされます。フローティング・ライセンスを使用するには，HP AutoPass License Server 8.3 以降がネットワークにインストールされている必要があります。

ライセンスを使用し終わったら，ライセンス・サーバにライセンスを返却して，他の NV インストールで使用できるようにします。

試用ライセンス

NV Analytics には，30 日間の試用ライセンスが付属しています。試用ライセンスでは，製品のすべての機能を利用できます。

VuGen での NV Analytics レポート：試用期間は，VuGen で NV Analytics の機能を初めて使用したときに開始されます。

NV Analytics：試用期間は，分析を初めて実行したときに開始されます。

注意: 仮想マシンに Network Virtualization 製品をインストールした場合，試用ライセンスが開始された後はマシンを複製しないでください。

NV License Manager へのアクセス

NV License Manager にアクセスするには，次の方法が使用できます。

- Windows の［**スタート**］メニューで，［**すべてのプログラム**］ > ［**HP Software**］ > ［**HP Network Virtualization**］ > ［**NV License Manager**］を選択します。

  Windows 8.x 以降では，［**スタート**］または［**アプリ**］画面から NV License Manager にアクセスできます。
- Web ブラウザから，次の URL にアクセスします。

  ```
  http:// < ホスト名 > : < ポート > /shunra/license/
  ```

  例：

  ```
  http://198.51.100.24:8182/shunra/license/
  ```

シート・ライセンスのインストール

シート・ライセンスを使用する場合，NV Analytics を使用するすべてのコンピュータにライセンスを適用する必要があります。

1. 必要なコンピュータ上で NV License Manager を開きます。
2. ［**ライセンスの更新**］をクリックします。
3. ［**更新手段**］：［**ファイルまたはキー**］をクリックします。
4. 下に表示される**マシン・コード**をコピーします。
5. **HP ライセンス・ポータル**をクリックして，HP ライセンス・サイトに接続します。
6. 有効なライセンス EON（Entitlement Order Number）を入力します。表示されるページで，さきほどコピーしたマシン・コードを入力し，ライセンス・ファイルを生成します。

7. ライセンス・キーを入力するか，［**ライセンス ファイル**］ボックスの右側のフォルダ・アイコン  をクリックし，ライセンス・ファイルを見つけてアップロードします。
8. ［**更新**］をクリックします。更新されたライセンスの詳細が NV License Manager メイン・ページに表示されます。

ライセンス・サーバのセットアップ

フローティング・ライセンス方法を使用する場合，ライセンスはライセンス・サーバに保持され，必要に応じてチェックアウトされます。Network Virtualization では，HP AutoPass ライセンス・サーバを使用して，フローティング・ライセンスを管理します。

1. ライセンス・サーバをホストするマシンを選択します。ライセンス・サーバは，NV Analytics レポートを生成するすべてのマシンからアクセス可能である必要があります。
2. HP AutoPass ライセンス・サーバをインストールします。インストール・フォルダ **autopass-8.3.zip** は，Network Virtualization のインストール・ファイルと同じ場所にあります。

   インストール・ファイルは次のフォルダにあります。

   - LoadRunner： < LoadRunner インストール DVD > \Additional Components\HP NV\
   - Performance Center： < Performance Center インストール DVD > \AdditionalComponents\HPNV\

   適切なセットアップ・ファイルを展開して実行します。詳細については，同じフォルダにある AutoPass ドキュメントを参照してください。
3. HP ライセンス・ポータル: http://h30580.www3.hp.com/に接続し，有効なライセンス EON（Entitlement Order Number）を入力します。HP サイト上の指示に従ってライセンスを取得し，ライセンス・サーバにインストールします。

フローティング・ライセンスの使用

フローティング・ライセンスはライセンス・サーバに保持され，必要に応じてチェックアウトされます。ライセンスをチェックアウトする際には日数を指定し，その日数に達するとライセンスは自動的にライセンス・サーバに返却されます。期日前にライセンスを返却することもできます。

関連作業：

- 「ライセンスのチェックアウト」(15ページ)
- 「ライセンスの返却」(15ページ)

ライセンスのチェックアウト

注: ライセンスをチェックアウトできる最大日数は，HP AutoPass ライセンス・サーバで設定できます。詳細については AutoPass のドキュメントを参照してください。

1. ライセンスをチェックアウトするマシンで NV License Manager を開きます。詳細については，「NV License Manager へのアクセス」(13ページ)を参照してください。
2. ［**ライセンスの更新**］をクリックします。
3. ［**更新手段**］：［**ライセンス サーバ**］を選択します。
4. ［**ライセンス サーバのアドレス**］フィールドで，ライセンス・サーバ がインストールされているマシンを選択します。ライセンス・サーバ がリストに表示されない場合は，アドレスを入力します。
5. ［**ライセンス期間（日）**］フィールドで，ライセンスをチェックアウトする期間を選択します。標準設定では，ライセンスをチェックアウトできる最大日数は 30 日です。
6. ［**詳細設定**］の下で，次の設定を行います。
   a. AutoPass ライセンス・サーバのポートを設定します。標準設定では 5814 です。ライセンスをチェックアウトするマシンは，ライセンス・サーバ にアクセスできる必要があります。
   b. NV License Manager と ライセンス・サーバ の間で保護された通信を使用するには，［**保護された通信を使用**］を選択します。
7. ［**ライセンスのチェックアウト**］をクリックします。ライセンスが ライセンス・サーバ からチェックアウトされます。

ライセンスの返却

1. ライセンスを返却するマシンで NV License Manager を開きます。
2. ［**ライセンスの更新**］ボタンをクリックします。
3. ［**更新手段**］：［**ライセンス サーバ**］を選択します。
4. ［**ライセンスの返却**］をクリックします。ライセンスが ライセンス・サーバ に返却されます。

第3章: 結果の分析

ここでは，NV Analytics から生成された結果を表示して分析する方法を説明します。

NV Analytics レポートは，ネットワーク仮想化パケット・リストを使用してキャプチャされたデータから構成されます。このデータは，後で処理されてわかりやすいレポートの形で表示されます。レポートを調べることで，問題の存在を突き止めることができます。

分析レポートには，各トランザクションのブレークダウンに関する詳細データが記載されています。トランザクションでアップロードまたはダウンロードされる各リソースの統計が，表とグラフの両方の形式で表示されます。ロード時間，コンポーネントのダウンロード分析，応答時間のブレークダウン，受信されたエラーの詳細といった精密なパフォーマンスデータが得られます。モバイルおよび非モバイル・トランザクションのパフォーマンスを改善するためのパフォーマンス最適化の推奨事項が示されます。

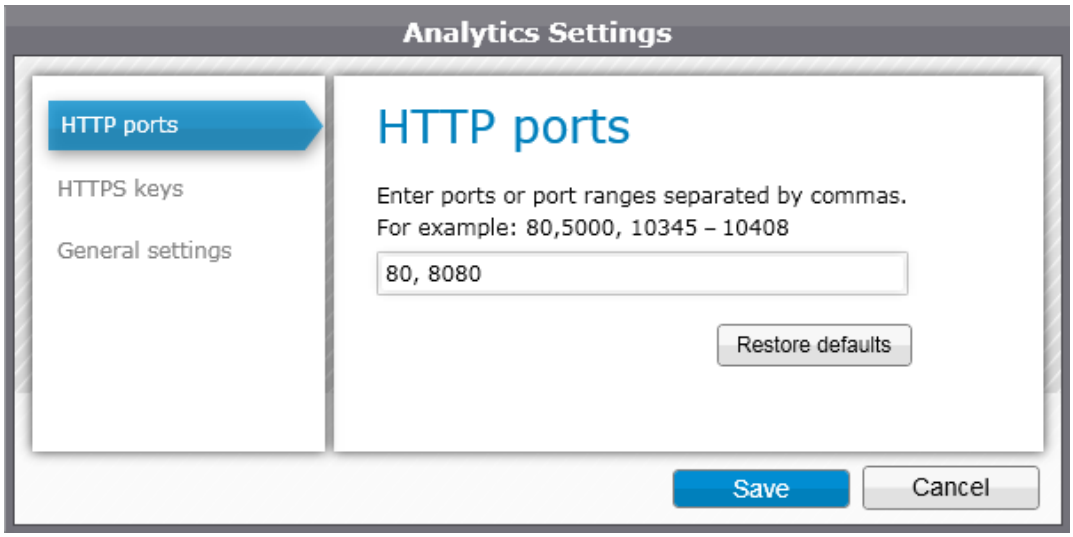
この項には，次の内容が含まれます。

設定

分析設定を指定するには，［Welcome（ようこそ）］ページの［Setting（設定）］アイコンをクリックします。設定はメイン・ページからも表示できます。

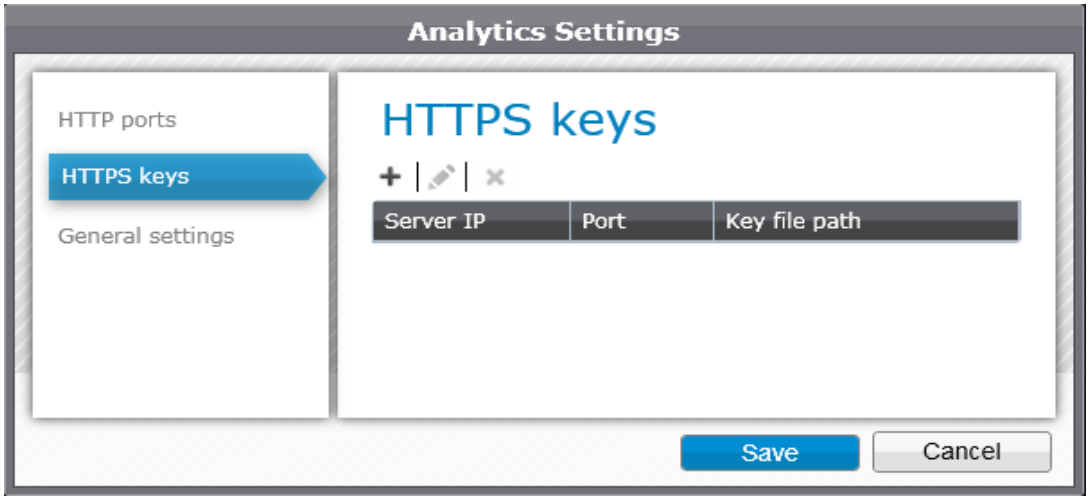
HTTP ポート設定

標準設定のポートを変更するには，［HTTP ports（HTTP ポート）］タブを選択して，1 つ以上のポートまたは範囲をカンマで区切って追加します。テスト対象のアプリケーションが使用しているポートを追加します。HTTP/S 分析は指定したポートに対して実行されます。



HTTPS キー

セキュア・データの分析を可能にするには，HTTPS キーを入力します。

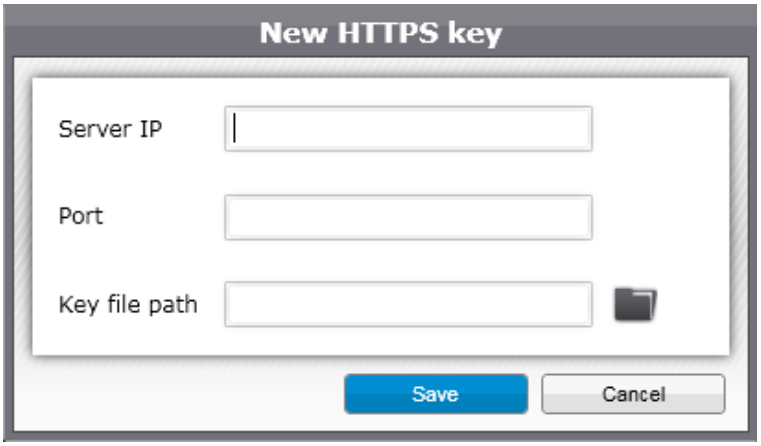


HTTPS キーを追加するには：

1. "+" 記号をクリックし，［New HTTPS Key（新規 HTTPS キー）］ウィンドウで，必要な情報を入力します。
2. 情報を編集するには，鉛筆アイコンをクリックします。キーを削除するには，"x" をクリックします。

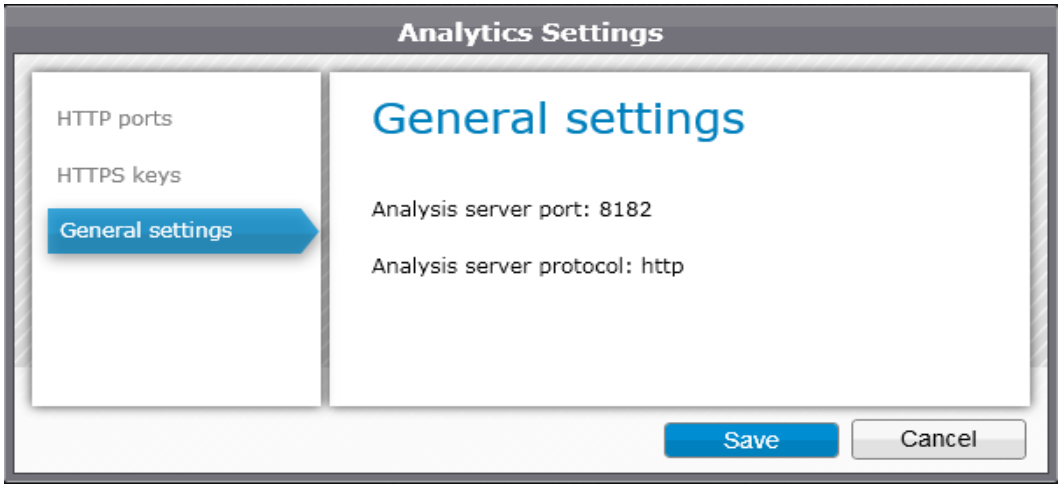
   アプリケーション・サーバまたはデバッグ用プロキシ（インストールしている場合）の IP，

ポート，秘密キーを指定する必要があります。



一般設定

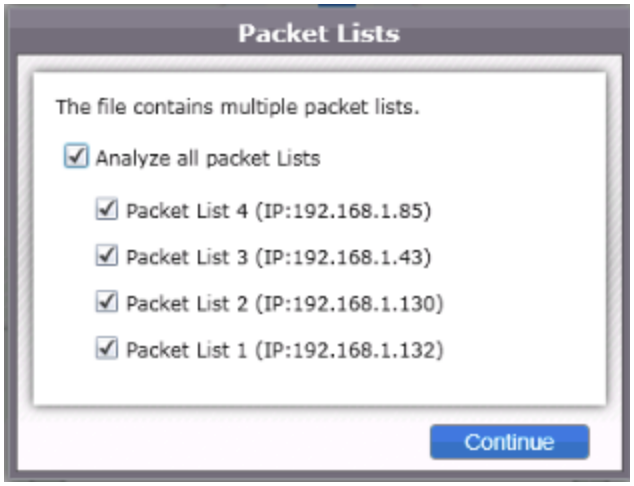
NV Analytics サーバ・ポートを表示します。



Network Virtualization テスト結果ファイルの分析

1. NV Analytics を開始し，［**Open File（ファイルを開く）**］をクリックします（ポートの指定やその他の設定の定義方法については，「設定」(16ページ)を参照してください）。サポートされるファイル・タイプとしては，*.shunra，*.ved，*.cap，*.pcap，*.enc があります。

2. テスト結果ファイルに複数のパケット・リストが含まれる場合は，一部または全部のパケット・リストを分析対象として選択します。



3. NV Analytics ウィンドウが開いたら，ツールバーで［All Transactions（すべてのトランザクション）］をクリックしてすべてのトランザクションの表示を選択するか，特定のトランザクションを選択します。メイン・ページにはすべてのトランザクションが表示され，1 つの四角形が 1 つのトランザクションを表します。

NV Analytics 結果のエクスポート

選択したビューおよび推奨事項のデータは，.csv ファイルおよび MS Word 形式でエクスポートできます。

.csv 形式でのエクスポート

.csv へのエクスポートには，スループット，ウォーターフォール・グラフに表示されたネットワーク条件，およびルール，達成グレード，各ルールの違反といった推奨事項が含まれます。

.csv 形式で結果をエクスポートするには：

1. ツールバーの  アイコンをクリックします。

> **注:** ズーム・ビューを使用した場合，トランザクションの選択した部分だけがエクスポートされ，トランザクション全体はエクスポートされません。

2. ［Export settings（エクスポート設定）］ダイアログで，レポートの全部または一部と，任意のオプションまたは全部のオプションを選択し，［Export（エクスポート）］をクリックします。

MS Word 形式でのエクスポート

MS Word 形式でエクスポートする場合，HTTP レポートおよび推奨事項をエクスポートできます。

注: ズーム・ビューを使用した場合，トランザクションの選択した部分だけがエクスポートされ，トランザクション全体はエクスポートされません。

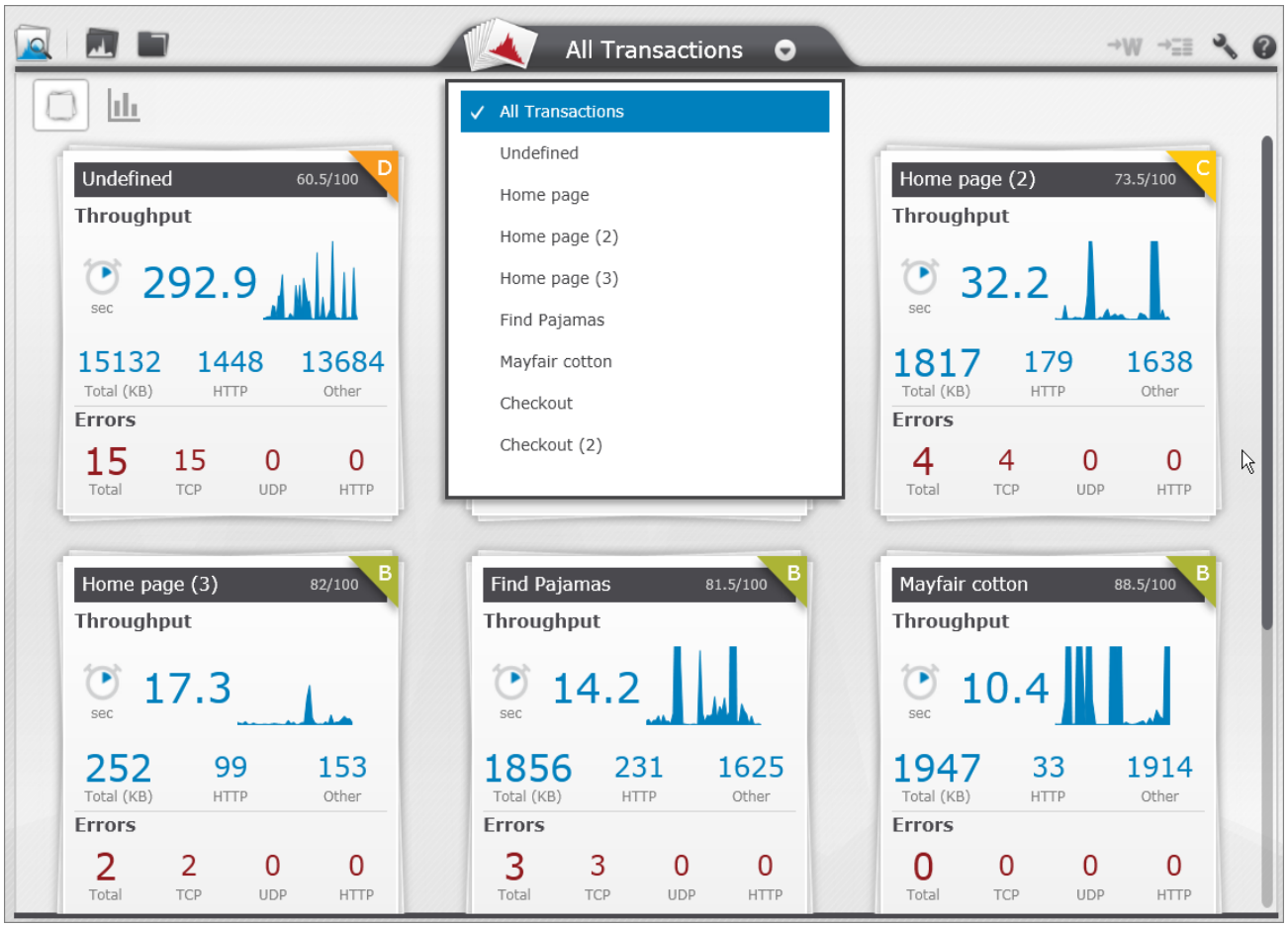
MS Word 形式で結果をエクスポートするには：

ツールバーの アイコンをクリックします。

ズーム・ビューを使用した場合，トランザクションの選択した部分だけがエクスポートされ，トランザクション全体はエクスポートされません。ハイライトされている検索結果は，エクスポートされた結果でもハイライトされます。

レポートの表示

［Overview（概要）］ページには，分析したテスト結果ファイルに含まれるトランザクションが表示されます。任意のトランザクションをクリックすると，そのトランザクションの詳細が表示されます。各トランザクションのセクションには，スループットおよびエラーの詳細と，文字とパーセンテージから成るパフォーマンス・スコアが表示されます。表示は対話型であり，［Total（合計）］などの任意のメトリックをクリックすると，そのメトリックの詳細なレポートが表示されます。



NV Analytics の［Overview（概要）］ページには，次のカテゴリに関する各トランザクションの詳細なブレークダウンが表示されます。

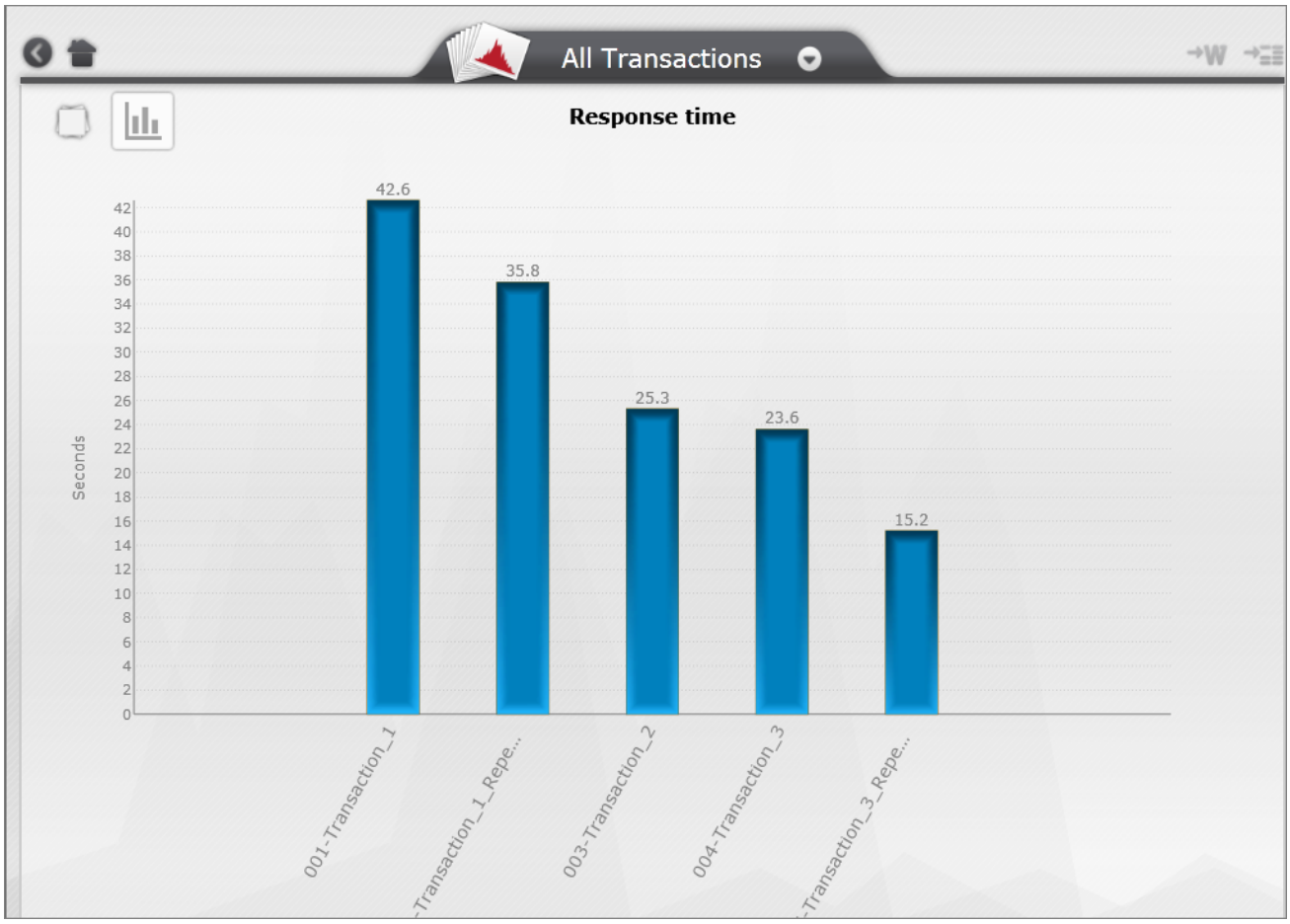
- 「応答時間」(21ページ)
- 「サマリ」(22ページ)
- 「一般分析」(24ページ)
- 「エンドポイント遅延」(27ページ)
- 「TCP/UDP エラーおよびセッション」(28ページ)
- 「HTTP 分析」(29ページ)
- 「HTTP 最適化」(33ページ)
- 「HTTP リソースおよび応答」(35ページ)

レポートのリスト（左側）の下には，トランザクションの合計時間とネットワーク時間が表示されます。トランザクションの作成時に説明が追加された場合は，それも表示されます。

各レポート・ビューで，メイン・ページに戻るにはホーム・アイコン，前のビューに戻るには戻るアイコン，次のビューに進むには進むアイコンをクリックします。

応答時間

［All Transactions（すべてのトランザクション）］タブで，左上の  アイコンをクリックすると，すべてのトランザクションのトランザクション応答時間が表示されます。

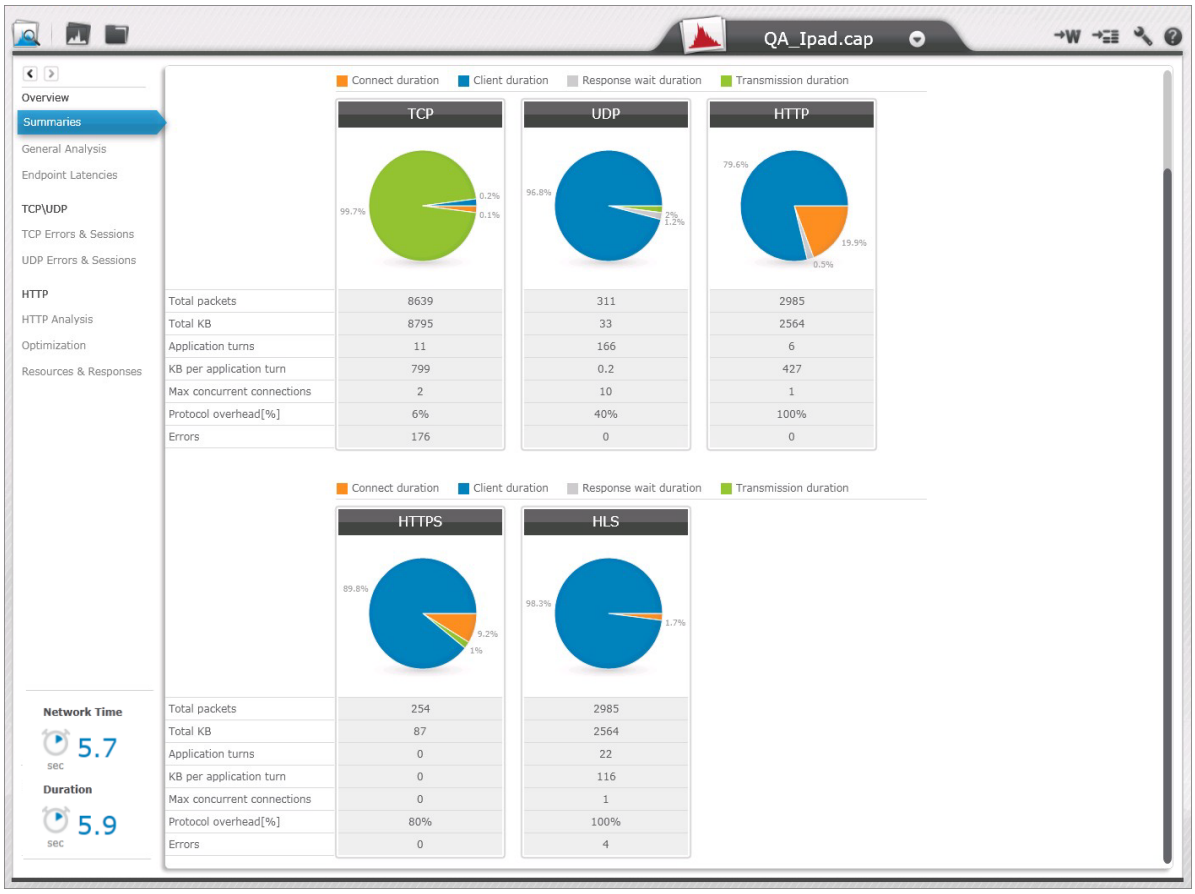


任意のバーをクリックすると，そのトランザクションの HTTP 分析が表示されます。

サマリ

プロトコルに基づくトランザクションのクライアント・ネットワーク・サーバの内訳を示します。結果は円グラフで示され，その他に次のプロトコルに関する追加詳細が表示されます。

- TCP
- UCP
- HTTP
- HTTPS（セキュア通信）
- HLS（HTTP ライブ・ストリーミング）



クライアント・ネットワーク・サーバの内訳

チャートの凡例は表の下に示されています。各円グラフに示されたフィールドの値は次のとおりです。

- **Connect duration（接続時間）**：クライアントがサーバに接続していた時間の割合。たとえば，TCP の要求や，SSL のトリプル・ハンドシェーク（セキュア・チャネルの確立）
- **Client duration（クライアント時間）**：クライアントが処理を行っていた時間の割合（サーバの応答を待っていた時間は含みません）
- **Response wait duration（応答待ち時間）**：サーバの応答を待っていた時間の割合
- **Transmission duration（伝送時間）**：データがダウンロードまたはアップロードされていた時間の割合

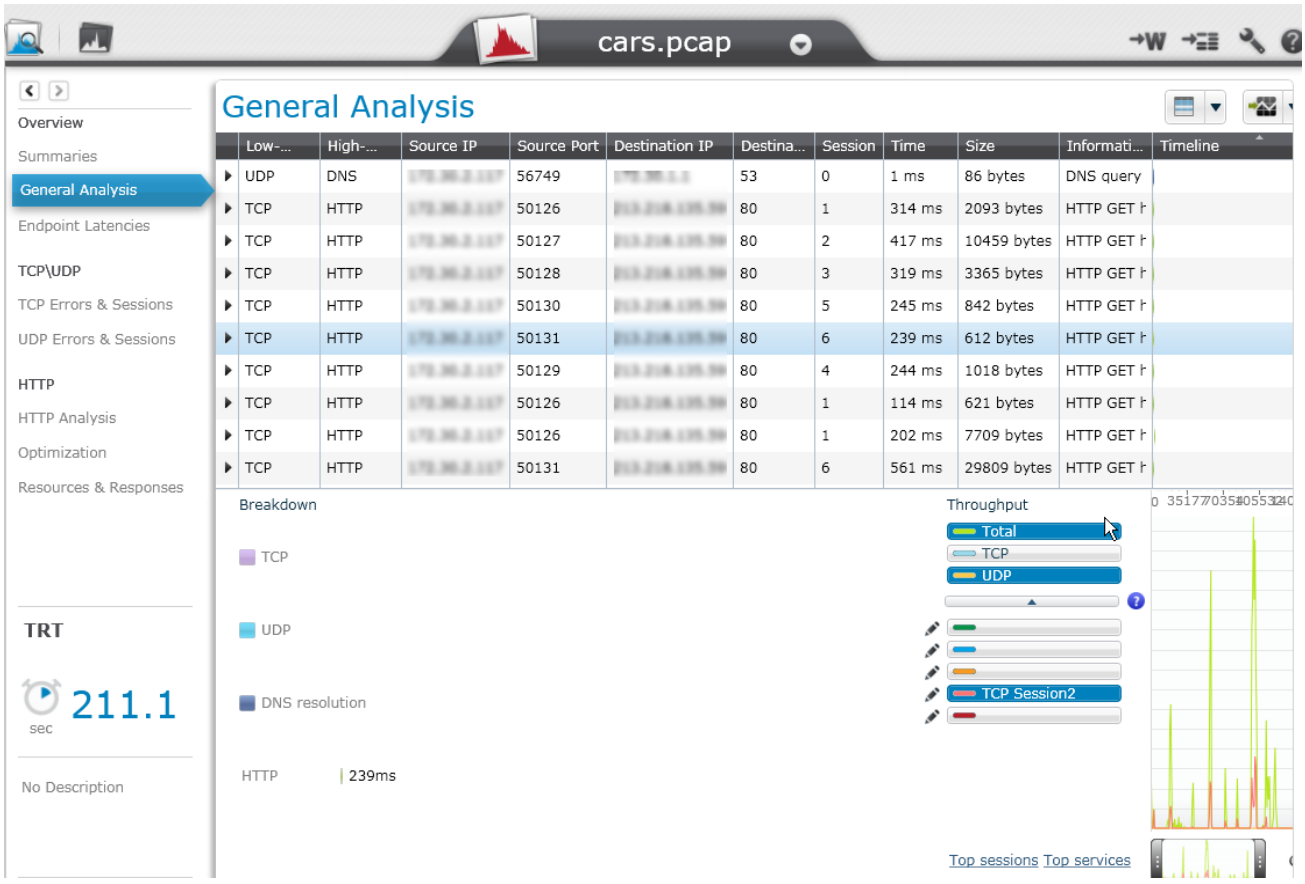
パラメータ

- **Total packets（合計パケット数）**：プロトコルに関連するパケットの合計数
- **Total KB（合計 KB）**：プロトコルに関連する合計スループット（TCP，IP などのヘッダを含む）

- **Application turns（アプリケーション転換数）**：プロトコルごとの，通信フローが要求から応答に転換した回数
- **KB per application turn（アプリケーション転換あたりの KB）**：プロトコルごとの，アプリケーション転換 1 回あたりのスループットの平均
- **Max concurrent connections（最大同時接続数）**：プロトコルごとの同時接続の最大数
- **Protocol overhead %（プロトコルオーバーヘッド %）**：プロトコルごとの，ヘッダなどの非データ要素に使用された合計スループットの割合
- **Errors（エラー）**：トランザクションで発生したエラーの数

一般分析

一般分析には，すべてのプロトコルのすべてのサブトランザクションの詳細が表示されます。



表示オプション

一般分析レポートと HTTP 分析レポートのツールバーから，次のオプションが利用可能です。最初の 2 つのオプションは，表の右クリックメニューでも利用可能です。

- ハイライト・オプション：  をクリックして，同じソース IP などに基づいてリソースをハイライトするオプションを選択します。
- グラフに表示オプション：  をクリックして，セッション，サービスなどをグラフに表示します。
- フィルタ・オプション：  をクリックして，サブトランザクションの表示を選択した基準に制限します。

サブトランザクション・パラメータ

サブトランザクションの表示を調整するには：

- 各列の昇順または降順に行をソートします（すべての表で利用可能）。
- 領域の境界をドラッグして，領域を拡大または縮小します（一般分析と HTTP 分析で利用可能）。

次のデータは表形式で表示されます。各行は 1 つのサブトランザクションを表します。

- **Low-level Protocol（低レベル・プロトコル）：**TCP および UDP を含みます。
- **High-level Protocol（高レベル・プロトコル）：**HTTP および HTTPS，DNS などを含みます。
- **Source IP（ソース IP）**
- **Source Port（ソース・ポート）**
- **Destination IP（宛先 IP）**
- **Destination Port（宛先ポート）**
- **Session（セッション）：**リソースがアップロードまたはダウンロードされた TCP または UDP セッションの数。ダウンロード中に発生したタスク（DNS 解決，TCP セットアップなど）は色分けされて示されます。
- **Time（時間）：**要求の応答時間（ミリ秒）
- **Size（サイズ）：**要求/応答に使用されたバイト数
- **Information（情報）：**プロトコル固有の情報（利用可能な場合）
- **Timeline（タイムライン）：**トランザクション・シーケンス内でのリソースの位置

要求/応答の詳細

要求/応答の詳細を表示するには，左の列の  アイコンを使用して行を展開するか折りたたみます。各要求/応答のクライアントおよびサーバ部分がバイト単位で表示されます。

ブレークダウン

各プロトコルがトラフィックに占める割合のブレークダウンが，ウィンドウ左下のグラフに表示されます。これには，TCP，UDP，DNS 解決，HTTP が含まれます。

スループット・グラフ

注: ブレークダウンおよびスループット領域を折りたたんだり展開したりするには，メインの表の下の二重矢印をクリックします。

スループット

スループット編集オプションは，グラフの表示を変更するために使用されます。

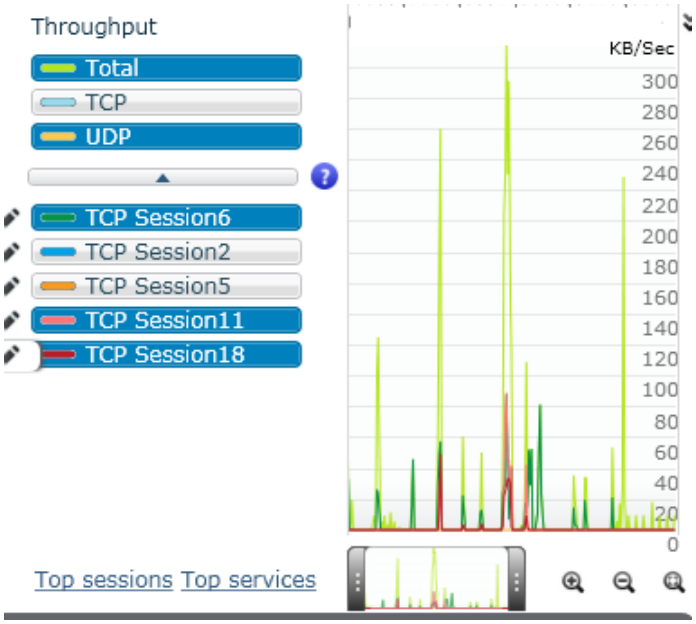
ズーム・バーを使用して，トランザクションの特定の期間に注目することができます。

- **Top Sessions（上位セッション）：**トラフィックが多いセッションを上位 5 つまで表示します
- **Top Services（上位サービス）：**トラフィックが多いサービスを上位 5 つまで表示します

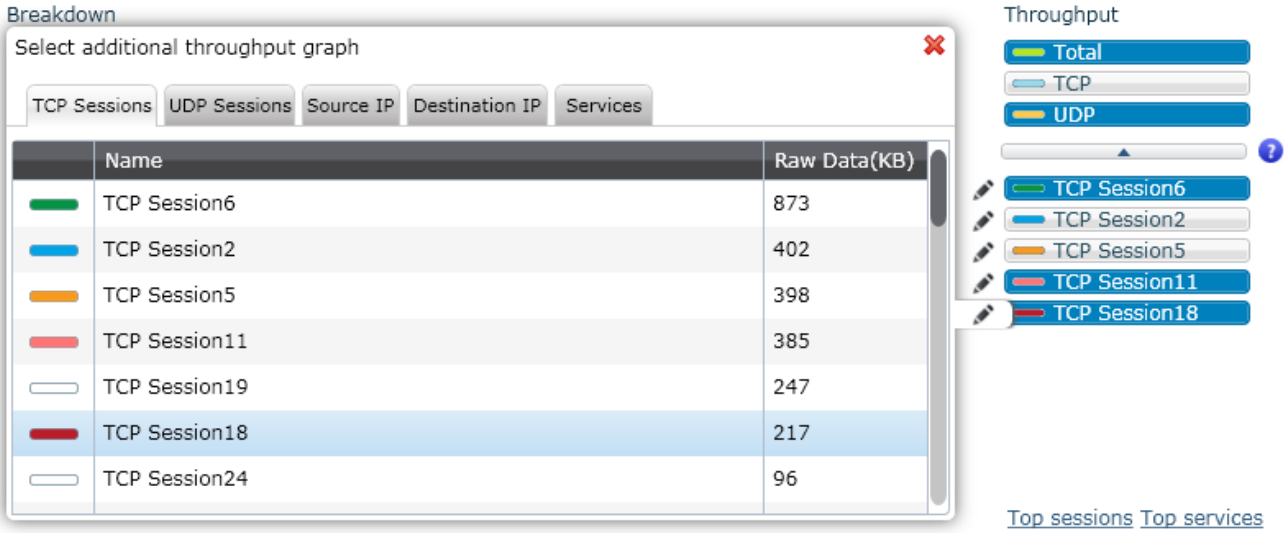
［Throughput（スループット）］領域の各タブで，バーをクリックすると，その項目（グラフ内の特定のセッションなど）が含められます。生データはスループット（KB/秒）を表します。

スループット・グラフ表示のコンポーネントを追加/削除するには：

1. タブの 1 つ（［TCP Session 18（TCP セッション 18）］など）をクリックします。グラフに TCP セッション 18 のスループットが栗色で表示されます。



2. 別のセッション，サービスなどを選択するには，表示する各セッションまたはホストの鉛筆アイコンをクリックします。



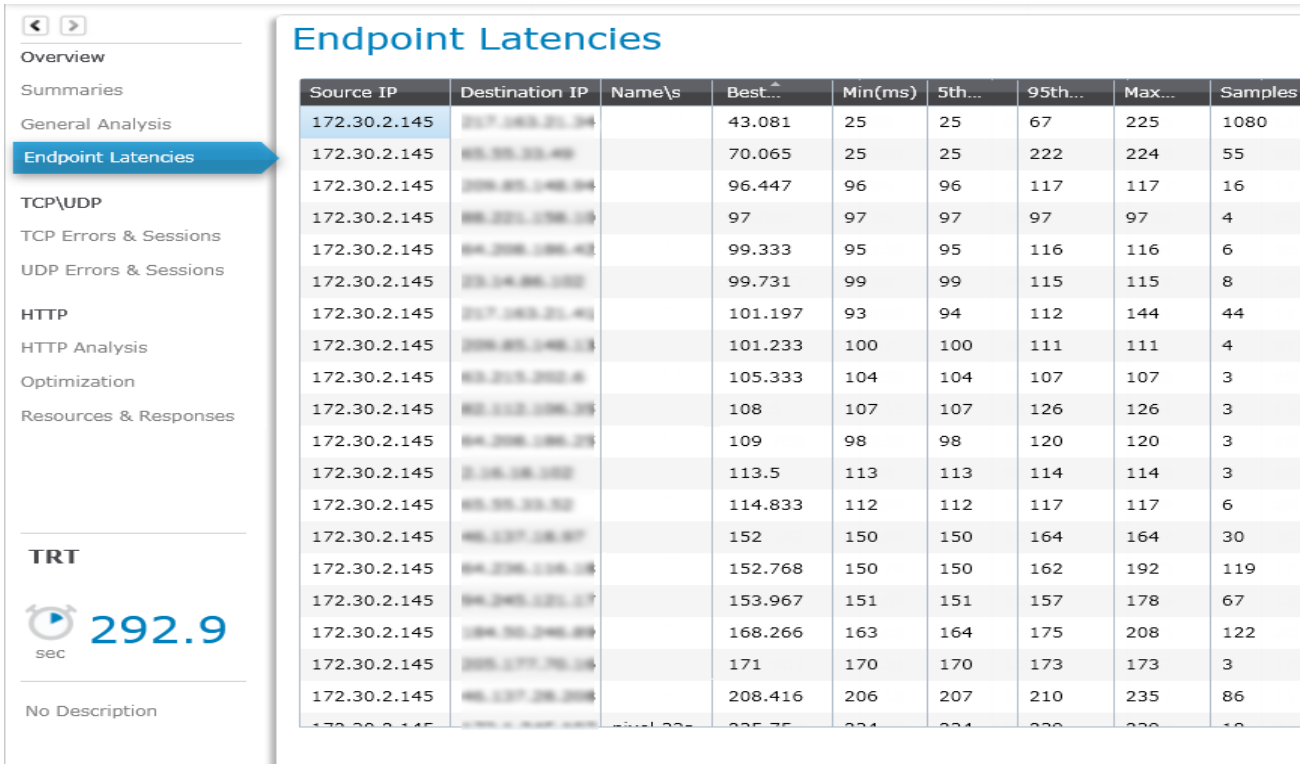
3. 次のタブが使用でき，それぞれ次の項目のトラフィック（KB/秒）を表示します。
   - TCP Sessions（TCP セッション）
   - UDP Session（UDP セッション）
   - Source IP（ソース IP）
   - Destination IP（宛先 IP）
   - Services（サービス）（IP アドレスとポート）
4. 必要な項目（［TCP セッション 16］など）またはサービス名を選択します。メトリックが選択した色でグラフに表示されます。

エンドポイント遅延

エンドポイント遅延レポートには，クライアントおよびサーバ・エンドポイントで観測された遅延の詳細が表示されます。

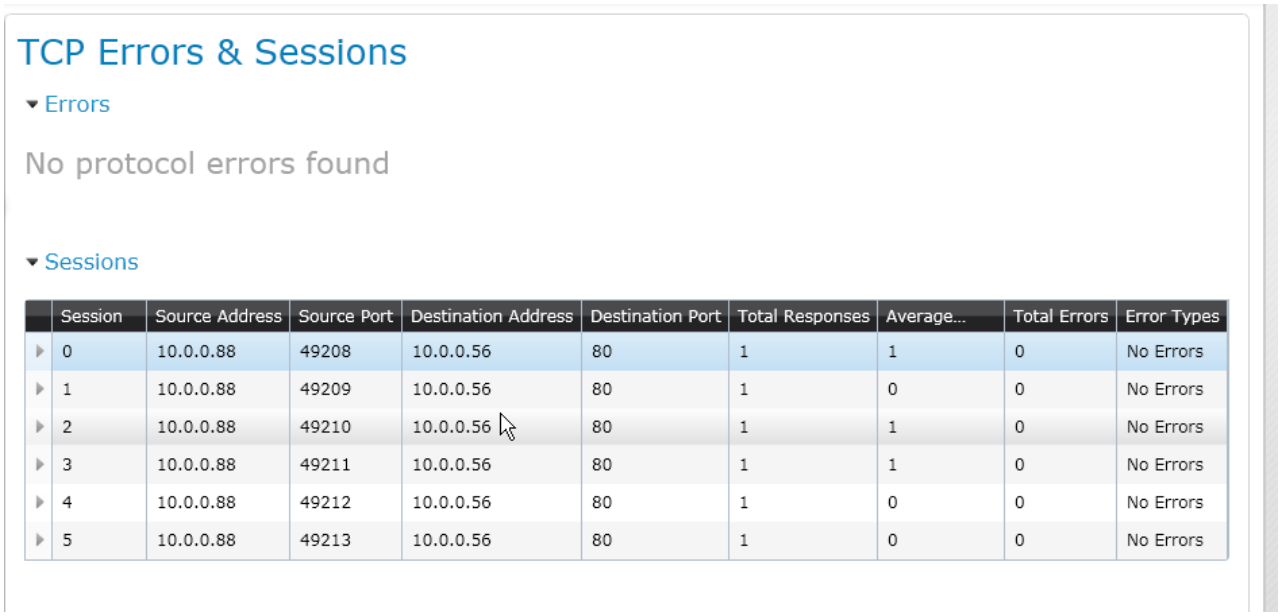
- **Source IP（ソース IP）**
- **Destination IP（宛先 IP）**
- **Names（名前）**：サーバの名前
- **Best estimate (ms)（最良推定値（ミリ秒））**：クライアントとサーバの間の遅延の最良推定値。これはクライアントとサーバの間の TCP 接続から導かれ，パケット・キャプチャで見つかった帯域幅の制約による追加の遅延を考慮しています（このため，この値は最小値より小さい可能性があります）。

- **Min (ms)（最小値（ミリ秒））**：パケット・キャプチャで観測された最小値。
- **95th percentile (ms)（95 パーセンタイル（ミリ秒））**：パケット・キャプチャで観測された最大値から再外部の条件を除いた値
- **Max (ms)（最大値（ミリ秒））**：パケット・キャプチャで観測された最大値。
- **Samples（サンプル数）**：遅延の計算に使用されたパケットの数。



TCP/UDP エラーおよびセッション

［**TCP errors & sessions（TCP エラーおよびセッション）**］を選択すると，TCP プロトコルを使用したトランザクションの詳細が表示されます。［**UDP errors & sessions（UDP エラーおよびセッション）**］を選択すると，UDP プロトコルを使用したトランザクションの詳細が表示されます。

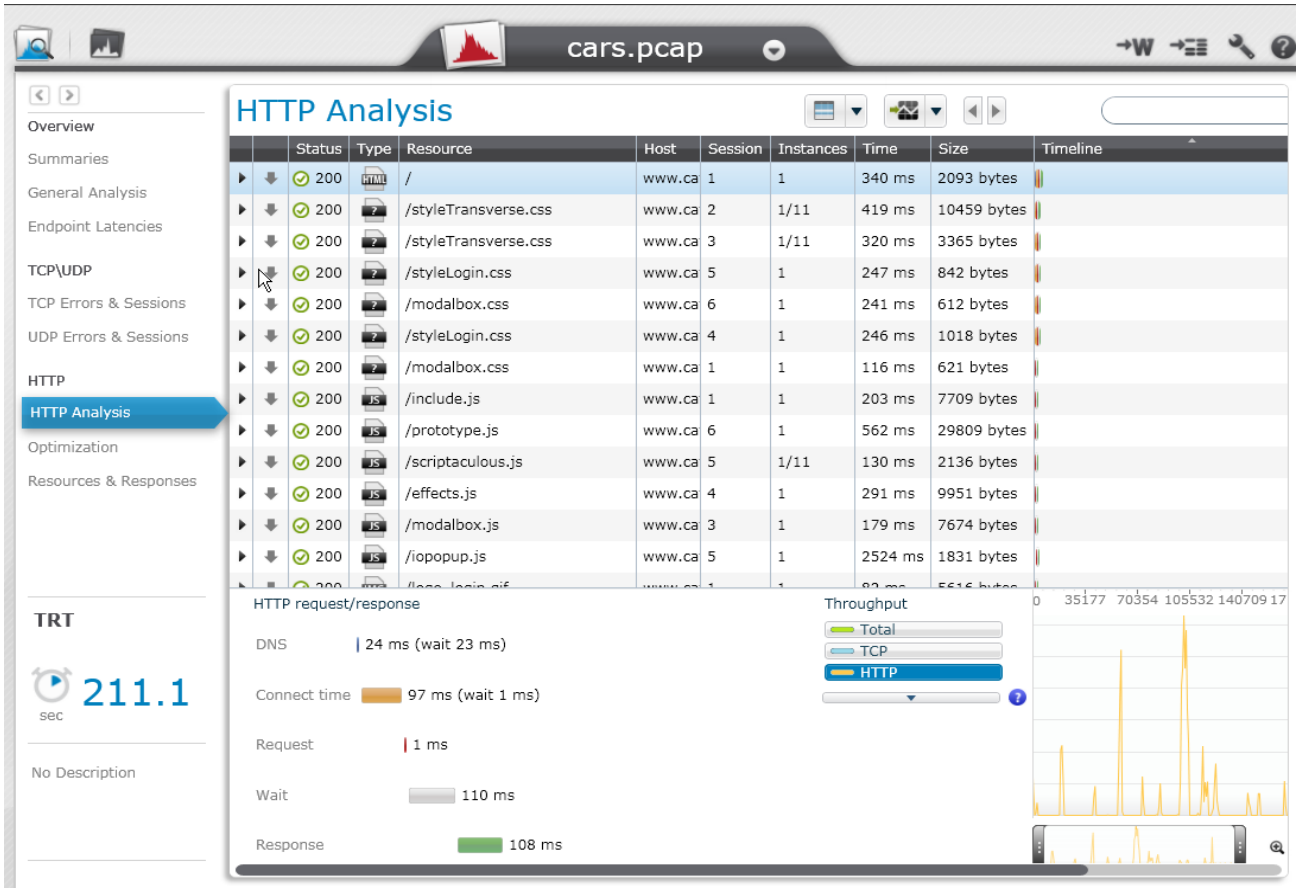


TCP と UDP のセッションの詳細は別々に表示され，それぞれに以下の項目が含まれます。

- **Session（セッション）**
- **Source Address（ソース・アドレス）**
- **Source Port（ソース・ポート）**
- **Destination Address（宛先アドレス）**
- **Destination Port（宛先ポート）**
- **Total Responses（合計応答数）**
- **Average Response Time（平均応答時間）**
- **Total Errors（合計エラー数）**
- **Error Types（エラー・タイプ）**

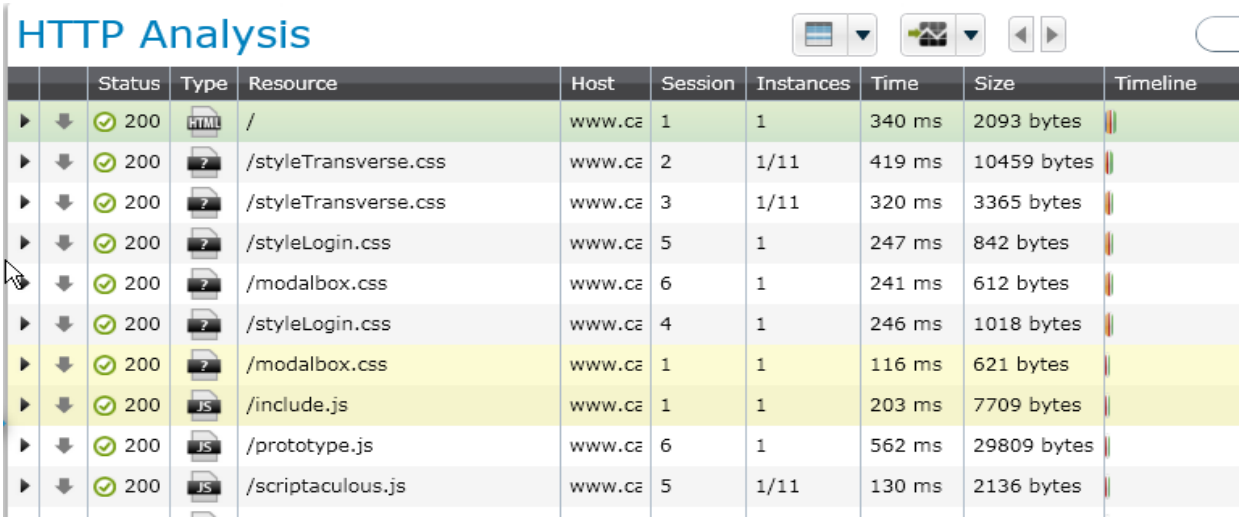
HTTP 分析

トランザクションの詳細は，表とグラフの両方の形式で表示されます。



一般分析レポートと HTTP 分析レポートのツールバーで，次のオプションが利用可能です。最初の 2 つのオプションは，表の右クリックメニューでも利用可能です。

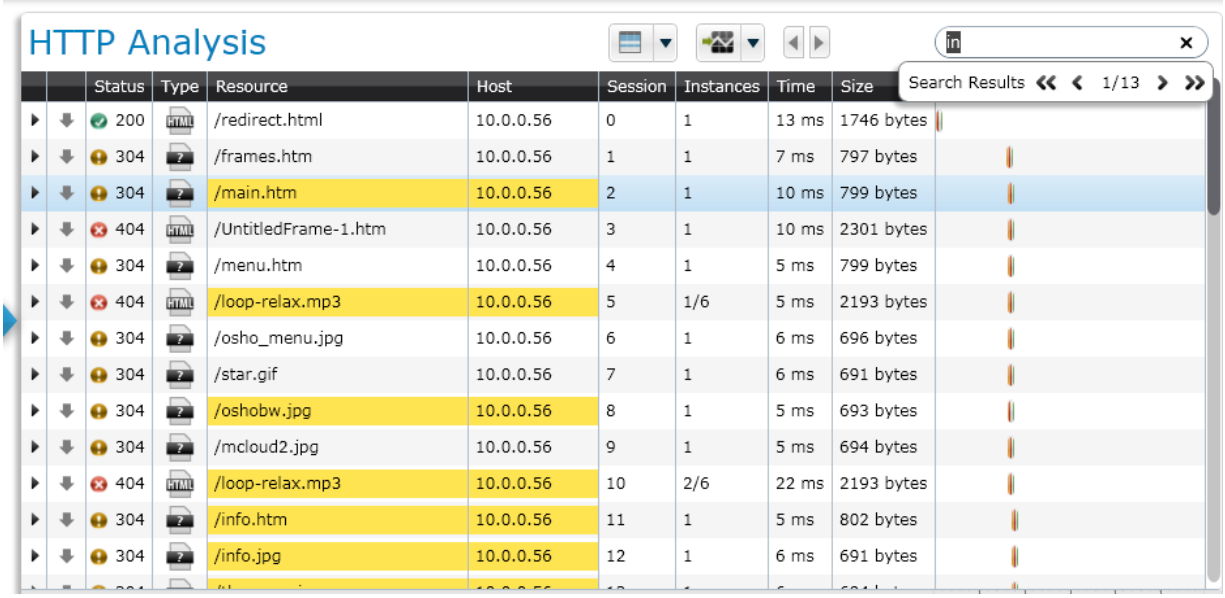
- ハイライト・オプション： をクリックして，同じソース IP などに基づいてリソースをハイライトするオプションを選択します。この例では，セッションのすべての参加者を表示しています。



- グラフに表示オプション：  をクリックして，セッション，サービスなどをグラフに表示します。

類似したリソースを URL に基づいて検索するには：

1. ツールバーの検索領域に，必要な文字列（この例では "in"）を入力します。



		Status	Type	Resource	Host	Session	Instances	Time	Size
▶	↓	200	HTML	/redirect.html	10.0.0.56	0	1	13 ms	1746 bytes
▶	↓	304	?	/frames.htm	10.0.0.56	1	1	7 ms	797 bytes
▶	↓	304	?	/main.htm	10.0.0.56	2	1	10 ms	799 bytes
▶	↓	404	HTML	/UntitledFrame-1.htm	10.0.0.56	3	1	10 ms	2301 bytes
▶	↓	304	?	/menu.htm	10.0.0.56	4	1	5 ms	799 bytes
▶	↓	404	HTML	/loop-relax.mp3	10.0.0.56	5	1/6	5 ms	2193 bytes
▶	↓	304	?	/osho_menu.jpg	10.0.0.56	6	1	6 ms	696 bytes
▶	↓	304	?	/star.gif	10.0.0.56	7	1	6 ms	691 bytes
▶	↓	304	?	/oshobw.jpg	10.0.0.56	8	1	5 ms	693 bytes
▶	↓	304	?	/mcloud2.jpg	10.0.0.56	9	1	5 ms	694 bytes
▶	↓	404	HTML	/loop-relax.mp3	10.0.0.56	10	2/6	22 ms	2193 bytes
▶	↓	304	?	/info.htm	10.0.0.56	11	1	5 ms	802 bytes
▶	↓	304	?	/info.jpg	10.0.0.56	12	1	6 ms	691 bytes

一致するすべてのリソース（この例では "in" を含む文字列）がハイライトされます。検索領域の下の矢印を使用すると，結果の間を移動できます。

2. 検索結果をクリアするには，検索領域を閉じます（"X" をクリック）。

サブトランザクションの詳細

要求/応答の表示を調整するには：

- 各列の昇順または降順に行をソートします。
- 領域の境界をドラッグして，領域を拡大または縮小します。

レポートには次の列が表示されます。

- **展開/折りたたみアイコン**  **：**各要求の詳細を表示または非表示にします。
- **上/下矢印：**上矢印は POST または PUT，下矢印は GET，頭の印はアイコン・ヘッダ，N/A は HTTPS，星印はその他すべてのタイプを示します。
- **Status（ステータス）：**HTTP ステータス。404（ページが見つかりません），200（OK）など。
- **Type（タイプ）：**アイコンは要求されたファイルのタイプ（グラフィック・ファイルなど）を示します。
- **Resource（リソース）：**サブトランザクションでアクセスされたパスを示します。
- **Host（ホスト）：**リソースのアップロードまたはダウンロード元のホスト（ドメイン，サーバなど）。

- **Session（セッション）：**リソースがアップロードまたはダウンロードされた TCP セッションの数。
- **Instances（インスタンス）：**同じ名前のリソースがトランザクションに出現する回数。
- **Time（時間）：**要求の応答時間（ミリ秒）。
- **Size（サイズ）：**要求/応答に使用されたバイト数。
- **Timeline（タイムライン）：**トランザクション・シーケンス内でのリソースの位置。

トランザクションごとに次のデータが（左側に）表示されます。

- **Network Time（ネットワーク時間）：**トランザクションの完了までの時間，すなわち要求の最初のパケットからトランザクションの最後のパケットまでの時間。
- **Duration（期間）：**トランザクションの開始から終了までの合計時間。
- **Description（説明）：**トランザクションの説明を表示します。

グラフの表示をカスタマイズするには：

- 右下のグラフには，スループットが表示されます。「ハンドル」を使用して，要求/応答内の特定の期間に注目できます。
- ［HTTP Throughput（HTTP スループット）］を選択すると HTTP データだけが表示され，［Total Throughput（合計スループット）］を選択すると要求/応答のすべてのデータが表示されます。
- この領域の上にマウスを移動すると，要求/応答領域とグラフ・ビューを結ぶ線がマウスとともに移動し，キャプチャ中にこれが発生した時間（秒）を示します。ズームアイコンを使用して，ズーム・イン，ズーム・アウト，またはズームなしを選択できます 。

HTTP パラメータ

リソースがハイライトされている場合，下の領域の表に次のブレークダウンが表示されます。

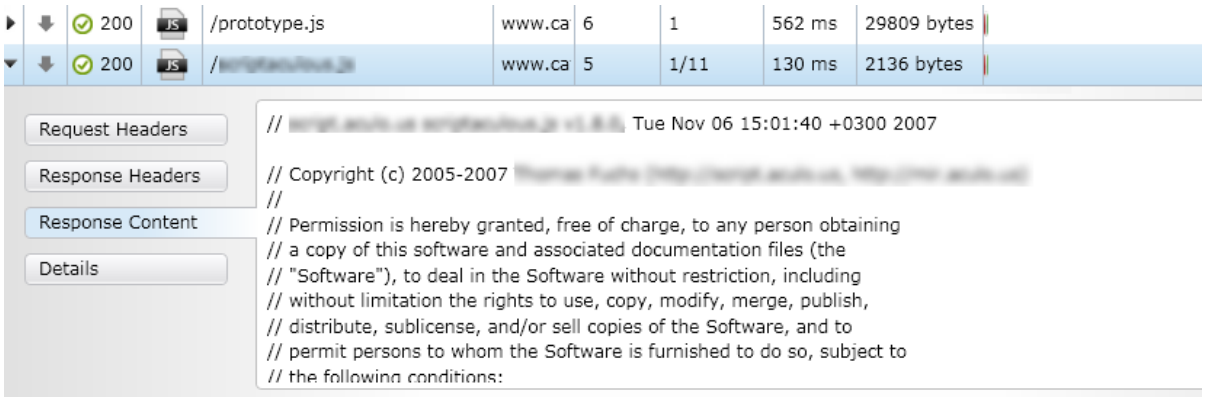
- **DNS Resolution（DNS 解決）：**解決時間から接続の開始までの待ち時間が含まれます（DNS クエリから，名前の問い合わせが行われたサーバへの最初の SYN までの時間）。
- **Connect Time（接続時間）：**TCP セットアップ時間と，接続の初期化から最初の要求データ・パケットの送信までの待ち時間。
- **TLS Time（TLS 時間）：**SSL/TLS セキュア・チャネルの確立。
- **Request（要求）：**クライアントがサーバに要求を送信するのに必要な時間。
- **Wait（待ち）：**要求の最後のパケットから応答の最初のパケットまでの時間。
- **Response（応答）：**応答の最初のパケットから最後のパケットまでの時間。
- **Encrypted Data Transmission（暗号化データ伝送）：**暗号化された HTTPS セッションの時間。

要求/応答の詳細を表示するには：

要求/応答名をダブルクリックするか，展開/折りたたみアイコンをクリックします。次の列が表示されます。

- **Request Headers（要求ヘッダ）：**要求ヘッダの構文。
- **Response Headers（応答ヘッダ）：**応答ヘッダの構文。

- **Request Content/Response Content（要求内容/応答内容）：**画像，HTML データなどを表示します（印刷不可能な文字は " ." で置き換えられることがあります。PDF，オーディオ，ビデオ，Flash，フォントなど，一部の形式の内容は表示されません）。
- **Details（詳細）：**要求/応答のスループット。HTTP POST の場合，完全な URL も表示されます。



注: リソースを選択（ハイライト）すると，リソースの詳細が下に表示されます。

HTTP 最適化

NV Analytics では，外部ソースから得られたデータと，HP 内部で現在のアプリケーション・テスト方法から得られたナレッジに基づいて，いくつかのベスト・プラクティス推奨事項を提供しています。

各トランザクションには，ベスト・プラクティスとの比較に基づくスコアが付けられます。合計スコア（100 点満点）は個別スコアのサマリであり，Web サイトのプログラミング・ルールに対する準拠のレベルに基づきます。

個別数値スコア：

優先度は，どのトランザクションが結果に最も影響するかを示します。それぞれの推奨事項は，それを実施した場合に得られるパフォーマンス向上の可能性に基づいて重み付けされ，負のポイント値で示されます。たとえば，個別スコアが -8 ポイントの場合は，該当する推奨事項の不遵守が，スコアが -2 ポイントの別の推奨事項の不遵守よりも，トランザクションの応答時間により大きな影響を与えることを示します。

文字グレード：

これに加えて，ベスト・プラクティスの遵守レベルを示す A ~ F の値が示されます。F が最も低い遵守レベルを表します。

例：

下の例では，いくつかのベストプラクティスに関してグレード F が付けられていますが，それぞれの数値スコアは異なります。その意味は，これらすべての推奨事項に関してトランザクションは低い遵守レベルを示していますが，そのうちいくつかは問題がより大きく，トランザクション結果により重

大な影響を与えるということです。「Reduce the size of your images（画像のサイズを小さくする）」というベスト・プラクティス推奨事項に関しては，画像は大きいもののその数はそれほど多くないのかもしれません。この場合，グレードがF でスコアが -8 の推奨事項を実施する方が，グレードがF でスコアが -4 の推奨事項を実施するよりも利益が大きいと言えます。

特定の推奨事項があるトランザクションのリストを表示するには：

1. ［**Optimization（最適化）**］を選択し，レポートに含めるルールを選択します。標準設定ではすべてのルールが選択されています。レポートは，異なるルールを使用して何度でも表示できます。
2. ［**Desktop（デスクトップ）**］または［**Mobile（モバイル）**］を選択します。また，結果をソートする基準を［**Priority（優先度）**］（各結果に付けられたポイント数の順）または［**Name（名前）**］（アルファベット順）から選択できます。
3. 追加の詳細を表示するには，推奨事項をクリックします。たとえば，「Don’t download the same data twice（同じデータを 2 回ダウンロードしない）」という推奨事項に関しては，ファイルが出現した回数が詳細に表示されます。それぞれをクリックして，HTTP 分析に表示できます。

> **注:** デスクトップとモバイルの結果では，スコアと推奨事項が異なる場合があります。これは，プラットフォームごとの最適化の違いによります。

モバイルとデスクトップの両方に関してルールのリストをフィルタするには：

1. 右上の編集（鉛筆）アイコン をクリックします。
2. ルールを選択するか選択解除し，［**Save（保存）**］をクリックします。一部のルールだけが選択されている場合，編集アイコンの隣に情報アイコンが表示されます 。

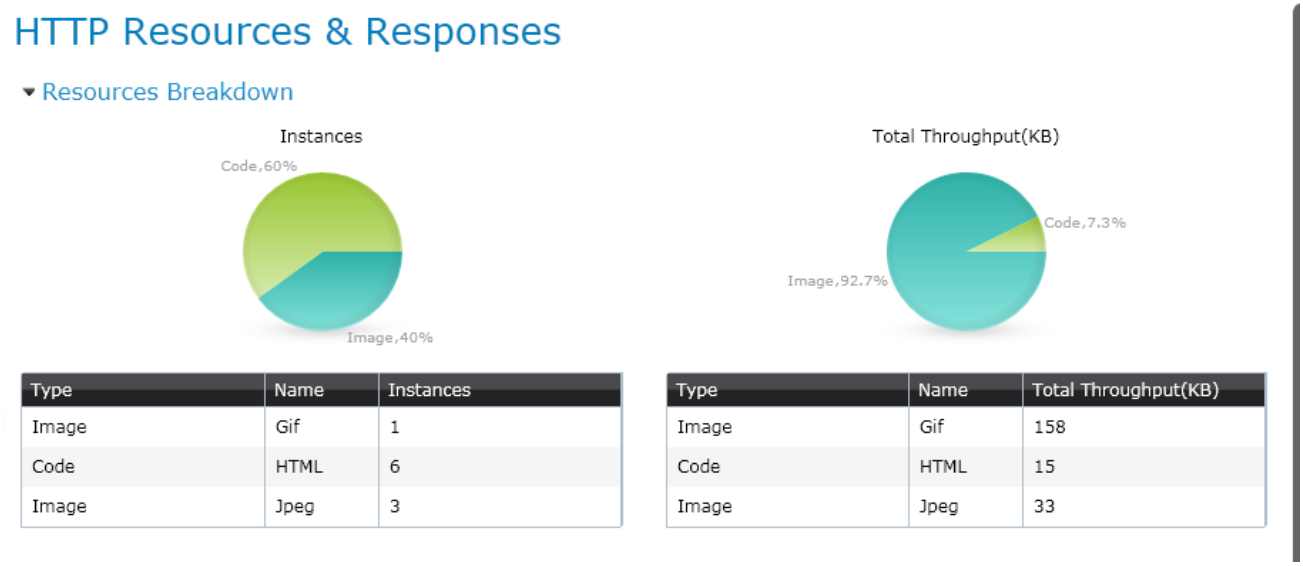
HTTP リソースおよび応答

リソース・ブレークダウンは，トランザクションに存在した各タイプのリソースに関して，インスタンス数と合計スループットを表示します。

リソース・ブレークダウン

［Instances（インスタンス）］円グラフは，各タイプのリソースがトランザクションに出現した回数を示します。グラフはリソースのタイプによって分割されており，たとえば .jpeg 画像が 1 つのカテゴリで，css ファイルが別のカテゴリです。

［Total Throughput（合計スループット）］グラフは，インスタンス・グラフの同じカテゴリに基づいて，合計スループット（KB）をサイズとするブレークダウンを示します。

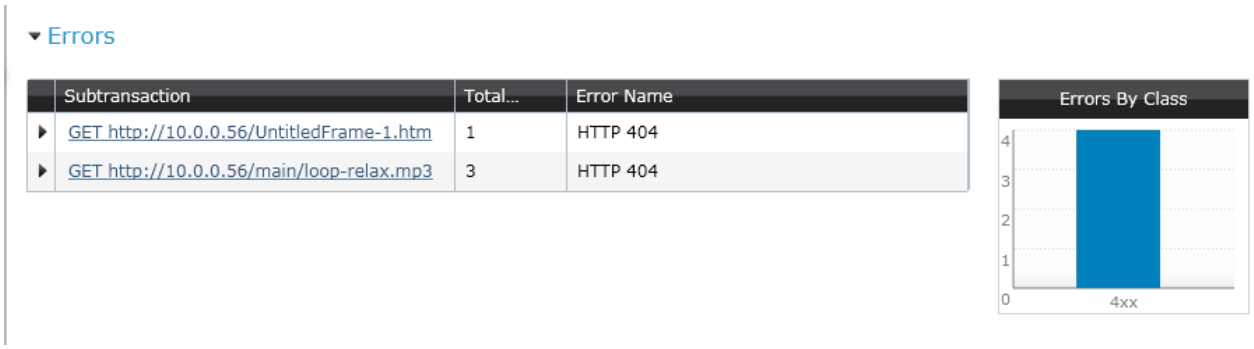


Type	Name	Instances
Image	Gif	1
Code	HTML	6
Image	Jpeg	3

Type	Name	Total Throughput(KB)
Image	Gif	158
Code	HTML	15
Image	Jpeg	33

HTTP エラー

この表には，各エラーが，発生したサブトランザクションに基づいて表示されます。［Errors by Class（クラス別エラー）］グラフには，クラス（HTTP クライアント・エラーなら 4xx，サーバ・エラーなら 5xx など）に基づく合計が表示されます。

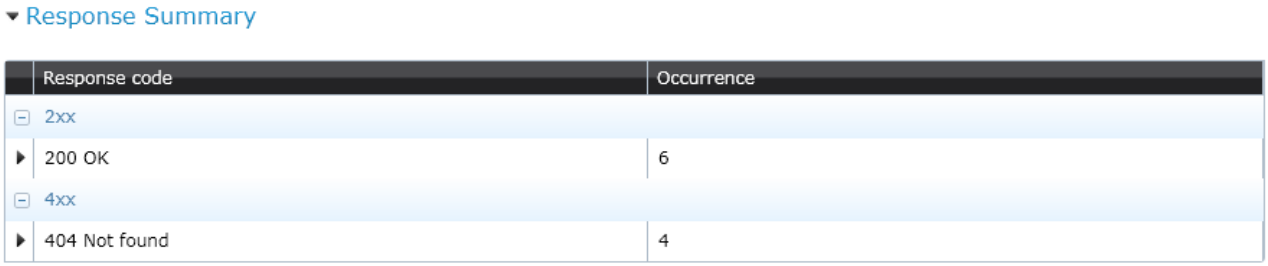


HLS エラー

この表には，ビデオ・ストリーミング中に発生したエラーのサマリが表示されます。

応答サマリ

各 HTTP 応答コードの発生回数を表示します。



セキュア通信

NV Analytics レポートでは，HTTPS セキュア通信に関して，ホスト名，TCP セッション確立，SSL セッション確立のデータを表示することができます。

秘密キーは，次の例のように，Privacy Enhanced Mail（PEM）形式のテキスト・ファイルである必要があります。

```
-----BEGIN RSA PRIVATE KEY-----
MIICXAIBAAKBgQC20yQBpiE1atCfluiOqa9fnPyHrCsYCoCJ/hKlS8z6IOcmBsNq
kZckzlMqcQP7i9o3SGVBXv1rAsGN3SnkqNUD4PFukUHQrjPpc0KPX9KdDCcLFu5f
Bxq+7/7hDzQA+rKWqixO3WmBVcKQm+3WvpF5M+jJuIsRY8O6rOxb+FrpvwIDAQAB
AoGBAKWpszHPUL4vmPTqU+KZ/bDc9rMVnkL9mTXxRQIFKqroT6vUaxTQ8i1GzHNj
zyELu+NmNWJD6cwixjJ/fap3HJNWMF0byZEwPyC5yKkEZQDKt3n549nTPxM
wDO9geHjqwJBANZu4lxS/+4WkqwN5yzh8VCKmMUc0PQvXw+niXrjSucc5k5VdHw4
qOobXalp1PyrafHj6YV8PfX9XHpPRAzs/j0CQBq8iJqHttBpzc+ObuCthtPR7XHt
5BTuuG3rPFoW+R/D8K2apQSoj2uEgxSFLcvpcaninPHEo0bO8SfqLqCmZxkCQADb
dKA13LwQ7wktDQ2K4bIWu8Gd+d/gCJtBajVJj1UZMnqBsPOGLnaxIVC6EZXpAYVs
CdT0yDKhjqsWggkjMWkCQHZvP1E28M51k1pLsQx43nq7zbueKZWkDg/biA3yOaLb
FJ9TSJeufAXAmG/US+zCfgLuzrSuJwHiCMnhRrBOm+Y=
-----END RSA PRIVATE KEY-----
```

秘密キーがこの形式でない場合は，Open SSL を使用して形式を変更します。サーバのオペレーティング・システムを判定します。Microsoft ® Windows の場合，キーは PKCS7/DER 形式で（ローカルに）保存されているか，.NET 形式で（任意のディレクトリ・サーバに）保存されているのが普通です。変換するには，次のコマンドを使用します。

```
# for PKCS7/DER keys (as held on disk)
openssl pkcs8 -nocrypt -in derfile.key -inform DER -out key.pem -outform PEM
# for NET keys (from the directory server)
openssl pkcs8 -nocrypt -in file.ick -inform NET -out key.pem -outform PEM
```

Mac OSX，Solaris，およびその他のシステムの場合，ファイル形式として PKCS#12 が多く用いられます。変換するには，次のコマンドを使用します。

```
openssl pkcs12 -nodes -in file.p12 -out key.pem -nocerts -nodes
```

Linux では，次のコマンドを使用します。

```
openssl x509 -nocrypt -in foo.der -informat DER -out key.pem -outformat PEM
openssl x509 -nocrypt -in foo.net -informat NET -out key.pem -outformat PEM
```

注: セキュア通信を分析するには，「HTTPS キー」(17ページ)の SSL オプションを参照してください。

第4章: NV Analytics API

NV Analytics API は，Representation State Transfer（REST）Web サービス・アーキテクチャを使用します。分析 API 要求はすべて同じ URL 構造を持ち，プレフィクスは次のとおりです。

[base address]/shunra/api/analysis

注: Python のコード例 "analyzer.py" がインストール・フォルダに用意されています。更新は https://gist.github.com/2773832 にあります。

これを使用して API にアクセスできます。このドキュメントにもいくつかのコード・セグメントが記載されています。

次のメソッドが公開されています。

分析エンジン

インストールされている分析エンジンの JSON コンテンツ・リストを返します。

GET

[base address]/shunra/api/analysis/engines

例： http://localhost:8182/shunra/api/analysis/engines

応答

応答には，すべての分析エンジンの ID と名前が含まれます。


```
{
"supportedAnalysisEngines":[{"name":"harExport","id":"harExport"},
{"name":"networkmeasurements","id":"networkmeasurements"},
{"name":"generalWaterfall","id":"generalWaterfall"},{"name":"http","id":"http"},
{"name":"iostats","id":"iostats"},{"name":"metrics","id":"metrics"},
{"name":"best practices","id":"best practices"}]}
}
```

戻り値

- 200 “OK”
- 404 “Not Found”
- 500 “Internal Server Error”

コード例

```
def get_engine_id(engine_name):
"""
  分析エンジンの名前を受け取ってその ID を返します。
  分析 API の動作テストとしても利用できます。


  >>> get_engine_id('best practices')
  u'best practices'
  """
  resp = get('/shunra/api/analysis/engines')
  engines = dict([(entry['name'], entry['id']) for entry in resp['supportedAnalysisEngines']])
  return engines[engine_name]
```

パケット・リストの抽出

パケット・リスト名，ID，エンドポイント，.pcap および .ved ファイルの固有 ID の JSON コンテンツを返します。

PUT

[base address]/shunra/api/analysis/packetlistmetadata

例： http://localhost:8182/shunra/api/analysis/packetlistmetadata

本文

JSON は分析されるエミュレーション結果（.ved または .pcap ファイル）の ID を定義します。これは NV Analytics のファイル・システム・パスです。

```
{
"id":"C:\\tmp\\Sample.ved"
 }
```

応答には，分析される実行結果の ID と，パケット・リストのメタデータ（名前，ID，エンドポイント）が含まれます。

```
{
    "packetLists":[{
        "endpoints":[{
            "name":"Tokyo Office",
            "id":"6d0652db88c349de9382a54dc350349f"
        }],
        "name":"Packet List 3",
        "id":"c6064d9bf25d405382e374795fef35fe"
    },
    {
        "endpoints":[{
            "name":"London Office",
            "id":"de358779547c4eea8caeef62bfbbb493"
        }],
        "name":"Packet List 2",
        "id":"59220e1cb4d248eba3b89a695918be91"
    },
    {
        "endpoints":[{
            "name":"NY Office",
            "id":"8c95498f7bb04c7598dde1d5e609082a"
        }],
        "name":"Packet List 1",
        "id":"620984c9a31b4ef694a1ac47d61b6a7e"
    }],
    "runResultId":"b80de7f5ffa97428b2324c8b3a9d469b"
}
```


戻り値

- 200 “OK”
- 404 “Not Found”
- 500 “Internal Server Error”

コード例

```
def get_packetlists(inputfilepath):
"""
    指定されたファイル内の利用可能なパケット・リストの辞書を返します。
    辞書のキーはパケット・リスト名，辞書の値はパケット・リスト ID です。

    >>> packetlists = get_packetlists(os.path.join(SAMPLE_FOLDER, 'Sample.ved'))
    >>> len(packetlists)
    3
    >>> 'Packet List 1' in packetlists
    True
    """
    resp = put('/shunra/api/analysis/packetlistmetadata', {'id':inputfilepath})
```

```
return dict([(entry['name'], entry['id']) for entry in resp['packetLists']])

def get_run_result_id(inputfilepath):
resp = put('/shunra/api/analysis/packetlistmetadata', {'id':inputfilepath})
return resp['runResultId']
```

分析要求

JSON で表され，パケット・リストごと，トランザクションごと，分析エンジンごとの分析プロセスの現在のステータスの内容を返します。

応答は，次のエントリから成る辞書です。

- **transactionAnalysisStatus** – 以下に示す内容のトランザクションのリスト
- **reportId** – 分析プロセスの ID
- **name** – 分析対象のパケット・リスト名
- **id** – 分析対象のパケット・リスト ID

トランザクションのリストには，パケット・リストに関連する各トランザクションのエントリが含まれます。

各エントリは辞書であり，トランザクション ID（id），名前（name），分析ステータス（analysisStatusPerEngine）を含みます。

分析ステータスは辞書であり，キーは分析エンジン，値はそのステータス（API ドキュメントに記述）です。

PUT

[base address]/shunra/api/analysis/request/{plid}

例： http://localhost:8182/shunra/api/analysis/request/620984c9a31b4ef694a1ac47d61b6a7e

ここで，"plid" はパケット・リスト固有の ID であり，パケット・リストの抽出要求から返されたものです。

本文

ポート，SSL 暗号化キー，分析されたエミュレーション結果（.ved または .pcap ファイル）の ID といった分析パラメータと，ファイル・システムのパスが含まれます。これは，ファイルがシステムによって永続化（保持?）されないからです。本文は JSON 形式です。

```
{
"ports":"80, 8080",
"sslEncryptionKey":"172.30.2.31,443,http,C:\\keys\\secret.key",
"runResultHandle":"C:\\tmp\\Sample.ved"
}
```

応答には，トランザクションごと，インストールされている分析エンジンごとの現在の分析ステータスと，分析パラメータを表す生成された分析レポート ID が含まれます。


```
{
    "transactionAnalysisStatus":[{
        "analysisStatusPerEngine":{
            "networkmeasurements":"Started",
            "harExport":"Started",
            "generalWaterfall":"Started",
            "http":"Started",
            "iostats":"Started",
            "metrics":"Started",
            "best practices":"Started"
        },
        "name":"Undefined",
        "id":"ccb8713e522241c9a691c4ed1ce72d27"
    }],
    "reportId":"-561678026",
    "name":"Packet List 1",
    "id":"620984c9a31b4ef694a1ac47d61b6a7e"
}
```


可能な分析ステータスは次のとおりです。

```
public enum WorkStatus {
// ジョブはまだ開始，実行，分析されていない
Idle(0),
// ジョブ（エミュレーションや分析など）は開始された
Started(1),
// ジョブ（エミュレーションや分析など）は完了した
Finished(2),
// ジョブ（分析など）は失敗した
Failed(3);
}
```

注: 分析プロセスが完了しているかどうかを判断するには，すべての項目のステータスが Finished または Failed になっているかどうかを見ます。そうでない場合は，分析ジョブ・プール内のいくつかの項目がまだ完了していません。

クライアント側では，分析プロセスが完了するまで，分析要求の処理を継続する必要があります。

戻り値

- 200 “OK”
- 404 “Not Found”

- 500 “Internal Server Error”

コード例

```
def analyze(inputfilepath, packetlist_id, settings={}):
    """
    指定したファイルに対して分析を呼び出します（ポート番号や SSL キーなどの特別な分析パラメータを渡すには設定を使用します）

    packetlist_id は特定のパケット・リストに対して get_packetlists から返された ID である必要があります



    応答は，次のエントリから成る辞書です。
        * transactionAnalysisStatus – 以下に示す内容のトランザクションのリスト
        * reportId – 分析プロセスの ID
        * name – 分析対象のパケット・リスト名
        * id – 分析対象のパケット・リスト ID

    トランザクションのリストには，パケット・リストに関連する各トランザクションのエントリが含まれます。
    各エントリは辞書であり，トランザクション ID（id），名前（name），分析ステータス（analysisStatusPerEngine）を含みます。
    分析ステータスは辞書であり，キーは分析エンジン，値はそのステータス（API ドキュメントに記述）です。

    >>> inputfilepath = os.path.join(SAMPLE_FOLDER, 'Sample.ved')
    >>> packetlists = get_packetlists(inputfilepath)
    >>> packetlist_id = packetlists['Packet List 1']
    >>> resp = start_analysis(inputfilepath, packetlist_id)['transactionAnalysisStatus']
    >>> len(resp) # only one tranasction is associated with this packet list
    1
    >> resp[0]['name']
    u'Undefined'
    >>> resp[0]['analysisStatusPerEngine']['http'] in ['Idle', 'Started', 'Finished', 'Failed']
    True
    """
    params = dict(settings)
    params['runResultHandle'] = inputfilepath
    resp = put('/shunra/api/analysis/request/'+packetlist_id, params)
    return resp
    def get_report_id(inputfilepath, packetlist_id, settings={}):
    return analyze(inputfilepath, packetlist_id, settings)['reportId']

    def get_transactions(inputfilepath, packetlist_id, settings={}):
"""
    指定されたパケット・リストに関連するすべてのトランザクションを取得します。
    結果はペアのリストであり，各ペアの最初の要素はトランザクション ID，2 番目の要素はトランザクション名です
```

```
    >>> inputfilepath = os.path.join(SAMPLE_FOLDER, 'Sample.ved')
    >>> packetlists = get_packetlists(inputfilepath)
    >>> packetlist_id = packetlists['Packet List 1']
    >>> result = get_transactions(inputfilepath, packetlist_id)
    >>> len(result) # only one transaction is associated with this packet list
    1
    >>> result[0][1]
    u'Undefined'
    """
    return [(transaction['id'], transaction['name']) for transaction in analyze(inputfilepath, packetlist_id, settings)
['transactionAnalysisStatus']]


    def start_analysis(inputfilepath, packetlist_id, settings={}):
"""
指定したファイルの分析を開始します。

    応答はリストであり，パケット・リストに関連する各トランザクションのエントリが含まれます。
    各エントリは辞書であり，トランザクション ID（id），名前（name），分析ステータス
（analysisStatusPerEngine）を含みます。
    分析ステータスは辞書であり，キーは分析エンジン，値はそのステータス（API ドキュメントに記
述）です。
    """
    return analyze(inputfilepath, packetlist_id, settings)

def is_analysis_done(inputfilepath, packetlist_id, settings={}):
    """
    指定したパケット・リストに関連するすべてのトランザクションが分析され，そのレポートが取得
可能な状態である場合に True を返します。
    """
    resp = analyze(inputfilepath, packetlist_id, settings)['transactionAnalysisStatus']
    for transaction in resp:
    for engine_status in transaction['analysisStatusPerEngine'].values():
if engine_status in ['Idle','Started']:
return False
return True
```

分析サマリ

JSON で表され，パケット・リストごと，トランザクションごと，分析エンジンごとの分析サマリの内容を返します。

GET

[base address]/shunra/api/analysis/summary/{runresulthandle}/{plid}/{reportId}/{engineId}

例：
http://localhost:8182/shunra/api/analysis/summary/b80de7f5ffa97428b2324c8b3a9d469b/620984c9a31b4ef694a1ac47d61b6a7e/-561678026/best%20practice

GET

[base address]/shunra/api/analysis/summary/{runresulthandle}/{plid}/{reportId}/{trId}/{engineId}

例：
http://localhost:8182/shunra/api/analysis/summary/b80de7f5ffa97428b2324c8b3a9d469b/620984c9a31b4ef694a1ac47d61b6a7e/-561678026/best%20practices

最初の呼び出しは，パケット・リスト内のすべてのトランザクションに関して，要求された分析レポートを返します。2 番目は，指定されたトランザクションのみに関するレポートを返します。

現在サポートされているレポート・タイプは次のとおりです。

- **http：**HTTP 分析
- **best practices：**最適化レポート
- **iostats：**スループット・レポート
- **general/waterfall：**一般分析
- **metrics：**プロトコルのサマリおよびメトリック・レポート
- **networkmeasurements：**エンドポイント遅延レポート
- **harExport：**HTTP サブトランザクションを HAR 形式で示すレポート（試験的）

応答

GET

’ /shunra/api/analysis/summary/%s/%s/%s/%s/%s'%(run_result_handle, packetlist_id, report_id, transaction_id, engine_id)

戻り値

resp['successfullTransactionAnalysis'][0]['result'] ???

- 200 “OK”
- 404 “Not Found”
- 500 “Internal Server Error”

コード例

```
def get_analysis_report(run_result_handle, packetlist_id, report_id, transaction_id, engine_id):
"""
指定したパケット・リストに対する分析エンジンの 1 つの実行結果を取得
"""

resp = get('/shunra/api/analysis/summary/%s/%s/%s/%s/%s'%(run_result_handle, packetlist_id, report_id, transaction_id, engine_id))
```

```
return resp['successfullTransactionAnalysis'][0]['result']
```

分析アーティファクト

トランザクション内に見つかった，画像，動画，文書，テキストなどのファイル。

ここには次の内容があります。

- 「分析レポートの構造」(46ページ)
- 「HTTP ウォーターフォール分析レポートの構造」(47ページ)
- 「ベスト・プラクティス分析レポートの構造」(50ページ)

GET

[base address]/shunra/api/analysis/artifact/{filehandle}

アーティファクト・ハンドルは分析レポートから得られます（「分析レポートの構造」(46ページ)を参照）。

例：
http://localhost:8182/shunra/api/analysis/artifact/620984c9a31b4ef694a1ac47d61b6a7e%2F-561678026%2Fccb8713e522241c9a691c4ed1ce72d27%2F94660f9c01724f63bedfefb370dc4575%2Fabf8bde63762421dbe29cab1cecae661

戻り値

- 200 “OK”
- 404 “Not Found”
- 500 “Internal Server Error”

分析レポートの構造

特定のトランザクションに対してステップ 4「分析結果の取得」を実行した結果として，API は次の形式の JSON ドキュメントを返します。

```
{
    "name":"Packet List 1",
    "successfullTransactionAnalysis":[{
        "status":"Finished",
        "result":{
            "type":"Best Practices Report",
            "subtype":"Web Applications Best Practices Report",
            "version":"0.5",
            -- Other, analysis engine dependent fields --
        },
```

```
            "name":"Undefined",
            "id":"fe15bdf3eafe4ec8bb1b055c49ca622b"
        }],
        "reportType":"best practices",
        "reportId":"129778102",
        "id":"1ae2d2ee02144e69801e0f3d1cb39d89",
        "failedTransactionAnalysis":[]
}
```


注: 青いテキストは実際のレポートを示します。要求は特定のトランザクション，特定のエンジンに関するものなので，"successfullTransactionAnalysis" の結果にはエントリが 1 つしか含まれません。これは，そのトランザクションのレポートにトランザクションの名前と ID を付加したものです。レポートはすべて共通の構造を持ち，"type"，"subtype"，"version" の各フィールドが含まれます。

HTTP ウォーターフォール分析レポートの構造

標準的な HTTP ウォーターフォール・レポートでは，"subTransactions" リストに多くのエントリが含まれています。これらはそれぞれ次の 2 つのタイプのいずれかです。

- **HTTP request/response：** HTTP セッションまたは復号化された HTTPS セッションから得られた 1 つの HTTP 要求
- **HTTPS session：**復号化されていない HTTPS セッションの詳細（赤でハイライト）

すべてのフィールドが必須というわけではありません。次に示すフィールドのうち，赤で記されたものだけが各エントリで利用できることが保証されています。たとえば，指定した要求に対する応答がパケット・リストでキャプチャされなかった場合，応答に関連するすべてのフィールドがエントリに出現しません。したがって，要求のタイムスタンプを示すコンポーネントだけが利用できることが保証されます（現時点では，レポートには要求がキャプチャされた HTTP 要求/応答ペアだけが含まれます）。

タイムスタンプは青で記され，1970 年 1 月 1 日からの秒数を表します。応答データのハンドルはオレンジで記されます。応答データ自体を取得するには，“Get Analysis Artifact” (5) API 呼び出しを使用します。


```
{
    "type":"Waterfall report",
    "subtype":"Http Waterfall report",
    "version":"0.80",
    "subTransactions":[{
        "type":"HTTP request/response",
        "start":1333054863953,
        "end":1333054864640,
        "recomendations":"",
        "attributes":{
            "RequestContentSize":0,
```

```
                "ResponseContentType":"application/json; charset\u003dUTF-8",
                "StatusCode":401,
                "TcpReset": false,
                "Method":"POST",
                "Scheme":"https",
                "ResponseContentSize":104,
                "RequestHeaders":"POST /setup/ws/1/validate HTTP/1.1\r\nHost: setup.example.com\r\ nAccept-Encoding:
gzip, deflate\r\nConnection: keep-alive\r\nProxy-Connection: keep-alive\r\n",
                "TcpSession":4,
                "RequestData":"",
                "RequestContentType":"text/plain",
                "URI":"/setup/ws/1/validate",
                "ResponseData":
                "\\74b85bbff75340a9b744bf8b4d1f5f6b\\-
1019702096\\5d35c39d1db84f3fa16786dc78eff622\\0703708ccbda491ba6d59944c1ef1114/78820e83d8
634265900999172f134389",
                "ResponseHeaders":"HTTP/1.1 401 Unauthorized\r\nDate:Thu, 29 Mar 2012 21:01:04 GMT
\r\nConnection:Keep-alive\r\n",
                "host":"setup.example.com",
                "Referer":"https://www.example.com/"
        },
        "components":[{
                "type":"DNSResolution",
                "start":1333054863853,
                "end":1333054863900
        },
        {
                "type":"TCPSetup",
                "start":1333054863900,
                "end":1333054863953
        },
        {
                "type":"ClientWaitAfterTCPSetup",
                "start":1333054863953,
                "end":1333054864418
        },
        {
                "type":"TLSHandshake",
                "start":1333054864418,
                "end":1333054864632
        },
        {
                "type":"request",
                "start":1333054864632,
                "end":1333054864632
        },
        {
                "type":"wait",
                "start":1333054864632,
```

```
                "end":1333054864640
            },
            {
                "type":"response",
                "start":1333054864640,
                "end":1333054864640
            }
        ]},

        {
            "type":"HTTPS session",
            "start":1333054861902,
            "end":1333054863281,
            "recomendations":"",
            "attributes":{
                "SentBytes":384,
                "ReceivedBytes":5792,
                "host":"www.example.com",
                "TcpReset": false,
                "TcpSession":0
            },
            "components":[{
                "type":"TCPSetup",
                "start":1333054861902,
                "end":1333054861902
            },
            {
                "type":"ClientWaitAfterTCPSetup",
                "start":1333054861902,
                "end":1333054862731
            },
            {
                "type":"TLSHandshake",
                "start":1333054862731,
                "end":1333054863230
            },
            {
                "type":"EncryptedDataTransmission",
                "start":1333054863230,
                "end":1333054863281
            }
        ]}
        --- OTHER HTTP and HTTPS ENTRIES --

    ]}
```

ベスト・プラクティス分析レポートの構造

ベスト・プラクティス・レポートは非常に単純です。レポートはエントリのリストであり，各エントリがベスト・プラクティスを表します。次の例では，2 つのベスト・プラクティスが青でハイライトされています。各ベスト・プラクティスには次のフィールドが含まれます。

- **名前**
- **説明**
- **List of applicable scenarios（適用可能なシナリオのリスト）**（現時点では DesktopWeb または MobileSafari）
- **Score（スコア）：**トランザクションが該当するベスト・プラクティスをどの程度満たすかを示す値（[0,1] の範囲内の数値）。
- **Weight（重み）：**トランザクションに対するベスト・プラクティスの影響の大きさを示す値（[0,1] の範囲内の数値）。
- **A dictionary of violations（違反の辞書）：**この辞書の各エントリは，ベストプラクティスに対する違反の特定のタイプと，その違反を犯しているリソース（または TCP セッション）のリストです。トランザクションには特定のベスト・プラクティスに対する違反がない場合もあります。次の例の "Compress Components" がそうです。


```
{
    "type":"Best Practices Report",
    "subtype":"Web Applications Best Practices Report",
    "version":"0.5",
    "report":[{
        "violations":{},
        "name":"Compress Components",
        "scenarios":[
            "DesktopWeb",
            "MobileSafari"
        ],
        "description":"Checks that textual elements are transferred in a compressed format.（テキスト要素が圧縮形式で転送されることを確認します）Compression usually reduces the response size by about 70%.（圧縮により，応答のサイズが通常 70% 程度減少します）Approximately 90% of current Internet traffic travels through browsers that claim to support gzip.（現在のインターネット・トラフィックの約 90% は，gzip をサポートすると表明しているブラウザを通過します）",
        "score":100.0,
        "weight":1.0
    },
    {
        "violations":{
            "An expiration header was not found":[
                "http://platform.example.com/widgets.js"
            ],
            "Expiration date is within the next two days":[
            "HTTP://media.example.com/media-proxy/picture1.jpg",
```

```
                "http://media.example.com/media-proxy/picture2.jpg",
                "http://media.example.com/media-proxy/picture3.jpg"
            ]},
            "name":"Add long term headers expiration dates",
            "scenarios":[
                "DesktopWeb",
                "MobileSafari"
            ],
            "description":"Near future headers expiration dates prevent effective caching.（ヘッダの有効期限が近い場合，キャッシュが有効に働きません）This results in a repeat visit to your site to be slower than necessary.（これにより，サイトの再表示の速度が不必要に遅くなります）",
            "score":65.0,
            "weight":0.8
        }]
}
```

第5章: NV Analytics プロトコル

ここでは，NV Analytics がレガシ・レポートの各プロトコルをどのように識別して処理するかを説明します。内容は次のとおりです。

サポートされているプロトコル

NV Analytics でサポートされ，分析されるプロトコルは次のとおりです。

レイヤ 2 ~ 3	Web
IP	HTTP
TCP	HTTPS
UDP	

会話の定義

会話の定義と識別は，実行する分析のタイプに応じて異なります。

定義（識別）は次の要素に基づきます。

- IP - IP アドレス・ペア（例：10.0.0.1 - 10.0.0.2）
- UDP - IP アドレスとポート番号のペア（例：10.0.0.1:6789 - 10.0.0.2:3456）
- TCP - IP アドレスとポート番号のペア（例：10.0.0.1:6789 - 10.0.0.2:3456）
- HTTP - URL（例：www.google.com/images）

会話統計の収集

NV Analytics は，会話インスタンス（たとえば，www.google.com という URL の 1 回の GET）ごとに統計を収集します。

表示されるメトリックは，次のグループに分けられます。

- すべてのアプリケーション
- アプリケーションごと
- アプリケーション会話ごと（サブ会話を含む）

TCP，UDP，IP の分類

会話が TCP に分類されるのは，それより上位レベルのプロトコルが存在しない場合です。この場合は TCP Other として識別されます。

会話が UDP に分類されるのは，それより上位レベルのプロトコルが存在しない場合です。この場合は UDP Other として識別されます。

会話が IP に分類されるのは，それより上位レベルのプロトコルが存在しない（TCP でも UDP でもない）場合です。この場合は IP Other として識別されます。

サブトランザクション・グループ

NV Analytics は，サブ会話を 1 つのフローにグループ化するので，会話全体のデータを取得することも，その中の各サブ会話のデータを取得することもできます。このグループ化がどのように行われるかは，NV Analytics のグループ化の設定によって異なります。

プロトコルの関連付けの理解

会話は，関連するアプリケーション・プロトコルに基づいて関連付けられます。たとえば，HTTP を観察している場合，基礎となる TCP/IP 通信および通信メトリックが，それによって構成される HTTP 会話に関連付けられます。HTTP Get 要求 - 応答会話中に TCP 再送信が見つかった場合，その TCP 再送信は HTTP 会話に関連付けられます。

受信したシーケンス番号が予期した値より小さい場合（再送信，高速再送信，セグメントの順序誤りのいずれか），次の条件が満たされれば NV Analytics は高速再送信であると判断します。

- このセグメントに対して 2 個以上の重複する ACK が見つかっている（すなわち 3 個以上の ACK）。

- このセグメントが次の ACK のないセグメントである。
- このセグメントが最後の重複する ACK から 20ms 以内に到着した（20ms は任意であり，再送信タイムアウトと混同するおそれがないくらいに小さい値であることが必要）。

フィードバックをお送りください

ユーザーズ・ガイド について何かお気づきのことがありませんか?

ご意見をお聞かせください。 SW-Doc@hp.com