PureFlow GSX

トラフィックシェーパー
NF7101C
コンフィギュレーションガイド

第9版

・製品を適切・安全にご使用いただくために，製品をご使用になる前に，本書を必ずお読みください。
・本書に記載以外の各種注意事項は，NF7101C トラフィックシェーパー取扱説明書（NF7101-W006J）に記載の事項に準じますので，そちらをお読みください。
・本書は製品とともに保管してください。

アンリツ株式会社

管理番号：NF7101-W008J-9.0

安全情報の表示について

当社では人身事故や財産の損害を避けるために，危険の程度に応じて下記のようなシグナルワードを用いて安全に関する情報を提供しています。記述内容を十分理解して機器を設置および操作するようにしてください。
下記の表示およびシンボルは，そのすべてが本器に使用されているとは限りません。また，外観図などが本書に含まれるとき，製品に貼り付けたラベルなどがその図に記入されていない場合があります。

本書中の表示について

⚠ 危険	回避しなければ，死亡または重傷に至る切迫した危険があることを示します。
⚠ 警告	回避しなければ，死亡または重傷に至る恐れがある潜在的な危険があることを示します。
⚠ 注意	回避しなければ，軽度または中程度の人体の傷害に至る恐れがある潜在的危険，または，物的損害の発生のみが予測されるような危険があることを示します。

機器に表示または本書に使用されるシンボルについて

機器の内部や操作箇所の近くに，または本書に，安全上および操作上の注意を喚起するための表示があります。
これらの表示に使用しているシンボルの意味についても十分理解して，注意に従ってください。

禁止行為を示します。丸の中や近くに禁止内容が描かれています。

守るべき義務的行為を示します。丸の中や近くに守るべき内容が描かれています。

警告や注意を喚起することを示します。三角の中や近くにその内容が描かれています。

注意すべきことを示します。四角の中にその内容が書かれています。

PureFlow GSX
トラフィックシェーパー　NF7101C
コンフィギュレーションガイド

2014年（平成26年）９月25日（初　版）
2021年（令和３年）10月25日（第９版）

当社へのお問い合わせ

本製品については，安全マニュアルに記載の「本製品についてのお問い合わせ窓口」へご連絡ください。

保守契約について

保守契約を結んでいただくと種々のサービスを受けることが可能です。保守契約の詳細については，ご購入いただいた販売店にお問い合わせください。

国外持出しに関する注意

本製品および添付マニュアル類は，輸出および日本国外持ち出しの際には，「外国為替及び外国貿易法」により，日本国政府の輸出許可や役務取引許可を必要とする場合があります。また，米国の「輸出管理規則」により，日本からの再輸出には米国政府の再輸出許可を必要とする場合があります。
本製品は日本国以外の安全規格などに準拠していない場合があります。
本製品や添付マニュアル類を輸出または日本国外持ち出しする場合は，事前に必ず弊社の営業担当までご連絡ください。
輸出規制を受ける製品やマニュアル類を廃棄処分する場合は，軍事用途等に不正使用されないように，破砕または裁断処理していただきますようお願い致します。

本書の内容

この取扱説明書は，PureFlow GSX トラフィックシェーパー NF7101C（以下，本装置）で動作するソフトウェアの設定方法と使用方法を説明します。本装置を設置，導入，管理を行うネットワーク管理者を対象としています。インターネットワーキングに対する以下のような基礎知識を持った読者を想定しています。

- ローカルエリアネットワーク（LAN）
- Ethernet
- インターネットプロトコル（IP）

本装置の取扱説明書は，以下の①～③で構成されています。本書は③です。

① **取扱説明書**（NF7101-W006J）
この説明書は，本装置の設置および取り扱いについて記述してあります。

② **コマンドリファレンス**（NF7101-W007J）
この説明書は，本装置で使用するコマンドの詳細について記述してあります。

③ **コンフィギュレーションガイド**（NF7101-W008J）
この説明書は，本装置の持つ基本的な機能およびその機能を使ってネットワークを構築する際の具体的な設定例について記述してあります。

また，本製品に関連する下記文書または機能に関する文書が発行された場合，必ずご一読ください。

リリースノート
（リリースノートの発行については，ご購入いただいた販売店にお問い合わせください）

目次

（空白ページ）

第1章　ソフトウェアの概要

ここでは，本装置ソフトウェアの概要について説明します。

基本機能を以下に列記します。

- トラフィックコントロール機能
- リンクダウン転送機能
- SSH 機能
- Simple Network Management Protocol（SNMP）機能
- 統計情報
- トップカウンタ機能
- WebAPI 機能
- RADIUS 機能
- ドメインフィルタ機能
- トラフィック分析機能

（空白ページ）

ここでは，本装置ソフトウェアの基本機能について説明します。

2.1　トラフィックコントロール機能

音声通信や TV 会議などのミッションクリティカルな業務は，回線帯域不足によるパケット消失や通信遅延が発生すると業務効率を低下させ，重大な支障をきたすことにつながります。このようなミッションクリティカルなトラフィックを回線帯域不足や通信遅延から守るために，回線帯域を拠点やユーザ，またはアプリケーションごとに分割し，必要な帯域を割り当てたり，トラフィックの優先制御を行う必要があります。本装置は，ネットワークの通信経路上に設置され，回線帯域を分割し，割り当てた帯域に対して最低帯域を保証したり，最大帯域制限を行うなどのトラフィックコントロールを行うことができます。

トラフィックコントロール機能についてのさらに詳細な説明は，「第 8 章 トラフィックコントロール機能」を参照してください。

2.2　リンクダウン転送機能

一方のリンクのダウンを検出すると他方のリンクをダウンさせ，リンク異常を通知することができます。

リンクダウン転送機能についてのさらに詳細な説明は，「第 9 章 リンクダウン転送機能」を参照してください。

2.3　SSH 機能

SSH サーバ機能により，本装置と SSH クライアント間の通信が暗号化され，安全性が保証されていないネットワークを経由する場合でも，セキュアな遠隔操作が可能になります。

SSH 機能についてのさらに詳細な説明は，「第 10 章 SSH 機能」を参照してください。

2.4　Simple Network Management Protocol（SNMP）機能

SNMP は，ルータやサーバなどのネットワーク機器に対してネットワークを通してリモートで管理するためのプロトコルです。

SNMP 機能についてのさらに詳細な説明は，「第 11 章 SNMP の設定」を参照してください。

2.5　統計情報

各カウンタ，キューバッファ情報などの統計情報があります。

統計情報についてのさらに詳細な説明は，「第 12 章 統計情報」を参照してください。

2.6　トップカウンタ機能

トップカウンタ機能は，トラフィックの利用状況を把握するための機能です。

トップカウンタ機能についてのさらに詳細な説明は，「第 13 章 トップカウンタ機能」を参照してください。

2.7　WebAPI 機能

WebAPI 機能は，本装置のトラフィックコントロール機能の設定を行う際に，HTTP（Hypertext Transfer Protocol：RFC2616）を使用して設定を行う機能です。

WebAPI についてのさらに詳細な説明は，「第 14 章 WebAPI 機能」を参照してください。

2.8　RADIUS 機能

RADIUS 機能は，TELNET，SSH，およびシリアルコンソールのログイン時に，RADIUS（RFC2865）を使用してユーザ認証する機能です。

RADIUS についてのさらに詳細な説明は，「第 15 章 RADIUS 機能」を参照してください。

2.9　ドメインフィルタ機能

ドメインフィルタ機能は，帯域制御のパケット分類識別子としてドメイン名を使用することができる機能です。

ドメインフィルタ機能についてのさらに詳細な説明は，「第 17 章 ドメインフィルタ機能」を参照してください。

2.10　トラフィック分析機能

トラフィック分析機能は，ネットワークのパケット損失、転送遅延やサーバクライアントの負荷状況などを測定する機能です。

トラフィック分析機能についてのさらに詳細な説明は，「第 18 章 トラフィック分析機能」を参照してください。

（空白ページ）

ここでは，設定の基本について説明します。

本装置の設定は Command Line Interface（CLI）を使用します。CLI はコンソールポートからコンソールケーブル経由で接続したターミナル（端末），またはシステムの IP ネットワークインタフェース（システムインタフェース）へのネットワーク経由で Telnet および SSH によるリモートアクセスが利用可能です。システムインタフェースへの通信は，Ethernet ポートまたは Network ポート経由のどちらかで行うことができます。

3.1 Command Line Interface（CLI）

CLI は，装置の動作パラメータの表示や設定を行うことができます。コマンドの詳細については，「PureFlow GSX トラフィックシェーパー NF7101C コマンドリファレンス」を参照してください。

（1）コンソールポート

コンソールポートの接続条件は次のとおりです。

通信速度:　9600 bit/s
キャラクタ長:　8 ビット
パリティ:　なし
ストップビット長: 1 ビット
フロー制御:　なし

コンソールを接続するシリアルインタフェースコネクタは本体の前面にあります。添付のコンソールケーブルを使用して接続してください。

（注）
通信速度を 115200 bit/s で使用する場合，お使いの環境（端末ハードウェア，ソフトウェア）によっては文字化けや文字抜けが発生する場合があります。文字化けや文字抜けが発生した場合は，通信速度を下げて使用してください。

（2）Telnet

Telnet を使用するためには，本装置のシステムインタフェースの設定を行う必要があります。SSH セッションと Telnet セッションを合わせ，最大 4 セッションまで同時利用可能です。なお，セッション数には WebAPI のセッション数は含みません。

Ethernet ポートまたは Network ポートに接続されたネットワークを経由した端末から Telnet を使用してください。

システムインタフェースの設定の説明は「第 7 章　システムインタフェースの設定」を参照してください。

Telnet を使用しない場合は，“set telnet”コマンドで Telnet を無効にしてください。

（3）SSH

本装置の SSH（Secure Shell）は，SSH Version2 をサポートしています。SSH セッションと Telnet セッションを合わせ，最大 4 セッションまで同時利用可能です。なお，セッション数には WebAPI のセッション数は含みません。

SSH を使用しない場合は，“set ssh”コマンドで SSH を無効にしてください。

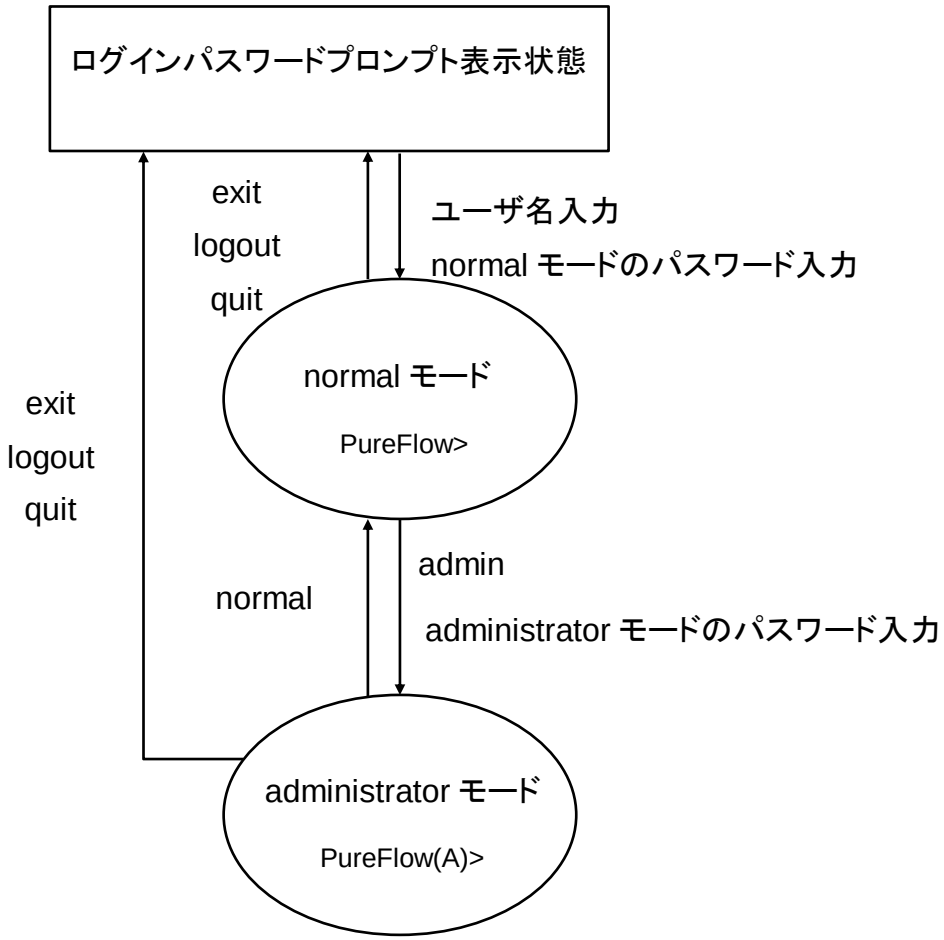
3.2　コマンド構造の説明

本装置の CLI には normal モードと administrator モードの 2 つがあります。normal モードでは，ステータスやカウンタ，および設定値の表示だけができます。administrator モードでは，すべての設定・変更・表示を行うことができます。

本装置のセキュリティを確保するため，normal モードに入るためのパスワードと administrator モードに入るためのパスワードを設定できます。パスワードが設定されている状態では，正しいパスワードを入力しないと，それぞれのモードに移行できません。

また，RADIUS 機能を使用しログイン認証を実施した場合，RADIUS サーバに設定されるユーザごとのサービスタイプに従って，normal モードまたは administrator モードに入ります。詳細は RADIUS 機能を参照してください。

CLI プロンプト表示	CLI モード
PureFlow>	normal モード
PureFlow(A)>	administrator モード



3.3　コマンドシンタックス

本装置 CLI のコマンドシンタックスは以下のような体系です。

アクション　設定項目　値

たとえば

アクション	設定項目	値
↓	↓	↓
設定する	時刻	数値
↓	↓	↓
set	date	20120630101010

また，機能に関する設定項目が多数あるので，「設定項目」は，「設定グループ＋設定項目」のように階層化している場合があります。

設定グループの例

ip
scenario
port

設定グループを伴うコマンドシンタックスの例を以下に示します。

アクション	設定グループ	設定項目	値
↓	↓	↓	↓
設定する	PORT グループ	1/2 の SPEED	100 M 固定
↓	↓	↓	↓
set	port	speed 1/2	100M

3.4　ヘルプ機能

システムプロンプト，またはコマンドの途中で疑問符(?)を入力すると，各コマンド入力モードで使用できるコマンドのリストを表示します。

```
PureFlow(A)> ?
Command                    Description
-------------------------------------------------------------------------------------------
?                          Lists the top-level commands available
add                        Adds some parameters, use 'add ?' for more
                           information
arp                        Shows address resolution table and control
clear                      Clears system statistics, use 'clear ?' for
                           more information
delete                     Deletes some parameters, use 'delete ?' for
                           more information
    .                          .
    .                          .
```

```
PureFlow(A)> set port ?
  flow_control             Sets the flow control parameters

  speed                    Sets the port speed
```

（注）
<?>キーによるヘルプ機能はコマンドラインの最後尾でのみ動作します。

3.5　コマンドの省略形と補完

各コマンドは判別可能な範囲で省略可能です。たとえば，s で始まるコマンドは save，set ，show などがありますが，2 文字目が異なっているので，se と入力されれば“set”コマンドであると判別可能です。以下の 2 つの入力は，同じコマンドを表します。

set port autonegotiation 1/2 disable ＝ se po au 1/2 d

判別可能な文字を入力した段階で，<TAB>キーを入力すると，キーワードを補完して表示します。

```
PureFlow(A)> set po<TAB>
                  ↓
PureFlow(A)> set port
```

（注）
<TAB>キーによる補完機能はコマンドラインの最後尾でのみ動作します。また，コマンドのキーワードによっては，省略および<TAB>キーが動作しないものがあります。その場合は，ヘルプ機能でキーワードを確認し，すべてのキーワードを入力してください。

3.6　ヒストリ機能

コマンド ヒストリの使用方法

CLI は，入力されたコマンドの履歴（記録）機能を持っています。
コマンドヒストリから，次に入力しようとするコマンドに類似した履歴コマンドを呼び出し，あとで説明するコマンド編集機能で編集後，実行することができます。

コマンドヒストリは下記のキー入力で履歴を呼び出すことができます。

Ctrl-Pまたは上矢印キー

最も新しいコマンドからヒストリ バッファのコマンドを呼び出します。このキー操作を繰り返すと，続けて古いコマンドが呼び出されます。

Ctrl-Nまたは下矢印キー

Ctrl-P または上矢印キーでコマンドが呼び出されてから，ヒストリ バッファの新しいコマンドに戻ります。このキー操作を繰り返すと，続けて新しいコマンドが呼び出されます。

また，“show history”コマンドにより，コマンド履歴を表示することができます。

3.7　コマンド編集機能

コマンドラインを編集するために必要なキーストロークを示します。

Ctrl-Bまたは左矢印キー

カーソルを 1 文字分後退させます。

Ctrl-Fまたは右矢印キー

カーソルを 1 文字分前に進めます。

Ctrl-Aキー

カーソルを行の先頭に戻します。

Ctrl-Eキー

カーソルを行の最後に進めます。

Ctrl-DまたはDeleteキー

カーソルの位置にある文字を削除します。

Ctrl-HキーまたはBSキー

カーソルの位置の前の文字を削除します。

Ctrl-Kキー

カーソル以降の文字列を削除するとともに，バッファにコピーします。

Ctrl-Wキー

カーソルで選択された単語を削除するとともに，バッファにコピーします。

Ctrl-Yキー

カーソル位置にバッファの内容をペーストします。

Ctrl-Uキー

カーソルの行を削除するとともに，バッファにコピーします。

（注）
コマンドライン編集機能はコマンドライン表示が 1 行に収まる場合のみ動作します。

3.8　ページャ機能

ターミナルへの表示を伴うコマンドを実行したとき表示内容が 24 行を超えるものについては，画面単位および行単位のページャ機能による表示を行います。その場合は画面の最終行に“—More—”と表示し，表示内容がその行以降も続いていることを表します。
“—More—”が表示されているときに入力可能なキーは，以下のとおりです。

スペースまたはFキー

次の画面を表示します。

Enterキー

次の行を表示します。

Qキー

表示を終了します。

3.9　起動とログイン

本装置の電源を投入すると，装置内部の内蔵フラッシュメモリ（以後，内蔵フラッシュメモリ）のソフトウェアオブジェクトを自動で読み込み起動します。また，ソフトウェアオブジェクト（ファイル名：nf7100.bin）が入った CF カードまたは USB メモリ（以下外部メディア）を挿入して電源を投入すると外部メディア内のソフトウェアオブジェクトを優先して読み込み起動します。外部メディアの優先順位は USB メモリ，CF カードの順です。

コンフィギュレーションについても同様に，設定ファイル（ファイル名：extcnf.txt）が入った外部メディアが挿入されている場合，外部メディア内の設定ファイルを優先して読み込みます。

外部メディアの読み込み中は，外部メディアに対しアクセスをしていますので，起動が完了する前に外部メディアを抜いたり電源をオフにすると，外部メディアが破損する恐れがあります。

本装置のコンソールポートに接続されている場合は，下記のような起動メッセージが表示されます（起動メッセージでの表示項目については，ソフトウェアバージョンによって，変更されることがあります）。

```
Anritsu PureFlow NF7101-S003A Software Version 1.5.2
Copyright 2015-2021 ANRITSU CORPORATION

Power Supply 0                  ... [OK]
Power Supply 1                  ... [NONE]
Fan 0                           ... [OK]
Fan 1                           ... [OK]
Serial Port                     ... [OK]
Backup Memory Checking          ... [OK]
Real Time Clock Checking        ... [OK]
File System Checking            ... [OK]
EEPROM Checking                 ... [OK]
Ethernet Controller Checking
        Management Port         ... [OK]
        Internal Port           ... [OK]

.
Loading Forwarding Processor module software
.....completed

Slot 1 boot up complete
      Medium type 10GBase-R 2 ports

System booting up
.....
Restoration in Progress
100 % done

Restoration completed

PureFlow login:
```

設定に際しては，まずシステムコンソールとしてコンソールポートに添付のコンソールケーブルを接続します。コンソールが接続され，［Enter］キーを入力すると，コンソール上に次のようなメッセージを表示し，ログイン受付状態となります。

```
PureFlow login:
```

本装置のユーザ名は“root”です。また，工場出荷時の初期状態において，ログインパスワードは未設定となっています。ログインが認証されると，プロンプトが表示され，コマンド受付状態になります。

```
PureFlow login:root
Password:([Enter]キーを入力)
PureFlow＞
```

この状態は normal モードで，設定内容を見ることはできますが，内容を変更することはできません。設定を行うためには administrator モードに移行する必要があります。この移行は“admin”コマンドで行います。

```
PureFlow＞admin
Password:([Enter]キーを入力)
PureFlow(A)＞
```

この administrator モードでは，各種パラメータの表示に加えて，動作パラメータの変更，パスワードの設定が可能になります。administrator モードは，同時に複数のユーザが移行でき，同時に設定変更が可能です。administrator モードの権限は，パスワードを設定するなど，ユーザ管理を行ってください。

3.10 設定の保存方法

本装置にて設定した内容は，コマンドによる設定後すみやかに有効となりますが，そのままでは電源断時に設定内容は失われ，再起動後は無効となります。本装置は内蔵フラッシュメモリに設定内容をコンフィギュレーションファイルとして保存することが可能です。次回電源投入後に設定内容を有効にするためには，内蔵フラッシュメモリに save コマンドにて設定内容を保存する必要があります。

保存方法は次のとおりです。

```
PureFlow(A)＞ save config
Do you wish to save the system configuration into the flash memory (y/n)? y
.....
Done
PureFlow(A)＞
```

3.11 設定のリストア方法

本装置の電源を投入すると，内蔵フラッシュメモリに保存されたコンフィギュレーションファイルを自動で読み込みます。また，コンフィギュレーションファイル（ファイル名：extcnf.txt）が格納されている CF カードまたは USB メモリ（以下外部メディア）を挿入して電源を投入すると，外部メディア内のコンフィギュレーションファイルを優先して読み込みます。外部メディアの優先順位は USB メモリ，CF カードの順です。

外部メディアの読み込み中は，外部メディアに対しアクセスをしていますので，起動が完了する前に外部メディアを抜いたり電源をオフにすると，外部メディアが破損する恐れがあります。

3.12 装置の起動時間

コンフィギュレーション情報量によって，save コマンド実行時間および電源投入時の起動時間が異なります。以下に，参考値を示します。

	save コマンド実行時間	起動時間
デフォルト	—	2 分 00 秒
シナリオ 100 件	2 秒	2 分 00 秒
シナリオ 1000 件	5 秒	2 分 20 秒
シナリオ 10000 件	50 秒	7 分 00 秒
シナリオ 100 件 フィルタ 100 件	2 秒	2 分 10 秒
シナリオ 1000 件 フィルタ 1000 件	20 秒	2 分 40 秒
シナリオ 10000 件 フィルタ 10000 件	15 分 50 秒	10 分 30 秒
ルールリストグループ 100 件 ルールリストエントリ 100 件	3 秒	2 分 20 秒
ルールリストグループ 100 件 ルールリストエントリ 1000 件	5 秒	2 分 30 秒
ルールリストグループ 100 件 ルールリストエントリ 10000 件	30 秒	5 分 20 秒

※　フィルタ／シナリオ／ルールリストの設定の説明は「第 8 章 トラフィックコントロール機能」を参照してください。

※　save コマンド実行時間と起動時間は，設定コマンドのライン数やパラメータの数によって異なります。

第4章　装置本体の情報表示と設定

ここでは，装置本体の情報表示と設定について説明します。

本装置には時刻，CLI パスワードなどの装置全体にかかわる設定や，ハードウェア，ソフトウェアのバージョン表示などの装置全体にかかわる情報があります。これらの情報表示と設定について説明します。

本装置には，下記の装置本体情報と設定項目があります。

日付／時刻	装置内蔵のカレンダ・クロックです。SYSLOG によるイベントの記録に使用されます。
SNTP	Simple Network Time Protocol（SNTP）クライアント機能です。
ユーザ名とパスワード	CLI による装置へのアクセス制御のためのユーザ名とパスワードです。
SYSLOG 設定	装置の状態変化イベントやエラーイベントを内蔵メモリ，バッテリバックアップメモリに保存したり，リモートホストに送信することができます。
モジュール情報	装置内の各モジュール情報（バージョンなど）です。

4.1　日付／時刻

本装置は，カレンダ機能に対応しています。日付，時刻はSYSLOGによるイベントの記録に使用されます。日付，時刻の設定は CLI コマンドで指定する方法と，SNTP クライアント機能により NTP サーバの時刻に自動同期させる方法があります。

CLIコマンドによる設定

CLI で設定する場合は以下のコマンドを使用します。

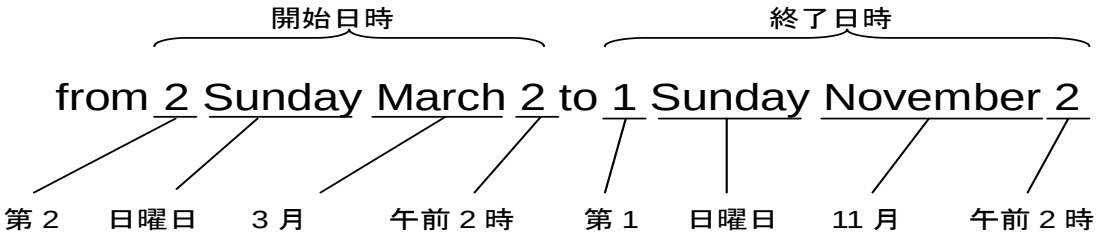
set date <yyyymmddhhmmss>	日付，時刻の設定を行います。
set timezone 　<hours-offset> [<minutes-offset>]	協定世界時（UTC：Coordinated Universal Time）からのタイムゾーンオフセットを設定します。 デフォルト値は+9[時間]0[分]です。
set summertime 　from <week> <day> <month> <hh> 　to <week> <day> <month> <hh> 　[offset]	夏時間の適用期間を設定します。 デフォルトでは設定されていません。
unset summertime	夏時間の設定を解除します。
show date	日付，時刻の表示を行います。

コマンドの実行例を示します。

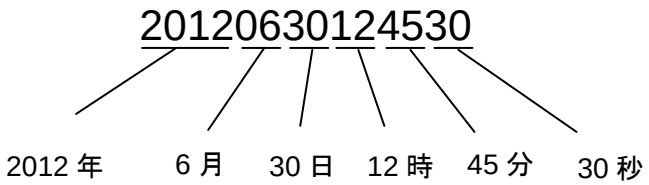
```
PureFlow(A)> set timezone +9
PureFlow(A)> set summertime from 2 Sunday March 2 to 1 Sunday November 2
PureFlow(A)> set date 20120630124530
PureFlow(A)> show date
May 18 2005(Mon) 12:45:32
UTC Offset      : +09:00
Summer Time     : From Second Sunday March 02:00
                  To     First Sunday November 02:00
                  Offset 60 minutes
PureFlow(A)>
```

タイムゾーンの設定は，UTC（協定世界時）からのオフセット時間を符号付きで入力します。必要ならば分単位のオフセットを入力します。

夏時間の設定は，夏時間の開始日時と終了日時を指定します。必要ならば夏時間である間時刻に加えるオフセットを分単位で入力します。オフセットを省略した場合は 60[分]が適用されます。
開始日時および終了日時は以下のフォーマットで指定します。



日付，時刻の設定は西暦年，月，日，時，分，秒を続けて 14 桁で入力します。



カレンダ・クロックに設定した時刻は，装置内部のバッテリで駆動され，装置電源がオフの状態でも止まらずに進み続けます。

4.2 Simple Network Time Protocol（SNTP）

本装置は，SNTP クライアント機能を実装しています。SNTP クライアントはシステムインタフェース経由で NTP サーバと通信し，本装置の日付および時刻を NTP サーバと同期させます。SNTP クライアントを使用するためには，本装置のシステムインタフェースの設定を行う必要があります。システムインタフェースの設定の説明は「第 7 章 システムインタフェースの設定」を参照してください。

SNTP クライアントの設定には以下のコマンドを使用します。

set sntp {enable | disable}	SNTP クライアント機能を有効化／無効化します。 有効化後，interval 設定時間が経過すると時刻同期を開始します。
set sntp server <IP_address>	NTP サーバの IP アドレスを設定します。NTP サーバは 1 つのみ指定できます。
set sntp interval <interval>	NTP サーバへ定期的に時刻の問い合わせを行う間隔を秒単位で設定します。設定範囲は 60～86400[秒]です。デフォルトは 3600[秒]です。設定可能な値は上記のとおりですが，実際の動作は 60 秒単位に端数切り上げで丸められます。 変更後の interval 設定時間が経過すると時刻同期を開始します。
sync sntp	NTP サーバへ時刻の問い合わせを行います。 SNTP クライアント機能が有効の場合のみ実行可能です。
show sntp	SNTP クライアント機能の状態および設定を表示します。

NTP サーバ 192.168.10.10，問い合わせ間隔 86400 秒を設定する場合，以下に示すコマンドを実行します。

```
PureFlow(A)> set sntp server 192.168.10.10
PureFlow(A)> set sntp interval 86400
PureFlow(A)> set sntp enable
PureFlow(A)> show sntp
Status          : enable
Server          : 192.168.10.10
Interval        : 86400
Sync            : kept
PureFlow(A)>
```

“show sntp”コマンドの Sync の表示が“kept”になっていれば，NTP サーバとの同期が取れている状態です。
時刻の修正は問い合わせ間隔ごとに行われます。

4.3　ユーザ名とパスワード

装置のセキュリティを保つために装置設定をシリアルコンソール，または Telnet で行う前にはユーザ名とパスワードによる認証が行われます。このパスワードはユーザが変更することができます。

set password	ログインパスワードを設定します。ログインパスワードは 16 文字以内です。
set adminpassword	administorator モードに移行するためのログインパスワードを設定します。ログインパスワードは 16 文字以内です。

コマンドの実行例を示します。

```
PureFlow(A)> set password
New Password:                  ← 設定したいパスワードを入力してください。
Retype the new Password:       ← 設定したいパスワードをもう一度入力してください。
```

ログインパスワードに設定できる文字は，以下の ASCII 文字です。

```
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
!#$%&'()=~-^|¥@`[]{}:*;+_/.,<>
```

ログインパスワードを設定解除する場合は，“New Password”の問いに対し，パスワードを入力せず，[Enter]キーを入力してください。

4.4 SYSLOG

装置に起きたエラーイベントやリンクアップ・ダウンなどのイベント(以後，ログデータと呼ぶ)を複数の方法で記録することができます。本装置はログデータを通電状態で内蔵メモリに最大 8000 イベント保持します。内蔵メモリに保持するログデータは電源が遮断されると消失します。ログデータは内蔵バックアップメモリと，ネットワークを介した SYSLOG ホストに記録することができます。内蔵バックアップメモリへは，前回と前々回の装置稼働時におけるログデータを，それぞれ最大 1200 イベント保持します。内蔵バックアップメモリに保持するログデータは，電源を遮断しても消失しません。

show syslog	内蔵メモリに記録されたログデータを表示します。
show backup syslog [last | second_last]	内蔵バックアップメモリに記録されたログデータを表示します。
clear syslog	内蔵メモリに記録されたログデータをクリアします。
set syslog severity <severity_level>	syslog ホストに送信するシステムログの最低レベル(重大度)を設定します。設定されたレベルより低いレベルのログは syslog ホストに送信されません。なお，装置のシステムログは本設定に関わらず，全ての重大度のシステムログが記録されます。
show syslog host	システムログ出力に関する設定を表示します。
set syslog host {enable | disable}	SYSLOG ホストへの記録を有効化／無効化します。
add syslog host <IP_address> [<udp_port>]	SYSLOGホストのIPアドレス／UDPポートを追加します。
delete syslog host <IP_address>	SYSLOGホストのIPアドレス／UDPポートを削除します。
set syslog facility {ccpu | fcpu} <facility_code>	システムログの facility を設定します。 ccpu： Control CPU(制御系 CPU)で検出，記録したログメッセージ fcpu： Forwarding CPU(フォワーディング系 CPU)で検出，記録したログメッセージ

ログデータはテキストデータとして以下のフォーマットで装置内に記録されています。

・show syslog コマンドで表示される内蔵メモリのログデータ

日時	Host	Ident	PID	メッセージ
Jun 30 16:51:19	PureFlow	System	[10330]	Port 1/1 changed Up from Down.

・show backup syslog コマンドで表示される内蔵バックアップメモリのログデータ

プライオリティ	日時	メッセージ
134	2012 Jun 30 16:51:19	Port 1/1 changed Up from Down.

日時

イベントが発生した日時です。

Host

Host はログメッセージを記録した装置名を示します。”PureFlow”固定です。

Ident

Ident はログメッセージを記録したプログラムの識別子を示します。”System”固定です。

PID

PID はログメッセージを記録したプロセスのプロセス ID 値を示します。

メッセージ

イベントの内容を示すメッセージが格納されます。

メッセージは show syslog コマンドで表示できます。

```
PureFlow> show syslog
-------------------------------------------------------------------------------------------------------
Date      Time     Host         Ident       [PID]     Message
-------------------------------------------------------------------------------------------------------
Jan 25 21:50:54 PureFlow   System      [10330]: Port 1/1 changed Up from Down.
```

データは装置の通電中，メモリに保持されていますが，オペレータがメッセージをクリアすることができます。

```
PureFlow> clear syslog
PureFlow> show syslog
-------------------------------------------------------------------------------------------------------
Date      Time     Host         Ident       [PID]     Message
-------------------------------------------------------------------------------------------------------
PureFlow>
```

プライオリティ

プライオリティはログメッセージの特徴を示すコードです。プライオリティのコードは RFC3164 で規定されている方式で計算し，格納されます。プライオリティコードはメッセージのカテゴリを表す Facility とメッセージの重大度を示す Severity の 2 つの数値を組み合わせたコードで表現されます。

プライオリティ＝Facility×8＋Severity

本装置の SYSLOG メッセージの Facility は設定が可能です。設定可能な Facility の範囲は 0～23 です。デフォルト値は以下となります。

control CPU：16
forwarding CPU：17

コマンドの実行例を以下に示します。

```
PureFlow(A)> set syslog facility ccpu 18
PureFlow(A)> set syslog facility fcpu 19
```

control cpuのFacilityを18にします。

forwarding cpuのFacilityを19にします。

Severity には 0 から 6 までの数値が格納されます。プライオリティ 0 が最も重大度が高く，数値が大きくなるほど低くなります。各メッセージの重大度は RFC 3164 に規定された以下の基準に従って割り当てられています。

Numerical Code	Severity	
0	Emergency:	system is unusable
1	Alert:	action must be taken immediately
2	Critical:	critical conditions
3	Error:	error conditions
4	Warning:	warning conditions
5	Notice:	normal but significant condition
6	Informational:	informational messages

たとえばプライオリティ 129（16×8＋1）のメッセージは Facility が 16，Severity が 1 です。つまり Control CPU で検出された Alert レベル（緊急）メッセージです。

4.5　モジュール情報

装置内の各モジュール情報を表示します。バージョン，製造番号などを確認することができます。

show module	各モジュール情報を表示します。

モジュール情報には，以下のものがあります。

MAC Address

装置の MAC アドレスを表します。

Chassis Model Name

本体の形名を表します。形名は，以下のとおりです。

NF7101C：PureFlow GSX

Chassis Serial Number

本体の製造番号を表します。

Control Module Version

Control モジュールのハードウェアバージョンを表します。

Shaper Module Version

Shaper モジュールのハードウェアバージョンを表します。

Software Version

インストールしたソフトウェアのバージョンを表します。

Management U-Boot Version，Forwarding U-Boot Version

U-Boot バージョンを表します。

MCU-C Version，MCU-S Version

MCU バージョンを表します。

Uptime

本装置が起動してからの動作時間を表します。

Temperature

入気温度を表します。

Power Supply Unit N

電源ユニットの状態を表します。

FAN Unit N

ファンユニットの状態を表します。

コマンドの実行例を示します。

```
PureFlow(A)> show module
Anritsu PureFlow NF7101-S003A Software Version 1.5.2
Copyright 2015-2021 ANRITSU CORPORATION

MAC Addres                          : 00-00-91-12-34-56

Chassis Model Name                  : NF7101C
Chassis Serial Number               : 1234567890

Control Module Version              : A00
Shaper Module Version               : A00
Software Version                    : 1.5.2
Management U-Boot Version           : 1.1.1
Forwarding U-Boot Version           : 1.1.1
MCU-C Version                       : 001
MCU-S Version                       : 001

Uptime                              : 0 days, 00:27:17
Temperature
  Intake   Temperature              : 32C
Power Supply Unit 0
  Operation Status                  : operational
  Fan Speed                         : 6240[rpm]
Power Supply Unit 1
  Operation Status                  : not present
  Fan Speed                         : 0[rpm]
FAN Unit 0
  Operation Status                  : operational
  Fan Speed                         : 3840[rpm]
FAN Unit 1
  Operation Status                  : operational
  Fan Speed                         : 3960[rpm]
PureFlow(A)>
```

4.6 ライセンスキー

ライセンスキーを購入することにより，本装置の機能や性能を拡張することが可能です。

ライセンスキーは，キーを記載したライセンス証書で提供されます。ライセンスキーを装置購入後に購入する場合は，装置シリアル番号をご指定ください。

ライセンスキーを本装置に設定するには，“set option”コマンドを入力してください。ライセンスキー入力を促すメッセージが表示されますので，ライセンスキーを入力してください。ライセンスキー入力の際，4 文字ごとにハイフンを入力しても，ハイフンを入力しなくても同じライセンスキーとして認識します。入力されたライセンスキーと装置のシリアル番号を比較し，一致した場合にライセンスが有効となります。

ライセンスキーに関するコマンドには以下ものがあります。

set option	ライセンスキーを本装置に設定します。
show option	有効になっているライセンスを表示します。

コマンドの実行例を示します。

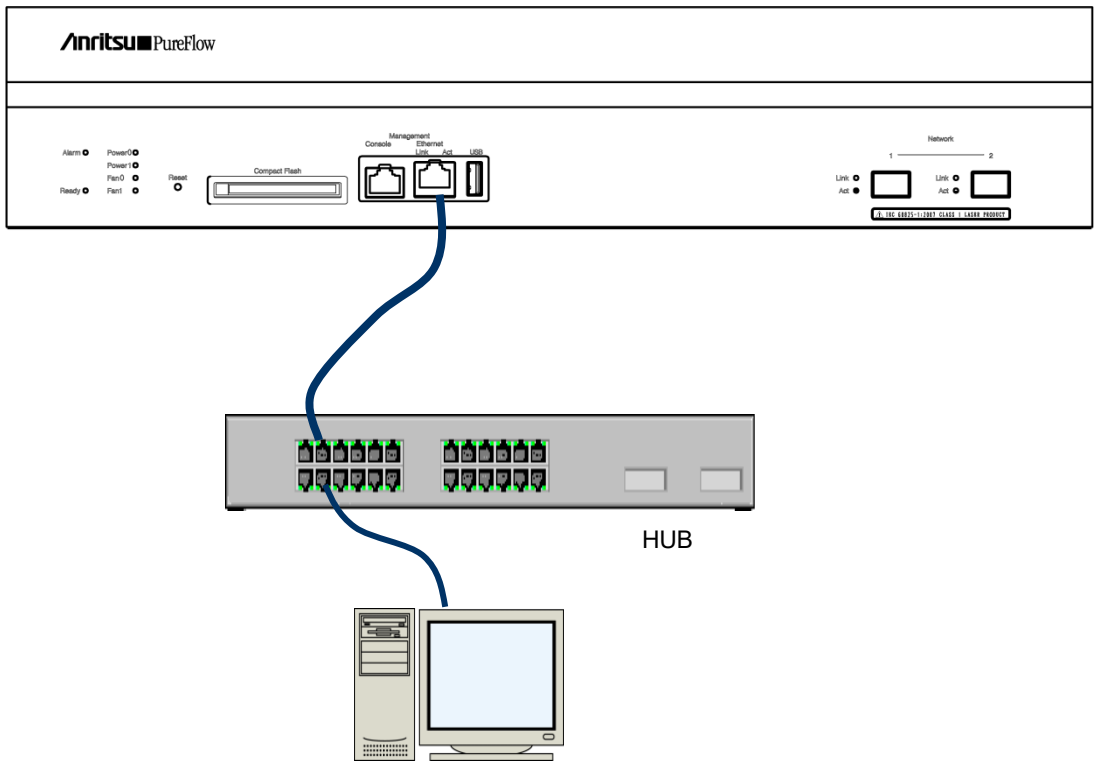
```
PureFlow(A)> set option
Enter the option key : XFS8wbFEFBNkfqLJ

Authentication succeed.

    Making be available : License Key NF7101-L004C (Extended Bandwidth 10Gbps)
Updation done.
Enter update scenario command to change port bandwidth.
PureFlow(A)>
PureFlow(A)> show option
    License Key NF7101-L004C available  (Extended Bandwidth 10Gbps)
PureFlow(A)>
```

（空白ページ）

本装置は，ネットワーク経由のリモートによる設定，制御を行うために，Ethernet ポートを装置前面に実装しています。
本ポートはマネジメント用のローカルポートで，Network ポートとは切り離されています。本ポートは Auto-MDIX をサポートした 10/100/1000BASE-T ポートです。



Ethernet ポートに接続されたネットワーク経由のリモートによる設定，制御を行うためには，本装置のシステムインタフェースの設定を行う必要があります。システムインタフェースの設定の説明は「第 7 章 システムインタフェースの設定」を参照してください。

（注 1）
1 Gbit/s 通信を行う場合は，AutoNegotiation 有効で使用してください。AutoNegotiation 無効で通信速度を 1Gbit/s 固定設定とすることはできません。AutoNegotiation 無効で通信速度を 1Gbit/s 設定とした場合，強制的に AutoNegotiation 有効となります。この場合，通信速度は 1Gbit/s のみアドバタイズされます。

（注 2）
通信速度／duplex モードの設定は，AutoNegotiation 無効のときのみ有効です。AutoNegotiation 有効のとき，AutoNegotiation の結果が反映されるため，これらの設定内容は適用されず，AutoNegotiation 無効に設定したときに反映されます。“show port”コマンドでリンク状態が半二重の場合，ポートの AutoNegotiation／通信速度／duplex モードの設定が接続装置と合っているか確認してください。

（注 3）
Ethernet ポートの最大フレーム長は 1518 Byte 固定です。

（空白ページ）

第6章　Network ポートの設定

ここでは，本装置の Network ポートの設定について説明します。

6.1　概要

Network ポートとは，ネットワーク上に流れるトラフィックをコントロール（トラフィックコントロール）するためのポートです。

本装置の Network ポートには，以下に示す SFP/SFP+を装着することが可能です。（注 1 参照）

SFP+ 10GBASE-SR／10GBASE-LR（LC コネクタ）
SFP　1000BASE-SX／1000BASE-LX（LC コネクタ）
SFP　1000BASE-T（RJ-45／Auto-MDIX）

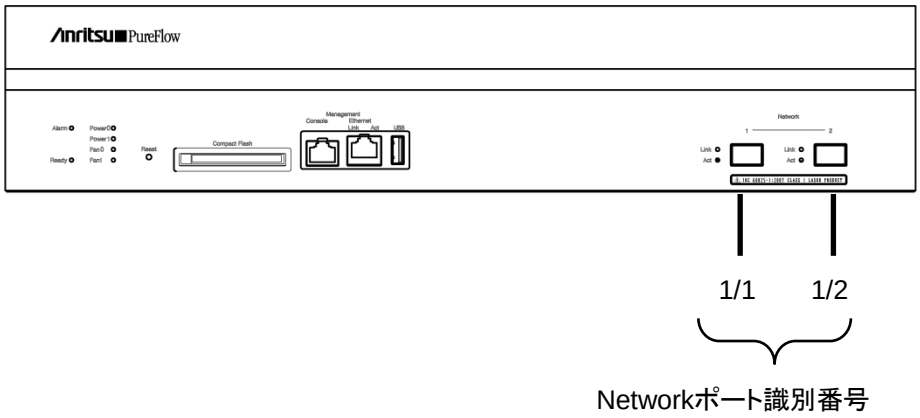
Network ポートには以下の設定が可能です。
AutoNegotiation の有効／無効（注 2，注 4 参照）
フローコントロール（auto，pause フレーム受信／送信）
通信速度（10 Mbit/s，100 Mbit/s，1 Gbit/s）（注 3 参照）
duplex モード（full，half）（注 3 参照）
最大パケット長（2048 Byte，10240 Byte）（注 5 参照）

上記設定は装着されている SFP によって適用範囲が異なります。

	10GBASE-SR/LR	1000BASE-SX/LX	1000BASE-T
AutoNegotiation	適用されない	有効／無効	有効／無効
通信速度	10G のみ	1G のみ	10M/100M/1G
duplex モード	Full のみ	Full のみ	Full/Half
フローコントロール	受信 ON/OFF 送信 ON/OFF	Auto 受信 ON/OFF 送信 ON/OFF	Auto 受信 ON/OFF 送信 ON/OFF
最大パケット長	2048/10240	2048/10240	2048/10240

CLI から Network ポートを指定するには＜スロット番号／ポート番号＞の組み合わせで指定します。
PureFlow GSX のスロット番号には 1 を指定します。

スロット内のポート番号は左から順番に 1/1，1/2 と番号付けられており，これにより，Network ポートの識別番号は以下のようになります。



Network ポートに接続されたネットワーク経由のリモートによる設定，制御を行うためには，本装置のシステムインタフェースの設定を行う必要があります。システムインタフェースの設定の説明は「第 7 章 システムインタフェースの設定」を参照してください。

（注 1）
PureFlow GSX では，装置起動中に Network ポートに実装されている SFP+または SFP を識別するため，装置起動後に SFP+と SFP の交換を行った場合，装置の再起動が必要です。装置起動後に SFP+と SFP の交換を行った場合，当該 Network ポートの Active/Link LED が両点滅し，再起動が必要であることを示します。
SFP+同士（10GBASE-SR と 10GBASE-LR）の交換，SFP 同士（1000BASE-SX/LX，1000BASE-T）の交換では装置再起動は必要ありません。
装置起動中に何も実装されていない Network ポートは，SFP+が実装されている場合の動作となります。

（注 2）
10GBASE-SR/LR の場合，AutoNegotiation の設定は適用されません。

（注 3）
10GBASE-SR/LR の場合，通信速度は 10G，duplex モードは Full のみとなります。
1000BASE-SX/LX の場合，AutoNegotiation の設定にかかわらず，通信速度は 1G，duplex モードは Full となります。
1000BASE-T SFP の通信速度／duplex モードの設定は，AutoNegotiation 無効のときのみ有効です。AutoNegotiation 有効のとき，これらの設定内容は無効です。

（注 4）
最大パケット長の設定は，システムインタフェースには適用されません。システムインタフェースの最大パケット長は 1518 Byte 固定です。

（注 5）
“show port”コマンドでリンク状態が半二重の場合，ポートの AutoNegotiation／通信速度／duplex モードの設定が接続装置と合っているか確認してください。

（注 6）
1000BASE-T SFP で 1 Gbit/s 通信を行う場合は，AutoNegotiation 有効で使用してください。AutoNegotiation 無効で通信速度を 1Gbit/s 固定設定とすることはできません。AutoNegotiation 無効で通信速度を 1Gbit/s 設定とした場合，強制的に AutoNegotiation 有効となります。この場合，通信速度は 1Gbit/s のみアドバタイズされます。

6.2 Network ポートの属性の設定

1000BASE-T SFP 使用時, AutoNegotiation 無効のときは, Network ポートの通信速度や duplex モードといったポートの動作属性を CLI から変更できます。これらの Network ポート属性は, 通常 AutoNegotiation により, 最も適切な動作モードに自動的に設定されます。接続先のスイッチやノードが AutoNegotiation をサポートしていない場合は, Network ポートの通信速度と duplex モードをマニュアル設定する必要があります。接続先が AutoNegotiation 設定になっている場合は, 本装置も AutoNegotiation 設定にしてください。片方がマニュアル設定で, 他方が AutoNegotiation 設定になっていると, 正しく接続できません。

set port autonegotiation <slot/port> {enable | disable}	Network ポートの AutoNegotiation の有効／無効を設定します。デフォルトは enable です。
set port speed <slot/port> {10M | 100M | 1G}	Network ポートの通信速度を設定します。本設定は, AutoNegotiation 無効のときの通信速度設定です。AutoNegotiation 有効のとき, この設定内容は無効です。デフォルトは 1G です。 注）1000BASE-T の 1 Gbit/s 通信は, AutoNegotiation 有効で使用してください。
set port duplex <slot/port> {full | half}	Network ポートの duplex モードを設定します。本設定は, AutoNegotiation 無効のときの duplex モード設定です。AutoNegotiation 有効のとき, この設定内容は無効です。デフォルトは full です。

Network ポート 1/2 の AutoNegotiation 無効, 通信速度 100 Mbit/s, duplex モード full を設定する場合, 以下に示すコマンドを実行します。

```
PureFlow(A)> set port autonegotiation 1/2 disable
PureFlow(A)> set port speed 1/2 100M
PureFlow(A)> set port duplex 1/2 full
```

また，いずれの SFP+/SFP 使用時においても，Network ポートのフローコントロールを CLI から変更できます。

set port flow_control <slot/port> auto set port flow_control <slot/port> {recv | send} {on | off}	Network ポートのフローコントロールを設定します。デフォルトは auto です。 なお、auto を指定した場合、ポートタイプにより次のように動作します。 ポートタイプ 1000BASE-T および 1000BASE-X の場合： AutoNegotiation により pause フレームの受信および送信を決定します。 AutoNegotiation 無効の場合は受信および送信ともに有効となります。 ポートタイプ 10GBASE-R の場合： 受信および送信ともに有効となります。

Network ポート 1/2 のフローコントロールで pause フレームを送受信しない設定にする場合，以下に示すコマンドを実行します。

```
PureFlow(A)> set port flow_control 1/2 recv off
PureFlow(A)> set port flow_control 1/2 send off
PureFlow(A)>
```

6.3　最大パケット長の設定

Network ポートで転送可能なパケットの最大長を CLI から変更できます。最大パケット長は Ethernet ヘッダ先頭から FCS までのパケット全体の長さで指定します。ただし，VLAN Tag および二重 VLAN Tag のサイズは除いた長さです。VLAN Tag が付加されたパケットは設定値+4 Byte，二重 VLAN Tag が付加されたパケットは設定値+8 Byte まで転送可能です。

set port maxpacketlen {2048 | 10240}	Network ポートの最大パケット長を設定します。 デフォルトは 2048 Byte です。

最大パケット長は各 Network ポートで共通の設定値です。また，10240 Byte に設定変更した場合，帯域制御の設定においてトラフィックアトリビュートの設定範囲が下記のとおりに変更されます。

	2048 [Byte]	10240 [Byte]
最小帯域	0, 1 k[bit/s]～10 G[bit/s] 1 k[bit/s]単位	0, 5 k[bit/s]～10 G[bit/s] 5 k[bit/s]単位
最大帯域	2 k[bit/s]～10 G[bit/s] 1 k[bit/s]単位	10 k[bit/s]～10 G[bit/s] 5 k[bit/s]単位
バッファサイズ	2 k[Byte]～100 M[Byte]	11 k[Byte]～100 M[Byte]

最大パケット長の変更によって，すでに設定済のトラフィックアトリビュートが範囲外となる場合，自動的に範囲内への丸めが行われます。また，追加で登録する場合は丸めが適用される旨の Warning メッセージが表示されます。いずれにおいても帯域制御は丸め後の値で実行されます。

最大パケット長を 10240 Byte に設定変更するには装置の再起動が必要です。最大パケット長を 10240 Byte に設定する場合，以下に示すコマンドを実行します。

```
PureFlow(A)> set port maxpacketlen 10240
Warning
This configuration change will be take effect on next boot.
Please save the system configuration and reboot the system.
If changed to 10240, some scenario parameters will be rounded as below.
   minimum value of minimum bandwidth      1k ->   5k
   minimum value of peak bandwidth         2k -> 10k
   bandwidth resolution                    1k ->   5k
   buffer size minimum                     2k -> 11k

Do you wish to save the system configuration into the flash memory (y/n)?y
.....
Done

Rebooting the system, ok (y/n)?y
```

コマンドを実行すると，再起動が必要であること，およびシナリオパラメータの設定範囲に関する Warning メッセージとともに，コンフィギュレーションの保存を確認するプロンプトが表示されます。“y”を入力してコンフィギュレーションを保存してください。次に，装置の再起動を確認するプロンプトが表示されます。“y”を入力して装置を再起動してください。装置を再起動すると 10240 バイトへの設定変更が適用されます。

6.4　設定，状態の確認

設定コマンドで設定した内容や，現在の Network ポートの動作状態を確認するには，“show port”コマンドを使用します。

```
PureFlow(A)>  show port
Port     Type              Status     Link   Autonego   Speed   Duplex
----     --------------    ------     ----   --------   -----   ------
1/1      1000BASE-T        Enabled    Up     Enabled    100M    Full
1/2      1000BASE-T        Enabled    Up     Enabled    100M    Full
system   1000BASE-T        Enabled    Up     Enabled    100M    Full
PureFlow(A)>
```

“show port”コマンドにより，実装されているすべての Network ポートの状態が確認できます。さらに詳細な情報を確認するには，Network ポート識別番号をコマンド引数で指定することにより，確認できます。

```
PureFlow> show port 1/1

Slot/Port              :1/1
Port type              :1000BASE-T
Admin status           :Enabled
Oper status            :Up
Auto negotiation       :Enabled
Admin speed            :100M
Admin duplex           :Full
Tx Flow control        :Auto
Rx Flow control        :Auto
Admin Max Packet Len   :2048
Oper Max Packet Len    :2048
PureFlow>
```

Network ポートの統計情報を確認するには，“show counter”コマンドを使用します。本コマンドで表示するカウンタ長は，32 ビットです。

```
PureFlow(A)> show counter
Port          Rcv Octets      Rcv Packets     Trs Octets      Trs Packets
-------       ---------------  ---------------  -------------  ---------------
 1/1          57566366          14194297                0              0
 1/2                 0                 0         59383412       14195494
system           58368               152            85424            152

Port          Rcv Broad       Rcv Multi       Trs Broad       Trs Multi
-------       -------------   -------------   -------------   ------------
 1/1              10000         14208097                0              0
 1/2                 0                 0            10000       14209615
system                5                 0               10              0

Port          Err Packets     Collision       Discard
-------       ---------------  ------------   -----------
 1/1                 0                 0              0
 1/2                 0                 0              0
system             N/A               N/A            N/A
```

また，Network ポート識別番号をコマンド引数で指定することにより，詳細内容を表示できます。本コマンドで表示するカウンタ長は，64 ビットです。“show counter”コマンドの 32 ビットカウンタがラップアラウンドした場合，“show counter <slot/port>”コマンドの 64 ビットカウンタと異なる値が表示されることに注意してください。

```
PureFlow(A)> show counter 1/1
Rcv Packets                    14194297
Rcv Broad                         10000
Rcv Multi                      14208097
Rcv Octets                     57566366
Rcv Rate                             16 [kbps]
Trs Packets                           0
Trs Broad                             0
Trs Multi                             0
Trs Octets                            0
Trs Rate                              0 [kbps]
Collision                             0
Drop                                  0
Discard                               0
Error Packets                         0
        CRC Align Error                       0
        Undersize Packet                      0
        Oversize Packet                       0
```

第7章　システムインタフェースの設定

ここでは，本装置のシステムインタフェースの設定について説明します。

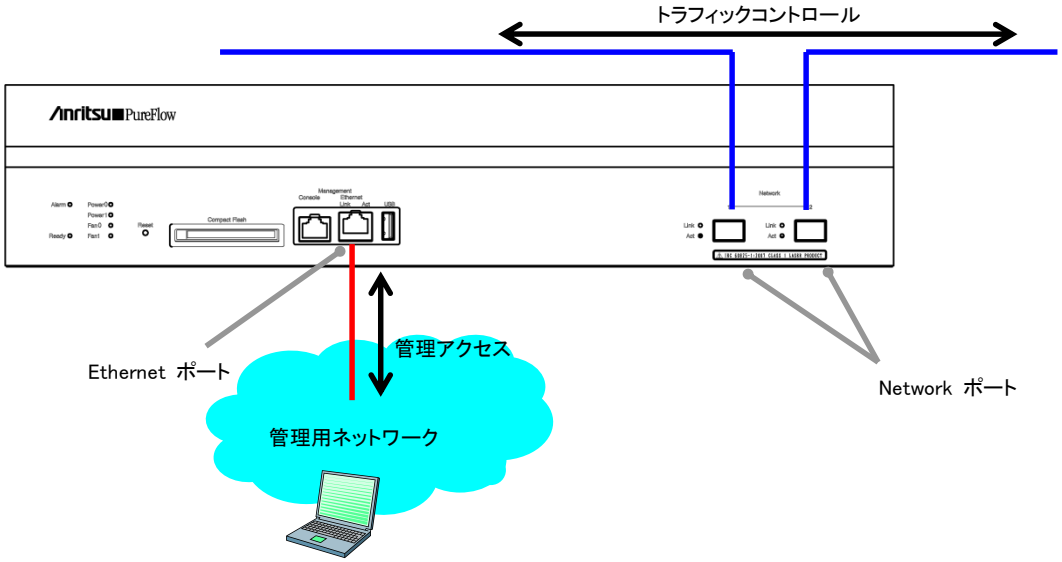
7.1　概要

システムインタフェースとは，管理者が本装置をネットワーク経由でリモートアクセスするための IP ネットワークインタフェースです。本装置へのリモートからの制御には Telnet，SNMP などの手段を用い，本装置の設定および状態監視を行うことができます。

以下に示すように，システムインタフェースは Ethernet ポート経由でアクセスするか，Network ポート経由でアクセスするかのいずれかを選択することができます。

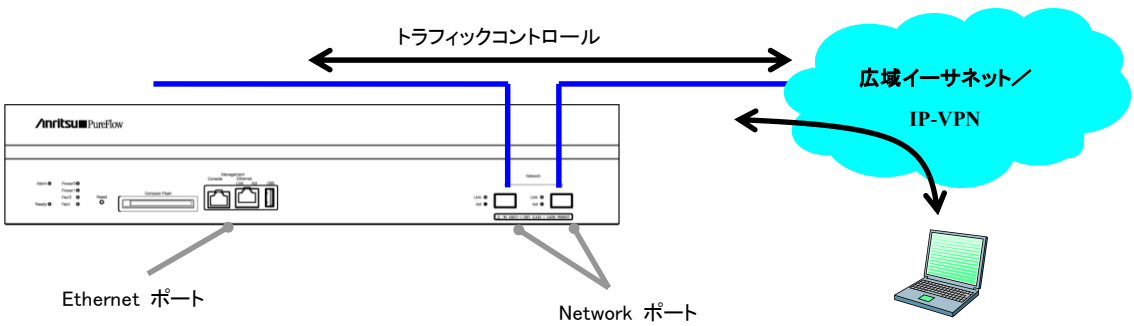
（1）Ethernet ポート経由によるリモート管理

トラフィックコントロールを行うネットワーク（Network ポートからの入出力）とは別の管理用ネットワークに管理者端末を配置し，Ethernet ポートを経由して制御することができます。セキュリティ上，トラフィックコントロールを行うネットワーク内から分離させたい場合などに有効です。



（2）Network ポート経由によるリモート管理

トラフィックコントロールを行うネットワーク内に管理者端末を配置し，Network ポートを経由して制御することができます。管理専用のネットワークを用意する必要がないため，ネットワーク構成をシンプルにすることができます。



7.2　システムインタフェース通信

システムインタフェースへの通信は，Ethernet ポート経由または Network ポート経由のどちらかで行うことができます。Ethernet ポート経由で行う場合は，VLAN Tag なしパケットの通信を行うことができます。また，Network ポート経由で行う場合は，通信を行う Network ポートを指定（1/1 のみ，1/2 のみ，すべて）でき，VLAN Tag なしパケット，VLAN Tag ありパケット，二重 VLAN Tag なしパケット，二重 VLAN Tag ありパケットの通信を行うことができます。
また，不特定多数の端末からシステムインタフェースへの通信を制限するためにフィルタ機能を使用することもできます。

システムインタフェース通信は IPv4 および IPv6 の同時利用が可能ですが，一部の機能は IPv4 のみのサポートとなります。

機能	IPv4	IPv6
Telnet	○	○
SSH	○	○
RADIUS	○	○
TFTP	○	○
FTP	○	○
SYSLOG	○	○
SNTP	○	○
SNMP	○	×
PING	○	○
Telnet クライアント	○	○
WebAPI	○	○
トラフィック分析	○	○
NF7201A NF7202A モニタリングマネージャ	○	×
システムインタフェース フィルタ	○	○

ファイアーウォールなどのセキュリティ設定を行っている場合は，以下のサービスが通信できるように設定を変更してください。

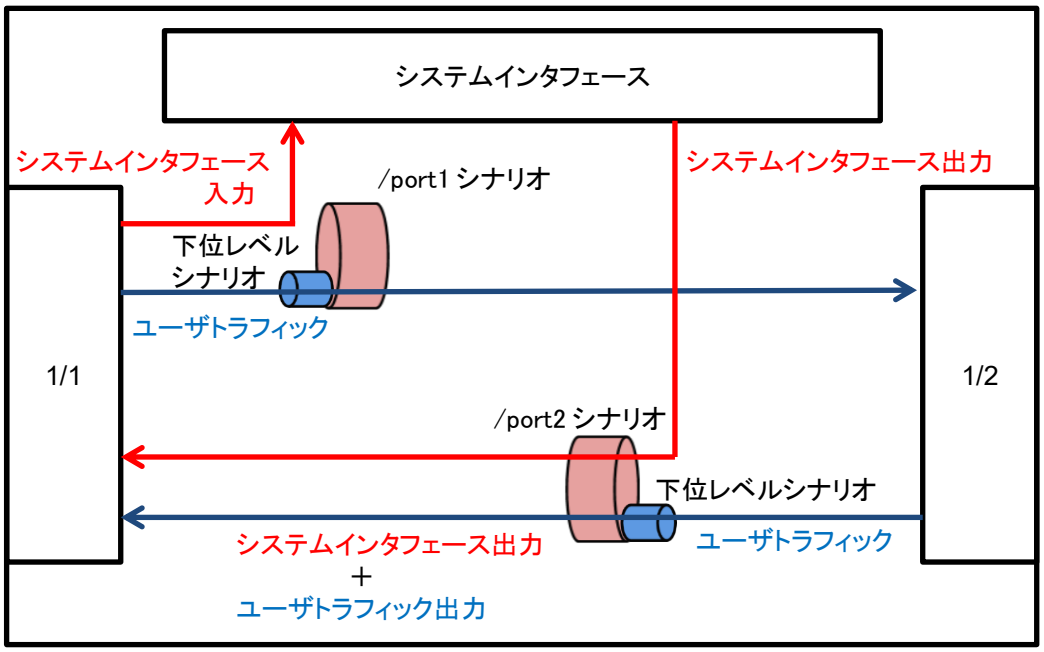
ポート番号	TCP/UDP	サービス名	備考
23	TCP	telnet	telnet 接続
22	TCP	ssh	SSH 接続
1812	UDP	radius	RADIUS 認証
69	UDP	tftp	TFTP 接続
21	TCP	ftp	FTP 制御
20	TCP	ftp	FTP データ転送
514	UDP	syslog	SYSLOG 送信
123	UDP	ntp	SNTP クライアント機能
161	UDP	snmp	SNMP 監視
162	UDP	snmptrap	SNMP TRAP 送信
80	TCP	http	WebAPI, トラフィック分析
443	TCP	https	WebAPI, トラフィック分析
51967	TCP	―	モニタリングマネージャとの接続

（注 1）
Ethernet ポートと Network ポートのどちらか一方でのみ通信を行うことができます。

（注 2）
Ethernet ポート経由の場合，VLAN Tag なしパケットのみ通信を行うことができます。

（注 3）
Network ポート経由の場合，システムインタフェースへの通信中は Network ポートの帯域を使用します。ネットワーク上を流れるトラフィックをコントロールするための帯域を割り当てるときは，システムインタフェース通信の帯域も考慮して設定してください。トラフィックコントロールの設定の説明は「第 8 章 トラフィックコントロール機能」を参照してください。システムインタフェースからの出力トラフィック転送動作は，ソフトウェアバージョンにより異なります。

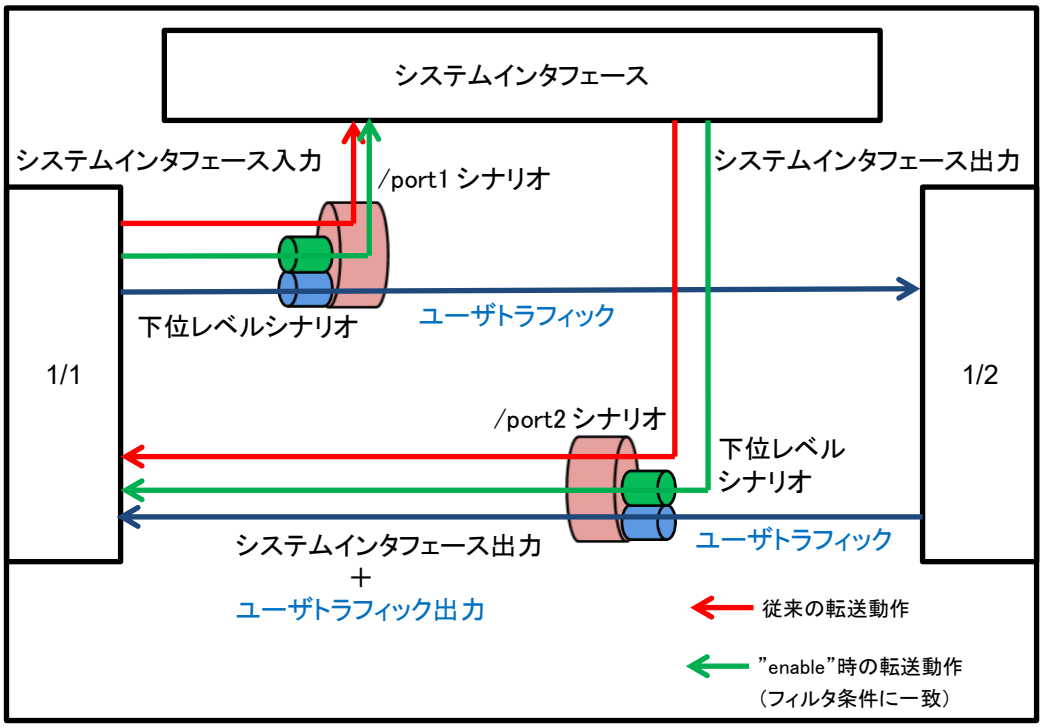
ソフトウェアバージョン Ver.1.4.2 以前は，システムインタフェースからの出力トラフィックは，出力ポートとは逆側ポートのポートシナリオ（ポート 1/1 への出力なら“/port2”シナリオ）の帯域を使用し，最優先のクラス 1 を割り当てています。また，当該シナリオの入出力カウンタにも加算されます。入力トラフィックについてはシナリオの帯域は使用せず，シナリオカウンタにも加算されません。



システムインタフェース	シナリオカウンタ		シナリオ動作
	受信	送信	
入力トラフィック	×	×	シナリオの帯域を使用しません。
出力トラフィック	○	○	出力ポートとは逆側ポートのポートシナリオの帯域を使用し，最優先のクラス 1 を割り当てています。

○：カウント対象，×：カウント対象外

ソフトウェアバージョン Ver.1.5.2 以降は，set ip system network port scenario コマンドで，従来のポートシナリオから最優先の転送，またはシステムインタフェースの通信に該当するフィルタとシナリオ（第 2 階層から第 8 階層）を設定している場合は，当該シナリオで制御とするかを選択可能です。デフォルト値は”disable”で従来通りの（ソフトウェアバージョン Ver.1.4.2 以前と同じ）転送動作です。トラフィック分析機能のトラフィック生成機能を使用する場合，同コマンドを”enable”にしてください。”enable”時は，システムインタフェースからの出力トラフィックは，出力ポートとは逆側ポートのフィルタ条件に一致するシナリオの帯域を使用します。また，当該シナリオの入出力カウンタにも加算されます。システムインタフェースへの入力トラフィックは，シナリオの帯域は使用せず，シナリオカウンタは受信のみ加算されます。なお，フィルタ条件に一致しない場合は，従来通りの転送動作です。



システムインタフェース	シナリオカウンタ		シナリオ動作
	受信	送信	
入力トラフィック	○	×	シナリオの帯域を使用しません。
出力トラフィック	○	○	出力ポートとは逆側ポートのフィルタ条件に一致するシナリオの帯域を使用します。 フィルタ条件に一致しない場合，出力ポートとは逆側ポートのポートシナリオの帯域を使用し，最優先のクラス 1 を割り当てています。

○：カウント対象，×：カウント対象外

システムインタフェースの設定には以下のコマンドを使用します。

set ip system <IP_address> netmask <netmask> [{up | down}]	システムインタフェースの IP アドレスを設定します。 IPv4 アドレスのデフォルト値は 192.168.1.1 です。サブネットマスクのデフォルト値は 255.255.255.0 です。 IPv6 アドレスのデフォルト値は::192.168.1.1（::C0A8:101）です。プレフィックス長のデフォルト値は 64 です。
set ip system port ethernet set ip system port network in {<slot/port> | all} vid {<VID> | none} [tpid <tpid>] inner-vid {<VID> | none} [inner-tpid <tpid>]	システムインタフェースの通信ポート（Ethernet ポート／Network ポート）を設定します。 また，システムインタフェースへの通信ポートとして Network ポートを指定した場合は，以下の内容も設定します。 - Network ポート識別番号（1/1，1/2，all） - VLAN ID（0～4094／none），出力 Tag Prptocol ID - Inner-VLAN ID（0～4094／none），出力 Tag Prptocol ID Network ポート識別番号のデフォルト値は“all”（すべての Network ポート）です。VLAN ID および Inner-VLAN ID のデフォルト値は“none”（VLAN Tag なしパケット通信）です。出力 Tag Protocol ID は VLAN Tag ありパケットの通信または二重 VLAN Tag ありパケットの通信を行う場合で，システムインタフェースが送信するパケットの Tag Protocol ID を指定するときに設定します。デフォルトではいずれも 0x8100 を使用します。 通信ポートのデフォルト値は Ethernet ポートです。
set ip system port network scenario {enable | disable}	システムインタフェースの通信ポート設定が Network ポート経由の場合，シナリオによるトラフィックコントロールの有効／無効を設定します。
set ip system gateway <gateway>	システムインタフェースのデフォルトゲートウェイアドレスを設定します。
unset ip system gateway <gateway>	システムインタフェースのデフォルトゲートウェイアドレスを解除します。
show ip system	システムインタフェース情報を表示します。

システムインタフェースに IPv4 アドレス（192.168.10.3），サブネットマスク（255.255.255.0），デフォルトゲートウェイ（192.168.10.1）を設定する場合，以下に示すコマンドを実行します。

```
PureFlow(A)> set ip system 192.168.10.3 netmask 255.255.255.0 up
PureFlow(A)> set ip system port ethernet
PureFlow(A)> set ip system gateway 192.168.10.1
```

システムインタフェースに IPv4 アドレス（192.168.10.3），サブネットマスク（255.255.255.0），通信ポート（Network ポート（1/1 のみ）），VLAN ID（10），二重 VLAN Tag なし，デフォルトゲートウェイ（192.168.10.1）を設定する場合，以下に示すコマンドを実行します。

```
PureFlow(A)> set ip system 192.168.10.3 netmask 255.255.255.0 up
PureFlow(A)> set ip system port network in 1/1 vid 10 inner-vid none
PureFlow(A)> set ip system gateway 192.168.10.1
```

IPv6を使用する場合でもIPv4の場合と同様に設定します。以下に示すコマンドを実行して，IPv6アドレス（2001:DB8::1），プレフィックス長（32），デフォルトゲートウェイ（2001:DB8::FE）を設定してください。
IPv6プレフィックス長はset ip systemコマンドのnetmask引数に指定します。

```
PureFlow(A)> set ip system 2001:db8::1 netmask 32 up
PureFlow(A)> set ip system gateway 2001:db8::fe
```

また，システムインタフェースでは以下のコマンドを使用して，ネットワークの疎通確認をすることができます。

ping <IP_address>	ICMP ECHO_REQUEST パケットを指定 IP アドレスに送信します。（IPv4／IPv6）
arp –a arp –d <IP_address>	ARP エントリの内容を表示（-a），または削除（-d）します。（IPv4 のみ）
delete ndp neighbor <IP_address>	NDP エントリの削除を行います。（IPv6 のみ）
show ndp neighbor	NDP エントリの内容を表示します。（IPv6 のみ）

IPv4 アドレス 192.168.10.100 との疎通確認を行う場合，以下に示すコマンドを実行します。

```
PureFlow(A)> ping 192.168.10.100
PING 192.168.10.100 (192.168.10.100) 56(84) bytes of data.
64 bytes from 192.168.10.100: icmp_seq=1 ttl=64 time=0.372 ms

--- 192.168.10.100 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.372/0.372/0.372/0.000 ms
PureFlow(A)> arp –a
IP address          MAC address                type
-------------------------------------------------------------------------------
192.168.10.3        00-00-91-01-11-23          permanent publish
192.168.10.100      00-00-91-01-23-45
PureFlow(A)>
```

疎通確認失敗時は，以下のように表示します。システムインタフェースの設定，およびネットワーク接続を確認してください。

```
PureFlow(A)> ping 192.168.10.101
PING 192.168.10.101 (192.168.10.101) 56(84) bytes of data.

--- 192.168.10.101 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 100ms
PureFlow(A)>
```

IPv4 アドレス 192.168.10.101 の ARP エントリを削除する場合，以下に示すコマンドを実行します。

```
PureFlow(A)> arp –d 192.168.10.100
PureFlow(A)> arp –a
IP address          MAC address                type
-------------------------------------------------------------------------------
192.168.10.3        00-00-91-01-11-23          permanent publish
PureFlow(A)>
```

IPv6 アドレス 2001:DB8::1 との疎通確認を行う場合，以下に示すコマンドを実行します。

```
PureFlow(A)> ping 2001:db8::1
PING 2001:db8::1 (2001:db8::1) 56(84) bytes of data.
64 bytes from 192.168.10.100: icmp_seq=1 ttl=64 time=0.372 ms

--- 2001:db8::1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.372/0.372/0.372/0.000 ms
PureFlow(A)> show ndp neighbor
IP address                    MAC address                type
-------------------------------------------------------------------------------------------
2001:db8::1                   00-00-91-01-23-45          reachable
PureFlow(A)>
```

疎通確認失敗時は，以下のように表示します。システムインタフェースの設定，およびネットワーク接続を確認してください。

```
PureFlow(A)> ping 2001:db8::10
PING 2001:db8::10 (2001:db8::10) 56(84) bytes of data.

--- 2001:db8::10 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 100ms
PureFlow(A)>
```

IPv6 アドレス 2001:db8::10 の NDP エントリを削除する場合，以下に示すコマンドを実行します。

```
PureFlow(A)> delete ndp neighbor 2001:db8::10
PureFlow(A)> show ndp neighbor
IP address                    MAC address                type
-------------------------------------------------------------------------------------------
PureFlow(A)>
```

7.3　システムインタフェースフィルタ

システムインタフェースへの通信を，ホストごとなどの単位で許可するか，拒否するかを選択することができます。

システムインタフェースへの通信を識別するルールは，システムフィルタにより定義します。IP パケットの以下のフィールド，およびその組み合わせで定義します。

- 送信元 IP アドレス
- 宛先 IP アドレス
- プロトコル番号
- 送信元ポート番号（Sport）
- 宛先ポート番号（Dport）

（注意）
ToS 値の指定が可能ですが，ToS 値によるフィルタリングは非サポートです。tos 指定を含むコマンドは受け付けますが，動作には反映されません

システムインタフェースフィルタの設定には以下のコマンドを使用します。

add ip system filter	システムインタフェースのフィルタを設定します。
delete ip system filter	システムインタフェースのフィルタを削除します。
show ip system	システムインタフェース情報を表示します。

システムインタフェースに IPv4 アドレス（192.168.10.3），サブネットマスク（255.255.255.0）を設定し，IPv4 アドレス（192.168.10.100）のパソコンからのみ装置にアクセスできるようにする場合は，以下に示すコマンドを実行します。

```
PureFlow(A)> set ip system 192.168.10.3 netmask 255.255.255.0 up
PureFlow(A)> set ip system gateway 192.168.10.1
PureFlow(A)> add ip system filter 20 sip 192.168.10.100 permit
PureFlow(A)> add ip system filter 30 deny
```

システムインタフェースフィルタをすべて解除する場合は，以下に示すコマンドを実行します。

```
PureFlow(A)> delete ip system filter all
```

システムインタフェースフィルタの 30 を解除する場合は，以下に示すコマンドを実行します。

```
PureFlow(A)> delete ip system filter 30
```

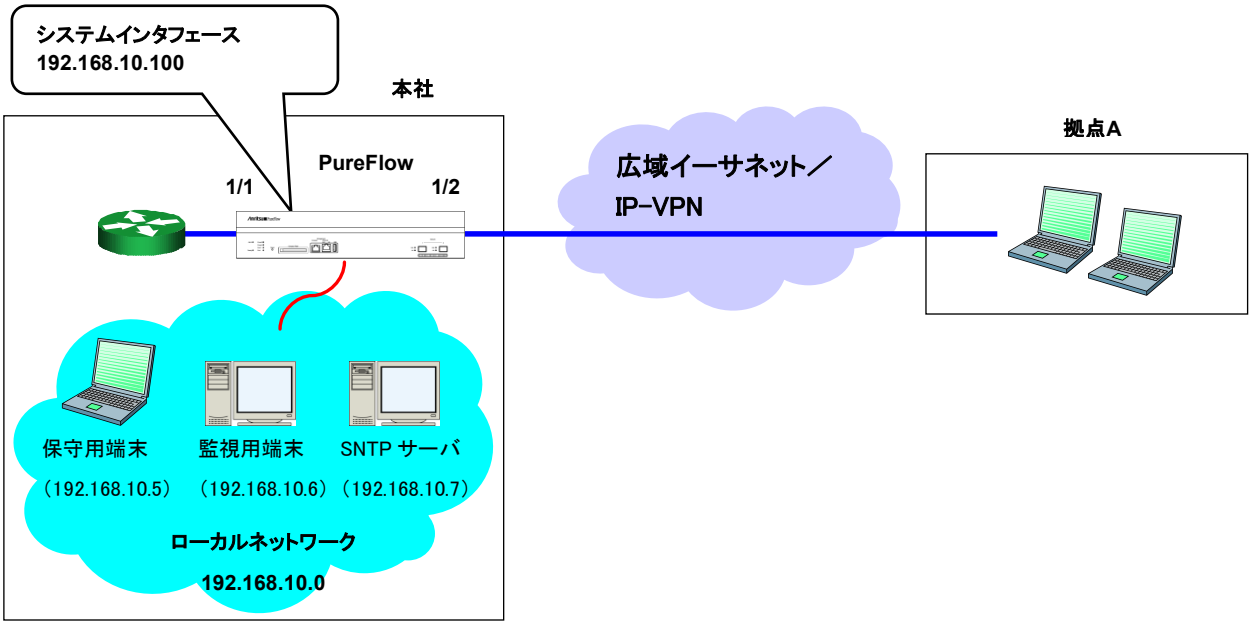
（注意）
システムインタフェースフィルタは十分に気をつけて設定してください。
機能を有効にする場合は permit を初めに設定し，そのあとに deny の設定を行ってください。機能を削除する場合は，deny を初めに削除し，そのあと permit の削除を行ってください。または，delete ip system filter all コマンドですべてのフィルタを削除してください。

7.4　コンフィギュレーション例

以下のネットワーク環境において，遠隔による保守／監視を行う場合のコンフィギュレーション例を示します。

[Case 1]ローカルネットワークから Ethernet ポートを経由して保守／監視を行う

- 本社内のローカルネットワークは 192.168.10.0/255.255.255.0 です。
- システムインタフェースの IPv4 アドレス 192.168.10.100，サブネットマスク 255.255.255.0 です。
- システムインタフェースのデフォルトゲートウェイアドレス 192.168.10.1 です。
- 保守用端末（CLI，ダウンロード／アップロード）の IPv4 アドレス 192.168.10.5 です。
- 監視用端末（SNMP，Syslog）の IPv4 アドレス 192.168.10.6 です。
- SNTP サーバの IPv4 アドレス 192.168.10.7 です。



以下のコマンドを実行します。

＜システムインタフェース設定＞

```
PureFlow(A)> set ip system 192.168.10.100 netmask 255.255.255.0 up
PureFlow(A)> set ip system gateway 192.168.10.1
```

＜SNMP ホスト設定＞

```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp community honsya_system_management view All
PureFlow(A)> add snmp host 192.168.10.6 version v2c
                community honsya_system_management trap
```

＜Syslog ホスト設定＞

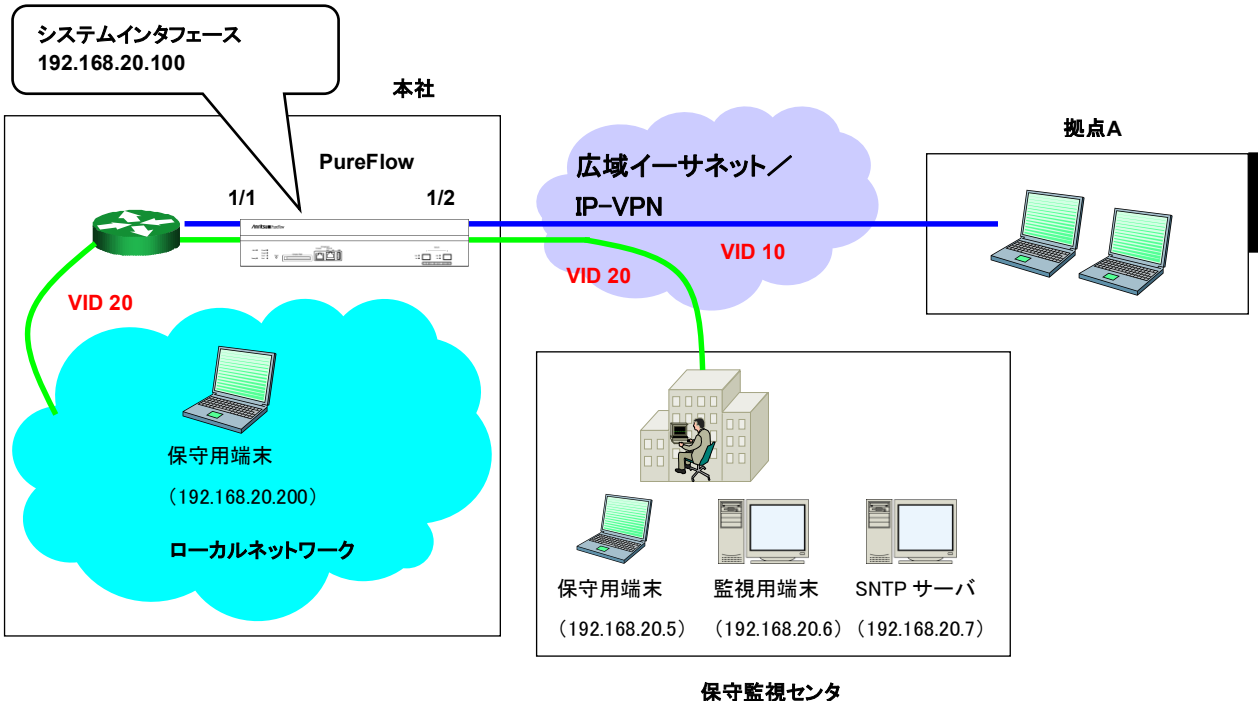
```
PureFlow(A)> add syslog host 192.168.10.6
PureFlow(A)> set syslog host enable
```

＜SNTP サーバ設定＞

```
PureFlow(A)> set sntp server 192.168.10.7
PureFlow(A)> set sntp enable
```

［Case 2］広域イーサネット／IP-VPNのネットワークとローカルネットワークから Networkポートを経由して保守／監視を行う（VLAN Tagありパケット通信）

- 拠点 A へのネットワークは VLAN ID 10 です。
- 保守監視センタへのネットワークは VLAN ID 20 です。
- システムインタフェースの IPv4 アドレス 192.168.20.100，サブネットマスク 255.255.255.0 です。
- システムインタフェースのデフォルトゲートウェイアドレス 192.168.20.1 です。
- すべての Network ポートからシステムインタフェースへの通信を行います。
- 保守用端末（CLI，ダウンロード／アップロード）の IPv4 アドレス 192.168.20.5，192.168.20.200 です。
- 監視用端末（SNMP，Syslog）の IPv4 アドレス 192.168.20.6 です。
- SNTP サーバの IPv4 アドレス 192.168.20.7 です。



以下のコマンドを実行します。

＜システムインタフェース設定＞

```
PureFlow(A)> set ip system 192.168.20.100 netmask 255.255.255.0 up
PureFlow(A)> set ip system port network in all vid 20 inner-vid none
PureFlow(A)> set ip system gateway 192.168.20.1
```

＜SNMP ホスト設定＞

```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp community honsya_system_management view All
PureFlow(A)> add snmp host 192.168.20.6 version v2c
                community honsya_system_management trap
```

＜Syslog ホスト設定＞

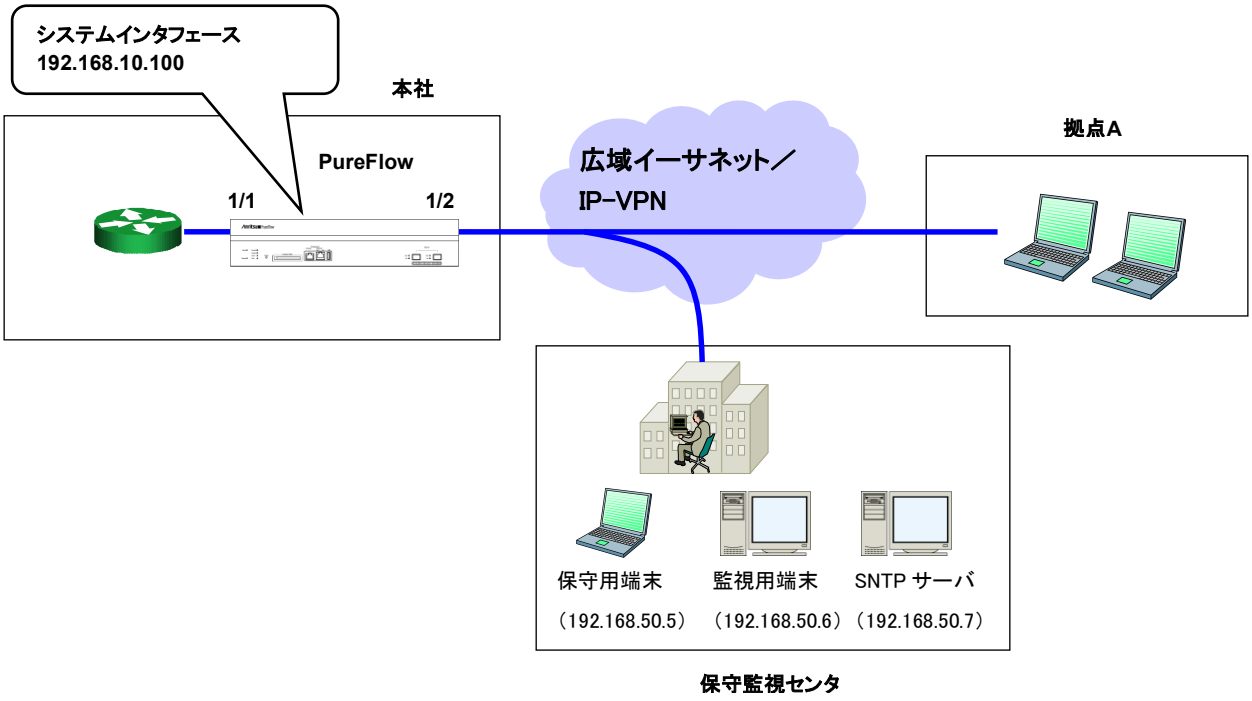
```
PureFlow(A)> set syslog host ip 192.168.20.6
PureFlow(A)> set syslog host enable
```

＜SNTP サーバ設定＞

```
PureFlow(A)> set sntp server 192.168.20.7
PureFlow(A)> set sntp enable
```

［Case 3］広域イーサネット／IP-VPNのネットワークからNetworkポートを経由して保守／監視を行う（VLAN Tagなしパケット通信）

- 拠点 A へのネットワークは 192.168.2.0/255.255.255.0 です。
- 保守監視センタへのネットワークは 192.168.50.0/255.255.255.0 です。
- システムインタフェースの IPv4 アドレス 192.168.10.100，サブネットマスク 255.255.255.0 です。
- システムインタフェースのデフォルトゲートウェイアドレス 192.168.10.1 です。
- Network ポート 1/2 からのみシステムインタフェースへの通信を行います。
- 保守用端末（CLI，ダウンロード／アップロード）の IPv4 アドレス 192.168.50.5 です。
- 監視用端末（SNMP，Syslog）の IPv4 アドレス 192.168.50.6 です。
- SNTP サーバの IPv4 アドレス 192.168.50.7 です。



以下のコマンドを実行します。

＜システムインタフェース設定＞

```
PureFlow(A)> set ip system 192.168.10.100 netmask 255.255.255.0 up
PureFlow(A)> set ip system port network in 1/2 vid none inner-vid none
PureFlow(A)> set ip system gateway 192.168.10.1
```

＜SNMP ホスト設定＞

```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp community honsya_system_management view All
PureFlow(A)> add snmp host 192.168.50.6 version v2c
                community honsya_system_management trap
```

＜Syslog ホスト設定＞

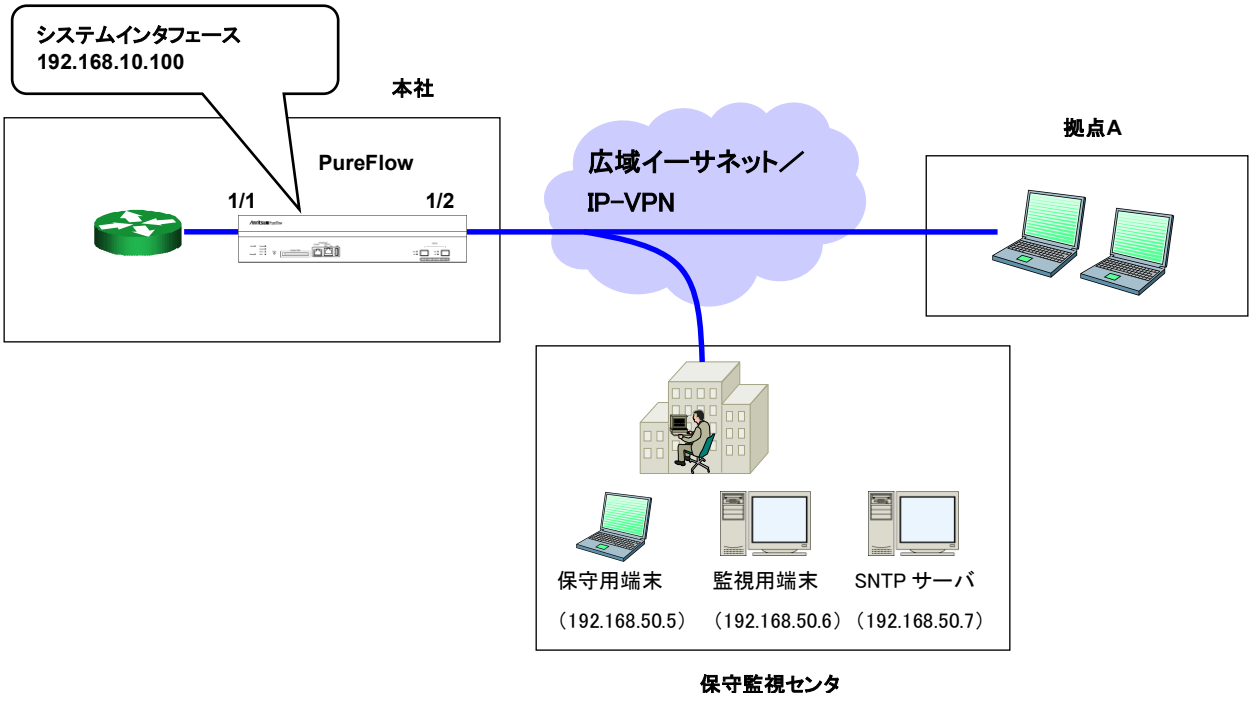
```
PureFlow(A)> set syslog host ip 192.168.50.6
PureFlow(A)> set syslog host enable
```

＜SNTP サーバ設定＞

```
PureFlow(A)> set sntp server 192.168.50.7
PureFlow(A)> set sntp enable
```

［Case 4］広域イーサネット／IP-VPNのネットワークからNetworkポートを経由して保守／監視を行う（特定ネットワークからのアクセスのみ許可）

- 拠点 A へのネットワークは 192.168.2.0/255.255.255.0 です。
- 保守監視センタへのネットワークは 192.168.50.0/255.255.255.0 です。
- システムインタフェースの IPv4 アドレス 192.168.10.100，サブネットマスク 255.255.255.0 です。
- システムインタフェースのデフォルトゲートウェイアドレス 192.168.10.1 です。
- Network ポート 1/2 からのみシステムインタフェースへの通信を行います。
- 保守用端末（CLI，ダウンロード／アップロード）の IPv4 アドレス 192.168.50.5 です。
- 監視用端末（SNMP，Syslog）の IPv4 アドレス 192.168.50.6 です。
- SNTP サーバの IPv4 アドレス 192.168.50.7 です。
- システムインタフェースへの通信は，保守監視センタからのみ許可します。



以下のコマンドを実行します。

＜システムインタフェース設定＞

```
PureFlow(A)> set ip system 192.168.10.100 netmask 255.255.255.0 up
PureFlow(A)> set ip system port network in 1/2 vid none inner-vid none
PureFlow(A)> set ip system gateway 192.168.10.1
```

＜システムインタフェースフィルタ設定＞

```
PureFlow(A)> add ip system filter 10 sip 192.168.50.0/255.255.255.0 permit
PureFlow(A)> add ip system filter 20 deny
```

＜SNMP ホスト設定＞

```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp community honsya_system_management view All
PureFlow(A)> add snmp host 192.168.50.6 version v2c
                community honsya_system_management trap
```

＜Syslog ホスト設定＞

PureFlow(A)> set syslog host ip 192.168.50.6

PureFlow(A)> set syslog host enable

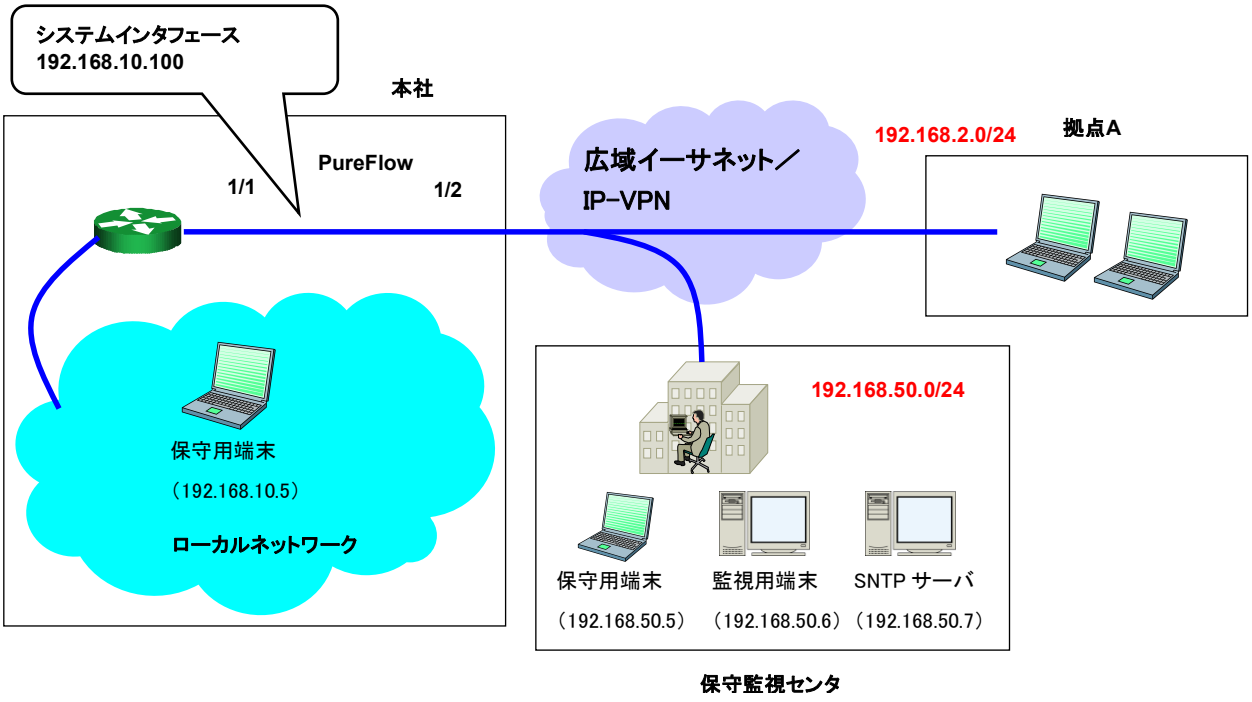
＜SNTP サーバ設定＞

PureFlow(A)> set sntp server 192.168.50.7

PureFlow(A)> set sntp enable

［Case 5］広域イーサネット／IP-VPN のネットワーク（Networkポート経由）とローカルネットワーク（Ethernetポート経由）から保守／監視を行う

- 本社内のローカルネットワークは 192.168.10.0/255.255.255.0 です。
- 拠点 A へのネットワークは 192.168.2.0/255.255.255.0 です。
- 保守監視センタへのネットワークは 192.168.50.0/255.255.255.0 です。
- システムインタフェースの IPv4 アドレス 192.168.10.100, サブネットマスク 255.255.255.0 です。
- システムインタフェースのデフォルトゲートウェイアドレス 192.168.10.1 です。
- 保守用端末（CLI, ダウンロード／アップロード）の IPv4 アドレス 192.168.50.5, 192.168.10.5 です。
- 監視用端末（SNMP, Syslog）の IPv4 アドレス 192.168.50.6 です。
- SNTP サーバの IPv4 アドレス 192.168.50.7 です。



以下のコマンドを実行します。

＜システムインタフェース設定＞

```
PureFlow(A)> set ip system 192.168.10.100 netmask 255.255.255.0 up
PureFlow(A)> set ip system gateway 192.168.10.1
```

＜SNMP ホスト設定＞

```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp community honsya_system_management view All
PureFlow(A)> add snmp host 192.168.50.6 version v2c
                community honsya_system_management trap
```

＜Syslog ホスト設定＞

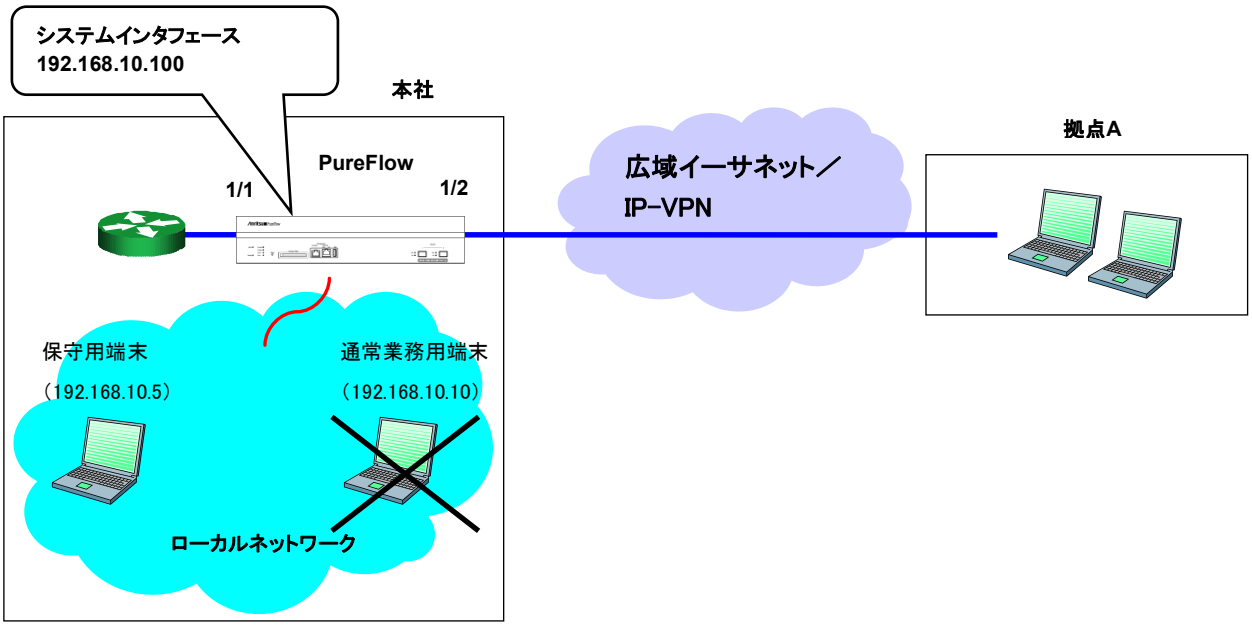
```
PureFlow(A)> set syslog host ip 192.168.50.6
PureFlow(A)> set syslog host enable
```

＜SNTP サーバ設定＞

```
PureFlow(A)> set sntp server 192.168.50.7
PureFlow(A)> set sntp enable
```

［Case 6］特定の端末から Ethernet ポートを経由して保守／監視を行う
不特定の端末からは監視を行わない

- 本社内のローカルネットワークは 192.168.10.0/255.255.255.0 です。
- システムインタフェースの IPv4 アドレス 192.168.10.100, サブネットマスク 255.255.255.0 です。
- システムインタフェースのデフォルトゲートウェイアドレス 192.168.10.1 です。
- 保守用端末（CLI, ダウンロード／アップロード）の IPv4 アドレス 192.168.10.5 です。
- 通常業務用端末の IPv4 アドレス 192.168.10.10 です。



以下のコマンドを実行します。

＜システムインタフェース設定＞

```
PureFlow(A)> set ip system 192.168.10.100 netmask 255.255.255.0 up
PureFlow(A)> set ip system gateway 192.168.10.1
```

＜システムインタフェースフィルタ設定＞

```
PureFlow(A)> add ip system filter 10 sip 192.168.10.5 permit
PureFlow(A)> add ip system filter 20 deny
```

7.5　設定，状態の確認

システムインタフェースの設定コマンドで設定した内容を確認するには，“show ip system”コマンドを使用します。

```
PureFlow(A)> show ip system
Status              : Up
IP Address          : 192.168.10.3
Netmask             : 255.255.255.0
Broadcast           : 192.168.10.255
Default Gateway     : 192.168.10.1
IPv6 Address        : 2001:DB8::1
Prefix              : 32
Default Gateway     : 2001:DB8::FE
Port                : Network (1/2)
VID                 : 20
TPID                : 0x8100
Inner-VID           : none
Inner-TPID          : ----

Number of system filter entries: 0
PureFlow(A)>
```

システムインタフェースの統計情報を確認するには，“show counter”コマンドを使用します。本コマンドで表示するカウンタ長は，32 ビットです。

```
PureFlow(A)> show counter
Port        Rcv Octets      Rcv Packets     Trs Octets      Trs Packets
-------     ------------    ------------    ------------    ------------
 1/1          57566366        14194297               0              0
 1/2                 0               0        59383412       14195494
system           58368             152           85424            152

Port        Rcv Broad       Rcv Multi       Trs Broad       Trs Multi
-------     ------------    ------------    ------------    ------------
 1/1             10000        14208097               0              0
 1/2                 0               0           10000       14209615
system             N/A             N/A             N/A            N/A

Port        Err Packets     Collision       Discard
-------     ------------    ------------    ------------
 1/1                 0               0              0
 1/2                 0               0              0
system             N/A             N/A            N/A
```

また，システムインタフェースをコマンド引数に指定することにより，詳細内容を表示できます。本コマンドで表示するカウンタ長は，64 ビットです。“show counter”コマンドの 32 ビットカウンタがラップアラウンドした場合，“show counter system”コマンドの 64 ビットカウンタと異なる値が表示されることに注意してください。

```
PureFlow(A)> show counter system
Rcv Packets                    152
Rcv Broad                      N/A
Rcv Multi                      N/A
Rcv Octets                   58368
Rcv Rate                       N/A
Trs Packets                    152
Trs Broad                      N/A
Trs Multi                      N/A
Trs Octets                   85424
Trs Rate                       N/A
Collision                      N/A
Drop                           N/A
Discard                        N/A
Error Packets                  N/A
        CRC Align Error                N/A
        Undersize Packet               N/A
        Oversize Packet                N/A
```

第8章　トラフィックコントロール機能

ここでは，トラフィックコントロール機能と設定について説明します。

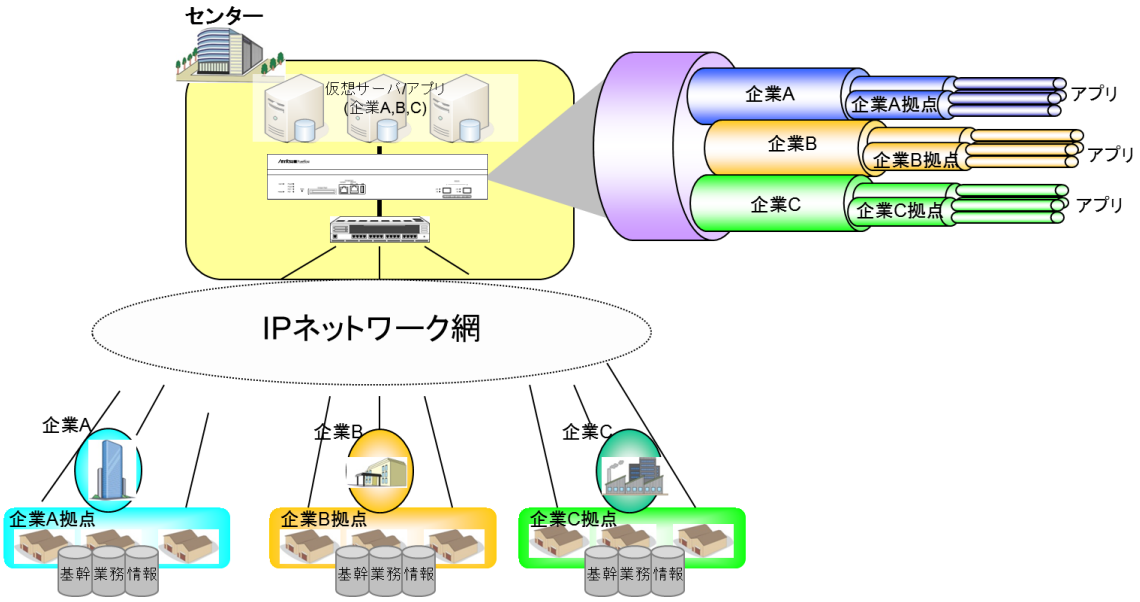
8.1　概要

PureFlow GSX は最大 10 GBit/s までの広帯域ネットワークでの帯域制御が可能です。多数拠点を持つ大規模な企業基幹ネットワークや，サービス事業者の大規模データセンタなどの利用に適しています。

企業の基幹システムネットワークにおける帯域制御は，拠点単位でグルーピングを行い帯域を確保，その配下でサービスやホストごとに帯域を保証もしくは制限する階層化制御が一般的です。サービス事業者のデータセンタでも同様に拠点単位，サービスホスト単位に階層化した帯域制御を行います。

クラウドサービスでは設備を構成するサーバ，ストレージ，ネットワークはすべて仮想化技術により企業で共有化されるため，拠点単位のグルーピングを施す前に企業ごとに帯域確保が必要になります。このように企業単位，拠点単位，サービスホスト単位と，制御を最大 8 レベルまで階層化することが可能です。性能，階層数をはじめ，キュー数，分類条件などについてもクラウド基盤に適したコンセプトとしております。

たとえば，まず企業ごとに帯域を確保した仮想回線を設定，その配下に企業各拠点ごとに帯域を確保した仮想回線を設定，さらにその配下にサービス，ホストに対し帯域保証，制限を設けるといった使い方になり，PureFlow GSX ではそれを容易に実現することができます。

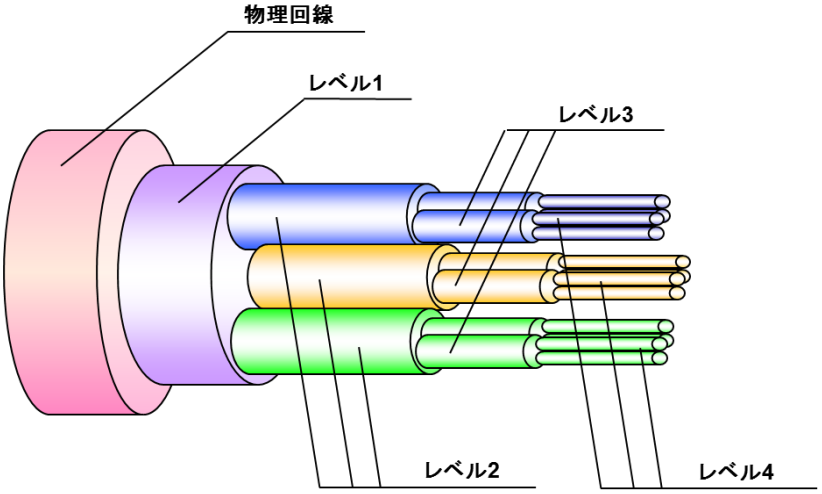


8.2　階層化シェーピング

階層化シェーピングとは，会社や拠点，ユーザといった任意のグループごとに回線帯域を分割し必要な帯域を割り当て，さらにその帯域内で任意のホスト，サービスごとに帯域を割り当てるといった階層構造のトラフィックコントロールを示します。

本装置では，階層ごとにシナリオとフィルタを追加することで，8 階層までの階層化シェーピングを行うことが可能です。第 1 階層（レベル 1）では，物理回線帯域を任意の帯域で制御（シェーピング）します。第 2 階層（レベル 2）では，会社や拠点，ユーザなどのトラフィックを分類し，レベル 2 の仮想回線に流すことで，仮想回線ごとに回線帯域を分割し，それぞれに個別の帯域を割り当てることができます。第 3 階層（レベル 3）以降でも同様に上位レベルに割り当てた帯域を分割し，制御することができます。

以下に，階層化シェーピングの概念図を示します。



レベル 1（第 1 階層）：

レベル 1 を通過する総帯域を制御（シェーピング）することができます。

レベル 1 は 1 つ，または複数のレベル 2 を集約できます。

レベル 2（第 2 階層）：

レベル 1 内の帯域を分割，制御します。

レベル 2 は 1 つ，または複数のレベル 3 を集約できます。

レベル 3（第 3 階層）：

レベル 2 内の帯域を分割，制御します。

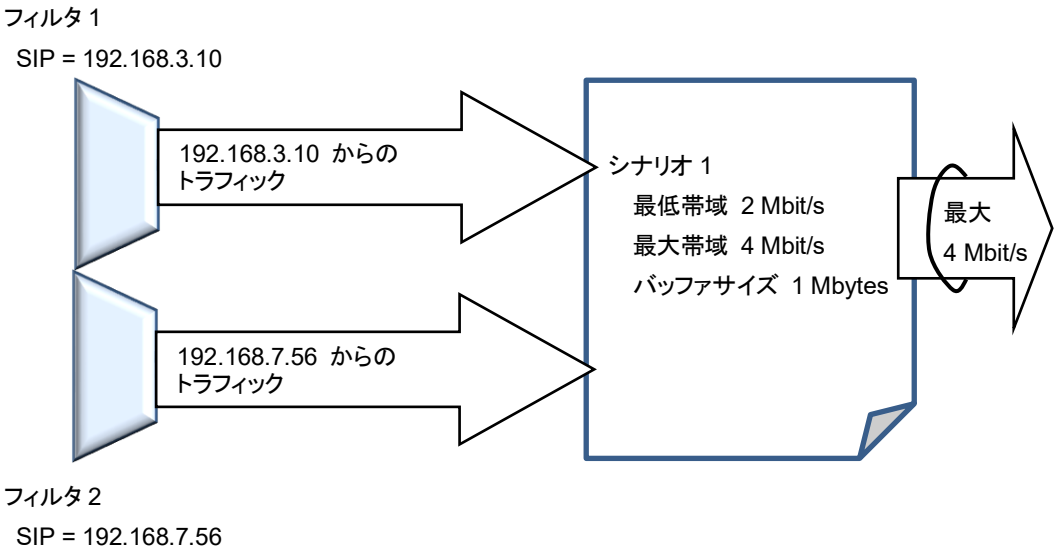
レベル 3 は 1 つ，または複数のレベル 4 を集約できます。

同様に，レベル 8（第 8 階層）まで帯域を分割，制御することが可能です。

8.3　フィルタとシナリオ

本装置は，通過するパケットをフィルタルールにより分類し，トラフィックを抽出します。フィルタルールにより分類されたトラフィックは，シナリオと呼ばれるトラフィックアトリビュート（最低帯域，最大帯域，バッファサイズ）に従ってトラフィックコントロールします。

複数のフィルタを同じシナリオに結び付けることが可能で，複数ユーザやアプリケーションで共有する帯域を割り当てることもできます。



SIP：送信元IPアドレス

8.3.1　フィルタ

パケットを分類するフィルタルールは，フィルタにより定義します。
Bridge-ctrl フレーム，Ethernet フレーム，IP パケットの以下のフィールド，およびその組み合わせで定義し，トラフィックを抽出します。

・Bridge-ctrl フレーム
　スイッチコントロール用特定宛先 MAC アドレスパケット（Bridge-ctrl フィルタ）

・Ethernet フレーム
　VLAN ID，CoS（二重 VLAN フレームの場合，双方についてフィルタ設定が可能です。）
　Ethernet ヘッダの length/type フィールド（Ethernet フィルタ）

・IPv4/IPv6 パケット（IP フィルタ）
　VLAN ID，CoS（二重 VLAN フレームの場合，双方についてフィルタ設定が可能です。）
　IPv4　　IP アドレス，プロトコル番号，ToS，ポート番号
　IPv6　　IP アドレス，プロトコル番号，トラフィッククラス，ポート番号

本装置は Tag Protocol ID 値が 0x8100 または 0x88A8 である VLAN Tag を認識します。IEEE 標準では Outer Tag に 0x88A8 を，Inner Tag に 0x8100 を使用しますが，本装置ではいずれの Tag においても 0x8100 または 0x88A8 を使用できます。

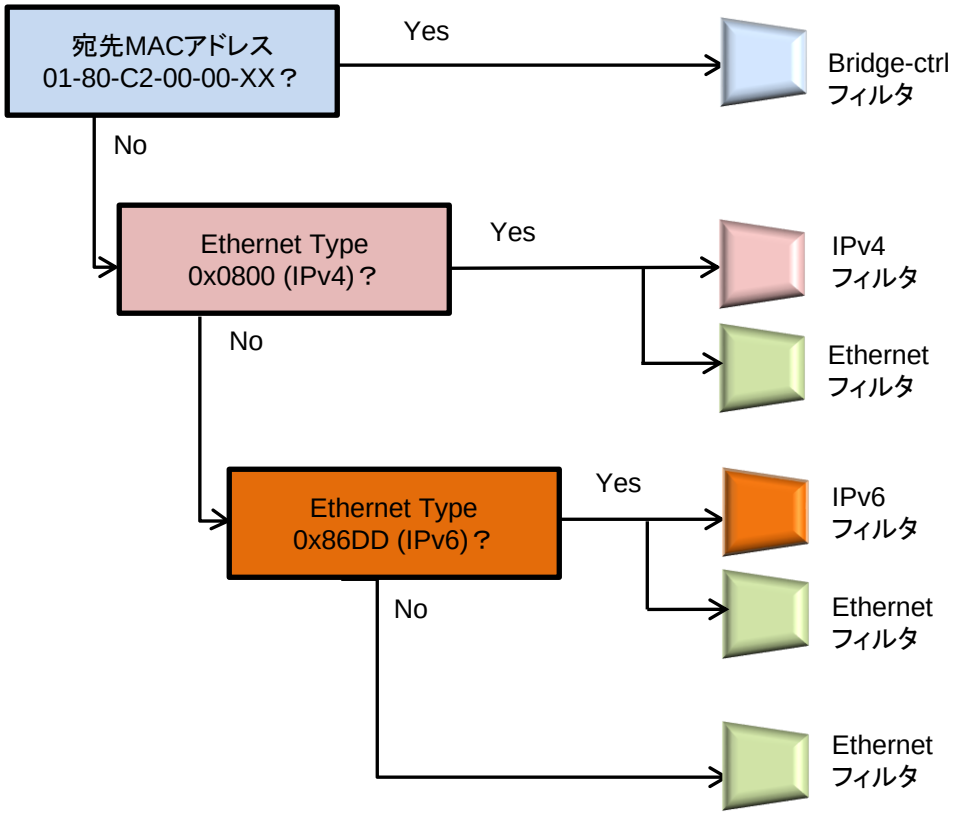
フィルタには，Bridge-ctrl フレームのみを分類する Bridge-ctrl フィルタ，VLAN Tag フィールド，Ethernet ヘッダの length/type フィールドを分類する Ethernet フィルタ，VLAN Tag フィールド，IP ヘッダ，TCP/UDP ヘッダを分類する IP フィルタの 3 種類があります。

1. Bridge-ctrl フィルタとは，スパニングツリープロトコルの BPDU やリンクアグリゲーションの LACP など，スイッチのコントロール用として予約されている MAC アドレスを対象としたフィルタです。
   たとえば，スパニングツリー構築環境下において BPDU を優先，もしくは帯域確保したい場合に使用します。
   対象となる宛先 MAC アドレスは以下です。
   - 宛先 MAC アドレス　01-80-C2-00-00-00～01-80-C2-00-00-FF

2. Ethernet フィルタとは，Ethernet フレーム全般を対象としたフィルタです。
   VLAN ごとに分類したい場合，パケット種別ごとに分類したい場合に使用します。
   たとえば，VLAN のみを指定することにより VLAN ごとの帯域制御が実現できます。
   また，ARP パケットを優先，もしくは帯域確保したい場合，Ethernet Type「0806」を指定することにより実現できます。

3. IP フィルタとは，IP パケットを対象にしたフィルタです。
   IP パケットフィールドにより IP パケットを分類したい場合に使用します。
   IP を IP フィルタにより分類する場合は，さらに以下の IP パケットフィールドの値を用いて細分化することができます。
   - VLAN ID（VLAN Tag あり／なしも識別）
   - CoS
   - 送信元 IP アドレス（SIP）
   - 宛先 IP アドレス（DIP）
   - ToS またはトラフィッククラス
   - プロトコル番号
   - 送信元ポート番号（Sport）
   - 宛先ポート番号（Dport）

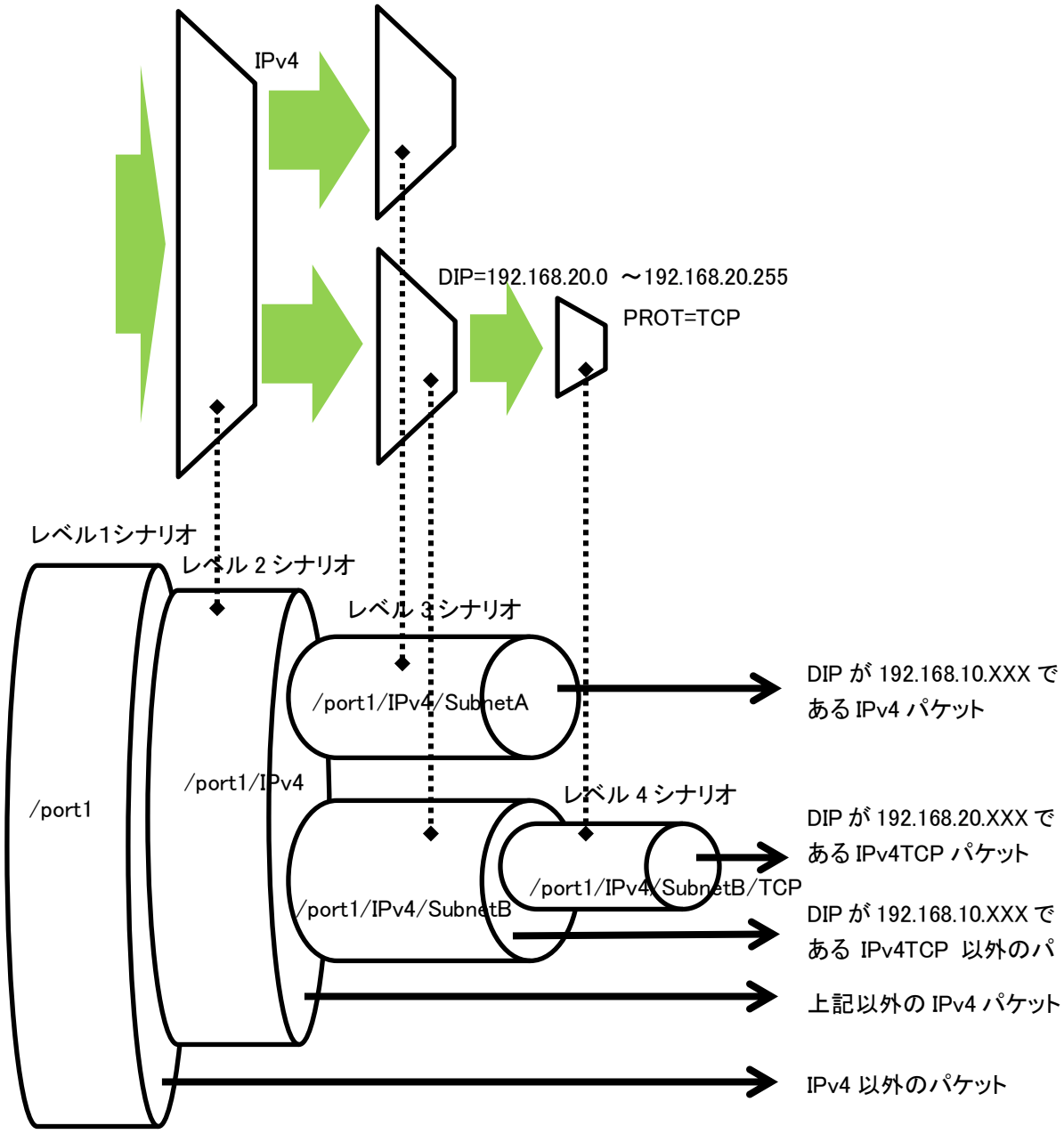
トラフィックをフィルタにより分類する際，適用するフィルタ種別はパケットの内容によって固定的です。宛先 MAC アドレスが 01-80-C2-00-00-XX であるフレームはそれ以外のフィールドの内容に関わらず Brdge-ctrl フィルタのみが適用されます。宛先 MAC アドレスが 01-80-C2-00-00-XX でなく，Ethernet Type 値が 0x0800 であるパケットは IPv4 フィルタおよび Ethernet フィルタのみが，0x86DD であるパケットは IPv6 フィルタおよび Ethernet フィルタのみが適用されます。上記いずれにも該当しないパケットは Ethernet フィルタのみが適用されます。



8.3.2　フィルタの階層関係

各シナリオのフィルタは，上位レベルシナリオのフィルタ条件を継承し，階層的にパケットを分類します。

上位レベルシナリオのフィルタ条件に一致し，下位レベルシナリオのフィルタ条件にも一致するトラフィックは，下位レベルシナリオのトラフィックとして分類します。上位レベルシナリオのフィルタ条件に一致し，下位レベルシナリオのフィルタ条件には一致しないトラフィックは，上位レベルシナリオのトラフィックとして分類され，上位レベルシナリオの空き帯域を使ってトラフィックを送出します。
以下の図は，パケットを階層的に分類した例です。レベル 2 シナリオのフィルタに，IPv4 を指定することにより IPv4 パケットと IPv4 以外のパケットを分類します。さらに，レベル 3 シナリオのフィルタに Subnet アドレスを指定することにより，SubnetA のパケットと SubnetB のパケットとそのほかの Subnet の IPv4 パケットに分類します。続いて，レベル 4 シナリオのフィルタに ProtocolTCP を指定することにより，SubnetB の TCP パケットと TCP 以外のパケットに分類します。



8.3.3　シナリオ

シナリオは，フィルタにより抽出されたトラフィックに対して，トラフィックコントロールを行うための設定（トラフィックアトリビュート）です。

シナリオには，トラフィックをひとつの固まりとしてコントロールする Aggregate（集約キューモード）と，トラフィック内をさらに個々のフロー（装置内で識別できるトラフィックの最小単位）ごとに識別し，各フローにトラフィックコントロールする Individual（個別キューモード）とフィルタに一致したトラフィックを上位階層のシナリオに転送する転送モード（Forward モード）があります。トラフィックは，複数のフローからなるグループと考えることができます（フローの詳細は「8.7.1 フロー識別モード」を参照してください）。

トラフィックアトリビュートは，レベル 1 からレベル 8 の各シナリオに対し設定する必要があり，指定できるパラメータは以下です。

-クラス　：シナリオの優先順位を指定します。同レベル内における優先度を設定することができます。クラス 1 が最優先とし以下クラス 2，3，4，5，6，7，8 の順となります。（クラスの詳細については「8.7.2 キュー」を参照してください。）

-保証帯域　：シナリオに割り当てる保証帯域です。同レベルの階層にトラフィックが流れている状態でも，この帯域を保証します。

-最大帯域　：シナリオに割り当てる最大帯域です。余剰帯域を設定値まで使用することができます。

-バッファサイズ　：シナリオに割り当てる許容入力バースト長です。
設定するシナリオ配下に下位シナリオが存在する場合，下位シナリオのフィルタ条件にマッチしたフローは下位シナリオで設定したバッファサイズが許容入力バースト長になります。
バッファサイズ以上パケットが滞留すると，パケットは廃棄されます。
（最適なバッファサイズを設定するには「12.2.4 シナリオパラメータ決定方法」を参照してください。）

個別キューモードのシナリオにはさらに以下のパラメータを指定できます。

-最大キュー数　：当該シナリオで生成する個別キューの最大数を指定します。
個別キューは全個別キューモードシナリオ合計で最大 300000 個です。

-キュー分割対象　：生成する個別キューの分割対象を指定します。フロー識別モードと同様にパケットのフィールドで指定します（フロー識別モードについては「8.7.1 フロー識別モード」を参照してください）。
指定されたフィールドを識別し，フィールドが異なるフローに個別のキューを割り当てます。

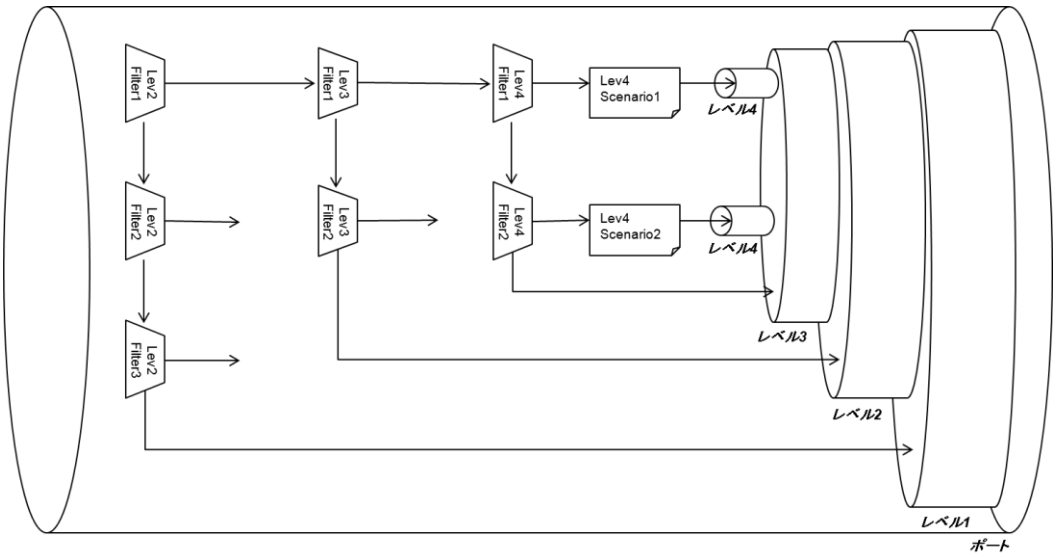
-キュー最大数超過アクション
：生成する個別キューが当該シナリオの最大キュー数，または全個別キューモードシナリオ合計で 300000 個を超えた場合，また，キュー分割対象に 5tuple（sip，dip，tos，proto，sport，dport）が含まれている場合の IP 以外のフローに適用する動作を指定します。

discard	廃棄します
forwardbesteffort	ベストエフォート（クラス 8）転送します
forwardattribute	トラフィックアトリビュートを指定して転送します

-キュー最大数超過時の最低帯域
-キュー最大数超過時の最大帯域
-キュー最大数超過時のクラス
：キュー最大数超過アクションに forwardattribute を指定した場合のトラフィックアトリビュートです。

8.3.4　フィルタとシナリオの関係

本装置は，物理回線内に流れるパケットをフィルタで分類し，トラフィックを抽出します。抽出したトラフィックを帯域，バッファサイズなどのトラフィックアトリビュートに従ってトラフィックコントロール転送します。



上図は，フィルタとシナリオの設定と，実際のトラフィックコントロール動作の関係を示した概念図です。

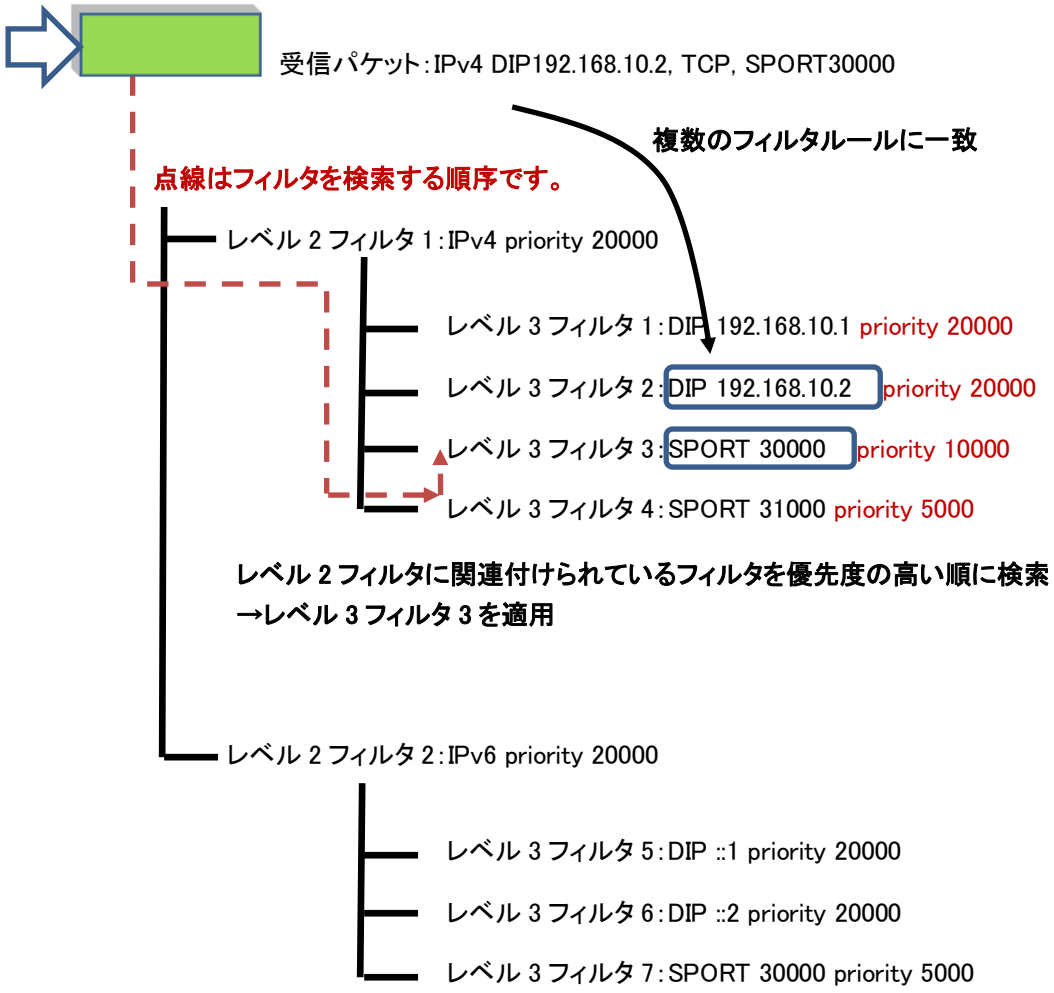
レベル 1 からレベル 8 での帯域制御，およびレベル 2 からレベル 8 でのフィルタ設定による廃棄，転送が制御できます。
フィルタの動作は，“aggregate”，“individual”，“discard” ，“foward”の指定が可能です。フィルタルールに一致したパケットは，フィルタに設定した動作に従います。
装置はパケットを受信すると，レベル 2 のフィルタにてフィルタ優先度の高い順からフィルタルールに一致するかどうか調べます。

レベル 2 フィルタルールに一致すると，フィルタに関連付けされているレベル 2 シナリオの動作が“aggregate”ならばそのシナリオで指定されたトラフィックアトリビュートに従ってパケットを転送します。また，そのシナリオに関連付けされているレベル 3 シナリオのレベル 3 フィルタにてフィルタ優先度の高い順からフィルタルールに一致するかどうか調べます。
“individual”ならばそのシナリオで指定されたトラフィックアトリビュートに従ってパケットを転送します。“individual”シナリオの下位レベルにシナリオおよびフィルタを登録することは可能ですが，“individual”シナリオより下位レベルのフィルタ検索は行いません。“individual”シナリオより下位レベルのシナリオでパケットが転送されることはなく，無効となります。
“discard”の場合，パケットを廃棄します。“discard”シナリオの下位レベルにシナリオおよびフィルタを登録することは可能ですが，“discard”シナリオより下位レベルのフィルタ検索は行いません。“individual”シナリオより下位レベルのシナリオでパケットが転送されることはなく，無効となります。
“forward”の場合，上位階層のシナリオに転送します。

レベル 3 から 8 までのフィルタに関しても同様となります。

フィルタには，Bridge-ctrl フィルタ，Ethernet フィルタ，IP フィルタの指定が可能です。各フィルタとも任意の文字列でフィルタ名を指定します。全フィルタ合計で 40000 個のフィルタルールを設定することができます。

また，各フィルタルールには優先度を設定することができます。
シナリオに関連付けされた同一レベルのフィルタ群のうち，一致するフィルタルールが複数ある場合，どのフィルタが適用されるかをフィルタ優先度に従って決定します。フィルタ優先度は値が小さいほど優先度が高くなります。



なお，一致するフィルタルールの優先度が同じである場合，どのフィルタが適用されるかは任意となります。
複数のフィルタルールに一致するようなフィルタ構成を行う場合，フィルタ優先度を調整して適用されるフィルタを明確にすることを推奨します。フィルタ優先度を指定しない場合，優先度 20000 が自動的に設定されます。

8.3.5　ルールリスト

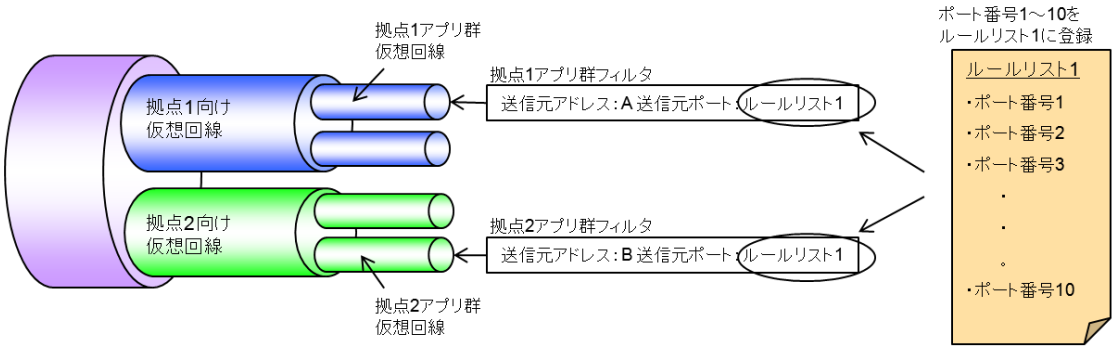
ルールリストは，複数のトラフィック分類条件（IP アドレス，ポート番号，ドメイン名[注 1]）をグループ化する機能です。これにより，複数のトラフィック分類条件を単一のルールリスト名で指定することができます。

ルールリスト名をフィルタ追加コマンドの引数に指定することで，トラフィック分類条件として設定することができます。

ルールリストに指定可能なトラフィック分類条件は，以下の通りです。

① IPv4 アドレス　　：IP アドレスとビットマスク
② IPv6 アドレス　　：IP アドレスとビットマスク
③ L4 ポート番号　　：ポート番号範囲
④ ドメイン　　：ドメイン名

ルールリストは複数のフィルタで繰り返し指定できます。ルールリストを使用することでフィルタ数や設定行数を削減できます。

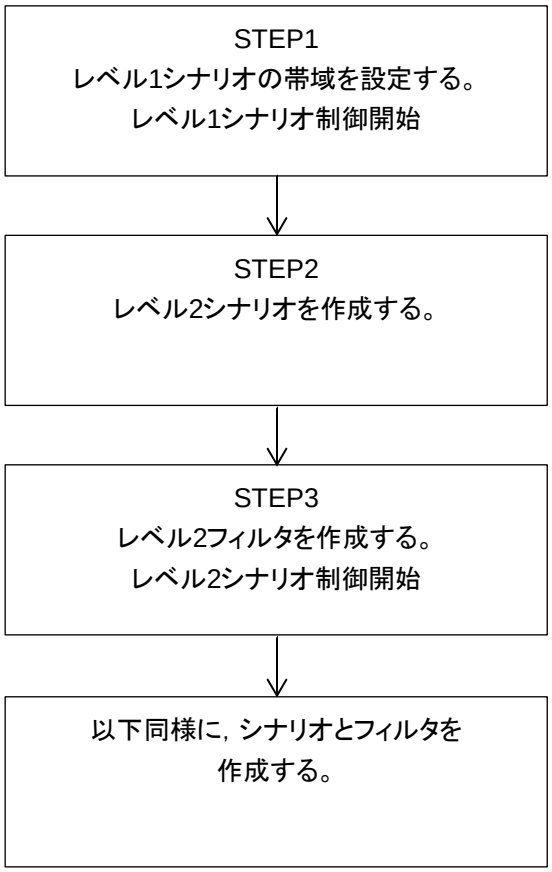


上図は，ルールリストの設定と，実際のトラフィックコントロール動作の関係を示した概念図です。この概念図では，ルールリスト 1 に，複数の TCP/UDP ポート番号を登録しておき，拠点 1 アプリ群仮想回線と拠点 2 アプリ群仮想回線のフィルタ設定コマンドにおいて，sport（送信元ポート番号）のパラメータとして利用しています。

（注 1）
ルールリストにドメインを指定する場合は，「ドメインフィルタ機能ライセンス(NF7101-L006A)」を購入していただく必要があります。

8.4　設定方法

設定方法の流れをまとめると下図のようになります。



次に，流れに沿って設定方法を説明します。

STEP 1： レベル1シナリオの帯域設定

本装置は，シナリオの設定により，各仮想回線のトラフィックアトリビュートを割り当てます。
レベル 1 シナリオに設定できるパラメータを以下に示します。

パラメータ	設定範囲	省略可能／不可	説明
シナリオ名 （scenario_name）	“/port1”または“/port2”	不可	省略，変更不可
最低帯域 （min_bandwidth）	0，1 k[bit/s]～10 G[bit/s]	可能	省略時および 0:最低帯域保証なし（最小値の確保） ※ 設定は可能ですが動作に影響しません。
最大帯域 （peak_bandwidth）	2 k[bit/s]～10 G[bit/s]	可能	省略時:1 G[bit/s]
バッファサイズ （bufsize）	2 k[Byte]～100 M[Byte]	可能	省略時:10 M[Byte]

以下にレベル 1 シナリオに関する CLI コマンドを示します。

update scenario <scenario_name> action aggregate [min_bw <min_bandwidth>] [peak_bw <peak_bandwidth>] [class <class>] [bufsize <bufsize>]	Network ポートから送出するトラフィックのトラフィックアトリビュートを変更することができます。レベル 1 シナリオ変更の場合，<scenario_name> に“ /port1 ”または“/port2”を指定します。 レベル 1 シナリオのクラスは変更できません。

レベル 1 シナリオ（“/port1”または“/port2”）はデフォルトで設定されているため，追加・削除することはできません。レベル 1 シナリオ（“/port1”または“/port2”）は，“update scenario”コマンドにより，パラメータを更新することができます。

以下に，レベル 1 シナリオの設定例を示します。

Sample 1） ポート 1 からポート 2 へのトラフィックを最大帯域 5 Gbit/s に設定する場合

```
PureFlow(A)> update scenario “/port1” action aggregate peak_bw 5G
```

Sample 2） ポート 2 からポート 1 へのトラフィックを最大帯域 3 Gbit/s に設定する場合

```
PureFlow(A)> update scenario “/port2” action aggregate peak_bw 3G
```

STEP 2： レベル2以降のシナリオの帯域設定

本装置は，シナリオの設定により，各仮想回線のトラフィックアトリビュートを割り当てます。
レベル 2 以降のシナリオに設定できるパラメータを以下に示します。

パラメータ	設定範囲	省略可能／不可	説明
シナリオ名 （scenario_name）	“/port1/xxxx”（2 階層） ”/port2/xxxx”（2 階層） “/port1/xxxx/xxxx”（3 階層） ”/port2/xxxx/xxxx”（3 階層） 以降同様に 8 階層まで	不可	update コマンドでの省略，変更不可 第 1 階層目には，Network ポートのポート番号を“/port1”または“/port2”のように指定し，第 2 階層以降に追加するシナリオ名を指定してください。設定範囲は全階層（/port1，/port2）を含めて 1～128 文字です。
アクションモード	aggregate：集約キューモード フィルタに一致したすべてのトラフィックを 1 つのキューでトラフィックコントロールします。 individual：個別キューモード フィルタに一致したトラフィックを個別のキューでトラフィックコントロールします。 discard　：廃棄モード フィルタに一致したトラフィックを廃棄します。 forward　：転送モード フィルタに一致したトラフィックを上位階層のシナリオに転送します。	不可	update コマンドでの省略，変更不可。
クラス （class）	1～8	可能	省略時：2 1（高）⇔（低）8 集約キューモード，個別キューモードで有効
最低帯域 （min_bandwidth）	0，1 k[bit/s]～10 G[bit/s]	可能	省略時および 0：最低帯域保証なし（最小値の確保） 集約キューモード，個別キューモードで有効
最大帯域 （peak_bandwidth）	2 k[bit/s]～10 G[bit/s]	可能	省略時：最大帯域制限なし 集約キューモード，個別キューモードで有効
バッファサイズ （bufsize）	2 k[Byte]～100 M[Byte]	可能	省略時：1 M[Byte] 集約キューモード，個別キューモードで有効
シナリオインデックス	1～40000	可能	update コマンドでの変更不可。 省略時：自動付与 すべてのアクションモードで有効

パラメータ	設定範囲	省略可能／不可	説明
最大キュー数 (maxquenum)	1～300000	可能	省略時: シナリオ拡張ライセンスが無効の場合，4096 シナリオ拡張ライセンス 10k が有効の場合，10000 シナリオ拡張ライセンス 40k が有効の場合，300000 個別キューモードのみ有効
キュー分割対象 (quedivision)	default　: 5tuple（ sip ， dip ， tos ， proto，sport，dport）の組み合わせでキューを分割します。 vlan　: VLAN ID でキューを分割します。 cos　: CoS でキューを分割します。 inner-vlan : インナーVLAN ID でキューを分割します。 inner-cos : インナーCoS でキューを分割します。 ethertype: EthernetType/Length でキューを分割します。 sip　: 送信元 IP アドレスでキューを分割します。 dip　: 宛先 IP アドレスでキューを分割します。 tos　: ToS または Traffic Class でキューを分割します。 proto　: プロトコル番号でキューを分割します。 sport　: 送信元ポート番号でキューを分割します。 dport　:宛先ポート番号でキューを分割します。	可能	省略時:default 個別キューモードのみ有効
キュー最大数超過アクション (failaction)	discard　: 廃棄します。 forwardbesteffort : ベストエフォート(クラス 8)転送します。 forwardattribute :トラフィックアトリビュートを指定して転送します。	可能	省略時:forwardbesteffort 個別キューモードのみ有効
キュー最大数超過時の最低帯域 (fail_min_bw)	0，1 k[bit/s]～10 G[bit/s]	可能	省略時:最低帯域保証なし 個別キューモードで“forwardattribute”指定時のみ有効

パラメータ	設定範囲	省略可能／不可	説明
キュー最大数超過時の最大帯域（fail_peak_bw）	2 k[bit/s]～10 G[bit/s]	可能	省略時:最大帯域制限なし 個別キューモードで“forwardattribute”指定時のみ有効
キュー最大数超過時のクラス（fail_class）	1～8	可能	省略時:8 1(高)⇔(低)8 個別キューモードで“forwardattribute”指定時のみ有効

以下にレベル 2 以降のシナリオに関する CLI コマンドを示します。

add scenario <scenario_name> action discard [scenario <scenario_id>]	廃棄モードのシナリオを登録します。 シナリオインデックスは自動付与されるので，通常は設定する必要はありません。
add scenario <scenario_name> action aggregate [min_bw <min_bandwidth>] [peak_bw <peak_bandwidth>] [class <class>] [bufsize <bufsize>] [scenario <scenario_id>]	集約キューモードのシナリオを登録します。 帯域，バッファサイズなどのトラフィックコントロールを行うためのトラフィックアトリビュートを設定します。 シナリオインデックスは自動付与されるので，通常は設定する必要はありません。
add scenario <scenario_name> action individual [min_bw <min_bandwidth>] [peak_bw <peak_bandwidth>] [class <class>] [bufsize <bufsize>] [scenario <scenario_id>] [maxquenum <quenum>] [quedivision <field>] [failaction {discard | forwardbesteffort | forwardattribute}] [fail_min_bw <min_bandwidth>] [fail_peak_bw <peak_bandwidth>] [fail_class <class>]	個別キューモードのシナリオを登録します。 帯域，バッファサイズなどのトラフィックアトリビュートを設定します。 また，個別キューの最大数，キュー分割対象，個別キュー数超過時の動作を設定します。 シナリオインデックスは自動付与されるので，通常は設定する必要はありません。
add scenario <scenario_name> action forward [scenario <scenario_id>]	転送モードのシナリオを登録します。 シナリオインデックスは自動付与されるので，通常は設定する必要はありません。
update scenario <scenario_name> action aggregate [min_bw <min_bandwidth>] [peak_bw <peak_bandwidth>] [class <class>] [bufsize <bufsize>]	集約キューモードのシナリオを変更します。 本コマンドにより，トラフィックコントロールされている状態でトラフィックアトリビュートを変更できます。各パラメータは省略可能ですが，すべてを省略することはできません。変更したいパラメータを 1 つ以上指定してください。

	シナリオ名，アクションモード，シナリオインデックスは変更できません。
update scenario <scenario_name> action individual [min_bw <min_bandwidth>] [peak_bw <peak_bandwidth>] [class <class>] [bufsize <bufsize>] [maxquenum <quenum>] [quedivision <field>] [failaction {discard | forwardbesteffort | forwardattribute}] [fail_min_bw <min_bandwidth>] [fail_peak_bw <peak_bandwidth>] [fail_class <class>]	個別キューモードのシナリオを変更します。 本コマンドにより，トラフィックコントロールされている状態でトラフィックアトリビュートを変更できます。各パラメータは省略可能ですが，すべてを省略することはできません。 変更したいパラメータを 1 つ以上指定してください。 シナリオ名，アクションモード，シナリオインデックスは変更できません。

以下に, レベル 2 シナリオの設定例を示します。

Sample 1）ポート 1 から受信した“Tokyo”拠点への集約キューモードシナリオについて最大帯域を 3 Gbit/s に設定する場合

```
PureFlow(A)> add scenario “/port1/Tokyo” action aggregate peak_bw 3G
```

Sample 2）ポート1から受信した“Osaka”拠点への集約キューモードシナリオについて最大帯域 1 Gbit/s に設定変更する場合

```
PureFlow(A)> update scenario “/port1/Osaka” action aggregate peak_bw 1G
```

Sample 3）ポート 1 から受信した“Nagoya”拠点への個別キューモードシナリオについて最大帯域 500 kbit/s, 最大キュー数 20 個に設定する場合

```
PureFlow(A)> add scenario “/port1/Nagoya” action individual peak_bw 500k maxquenum 20
```

レベル 3 以降のシナリオについても同様に, シナリオ名で上位シナリオと階層を指定します。

Sample 4）“Tokyo”拠点配下の“Shinjuku”エリアを集約キューモードシナリオで登録し, 最大帯域を 100 Mbit/s に設定する場合

```
PureFlow(A)> add scenario “/port1/Tokyo/Shinjuku” action aggregate peak_bw 100M
```

STEP 3： レベル2以降のフィルタの設定

本装置は，Bridge-ctrl フレーム，Ethernet フレーム，IPv4 パケット，IPv6 パケットのトラフィックをフィルタにより識別します。

レベル 2 以降のフィルタに設定できるパラメータを，以下に示します。

パラメータ		設定範囲	省略可能／不可
フィルタ名		1～48 文字	不可
シナリオ名		全階層を含めて 1～128 文字 （“add scenario”コマンドで設定したもの）	不可
フィルタ種類		bridge-ctrl, ethernet, ipv4, ipv6	不可
イーサタイプ		Ethernet ヘッダ内 Type フィールド値指定 0x0000～0xFFFF	可能 Ethernet フィルタのみ有効
VLAN ID		IEEE802.1Q VLAN ID を指定 0～4094（範囲指定可能）， none（VLAN Tag なし）	可能
インナーVLAN ID		QinQ におけるインナーVLAN ID を指定 0～4094（範囲指定可能）， none（VLAN Tag なし）	可能
CoS		IEEE802.1Q VLAN 内 CoS を指定 0～7	可能
インナーCoS		QinQ におけるインナーVLAN 内 CoS を指定 0～7	可能
送信元 IP アドレス	IPv4	0.0.0.0～255.255.255.255 （範囲指定「start-end」可能） ルールリスト名	可能 IP フィルタのみ有効
	IPv6	0::0～FFFF:…:FFFF（小文字入力，範囲指定「start-end」可能） ルールリスト名	可能 IP フィルタのみ有効
宛先 IP アドレス	IPv4	0.0.0.0～255.255.255.255 （範囲指定「start-end」可能） ルールリスト名	可能 IP フィルタのみ有効
	IPv6	0::0～FFFF:…:FFFF（小文字入力，範囲指定「start-end」可能） ルールリスト名	可能 IP フィルタのみ有効
ToS，または Traffic Class	IPv4	0～255 （範囲指定「start-end」可能）	可能 IP フィルタのみ有効
	IPv6	0～255 （範囲指定「start-end」可能）	可能 IP フィルタのみ有効
プロトコル番号		0～255 （範囲指定「start-end」可能） （tcp，udp，icmp は文字入力可能）	可能 IP フィルタのみ有効
送信元ポート番号		0～65535 （範囲指定「start-end」可能） ルールリスト名	可能 IP フィルタのみ有効

パラメータ	設定範囲	省略可能／不可
宛先ポート番号	0～65535 （範囲指定「start-end」可能） ルールリスト名	可能 IP フィルタのみ有効
フィルタ優先度	1～40000	可能 省略時は 20000

以下にレベル 2 以降のフィルタに関する CLI コマンドを示します。

add filter scenario <scenario_name> filter <filter_name> bridge-ctrl [priority <filter_pri>]	宛先 MAC アドレスが 01-80-C2-00-00-00～01-80-C2-00-00-FF（スパニングツリープロトコル，リンクアグリゲーション，EAPoL（認証プロトコル）などを含む）であるフレームを識別します。
add filter scenario <scenario_name> filter <filter_name> ethernet [vid {<VID> | none}] [cos <user_priority>] [inner-vid {<VID> | none}] [inner-cos <user_priority>] [ethertype <type>] [priority <filter_pri>]	Ethernet ヘッダの length/type フィールドを対象としてフレームを識別します。また，VLAN タグ内の VLAN ID，CoS においても指定することができます。 各パラメータは省略可能ですが，すべてを省略することはできません。“priority”以外のパラメータを 1 つ以上指定してください。
add filter scenario <scenario_name> filter <filter_name> ipv4 [vid {<VID> | none}] [cos <user_priority>] [inner-vid {<VID> | none}] [inner-cos <user_priority>] [sip [list] {<src_IP_address> | <list_name>}] [dip [list] {<dst_IP_address> | <list_name>}] [tos <type_of_service>] [proto <protocol>] [sport [list] {<sport> | <list_name>}] [dport [list] {<dport> | <list_name>}] [priority <filter_pri>]	IPv4 パケットの IP アドレス，プロトコル番号，ポート番号などを対象としてパケットを識別します。また，VLAN タグ内の VLAN ID，CoS においても指定することができます。 各パラメータは省略可能です。すべてを省略した場合は，IPv4 パケットすべてとなります。
add filter scenario <scenario_name> filter <filter_name> ipv6 [vid {<VID> | none}] [cos <user_priority>] [inner-vid {<VID> | none}] [inner-cos <user_priority>] [sip [list] {<src_IP_address> | <list_name>}] [dip [list] {<dst_IP_address> | <list_name>}] [tos <type_of_service>] [proto <protocol>] [sport [list] {<sport> | <list_name>}] [dport [list] {<dport> | <list_name>}] [priority <filter_pri>]	IPv6 パケットの IP アドレス，プロトコル番号，ポート番号などを対象としてパケットを識別します。また，VLAN タグ内の VLAN ID，CoS においても指定することができます。 各パラメータは省略可能です。すべてを省略した場合は，IPv4 パケットすべてとなります。

以下に，レベル 2 フィルタの設定例を示します。

Sample 1）レベル 2 シナリオ“/port1/bpdu”に流すフィルタ条件として，BPDU を設定する場合。

```
PureFlow(A)> add filter scenario “/port1/bpdu” filter “bpdu” bridge-ctrl priority 1
```

Sample 2）レベル 2 シナリオ“/port1/arp”に流すフィルタ条件として，ARP を設定する場合。

```
PureFlow(A)> add filter scenario “/port1/arp” filter “arp” ethernet ethertype 0x0806
```

Sample 3）レベル 2 シナリオ“/port1/Tokyo”に流すフィルタ条件として，IPv4 における VLAN ID を“10”に設定する場合。

```
PureFlow(A)> add filter scenario “/port1/Tokyo” filter “Tokyo” ipv4 vid 10
```

Sample 4）レベル 2 シナリオ“/port1/Osaka”に流すフィルタ条件として，IPv6 における VLAN ID を“20”に設定する場合。

```
PureFlow(A)> add filter scenario “/port1/Osaka” filter “Osaka” ipv6 vid 20
```

レベル 3 以降のフィルタについても同様に，対象シナリオを指定してフィルタを設定します。

Sample 4）レベル 3 シナリオ“/port1/Tokyo/Shinjuku”に流すフィルタ条件として，IPv4 における送信元 IP アドレス“192.168.10.0 ～ 192.168.10.255”に設定する場合。

```
PureFlow(A)> add filter scenario “/port1/Tokyo/Shinjuku” filter “Shinjuku” ipv4
                sip 192.168.10.0-192.168.10.255
```

8.5　ルールリストの設定方法

本章ではルールリストの設定方法について説明します。

ルールリストを利用するには以下の手順で設定します。
手順 1 ルールリストを設定する。
手順 2 ルールリストに対してルールリストエントリを追加する。
手順 3 フィルタ追加コマンドにルールリストを指定する。

ルールリストおよびルールリストエントリのパラメータを，以下に示します。

ルールリストのパラメータ

パラメータ	設定範囲
ルールリスト名	1 文字-32 文字
ルールリストタイプ	ipv4, ipv6, l4port, domain

ルールリストエントリのパラメータ

パラメータ		設定範囲
ルールリスト名		登録済みのルールリスト名を指定する
ルールリストタイプ		ipv4, ipv6, l4port, domain
トラフィック分類条件	IPv4 アドレス	0.0.0.0 -255.255.255.255
	IPv6 アドレス	0::0 – FFFF:…:FFFF （小文字入力可能）
	TCP/UDP ポート番号	0 – 65535（範囲指定可能）
	ドメイン名	・使用可能文字: 1234567890 abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ -.* （「*」は，後方一致のワイルドカード） ・ドメインの長さ: 「.」「*」を含め，253 文字以内 ・ラベルの長さ: 63 文字以内 ・大文字小文字の区別なし

ルールリストの設定に関する CLI は以下のコマンドです。

add rulelist group <list_name> {ipv4 | ipv6 | l4port | domain }	ルールリストを追加します。 ipv4，ipv6，l4port，domain のいずれかを対象にします。
add rulelist entry <list_name> ipv4 <IP_address>	IPv4 アドレスをルールリストに追加します。
add rulelist entry <list_name> ipv6 <IP_address>	IPv6 アドレスをルールリストに追加します。
add rulelist entry <list_name> l4port <port>	l4port ポート番号をルールリストに追加します。
add rulelist entry <list_name> domain <domain_ name>	ドメイン名のルールリストエントリを登録します。
delete rulelist group {<list_name> | all}	ルールリストを削除します。 対象ルールリストに登録済のルールリストエントリも削除します。
delete rulelist entry <list_name> ipv4 <IP_address>	IPv4 アドレスをルールリストから削除します。
delete rulelist entry <list_name> ipv6 <IP_address>	IPv6 アドレスをルールリストから削除します。
delete rulelist entry <list_name> l4port <port>	l4port ポート番号をルールリストから削除します。
delete rulelist entry <list_name> domain <domain_ name>	ドメイン名のルールリストエントリを削除します。
show rulelist [<list_name>]	ルールリストを表示します。

ルールリストは，以下のルールに従って設定してください。

(1) 装置内で重複しないユニークなルールリスト名を設定してください。

(2) “delete rulelist group”コマンドは，フィルタに登録されていないルールリストに対してのみ行うことができます。

(3) ルールリスト名には，“all”は指定できません。

以下に，ルールリストのサンプルを設定するコマンド手順を示します。

手順 1）　**ルールリスト**“TVCservers”を登録する。

```
PureFlow（A）＞　add rulelist group “TVCservers” ipv4
```

手順 2）　**ルールリスト**“TVCservers”に，ルールリストエントリを追加する。

```
PureFlow（A）＞　add rulelist entry “TVCservers” ipv4 172.16.111.11
PureFlow（A）＞　add rulelist entry “TVCservers” ipv4 172.16.112.11
                                ・
                                ・　（リスト化するホスト IP の追加）
                                ・
```

手順 3）　フィルタ追加コマンドの sip にルールリスト名“TVCservers”を指定する。

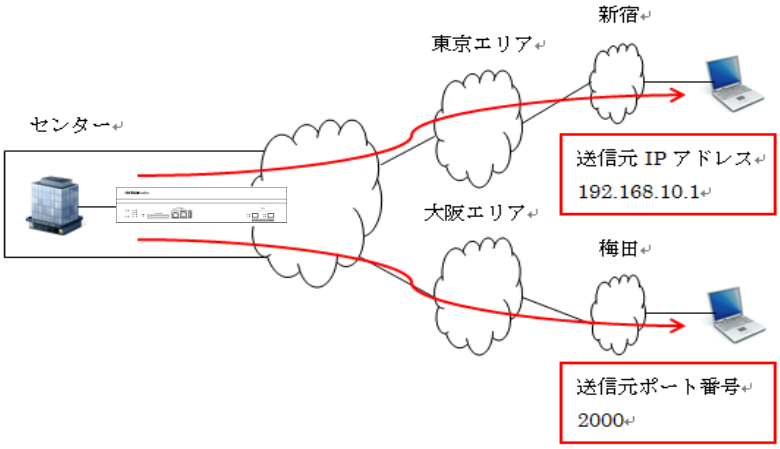
```
PureFlow（A）＞ add filter scenario “/port1/Tokyo/TVC” filter “TVC” ipv4 sip list
                “TVCservers”
```

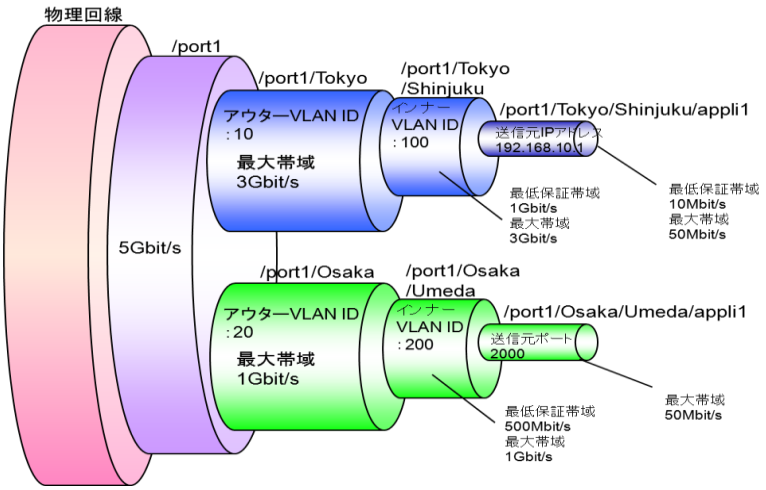
8.6　コンフィギュレーション例

以下のネットワーク環境の設定を行う場合のコンフィギュレーション例を示します。
レベル 2 にエリア，レベル 3 にネットワーク，レベル 4 にアプリを割り当てるケースです。

[Case 1]エリアおよびアプリの帯域確保を行う

- 東京のネットワークはアウターVLAN ID 10 です(レベル 2 フィルタの設定)。
- 大阪のネットワークはアウターVLAN ID 20 です(レベル 2 フィルタの設定)。
- 東京内新宿のネットワークはインナーVLAN ID 100 です(レベル 3 フィルタの設定)。
- 大阪内梅田のネットワークはインナーVLAN ID 200 です(レベル 3 フィルタの設定)。
- 新宿内制御対象アプリの送信元 IP アドレスは"192.168.10.1"です(レベル 4 フィルタの設定)。
- 梅田内制御対象アプリの送信元ポート番号は"2000"です(レベル 4 フィルタの設定)。
- GSX から送出する最大帯域を 5 Gbit/s とします(レベル 1 回線の設定)。
- センターから東京エリアへの最大帯域を 3 Gbit/s，大阪エリアへの最大帯域を 1 Gbit/s とします(レベル 2 シナリオの設定)。
- 東京エリア確保帯域 3 Gbit/s のうち，新宿エリアに対し最低保証帯域を 1 Gbit/s，最大帯域を 3 Gbit/s とします(レベル 3 シナリオの設定)。
- 大阪エリア確保帯域 1 Gbit/s のうち，梅田エリアに対し最低保証帯域を 500 Mbit/s，最大帯域を 1 Gbit/s とします(レベル 3 シナリオの設定)。
- 新宿エリア内対象アプリに対し，最低保証帯域を 10 Mbit/s，最大帯域を 50 Mbit/s とします(レベル 4 シナリオの設定)。
- 梅田エリア内対象アプリに対し，最大帯域を 50 Mbit/s とします(レベル 4 シナリオの設定)。





以下のコマンドを実行します。

＜レベル 1 シナリオ設定＞

```
PureFlow(A)> update scenario “/port1” action aggregate peak_bw 5G
```

＜レベル 2 シナリオ設定＞

```
PureFlow(A)> add scenario “/port1/Tokyo” action aggregate peak_bw 3G
PureFlow(A)> add scenario “/port1/Osaka” action aggregate peak_bw 1G
```

＜レベル 2 フィルタ設定＞

```
PureFlow(A)> add filter scenario “/port1/Tokyo” filter “Tokyo” ipv4 vid 10
PureFlow(A)> add filter scenario “/port1/Osaka” filter “Osaka” ipv4 vid 20
```

＜レベル 3 シナリオ設定＞

```
PureFlow(A)> add scenario “/port1/Tokyo/shinjuku” action aggregate min_bw 1G
                 peak_bw 3G
PureFlow(A)> add scenario “/port1/Osaka/Umeda” action aggregate min_bw
                 500M peak_bw 1G
```

＜レベル 3 フィルタ設定＞

```
PureFlow(A)> add filter scenario “/port1/Tokyo/Shinjuku” filter “Shinjuku” ipv4
                 inner-vid 100
PureFlow(A)> add filter scenario “/port1/Osaka/Umeda” filter “Umeda” ipv4 inner-vid 200
```

＜レベル 4 シナリオ設定＞

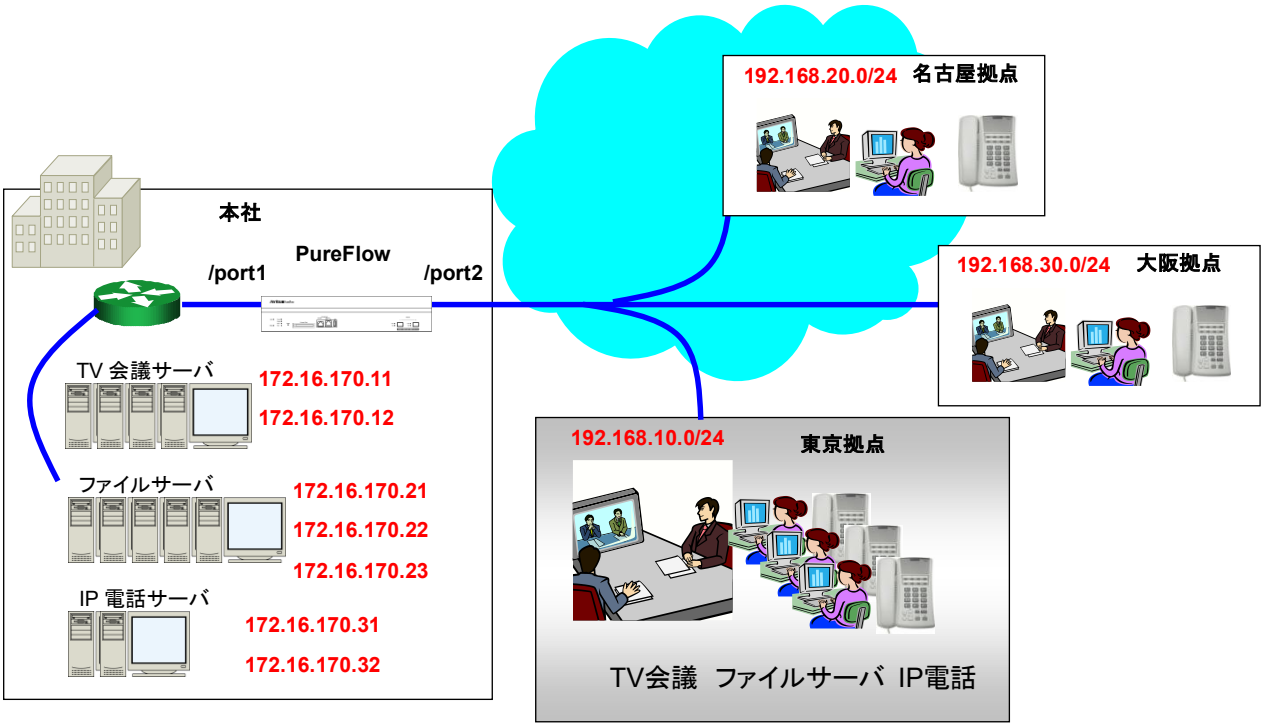
```
PureFlow(A)> add scenario “/port1/Tokyo/Shinjuku/appli1” action aggregate min_bw 10M
                 peak_bw 50M
PureFlow(A)> add scenario “/port1/Osaka/Umeda/appli1” action aggregate peak_bw 50M
```

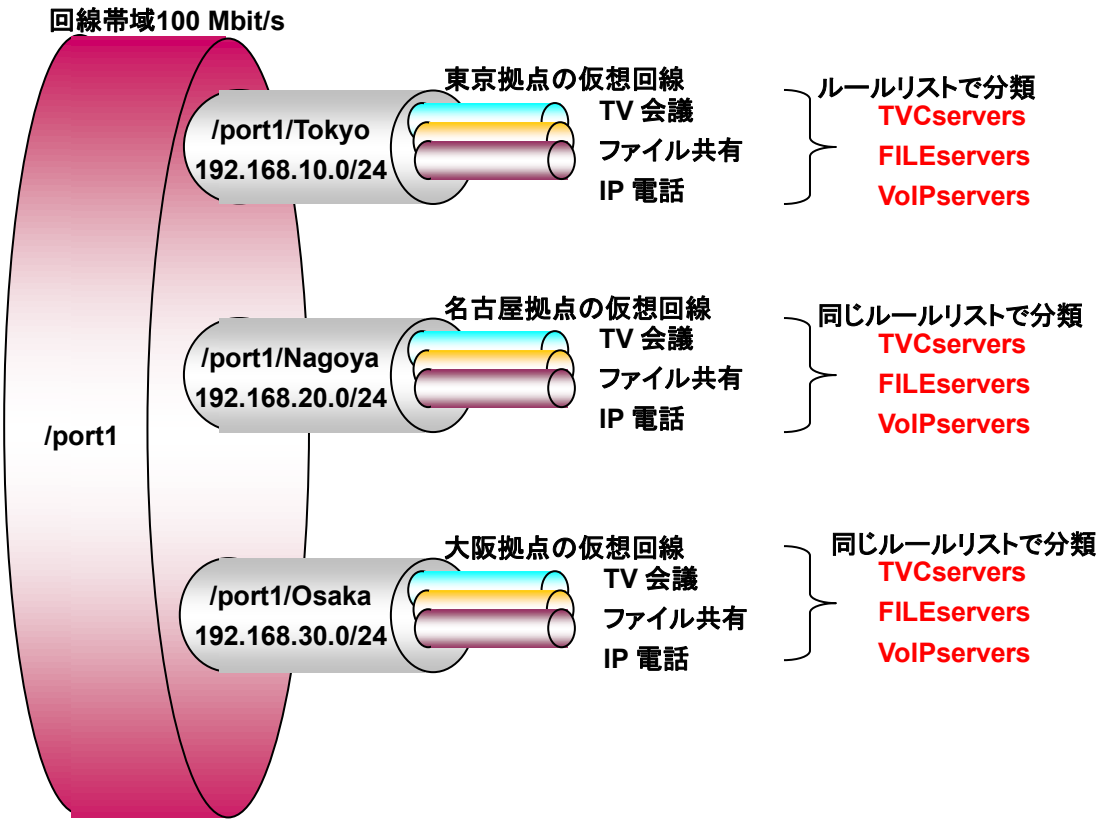
＜レベル 4 フィルタ設定＞

```
PureFlow(A)> add filter scenario “/port1/Tokyo/Shinjuku/appli1” filter “Shin_appli1” ipv4
                 sip 192.168.10.1
PureFlow(A)> add filter scenario “/port1/Osaka/Umeda/appli1” filter “Ume_appli1” ipv4
                 sport 2000
```

［Case 2］ルールリストを使って，フィルタ設定を簡素化する。

- 各拠点は，本社に設置されたサーバを利用している（TV 会議，ファイルサーバ，VoIP）。
- PureFlowGSX は，各拠点に，通信帯域を割り当て，さらにサービスごとに通信帯域を割り当てる。





＜サービスごとにルールリストに登録する＞

・ TV 会議サーバの IP アドレスをルールリストに登録する。

```
PureFlow(A)> add rulelist group “TVCservers” ipv4
PureFlow(A)> add rulelist entry “TVCservers” ipv4 172.16.170.11
PureFlow(A)> add rulelist entry “TVCservers” ipv4 172.16.170.12
```

・ ファイルサーバの IP アドレスをルールリストに登録する。

```
PureFlow(A)> add rulelist group “FILEservers” ipv4
PureFlow(A)> add rulelist entry “FILEservers” ipv4 172.16.170.21
PureFlow(A)> add rulelist entry “FILEservers” ipv4 172.16.170.22
PureFlow(A)> add rulelist entry “FILEservers” ipv4 172.16.170.23
```

・ IP 電話サーバの IP アドレスをルールリストに登録する。

```
PureFlow(A)> add rulelist group “VoIPservers” ipv4
PureFlow(A)> add rulelist entry “VoIPservers” ipv4 172.16.170.31
PureFlow(A)> add rulelist entry “VoIPservers” ipv4 172.16.170.32
```

＜東京拠点への仮想回線を登録する＞

・ 東京拠点へのトラフィック総量を設定する。

```
PureFlow(A)> add scenario “/port1/Tokyo” action aggregate peak_bw 10M
PureFlow(A)> add filter scenario “/port1/Tokyo” filter “Tokyo” ipv4 dip
                  192.168.10.0-192.168.10.255
```

・ TV 会議サーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Tokyo/TVC” action aggregate min_bw 5M
PureFlow(A)> add filter scenario “/port1/Tokyo/TVC” filter “Tokyo_TVC” ipv4
                  sip list “TVCservers”
```

・ ファイルサーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Tokyo/FILE” action aggregate min_bw 4M
PureFlow(A)> add filter scenario “/port1/Tokyo/FILE” filter “Tokyo_FILE” ipv4
                  sip list “FILEservers”
```

・ IP 電話サーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Tokyo/VoIP” action aggregate min_bw 1M
PureFlow(A)> add filter scenario “/port1/Tokyo/VoIP” filter “Tokyo_VoIP” ipv4
                  sip list “VoIPservers”
```

同じルールリストを使って，名古屋拠点と大阪拠点のトラフィックを登録します。

＜名古屋拠点への仮想回線を登録する＞

・ 名古屋拠点へのトラフィック総量を設定する。

```
PureFlow(A)> add scenario “/port1/Nagoya” action aggregate peak_bw 10M
PureFlow(A)> add filter scenario “/port1/Nagoya” filter “Nagoya” ipv4 dip
                192.168.20.0-192.168.20.255
```

・ TV 会議サーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Nagoya/TVC” action aggregate min_bw 5M
PureFlow(A)> add filter scenario “/port1/Nagoya/TVC” filter “Nagoya_TVC” ipv4
                sip list “TVCservers”
```

・ ファイルサーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Nagoya/FILE” action aggregate min_bw 4M
PureFlow(A)> add filter scenario “/port1/Nagoya/FILE” filter “Nagoya_FILE” ipv4
                sip list “FILEservers”
```

・ IP 電話サーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Nagoya/VoIP” action aggregate min_bw 1M
PureFlow(A)> add filter scenario “/port1/Nagoya/VoIP” filter “Nagoya_VoIP” ipv4
                sip list “VoIPservers”
```

＜大阪拠点への仮想回線を登録する＞

・ 大阪拠点へのトラフィック総量を設定する。

```
PureFlow(A)> add scenario “/port1/Osaka” action aggregate peak_bw 10M
PureFlow(A)> add filter scenario “/port1/Osaka” filter “Osaka” ipv4 dip
                192.168.30.0-192.168.30.255
```

・ TV 会議サーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Osaka/TVC” action aggregate min_bw 5M
PureFlow(A)> add filter scenario “/port1/Osaka/TVC” filter “Osaka_TVC” ipv4
                sip list “TVCservers”
```

・ ファイルサーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Osaka/FILE” action aggregate min_bw 4M
PureFlow(A)> add filter scenario “/port1/Osaka/FILE” filter “Osaka_FILE” ipv4
                sip list “FILEservers”
```

・ IP 電話サーバのルールリストを使ってトラフィックを登録する。

```
PureFlow(A)> add scenario “/port1/Osaka/VoIP” action aggregate min_bw 1M
PureFlow(A)> add filter scenario “/port1/Osaka/VoIP” filter “Osaka_VoIP” ipv4
                sip list “VoIPservers”
```

8.7　さらに高度な設定

本装置には，さらに高度な設定として，以下の設定があります。

- フロー識別モード
- キュー
- 通信ギャップモード

8.7.1　フロー識別モード

フローとは，装置内で識別できるトラフィックの最小単位です。トラフィックは，複数のフローからなるグループと考えることができます。

本装置は，パケットを受信すると，そのパケットを転送するためのフローを登録します。登録したフローは，フィルタに設定した動作に従ってキューにパケットを格納し，トラフィックコントロールします。

フローには，BridgeControl フロー，EthernetType フロー，IPv4 フロー，および IPv6 フローの 4 種類があります。

（1）BridgeControl フロー

BridgeControl フローは，Bridge-ctrl フィルタによって識別するフローです。宛先 MAC アドレスが 01-80-C2-00-00-00～01-80-C2-00-00-FF であるフレームを，入力ポートごとにひとつのフローに集約します。

（2）EthernetType フロー

EthernetType フローは，Ethernet フィルタによって識別するフローです。以下の Ethernet フィールドをフロー識別します。

- VLAN ID（VLAN Tag あり／なしも識別）
- CoS
- インナーVLAN ID
- インナーCoS
- Ethernet Type

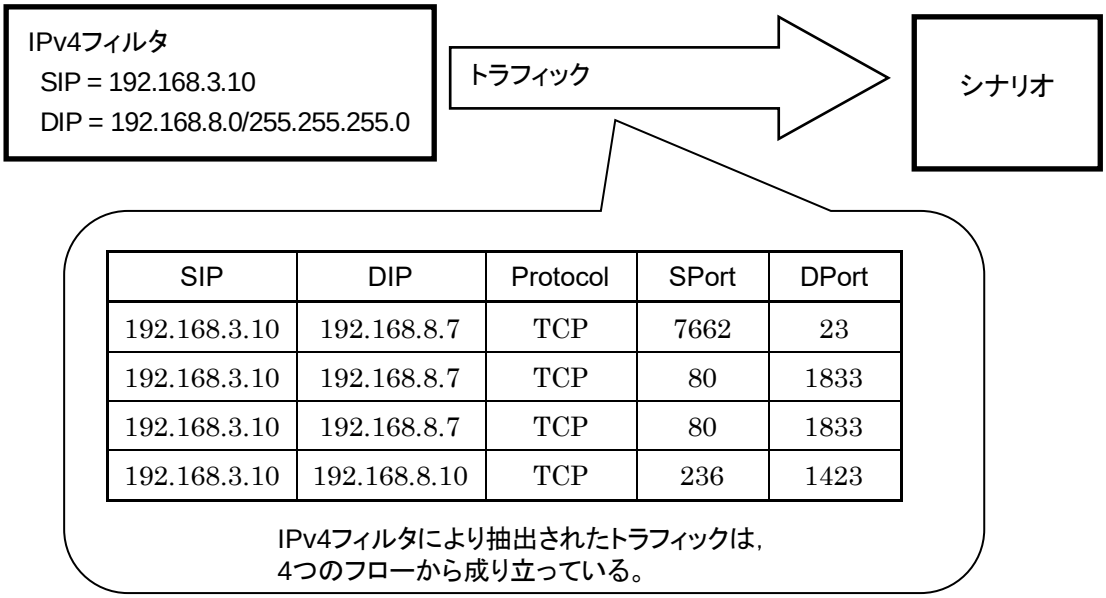
（3）IPv4/IPv6 フロー

IPv4/IPv6 フローは，IPv4/IPv6 フィルタによって識別するフローです。以下の IP パケットフィールドをフロー識別します。

- VLAN ID（VLAN Tag あり／なしも識別）
- CoS
- インナーVLAN ID
- インナーCoS
- 送信元 IP アドレス（SIP）
- 宛先 IP アドレス（DIP）
- ToS またはトラフィッククラス
- プロトコル番号
- 送信元ポート番号（Sport）
- 宛先ポート番号（Dport）

フロー識別モードを設定することにより，EthernetType フローおよび IPv4/IPv6 フローでフロー識別するフィールドを選択することができます。

たとえば，通常 IPv4 フローは，送信元 IP アドレス（SIP），宛先 IP アドレス（DIP），プロトコル番号（Protocol），送信元ポート番号（SPort），宛先ポート番号（DPort）がすべて一致するトラフィックです。



SIP	DIP	Protocol	SPort	DPort
192.168.3.10	192.168.8.7	TCP	7662	23
192.168.3.10	192.168.8.7	TCP	80	1833
192.168.3.10	192.168.8.7	TCP	80	1833
192.168.3.10	192.168.8.10	TCP	236	1423

IPv4フィルタにより抽出されたトラフィックは，
4つのフローから成り立っている。

本装置は，このフローを識別するフィールドの組み合わせ（フロー識別モード）を変更することが可能です。各フィールドが異なるパケットを異なるフローとして転送したり，同じフローとして転送したりすることができます。

フロー識別モードに設定できるパラメータを以下に示します。

パラメータ	設定範囲	省略可能／不可
入力 Network ポート	1/1／1/2	不可
フィールド名	default　：フローの識別フィールドをデフォルトにします。送信元 IP アドレス, 宛先 IP アドレス, プロトコル番号, 送信元ポート番号, 宛先ポート番号をフロー識別します。 vid　：VLAN ID（IEEE802.1q）または 2 重 VLAN タグ（IEEE802.1ad）の外側 VLAN ID をフロー識別します。 cos　：CoS（IEEE802.1q）または 2 重 VLAN タグ（IEEE802.1ad）の外側 CoS をフロー識別します。 inner-vid：2 重 VLAN タグの内側 VLAN ID をフロー識別します。 inner-cos：2 重 VLAN タグの内側 CoS をフロー識別します。 sip　：送信元 IP アドレスをフロー識別します。 dip　：宛先 IP アドレスをフロー識別します。 tos　：ToS または Traffic Class をフロー識別します。 proto　：プロトコル番号をフロー識別します。 sport　：送信元ポート番号をフロー識別します。 dport　：宛先ポート番号をフロー識別します。	不可

本パラメータは, カンマ(,)で区切って複数指定することができます。

フロー識別モードに関する CLI は以下のコマンドがあります。

set filter mode in <slot/port> <field>	フローの識別フィールドを選択します。<field> のデフォルト値は“default”です。

コマンドの実行例を示します。

```
PureFlow(A)> set filter mode in 1/1 cos
PureFlow(A)> set filter mode in 1/2 sip,dip
PureFlow(A)>
```

注:

装置内部には最大 1,280,000 フロー（BridgeControl フロー, EthernetType フロー, および IPv4/IPv6 フローの合計）を同時に作成, 帯域制御に用いることができます。フィルタルールによってパケット分類に用いないパラメータは, フロー識別モードから除外することで内部リソースの消費を抑えることができます。

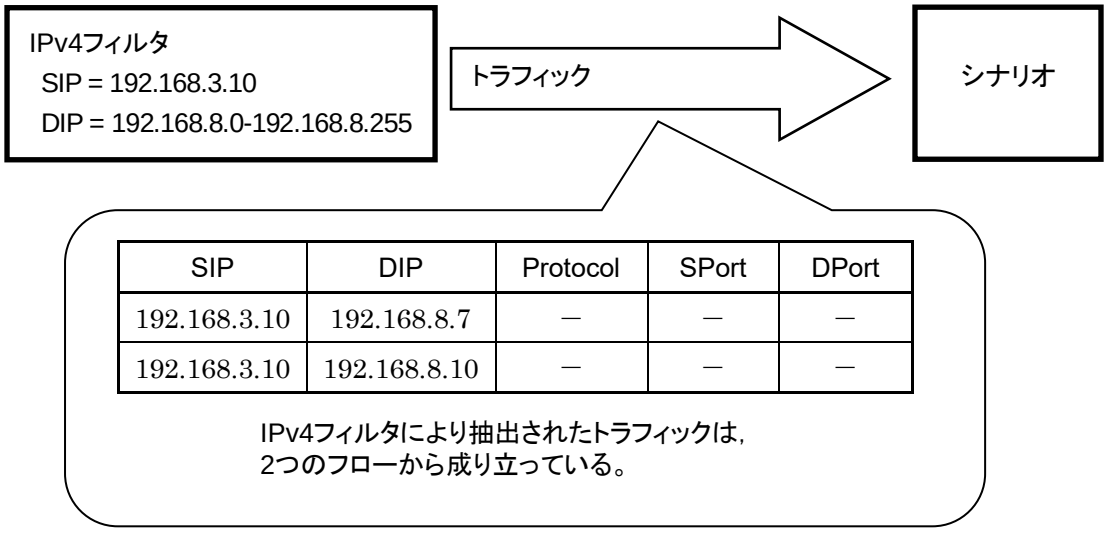
注:

BridgeControl フローは, フロー識別モードの設定にかかわらず, ポートに対して 1 つのみです。

注:

フラグメントパケットは, フラグメントされたすべてのパケットが本装置を経由するようにしてください。フラグメントパケットの先頭パケットが無い場合, フロー識別されないため, 後続パケットは転送されません。

たとえば，送信元 IP アドレスと宛先 IP アドレスのみでフローを識別し，その他のフィールドが異なる IPv4 パケットは同じ IPv4 フローとしてトラフィックコントロールしたい場合，sip と dip を有効にします。このフロー識別モードの場合，“add filter”で登録した IPv4 フィルタの条件は，送信元 IP アドレス，宛先 IP アドレスがフィルタ対象となります。フロー識別モードで指定したフィールド以外のフィールドが設定されている IPv4 フィルタは，無効と見なします。



指定フィールド名とフローで識別するフィールドの関係を，以下に示します。

指定フィールド名	フロー識別フィールド									
	VLAN ID	CoS	インナー VLAN ID	インナー CoS	SIP	DIP	ToS	プロトコル番号	Sport 番号	Dport 番号
default	×	×	×	×	○	○	×	○	○	○
vid	○	×	×	×	×	×	×	×	×	×
cos	×	○	×	×	×	×	×	×	×	×
inner-vid	×	×	○	×	×	×	×	×	×	×
inner-cos	×	×	×	○	×	×	×	×	×	×
sip	×	×	×	×	○	×	×	×	×	×
dip	×	×	×	×	×	○	×	×	×	×
tos	×	×	×	×	×	×	○	×	×	×
proto	×	×	×	×	×	×	×	○	×	×
sport	×	×	×	×	×	×	×	×	○	×
dport	×	×	×	×	×	×	×	×	×	○

○:フロー識別する
×:フロー識別しない

8.7.2　キュー

本装置は，各フローに対してキューを割り当て，受信したパケットを割り当てたキューに格納します。キューに格納したパケットはスケジューリングされ，トラフィックコントロール転送されます。

(1) デフォルトキュー

任意のレベル n シナリオ内で，それに属する下位のレベル n シナリオに該当しないフローを転送するためのキューです。デフォルトキューは，ベストエフォートクラス(クラス 8)となります。

任意のレベルフィルタに一致し，それに属する下位のレベルフィルタに一致しないすべてのフローを同じデフォルトキューに割り当て，トラフィックコントロールを行います。

たとえば，レベル 2 シナリオで保証帯域 100 Mbit/s に設定した場合は，以下のようになります。

本装置に，以下のフィルタを登録したと仮定します。

- レベル 2 フィルタ
  送信元 IP アドレス ： 192.168.0.0 - 192.168.255.255
  宛先 IP アドレス ： 192.168.0.0 - 192.168.255.255
- レベル 3 フィルタ
  送信元 IP アドレス ： 192.168.10.0 - 192.168.10.255
  宛先 IP アドレス ： 192.168.10.0 - 192.168.10.255

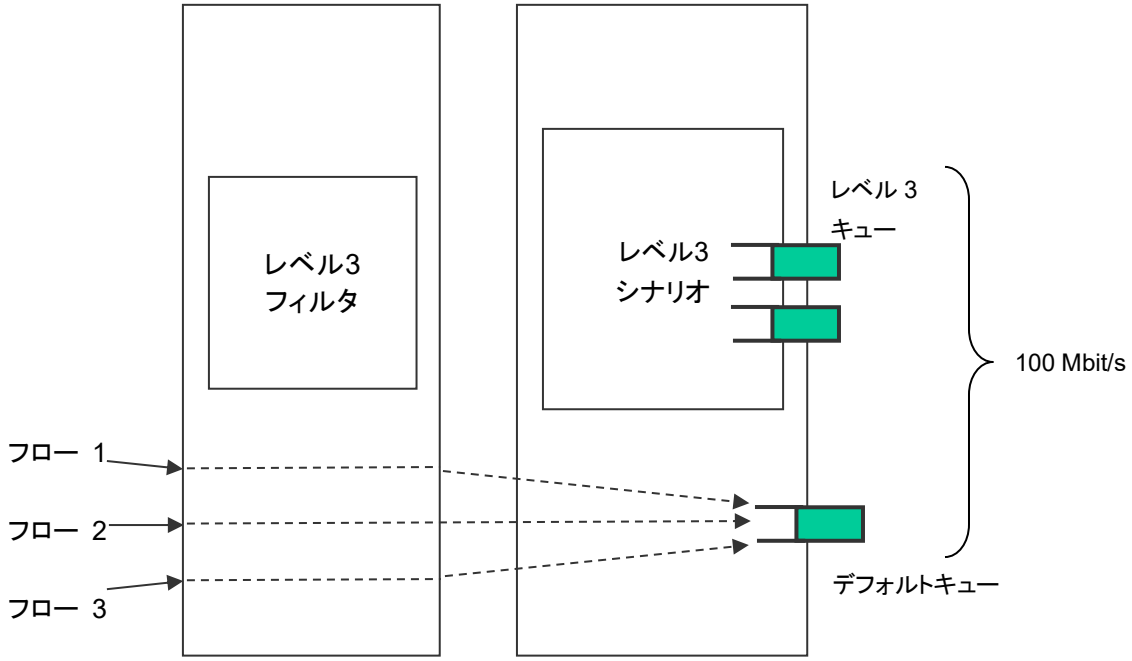
また，以下の 3 つのトラフィックが入力されたと仮定します。

- 192.168.1.1 から 192.168.1.100 へのトラフィック(フロー 1)
- 192.168.1.1 から 192.168.1.150 へのトラフィック(フロー 2)
- 192.168.1.1 から 192.168.1.200 へのトラフィック(フロー 3)

これらのフローは，レベル 2 フィルタに一致し，レベル 3 フィルタに一致しないため，デフォルトキューにパケットを格納します。

- フロー 1～3 の合計が 100 Mbit/s

レベル 2 シナリオとして，合計 100 Mbit/s の帯域を保証します。



ただし，優先度が高いクラスのレベル 3 キューに割り当てられたフローが流れている場合，デフォルトキューに割り当てられたフローの合計は 100 Mbit/s の帯域を保証することができません。

(2) 集約キュー(レベル n キュー)

Aggregate(集約キューモード)のレベル n シナリオとは, レベル n フィルタに一致した複数のフローを 1 つのレベル n キューに集約して割り当てる方式です。

レベル n フィルタに一致し, その下位に属するレベル n フィルタにも一致したすべてのフローを同じレベル n キューに割り当て, トラフィックコントロールを行います。

たとえば, 送信元 IP アドレスが 192.168.10.1 で, 宛先 IP アドレスが 192.168.10.100, 192.168.10.150, 192.168.10.200 の場合に, レベル n シナリオの集約キューで最大帯域 10 Mbit/s に設定した場合は, 以下のようになります。

本装置に, 以下のフィルタを登録したと仮定します。

- レベル 2 フィルタ
  送信元 IP アドレス ： 192.168.0.0 - 192.168.255.255
  宛先 IP アドレス ： 192.168.0.0 - 192.168.255.255
- レベル 3 フィルタ
  送信元 IP アドレス ： 192.168.10.0 - 192.168.10.255
  宛先 IP アドレス ： 192.168.10.0 - 192.168.10.255

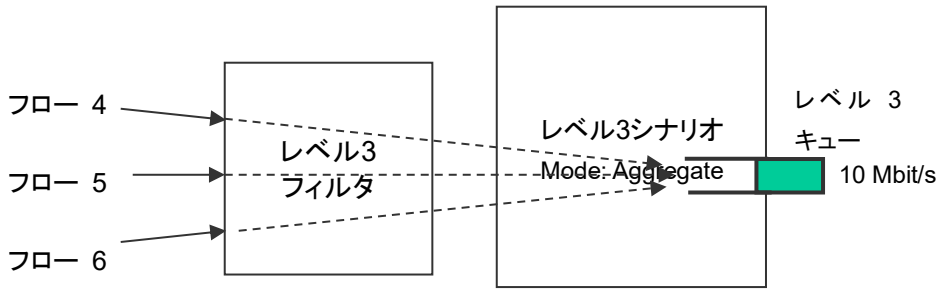
また, 以下の 3 つのトラフィックが入力されたと仮定します。

- 192.168.10.1 から 192.168.10.100 へのトラフィック(フロー 4)
- 192.168.10.1 から 192.168.10.150 へのトラフィック(フロー 5)
- 192.168.10.1 から 192.168.10.200 へのトラフィック(フロー 6)

これらのフローは, レベル 2 フィルタに一致し, レベル 3 フィルタにも一致するため, レベル 3 キュー(集約キュー)にパケットを格納します。

- フロー 4～6 の合計が 10 Mbit/s

レベル 3 シナリオとして, 合計 10 Mbit/s の帯域を使用します。



(3) 個別キュー(レベル n キュー)

Individual(個別キューモード)のレベル n シナリオとは, レベル n フィルタに一致した複数のフローに対して, 個別のレベル n キューを割り当てる方式です。

レベル n フィルタに一致したすべてのフローごとに個別のレベル n キューを割り当て, トラフィックコントロールを行います。下位レベルにシナリオを登録することはできますが, Individual シナリオの下位レベルシナリオにフローが割り当てられることはありません。

たとえば, 送信元 IP アドレスが 192.168.20.1 で, 宛先 IP アドレスが 192.168.20.100, 192.168.20.150, 192.168.20.200 の場合に, レベル n シナリオの個別キューで最大帯域 10 Mbit/s に設定した場合は, 以下のようになります。

本装置に, 以下のフィルタを登録したと仮定します。

- レベル 2 フィルタ
  送信元 IP アドレス ： 192.168.0.0 - 192.168.255.255
  宛先 IP アドレス ： 192.168.0.0 - 192.168.255.255
- レベル 3 フィルタ
  送信元 IP アドレス ： 192.168.20.0 - 192.168.20.255
  宛先 IP アドレス ： 192.168.20.0 - 192.168.20.255

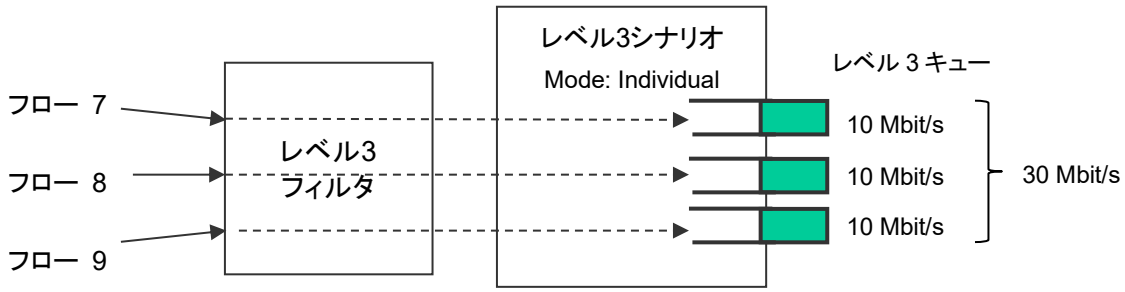
また, 以下の 3 つのトラフィックが入力されたと仮定します。

- 192.168.20.1 から 192.168.20.100 へのトラフィック(フロー 7)
- 192.168.20.1 から 192.168.20.150 へのトラフィック(フロー 8)
- 192.168.20.1 から 192.168.20.200 へのトラフィック(フロー 9)

これらのフローは, レベル 2 フィルタに一致し, レベル 3 フィルタにも一致するため, レベル 3 キュー(個別キュー)にパケットを格納します。

- フロー 7 は 10 Mbit/s
- フロー 8 は 10 Mbit/s
- フロー 9 は 10 Mbit/s

レベル 3 シナリオとして, 合計 30 Mbit/s の帯域を使用します。



注:

モニタリングマネージャ 2 において, 個別キューモードのシナリオは集約キューモードと同様に 1 つのキューとして表示されます。個別キューは表示されません。

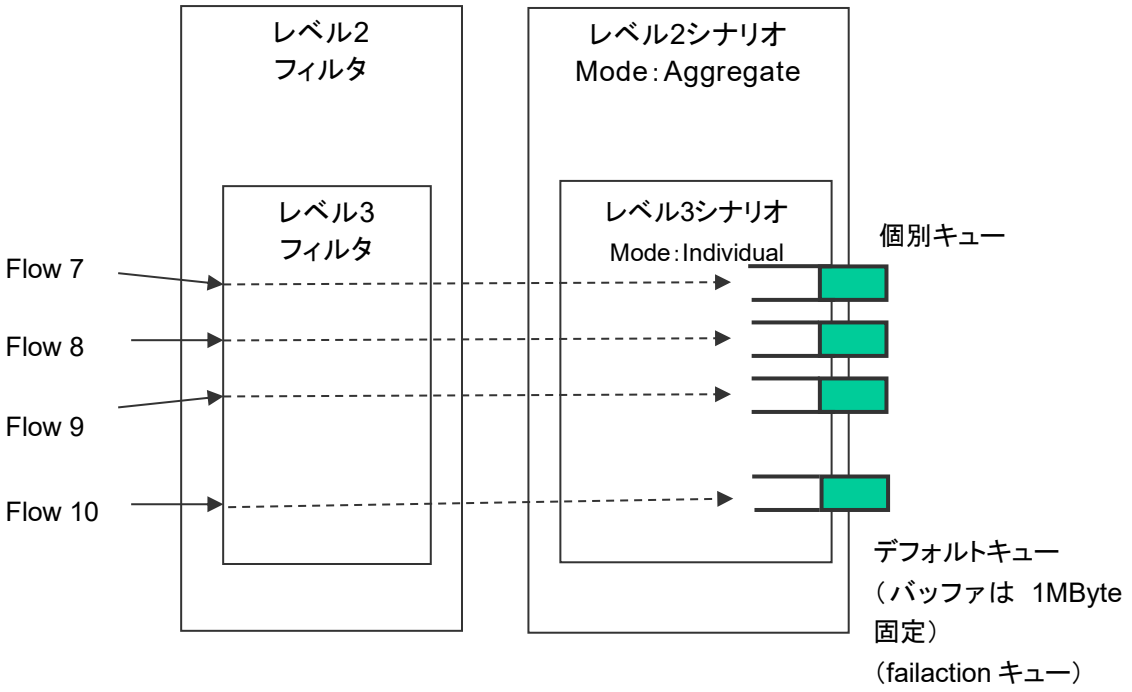
個別キューモードの場合，シナリオで割り当てる個別キューの最大数を設定することが可能です。個別キューの最大数を超えてフローを作成する場合は，キュー最大数超過アクション（ベストエフォート転送，トラフィックアトリビュート転送，または廃棄）に従います。

たとえば，上記の例で，レベル 3 シナリオの個別キュー最大数 3，キュー最大数超過アクション forwardbesteffort とします。

本装置に，Flow 7～Flow 9 に加えて，以下のトラフィックが入力されたと仮定します。

- 192.168.20.1 から 192.168.20.250 へのトラフィック（Flow 10）

このフローは，レベル 2 フィルタに一致し，レベル 3 フィルタにも一致しますが，すでに個別キューを 3 個割り当てているため，キュー最大数超過アクション（ベストエフォート転送）に従います。



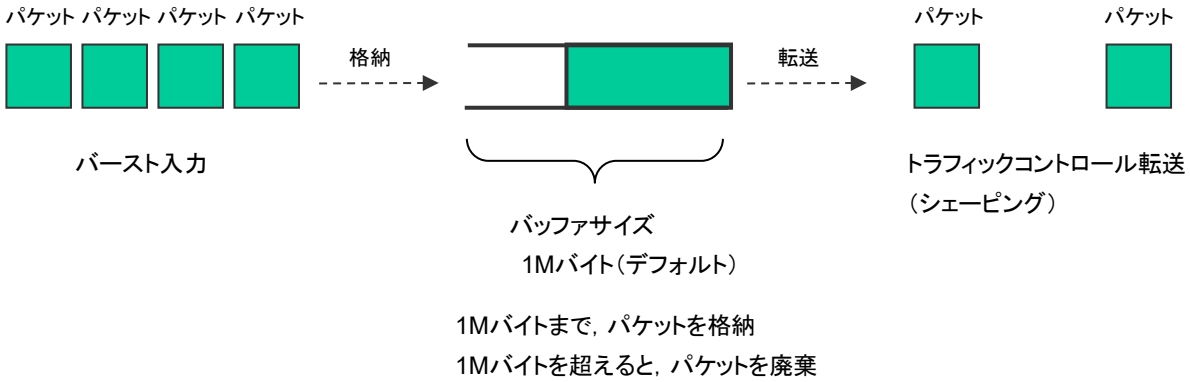
なお，デフォルトキューのキューバッファサイズは 1Mbyte（固定）です。

また，キュー最大数超過アクション discard の場合は，Flow 10 のトラフィックを廃棄します。

（4）バッファサイズ

レベル n キューは，バッファサイズを設定することが可能です。

バッファサイズは，キューで許容できる入力バースト長です。バーストでパケット受信したときに，キューに格納できるバイト数です。



入力バースト長が，バッファサイズを超えてしまうと，パケットを廃棄します。バッファサイズ不足により，パケットが廃棄されてしまう場合，レベル n シナリオ（トラフィックアトリビュート）でバッファサイズを設定してください。

パケットが廃棄されているかどうかは，キュー統計情報で確認することができます。（詳細は「第 10 章 SSH 機能」を参照してください。）

デフォルトキューおよびレベル n シナリオで割り当てるレベル n キューのバッファサイズは，バイト指定で設定します。

以下にレベル n シナリオで割り当てるレベル n キューのバッファサイズを変更するコマンドを示します。

Sample 1）　すでに存在するレベル 2 シナリオに対して，バッファサイズ 5M バイトに変更する場合

PureFlow（A）＞ update scenario “/port1/Tokyo” action aggregate bufsize 5M

Sample 2）　すでに存在するレベル 3 シナリオに対して，バッファサイズ 2M バイトに変更する場合

PureFlow（A）＞ update scenario “/port1/Tokyo/Shinjuku” action aggregate bufsize 2M

注：

システムパケットバッファとキューバッファの関係性

各キューは，キューで使用可能な帯域を超過してしまったパケットをバッファに蓄積する際，設定したバッファサイズまでシステムパケットバッファを動的に使用します。キューバッファは，システムパケットバッファの中から固定に確保されるわけではありません。シナリオのキューバッファサイズの設定値合計は，システムパケットバッファサイズを超過し設定することができます。ただし，同時に使用できるシナリオのキューバッファサイズの合計は，システムパケットバッファサイズまでです。各シナリオのキューバッファは，パケットの入力順に，設定したバッファサイズを上限とし，システムパケットバッファを使用します。

(5) クラス

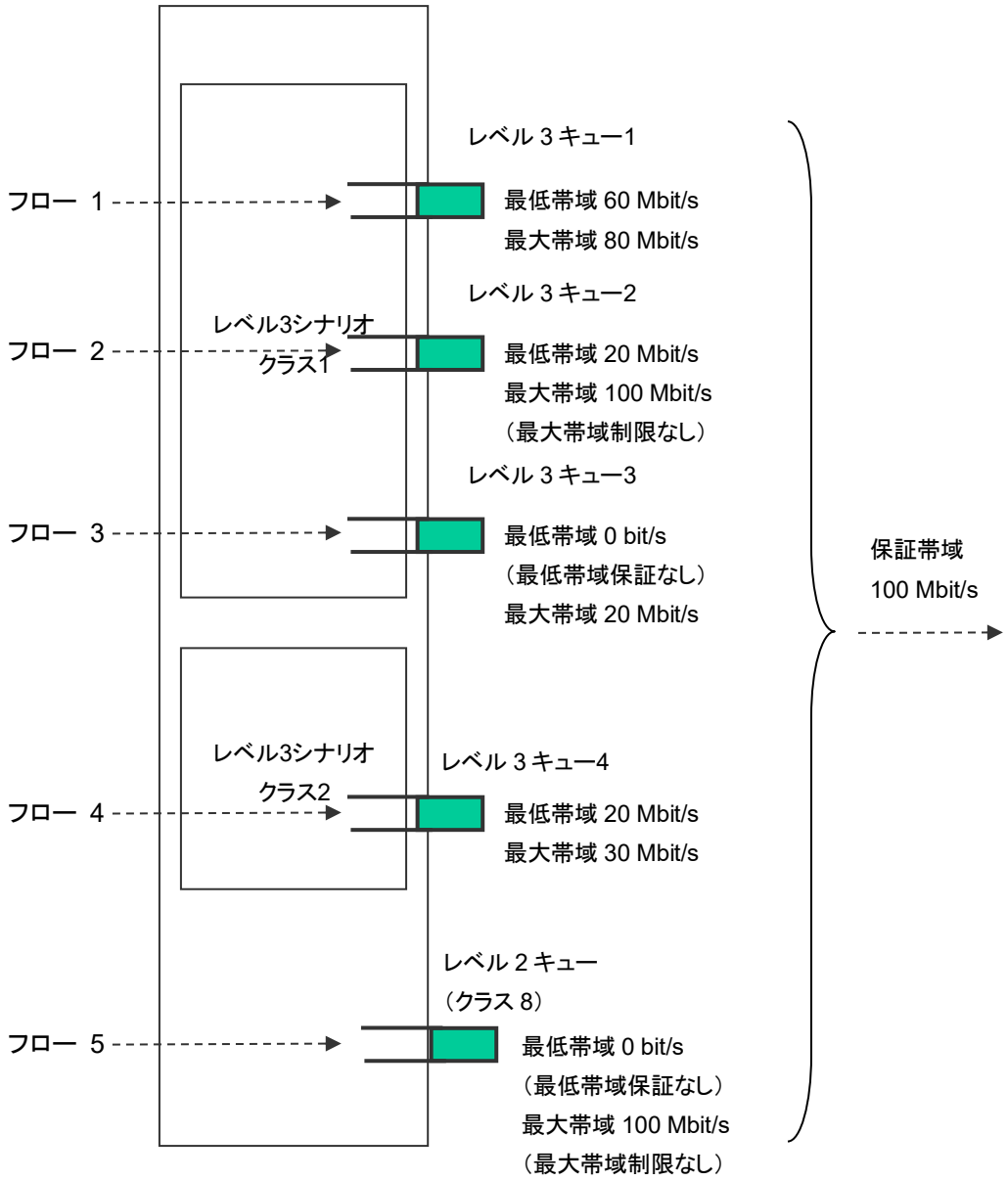
レベル 2 以降のキューには，クラス(キューの優先順位)を設定することが可能です。

本装置のトラフィックコントロール方式は，8 クラス(クラス 1～8)の優先度に基づくキュー間を優先度の高いものから出力していく方式(Strict Priority)です。

以下に Strict Priority 動作を示します。

本装置に以下のレベル 2，3 キューを割り当てたものと仮定します。

- レベル 2 キュー(クラス 8，保証帯域 100 Mbit/s)
- レベル 3 キュー1(クラス 1，最低帯域 60 Mbit/s／最大帯域 80 Mbit/s)
- レベル 3 キュー2(クラス 1，最低帯域 10 Mbit/s／最大帯域制限なし)
- レベル 3 キュー3(クラス 1，最低帯域保証なし／最大帯域 10 Mbit/s)
- レベル 3 キュー4(クラス 2，最低帯域 20 Mbit/s／最大帯域 30 Mbit/s)



a)　レベル 2 シナリオは，帯域を保証します。

たとえば，レベル 2 シナリオ外で 990 Mbit/s のフローが流れている場合でも，レベル 2 シナリオ内のフローは 100 Mbit/s を保証します。

ただし，各レベル 2 シナリオに割り当てた保証帯域の合計がレベル 1 シナリオの帯域を超えている場合，レベル 2 シナリオの帯域を保証できません。

b)　最低帯域保証ありのレベル 3 キューに割り当てられたフローは，最低帯域を保証します。

たとえば，フロー3（100 Mbit/s）のフローが流れている場合でも，フロー1（60 Mbit/s）のフローは 60 Mbit/s，フロー2（20 Mbit/s）のフローは 20 Mbit/s でトラフィックコントロールします。

ただし，各レベル 3 シナリオに割り当てた最低帯域の合計が，レベル 2 シナリオの保証帯域を超えている場合，レベル 3 シナリオの最低帯域を保証できません。

c)　同じレベル 2 シナリオ内に，複数のクラスのレベル 3 キューを割り当てた場合，優先度が低いクラスのレベル 3 キューのフローは，最低帯域を保証できません。優先度が低いクラスのレベル 3 キューは，優先度が高いクラスの余剰帯域でトラフィックコントロールします。

たとえば，フロー1（60 Mbit/s），フロー2（20 Mbit/s），フロー3（15 Mbit/s）のフロー（クラス 1）と，フロー4（20 Mbit/s）のフロー（クラス 2）が流れている場合，フロー4 は 5 Mbit/s でトラフィックコントロールします。

d)　最大帯域制限ありのレベル 3 キューに割り当てられたフローは，その最大帯域で制限します。

たとえば，フロー3（30 Mbit/s）が流れている場合は，フロー3 は 20 Mbit/s でトラフィックコントロールします。

また，レベル 3 キューの最大帯域がレベル 2 シナリオの保証帯域を超えている場合，レベル 2 シナリオの保証帯域でトラフィックコントロールします。

e)　最大帯域制限なしのレベル 3 キューに割り当てられたフローは，レベル 2 シナリオの保証帯域でトラフィックコントロールします。

たとえば，フロー2（120 Mbit/s）が流れている場合，フロー2 は 100 Mbit/s でトラフィックコントロールします。

レベル 3 キューに優先度をつけると，優先度の高いクラスのキューに格納されたパケットを優先して転送しますので，優先度が低いクラスに比べて揺らぎが小さくなります。レベル 3 キューに優先度をつけたい場合，レベル 3 シナリオ（トラフィックアトリビュート）でクラスを設定してください。

以下にレベル 3 シナリオのクラスを変更するコマンドを示します。

Sample)　　すでに存在するレベル 3 シナリオに対して，クラス 1 に変更する場合

```
PureFlow(A)> update scenario "/port1/Tokyo/Shinjuku" action aggregate class 1
```

注:

CLI コマンドなどによるシナリオのクラス変更は，対象シナリオが 1 パケット送信したあとに反映されます。優先度の高い他シナリオが帯域を占有している状態では，対象シナリオがパケットを送出できないため，クラス変更が反映されません。クラスの変更は帯域に余裕がある（最大帯域に達していない）状態で行ってください。

8.7.3　通信ギャップモード

Ethernetは，フレームを連続して送信する場合，フレームとフレームの間にギャップとプリアンブルが挿入されます。トラフィックアトリビュート（シナリオ，Network ポート）の帯域を設定するときに，これらを含めてトラフィックコントロール（ネットワーク帯域全体）を行うか，または含めないでトラフィックコントロール（フレームのみを対象）を行うかを選択することができます。本設定は装置全体に適用します。

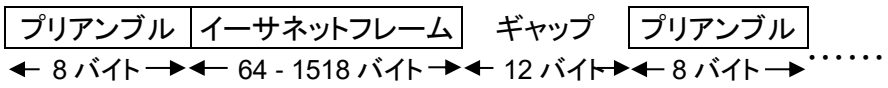


図　イーサネットフレームのギャップとプリアンブルについて

通信ギャップモードに関するCLIは以下のコマンドがあります。

set bandwidth mode {gap [<size>] | no_gap}	通信帯域設定で，フレーム間ギャップとプリアンブルの有効／無効を選択します。デフォルト値は“no_gap”（無効）です。 gap の場合は，フレーム間ギャップおよびプリアンブルを帯域に含み，サイズを指定することができます。サイズの設定範囲は–100[Byte]～100[Byte]です。サイズを 0 に設定するとno_gapと同意になります。

コマンドの実行例を示します。

```
PureFlow(A)> set bandwidth mode gap
PureFlow(A)>
```

通信ギャップモードを有効としたときは，トラフィックアトリビュート（シナリオ，Network ポート）の帯域設定値による制御がフレーム間ギャップとプリアンブルを含めた制御となります。この設定は，帯域設定値が物理回線と同じ数値の意味を示しますので，出力 WAN 回線の帯域に対する輻輳回避や，トラフィックの優先制御を実施する場合に有効です。

通信ギャップモードを無効としたときは，トラフィックアトリビュート（シナリオ，Network ポート）の帯域設定値による制御がフレーム間ギャップとプリアンブルを含めないイーサネットフレームのみのデータレートとして制御します。この設定は，一般的にフレーム間ギャップやプリアンブルを含まないデータレートで示されているコンテンツ，映像，音声などのバースト回避のための平滑化，サーバに対して受信レートを制御するなどのコンテンツレート制御に有効です。

通信ギャップモードを無効で使用する場合は，トラフィックアトリビュート（シナリオ，Network ポート）の帯域設定値が回線帯域と異なる出力レートとなるので，通信ギャップを考慮した帯域設定値にする必要があります。たとえば回線帯域が 100 Mbit/s の場合，すべてのフレーム長（64 バイト～1522 バイト）においてフレーム落ちなく転送できる設定値は約 76 Mbit/s（(100 Mbit/s) ×（64 byte/84 byte））になります。この場合，いかなるフレーム長においても 76 Mbit/s に制限するので，フレーム長が長いほど転送量に無駄が生じることになります。回線帯域を無駄なく使用する場合は，通信ギャップモードを有効に設定し，フレーム間ギャップを含めた帯域を設定してください。

注：

通信ギャップモードの設定値はパケットごとにパケット受信時に適用されます。通信ギャップモード変更時に各シナリオバッファに滞留しているパケットには適用されません。このため，通信ギャップモードの変更は，変更時に滞留していたパケットを排出したあとに反映されます。

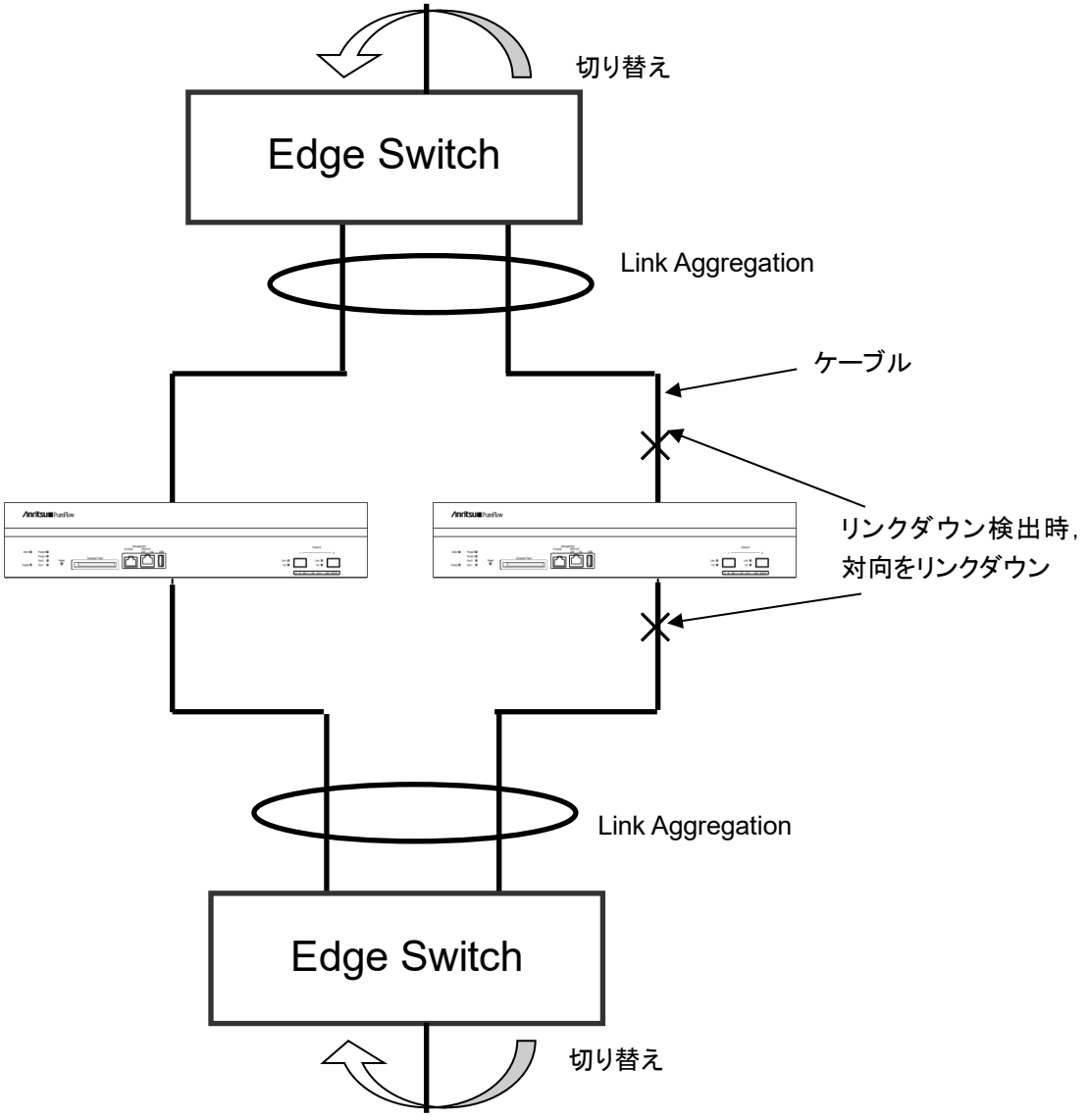
第9章　リンクダウン転送機能

ここでは，リンクダウン転送機能について説明します。

9.1　リンクダウン転送機能

本装置のリンクダウン転送機能を使用すると，「IEEE802.3ad Link Aggregation」などの回線冗長機能を使用している装置の間に本装置を挿入しても外部装置間の回線冗長機能を妨げることなく協調動作を行います。

本装置では，リンクダウンを検出すると対向のリンクをダウンさせることにより対向装置に対して警報の転送を行います。対向の装置は，そのリンクダウンを検出することにより回線を切り替えることが可能となります。



リンクダウン転送機能の設定を以下に示します。

set lpt {enable | disable}	ポート(1/1)のリンクがダウンした場合に対向のポート(1/2)をリンクダウンさせて警報転送を行う機能と，ポート(1/2)のリンクがダウンした場合に対向のポート(1/1)をリンクダウンさせて警報転送を行う機能の有効／無効を設定します。
show lpt	リンクダウン転送機能の有効／無効状態を表示します。

コマンドの実行例を示します。

PureFlow(A)> set lpt enable
PureFlow(A)>

(注意)
以下の場合は，リンクダウン転送機能有効時でもケーブルを接続しているポートが一時的にリンクアップし通信できる状態となり，約 10 秒後にリンクダウンします。

・対となるポートの両方にケーブルを接続せずに本装置を起動し，片方のポートのみにケーブルを接続した場合

・対となる両方のポートがリンクアップしている状態から片方のケーブルを抜き、次にもう片方のケーブルを抜いてから先に抜いたケーブルを接続した場合

・初期設定(リンクダウン転送機能無効)で片側ポートのみケーブルを接続して本装置を起動し，リンクダウン転送機能を有効とした場合

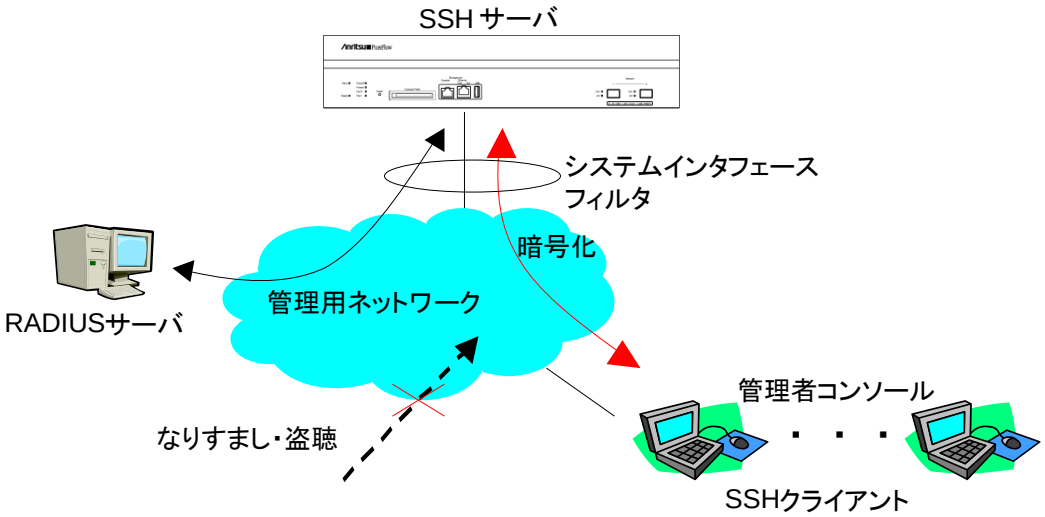
（空白ページ）

ここでは，SSH（Secure SHell）機能について説明します。

10.1 概要

本装置は，SSH バージョン 2 に準拠した SSH サーバ機能を提供します。SSH サーバ機能により，本装置と SSH クライアント間の通信が暗号化され，安全性が保証されていないネットワークを経由する場合でも，セキュアな遠隔操作が可能になります。また，強力なサーバ認証機能を有し，第 3 者による「盗聴」や「なりすまし」を防止することができます。

SSH サーバによる接続を利用する場合も，不特定多数の端末から本装置への通信を制限するためのシステムインタフェースフィルタを設定することができます。詳細は，「第 7 章 システムインタフェースの設定」を参照してください。また，Telnet と同様に，ローカルに設定された root ユーザのパスワード認証だけでなく，RADIUS サーバ経由でのパスワード認証が利用できます。RADIUS 機能の詳細は，「第 15 章 RADIUS 機能」を参照してください。



10.2 仕様一覧

本装置の SSH サーバ機能の仕様一覧を記載します。

項目	内容
SSH バージョン	SSH Ver.2 準拠
ユーザ認証方式	パスワード認証
鍵交換アルゴリズム	ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521, diffie-hellman-group-exchange-sha256, diffie-hellman-group-exchange-sha1, diffie-hellman-group14-sha1, diffie-hellman-group1-sha1
公開鍵アルゴリズム	RSA 2048bit, DSA 1024bit, ECDSA 256bit
暗号化アルゴリズム	aes128-ctr, aes192-ctr, aes256-ctr, arcfour256, arcfour128, aes128-gcm@openssh.com, aes256-gcm@openssh.com, aes128-cbc, blowfish-cbc, cast128-cbc, aes192-cbc, aes256-cbc, arcfour, rijndael-cbc@lysator.liu.se
MAC アルゴリズム	hmac-md5-etm@openssh.com, hmac-sha1-etm@openssh.com, umac-64-etm@openssh.com, umac-128-etm@openssh.com, hmac-sha2-256-etm@openssh.com, hmac-sha2-512-etm@openssh.com, hmac-ripemd160-etm@openssh.com, hmac-sha1-96-etm@openssh.com, hmac-md5-96-etm@openssh.com, hmac-md5, hmac-sha1, umac-64@openssh.com, umac-128@openssh.com, hmac-sha2-256, hmac-sha2-512, hmac-ripemd160, hmac-ripemd160@openssh.com, hmac-sha1-96, hmac-md5-96
接続ポート番号	22
クライアント最大接続数	4（telnet 接続数と合わせて）

10.3 SSH の利用方法

10.3.1 本体の設定

本装置の SSH サーバ機能を使用するには，以下の設定が必要です。

(1) システムインタフェースの設定

本装置の IP アドレスや Gateway を設定します。接続する端末を制限する場合は，システムインタフェースフィルタを設定します。詳細は，「第 7 章 システムインタフェースの設定」を参照してください。

(2) 公開鍵（ホスト鍵）の生成

SSH サーバは，SSH クライアントとの接続を確立するために，ホスト鍵を必要とします。このホスト鍵は，工場出荷時に無作為に生成された鍵が設定されており，装置外部からは参照できない状態で装置内部に保存しています。特に，新しく生成する必要はありませんが，必要に応じてシリアルコンソールから変更することができます。

10.3.2 SSHクライアントの準備

SSH バージョン 2 に準拠した SSH クライアントを用意してください。

10.3.3 注意事項

(1) 初めて SSH 接続を行うときの注意事項

SSH クライアントからリモートホストに初めて接続するとき，そのホストを信用していいかどうかを確認するサーバ認証を行います。このとき，SSH クライアントは，リモートホストが通知してきた認証鍵の fingerprint を表示し，このホストに接続していいかの確認を求めます。この場合は，SSH クライアントが表示したリモートホストの fingerprint と本装置の fingerprint が一致しているかどうかを確認することを推奨します。

本装置のホスト鍵の fingerprint は，“show ssh”コマンドで表示可能です。

(2) ホスト鍵の再生成

本装置の SSH サーバが使用するホスト鍵は，工場出荷時に生成され本装置内部に保存されています。このホスト鍵は，“set ssh server key”コマンドで変更することが可能ですが，このコマンドは，シリアルコンソールからログインしたときだけ実行可能です。

(3) ホスト鍵を再生成したあとの SSH 接続

SSH クライアントは，過去に接続したリモートホストの fingerprint を記憶しており，過去に通知してきた fingerprint が異なる場合，SSH クライアントは，ワーニングを表示し，リモートホストへの SSH 接続を切断します。これは，リモートホストの「なりすまし」を防止するための動作であり，多くの SSH クライアントが同様な動作をします。

本装置のホスト鍵を再生成した場合は，本装置に SSH で接続したことがある SSH クライアントから，本装置の fingerprint を削除または更新する必要があります。詳細は，SSH クライアントのマニュアルを参照してください。

(4) RADIUS 機能を有効にした場合の SSH 接続

本装置の RADIUS 機能を有効にした場合，本装置は，ログイン認証時に RADIUS サーバに問い合わせます。SSH クライアントから本装置に新しい SSH セッションの接続を試みた場合，SSH クライアントと本装置の通信は SSH 機能により暗号化されますが，RADIUS サーバと本装置の通信は暗号化されません。RADIUS サーバとの通信を傍受された場合，パスワードは RADIUS プロトコルにより秘匿されますが，ログイン名が第三者によって解読される可能性があります。

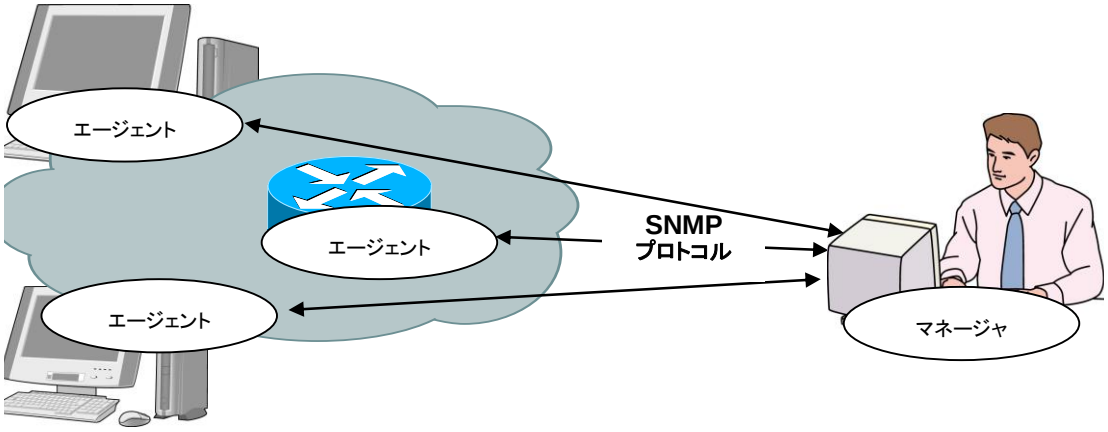
（空白ページ）

第11章　SNMP の設定

ここでは，SNMP の機能と設定について説明します。

11.1 SNMP の概要

SNMP は，ルータやサーバなどのネットワーク機器に対してネットワークを通してリモートで管理するためのプロトコルです。SNMP では，ルータやサーバなどの管理される側をエージェントノード（またはエージェント），管理用のアプリケーションソフトウェアをインストールした PC や EWS をマネジメントノード（またはマネージャ）と呼んでいます。ネットワーク管理者はマネジメントノードのコンソールを使って，ネットワーク機器（エージェントノード）の障害を発見したり，設定を変更することで，日々のネットワーク管理業務を遂行します。



SNMP には SNMPv1，SNMPv2c，SNMPv3 の 3 つのバージョンが存在します。
本装置は SNMPv1，SNMPv2c，SNMPv3 の 3 つのバージョンをすべてサポートしています。それぞれのバージョンによる違いは以下のとおりです。

・ SNMPv1：最もシンプルで簡単なプロトコルで，管理情報の取得，設定，トラップ（警報）の 3 つのオペレーションから成り立っています。セキュリティはコミュニティ名と呼ばれる文字列（パスワードのようなもの）で実現されています。コミュニティ名は SNMPv1 データ要求とともにパケットに含まれてしまうため，ネットワークを測定器などでモニタされると盗み見されてしまいます。コミュニティ名は暗号化されないため，安全とみなすことはできません。外部の人間がネットワークに接続しないイントラネットなどでしか用いることができません。

・ SNMPv2c：管理情報の取得に，バルク転送と呼ばれるデータの一括取得処理をサポートすることで，プロトコルのオーバヘッドを軽減しました。アクセス・セキュリティは SNMPv1 と同様にコミュニティ文字列で行うため，セキュリティ強度は SNMPv1 と同等です。

・ SNMPv3：最新のプロトコルで，ユーザ名とそれに対応した暗号化パスワードでアクセスを認証します。エージェントへのアクセスはユーザ名が必要です。ユーザ名はグループという単位でまとめられ，グループごとに管理情報の取得，設定の権限の範囲を変えておくことで，コーポレートごとの管理者グループ，部門管理者グループ，一般ユーザグループという具合いに，権限を階層構造にもたせることができます。大規模イントラネットからインターネットまで，一般的な用途で利用可能です。SNMPv3 のセキュリティは暗号化機能も持ちますが，本装置では暗号化機能をサポートしていません。

一般的なマネジメントソフトウェアは，エージェントがサポートできるバージョンを自動検知し，最も高いバージョンを優先使用します。

11.2 SNMPv1/SNMPv2c の設定

SNMPv1 および SNMPv2c はどちらもコミュニティ名と呼ばれる文字列（パスワードのようなもの）を設定することでマネジメントノードからのアクセスが可能となります。

add snmp community <community_string> [version {v1 | v2c}] [view <view_name>] [permission {ro | rw}]	SNMPv1/v2c のコミュニティを追加します。
delete snmp community <community_string>	コミュニティを削除します。
add snmp view <view_name> <oid> {included | excluded}	SNMP の View（管理範囲の制限）を設定します。 注）snmpv2 グループは，本コマンドで指定可能ですが SNMP によるアクセスはできません。
delete snmp view <view_name> [<oid>]	SNMP の View（管理範囲の制限）を削除します。
show snmp community [<community_string>]	設定されているコミュニティを表示します。
show snmp view [<view_name>]	設定されている View を表示します。

最初に SNMPv1 コミュニティに“netman1”，SNMPv2c コミュニティに“netman2”というコミュニティ名を設定します。

```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp community netman1 version v1 permission rw
PureFlow(A)> add snmp community netman2 version v2c permission rw
```

View はそのコミュニティ名でアクセスするマネジメントノードが本装置のどの MIB Tree に対してアクセス可能かを許可／制限する機構です。add snmp community で view を省略時は“All”の View 名に対してアクセスが可能となります。また，v2c のトラップ送信を使用する場合，<oid>パラメータに，“private”を指定する際は “system”と“snmpmodules”の“included”設定も追加してください。

SNMPv1 コミュニティ netman1 を interfaces グループだけにアクセス制限をかけるには，以下のコマンドを実行します。

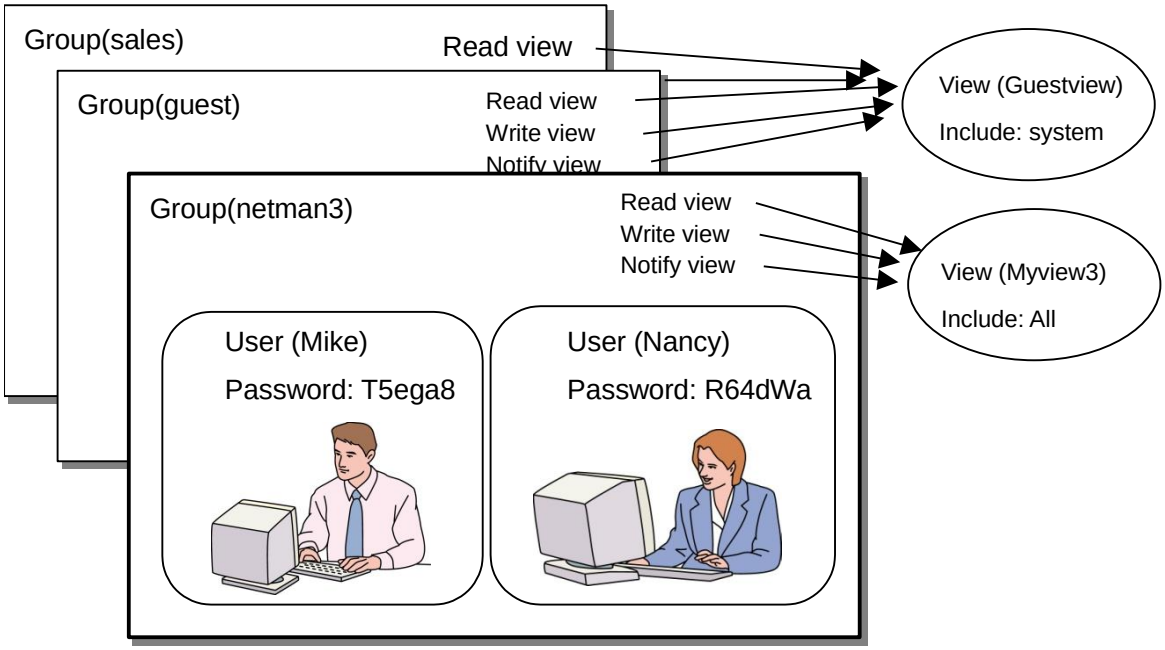
```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp view myview1 interface included
PureFlow(A)> add snmp community netman1 version v1 view myview1 permission rw
```

設定コマンドで設定した community 名や view の内容を確認するには，show snmp community コマンドと show snmp view コマンドを使用します。

```
PureFlow> show snmp community
-----------------------------------------------------------------
Community Name                  :netman1
Version                         :v1
Read View                       :myview1
Write View                      :myview1
-----------------------------------------------------------------
Community Name                  :netman2
Version                         :v2c
Read View                       :All
Write View                      :All
-----------------------------------------------------------------
PureFlow>
PureFlow> show snmp view
-------------------------------------------------------------------------------
View name                       :All
Subtree                         :iso
Access State                    :included
-------------------------------------------------------------------------------
View name                       :myview1
Subtree                         :interface
Access State                    :Included
PureFlow>
```

11.3 SNMPv3 の設定

SNMPv3 の管理フレームワークは，ユーザごとにセキュリティを設定するユーザベースセキュリティです。各ユーザはグループに属し，グループの属性として View を設定します。



SNMPv3 を使用するためには，グループ，ユーザ，View の設定が必要です。以下のコマンドを使用します。

add snmp group <group_name> [auth_type {auth | noauth}] [read <readview>] [write <writeview>] [notify <notifyview>]	SNMPv3 のグループを追加します。
delete snmp group <group_name>	グループを削除します。
add snmp user <user_name> <group_name> [auth_type {auth | noauth}] [password <auth_password>]	SNMPv3 のユーザを追加します。パスワードを指定する場合，8 文字以上 24 文字以下で指定してください。
delete snmp user <user_name>	ユーザを削除します。
add snmp view <view_name> <oid> {included | excluded}	SNMP の View（管理範囲の制限）を設定します。 注）snmpv2 グループは，本コマンドで指定可能ですが SNMP によるアクセスはできません。
delete snmp view <view_name> [<oid>]	SNMP の View（管理範囲の制限）を削除します。
show snmp group [<group_name>]	設定されているグループを表示します。
show snmp user [<user_name>]	設定されているユーザを表示します。
show snmp view [<view_name>]	設定されている View を表示します。

View はそのグループ，ユーザ名でアクセスするマネジメントノードが本装置のどの MIB Tree に対してアクセス可能かを許可／制限する機構です。add snmp group で view を省略時は“All”の View 名に対してアクセスが可能となります。また，v3 のトラップ送信を使用する場合，<oid>パラメータに，“private”を指定する際は “system”と“snmpmodules”の“included”設定も追加してください。

以下のコマンド例は SNMPv3 ユーザ Mike と Nancy をグループ netman3 の一員として設定します。

```
PureFlow(A)> add snmp view myview3 iso included
PureFlow(A)> add snmp group netman3 auth_type auth read  myview3 write  myview3
notify myview3
PureFlow(A)> add snmp user Mike netman3 auth_type auth password T5ega8GH
PureFlow(A)> add snmp user Nancy netman3 auth_type auth password R64dWa99
```

11.4 TRAP の設定

SNMP ではエージェントノードの状態変化を検出して，マネジメントノードへ通知する機能があります。通知用の View とマネジメントノード（ホスト）のアドレスを設定することでマネジメントノードへの TRAP（ノーティフィケーション）の送信が可能となります。

add snmp view <view_name> <oid> {included | excluded}	SNMP の View（管理範囲の制限）を設定します。
add snmp host <host_address> version {v1 | v2c | v3 [auth_type { auth | noauth}] } {user | community} <community_string / username> } {trap | inform} [udp_port <port_number>] [<notification_type>]	SNMP TRAP（ノーティフィケーション）の送信先を示すホストを追加します。
delete snmp host <host_address>	TRAP の送信先を示すホストを削除します。
set snmp traps {authentication | linkup | linkdown | warmstart | coldstart | modulefailurealarm | modulefailurerecovery | systemheatalarm | systemheatrecovery | powerinsert | powerextract | powerfailure | powerrecovery | faninsert | fanextract | fanfailure | fanrecovery | queuebuffalarm | queuebuffrecovery | systembuffalarm | systembuffrecovery | queueallocalarm | queueallocrecovery | maxqnumalarm | maxqnumrecovery} {enable | disable}	SNMP の TRAP 送信を有効／無効に設定します。トラップ種別ごとに設定することができます。 <trapname>には， “authentication”， “linkup”， “linkdown”， “coldstart”， “modulefailurealarm”， “modulefailurerecovery”， “systemheatalarm”， “systemheatrecovery”， “powerinsert”， “powerextract”， “powerfailure”， “powerrecovery”， “faninsert”， “fanextract”， “fanfailure”， “fanrecovery”， “queuebuffalarm”， “queuebuffrecovery”， “systembuffalarm”， “systembuffrecovery” “queueallocalarm” “queueallocrecovery” “maxqnumalarm” “maxqnumrecovery” を指定することができます。
show snmp host [<host_address>]	TRAP の送信先を示すホストの一覧を表示します。

最初に SNMP TRAP 送信用に View を設定します。SNMP 基本 TRAP は snmpv2 オブジェクト，Enterprise TRAP は private オブジェクトに含まれています。snmpv2 オブジェクト，private オブジェクトへのアクセスを有効にすることで TRAP をマネジメントノードの送信することが可能となります。

```
PureFlow(A)> add snmp view All iso included
PureFlow(A)> add snmp host 192.168.1.10 version v1 community public trap udp_port 162
```

authenticationFailure TRAP の送信を無効にするには下記を設定します。

```
PureFlow(A)> set snmp traps authentication disable
```

設定コマンドで設定したホストの内容を確認するには，show snmp host コマンドを使用します。

```
PureFlow(A)> show snmp host
-------------------------------------------------------
Host Address                :192.168.1.10
Version                     :v1
Security                    :No Authentication
Security Name               :public
UDP port                    :162
Notification Type           :all
-------------------------------------------------------
Host Address                :192.168.1.11
Version                     :v2c
Security                    :No Authentication
Security Name               :public
UDP port                    :162
Notification Type           :all
PureFlow(A)>
```

設定コマンドで設定した TRAP の有効／無効の内容を確認するには，show snmp system コマンドを使用します。

```
PureFlow(A)> show snmp system
-------------------------------------------------------------------------------
System Location                 :Not Yet Set
System Contact                  :Not Yet Set
System Name                     :Not Yet Set
Engine ID                       :00:00:04:7f:00:00:00:a1:c0:a8:01:01

Traps
  authentication                :disable
  linkup                        :enable
  linkdown                      :enable
  warmstart                     :enable
  coldstart                     :enable
  modulefailurealarm            :enable
  modulefailurerecovery         :enable
  systemheatalarm               :enable
  systemheatrecovery            :enable
  powerinsert                   :enable
  powerextract                  :enable
  powerfailure                  :enable
  powerrecovery                 :enable
  faninsert                     :enable
  fanextract                    :enable
  fanfailure                    :enable
  fanrecovery                   :enable
  queuebuffalarm                :enable
  queuebuffrecovery             :enable
  systembuffalarm               :enable
  systembuffrecovery            :enable
  queueallocalarm               :enable
  queueallocrecovery            :enable
  maxqnumalarm                  :enable
  maxqnumrecovery               :enable
-------------------------------------------------------------------------------
PureFlow(A)>
```

（空白ページ）

第12章　統計情報

ここでは，統計情報について説明します。
本装置には，ポート統計情報，シナリオ統計情報があります。

12.1 ポート統計情報

ポート統計情報には，Network ポートカウンタおよびシステムインタフェースカウンタがあります。
この情報は，Network ポートごと，およびシステムインタフェースの統計情報です。

12.1.1 ポートカウンタ

Network ポートごと，およびシステムインタフェースのカウンタです。
ポートカウンタでは，以下の内容を表示します。

- 受信バイト数
- 受信パケット数
- 受信ブロードキャストパケット数
- 受信マルチキャストパケット数
- 送信バイト数
- 送信パケット数
- 送信ブロードキャストパケット数
- 送信マルチキャストパケット数
- 受信エラーパケット数
- Collision（パケットの衝突）発生回数
- 廃棄パケット数
- 受信したパケットの平均レート（単位 kbit/s）
- 送信したパケットの平均レート（単位 kbit/s）

システムインタフェースカウンタでは，以下の内容を表示します。

- 受信バイト数
- 受信パケット数
- 送信バイト数
- 送信パケット数

ポートカウンタに関する CLI は以下のコマンドがあります。

show counter [brief]	すべての Network ポートおよびシステムインタフェースのカウンタを表示します。brief を指定した場合は，概要を表示します。
show counter {<slot/port> | system}	指定 Network ポートまたはシステムインタフェースのカウンタを表示します。
clear counter [<slot/port> | system]	Network ポートシステムインタフェースのカウンタをクリアします。

12.2 シナリオ統計情報

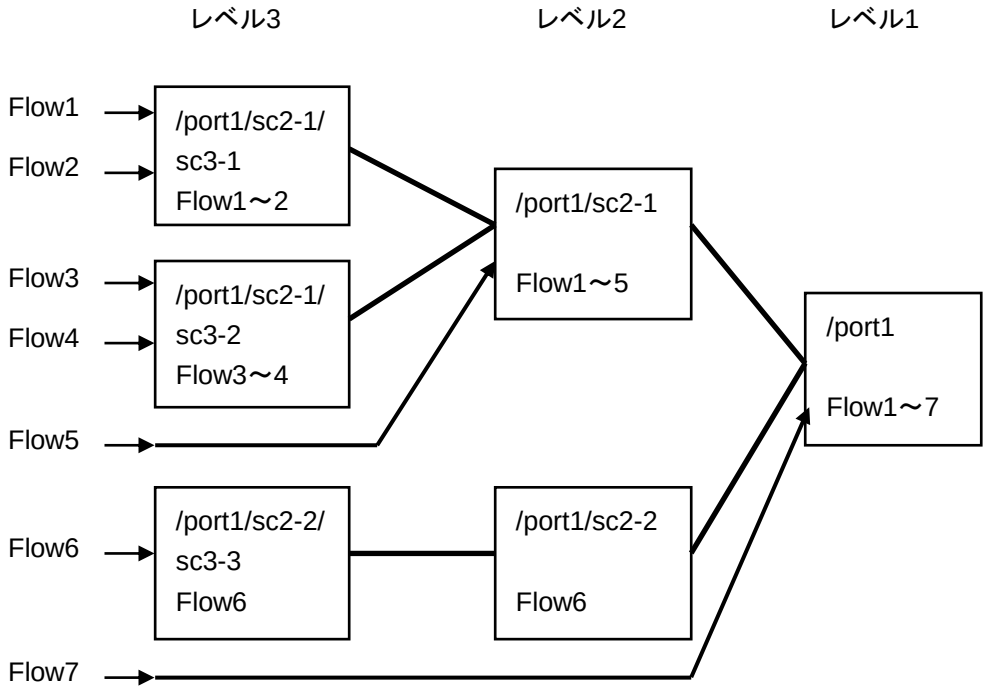
シナリオ統計情報には，シナリオカウンタ，シナリオ動作情報，レート測定があります。
この情報は，シナリオごとの統計情報です。

12.2.1 シナリオカウンタ

シナリオごとのカウンタです。
シナリオカウンタでは，以下の内容を表示します。

- 受信バイト数，受信パケット数
- 送信バイト数，送信パケット数
- 廃棄バイト数，廃棄パケット数

シナリオカウンタは，関連する下位レベルシナリオカウンタを含めた合計値となります。



シナリオカウンタに関する CLI は以下のコマンドがあります。

show scenario counter name <scenario_name>	シナリオのカウンタを表示します。
show scenario counter summary	シナリオのカウンタを一覧で表示します。
clear scenario counter name <scenario_name>	シナリオのカウンタをクリアします。
clear scenario counter all	すべてのシナリオのカウンタをクリアします。

<scenario_name>は，“add scenario”コマンドで指定したシナリオ名を指定します。

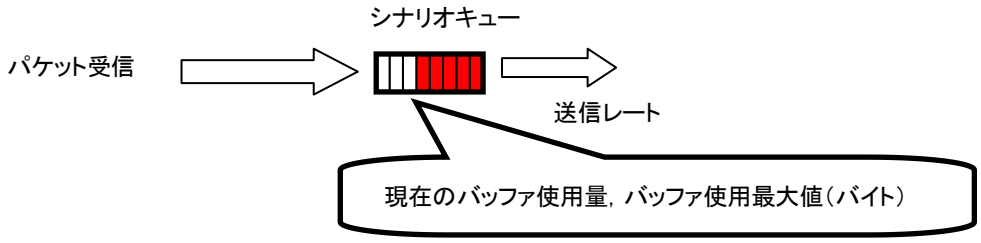
12.2.2 シナリオ動作情報

シナリオごとの動作情報です。
シナリオ動作情報では，以下の内容を表示します。

＜シナリオのデフォルトキューに関する情報＞
- バッファ使用量とバッファ使用率
- バッファピークホールド（バッファ使用最大値）
- フロー数（forward シナリオの場合は総フロー数と同じ値を表示します）

＜シナリオの個別キューに関する情報（個別キューモードシナリオのみ）＞
- 個別キュー数
- バッファ使用量とバッファ使用率（現在のバッファ使用量の最大／最小／平均値も表示）
- バッファピークホールド（バッファ使用最大値）
- 今までに割り当てた個別キューの中で，バッファ使用最大値が最大の個別キュー

＜シナリオの送信レートに関する情報＞
- 送信ピークレート（直近 1 分間の最大送信レート）
- 送信平均レート（直近 1 分間の平均送信レート）
- シナリオに関連する総フロー数



シナリオ動作情報に関する CLI は以下のコマンドがあります。

show scenario info name <scenario_name>	シナリオに関する動作情報を表示します。
show scenario info summary	シナリオに関する動作情報を一覧で表示します。
clear scenario peakhold buffer name <scenario_name>	シナリオに関するバッファ使用最大値をクリアします。
clear scenario peakhold buffer all	すべてのシナリオのバッファ使用最大値をクリアします。

<scenario_name>は，“add scenario”コマンドで指定したシナリオ名を指定します。

注：

モニタリングマネージャ 2 V1.1.1 において，送信レート情報は表示されません。送信レート情報を表示する場合はモニタリングマネージャ 2 V1.2.1 以降を使用してください。

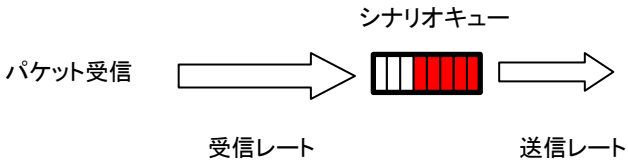
注：

個別キューモードシナリオにおいては，個別キューのバッファ情報は表示されません。キュー最大数超

過時転送キューのバッファ情報が表示されます。
次ページ以降の測定は集約モードシナリオで実施してください。

12.2.3 レート測定

シナリオの受信／送信レートを測定します。受信／送信レートは，約 1 秒ごとに測定を行い，指定回数分表示します。
表示単位は kbit/s で，小数点以下 3 桁まで表示します。また，受信／送信レートの測定は，パケットのみを対象とし，フレーム間ギャップとプリアンブルを含みません。



レート測定に関する CLI は以下のコマンドがあります。

monitor rate <scenario_name> [<num>]	シナリオの受信／送信レートを測定します。

コマンドの実行例を示します。

```
PureFlow(A)> monitor rate /port1/Tokyo 3
Scenario Name : "/port1/Tokyo"

  Times[s]          Rcv Rate[kbps]          Trs Rate[kbps]
  --------------------------------------------------------------------------
         1              3587.562                1254.531
         2              3482.826                1198.426
         3              3624.692                1217.879
  --------------------------------------------------------------------------
  Average               3565.026                1223.612
PureFlow(A)>
```

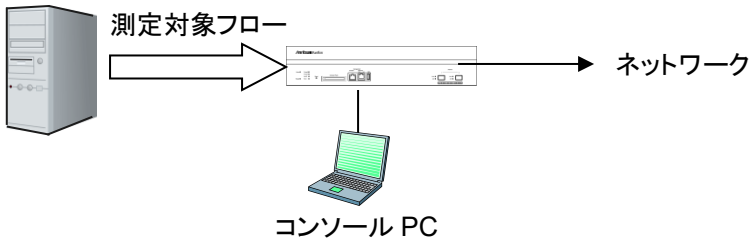
注）CLI 中の"bps"は bit/s を表します。

12.2.4 シナリオパラメータ決定方法

シナリオ統計情報を用いることで，シナリオの平均レート，バーストサイズを測定し，パラメータ決定の参考にすることができます。以下に，決定方法を説明します。

STEP1　レート測定機能を用いた平均レートの測定方法

レート測定するためには，シナリオを割り当てる必要があります。測定対象フローに対し，シナリオとフィルタを設定します。



まず，測定用のシナリオをレベル 2 にバッファサイズ 100 Mbyte（設定可能最大値）で設定します。測定用シナリオに，測定対象フローのみがヒットするフィルタを設定します。

設定例：

```
PureFlow(A)> add scenario /port1/measscenario action aggregate bufsize 100M
PureFlow(A)> add filter scenario /port1/measscenario filter measflow ipv4 sip 192.168.10.9
```

実際にフローを流し，測定シナリオに対してレート測定を実行します。

```
PureFlow(A)> monitor rate /port1/measscenario 3
Scenario Name : "/port1/measscenario"

  Times[s]          Rcv Rate[kbps]          Trs Rate[kbps]
----------------------------------------------------------------------------
         1               3587.562                3587.562
         2               3482.826                3482.826
         3               3624.692                3624.692
----------------------------------------------------------------------------
   Average               3565.026                3565.026
PureFlow(A)>
```

注）CLI 中の"bps"は bit/s を表します。

レート測定の結果，平均受信レートが約 3.6 Mbit/s であることが分かります。

STEP2　バッファピークホールドを用いたバッファ使用最大値の測定方法

次にバッファサイズを決定するためにバーストサイズの測定を行います。STEP1 の測定により得られた平均受信レートに 10%程度のマージンを加えたレートをトラフィックアトリビュートに再設定します。
下記の例では，4 Mbit/s のレートをトラフィックアトリビュートに再設定しています。

```
PureFlow(A)> update scenario /port1/measscenario action aggregate peak_bw 4M
```

次にフローを流している状態で，バッファ使用最大値をクリアします。

```
PureFlow(A)> clear scenario peakhold buffer name /port1/measscenario
```

この状態で，バッファ使用最大値の再記録が行われます。通常の映像トラフィックであれば 1 分程度で映像のバーストサイズがバッファ使用最大値として記録されます。
記録されたバッファ使用最大値を以下のように表示させます。

```
PureFlow(A)> show scenario info name /port1/measscenario
Scenario 1:  "/port1/measscenario"
  Rate Control Unit:
            Create Mode          :Aggregate
            Class                :2
            Min Bandwidth        :-----
            Peak Bandwidth       :4M[bps]
  Default Queue:
            Class                :8
            Buf Size             :100M[Bytes]

  Attached Filters:
            "measflow"

Scenario Rate Information
  Recent interval Tx peak        :0[bps]
  Recent interval Tx average     :0[bps]

Default Queue Information
  Buffer Utilization
            Current              :105384( 10%)[Bytes(%)]
            Peak Hold            :149504( 14%)[Bytes(%)]
  Related Flow
            Flow Num             :1[flows]
PureFlow(A)>
```

バッファ使用最大値が 149504 バイトであることが分かります。測定により得られたバッファ使用最大値に安全率 2 を与え，bufsize を 300000 バイトとします。

```
PureFlow(A)> update scenario /port1/measscenario action aggregate bufsize 300000
```

以上で，対象フローへのトラフィックアトリビュートは下記の値となります。

PeakBandwidth： 4 Mbit/s
BufSize　：　　300000 bytes

注:
安全率は，ネットワーク環境やトラフィックにより適性値を与えてください。

（空白ページ）

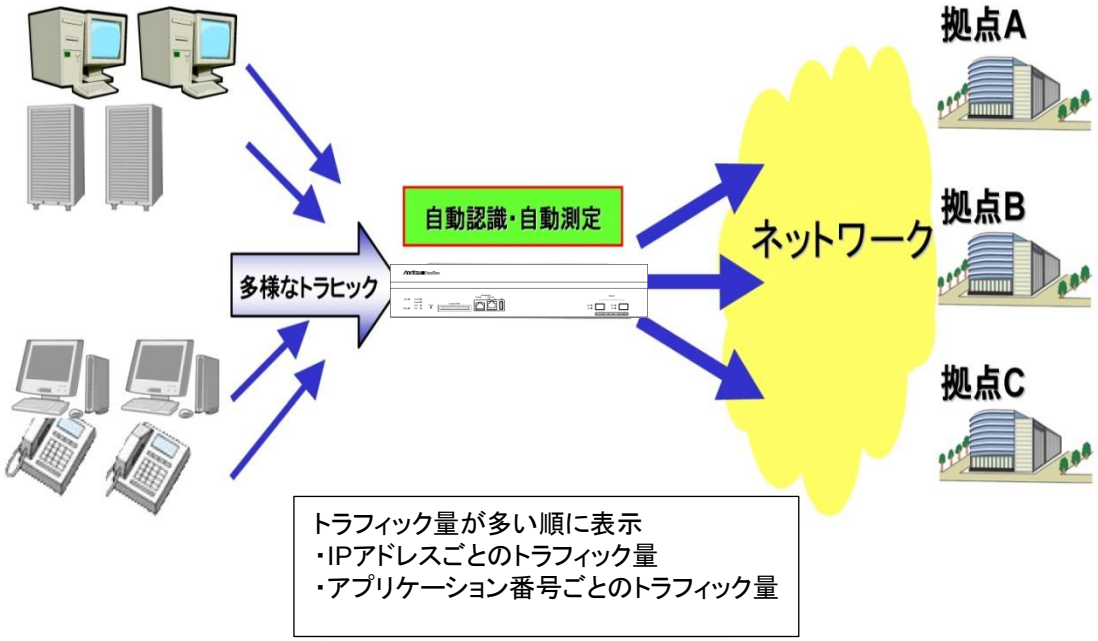
第13章　トップカウンタ機能

ここでは，トップカウンタ機能について説明します。

13.1 概要

トップカウンタ機能は，トラフィックの利用状況を把握するための機能です。この機能は，IP アドレスごとまたはアプリケーションポート番号ごとにトラフィック量を自動認識，流量測定し，トラフィック量が多い順に上位 25 位までのトラフィック量を表示します。

また，モニタリングマネージャ 2 を使用することにより，利用状況をリアルタイムにグラフ表示し，過去のデータを含めたレポートを作成することができます。詳細はモニタリングマネージャ 2 の取扱説明書を参照してください。



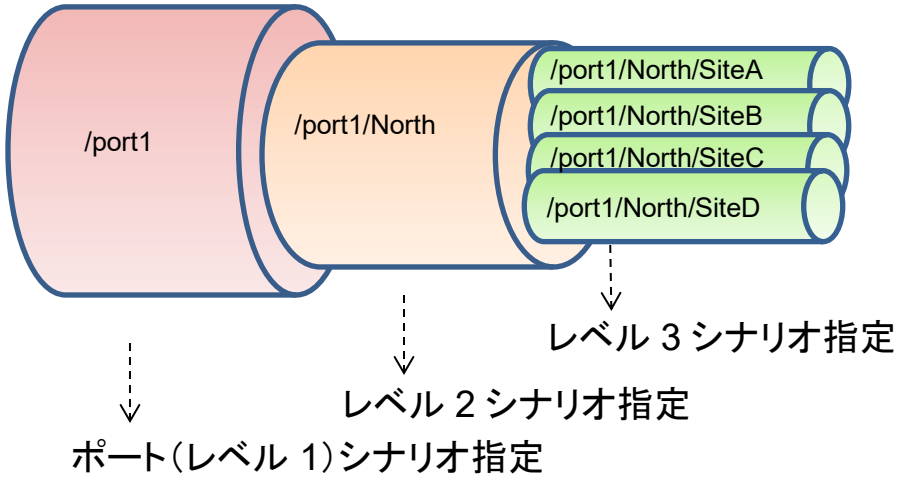
13.2 トップカウンタの表示単位について

トップカウンタ機能は，以下の 4 種の表示単位でトラフィックを計測し，それぞれの表示単位ごとに，上位 25 位までのトラフィック量を表示します。

- 送信元 IP アドレス（SIP）
- 宛先 IP アドレス（DIP）
- 送信元 IP アドレスと宛先 IP アドレスの組（SIP_DIP）
- アプリケーションポート番号（APPLI）

13.3 トップカウンタの測定範囲について

トップカウンタ機能は，本装置を通過する全トラフィックの中から，トップカウンタを測定する範囲を指定することができます。測定範囲として，任意のシナリオを指定でき，最大で 200 個まで登録できます。

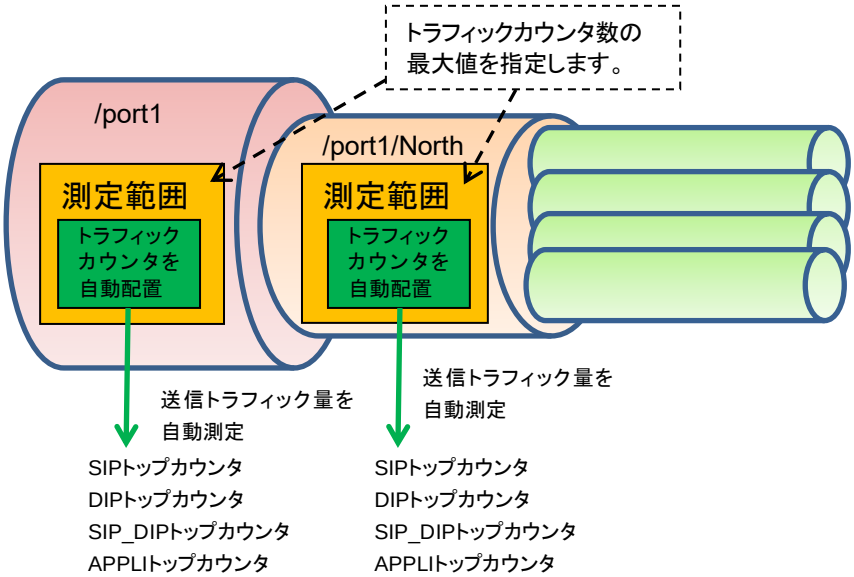


たとえば，あるレベル n シナリオを通過するトラフィックにおいて，通信帯域をより多く消費しているトラフィックを観測する場合は，測定範囲に該当するレベル n シナリオを指定します。これにより，シナリオに入力されたトラフィックの中から，送出量が最も多いトラフィックを把握することができます。

13.4 トラフィックカウンタについて

トラフィックカウンタは，トラフィックの IP アドレスやアプリケーションポート番号ごとなど，自動認識したトラフィックごとに自動配置され，それぞれの送信トラフィック量を測定するカウンタです。

トップカウンタ機能を使用する場合，あらかじめ，利用可能なトラフィックカウンタの最大数をそれぞれの測定範囲ごとに指定する必要があります。トラフィックカウンタの総数は，全測定対象合計で 1,000,000 個までです。



13.5 アプリケーションポート番号の測定について

トップカウンタ機能は，特定のアプリケーションポート番号だけにトラフィックカウンタを割り当て，トラフィック量を測定します。公知のアプリケーションについては測定を実施するようにデフォルトで登録済です。デフォルト状態で測定を実施するアプリケーションポート番号は，“show topcounter config all”コマンドで確認してください。

また，任意のアプリケーションポート番号も測定することができます。測定したいアプリケーションポート番号を“add topcounter config appli port”コマンドで追加してください。

アプリケーションポート番号の測定では，任意のアプリケーションを常時監視することもできます。常時監視に指定すると，当該アプリケーションポート番号のトラフィックカウンタを固定的に確保します。また，その実際の順位が上位 25 位以内でなくても，“show topcounter target”コマンドの測定結果に常に表示します。常時監視したいアプリケーションポート番号は，測定範囲（シナリオ）ごとに，“add topcounter config appli port static”コマンドで登録してください。

13.6 操作コマンド一覧

トップカウンタ機能の操作は，以下のコマンドで行います。

set topcounter	トップカウンタの有効／無効を設定します。
set topcounter config interval time	トップカウンタの収集周期を設定します。
add topcounter target	トップカウンタの測定範囲を追加します。
update topcounter target	トップカウンタの測定範囲に指定されているパラメータを変更します。
delete topcounter target	トップカウンタの測定範囲を削除します。
show topcounter config	トップカウンタの設定を表示します。
show topcounter target	トップカウンタを表示します。
add topcounter config appli port	トップカウンタを測定するアプリケーションポート番号を追加します。
delete topcounter config appli port	トップカウンタを測定するアプリケーションポート番号を削除します。
add topcounter config appli port static	常時監視するアプリケーションポート番号を登録します。
delete topcounter config appli port static	常時監視するアプリケーションポート番号を削除します。

13.7 操作手順

トップカウンタ機能を使用するための操作手順は以下のとおりです。

(1) トップカウンタの測定範囲を設定する。
“add topcounter target” コマンドを使用し，トップカウンタを測定するトラフィックを指定してください。測定範囲として，任意のシナリオのトラフィックを指定することができます。

(2) 必要に応じて，トップカウンタの収集周期を設定する。
“set topcounter config interval time”コマンドを使用し，トップカウンタの収集周期を変更することができます。ただし，モニタリングマネージャ 2 を接続している場合，収集周期が変更される場合があります(「13.9 注意事項(2)」参照)。動作中の収集周期は，“show topcounter config”コマンドで確認することができます。

(3) 必要に応じて，トップカウンタで測定するアプリケーションポート番号を追加する。
デフォルト設定以外のアプリケーションポート番号を測定する場合は，“add topcounter config appli port”コマンドを使用し，任意のポート番号を追加することができます。デフォルト設定のポート番号は，“show topcounter config all”コマンドで確認することができます。

(4) 必要に応じて，常時監視するアプリケーションポート番号を登録する。
任意のアプリケーションポート番号を“add topcounter config appli port static”コマンドを使用し，常時監視するように登録することができます。常時監視するアプリケーションポート番号の登録は測定範囲(シナリオ)ごとに行ってください。

(5) トップカウンタの収集を有効にする。
“set topcounter enable”コマンドを使用し，トップカウンタ機能を有効にしてください。トップカウンタ機能が有効になってから，収集周期が経過した後，次の(6)でトップカウンタを表示します。

(6) トップカウンタを表示する。
“show topcounter target”コマンドを使用し，トップカウンタを表示します。送信元 IP アドレスごと，宛先 IP アドレスごと，送信元 IP アドレスと宛先 IP アドレスの組み合わせごと，アプリケーションポート番号ごとなど，それぞれのトップカウンタを表示することができます。

13.8 操作例

以下の表に示す設定で，トップカウンタ機能を使用するときのコマンド設定例を記載します。

ユーザ設定項目	設定値	備考
測定範囲	Network ポート 1/1 /port1	トラフィックカウンタ数を任意に設定
	レベル 2 シナリオ /port1/North	トラフィックカウンタ数をデフォルト設定
	レベル 3 シナリオ /port1/North/SiteA	トラフィックカウンタ数をデフォルト設定
収集周期	5 分	モニタリングマネージャ 2 を接続した場合，収集周期が変更される場合があります（「13.9 注意事項(2)」参照）。
アプリケーションポート番号	測定するアプリケーションポート番号を追加 10000 20000～20003	デフォルト設定のアプリケーションポート番号に加えて，10000，20000，20001，20002，20003 のアプリケーションポート番号を測定する。
	常時監視するアプリケーションポート番号を登録 シナリオ /port1 ポート番号 80	HTTP（ポート番号 80）トラフィックを常時監視する。

設定コマンドは，以下のとおりです。

```
PureFlow(A)> add topcounter target scenario /port1 sip 10000 dip 10000 sip_dip 10000
             appli 250
PureFlow(A)> add topcounter target scenario /port1/North
PureFlow(A)> add topcounter target scenario /port1/North/SiteA
PureFlow(A)> set topcounter config interval time 5
PureFlow(A)> add topcounter config appli port 10000
PureFlow(A)> add topcounter config appli port 20000-20003
PureFlow(A)> add topcounter config appli port static /port1 80
PureFlow(A)> set topcounter enable
PureFlow(A)>
```

トップカウンタは，以下のように表示されます。

```
PureFlow(A)> show topcounter target scenario /port1 group sip
From          : 2013 Jan 02 19:47:55   To            : 2013 Jan 02 19:57:55
Total Octet: 1475806000                Total Packet: 1475806

Order   IP Address                                  Tx Octet          Tx Packet
-----   ----------------------------------------    ----------------  ----------------
    1   192.168.101.121                                     8214               111
    2   192.168.101.122                                     5846                79
    3   fe80:0000:0000:0000:0290:ccff:fe22:8b4c             5772                78
    4   fe80:0000:0000:0000:0290:ccff:fe22:8b4d             5698                77
    5   fe80:0000:0000:0000:0290:ccff:fe22:8b4e             3848                52
PureFlow(A)>

PureFlow(A)> show topcounter target scenario /port1 group appli
From          : 2013 Jan 02 19:47:55   To            : 2013 Jan 02 19:57:55
Total Octet: 1475806000                Total Packet: 1475806

Order   TCP/UDP Port                    Type            Tx Octet          Tx Packet
-----   ---------------------------     ----------      ----------------  ----------------
    1                        10000                             22625               276
    2                        20000                              1288                46
    3                        20001                               446                12
    4                        20002                               446                12
    5                        20003                               240                20
    6                           80   static                       0                 0
PureFlow(A)>
```

13.9 注意事項

(1) トラフィックカウンタが不足した場合，正確なトップカウンタを表示しない場合があります。
割り当てたトラフィックカウンタの数よりも，実際に通信している通信ノードが多い場合，トラフィックカウンタが不足する場合があります。トラフィックカウンタが割り当てられていない通信ノードは，個別の流量を測定することができないため，トップカウンタとして表示されません。

(2) モニタリングマネージャ 2 を使用している場合，CLI で設定した収集周期とは異なる周期でトップカウンタを集計する場合があります。
本装置にモニタリングマネージャ 2 が接続された場合，トップカウンタの収集周期がモニタリングマネージャによって変更される場合があります。CLI で設定された収集周期と，モニタリングマネージャ 2 の GUI で設定された収集周期を比較し，より長いほうの周期でトップカウンタを収集します。動作中の収集周期は，“show topcounter config”コマンドで確認してください。

(3) モニタリングマネージャ 2 V1.1.1 ではトップカウンタ情報は表示されません。
トップカウンタ情報を表示するにはモニタリングマネージャ 2 V1.2.1 以降を使用してください。

(4) 受信した TCP/IP パケットにおいて，送信元ポート番号と宛先ポート番号の両方が，トップカウンタを測定するアプリケーションポート番号として登録されている場合，そのパケットは，宛先ポート番号のトラフィックカウンタに計上されます。送信元ポート番号のトラフィックカウンタには計上されません。

(5) トップカウンタを測定するアプリケーションポート番号を必要に応じて追加できますが，デフォルトで設定されているアプリケーションポート番号は削除できません。

(6) CLI またはモニタリングマネージャ 2 からトップカウンタの収集周期を変更した場合，一度だけ，設定されている収集周期よりも短い期間で集計されたトップカウンタを表示する場合があります。これは，前回の収集周期に達した時刻から，収集周期を変更した時刻までのトップカウンタの集計結果です。

(7) トップカウンタは，トップカウンタの収集周期に到達してから約 1 分経過したときに更新されます。

(8) トップカウンタの収集周期が 1 分の場合は，全測定対象合計は 100,000 個までに制限されます。

(9) トップカウンタ有効時に測定範囲を追加した場合，追加した対象シナリオは，即座に測定対象にはなりません。即座に測定を行いたい場合は，トップカウンタを無効にしてから測定範囲を追加し，トップカウンタを有効にしてください。

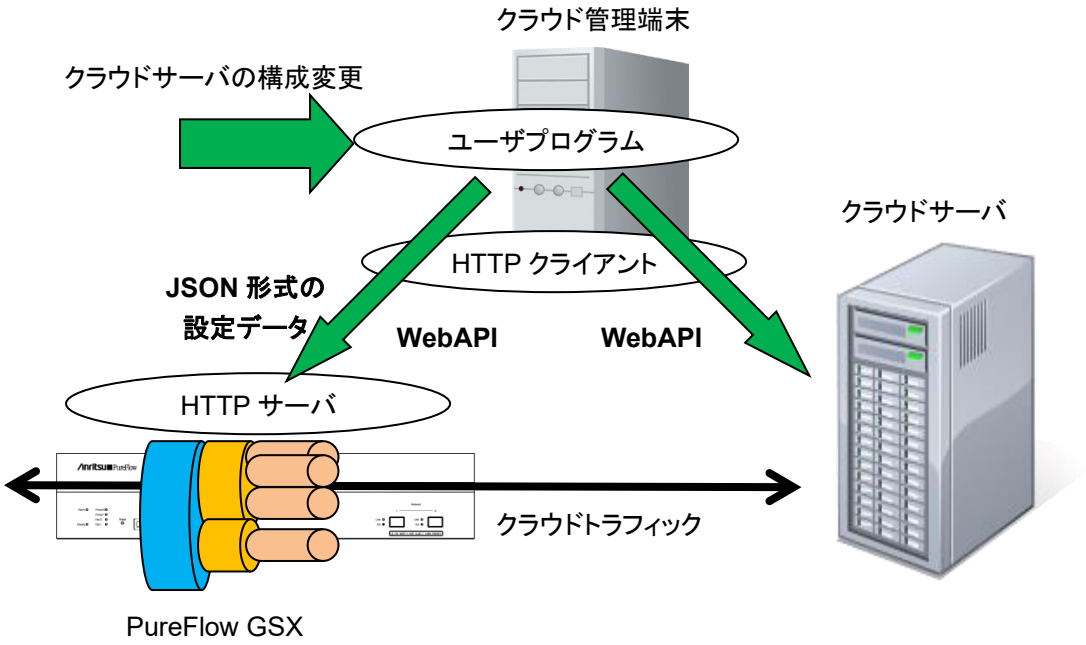
第14章　WebAPI 機能

ここでは，WebAPI（Web Application Program Interface）機能について説明します。

14.1 概要

WebAPI 機能は，本装置のトラフィックコントロール機能の設定を行う際に，HTTP（Hypertext Transfer Protocol：RFC2616）を使用して設定を行う機能です。本装置は，HTTP サーバとして動作し，外部に設置した管理端末の HTTP クライアントから JSON（JavaScript Object Notation：RFC4627）形式で設定を行うことができます。

クラウド環境において，クラウドサーバの構成変更に連動して手動で帯域制御装置のトラフィックコントロール設定を更新することは困難になってきています。クラウド管理端末上で JSON 形式をサポートしたプログラミング言語を利用し，クラウドサーバの構成変更に連動して本装置のトラフィックコントロール設定を更新するユーザプログラムを作成することにより，本装置の設定更新を自動化することができます。



また，SSL 暗号化通信による HTTP 接続（HTTPS: Hypertext Transfer Secure）を使用することができます。HTTPS では WebAPI の通信が暗号化され，盗聴やなりすましを防ぐことができます。

WebAPI は同時に 4 セッションまで実行可能です。
同時に 5 セッション以上の WebAPI を実行した場合，5 セッション以上の接続は可能ですが，要求発行時にいずれかのセッションでエラーが発生します。例えば，セッション 1～4 で WebAPI を実行中に 5 セッション目の要求を発行すると，セッション 1～5 のいずれかでセッション数超過エラーやコネクションの切断が発生します。WebAPI は 4 セッション以内でご利用ください。なお，セッション数には Telnet セッションと SSH セッションのセッション数は含みません。

HTTP リクエストと次の HTTP リクエストまでのタイムアウト時間は 15 秒です。

14.2 通信プロトコル

WebAPI 機能では通信プロトコルとして HTTP または HTTPS を使用します。通信プロトコルの設定には以下のコマンドを使用します。

set webapi protocol {normalhttp | httpsecure}	WebAPI で使用する通信プロトコルを設定します。デフォルトは normalhttp です。 normalhttp： HTTP を使用します。 httpsecure： HTTPS を使用します。 HTTP と HTTPS の同時利用はできません。
show webapi	WebAPI の設定を表示します。

14.3 HTTP メソッド

WebAPI 機能がサポートする HTTP メソッドは以下のとおりです。

HTTP メソッド	用途
HEAD	アクセス可否の判断等に使用されます。
GET	情報の取得に使用されます。 本装置では情報取得系の要求で使用します。
POST	情報の設定に使用されます。 本装置では追加，更新，削除系の要求で使用します。

なお，HTTP クライアントから上記以外のメソッドが指定された場合，HTTP ステータスコード 405（Method Not Allowed）を返します。

14.4 JSON 形式

WebAPI 機能は GET や POST メソッドで JSON 形式のデータを利用します。JSON とはデータを表現するためのデータ記述言語です。JSON の記述方法では，パラメータのキーと値の組をコロン":"でペアにします。パラメータが複数ある場合はコンマ", "で区切ります。これらの全体を中括弧"{"および"}"で括ります。

WebAPI 機能ではキーや値はすべて文字列で記述してください。API の種別を示すキー"command"と，API に相当する CLI コマンドのパラメータを指定します。WebAPI においてはキーの記述順序は順不同です。CLI コマンドのパラメータ順序と合わせる必要はありません。

下記にシナリオ追加 API の JSON 記述例を示します。

```
{
        “command”: ”add scenario”,
        "scenario_name": "/port1/North",
        "action": "aggregate",
        "min_bandwidth": "5M",
        "peak_bandwidth": "8M",
        "class": "2",
        "bufsize": "512k"
}
```

JSON の記述方法の詳細については「付録 E　JSON の記述方法」を参照してください。

14.5 API一覧

WebAPIは，シナリオ，フィルタ，ルールリストに関する設定，および情報取得のAPIを提供します。それぞれの機能は，相当するCLIコマンドと同等です。APIで指定するパラメータ，値の範囲や省略可能／不可についても同等です。各APIの詳細については「付録F　WebAPI詳細」を参照してください。

対象	操作	相当するCLIコマンド
シナリオ	追加	add scenario
	更新	update scenario
	削除	delete scenario
	情報取得	show scenario
フィルタ	追加	add filter
	削除	delete filter
	情報取得	show filter
ルールリスト	グループ追加	add rulelist group
	グループ削除	delete rulelist group
	エントリ追加	add rulelist entry
	エントリ削除	delete rulelist entry
	情報取得	show rulelist
コンフィギュレーション	保存	save config
	情報取得	show save status※

※コンフィギュレーションの情報取得 APIは，コンフィギュレーションの保存が実行中であるかどうかのステータスを取得するAPIです。コンフィギュレーションの保存が実行中である間はコンフィギュレーションの保存を重複して実行できません。保存の所要時間については「第3章　設定の基本」を参照してください。

14.6 共通エラーメッセージ

HTTPメソッド，JSONフォーマットおよび指定内容が正しい場合，HTTPステータスコード200（OK）に加えて，"status": "OK."　を返します。HTTPメソッドおよびJSONフォーマットが正しいが，指定内容が不正な場合，HTTPステータスコード200（OK）に加えてエラーメッセージを返します。共通エラーメッセージは以下のとおりです。

エラーメッセージ	説明
Specified command is invalid.	APIコマンドが不正です。
Required parameter is not specified.	必須のパラメータが指定されていません。
Specified command is invalid when GET request.	GETメソッドでは指定できないコマンド（追加・更新・削除）です。
Specified command is invalid when POST request.	POSTメソッドでは指定できないコマンド（情報取得）です。
WebAPI session is full.	WebAPIの最大セッション数が超過しました。
Failed to create pipe.	内部通信用のPIPE作成でエラーが発生しました。
No response message from LR.	内部通信で応答がありません。

14.7 エラーメッセージ一覧

各 API 個別のエラーメッセージは以下のとおりです。

API	エラーメッセージ
シナリオ追加	Specified Scenario Class is invalid. It must be either of 1,2,3,4,5,6,7,8. ・class の指定が不正です。
	Specified Minimum Bandwidth is invalid. (Valid from 0, 1k to 10G) ・Minimum Bandwidth の指定が不正です。
	Specified Peak Bandwidth is invalid. (Valid from 2k to 10G) ・Peak Bandwidth の指定が不正です。
	Peak Bandwidth should be greater than Minimum Bandwidth. ・peak_bandwidth は min_bandwidth 以上に設定する必要があります。
	Specified Buff Size is invalid. (Valid from 2k to 100M) ・bufsize の指定が不正です。
	Specified Scenario Name is invalid. ・シナリオ名の指定が不正です。
	Specified Scenario Name is already used. ・指定のシナリオ名はすでに別のシナリオで使われています。
	Specified Scenario of upper level hierarchy is not found. ・上位階層のシナリオが存在しません。
	maximum number of scenario was exceeded. ・シナリオの最大登録件数を超えました。
	Could not Add the Scenario. ・シナリオが登録できません。
	Specified Scenario ID is invalid. (Valid from 1 to 40000) ・シナリオインデックスが範囲外です。
	Specified Scenario ID is already used. ・指定のシナリオインデックスはすでに別のシナリオで使われています。
	Specified Max Q Num is invalid. (Valid from 1 to 300000) ・maxquenum が範囲外です。
	Specified Q Division Field is invalid. Valid fields: default, vlan, cos, inner-vlan, inner-cos, ethertype, sip, dip, tos, proto, sport, dport ・quedivision のフィールド指定が不正です。
	failaction is not specified. ・failaction の指定せずに fail_min_bw, fail_peak_bw, fail_class を設定することはできません。
	Specified Failaction is invalid. ・ fail_min_bw, fail_peak_bw, fail_class は failaction として forwardattribute を指定した場合のみ設定可能です。

API	エラーメッセージ
シナリオ追加(続き)	Specified scenario has packets in buffer. Please wait until the buffer becomes empty, and try again. ・指定のシナリオはパケットの送出中です。送出が完了するまで待ってから，再度実行してください。

API	エラーメッセージ
シナリオ更新	Specified Scenario Name is invalid. ・シナリオ名の指定が不正です。
	Specified scenario name is not used. ・指定シナリオが存在しません。
	Specified Scenario Class is invalid. It must be either of 1,2,3,4,5,6,7,8. ・class の指定が不正です。
	Specified Minimum Bandwidth is invalid. (Valid from 0, 1k to 10G) ・Minimum Bandwidth の指定が不正です。
	Specified Peak Bandwidth is invalid. (Valid from 2k to 10G) ・Peak Bandwidth の指定が不正です。
	Peak Bandwidth should be greater than Minimum Bandwidth. ・peak_bandwidth は min_bandwidth 以上に設定する必要があります。
	Specified Buff Size is invalid. (Valid from 2k to 100M) ・bufsize の指定が不正です。
	It is necessary to set one or more parameters. ・1 つ以上のパラメータを設定する必要があります。
	Specified Scenario Mode is invalid. ・シナリオモードの指定が不正です。
	Could not Update the Scenario. ・シナリオが変更できません。
	Specified Max Q Num is invalid. (Valid from 1 to 300000) ・maxquenum が範囲外です。
	Extended number of scenario is not licensed. ・シナリオ拡張ライセンスの制限数を超えてシナリオを登録することはできません。 ・シナリオ拡張ライセンスの制限数を超えた maxquenum を設定することはできません。
	Specified Q Division Field is invalid. Valid fields: default, vlan, cos, inner-vlan, inner-cos, ethertype, sip, dip, tos, proto, sport, dport ・quedivision のフィールド指定が不正です。
	Specified Failaction is invalid. ・ fail_min_bw, fail_peak_bw, fail_class は failaction として forwardattribute を指定した場合のみ設定可能です。

API	エラーメッセージ
シナリオ削除	Specified Scenario Name is invalid. ・シナリオ名の指定が不正です。
	Specified scenario name is not used. ・指定シナリオが存在しません。
	Down level hierarchy scenario exists. ・下位階層のシナリオが存在します。
	Could not Delete the Scenario. ・シナリオが削除できません。

API	エラーメッセージ
シナリオ情報取得	Specified Scenario Name is invalid. ・シナリオ名の指定が不正です。
	Specified scenario name is not used. ・指定シナリオが存在しません。

API	エラーメッセージ
フィルタ追加	Specified Scenario Name is invalid. ・シナリオ名の指定が不正です。
	Specified scenario name is not used. ・指定シナリオが存在しません。
	Specified filter Name is invalid. (Number only cannot be specified. "all" cannot be specified.) (Valid Filter Name length is from 1 to 48.) ・フィルタ名の指定が不正です。
	Specified filter Name is already used. ・指定のフィルタ名はすでに別のフィルタで使われています。
	Specified Ether type is invalid. (Valid from 0x0000 to 0xFFFF) ・Ether type の指定が不正です。
	Specified vid is invalid. (Valid from 0 to 4094, Or Start - End) ・VLAN ID の指定が不正です。
	Specified cos is invalid. (Valid from 0 to 7, Or Start - End) ・CoS 値の指定が不正です。
	Specified inner-vid is invalid. (Valid from 0 to 4094, Or Start - End) ・VLAN ID の指定が不正です。
	Specified inner-cos is invalid. (Valid from 0 to 7, Or Start - End) ・CoS 値の指定が不正です。
	The format or value of the specified source IP address is invalid. ・Source IP address の指定が不正です。

API	エラーメッセージ
フィルタ追加（続き）	The format or value of the specified destination IP address is invalid. ・Destination IP address の指定が不正です。
	The format or value of the specified source IPv6 address is invalid. ・Source IPv6 address の指定が不正です。
	The format or value of the specified destination IPv6 address is invalid. ・Destination IPv6 address の指定が不正です。
	Specified rulelist name of source IP address is invalid. Specified rulelist name of destination IP address is invalid. Specified rulelist name of source port is invalid. Specified rulelist name of destination port is invalid. ・ルールリスト名が不正です。
	Specified rulelist name of source IP address is not used. Specified rulelist name of destination IP address is not used. Specified rulelist name of source port is not used. Specified rulelist name of destination port is not used. ・指定ルールリストが存在しません。
	IP Filter and rulelist of source IP address is not same type. IP Filter and rulelist of destination IP address is not same type. IP Filter and rulelist of source port is not same type. IP Filter and rulelist of destination port is not same type. ・対象ルールリストと種別が異なります。
	Specified ToS is invalid. (Valid from 0 to 255, Or Start - End) ・ToS 値　または　Traffic Class 値の指定が不正です。
	Specified protocol number is invalid. (Valid from 0 to 255, Start - End, Or tcp/udp/icmp) ・プロトコル番号の指定が不正です。
	Specified Source TCP/UDP port number is invalid. (Valid from 0 to 65535. Or Start - End) ・sport 番号の指定が不正です。
	Specified Destination TCP/UDP port number is invalid. (Valid from 0 to 65535. Or Start - End) ・dport 番号の指定が不正です。
	Specified Filter Priority is invalid. (Valid from 1 to 40000, Or Start - End) ・フィルタ優先度の指定が不正です。
	maximum number of filter was exceeded. ・フィルタの最大登録件数を超えました。
	It is necessary to set one or more parameters other than Priority. ・Ethernetフィルタは Priority 以外で少なくとも 1 つのパラメータを指定する必要があります。
	Could not Add the Filter. ・フィルタが登録できません。

API	エラーメッセージ
フィルタ削除	Specified scenario name is invalid. ・シナリオ名の指定が不正です。
	Specified scenario name is not used. ・指定シナリオが存在しません。
	Specified filter name is invalid. (Number only cannot be specified. "all" cannot be specified.) (Valid Filter Name length is from 1 to 48.) ・フィルタ名の指定が不正です。
	Specified filter name is not used. ・指定フィルタが存在しません。
	Could not Delete the Filter. ・フィルタが削除できません。

API	エラーメッセージ
フィルタ情報取得	Specified scenario name is invalid. ・シナリオ名の指定が不正です。
	Specified scenario name is not used. ・指定シナリオが存在しません。
	Specified filter name is invalid. (Number only cannot be specified. "all" cannot be specified.) (Valid Filter Name length is from 1 to 48.) ・フィルタ名の指定が不正です。
	Specified filter name is not used. ・指定フィルタが存在しません。

API	エラーメッセージ
ルールリストグループ追加	Specified rulelist name is invalid. (Number only cannot be specified. "all" cannot be specfied.) (Valid rulename length is from 1 to 32.) ・ルールリスト名が不正です。
	Specified rulelist name is already in use. ・同一名のルールリストがすでに存在します。
	Maximum number of rulelist was exceeded. ・ルールリストの最大登録件数を超えました。
	Could not add the rulelist. ・ルールリストが登録できません。

API	エラーメッセージ
ルールリストグループ削除	Specified rulelist name is invalid. (Number only cannot be specified. "all" cannot be specfied.) (Valid rulename length is from 1 to 32.) ・ルールリスト名が不正です。
	Specified rulelist name is not used. ・指定ルールリストが存在しません。
	Rulelist is used by filter. ・ルールリストがフィルタに設定されています。
	Could not delete the rulelist. ・ルールリストを削除できません。

API	エラーメッセージ
ルールリストエントリ追加	Specified rulelist name is invalid. (Number only cannot be specified. "all" cannot be specfied.) (Valid rulename length is from 1 to 32.) ・ルールリスト名が不正です。
	Specified rulelist name is not used. ・指定ルールリストが存在しません。
	The format or value of the specified IP address is invalid. ・IP address の指定が不正です。
	Specified TCP/UDP port number is invalid. (Valid from 0 to 65535. Or Start - End) ・TCP/UDP ポート番号の指定が不正です。
	Maximum number of rulelist entry was exceeded. ・指定ルールリストのルールリストエントリ最大登録件数(512 件)を超えました。
	Maximum number of total rulelist entry was exceeded. ・全ルールリスト合計のルールリストエントリ最大登録件数(64000 件)を超えました。
	Specified rulelist entry is already in use. ・指定ルールリストエントリはすでに登録されています。
	Rulelist entry and rulelist is not same type. ・対象ルールリストと種類が異なります。
	Could not add the rulelist entry. ・ルールリストエントリが登録できません。

API	エラーメッセージ
ルールリストエントリ 削除	Specified rulelist name is invalid. (Number only cannot be specified. "all" cannot be specfied.) (Valid rulename length is from 1 to 32.) ・ルールリスト名が不正です。
	Specified rulelist name is not used. ・指定ルールリストが存在しません。
	The format or value of the specified IP address is invalid. ・IP address の指定が不正です。
	Specified TCP/UDP port number is invalid. (Valid from 0 to 65535. Or Start - End) ・TCP/UDP ポート番号の指定が不正です。
	Rulelist entry and rulelist is not same type. ・対象ルールリストと種別が異なります。
	Specified rulelist entry is not used. ・指定ルールリストエントリが存在しません。
	Could not delete the rulelist entry. ・ルールリストエントリが削除できません。

API	エラーメッセージ
ルールリスト情報取得	Specified rulelist name is invalid. (Number only cannot be specified. "all" cannot be specfied.) (Valid rulename length is from 1 to 32.) ・ルールリスト名が不正です。
	Specified rulelist name is not used. ・指定ルールリストが存在しません。

API	エラーメッセージ
コンフィギュレーション 保存	configuration save is in progress. ・コンフィギュレーション保存中です。

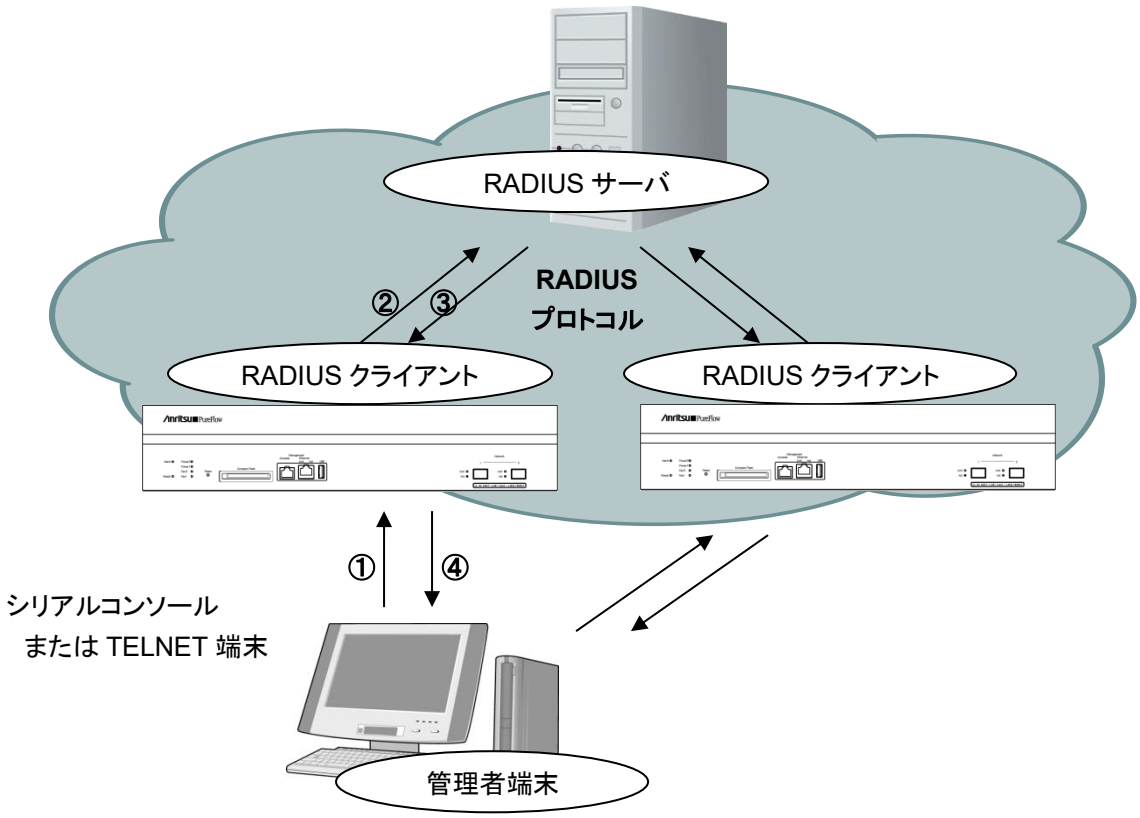
API	エラーメッセージ
コンフィギュレーション 情報取得	なし

第15章　RADIUS 機能

ここでは，RADIUS（Remote Authentication Dial In User Service）機能について説明します。

15.1 概要

RADIUS 機能は，TELNET，SSH，およびシリアルコンソールのログイン時に，RADIUS（RFC2865）を使用してユーザ認証する機能です。本装置は，RADIUS クライアントとして動作し，外部に設置した RADIUS サーバのユーザ情報に基づいたユーザ認証が可能です。



① ユーザが管理者端末からユーザ名とパスワードを入力する。

② PureFlowGSX の RADIUS クライアントから RADIUS サーバに認証要求を送信する。

③ RADIUS サーバから RADIUS クライアントに認証応答を送信する。

④ PureFlowGSX は受信した認証応答に基づいて管理者端末からの接続を許可する。

15.2 ログイン認証の制御

RADIUS機能を有効にした場合のログイン認証の制御について説明します。RADIUS機能が有効な場合と無効な場合におけるログイン認証の制御は以下のとおりです。

RADIUS 認証有効時の ログイン認証手順	RADIUIS 認証無効時の ログイン認証手順
1）　本装置に設定されたユーザ名とログインパスワードでログイン認証を実施します。 2）　ログイン認証が拒否された場合，RADIUS サーバに登録されたユーザ名とログインパスワードでログイン認証を実施します。	1）　本装置に設定されたユーザ名とログインパスワードでログイン認証を実施します。

15.3 ログインモードの制御

本装置は，RADIUS サーバに設定されるユーザごとのサービスタイプに従って，ユーザがログインしたときのログインモードを切り替えます。本装置がサポートするサービスタイプは以下のとおりです。

サービスタイプ	ログインモード
Login-User(1)	normal モード
Administrative-User(6)	administrator モード

なお，RADIUS サーバから上記以外のサービスタイプが指定された場合，Normal モードでログインします。

15.4 RADIUS 機能の設定

RADIUS 認証サーバの情報および認証用パラメータを設定することで RADIUS クライアントとしてユーザ認証することが可能となります。

set radius auth { enable | disable }	RADIUS 認証の有効／無効を設定します。
set radius auth timeout <timeout>	RADIUS 認証応答パケットの受信タイムアウト値を設定します。設定範囲は 1～30[秒]です。デフォルトは 5[秒]です。
set radius auth retransmit <retry>	RADIUS 認証要求パケットの再送信回数を設定します。設定範囲は 0～10[回]です。デフォルトは 3[回]です。
set radius auth method {PAP | CHAP | default}	RADIUS 認証方法を設定します。
add radius auth server <IP_address> [port <port>] key <string > [Primary]	RADIUS 認証サーバを追加します。
update radius auth server <IP_address> [port <port>] [key <string>] [Primary]	すでに存在している RADIUS 認証サーバの設定内容を変更します。
delete radius auth server <IP_address>	RADIUS 認証サーバの設定を削除します。
show radius	RADIUS 設定情報を表示します。

以下に RADIUS 機能の設定例を記述します。

① RADIUS 認証方法を設定します。例では，PAP 認証方式を設定しています。

```
PureFlow(A)> set radius auth method PAP
```

② RADIUS 認証サーバを追加します。例では，2 つのサーバを登録しています。ひとつは，サーバ IP アドレス 192.168.1.10，RADIUS 共有鍵“testing123”で設定しています。もうひとつは，サーバ IP アドレス 192.168.1.11，RADIUS 共有鍵“testing789”で設定しています。 Primary 指定は，最初にログイン認証を問い合わせする RADIUS サーバに設定します。Primary 指定がない場合は，RADIUS サーバが登録された順番にログイン認証を問い合わせします。

```
PureFlow(A)> add radius auth server 192.168.1.10 key testing123 Primary
PureFlow(A)> add radius auth server 192.168.1.11 key testing789
```

③ RADIUS 機能を有効にします。

```
PureFlow(A)> set radius auth enable
```

④　設定内容を確認します。

```
PureFlow(A)> show radius
RADIUS Authentication  : Enable
RADIUS method          : PAP
RADIUS server entries  : 2
Retry retransmit       : 5
Retry timeout          : 3

Type Pri Server              Port    key
---- --- --------------- ----- ------------------------------------------
auth   *   192.168.1.10      1812   "testing123"
auth       192.168.1.11      1812   "testing789"
PureFlow(A)>
```

15.5 RADIUS サーバの設定

RADIUS サーバの設定方法を説明します。RADIUS サーバには，以下のユーザ情報を設定します。

RADIUS 共有鍵

PureFlowGSX に設定した RADIUS 共有鍵と同一の文字列を指定します。

ユーザ ID

ユーザ ID を設定します。

認証方法

PureFlowGSX に設定した認証方法と同じ認証方法（CHAP または PAP）を指定します。

パスワード

パスワードを設定します。

サービスタイプ

このパラメータは必要に応じて設定します。RADIUS サーバからサービスタイプが通知されない場合，PureFlowGSX は normal モードでのログインをユーザに許可します。RADIUS サーバからサービスタイプが通知され，そのサービスタイプが Administrative-User の場合，administrator モードでのログインをユーザに許可します。

本書では，RADIUS サーバとして FreeRADIUS バージョン 1 を使用した場合を説明しますが，実際の設定についてはお使いの RADIUS サーバの種類によって異なる設定が必要となります。また，FreeRADIUS をご利用の場合でも，FreeRADIUS のバージョンによって設定方法が異なります。FreeRADIUS は，LDAP（Lightweight Directory Access Protocol），SQL Server，UNIX システムのユーザ情報などのさまざまなユーザ情報と統合可能であり，企業内の多数のユーザの管理，認証，認可に使用することができます。

（注）
Linux に FreeRADIUS がインストールされていることを前提としています。FreeRADIUS の設定方法および，使用方法の詳細は，インストールされているソフトウエアのマニュアルを参照してください。

FreeRADIUS バージョン 1 の設定方法

① RADIUS 共有鍵の設定

RADIUS サーバに RADIUS クライアントとして登録する装置の IP アドレスおよび, RADIUS 共有鍵を以下の形式で設定します。

RADIUS サーバの/usr/local/etc/raddb/clients.conf ファイルを開き, 適切なセクションに以下の設定を追加してください。

```
client 192.168.37.10 {
        secret =  testing123
        shortname =  gsx
}
```

② ユーザの設定

RADIUS サーバに PureFlowGSX へのログインを許可するユーザ情報を設定します。ユーザごとに, ユーザ ID, 認証方法, パスワード, サービスタイプを設定します。

RADIUS サーバの/usr/local/etc/raddb/users ファイルを開き, 適切なセクションに以下の設定を追加してください。

1） 認証方法に CHAP を使用する場合

normal モードでのログインを許可するユーザの設定

```
user1 Cleartext-Password:=" user1passwd "
        Auth-Type:=CHAP,
        Service-Type=Login-User
```

Administrator モードでのログインを許可するユーザの設定

```
user2 Cleartext-Password:=" user2passwd "
        Auth-Type:=CHAP,
        Service-Type= Administrative-User
```

2） 認証方法に PAP を使用する場合

normal モードでのログインを許可するユーザの設定

```
user3 Cleartext-Password:=" user3passwd "
        Auth-Type:=PAP,
        Service-Type=Login-User
```

Administrator モードでのログインを許可するユーザの設定

```
user4 Cleartext-Password:=" user4passwd "
        Auth-Type:=PAP,
        Service-Type=Administrative-User
```

第16章　ダウンロードとアップロード

ここでは，ソフトウェアやコンフィギュレーションのダウンロード／アップロードについて説明します。

ソフトウェアやコンフィギュレーションをダウンロード／アップロードする場合は，Compact Flash（以下 CF）カードまたは USB メモリを使用します。ファイルシステムは FAT16／FAT32 を対象とします。また，ソフトウェアのダウンロード，コンフィギュレーションのダウンロード／アップロードについては，システムインタフェースから TFTP または FTP により実行することもできます。システムインタフェースを使用する場合には TFTP サーバまたは FTP サーバ機能を備えた PC などを用意してください。

また，CF カードをご使用になる場合，弊社推奨 CF カードをご使用ください。推奨 CF カード以外の動作は保証対象外です。弊社動作確認済の USB メモリの詳細につきましては，取扱説明書をご覧ください。

ソフトウェアやコンフィギュレーションのロードは，Command Line Interface（CLI）を使用します。CLI については，「第 3 章 設定の基本」を参照してください。

16.1 ソフトウェアのダウンロード／アップロード

ソフトウェアをダウンロードするときの注意事項

弊社指定の正規オブジェクトファイル以外をダウンロードすると，装置が起動しません。上記のコマンドで正規のオブジェクトファイル以外の誤ったファイルをダウンロードしないようにご注意ください。誤ったオブジェクトファイルをダウンロードした場合は，正規のオブジェクトファイルが入った CF カードまたは USB メモリを挿入して，装置を起動してください。その後，正規のオブジェクトファイルを再度ダウンロードしてください。正規オブジェクトファイルの入手方法は，当社営業窓口へお問い合わせください。

16.1.1 ソフトウェアをCFカードよりダウンロードする

CF カードスロットに，新しいソフトウェアオブジェクトが入った CF カードを挿入して，新しいソフトウェアを装置にダウンロードします。ダウンロードしたソフトウェアは自動的に内蔵フラッシュメモリに保存されます。このとき古いバージョンのソフトウェアは別領域に待避し，新しいソフトウェアの書き込みを行います。バージョンアップ作業中は，CF カードを抜いたり，装置の電源が切断されないようにご注意ください。万が一作業中に，CF カードを抜いたり，装置の電源を切断してしまった場合は，別領域に待避してある古いバージョンのソフトウェアを再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。

```
PureFlow(A)> download cf obj nf7100.bin
Download "nf7100.bin" from Flash Memory Card (y/n)? y
Loading .....................
creating Backup from Master file.....................completed.
Done.
PureFlow(A)>
```

ダウンロードが完了しても，新しいソフトウェアはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.1.2 ソフトウェアをCFカードにアップロードする

CF カードスロットに CF カードを挿入してソフトウェアを CF カードにアップロードします。アップロードしたソフトウェアは挿入した CF カードに保存されます。

```
PureFlow(A)> upload cf obj nf7100.bin
Upload as "nf7100.bin" to Flash Memory Card (y/n)? y
Loading .....................
Done.
PureFlow(A)>
```

16.1.3 ソフトウェアをUSBメモリよりダウンロードする

USB ポートに，新しいソフトウェアオブジェクトが入った USB メモリを挿入して，新しいソフトウェアを装置にダウンロードします。ダウンロードしたソフトウェアは自動的に内蔵フラッシュメモリに保存されます。このとき古いバージョンのソフトウェアは別領域に待避し，新しいソフトウェアの書き込みを行います。バージョンアップ作業中は，USB メモリを抜いたり，装置の電源が切断されないようにご注意ください。万が一作業中に，USB メモリを抜いたり，装置の電源を切断してしまった場合は，別領域に待避してある古いバージョンのソフトウェアを再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。

```
PureFlow(A)> download usb obj nf7100.bin
Download "nf7100.bin" from USB Memory (y/n)? y
Loading .....................
creating Backup from Master file.....................completed.
Done.
PureFlow(A)>
```

ダウンロードが完了しても，新しいソフトウェアはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.1.4 ソフトウェアをUSBメモリにアップロードする

USB ポートに USB メモリを挿入してソフトウェアを USB メモリにアップロードします。アップロードしたソフトウェアは挿入した USB メモリに保存されます。

```
PureFlow(A)> upload usb obj nf7100.bin
Upload as "nf7100.bin" to USB Memory (y/n)? y
Loading .....................
Done.
PureFlow(A)>
```

16.1.5 ソフトウェアをTFTPによりダウンロードする

TFTP によりソフトウェアを装置にダウンロードします。ダウンロードしたソフトウェアは自動的に内蔵フラッシュメモリに保存されます。このとき古いバージョンのソフトウェアは別領域に待避し，新しいソフトウェアの書き込みを行います。バージョンアップ作業中は装置の電源が切断されないようにご注意ください。万が一作業中に装置の電源を切断してしまった場合は，別領域に待避してある古いバージョンのソフトウェアを再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。また，ダウンロード中に通信が切断された場合は，再度ダウンロード作業をやり直してください。

ソフトウェアを装置にダウンロードするには以下のコマンドを使用します。あらかじめ TFTP サーバと通信できるようにシステムインタフェースに正しい IP アドレスを設定してください。システムインタフェースの設定の説明は「第 7 章 システムインタフェースの設定」を参照してください。

ソフトウェアのファイルサイズが 32MByte を超えるため，RFC2349 に規定される tsize オプションに対応した TFTP サーバをお使いください。

```
PureFlow(A)> download tftp obj 192.168.100.40 nf7100.bin
Download "nf7100.bin" from 192.168.100.40 (y/n)? y
Loading ...
creating Backup from Master file.....................completed.
Done.
PureFlow(A)>
```

ダウンロードが完了しても，新しいソフトウェアはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.1.6 ソフトウェアをFTPによりダウンロードする

FTP によりソフトウェアを装置にダウンロードします。ダウンロードしたソフトウェアは自動的に内蔵フラッシュメモリに保存されます。このとき古いバージョンのソフトウェアは別領域に待避し，新しいソフトウェアの書き込みを行います。バージョンアップ作業中は装置の電源が切断されないようにご注意ください。万が一作業中に装置の電源を切断してしまった場合は，別領域に待避してある古いバージョンのソフトウェアを再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。また，ダウンロード中に通信が切断された場合は，再度ダウンロード作業をやり直してください。

ソフトウェアを装置にダウンロードするには以下のコマンドを使用します。あらかじめ FTP サーバと通信できるようにシステムインタフェースに正しい IP アドレスを設定してください。システムインタフェースの設定の説明は「第 7 章 システムインタフェースの設定」を参照してください。また，ダウンロードで使用する FTP サーバのユーザ名とパスワードを用意してください。

```
PureFlow(A)> download ftp obj 192.168.100.40 nf7100.bin
Name:ftpuser  (ユーザ名を入力)
Password:     (パスワードを入力)
Download "nf7100.bin" from 192.168.100.40 (y/n)? y
Loading ...
creating Backup from Master file.....................completed.
Done.
PureFlow(A)>
```

ダウンロードが完了しても，新しいソフトウェアはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.2 ソフトウェアアップデートパッチの適用

本装置のソフトウェアは，ソフトウェアアップデートパッチの適用により新しいソフトウェアに更新することもできます。パッチ適用は，ソフトウェアオブジェクトのダウンロードと同様の手順で行います。ただし，パッチ適用を TFTP または FTP 経由で行うことはできません。

ソフトウェアアップデートパッチの入手方法は，当社営業窓口へお問い合わせください。

16.2.1 ソフトウェアアップデートパッチをCFカードより適用する

CF カードスロットに，ソフトウェアアップデートパッチが入った CF カードを挿入して，装置内部のソフトウェアに適用します。パッチ適用作業中は，CF カードを抜いたり，装置の電源が切断されないようにご注意ください。万が一作業中に，CF カードを抜いたり，装置の電源を切断してしまった場合は，別領域に待避してある古いバージョンのソフトウェアを再ロードしますので，再度装置を起動してパッチ適用作業をやり直してください。

```
PureFlow(A)> download cf patch
Apply patch from Flash Memory Card (y/n)? y
Appling file system patch ..............  done
Appling apps patch ..............  done
Appling fcpu patch ..............  done
creating Backup from Master file.....................completed.
Done.
PureFlow(A)>
```

パッチ適用が完了しても，新しいソフトウェアはすぐに反映されません。パッチ適用が完了したあとで，装置を再起動してください。

16.2.2 ソフトウェアアップデートパッチをUSBメモリより適用する

USB ポートに，ソフトウェアアップデートパッチが入った USB メモリを挿入して，装置内部のソフトウェアに適用します。パッチ適用作業中は，USB メモリを抜いたり，装置の電源が切断されないようにご注意ください。万が一作業中に，USB メモリを抜いたり，装置の電源を切断してしまった場合は，別領域に待避してある古いバージョンのソフトウェアを再ロードしますので，再度装置を起動してパッチ適用作業をやり直してください。

```
PureFlow(A)> download usb patch
Apply patch from USB Memory (y/n)? y
Appling file system patch ..............  done
Appling apps patch ..............  done
Appling fcpu patch ..............  done
creating Backup from Master file.....................completed.
Done.
PureFlow(A)>
```

パッチ適用が完了しても，新しいソフトウェアはすぐに反映されません。パッチ適用が完了したあとで，装置を再起動してください。

16.3 コンフィギュレーションのダウンロード／アップロード

コンフィギュレーションをダウンロードするときの注意事項

弊社指定の正規コンフィギュレーションファイル以外をダウンロードすると，装置が起動しない場合があります。上記のコマンドで正規のコンフィギュレーションファイル以外の誤ったファイルをダウンロードしないようにご注意ください。誤ったコンフィギュレーションファイルをダウンロードした場合は，正規のコンフィギュレーションファイルが入った CF カードまたは USB メモリを挿入して，装置を起動してください。その後，正規のコンフィギュレーションファイルを再度ダウンロードしてください。正規コンフィギュレーションファイルの入手方法は，当社営業窓口へお問い合わせください。

16.3.1 コンフィギュレーションをCFカードよりダウンロードする

CF カードスロットに CF カードを挿入して新しいコンフィギュレーションファイルを装置にダウンロードします。ダウンロードしたコンフィギュレーションファイルは自動的に内蔵フラッシュメモリに保存されます。このとき古いコンフィギュレーションファイルは別領域に待避し，新しいコンフィギュレーションファイルの書き込みを行います。ダウンロードが完了しても，新しいコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。ダウンロード作業中は，CF カードを抜いたり，装置の電源が切断されないようにご注意ください。万が一作業中に，CF カードを抜いたり，装置の電源を切断してしまった場合は，別領域に待避してある古いコンフィギュレーションファイルを再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。

```
PureFlow(A)> download cf conf config.txt
Download ”config.txt” from Flash Memory Card (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

ダウンロードが完了しても，ダウンロードしたコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.3.2 コンフィギュレーションをCFカードにアップロードする

CF カードスロットに CF カードを挿入してコンフィギュレーションファイルを CF カードにアップロードします。アップロードしたコンフィギュレーションファイルは挿入した CF カードに保存されます。

```
PureFlow(A)> upload cf conf config.txt
Upload as ”config.txt” to Flash Memory Card (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

動作中のコンフィギュレーション情報ではなく，内部フラッシュメモリにセーブされたコンフィギュレーション情報がアップロードされます。コンフィギュレーション情報は，save config コマンドを実行したとき，内部フラッシュメモリに保存されます。

16.3.3 コンフィギュレーションをUSBメモリよりダウンロードする

USB スロットに USB メモリを挿入して新しいコンフィギュレーションファイルを装置にダウンロードします。ダウンロードしたコンフィギュレーションファイルは自動的に内蔵フラッシュメモリに保存されます。このとき古いコンフィギュレーションファイルは別領域に待避し，新しいコンフィギュレーションファイルの書き込みを行います。ダウンロードが完了しても，新しいコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。ダウンロード作業中は，USB メモリを抜いたり，装置の電源が切断されないようにご注意ください。万が一作業中に，USB メモリを抜いたり，装置の電源を切断してしまった場合は，別領域に待避してある古いコンフィギュレーションファイルを再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。

```
PureFlow(A)> download usb conf config.txt
Download "config.txt" from USB Memory (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

ダウンロードが完了しても，ダウンロードしたコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.3.4 コンフィギュレーションをUSBメモリにアップロードする

USB ポートに USB メモリを挿入してコンフィギュレーションファイルを USB メモリにアップロードします。アップロードしたコンフィギュレーションファイルは挿入した USB メモリに保存されます。

```
PureFlow(A)> upload usb conf config.txt
Upload as "config.txt" to USB Memory (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

動作中のコンフィギュレーション情報ではなく，内部フラッシュメモリにセーブされたコンフィギュレーション情報がアップロードされます。コンフィギュレーション情報は，save config コマンドを実行したとき，内部フラッシュメモリに保存されます。

16.3.5 コンフィギュレーションをTFTPによりダウンロードする

TFTP によりコンフィギュレーションファイルを装置にダウンロードします。ダウンロードしたコンフィギュレーションファイルは自動的に内蔵フラッシュメモリに保存されます。このとき古いバージョンのコンフィギュレーションファイルは別領域に待避し，新しいコンフィギュレーションファイルの書き込みを行います。ダウンロードが完了しても，新しいコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。ダウンロード作業中は装置の電源が切断されないようにご注意ください。万が一作業中に装置の電源を切断してしまった場合は，別領域に待避してある古いコンフィギュレーションファイルで再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。また，ダウンロード中に通信が切断された場合は，再度ダウンロード作業をやり直してください。

コンフィギュレーションファイルを装置にダウンロードするには以下のコマンドを使用します。あらかじめ TFTP サーバと通信できるようにシステムインタフェースに正しい IP アドレスを設定してください。システムインタフェースの設定の説明は「第 7 章　システムインタフェースの設定」を参照してください。

```
PureFlow(A)> download tftp conf 192.168.100.40 config.txt
Download "config.txt" from 192.168.100.40 (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

ダウンロードが完了しても，ダウンロードしたコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.3.6 コンフィギュレーションをTFTPによりアップロードする

TFTP によりコンフィギュレーションファイルを TFTP サーバにアップロードします。アップロードしたコンフィギュレーションファイルは TFTP サーバに保存されます。

```
PureFlow(A)> upload tftp conf 192.168.100.40 config.txt
Upload as "config.txt" to 192.168.100.40 (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

動作中のコンフィギュレーション情報ではなく，内部フラッシュメモリにセーブされたコンフィギュレーション情報がアップロードされます。

16.3.7 コンフィギュレーションをFTPによりダウンロードする

FTP によりコンフィギュレーションファイルを装置にダウンロードします。ダウンロードしたコンフィギュレーションファイルは自動的に内蔵フラッシュメモリに保存されます。このとき古いバージョンのコンフィギュレーションファイルは別領域に待避し，新しいコンフィギュレーションファイルの書き込みを行います。ダウンロードが完了しても，新しいコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。ダウンロード作業中は装置の電源が切断されないようにご注意ください。万が一作業中に装置の電源を切断してしまった場合は，別領域に待避してある古いコンフィギュレーションファイルで再ロードしますので，再度装置を起動してダウンロード作業をやり直してください。また，ダウンロード中に通信が切断された場合は，再度ダウンロード作業をやり直してください。

コンフィギュレーションファイルを装置にダウンロードするには以下のコマンドを使用します。あらかじめ FTP サーバと通信できるようにシステムインタフェースに正しい IP アドレスを設定してください。システムインタフェースの設定の説明は「第 7 章 システムインタフェースの設定」を参照してください。また，ダウンロードで使用する FTP サーバのユーザ名とパスワードを用意してください。

```
PureFlow(A)> download ftp conf 192.168.100.40 config.txt
Name:ftpuser  (ユーザ名を入力)
Password:     (パスワードを入力)
Download "config.txt" from 192.168.100.40 (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

ダウンロードが完了しても，ダウンロードしたコンフィギュレーションはすぐに反映されません。ダウンロードが完了したあとで，装置を再起動してください。

16.3.8 コンフィギュレーションをFTPによりアップロードする

FTP によりコンフィギュレーションファイルを FTP サーバにアップロードします。アップロードしたコンフィギュレーションファイルは FTP サーバに保存されます。

```
PureFlow(A)> upload ftp conf 192.168.100.40 config.txt
Name:ftpuser  (ユーザ名を入力)
Password:     (パスワードを入力)
Upload as "config.txt" to 192.168.100.40 (y/n)? y
Loading ...
Done.
PureFlow(A)>
```

動作中のコンフィギュレーション情報ではなく，内部フラッシュメモリにセーブされたコンフィギュレーション情報がアップロードされます。

16.4 ソフトウェアを再起動する

ダウンロードが完了したあとは新しいソフトウェアで再起動させます。

(1) 装置を再起動する

装置の再起動方法です。電源を再投入するか以下のコマンドを使用してください。

```
PureFlow(A)> reboot system
Rebooting the system, ok(y/n)? y
```

(2) 起動ファイルを確認する

シリアルコンソールのボーレートを 9600bps に設定している場合，装置の起動時に起動ファイル種別および CRC チェックの結果が表示されます。

```
reading :Object
checkCRC:OK
```

ダウンロード中の電源断等でダウンロードが異常終了すると，Master ファイルが CRC エラーとなり，Backup ファイルで起動します。Backup ファイルでの起動後に再度ダウンロードしてください。

```
reading :Object
checkCRC:NG
reading :Backup
checkCRC:OK
```

起動ファイル種別は以下の通りです。

表示	説明	優先度
/dev/usb1	USB メモリ上のファイル	高 ↑ ↓ 低
/dev/externalcf1	CF カード上のファイル	
Object	Master ファイル	
Backup	Backup ファイル	

(3) 再起動の完了確認

再起動は Telnet/SSH の接続がいったん切断されます。装置起動後，再度 Telnet/SSH によりログインし直してください。

（空白ページ）

第17章　ドメインフィルタ機能

ここでは，ドメインフィルタ機能について説明します。

17.1 概要

ドメインフィルタ機能は，帯域制御のパケット分類識別子としてドメイン名を使用することができる機能です。本機能を使用することで，制御対象通信識別を「example.com」のようなドメイン名で指定することができます。その結果，IPアドレスが動的に変わるクラウドサービスのような通信でもIPアドレスを意識することなく帯域制御が可能となります。

ドメインフィルタ機能は，装置を通過するDNSレスポンスパケットにより，指定したドメイン名からIPアドレスを自動検索して学習します。そのIPアドレス情報をフィルタに設定することで対象通信の帯域制御が可能となります。IPアドレスの自動検索および学習はDNSレスポンスパケットが発生するごとに行うため，対象となるサイトのIPアドレスが変更されても追従します。また一度学習したIPアドレスは一定時間保持します。

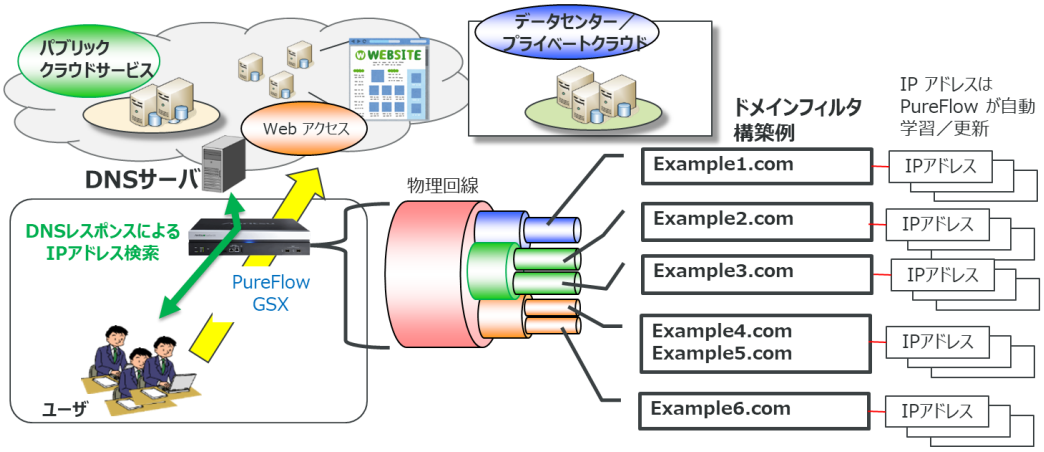


図 17.1-1 ドメインフィルタ機能概要

ドメインフィルタ機能を使用する場合は，「NF7101-L006A ドメインフィルタ機能ライセンス」が必要です。

17.2 ドメインフィルタ機能仕様について

本章ではドメインフィルタ機能に関する仕様を説明します。

（1）ドメイン名

以下に，設定可能なドメイン名の仕様を示します。

表 17.2-1 ドメイン名仕様

項目	仕様
ドメイン名の長さ	253 文字以下
ラベルの長さ	63 文字以下
使用可能文字	1234567890 abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ -.* 「*」は、ワイルドカード設定時に使用 日本語ドメインは対応しません。
大文字小文字の区別	大文字と小文字の混在設定は可能ですが区別しません。 （例） 「example.com」と「EXAMPLE.com」は同じ文字列として扱います。
文字列一括設定	後方一致のみ，「*」によるワイルドカード設定が可能です。 （例） 「*.com」「*.example.com」「*example1.example2.com」
その他	トップドメインは設定必須です。 「.」「*」「*.」は設定できません。

(2)　設定可能なドメイン名の数

ドメインフィルタ機能は「ルールリスト」により設定します。まず，制御対象ドメイン名群を一括管理するためのルールリストグループを作成します。その中にエントリとして制御対象ドメイン名を設定します。そのため，各項目の設定数はルールリストの仕様に準じていますが，ドメイン名で検索，学習したIPアドレス(以下，エントリ IP アドレス)はグループあたり 512 件，全体で 64,000 件保持することが可能です。

表 17.2-2 ドメイン名／エントリーIP アドレス設定数

項目	仕様
最大グループ数	1024 件(全種別(ipv4, ipv6, l4port, domain)合計)
最大エントリ数	512 件／グループ 装置全体 64,000 件(全種別(ipv4, ipv6, l4port, domain)合計)
最大エントリ IP アドレス数	512 件／グループ 装置全体 64,000 件 学習した IP アドレスの保持時間は学習時の TTL 値＋86400(1 日)秒です。

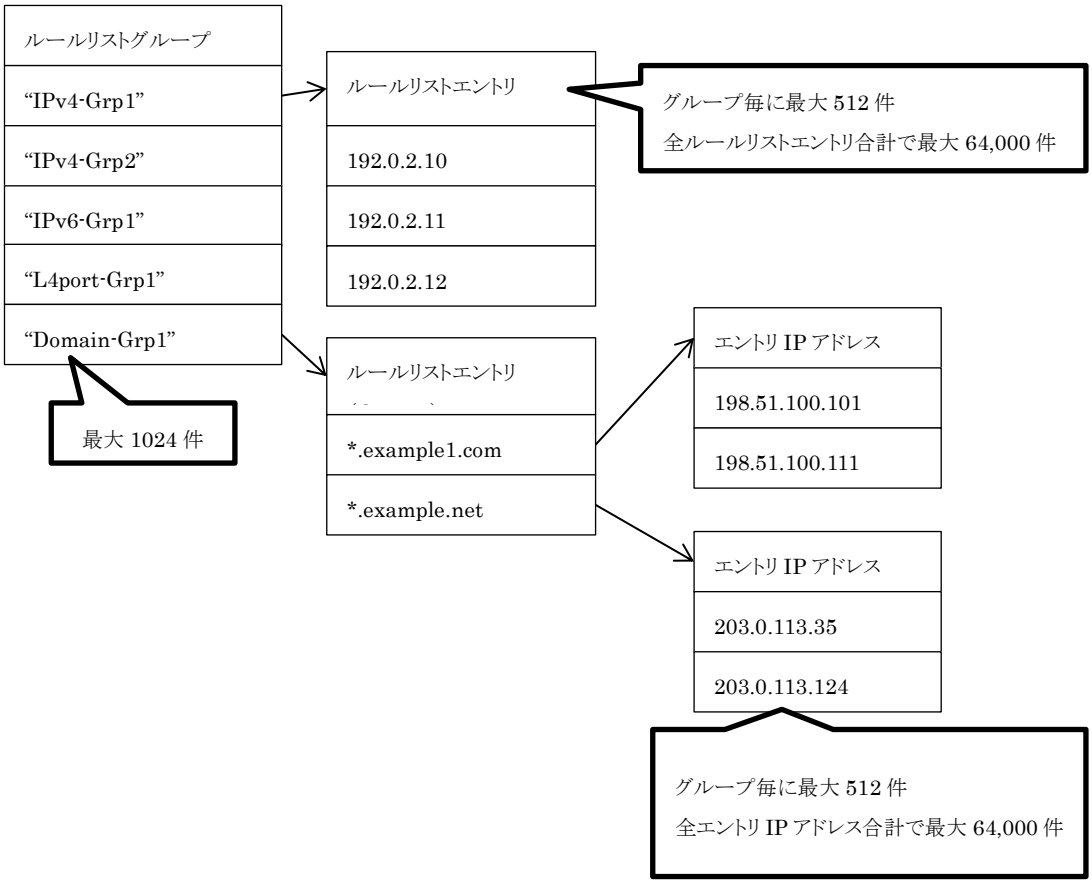


図 17.2-1 ドメインフィルタ機能対応ルールリスト体系

なお，ルールリストについては「8.5 ルールリストの設定方法」を参照してください

17.3 設定手順

ドメインフィルタ機能を使用する際の設定はルールリストで行います。その手順を以下に記します。

（例 1）ドメイン名「example1.com」と「example2.com」向けの通信を一括して制御するため，ドメインフィルタ機能を使用する。

手順 1） ルールリストにて，対象ドメイン名を制御するためのルールリストグループ「1-site-EX」を作成します。

```
PureFlow (A)> add rulelist group 1-site-EX domain
```

手順 2） 手順 1 で作成したグループ「1-site-EX」にエントリとして対象ドメイン名を登録します。

```
PureFlow(A)> add rulelist entry 1-site-EX domain example1.com
PureFlow(A)> add rulelist entry 1-site-EX domain example2.com
```

これで機能が有効となり，DNS レスポンスパケットにより対象ドメイン名から IP アドレスを自動的に検索して学習します。
帯域制御シナリオ設定後，ここで作成したルールリストグループをフィルタ条件として設定することで帯域制御が行われます。
以下は，「1-site-EX」で学習した IP アドレス向けに上限 50Mbps で制御するための設定を示します。

```
PureFlow(A)> add scenario /port1/1-site-ex action aggregate peak_bw 50M
PureFlow(A)> add filter scenario /port1/1-site-ex filter site ipv4 dip list 1-site-EX
```

（例 2） トップレベルドメインとセカンドレベルドメインのラベルが「example3.com」である場合の通信を制御するため，ドメインフィルタ機能を使用する。

手順 1） ルールリストにて，対象ドメイン名を制御するためのルールリストグループを作成します。

```
PureFlow (A)> add rulelist group 2-site-EX3 domain
```

手順 2） 手順 1 で作成したグループにエントリとして対象ドメイン名をワイルドカード「*」付きで登録します。

```
PureFlow(A)> add rulelist entry 2-site-EX3 domain *.example3.com
```

ここで DNS レスポンスパケットから「example3.com」が存在するすべての IP アドレスを自動的に検索して学習します。
（例 1）と同様に，帯域制御シナリオ設定後，ここで作成したルールリストグループをフィルタ条件として設定することで帯域制御が行われます。
以下は，「2-site-EX3」で学習した IP アドレス向けに上限 70Mbps で制御するための設定を示します。

```
PureFlow(A)> add scenario /port1/2-site-ex3 action aggregate peak_bw 70M
PureFlow(A)> add filter scenario /port1/2-site-ex3 filter site ipv4 dip list 2-site-EX3
```

17.4 確認手順

ドメインフィルタ機能で学習したドメイン名とIPアドレス情報は，ルールリストの”show rulelist”コマンドで確認することが可能です。

（例 1） 設定したルールリストすべての情報を確認する。

```
PureFlow(A)> show rulelist all
Total rulelist groups: 2

ListName: 1-site-EX
        Type                    : domain
        Rulelist Index          : 1
Number of Rules  :
        Total                   : 512
        Used                    : 1
        Available    : 511
Number of IP Address Learning:
        Total                   : 512
        Used                    : 1
        Available    : 511
Rules:
        [   1]          : example1.com
        <Domain IP>
        NAME        : example1.com          ←自動登録したドメイン名を表示します。
        CNAME       : abc.example1.com      ←自動登録した CNAME を表示します。
        Address     : 192.0.2.10            ←自動登録した IP アドレスを表示します。
        TTL         : 87000[s]              ←保存時間を表示します。
ListName: 1-site-EX3
        Type                    : domain
        Rulelist Index      : 2
Number of Rules:
        Total                   : 512
        Used                    : 1
        Available    : 511
Number of IP Address Learning:
        Total                   : 512
        Used                    : 1
        Available    : 511
Rules:
        [   1]          : *.example3.com
        <Domain IP>
        NAME            : zzz.example3.com
        CNAME       : yy.zzz. example.com
        Address     : 198.51.100.10
        TTL         : 86400[s]
```

（例 2） 任意のルールリスト情報を確認する。

```
PureFlow(A)> show rulelist name 1-site-EX
Total rulelist groups: 2

ListName: 1-site-EX
        Type                : domain
        Rulelist Index      : 1
        Number of Rules     :
          Total             : 512
          Used              : 1
          Available         : 511
        Number of IP Address Learning:
          Total             : 512
          Used              : 1
          Available         : 511
        Rules:
              [   1]        : example1.com
            <Domain IP>
            NAME    : example1.com          ←自動登録したドメイン名を表示します。
            CNAME   : abc.example1.com      ←自動登録した CNAME を表示します。
            Address : 192.0.2.10            ←自動登録した IP アドレスを表示します。
            TTL     : 87000[s]              ←保存時間を表示します。

            NAME    : example1.com          ←自動登録したドメイン名を表示します。
            CNAME   : def.example1.com      ←自動登録した CNAME を表示します。
            Address : 192.0.2.20            ←自動登録した IP アドレスを表示します。
            TTL     : 88000[s]              ←保存時間を表示します。
```

また，エントリ IP アドレスのリソース状況については”show resource”コマンドで確認することができます。

```
PureFlow(A)> show resource
Resource information
                                        Total       Used      Available
-------------------------------------------------------------------------
Scenario                        :        4100          4     4096 [entry]
  Individual Que                :        4096          0     4096 [entry]
  Second Peer                   :         100          0      100 [entry]
  Keep Alive                    :         100          0      100 [entry]
Filter                          :       10000          0    10000 [entry]
Rulelist                        :        1024          2     1022 [group]
Total Rulelist Entry            :       10000          2     9998 [entry]
Total Domain IP Entry           :       10000        356     9644 [entry]
                                                ↑エントリ IP アドレスのリソース状況が確認できます。
        :                               :           :           :
        :                               :           :           :
```

エントリ IP アドレスが装置上限に達した場合は syslog に記録します。syslog に関しては「付録B SYSLOG 一覧」を参照してください。

17.5 注意事項

(1) クラウドサービスやアップデートサービスなどでお使いになる場合は，対象サービスのドメイン名がプロバイダーやWebサイト等で公開されている必要があります。本装置は，あらかじめ制御対象のドメイン名を登録しておくことで，名前解決発生時にIPアドレスを自動学習し，そのIPアドレスをフィルタ条件として通信を制御します。

(2) 本機能で動作するDNSレスポンスパケットはUDPでかつデータサイズ（UDPヘッダ＋データ）が512バイト以下（RFC1035準拠）のものです。DNSSECおよび，ゾーン転送などで用いられるTCPパケットでは動作しません。また，DNSの拡張方式であるEDNS0（RFC6891）にも対応しておりません。

(3) 本機能の設定および確認はCLIのみで，WebAPIには対応していません。
ただし，設定したグループおよびエントリは削除できます。

(4) 本機能は，IPv6パケットには対応していません。

(5) エントリにワイルドカードを使用した場合，本機能において発見したドメイン名が設定した複数のドメイン名に一致する場合があります。その場合，一致する文字数の多いドメイン名を設定したルールリストグループに登録します。

（例）

・エントリとして設定したドメイン名

設定ドメイン名①　*.example.com　——ルールリストグループAに設定

設定ドメイン名②　aaa.example.com　——ルールリストグループBに設定

・ドメインフィルタ機能により発見したドメイン名およびIPアドレス

発見ドメイン名／IPアドレス　aaa.example.com／203.0.113.10

上記の場合，発見ドメイン名／IPアドレス「aaa.example.com／203.0.113.10」はラベル数が多いほうのルールリストグループBに登録されます。

(6) インターネットアクセスにプロキシサーバを適用した構成で，装置をプロキシサーバよりユーザ側に設置した場合，プロキシサーバ宛ての通信はドメインフィルタ機能で分類することができません。

(7) 異なるドメイン名を設定して検索したIPアドレスが両方とも同じであった場合，それぞれのドメイン名に対して同じIPアドレスを学習します。この場合，ドメイン名ごとに帯域制御は行えません。

（空白ページ）

第18章　トラフィック分析機能

ここでは，トラフィック分析機能について説明します。

18.1 概要

トラフィック分析機能は，ネットワークの転送性能，および，アプリケーションの転送性能を測定するための機能です。本装置を経由するアプリケーショントラフィックのパケットヘッダを参照し，パケット損失や転送遅延を測定します。これらを継続して測定することで,ネットワークの状態変化を監視するための指標として使用します。

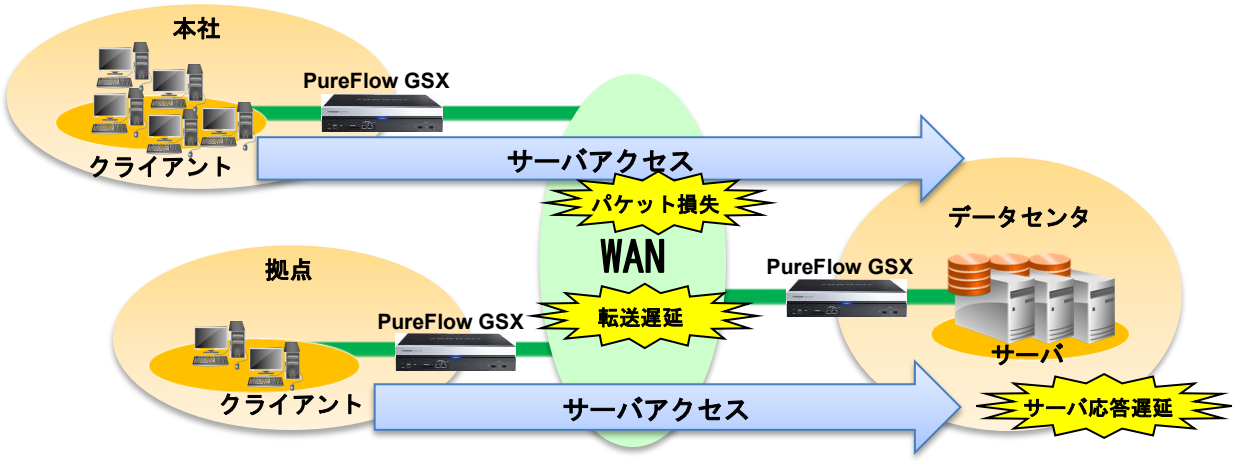


図 18.1-1　トラフィック分析の概要

また，PureFlow Profiler を使用することにより，トラフィック分析結果をグラフ表示し，過去のデータを含めたレポートを作成することができます。詳細は，「PureFlow Profiler モニタリングマネージャ3 NF7202A 取扱説明書」を参照してください。

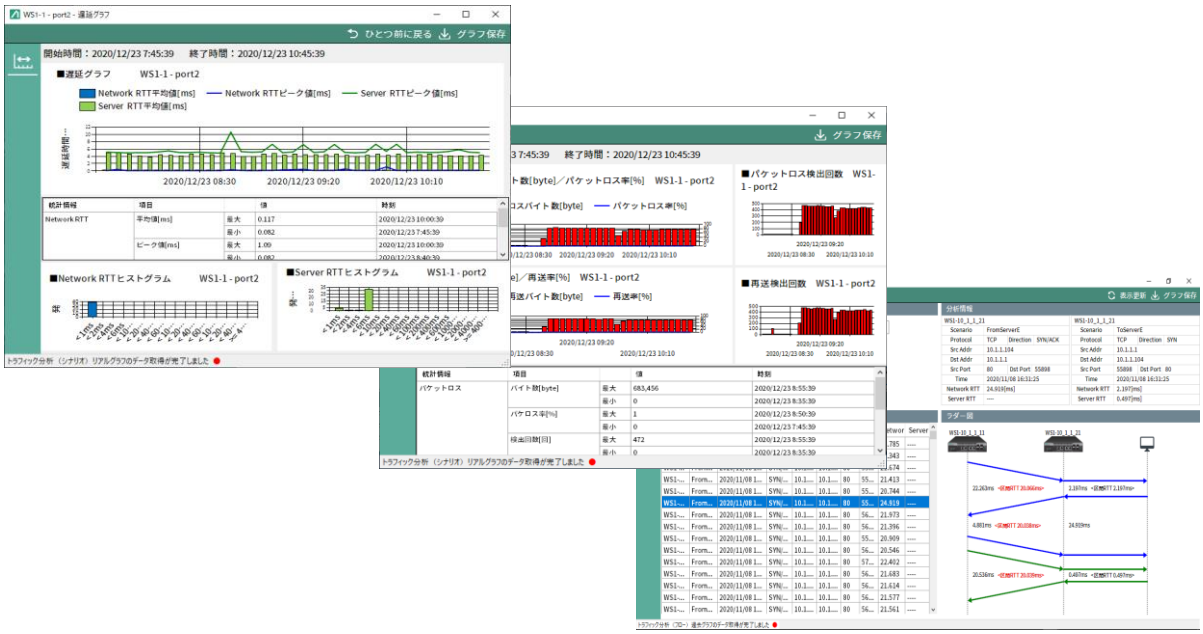


図 18.1-2　PureFlow Profiler の表示例

18.2 トラフィック分析の測定項目

トラフィック分析機能において,アプリケーションの種別ごとに測定項目が異なります。ここでは,アプリケーション種別ごとの測定項目を説明します。

18.2.1 アプリケーション種別

トラフィック分析機能は，いくつかのアプリケーションを測定対象とします。測定対象の一覧を以下に記載します。

表 18.2.1-1　測定対象のアプリケーション種別

アプリケーション種別	概要
TCP	IPv4 および IPv6 の TCP 通信を測定します。
ICMP	IPv4 および IPv6 の ICMP 通信を測定します。測定対象は ICMP echo 要求と応答のみです。

各アプリケーションに合わせ，様々な項目を測定します。測定項目は以下の説明を参照してください。

18.2.2 TCP の測定項目

TCP 通信を検出したとき，トラフィック分析機能は，以下の項目を測定します。

表 18.2.2-1　TCP の測定項目

測定項目	測定方法	用途
ネットワーク遅延 Network RTT	端末が送信した SYN パケットを転送してから，対向側の端末が送信した応答パケット（ACK パケットまたは RST パケット）を転送するまでの時間を測定する。	ネットワークの往復遅延を予測する指標として使用する。
サーバ遅延 ServerRTT	端末が送信した最初のデータパケットを転送してから，対向側の端末が送信したデータパケットを転送するまでの時間を測定する。	アプリケーションの起動時間を予測する指標として使用する。
データ往復遅延 Data RTT	端末がデータを送信してから，対向側の端末がデータを送信し，本装置に到達するまでの時間を測定する。	アプリケーションの応答時間の変化を予測する指標しとして使用する。
データ ACK 遅延 Data ACK RTT	端末がデータを送信してから，対向側の端末がデータ ACKを送信し，本装置に到達するまでの時間を測定する。パケット再送があった場合は，再送時間も含めた時間を測定する。	データ転送に要した時間を予測する指標として使用する。
データ損失 Segments lost	TCP プロトコルのシーケンス番号をパケットごとに参照し，データ損失を測定する。TCP ヘッダのシーケンス番号が，次に来るべきシーケンス番号よりも大きいとき，データ損失と判定する。	ネットワークのデータ損失を予測する指標として使用する。
データ再送 Segments Retransmitted	TCP プロトコルのシーケンス番号をパケットごとに参照し，データ再送を測定する。TCP ヘッダのシーケンス番号が，次に来るべきシーケンス番号よりも小さいとき，データ再送と判定する。	TCP によるデータ再送を予測する指標として使用する。
SYN 受信回数 SYN received	TCP ヘッダの SYN フラグがセットされたパケットを計上する。	通信の有無を確認するために使用する。
ACK 受信回数 ACK received	TCP ヘッダの ACK フラグがセットされたパケットを計上する。	通信の有無を確認するために使用する。

DATA 受信回数 DATA received	TCP パケットにおいてデータ長が1以上のパケットを計上する。再送パケットも含む。	通信の有無を確認するために使用する。
FIN 受信回数 FIN received	TCP ヘッダの FIN フラグがセットされたパケットを計上する。	通信の有無を確認するために使用する。
RST 受信回数 RST received	TCP ヘッダの Reset フラグがセットされたパケットを計上する。	通信の有無を確認するために使用する。

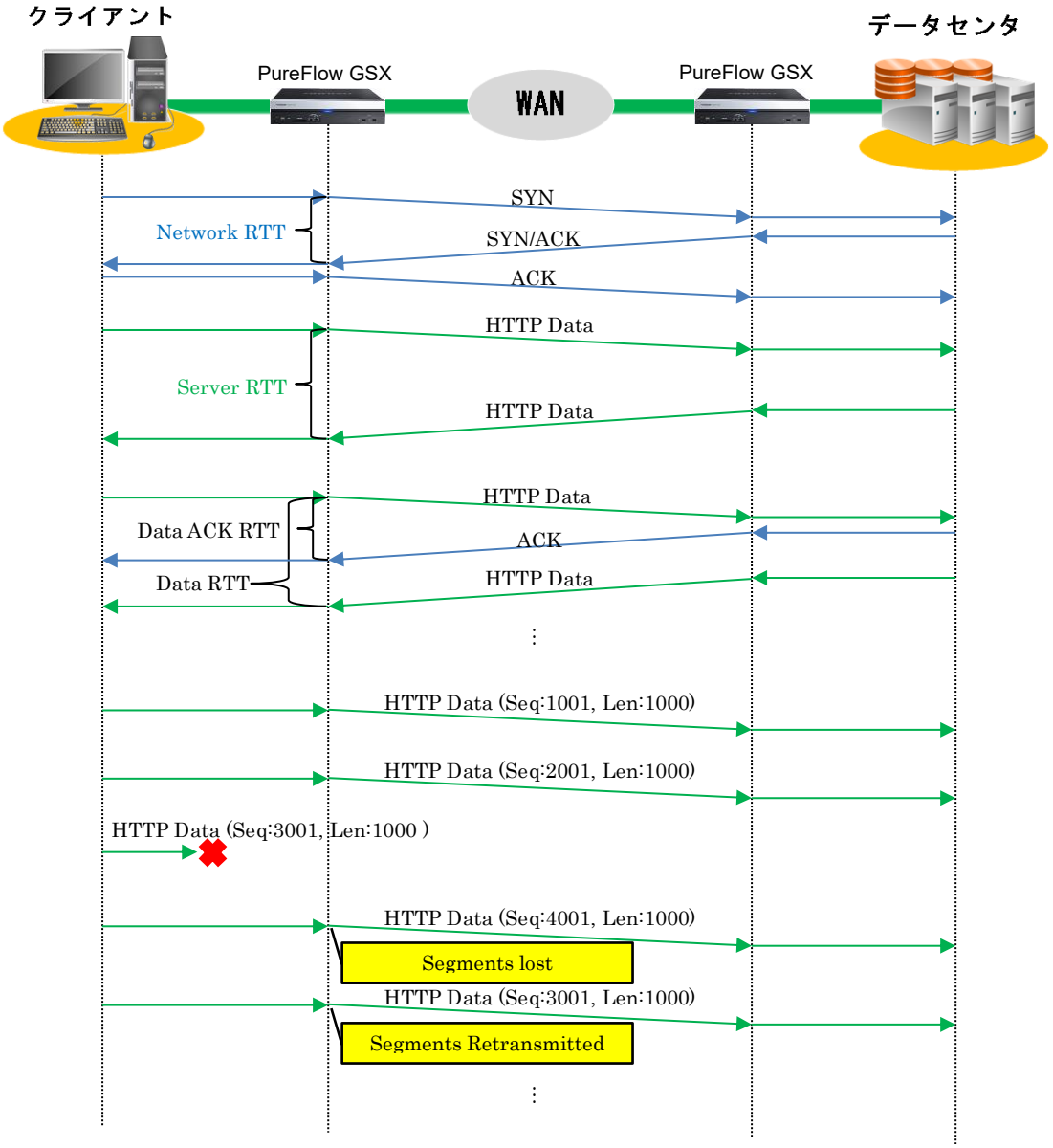


図 18.2.2-1　TCP の測定項目

18.2.3 ICMP の測定項目

ICMP echo 要求を検出したとき，トラフィック分析機能は，以下の項目を測定します。

表 18.2.3-1　ICMP の測定項目

測定項目	測定方法	用途
ネットワーク遅延 Network RTT	端末が送信した ICMP echo 要求を転送してから，対向側の端末が送信した ICMP echo 応答を転送するまでの時間を計測します。	ネットワークの往復遅延を測定する指標として使用する。

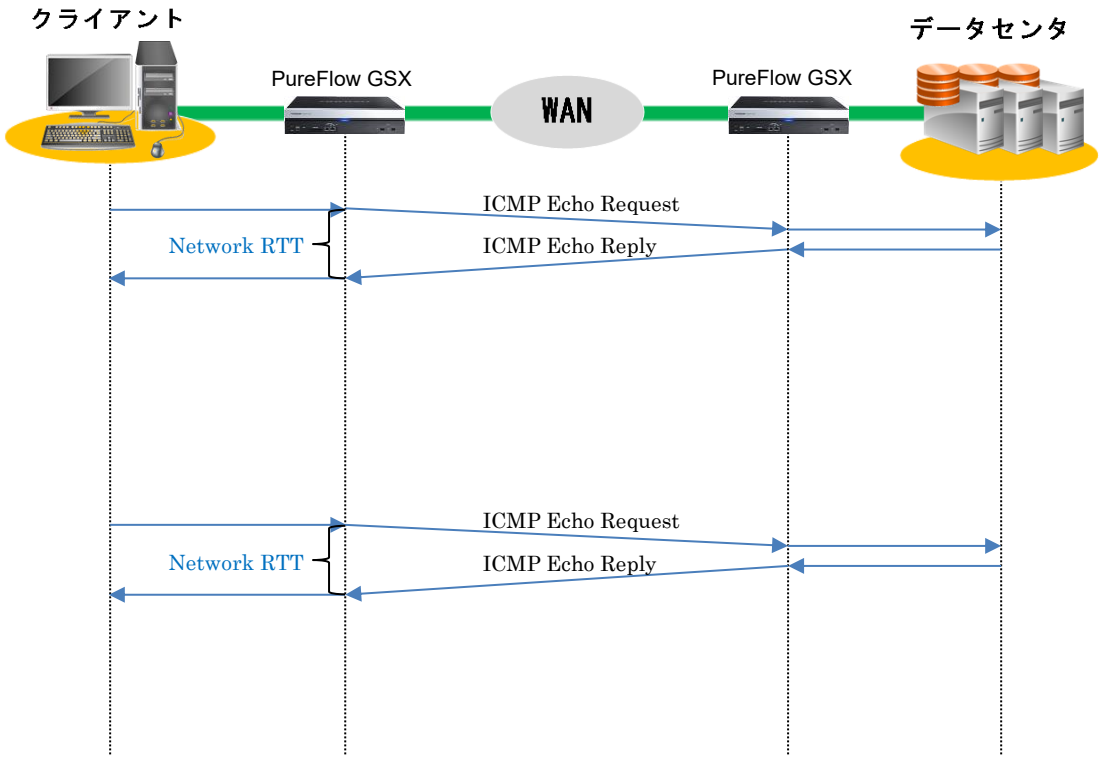


図 18.2.3-1　ICMP の測定項目

18.3 トラフィック分析の集計方法

トラフィック分析の集計方法として，2種類あります。ここでは，トラフィック分析のシナリオ集計とトップ集計について説明します。

18.3.1 シナリオ集計

シナリオ集計は，シナリオを通過するトラフィックを対象に，トラフィック分析で得られた NetworkRTT などの測定値を集計し，最大値，平均値，最小値，ヒストグラムなどの統計情報を生成します。シナリオを階層的に指定することにより，統計情報も階層的に集計することができます。

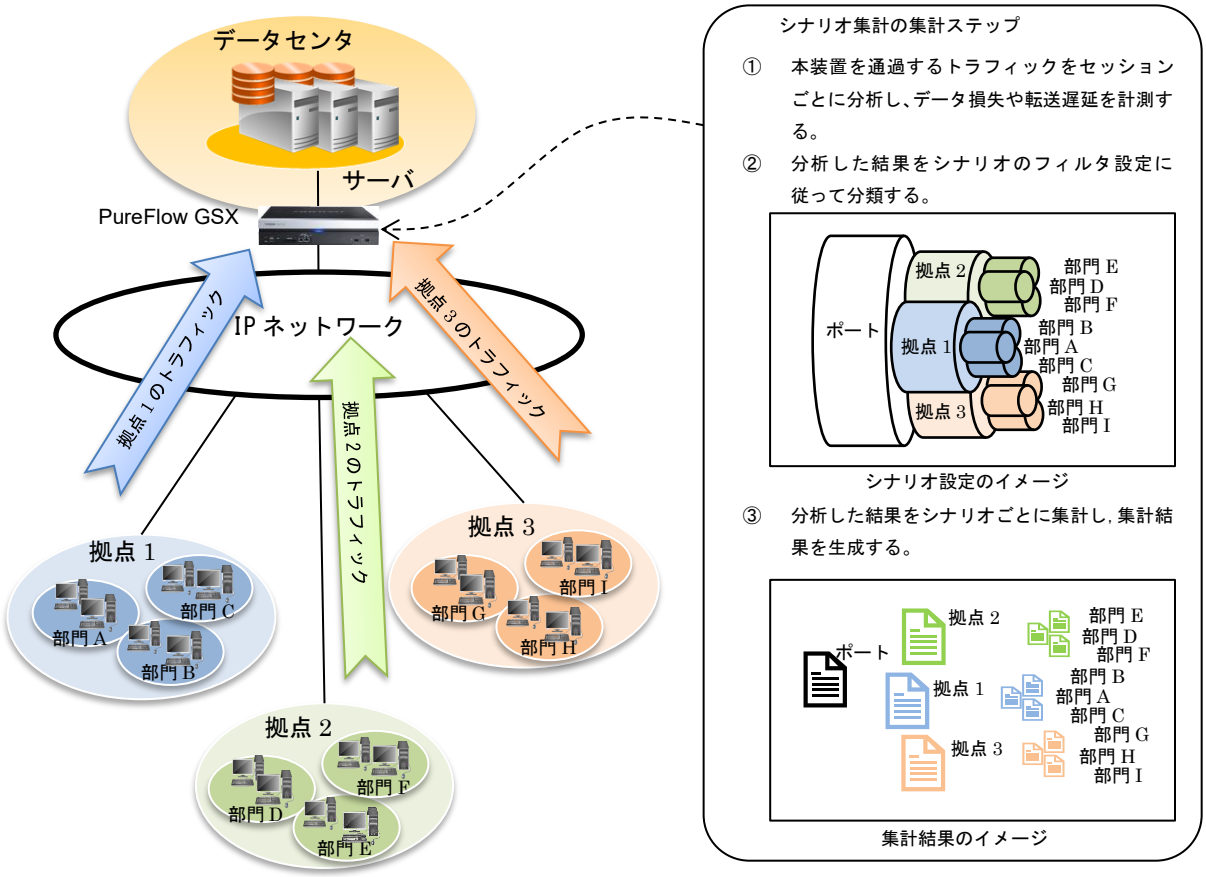


図 18.3.1-1　シナリオ集計

シナリオ集計は，トラフィック種別ごとに様々な統計情報を生成します。シナリオ集計で生成する統計情報を以下の表に記します。

表 18.3.1-1　シナリオ集計の統計情報

トラフィック種別	測定項目	統計情報					
		最大値	最小値	平均値	合計値	比率	ヒストグラム
TCP	Network RTT ネットワーク遅延	○	○	○	－	－	○
	Server RTT サーバ遅延	○	○	○	－	－	○
	Data RTT データ RTT	○	○	○	－	－	○
	Data ACK RTT データ ACK RTT	○	○	○	－	－	○
	Segments sent out データ送信数	－	－	－	○	－	－
	Segments lost データ損失数	－	－	－	○	○※1	－
	Segments retransmitted データ再送数	－	－	－	○	○※2	－
	Number of Flow フロー数	－	－	－	○	－	－
	Number of Flow with loss データ損失が発生したフロー数	－	－	－	○	－	－
	Number of Flow with retransmit データ再送が発生したフロー数	－	－	－	○	－	－
	SYN received TCP SYN フラグパケットの受信数	－	－	－	○	－	－
	ACK received TCP ACK フラグパケットの受信数	－	－	－	○	－	－
	DATA received TCP データパケットの受信数	－	－	－	○	－	－
	FIN received TCP FIN フラグパケットの受信数	－	－	－	○	－	－
	RST received TCP RST フラグパケットの受信数	－	－	－	○	－	－

※1）データ損失率を表示します。「Segments lost÷Segments send out」で算出した結果です。

※2）データ再送率を表示します。「Segments retransmitted÷Segments send out」で算出した結果です。

18.3.2 トップ集計

トップ集計は，トラフィック分析で得られた測定値を送受信 IP アドレスなどのセッション単位で細分化して集計します。トップ集計は，シナリオごとに最大で 100 セッションまでを集計し，Network RTT が大きい順に各セッションをソートして表示します。

トップ集計が生成する統計情報を以下に記します。

表 18.3.2-1　トップ集計の集計項目

トラフィック種別	集計項目	説明
TCP	Time セッション生成時刻	TCP セッションが生成された時刻を記録します。
	Direction フロー方向	TCP セッションの方向を記録します。TCP クライアントのフローを TCP SYN，TCP サーバのフローを TCP SYNACK と表示します。
	Type IP プロトコルのバージョン	IPv4，IPv6 などの IP バージョンを記録します。
	Src Addr Source IP アドレス	IP ヘッダの Source IP アドレスを記録します。
	Dst Addr Destination IP アドレス	IP ヘッダの Destination IP アドレスを記録します。
	Protocol プロトコル番号	IP ヘッダのプロトコル名（TCP）を記録します。
	Src Port Source Port 番号	TCP ヘッダの Source Port 番号を記録します。
	Dst Port Destination Port 番号	TCP ヘッダの Destination Port 番号を記録します。
	Network RTT ネットワーク遅延	Network RTT の測定値を記録します。
	Server RTT サーバ遅延	Server RTT の測定値を記録します。
ICMP	Time セッション生成時刻	ICMP echo 要求を転送した時刻を記録します。複数の ICMP echo 要求があった場合，集計周期内の最後の ICMP echo 要求を転送した時刻を記録します。
	Direction フロー方向	ICMP Request，および，ICMP Reply など，フローの方向を記録します。
	Type IP プロトコルのバージョン	IPv4，IPv6 などの IP バージョンを記録します。
	Src Addr Source IP アドレス	IP ヘッダの Source IP アドレスを記録します。
	Dst Addr Destination IP アドレス	IP ヘッダの Destination IP アドレスを記録します。
	Protocol プロトコル番号	IP ヘッダのプロトコル名（ICMP）を記録します。
	Network RTT ネットワーク遅延	測定した Network RTT を記録します。

18.4 トラフィック分析の設定方法

トラフィック分析の設定方法と表示方法について説明します。

18.4.1 シナリオ集計の設定方法

シナリオ集計を使用するには，あらかじめ，トラフィック分析有効設定（“set analysis”コマンド）と測定対象シナリオの追加（“add analysis target”コマンド）を実行してください。

まず，トラフィック分析を有効に設定します。

```
PureFlow(A)＞ set analysis enable
```

次に，測定対象シナリオを追加します。測定対象シナリオを追加するとき，シナリオ集計を行うトラフィック種別を指定します。例えば，シナリオ“/port1”を通過する TCP トラフィックの統計情報を集計するとき．測定対象シナリオを以下のように追加します。

```
PureFlow(A)＞ add analysis target scenario /port1 tcp
```

反対方向のトラフィックについても統計情報を集計する場合は，測定対象シナリオとして反対方向のシナリオも追加します。例えば，TCP トラフィックが“/port1”と “/port2”を通過する場合は，測定対象シナリオとして“/port2”も追加します。シナリオ集計において，測定対象シナリオは最大 200 個まで追加できます。

```
PureFlow(A)＞ add analysis target scenario /port2 tcp
```

集計周期（デフォルト5分）が経過するまで待ち，結果を表示します。例えば，シナリオ“/port1”を通過するトラフィックの統計情報を表示した場合，トラフィック分析表示(“show analysis target”コマンド)で，以下のように表示します。

```
PureFlow(A)> show analysis target scenario /port1 histogram
From       : 2020 Oct 27 11:15:24  To          : 2020 Oct 27 11:20:24

TCP
  Network RTT   (Min/Avg/Max)      :       0.144/    0.290/    0.485[msec]      6[times]
  Server RTT    (Min/Avg/Max)      :       0.912/    7.191/   19.802[msec]      6[times]
  Data RTT      (Min/Avg/Max)      :       0.561/   57.653/  446.927[msec]     15[times]
  Data Ack RTT  (Min/Avg/Max)      :       0.033/    2.836/   33.982[msec]     15[times]
  Segments sent out                :          6936[bytes]
  Segments lost                    :             0[bytes]          0[%]         0[times]
  Segments retransmitted           :             0[bytes]          0[%]         0[times]
  Number of Flow                   :             9[flows]
  Number of Flow with loss         :             0[flows]
```

```
  Number of Flow with retransmit  :             0[flows]
  SYN  received                   :             6[times]
  ACK  received                   :           714[times]
  DATA received                   :            15[times]
  FIN  received                   :             9[times]
  RST  received                   :             0[times]

Histogram (Network RTT)                 Histogram (Server RTT)
  Time Interval  Count                    Time Interval  Count
  -------------  --------------------     -------------  --------------------
           1ms                       6            1ms                       1
           2ms                       0            2ms                       1
           4ms                       0            4ms                       0
           6ms                       0            6ms                       1
          10ms                       0           10ms                       2
          20ms                       0           20ms                       1
          40ms                       0           40ms                       0
          60ms                       0           60ms                       0
         100ms                       0          100ms                       0
         200ms                       0          200ms                       0
         400ms                       0          400ms                       0
         600ms                       0          600ms                       0
        1000ms                       0         1000ms                       0
        2000ms                       0         2000ms                       0
        4000ms                       0         4000ms                       0
  above 4000ms                       0   above 4000ms                       0
  Histogram (Data RTT)                   Histogram (Data Ack RTT)
  Time Interval  Count                    Time Interval  Count
  -------------  --------------------     -------------  --------------------
           1ms                       2            1ms                      10
           2ms                       4            2ms                       3
           4ms                       1            4ms                       1
           6ms                       1            6ms                       0
          10ms                       2           10ms                       0
          20ms                       1           20ms                       0
          40ms                       0           40ms                       1
          60ms                       1           60ms                       0
         100ms                       1          100ms                       0
         200ms                       0          200ms                       0
         400ms                       1          400ms                       0
         600ms                       1          600ms                       0
        1000ms                       0         1000ms                       0
        2000ms                       0         2000ms                       0
        4000ms                       0         4000ms                       0
  above 4000ms                       0   above 4000ms                       0
```

統計情報の表示フォーマットについては，「PureFlow GSX トラフィックシェーパーNF7101C コマンドリファレンス」の”show analysis target”コマンドの表示を参照してください。

なお，シナリオ集計は測定対象のトラフィックが流れてから，次の集計周期が経過した時に測定結果を集計します。トラフィック分析表示(“show analysis target”コマンド)を連続で実行した場合，次の集計周期が経過するまで，同じ集計結果を表示します。

18.4.2 トップ集計の設定方法

トップ集計を使用するには，あらかじめ，トラフィック分析有効設定(“set analysis”コマンド)と測定対象シナリオの追加(“add topanalysis target”コマンド)を実行してください。

まず，トラフィック分析を有効に設定します。

```
PureFlow(A)＞ set analysis enable
```

次に,測定対象シナリオを追加します。測定対象シナリオを追加するとき，トップ集計で集計する集計単位を指定します。例えば，シナリオ“/port1”を通過するトラフィックについて，フロー単位で細分化して表示する場合は，以下のように設定します。

```
PureFlow(A)＞ add topanalysis target scenario /port1 flow
```

反対方向のトラフィックについても統計情報を集計する場合は，測定対象シナリオとして反対方向のシナリオも追加します。例えば，TCP トラフィックが“/port1”と “/port2”を通過する場合は，測定対象シナリオとして“/port2”も追加します。トップ集計において，測定対象シナリオを最大 25 個まで設定できます。

```
PureFlow(A)＞ add topanalysis target scenario /port2 flow
```

集計周期(デフォルト5分)が経過するまで待ち，結果を表示します。例えば，シナリオ“/port1”を通過するトラフィックの統計情報を表示した場合，トラフィック分析表示(“show topanalysis target”コマンド)は，以下のように表示します。測定結果をフロー単位で集計し，NetworkRTT が大きい順にソートして表示します。

```
PureFlow(A)> show topanalysis target scenario /port1
From       : 2020 Oct 27 11:15:00  To          : 2020 Oct 27 11:20:00

Sort Type    : Network RTT
Flow
    Flow 1:
      Time                      : 2020 Oct 27 11:15:38
      Direction                 : TCP SYN
      Type                      : IPv4
      Src Addr                  : 192.168.37.15
      Dst Addr                  : 192.168.37.1
      Protocol                  : TCP
      Src Port                  : 56647
      Dst Port                  : 80
      Network RTT               : 5.485     [msec]
      Server RTT                : 19.802    [msec]
    Flow 2:
      Time                      : 2020 Oct 27 11:16:42
```

```
      Direction                 : TCP SYN
      Type                      : IPv4
      Src Addr                  : 192.168.37.16
      Dst Addr                  : 192.168.37.1
      Protocol                  : TCP
      Src Port                  : 56649
      Dst Port                  : 80
      Network RTT               : 2.312      [msec]
      Server RTT                : 18.640     [msec]
PureFlow(A)>
```

統計情報の表示フォーマットは,「PureFlow GSX トラフィックシェーパーNF7101C コマンドリファレンス」の”show topanalysis target”コマンドの表示を参照してください。

なお,トップ集計もシナリオ集計と同様に,測定対象のトラフィックが流れてから,次の集計周期が経過した時に測定結果を集計します。トラフィック分析表示(“show topanalysis target”コマンド)を連続で実行した場合,次の集計周期が経過するまで,同じ集計結果を表示します。

18.4.3 トラフィック分析の測定対象シナリオ

各ポート方向のシナリオ(/port1 や/port2 など)を測定対象にした場合，各シナリオで確認できるネットワーク遅延(Network RTT)とサーバ遅延(Server RTT)を，以下に示します。

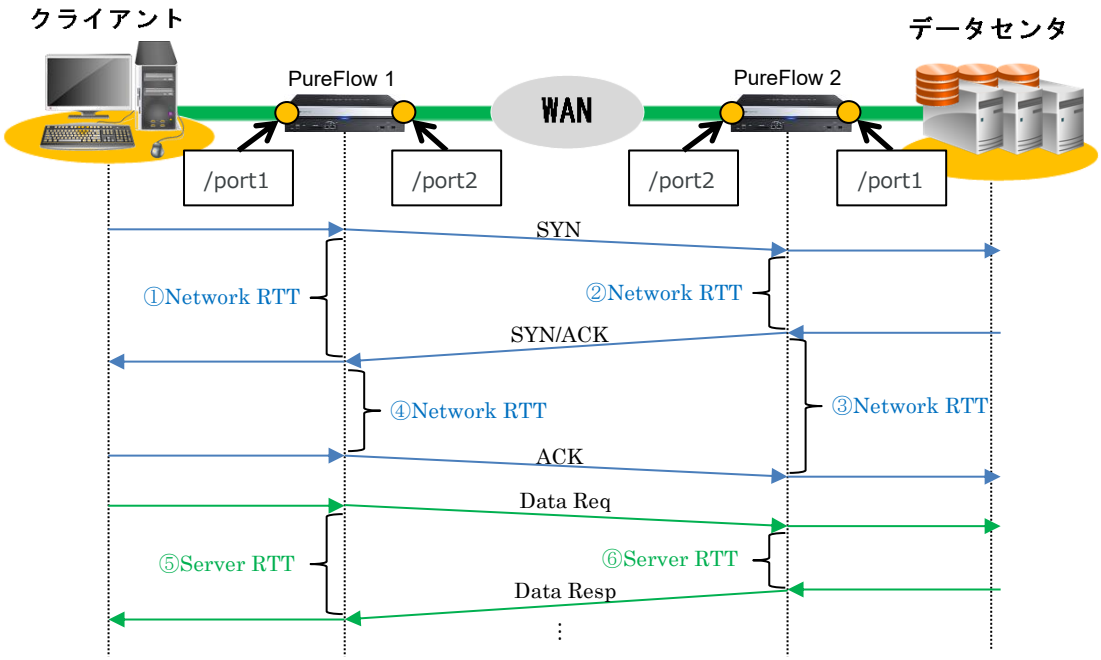


図 18.4.3-1　ネットワーク遅延とサーバ遅延の測定項目

表 18.4.3-1　測定対象シナリオ

パケット種別	PureFlow 1		PureFlow 2	
	/port1 シナリオ (LAN⇒WAN)	/port2 シナリオ (WAN⇒LAN)	/port2 シナリオ (WAN⇒LAN)	/port1 シナリオ (LAN⇒WAN)
SYN	①Network RTT	－	②Network RTT	
SYN/ACK		④Network RTT		③Network RTT
ACK	－			
Data Req	⑤Server RTT	－	⑥Server RTT	
Data Resp		－		

なお，ICMP のネットワーク遅延(Network RTT)も同様です。

次にデータ損失（Segments lost）とデータ再送（Segments Retransmitted）を，以下に示します。

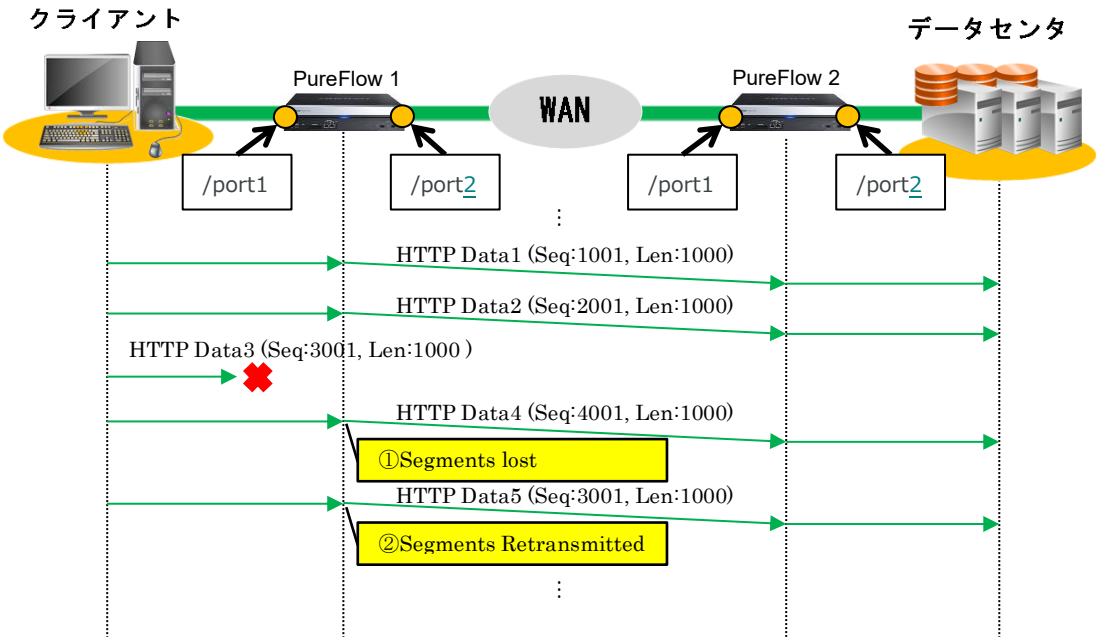


表 20.4.3-2　測定対象シナリオ

パケット種別	PureFlow 1		PureFlow 2	
	/port1 シナリオ (LAN⇒WAN)	/port2 シナリオ (WAN⇒LAN)	/port2 シナリオ (WAN⇒LAN)	/port1 シナリオ (LAN⇒WAN)
HTTD Data1	—	—	—	—
HTTP Data2	—	—	—	—
HTTP Data3	—	—	—	—
HTTP Data4	①Segments lost	—	①Segments lost	—
HTTP Data5	②Segments Retransmitted	—	②Segments Retransmitted	—

18.5 トラフィック生成機能

トラフィック生成機能は，ネットワークに接続されたサーバや端末などに対し，本装置から定期的にトラフィックを生成する機能です。この機能を使うことにより，端末からの通信が全くない状態が継続する場合でも，トラフィック分析を実施し，転送遅延などの測定を継続することができます。

18.5.1 生成するトラフィックの種類

トラフィック生成機能は， HTTP/HTTPS トラフィックと ICMP トラフィックを生成します。これらのトラフィックについて説明します。

(1)HTTP/HTTPS トラフィック

TCP を使った HTTP/HTTPS トラフィックを定期的に生成します。生成する周期はデフォルトで1分です。生成周期が経過するたびに TCP セッション接続，HTTP/HTTPS コマンドの送信， TCP セッション切断を行います。

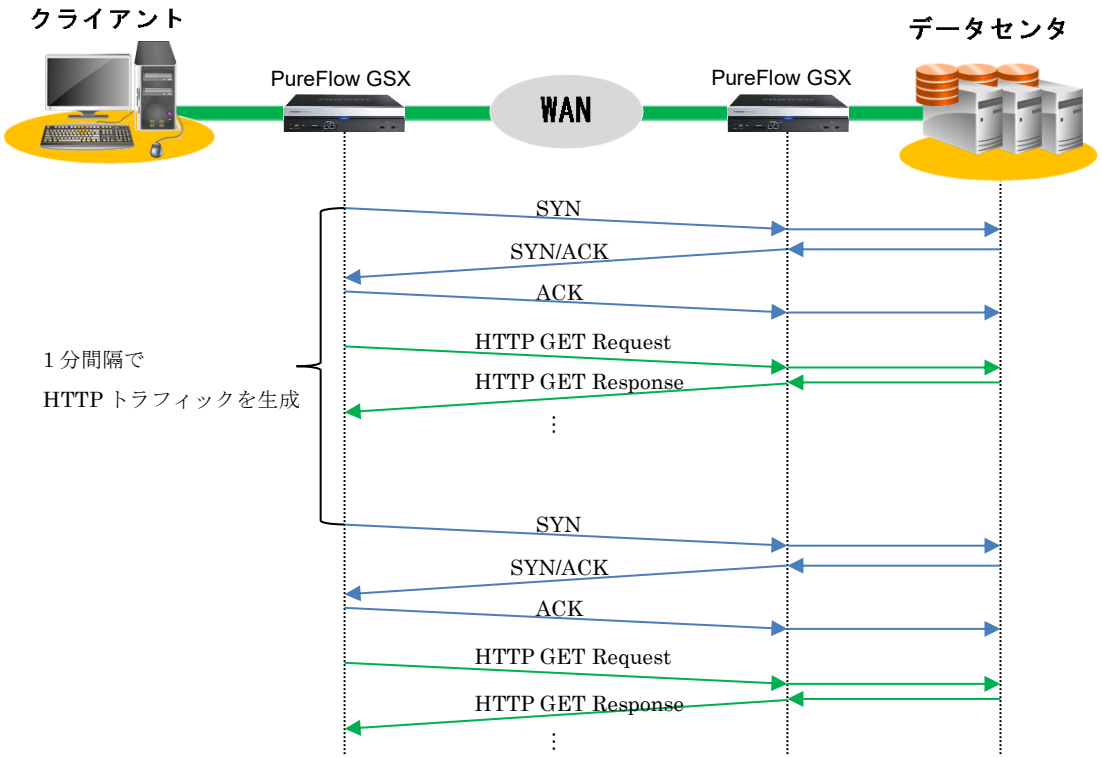


図 18.5.1-1　HTTP トラフィック

(2)ICMP echoトラフィック

ICMP echo トラフィックを定期的に生成します。生成する周期はデフォルトで1分です。生成周期が経過するたびに ICMP echo 要求を送信します。

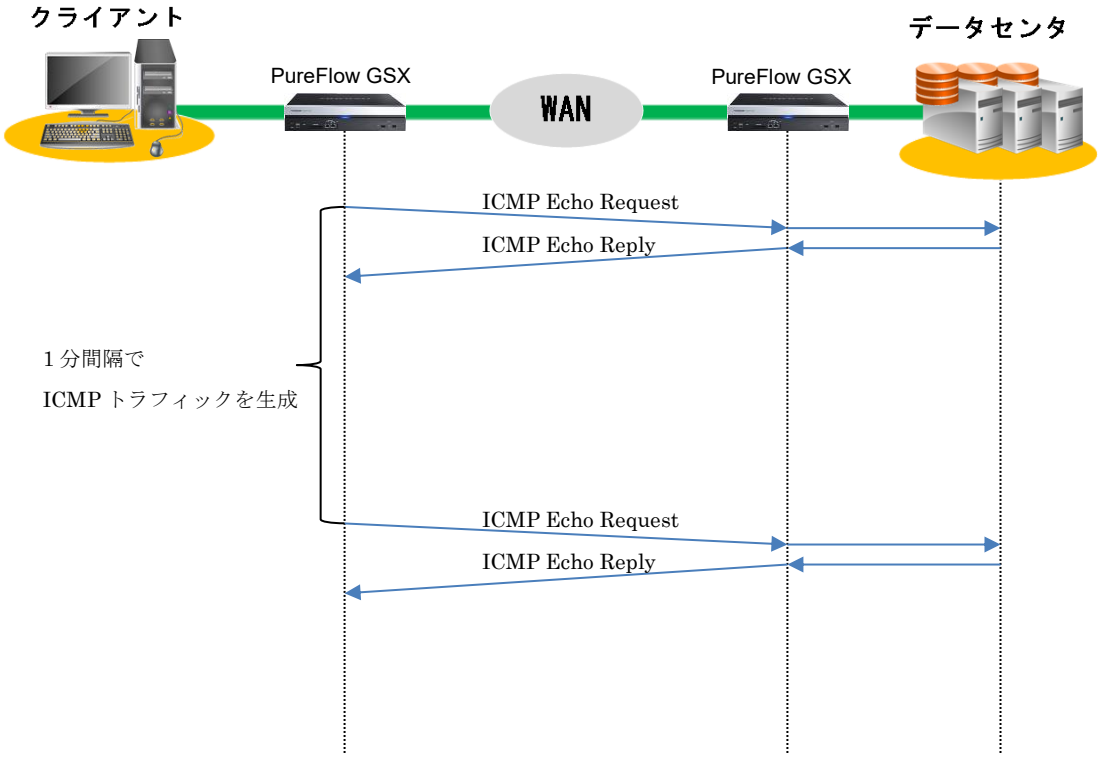


図 18.5.1-2　ICMP トラフィック

18.5.2 トラフィックを送信するインタフェース

トラフィック生成機能は，本装置のシステムインタフェースからトラフィックを送信します。

トラフィック生成機能で送信するトラフィックをトラフィック分析の測定対象とする場合は，システムインタフェースのポート設定(“set ip sytsem port”コマンド)において，“network”を指定してご利用ください。”ethernet“を指定されている場合は，システムインタフェースから送信されるトラフィックが，本装置のNetwork ポートを経由するように，ネットワークケーブルを接続してください。

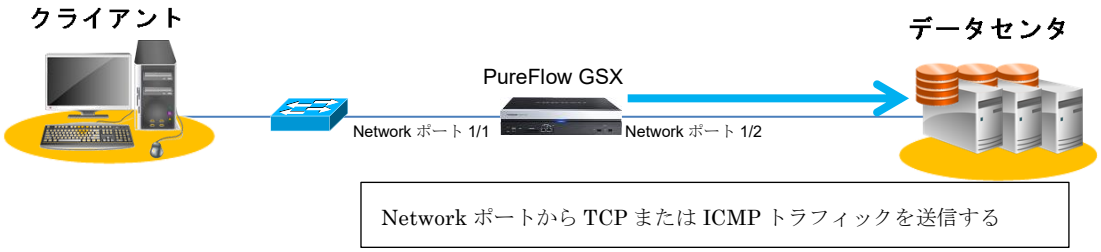


図 18.5.2-1　システムインタフェースのポート設定”が”network“の場合

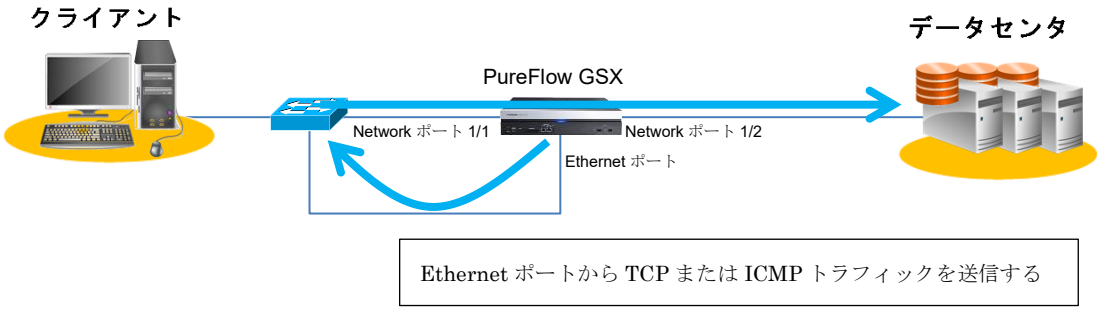


図 18.5.2-2　システムインタフェースのポート設定”が”ethernet“の場合

生成したトラフィックを送受信する両 Network ポート，または，両ポートに属するシナリオをトラフィック分析の測定対象として追加してください。

18.5.3 トラフィック生成の設定方法

トラフィック生成機能の設定方法を説明します。

まず，システムインタフェースを設定します。システムインタフェースのパケットを Network ポートで送受信する場合の例です。すでに設定されている場合は，不要です。

生成するトラフィックをトラフィック分析の測定対象とする場合，シナリオによるトラフィックコントロールを有効に設定します。

```
PureFlow(A)＞ set ip system port network scenario enable
PureFlow(A)＞ set ip system port network in all
PureFlow(A)＞ set ip system 192.168.37.10 netmask 255.255.255.0 up
PureFlow(A)＞ set ip system gateway 192.168.37.1
```

トラフィック分析を有効に設定します。

```
PureFlow(A)＞ set analysis enable
```

次に，トラフィックを生成する設定(“add analysis traffic_generator”コマンド)を追加します。

HTTP サーバ 192.168.37.20 に対して HTTP コマンドを定期的に生成する場合，以下のように設定します。

```
PureFlow(A)＞ add analysis traffic_generator ipv4 dip 192.168.37.20 normalhttp url
index.html
```

送信するトラフィックの種類を追加するときは，設定を追加します。以下は，ICMP トラフィックを追加するときの例です。

```
PureFlow(A)＞ add analysis traffic_generator ipv4 dip 192.168.37.20 icmp
```

送信するトラフィックは最大 25 個まで追加できます。

生成するトラフィックをトラフィック分析の測定対象とする場合，両 Network ポートに属するシナリオとフィルタを追加し，測定対象シナリオを追加します。

例えば，生成する HTTP トラフィックを測定対象とする場合は，以下のように両 Network ポートにシナリオとフィルタを設定し，これらのシナリオを測定対象シナリオとして追加します。フィルタ設定においては，宛先IPアドレス，送信元IPアドレス，プロトコル，HTTP のポート番号を指定します。

```
PureFlow(A)＞ add scenario "/port1/ana1" action forward
PureFlow(A)＞ add filter scenario "/port1/ana1" filter "http1" ipv4 sip 192.168.37.10 dip
192.168.37.20 proto tcp dport 80
```

```
PureFlow(A)＞ add filter scenario "/port1/ana1" filter "http2" ipv4 sip 192.168.37.20 dip
192.168.37.10 proto tcp sport 80
PureFlow(A)＞ add scenario "/port2/ana1" action forward
PureFlow(A)＞ add filter scenario "/port2/ana1" filter "http1" ipv4 sip 192.168.37.10 dip
192.168.37.20 proto tcp dport 80
PureFlow(A)＞ add filter scenario "/port2/ana1" filter "http2" ipv4 sip 192.168.37.20 dip
192.168.37.10 proto tcp sport 80
PureFlow(A)＞
PureFlow(A)＞ add topanalysis target scenario "/port1/ana1"  flow
PureFlow(A)＞ add topanalysis target scenario "/port2/ana1"  flow
```

また，生成するHTTPトラフィックに加え，ICMPトラフィックも測定対象とする場合は、上記の設定に加え，以下のようにICMPのフィルタ設定を追加します。

```
PureFlow(A)＞ add filter scenario "/port1/ana1" filter "icmp1" ipv4 sip 192.168.37.10 dip
192.168.37.20 proto icmp
PureFlow(A)＞ add filter scenario "/port1/ana1" filter "icmp2" ipv4 sip 192.168.37.20 dip
192.168.37.10 proto icmp
PureFlow(A)＞ add filter scenario "/port2/ana1" filter "icmp1" ipv4 sip 192.168.37.10 dip
192.168.37.20 proto icmp
PureFlow(A)＞ add filter scenario "/port2/ana1" filter "icmp2" ipv4 sip 192.168.37.20 dip
192.168.37.10 proto icmp
```

18.6 注意事項

トラフィック分析を使用する際の注意事項を説明します。

(1)　フィルタモードの設定について

フィルタモード設定(“set filter mode”コマンド)において，sip,dip,proto,sport,dport のすべてが含まれない場合，トラフィック分析を実行しません。また，フィルタモード設定(“set filter mode”コマンド)において，tos,cos が含まれる場合，トラフィック分析を実行しません。

(2)　測定対象シナリオのシナリオモードについて

トラフィック分析は，シナリオの種別として”aggregate”，”individual”，”forward”のいずれかが設定されたシナリオを測定対象としたとき，トラフィック分析を実行します。シナリオ種別として”discard”が指定されたシナリオの場合，トラフィック分析を実行しません。

(3)　トップ集計のセッション数について

トップ集計において，シナリオに 100 セッション以上のトラフィックがある場合，集計できないセッションが発生します。トップ集計を使用する場合は，シナリオに設定するフィルタ設定(“add filter scenario”コマンド)でパケット分類を細分化し，シナリオを通過するセッション数を調整してください。

(4)　トラフィック生成機能について

トラフィック分析が無効の場合は，トラフィック生成設定(“add analysis traffic_generator”コマンド)で設定されたトラフィックは送信されません。また，システムインタフェースの通信ポート設定が Network ポート経由で，トラフィック生成機能を使用する場合，測定対象シナリオで制御する必要があるため，“set ip system network port scenario”コマンドを“enable(有効)”に設定してください。パケット送受信に対する測定対象シナリオ方向(/port1 や/port2 など)については「第 7 章 システムインタフェースの設定」を参照してください。

(5)　システムインタフェース通信のポート設定との関係

Network ポート経由のシステムインタフェース通信は，当該出力ポートが無効設定の場合，システムインタフェース通信不可のため，トラフィック生成機能によるトラフィック分析もできません。
例えば，出力ポートが Network ポート 1/2 の場合，以下のようになります。
①Network ポート経由ですべてのポート(1/1, 1/2, 1/3, 1/4)を有効
　set ip system port network in all vid none inner-vid none　⇒トラフィック分析可能

②Network ポート経由でポート(1/1)のみ有効
　set ip system port network in 1/1 vid none inner-vid none　⇒トラフィック分析不可

③Network ポート経由でポート(1/2)のみ有効
　set ip system port network in 1/2 vid none inner-vid none　⇒トラフィック分析可能

（空白ページ）

付録A　デフォルト値

本装置には，機能に応じていくつもの設定項目があります。項目の中にはその機能を使用しない限り設定する必要のないものもありますが，設定が必須なものもあります。設定値を必要とする項目については，あらかじめ値が設定されています。表 1 に設定項目と設定値を示します。コマンドの詳細については「PureFlow GSX トラフィックシェーパーNF7101C コマンドリファレンス」を参照してください。

表 1　デフォルト値一覧

設定項目	コマンド	既設定値	設定範囲
ユーザ名	ユーザ名	root	設定なし
プロンプト	set prompt	PureFlow	最大 15 文字
ボーレート	set console baudrate	9600 bps	9600/19200/38400/ 115200 bps
ページャ	set pager	enable	enable/disable
オートログアウト	set autologout time	10 分	1～30 分
パスワード	set password	（なし）	最大 16 文字
	set adminpassword	（なし）	最大 16 文字
Network ポート設定 （1000BASE-T SFP のみ適用）	set port autonegotiation	enable	enable/disable
	set port speed	1G	1G/100M/10M
	set port duplex	full	full/half
フロー コントロール	set port flow_control	auto	auto Pause 受信 on/off Pause 送信 on/off
最大パケット長	set port maxpacketlen	2048	2048/10240
Ethernet ポート設定	set port autonegotiation system	enable	enable/disable
	set port speed system	1G	1G/100M/10M
	set port duplex system	full	full/half
SYSLOG	set syslog host	disable	enable/disable
	add syslog host (IP Address)	（なし）	IP Address
	add syslog host (UDP port)	514	1～65534
	set syslog severity	notice（5）	0～6
	set syslog facility ccpu	16（local0）	0～23
	set syslog facility fcpu	17（local1）	0～23

設定項目	コマンド	既設定値	設定範囲
SNMP	set snmp syscontact	Not Yet Set	最大 200 文字
	set snmp syslocation	Not Yet Set	最大 200 文字
	set snmp sysname	Not Yet Set	最大 200 文字
	set snmp traps	すべて enable	トラップごとに enable/disable
	add snmp view	(なし)	view レコード名 OID included/excluded
	add snmp community	(なし)	コミュニティ名 バージョン View 名 ReadOnly/ReadWrite
	add snmp group	(なし)	グループ名 認証方式 ReadView WriteView NotifyView
	add snmp user	(なし)	ユーザ名 グループ名 認証方式 パスワード
	add snmp host	(なし)	IPv4 Address バージョン 認証方式 ユーザ名/コミュニティ名 Trap/Inform UDP ポート番号 送信ノーティフィケーション
タイムゾーン	set timezone	UTC +09:00	UTC からのオフセット
夏時間	set summertime	(なし)	開始日時 終了日時 オフセット
SNTP	set sntp	disable	enable/disable
	set sntp server	(なし)	IP Address
	set sntp interval	3600 秒	60～86400 秒
RADIUS	set radius auth	disable	enable/disable
	set radius auth timeout	5	1～30 秒
	set radius auth retransmit	3	0～10 回
	set radius auth method	CHAP	CHAP/PAP
RADIUS サーバ	add radius auth server	(なし)	IP アドレス ポート番号 共通鍵 Primary

設定項目	コマンド	既設定値	設定範囲
システム インタフェース	set ip system(IPv4 Address)	192.168.1.1	IPv4 Address
	set ip system(IPv4 netmask)	255.255.255.0	IPv4 Address
	set ip system(IPv4 up/down)	up	up/down
	set ip system(IPv6 Address)	::192.168.1.1	IPv6 Address
	set ip system(IPv6 prefixlen)	64	0～128
	set ip system(IPv6 up/down)	up	up/down
	set ip system port (ethernet/network)	ethernet	ethernet/network
	set ip system port network (in <slot/port>/all)	all (すべての Network ポート)	1/1, 1/2, all ※ 通信ポートが network のとき
	set ip system port network (vid <VID>/none)	none (VLAN Tag なし)	0～4094/none ※ 通信ポートが network のとき
	set ip system port network (tpid <tpid>)	0x8100 (IEEE802.1Q VLAN Tag)	0x0000～0xffff ※ 通信ポートが network のとき
	set ip system port network (inner-vid <VID>/none)	none (Inner-VLAN Tag なし)	0～4094/none ※ 通信ポートが network のとき
	set ip system port network (inner-tpid <tpid>)	0x8100 (IEEE802.1Q VLAN Tag)	0x0000～0xffff ※ 通信ポートが network のとき
	set ip system port network scenario	disable	enable/disable
	set ip system gateway(IPv4)	(なし)	IPv4 Address
	set ip system gateway(IPv6)	(なし)	IPv6 Address
自動リブート	set autoreboot	enable	enable/disable
フロー識別モード	set filter mode	default	default,vid,cos,inner-vid,inner-cos,sip,dip,tos,proto
フローエージングタイム	set agingtime	300 秒	1～1800 秒
通信ギャップモード設定	set bandwidth mode	no_gap	gap/no_gap
リンクダウン転送機能	set lpt	disable	enable/disable
Telnet 接続設定	set telnet	enable	enable/disable
SSH 接続設定	set ssh	enable	enable/disable
トップカウンタ	set topcounter	disable	enable/disable
	set topcounter config interval time	5 分	1/5/60/180/1440 分

設定項目	コマンド	既設定値	設定範囲
トラフィック分析	set analysis	disable	enable/disable

付録B　SYSLOG 一覧

syslog の一覧を表 2 に示します。表 2 は severity（カッコ内は重大度）ごとにまとめています。

（参考）

syslog メッセージには括弧（[ ]や< >）で囲まれた 16 進数が付加されるものがあります。括弧内の 16 進数はソースコード上の位置や変数値を表しており，当社内でのトラブルシューティングで参照します。

表 2　syslog 一覧

Severity	syslog メッセージ	発生条件	対応方法
Emergency (0)	Temperature #N of the system is critical：xx.xx	システムの温度が危険域 (#N は 1～5) (xx.xx は温度(℃))	このまま使用を続けるとハードウェアが損傷を受ける可能性があります。ただちに電源を落としてください。
Alert (1)	Temperature #N of the system is OK：xx.xx	システムの温度範囲が正常値に復帰 (#N は 1～5) (xx.xx は温度(℃))	回復措置は不要です。
	Temperature #N of the system is abnormal：xx.xx	システムの温度が異常 (#N は 1～5) (xx.xx は温度(℃))	設置環境の温度が範囲内(0～40℃)であることを確認してください。 範囲内である場合は装置を交換してください。範囲外である場合は設置場所を変えてください。
	Power #N inserted	電源ユニットの装着 (#N は 0 または 1)	回復措置は不要です。
	Power #N removed	電源ユニットの抜去 (#N は 0 または 1)	回復措置は不要です。
	Power #N failed	電源ユニットの異常検出 (#N は 0 または 1)	下記を確認してください。 ・ 電源ケーブルは接続されているか。 ・ 供給電圧は規定内(AC 100 V～AC 127 V/ AC 200 V～AC 240 V)か。 ・ 電源ファンは回転しているか。
	Power #N OK	電源ユニットの異常回復 (#N は 0 または 1)	回復措置は不要です。
	Fan #N inserted	ファンユニットの装着 (#N は 0 または 1)	回復措置は不要です。
	Fan #N removed	ファンユニットの抜去 (#N は 0 または 1)	回復措置は不要です。
	Fan #N failed	ファンユニットの異常検出 (#N は 0 または 1)	下記を確認してください。 ・ ファンは回転しているか。
	Fan #N OK	ファンユニットの異常回復 (#N は 0 または 1)	回復措置は不要です。
	No response from Slot #N	モジュールからの応答なし (#N は 1)	弊社サポートまでご連絡ください。
	Slot #N response is OK	モジュールからの応答が回復 (#N は 1)	回復措置は不要です。

Severity	syslog メッセージ	発生条件	対応方法
Alert (1) (続き)	System Buffer %s almost full	システムバッファ%s のバッファ使用量が 90%を超過した。	トラフィック状況および各種設定をチェックしてください。
	System Buffer %s recoverd	システムバッファ%s のバッファ使用量が 90%を超えたあと，50%を下回った。	回復措置は不要です。
	Critical error on FCPU Core[#N], Code[#M] Data1[0xxxxxxxxx] Data2[0xxxxxxxxx]	FCPU でコア停止異常が発生した。	弊社サポートまでご連絡ください。
	Queue blocktime exceeded. [S:#M Q:#Q]	シナリオ M で生成されたキューQ のパケット送出停止を検出した。	弊社サポートまでご連絡ください。
	Detected FCPU IIC error on port[#N/#M]	FCPU で IIC インタフェースの異常が発生した。 (#N は 1) (#M は 1～2)	弊社サポートまでご連絡ください。
Error (3)	CLI Command %s, failed during restoration %msg	起動時のコンフィギュレーションリストアでコマンド%s のエラーが発生した。エラーメッセージは%msg。	弊社サポートまでご連絡ください。
	Detected Queue blocking.	装置内の個別キューでパケット滞留が検出された。	弊社サポートまでご連絡ください。
Notice (5)	The buffer of queue exceeded the limit. [S:#M,Q:#Q]	シナリオ M で生成されたキューQ のパケットバッファ使用量が制限値を超過した。	キューバッファフルのためパケット廃棄が発生しています。入力バースト長の設定をチェックしてください。
	The buffer of queue is less than 50% of the limit.[S:#M,Q:#Q]	シナリオ M で生成されたキューQ のパケットバッファ使用量が制限値を超えたあと，制限値の 50%を下回った。	回復措置は不要です。
	Flow registration failure for the system.	装置内のフローが最大数を超えた。	トラフィック状況および各種設定をチェックしてください。
	Flow registration available for the system.	装置内のフローが最大数に達したあと，最大数の 50%を下回った。	回復措置は不要です。
	Queue allocation failure for the system.	装置内の個別キューが最大数を超えた。	個別キューが装置の最大数に達しているため最大数超過時のアクションを適用しています。トラフィック状況をチェックしてください。
	Queue allocation available for the system.	装置内の個別キューが最大数に達したあと，最大数の 90%を下回った。	回復措置は不要です。
	Queue allocation failure for the scenario.[S:#M]	シナリオ M の個別キュー数が制限値を超過した。	個別キューがシナリオの制限数に達しているため最大数超過時のアクションを適用しています。トラフィック状況をチェックしてください。

Severity	syslog メッセージ	発生条件	対応方法
Notice (5) (続き)	Queue allocation available for the scenario. [S:#M]	シナリオ M の個別キューが制限値に達したあと，制限値の 50%を下回った。	回復措置は不要です。
	Domain IP Entry exceeded the limit for the system.	装置内のドメイン IP エントリが最大数を超えた。	ドメイン IP エントリが装置の最大数に達しています。トラフィック状況を確認してください。
	Domain IP Entry is more than 80% of the limit for the system.	装置内のドメイン IP エントリが 80%を超えた。	ドメイン IP エントリが装置の最大数の 80%に達しています。トラフィック状況を確認してください。
	Domain IP Entry is less than 70% of the limit for the system.	装置内のドメイン IP エントリが 80%に達したあと，最大数の 70%を下回った。	回復措置は不要です。
	Domain IP Entry exceeded the limit for the rule list.[#M]	ルールリスト M のドメイン IP エントリが最大数を超えた。 #M は，ルールリスト名	ルールリスト M のドメイン IP エントリが装置の最大数に達しています。IP エントリの学習状況を確認し，複数のルールリストエントリやワイルドカードで指定している場合はルールリストを分割するなどの措置を行ってください。
	Detected MCU-C failure[xx]	MCU-C でエラーを検出した。	弊社サポートまでご連絡ください。
	Detected MCU-C recovery	MCU-C で検出したエラーが回復した。	回復措置は不要です。
	Detected MCU-S failure[xx]	MCU-S でエラーを検出した。	弊社サポートまでご連絡ください。
	Detected MCU-S recovery	MCU-S で検出したエラーが回復した。	回復措置は不要です。
	Session limits between monitoring manager occurred.	モニタリングマネージャ 2 の接続数制限を超過した。	下記の制限値を超えるとモニタリングマネージャ 2 での情報収集ができない場合があります。制限値を超過しないように使用してください。 収集周期 シナリオ数 モニタリングマネージャ 2 接続数 10 秒 2000 2 10 秒 4000 1 30 秒 10000 2 30 秒 20000 1 60 秒 制限なし 4
	Session limits between monitoring manager is released.	モニタリングマネージャ 2 の接続数制限を超過したあと，制限を下回った。	回復措置は不要です。
	Terminate monitoring manager session due to System Packet Buffer full.	システムインタフェースの通信ポートが Network であり，System Packet Buffer の使用量が制限値を超過したため，モニタリングマネージャ 2 との接続を切断した。	System Packet Buffer の使用量が制限値を超過している状態ではシステムインタフェースの通信ができません。トラフィック状況および各種設定をチェックしてください。 System Packet Buffer の使用量回復後は自動的に再接続されます。

Severity	syslog メッセージ	発生条件	対応方法
Notice (5) （続き）	Monitoring manager session connected. (xxx.xxx.xxx.xxx)	モニタリングマネージャ 2（xxx.xxx.xxx.xxx）と接続した。	回復措置は不要です。
	Monitoring manager session disconnected[State:#N]. (xxx.xxx.xxx.xxx)	モニタリングマネージャ 2（xxx.xxx.xxx.xxx）との接続を切断した。（State:#N は通信状態）	モニタリングマネージャ 2 のノード登録や接続状況をチェックしてください。
	Exceeds max no. of sessions.	Telnet または SSH セッションの接続数制限を超過した。	Telnet と SSH を合わせ，最大 4 セッションまで同時利用可能です。制限値を超過しないように使用してください。
Informational (6)	Pipe #N changed Operate from Down.	パイプがリンクアップ（#N は 1）	回復措置は不要です。
	Pipe #N changed Down from Operate.	パイプがリンクダウン（#N は 1）	下記を確認してください。 ・ 該当箇所が LinkDown を起こしていないか。

Severity	syslog メッセージ	発生条件	対応方法
Informational (6) (続き)	Port #N/#M changed Up from Down.	ポートがリンクアップ (#N は 1) (#M は 1～2)	回復措置は不要です。
	Port #N/#M changed Down from Up.	ポートがリンクダウン (#N は 1) (#M は 1～2)	下記を確認してください。 ・ ケーブル断は起きていないか。 ・ 正しいケーブル(マルチモード／シングルモード, ストレート／クロス)を使用しているか。 ・ Network ポートの Speed／Duplex および Pause の設定が接続装置と合っているか。
	Port #N/#M changed PowerDown with Link Pass Through.	リンクダウン転送機能が動作 (#N は 1) (#M は 1～2)	下記を確認してください。 ・ ケーブル断は起きていないか。 ・ 正しいケーブル(マルチモード／シングルモード, ストレート／クロス)を使用しているか。 ・ Network ポートの Speed／Duplex および Pause の設定が接続装置と合っているか。
	Warning. Port #N/#M Oper duplex is Half.	ポートが半二重でリンクアップ (#N は 1) (#M は 1～4)	下記を確認してください。 ・ Network ポートの Speed／Duplex の設定が接続装置と合っているか。
	Management Ethernet Port changed Up from Down.	Management Ethernet ポートがリンクアップ	回復措置は不要です。
	Management Ethernet Port changed Down from Up.	Management Ethernet ポートがリンクダウン	下記を確認してください。 ・ ケーブル断は起きていないか。 ・ 正しいケーブルを使用しているか。
	AnritsuPureFlow Software Version x.x.x.x	装置起動	回復措置は不要です。
	User %s authentication from RADIUS server was Accept	ユーザ名%s の RADIUS 認証が accept された。	回復措置は不要です。
	User %s authentication from RADIUS server was Reject	ユーザ名%s の RADIUS 認証が reject された。	回復措置は不要です。
	User %s authentication from RADIUS server was Timeout	ユーザ名%s の RADIUS 認証がタイムアウトした。	回復措置は不要です。
	User root logged in by SSH(xxx.xxx.xxx.xxx)	SSH host のユーザが本装置にログイン	回復措置は不要です。
	User root logged in by TELNET	TELNET host のユーザが本装置にログイン	回復措置は不要です。

（空白ページ）

付録C　SNMP Trap 一覧

SNMP Trap の一覧を表 3 に示します。

Trap は有効に設定されているもののみ送出されます。Trap の有効／無効の設定は，“set snmp traps”コマンドを使用して設定します。コマンドの詳細については，「PureFlow GSX トラフィックシェーパー NF7101C コマンドリファレンス」を参照してください。

表 3　SNMP Trap 一覧

MIB オブジェクト名	コマンドでの設定名	発生条件	対応方法
coldStart(1.3.6.1.6.3.1.1.5.1)	coldstart	装置起動完了	下記を確認してください。 ・電源断は発生していないか。 ・リセットボタンが押されていないか。 ・再起動コマンドを実行していないか。 ・自動リブート機能が働いていないか。
warmStart(1.3.6.1.6.3.1.1.5.2)	warmstart	出力されません。	
linkDown(1.3.6.1.6.3.1.1.5.3)	linkdown	ポートのリンクダウン	下記を確認してください。 ・ケーブルが切断されていないか。 ・正しいケーブル（シングルモード／マルチモード，ストレート／クロス）を使用しているか。 ・Network ポートの Speed/Duplex および Pause の設定が接続装置と整合が取れているか。
linkUp(1.3.6.1.6.3.1.1.5.4)	linkup	リンクアップ	回復措置は不要です。
authenticationFailure(1.3.6.1.6.3.1.1.5.5)	authentication	SNMP の不正アクセス検出	本装置に設定したアクセス許可 comunity 名，IP address，レベル（get/set）が，SNMP manager 側と整合が取れているか確認してください。
pfGsPowerInsertEvent(1.3.6.1.4.1.1151.2.1.7.20.0.3)	powerinsert	電源ユニットの装着	回復措置は不要です。
pfGsPowerExtractEvent(1.3.6.1.4.1.1151.2.1.7.20.0.4)	powerextract	電源ユニットの抜去	回復措置は不要です。

MIB オブジェクト名	コマンドでの設定名	発生条件	対応方法
pfGsPowerFailureEvent(1.3.6.1.4.1.1151.2.1.7.20.0.5)	powerfailure	電源ユニットの異常検出	下記を確認してください。 ・ 電源ケーブルは接続されているか。 ・ 供給電圧は規定内（AC 100 V~ AC 127 V/ AC 200 V ~ AC 240 V）か。 ・ 電源ファンは回転しているか。
pfGsPowerRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.6)	powerrecovery	電源ユニットの異常回復	回復措置は不要です。
pfGsModuleFailureAlarmEvent(1.3.6.1.4.1.1151.2.1.7.20.0.7)	modulefailurealarm	モジュール異常の検出	弊社サポートまでご連絡ください。
pfGsModuleFailureRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.8)	modulefailurerecovery	モジュール異常の回復	回復措置は不要です。
pfGsFanInsertEvent(1.3.6.1.4.1.1151.2.1.7.20.0.11)	faninsert	ファンユニットの装着	回復措置は不要です。
pfGsFanExtractEvent(1.3.6.1.4.1.1151.2.1.7.20.0.12)	fanextract	ファンユニットの抜去	回復措置は不要です。
pfGsFanFailureEvent(1.3.6.1.4.1.1151.2.1.7.20.0.13)	fanfailure	ファンユニットの異常検出	下記を確認してください。 ・ ファンは回転しているか。
pfGsFanRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.14)	fanrecovery	ファンユニットの異常回復	回復措置は不要です。
pfGsQueueBuffAlarmEvent(1.3.6.1.4.1.1151.2.1.7.20.0.15)	queuebuffalarm	当該シナリオのパケットバッファ使用量が制限値を超過した。	キューバッファフルのためパケット廃棄が発生しています。入力バースト長の設定をチェックしてください。
pfGsQueueBuffRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.16)	queuebuffrecovery	当該シナリオのパケットバッファ使用量が制限値を超えたあと，制限値の 50%を下回った。	回復措置は不要です。
pfGsSystemBuffAlarmEvent(1.3.6.1.4.1.1151.2.1.7.20.0.17)	systembuffalarm	当該システムバッファのバッファ使用量が 90%を超過した。	トラフィック状況，および各種設定をチェックしてください。
pfGsSystemBuffRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.18)	systembuffrecovery	当該システムバッファのバッファ使用量が 90%を超えたあと，50%を下回った。	回復措置は不要です。
pfGsxSystemHeatAlarmEvent(1.3.6.1.4.1.1151.2.1.7.20.0.19)	systemheatalarm	システム温度が 50℃を超えた，または–5 ℃を下回った。	環境温度が 40℃以下，および 0℃以上になるように空調または機器配置を見直してください。

MIB オブジェクト名	コマンドでの設定名	発生条件	対応方法
pfGsxSystemHeatRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.20)	systemheatrecovery	システム温度が 50℃を超えたあと，45℃を下回った。または–5℃を下回ったあと，0℃を超えた。	回復措置は不要です。
pfGsIndividualQueueAlarmEvent(1.3.6.1.4.1.1151.2.1.7.20.0.21)	queueallocalarm	装置内の個別キューが最大数を超えた。	個別キューが装置の最大数に達しているため最大数超過時のアクションを適用しています。トラフィック状況をチェックしてください。
pfGsIndividualQueueRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.22)	queueallocrecovery	装置内の個別キューが最大数に達したあと，最大数の 90%を下回った。	回復処置は不要です。
pfGsMaxQnumAlarmEvent(1.3.6.1.4.1.1151.2.1.7.20.0.23)	maxqnumalarm	当該シナリオの個別キュー数が制限値を超過した。	個別キューがシナリオの制限数に達しているため最大数超過時のアクションを適用しています。トラフィック状況をチェックしてください。
pfGsMaxQnumRecoveryEvent(1.3.6.1.4.1.1151.2.1.7.20.0.24)	maxqnumrecovery	当該シナリオの個別キューが制限値に達したあと，制限値の 50%を下回った。	回復処置は不要です。

（空白ページ）

付録D　Enterprise MIB 一覧

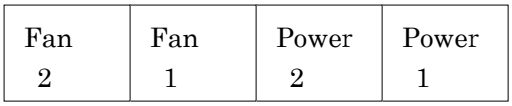
PureFlow GSX の Enterprise MIB オブジェクト一覧を表 4 に示します。

表 4　PureFlow GSX Enterprise MIB 一覧

MIB グループ	MIB オブジェクト名	説明
pureFlowGsMib		PureFlow GS Enterprise MIB ツリーです。オブジェクト ID は 1.3.6.1.4.1.1151.2.1.7 です。 以下にツリー内のオブジェクトと，そのオブジェクト ID（カッコ内の値）を示します。 PureFlow GS Enterprise MIB ツリーは PureFlow GS シリーズ共通の MIB ツリーです。本書では PureFlow GSX の MIB オブジェクトを示します。
pfGsSystem(1.3.6.1.4.1.1151.2.1.7.1)	pfGsSystemType(1.3.6.1.4.1.1151.2.1.7.1.1)	システムソフトウェアの形名を表します。 nf7101s003a(3)　：NF7101-S003A
	pfGsSystemSlotNumber(1.3.6.1.4.1.1151.2.1.7.1.2)	モジュールを実装するスロットの数を表します。
	pfGsSystemSoftwareRev(1.3.6.1.4.1.1151.2.1.7.1.3)	システムソフトウェアのバージョンを表します。
	pfGsSystemOperationTime(1.3.6.1.4.1.1151.2.1.7.1.5)	装置が起動してからの経過時間を表します。単位は 10 ms です。この MIB オブジェクトは 1 時間ごとに更新されます。したがって，時間以下の単位は常に 0 となります。
	pfGsSystemCcpu5sec(1.3.6.1.4.1.1151.2.1.7.1.6)	制御系 CPU の CPU 使用率を，最近 5 秒の平均値で表します。
	pfGsSystemCcpu1min(1.3.6.1.4.1.1151.2.1.7.1.7)	制御系 CPU の CPU 使用率を，最近 1 分の平均値で表します。
	pfGsSystemCcpu5min(1.3.6.1.4.1.1151.2.1.7.1.8)	制御系 CPU の CPU 使用率を，最近 5 分の平均値で表します。
	pfGsSystemCcpuMemory5sec(1.3.6.1.4.1.1151.2.1.7.1.9)	制御系 CPU のメモリ使用率を，最近 5 秒の平均値で表します。
	pfGsSystemCcpuMemory1min(1.3.6.1.4.1.1151.2.1.7.1.10)	制御系 CPU のメモリ使用率を，最近 1 分の平均値で表します。
	pfGsSystemCcpuMemory5min(1.3.6.1.4.1.1151.2.1.7.1.11)	制御系 CPU のメモリ使用率を，最近 5 分の平均値で表します。
	pfGsSystemFcpuTable(1.3.6.1.4.1.1151.2.1.7.1.12)	フォワーディング系 CPU の CPU およびメモリ使用率のテーブルです。 このテーブルには以下のオブジェクトが含まれています。
	pfGsSystemFcpuEntry(1.3.6.1.4.1.1151.2.1.7.1.12.1)	フォワーディング系 CPU の CPU およびメモリ使用率のエントリテーブルです。テーブルインデックスは pfSystemFcpuIndex です。 このテーブルには以下のオブジェクトが含まれています。

MIB グループ	MIB オブジェクト名	説明
pfGsSystem(1.3.6.1.4.1.1151.2.1.7.1)（続き）	pfGsSystemFcpuIndex(1.3.6.1.4.1.1151.2.1.7.1.12.1.1)	フォワーディング系 CPU の番号を表します。 正面図 1
	pfGsSystemFcpu5sec(1.3.6.1.4.1.1151.2.1.7.1.12.1.2)	フォワーディング系 CPU の CPU 使用率を，最近 5 秒の平均値で表します。
	pfGsSystemFcpu1min(1.3.6.1.4.1.1151.2.1.7.1.12.1.3)	フォワーディング系 CPU の CPU 使用率を，最近 1 分の平均値で表します。
	pfGsSystemFcpu5min(1.3.6.1.4.1.1151.2.1.7.1.12.1.4)	フォワーディング系 CPU の CPU 使用率を，最近 5 分の平均値で表します。
	pfGsSystemFcpuMemory5sec(1.3.6.1.4.1.1151.2.1.7.1.12.1.5)	フォワーディング系 CPU のメモリ使用率を，最近 5 秒の平均値で表します。
	pfGsSystemFcpuMemory1min(1.3.6.1.4.1.1151.2.1.7.1.12.1.6)	フォワーディング系 CPU のメモリ使用率を，最近 1 分の平均値で表します。
	pfGsSystemFcpuMemory5min(1.3.6.1.4.1.1151.2.1.7.1.12.1.7)	フォワーディング系 CPU のメモリ使用率を，最近 5 分の平均値で表します。
	pfGsSystemBuffTable(1.3.6.1.4.1.1151.2.1.7.1.13)	システムバッファのテーブルです。 このテーブルには以下のオブジェクトが含まれています。
	pfGsSystemBuffEntry(1.3.6.1.4.1.1151.2.1.7.1.13.1)	システムバッファのエントリテーブルです。テーブルインデックスは pfGsSystemBuffIndex です。 このテーブルには以下のオブジェクトが含まれています。
	pfGsSystemBuffIndex(1.3.6.1.4.1.1151.2.1.7.1.13.1.1)	システムバッファの番号を表します。 1 :パケットバッファ 2 :帯域制御エンジンのメッセージブロック 3 :パケット出力コマンド領域 4 :インバンドで送信するパケットのパケットバッファ 5 :未使用 6 :未使用 7 :未使用 8 :未使用 9 :処理中のパケットの一時領域
	pfGsSystemBuffMax(1.3.6.1.4.1.1151.2.1.7.1.13.1.2)	システムバッファの最大容量を表します。
	pfGsSystemBuffRemaining(1.3.6.1.4.1.1151.2.1.7.1.13.1.3)	システムバッファの残容量を表します。
	pfGsSystemTempTable(1.3.6.1.4.1.1151.2.1.7.1.14)	システム温度のテーブルです。 このテーブルには以下のオブジェクトが含まれています。

MIB グループ	MIB オブジェクト名	説明
pfGsSystem(1.3.6.1.4.1.1151.2.1.7.1) （続き）	pfGsSystemTempEntry(1.3.6.1.4.1.1151.2.1.7.1.14.1)	システム温度のエントリテーブルです。テーブルインデックスは pfGsSystemTempIndex です。 このテーブルには以下のオブジェクトが含まれています。
	pfGsSystemTempIndex(1.3.6.1.4.1.1151.2.1.7.1.14.1.1)	システム温度の番号を表します。 1 :吸気 2 :未使用 3 :未使用 4 :未使用 5 :未使用 6 :未使用 7 :未使用 8 :未使用 9 :未使用
	pfGsSystemTempValue(1.3.6.1.4.1.1151.2.1.7.1.14.1.2)	システム温度の値を表します。 単位は摂氏です。
pfGsModule(1.3.6.1.4.1.1151.2.1.7.2)	pfGsModuleTable(1.3.6.1.4.1.1151.2.1.7.2.1)	モジュール情報のテーブルです。 このテーブルには以下のオブジェクトが含まれています。
	pfGsModuleEntry(1.3.6.1.4.1.1151.2.1.7.2.1.1)	モジュール情報のエントリテーブルです。テーブルインデックスは pfGsModuleIndex です。 このテーブルには以下のオブジェクトが含まれています。
	pfGsModuleIndex(1.3.6.1.4.1.1151.2.1.7.2.1.1.1)	モジュールの番号を表します。 正面図 1
	pfGsModuleLocation(1.3.6.1.4.1.1151.2.1.7.2.1.1.2)	モジュールの実装スロット番号を表します。 (モジュール番号と同じ値になります) 正面図 1
	pfGsModuleType(1.3.6.1.4.1.1151.2.1.7.2.1.1.3)	モジュールの種別を表します。 unknown(1) :下記以外 empty(2) :未実装 ge2gt(3) :GbE/2T fe2ft(4) :FE/2T xge2sfp(5) :XGE/2SFP
	pfGsModuleDescr(1.3.6.1.4.1.1151.2.1.7.2.1.1.4)	モジュールの名前を表します。
	pfGsModulePortNumber(1.3.6.1.4.1.1151.2.1.7.2.1.1.5)	モジュールの実装ポート数を表します。

<table>
<tr><th>MIB グループ</th><th>MIB オブジェクト名</th><th>説明</th></tr>
<tr><td>pfGsModule(1.3.6.1.4.1.1151.2.1.7.2)
（続き）</td><td>pfGsModuleOperStatus(1.3.6.1.4.1.1151.2.1.7.2.1.1.6)</td><td>モジュールの状態を表します。
other(1) : 下記以外
operational(2) : 正常
malfunctioning(3) : 6 以外の異常
notPresent(4) : 未実装
standby(5) : （未使用です）
notResponding(6) : 応答なし</td></tr>
<tr><td></td><td>pfGsModuleRevision(1.3.6.1.4.1.1151.2.1.7.2.1.1.7)</td><td>モジュールのハードウェアレビジョンを表します。</td></tr>
<tr><td></td><td>pfGsModuleSerialNumber(1.3.6.1.4.1.1151.2.1.7.2.1.1.8)</td><td>モジュールのシリアル番号を表します。</td></tr>
<tr><td>pfGsPower(1.3.6.1.4.1.1151.2.1.7.3)</td><td>pfGsPowerTable(1.3.6.1.4.1.1151.2.1.7.3.1)</td><td>電源ユニット情報のテーブルです。
このテーブルには以下のオブジェクトが含まれています。</td></tr>
<tr><td></td><td>pfGsPowerEntry(1.3.6.1.4.1.1151.2.1.7.3.1.1)</td><td>電源ユニット情報のエントリテーブルです。テーブルインデックスは pfGsPowerIndex です。
このテーブルには以下のオブジェクトが含まれています。</td></tr>
<tr><td></td><td>pfGsPowerIndex(1.3.6.1.4.1.1151.2.1.7.3.1.1.1)</td><td>電源ユニットの番号を表します。
背面図
 
</td></tr>
<tr><td></td><td>pfGsPowerOperStatus(1.3.6.1.4.1.1151.2.1.7.3.1.1.2)</td><td>電源ユニットの状態を表します。
other(1) : 下記以外
operational(2) : 正常
malfunctioning(3) : 異常（入力異常またはファン停止）
notPresent(4) : 未実装
outputerror(5) : （未使用です）
inputerror(6) : （未使用です）
fanfailure(7) : （未使用です）</td></tr>
<tr><td></td><td>pfGsPowerUpTime(1.3.6.1.4.1.1151.2.1.7.3.1.1.3)</td><td>電源ユニットが装着されてからの経過時間を表します。単位は 10 ms です。</td></tr>
<tr><td></td><td>pfGsPowerFanSpeed(1.3.6.1.4.1.1151.2.1.7.3.1.1.4)</td><td>電源ユニットのファンの回転数を表します。単位は RPM です。</td></tr>
<tr><td>pfGsxFan(1.3.6.1.4.1.1151.2.1.7.4)</td><td>pfGsxFanTable(1.3.6.1.4.1.1151.2.1.7.4.1)</td><td>ファンユニット情報のテーブルです。
このテーブルには以下のオブジェクトが含まれています。</td></tr>
<tr><td></td><td>pfGsxFanEntry(1.3.6.1.4.1.1151.2.1.7.4.1.1)</td><td>ファンユニット情報のエントリテーブルです。テーブルインデックスは pfGsFanIndex です。
このテーブルには以下のオブジェクトが含まれています。</td></tr>
</table>

<table>
<tr><th>MIB グループ</th><th>MIB オブジェクト名</th><th>説明</th></tr>
<tr><td rowspan="4">pfGsxFan(1.3.6.1.4.1.1151.2.1.7.4)
（続き）</td><td>pfGsxFanIndex(1.3.6.1.4.1.1151.2.1.7.4.1.1.1)</td><td>ファンユニットの番号を表します。
背面図<table><tr><td>Fan
2</td><td>Fan
1</td><td>Power
2</td><td>Power
1</td></tr></table></td></tr>
<tr><td>pfGsxFanOperStatus(1.3.6.1.4.1.1151.2.1.7.4.1.1.2)</td><td>ファンユニットの状態を表します。
other(1) : 下記以外
operational(2) : 正常
malfunctioning(3) : 異常（ファン停止）
notPresent(4) : 未実装</td></tr>
<tr><td>pfGsxFanUpTime(1.3.6.1.4.1.1151.2.1.7.4.1.1.3)</td><td>ファンユニットが装着されてからの経過時間を表します。単位は 10 ms です。</td></tr>
<tr><td>pfGsxFanSpeed(1.3.6.1.4.1.1151.2.1.7.4.1.1.4)</td><td>ファンユニットのファンの回転数を表します。単位は RPM です。</td></tr>
<tr><td rowspan="3">pfGsFlowInformation(1.3.6.1.4.1.1151.2.1.7.8)</td><td>pfGsFlowInformationResourceTotal(1.3.6.1.4.1.1151.2.1.7.8.1)</td><td>装置で使用可能なフロー数の総数を表示します。</td></tr>
<tr><td>pfGsFlowInformationResourceUsed(1.3.6.1.4.1.1151.2.1.7.8.2)</td><td>装置で使用中のフロー数を表示します。</td></tr>
<tr><td>pfGsFlowInformationResourceAvailable(1.3.6.1.4.1.1151.2.1.7.8.3)</td><td>装置で使用前のフロー数を表示します。</td></tr>
<tr><td rowspan="7">pfGsxScenarioStatistics(1.3.6.1.4.1.1151.2.1.7.9)</td><td>pfGsxScenarioStatisticsTable(1.3.6.1.4.1.1151.2.1.7.9.1)</td><td>シナリオカウンタのテーブルです。
このテーブルには以下のオブジェクトが含まれています。</td></tr>
<tr><td>pfGsxScenarioStatisticsEntry(1.3.6.1.4.1.1151.2.1.7.9.1.1)</td><td>シナリオカウンタのエントリテーブルです。テーブルインデックスは pfGsxScenarioStatisticsScenarioSortIndex です。
このテーブルには以下のオブジェクトが含まれています。
参考）このテーブル内オブジェクトの OID を求める方法をこの表の次に示します。</td></tr>
<tr><td>pfGsxScenarioStatisticsScenarioSortIndex(1.3.6.1.4.1.1151.2.1.7.9.1.1.1)</td><td>シナリオのソート番号を表します。
ソート番号はシナリオ登録／削除時に自動付加されます。
シナリオツリーの並び順に対応した番号になります。</td></tr>
<tr><td>pfGsxScenarioStatisticsScenarioName(1.3.6.1.4.1.1151.2.1.7.9.1.1.2)</td><td>シナリオのシナリオ名を表します。</td></tr>
<tr><td>pfGsxScenarioStatisticsScenarioType(1.3.6.1.4.1.1151.2.1.7.9.1.1.3)</td><td>シナリオのタイプを表します。
discard(0) : 廃棄シナリオ
individual(1) : 個別キューシナリオ
aggregate(2) : 集約キューシナリオ
forward(5) : 転送シナリオ</td></tr>
<tr><td>pfGsxScenarioStatisticsRxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.4)</td><td>シナリオの受信オクテット数を表します。</td></tr>
</table>

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioStatistics(1.3.6.1.4.1.1151.2.1.7.9) （続き）	pfGsxScenarioStatisticsRxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.5)	シナリオの受信パケット数を表します。
	pfGsxScenarioStatisticsTxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.6)	シナリオの送信オクテット数を表します。
	pfGsxScenarioStatisticsTxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.7)	シナリオの送信パケット数を表します。
	pfGsxScenarioStatisticsDiscardOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.8)	シナリオの廃棄オクテット数を表します。
	pfGsxScenarioStatisticsDiscardPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.9)	シナリオの廃棄パケット数を表します。
	pfGsxScenarioStatisticsHCRxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.10)	シナリオの受信オクテット数を 64 ビットで表します。 注)このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsHCRxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.11)	シナリオの受信パケット数を 64 ビットで表します。 注)このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsHCTxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.12)	シナリオの送信オクテット数を 64 ビットで表します。 注)このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsHCTxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.13)	シナリオの送信パケット数を 64 ビットで表します。 注)このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsHCDiscardOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.14)	シナリオの廃棄オクテット数を 64 ビットで表します。 注)このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsHCDiscardPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.15)	シナリオの廃棄パケット数を 64 ビットで表します。
	pfGsxScenarioStatisticsDefaultQueRxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.16)	シナリオのデフォルトキューの受信パケット数を表します。
	pfGsxScenarioStatisticsDefaultQueRxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.17)	シナリオのデフォルトキューの受信パケット数を表します。
	pfGsxScenarioStatisticsDefaultQueTxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.18)	シナリオのデフォルトキューの送信オクテット数を表します。
	pfGsxScenarioStatisticsDefaultQueTxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.19)	シナリオのデフォルトキューの送信パケット数を表します。

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioStatistics(1.3.6.1.4.1.1151.2.1.7.9) （続き）	pfGsxScenarioStatisticsDefaultQueDiscardOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.20)	シナリオのデフォルトキューの廃棄オクテット数を表します。
	pfGsxScenarioStatisticsDefaultQueDiscardPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.21)	シナリオのデフォルトキューの廃棄パケット数を表します。
	pfGsxScenarioStatisticsDefaultQueHCRxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.22)	シナリオのデフォルトキューの受信オクテット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsDefaultQueHCRxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.23)	シナリオのデフォルトキューの受信パケット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsDefaultQueHCTxOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.24)	シナリオのデフォルトキューの送信オクテット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsDefaultQueHCTxPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.25)	シナリオのデフォルトキューの送信パケット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsDefaultQueHCDiscardOctets(1.3.6.1.4.1.1151.2.1.7.9.1.1.26)	シナリオのデフォルトキューの廃棄オクテット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatisticsDefaultQueHCDiscardPackets(1.3.6.1.4.1.1151.2.1.7.9.1.1.27)	シナリオのデフォルトキューの廃棄パケット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
pfGsxScenarioInformation(1.3.6.1.4.1.1151.2.1.7.10)	pfGsxScenarioInformationTable(1.3.6.1.4.1.1151.2.1.7.10.1)	シナリオ情報のテーブルです。 このテーブルには以下のオブジェクトが含まれています。
	pfGsxScenarioInformationEntry(1.3.6.1.4.1.1151.2.1.7.10.1.1)	シナリオ情報のエントリテーブルです。テーブルインデックスは pfGsxScenarioInformationScenarioSortIndex です。 このテーブルには以下のオブジェクトが含まれています。 参考）このテーブル内オブジェクトの OID を求める方法をこの表の次に示します。
	pfGsxScenarioInformationScenarioSortIndex(1.3.6.1.4.1.1151.2.1.7.10.1.1.1)	シナリオのソート番号を表します。 ソート番号はシナリオ登録／削除時に自動付加されます。 シナリオツリーの並び順に対応した番号になります。
	pfGsxScenarioInformationScenarioName(1.3.6.1.4.1.1151.2.1.7.10.1.1.2)	シナリオのシナリオ名を表します。
	pfGsxScenarioInformationScenarioType(1.3.6.1.4.1.1151.2.1.7.10.1.1.3)	シナリオのタイプを表します。 discard(0) :廃棄シナリオ individual(1) :個別キューシナリオ aggregate(2) :集約キューシナリオ forward(5) :転送シナリオ

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioInformation(1.3.6.1.4.1.1151.2.1.7.10) （続き）	pfGsxScenarioInformationDefFlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.4)	シナリオに関連して生成されたデフォルトフローの数を表します。forward シナリオの場合はフローの総数と同じ値になります。
	pfGsxScenarioInformationClass1FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.5)	シナリオに関連して生成されたクラス 1 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationClass2FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.6)	シナリオに関連して生成されたクラス 2 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationClass3FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.7)	シナリオに関連して生成されたクラス 3 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationClass4FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.8)	シナリオに関連して生成されたクラス 4 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationClass5FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.9)	シナリオに関連して生成されたクラス 5 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationClass6FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.10)	シナリオに関連して生成されたクラス 6 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationClass7FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.11)	シナリオに関連して生成されたクラス 7 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationClass8FlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.12)	シナリオに関連して生成されたクラス 8 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInformationTotalFlowNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.13)	シナリオに関連して生成されたフローの総数を表します。
	pfGsxScenarioInformationMaxBuffScenarioId(1.3.6.1.4.1.1151.2.1.7.10.1.1.14)	現在のバッファ使用量が最大のキューについて，該当するキューの QID を示します。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationMaxBuffRatio(1.3.6.1.4.1.1151.2.1.7.10.1.1.15)	現在のバッファ使用量が最大のキューについて，該当するキューの最大バッファサイズに対するバッファ使用率を表します。単位は%です。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationMaxBuff(1.3.6.1.4.1.1151.2.1.7.10.1.1.16)	現在のバッファ使用量が最大のキューについて，該当するキューのバッファ使用量を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationMinBuffScenarioId(1.3.6.1.4.1.1151.2.1.7.10.1.1.17)	現在のバッファ使用量が最小のキューについて，該当するキューの QID を示します。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationMinBuffRatio(1.3.6.1.4.1.1151.2.1.7.10.1.1.18)	現在のバッファ使用量が最小のキューについて，該当するキューの最大バッファサイズに対するバッファ使用率を表します。単位は%です。 個別キューモード以外のシナリオでは 0 固定です。

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioInformation(1.3.6.1.4.1.1151.2.1.7.10) （続き）	pfGsxScenarioInformationMinBuff(1.3.6.1.4.1.1151.2.1.7.10.1.1.19)	現在のバッファ使用量が最小のキューについて，該当するキューのバッファ使用量を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationAveBuffRatio(1.3.6.1.4.1.1151.2.1.7.10.1.1.20)	現在のバッファ使用率の平均値を表します。単位は％です。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationAveBuff(1.3.6.1.4.1.1151.2.1.7.10.1.1.21)	現在のバッファ使用量の平均値を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationPeakBuffScenarioId(1.3.6.1.4.1.1151.2.1.7.10.1.1.22)	今までに割り当てたキューの中で，バッファ使用量ピークが最大のキューについて，該当するキューの QID を示します。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationPeakBuffRatio(1.3.6.1.4.1.1151.2.1.7.10.1.1.23)	今までに割り当てたキューの中で，バッファ使用量ピークが最大のキューについて，該当するキューの最大バッファサイズに対するバッファ使用率を表します。単位は％です。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationPeakBuff(1.3.6.1.4.1.1151.2.1.7.10.1.1.24)	今までに割り当てたキューの中で，バッファ使用量ピークが最大のキューについて，該当するキューのバッファ使用量を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInformationDefBuffRatio(1.3.6.1.4.1.1151.2.1.7.10.1.1.25)	シナリオのデフォルトキューの，現在のバッファ使用率を表します。単位は％です。
	pfGsxScenarioInformationDefBuff(1.3.6.1.4.1.1151.2.1.7.10.1.1.26)	シナリオのデフォルトキューの，現在のバッファ使用量を表します。単位はバイトです。
	pfGsxScenarioInformationDefPeakBuffRatio(1.3.6.1.4.1.1151.2.1.7.10.1.1.27)	シナリオのデフォルトキューの，現在のバッファ使用率ピークを表します。単位は％です。
	pfGsxScenarioInformationDefPeakBuff(1.3.6.1.4.1.1151.2.1.7.10.1.1.28)	シナリオのデフォルトキューの，現在のバッファ使用量ピークを表します。単位はバイトです。
	pfGsxScenarioInformationTxPeakRateBps(1.3.6.1.4.1.1151.2.1.7.10.1.1.29)	シナリオの直近 1 分間の送信レートピークを表します。単位は bit/s です。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioInformationTxAveRateBps(1.3.6.1.4.1.1151.2.1.7.10.1.1.31)	シナリオの直近 1 分間の送信レート平均を表します。単位は bit/s です。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioInformationIndQueNum(1.3.6.1.4.1.1151.2.1.7.10.1.1.33)	個別キューモードシナリオの現在の個別キュー数を表します。 個別キューモード以外のシナリオでは 0 固定です。

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioStatByScId(1.3.6.1.4.1.1151.2.1.7.11)	pfGsxScenarioStatByScIdTable(1.3.6.1.4.1.1151.2.1.7.11.1)	シナリオカウンタのテーブルです。 このテーブルには以下のオブジェクトが含まれています。
	pfGsxScenarioStatByScIdEntry(1.3.6.1.4.1.1151.2.1.7.11.1.1)	シナリオカウンタのエントリテーブルです。テーブルインデックスは pfGsxScenarioStatByScIdScenarioId です。 このテーブルには以下のオブジェクトが含まれています。 参考)このテーブル内オブジェクトの OID を求める方法をこの表の次に示します。
	pfGsxScenarioStatByScIdScenarioId(1.3.6.1.4.1.1151.2.1.7.11.1.1.1)	シナリオのシナリオ ID を表します。 シナリオ ID はシナリオ登録時に指定可能です。 シナリオ登録時にシナリオ ID の指定を省略した場合，シナリオ ID は自動割り当てされます。
	pfGsxScenarioStatByScIdScenarioName(1.3.6.1.4.1.1151.2.1.7.11.1.1.2)	シナリオのシナリオ名を表します。
	pfGsxScenarioStatByScIdScenarioType(1.3.6.1.4.1.1151.2.1.7.11.1.1.3)	シナリオのタイプを表します。 discard(0) :廃棄シナリオ individual(1) :個別キューシナリオ aggregate(2) :集約キューシナリオ forward(5) :転送シナリオ
	pfGsxScenarioStatByScIdRxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.4)	シナリオの受信オクテット数を表します。
	pfGsxScenarioStatByScIdRxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.5)	シナリオの受信パケット数を表します。
	pfGsxScenarioStatByScIdTxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.6)	シナリオの送信オクテット数を表します。
	pfGsxScenarioStatByScIdTxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.7)	シナリオの送信パケット数を表します。
	pfGsxScenarioStatByScIdDiscardOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.8)	シナリオの廃棄オクテット数を表します。
	pfGsxScenarioStatByScIdDiscardPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.9)	シナリオの廃棄パケット数を表します。
	pfGsxScenarioStatByScIdHCRxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.10)	シナリオの受信オクテット数を 64 ビットで表します。 注)このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioStatByScId(1.3.6.1.4.1.1151.2.1.7.11) （続き）	pfGsxScenarioStatByScIdHCRxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.11)	シナリオの受信パケット数を 64 ビットで表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdHCTxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.12)	シナリオの送信オクテット数を 64 ビットで表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdHCTxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.13)	シナリオの送信パケット数を 64 ビットで表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdHCDiscardOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.14)	シナリオの廃棄オクテット数を 64 ビットで表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdHCDiscardPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.15)	シナリオの廃棄パケット数を 64 ビットで表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdDefaultQueRxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.16)	シナリオのデフォルトキューの受信オクテット数を表します。
	pfGsxScenarioStatByScIdDefaultQueRxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.17)	シナリオのデフォルトキューの受信パケット数を表します。
	pfGsxScenarioStatByScIdDefaultQueTxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.18)	シナリオのデフォルトキューの送信オクテット数を表します。
	pfGsxScenarioStatByScIdDefaultQueTxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.19)	シナリオのデフォルトキューの送信パケット数を表します。
	pfGsxScenarioStatByScIdDefaultQueDiscardOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.20)	シナリオのデフォルトキューの廃棄オクテット数を表します。
	pfGsxScenarioStatByScIdDefaultQueDiscardPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.21)	シナリオのデフォルトキューの廃棄パケット数を表します。
	pfGsxScenarioStatByScIdDefaultQueHCRxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.22)	シナリオのデフォルトキューの受信オクテット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdDefaultQueHCRxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.23)	シナリオのデフォルトキューの受信パケット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioStatByScId(1.3.6.1.4.1.1151.2.1.7.11) （続き）	pfGsxScenarioStatByScIdDefaultQueHCTxOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.24)	シナリオのデフォルトキューの送信オクテット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdDefaultQueHCTxPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.25)	シナリオのデフォルトキューの送信パケット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdDefaultQueHCDiscardOctets(1.3.6.1.4.1.1151.2.1.7.11.1.1.26)	シナリオのデフォルトキューの廃棄オクテット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioStatByScIdDefaultQueHCDiscardPackets(1.3.6.1.4.1.1151.2.1.7.11.1.1.27)	シナリオのデフォルトキューの廃棄パケット数を表します。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
pfGsxScenarioInfoByScId(1.3.6.1.4.1.1151.2.1.7.12)	pfGsxScenarioInfoByScIdTable(1.3.6.1.4.1.1151.2.1.7.12.1)	シナリオ情報のテーブルです。 このテーブルには以下のオブジェクトが含まれています。
	pfGsxScenarioInfoByScIdEntry(1.3.6.1.4.1.1151.2.1.7.12.1.1)	シナリオ情報のエントリテーブルです。テーブルインデックスは pfGsxScenarioInfoByScIdScenarioId です。 このテーブルには以下のオブジェクトが含まれています。 参考）このテーブル内オブジェクトの OID を求める方法をこの表の次に示します。
	pfGsxScenarioInfoByScIdScenarioId(1.3.6.1.4.1.1151.2.1.7.12.1.1.1)	シナリオのシナリオ ID を表します。 シナリオ ID はシナリオ登録時に指定可能です。 シナリオ登録時にシナリオ ID の指定を省略した場合，シナリオ ID は自動割り当てされます。
	pfGsxScenarioInfoByScIdScenarioName(1.3.6.1.4.1.1151.2.1.7.12.1.1.2)	シナリオのシナリオ名を表します。
	pfGsxScenarioInfoByScIdScenarioType(1.3.6.1.4.1.1151.2.1.7.12.1.1.3)	シナリオのタイプを表します。 discard(0)　：廃棄シナリオ individual(1)　：個別キューシナリオ aggregate(2)　：集約キューシナリオ forward(5)　：転送シナリオ
	pfGsxScenarioInfoByScIdDefFlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.4)	シナリオに関連して生成されたデフォルトフローの数を表します。forward シナリオの場合はフローの総数と同じ値になります。
	pfGsxScenarioInfoByScIdClass1FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.5)	シナリオに関連して生成されたクラス 1 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInfoByScIdClass2FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.6)	シナリオに関連して生成されたクラス 2 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInfoByScIdClass3FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.7)	シナリオに関連して生成されたクラス 3 フローの数を表します。 注）未サポートです。値は 0 固定です。

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioInfoByScId(1.3.6.1.4.1.1151.2.1.7.12)（続き）	pfGsxScenarioInfoByScIdClass4FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.8)	シナリオに関連して生成されたクラス 4 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInfoByScIdClass5FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.9)	シナリオに関連して生成されたクラス 5 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInfoByScIdClass6FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.10)	シナリオに関連して生成されたクラス 6 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInfoByScIdClass7FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.11)	シナリオに関連して生成されたクラス 7 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInfoByScIdClass8FlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.12)	シナリオに関連して生成されたクラス 8 フローの数を表します。 注）未サポートです。値は 0 固定です。
	pfGsxScenarioInfoByScIdTotalFlowNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.13)	シナリオに関連して生成されたフローの総数を表します。
	pfGsxScenarioInfoByScIdMaxBuffScenarioId(1.3.6.1.4.1.1151.2.1.7.12.1.1.14)	現在のバッファ使用量が最大のキューについて，該当するキューの QID を示します。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdMaxBuffRatio(1.3.6.1.4.1.1151.2.1.7.12.1.1.15)	現在のバッファ使用量が最大のキューについて，該当するキューの最大バッファサイズに対するバッファ使用率を表します。単位は%です。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdMaxBuff(1.3.6.1.4.1.1151.2.1.7.12.1.1.16)	現在のバッファ使用量が最大のキューについて，該当するキューのバッファ使用量を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdMinBuffScenarioId(1.3.6.1.4.1.1151.2.1.7.12.1.1.17)	現在のバッファ使用量が最小のキューについて，該当するキューの QID を示します。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdMinBuffRatio(1.3.6.1.4.1.1151.2.1.7.12.1.1.18)	現在のバッファ使用量が最小のキューについて，該当するキューの最大バッファサイズに対するバッファ使用率を表します。単位は%です。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdMinBuff(1.3.6.1.4.1.1151.2.1.7.12.1.1.19)	現在のバッファ使用量が最小のキューについて，該当するキューのバッファ使用量を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdAveBuffRatio(1.3.6.1.4.1.1151.2.1.7.12.1.1.20)	現在のバッファ使用率の平均値を表します。単位は%です。 個別キューモード以外のシナリオでは 0 固定です。

MIB グループ	MIB オブジェクト名	説明
pfGsxScenarioInfoByScId(1.3.6.1.4.1.1151.2.1.7.12)（続き）	pfGsxScenarioInfoByScIdAveBuff(1.3.6.1.4.1.1151.2.1.7.12.1.1.21)	現在のバッファ使用量の平均値を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdPeakBuffScenarioId(1.3.6.1.4.1.1151.2.1.7.12.1.1.22)	今までに割り当てたキューの中で，バッファ使用量ピークが最大のキューについて，該当するキューの QID を示します。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdPeakBuffRatio(1.3.6.1.4.1.1151.2.1.7.12.1.1.23)	今までに割り当てたキューの中で，バッファ使用量ピークが最大のキューについて，該当するキューの最大バッファサイズに対するバッファ使用率を表します。単位は%です。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdPeakBuff(1.3.6.1.4.1.1151.2.1.7.12.1.1.24)	今までに割り当てたキューの中で，バッファ使用量ピークが最大のキューについて，該当するキューのバッファ使用量を表します。単位はバイトです。 個別キューモード以外のシナリオでは 0 固定です。
	pfGsxScenarioInfoByScIdDefBuffRatio(1.3.6.1.4.1.1151.2.1.7.12.1.1.25)	シナリオのデフォルトキューの，現在のバッファ使用率を表します。単位は%です。
	pfGsxScenarioInfoByScIdDefBuff(1.3.6.1.4.1.1151.2.1.7.12.1.1.26)	シナリオのデフォルトキューの，現在のバッファ使用量を表します。単位はバイトです。
	pfGsxScenarioInfoByScIdDefPeakBuffRatio(1.3.6.1.4.1.1151.2.1.7.12.1.1.27)	シナリオのデフォルトキューの，現在のバッファ使用率ピークを表します。単位は%です。
	pfGsxScenarioInfoByScIdDefPeakBuff(1.3.6.1.4.1.1151.2.1.7.12.1.1.28)	シナリオのデフォルトキューの，現在のバッファ使用量ピークを表します。単位はバイトです。
	pfGsxScenarioInfoByScIdTxPeakRateBps(1.3.6.1.4.1.1151.2.1.7.12.1.1.29)	シナリオの直近 1 分間の送信レートピークを表します。単位は bit/s です。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioInfoByScIdTxAveRateBps(1.3.6.1.4.1.1151.2.1.7.12.1.1.31)	シナリオの直近 1 分間の送信レート平均を表します。単位は bit/s です。 注）このオブジェクトに SNMPv1 でアクセスすることはできません。v2c 以上でアクセスしてください。
	pfGsxScenarioInfoByScIdIndQueNum(1.3.6.1.4.1.1151.2.1.7.12.1.1.33)	個別キューモードシナリオの現在の個別キュー数を表します。 個別キューモード以外のシナリオでは 0 固定です。

参考）

シナリオカウンタ，シナリオインフォメーションテーブルの OID を求める方法

テーブル内オブジェクトの OID を求めるには以下を参考にしてください。

pfGsxScenarioStatisticsTable の場合
pfGsxScenarioStatisticsEntry の OID は次のようになります。
1.3.6.1.4.1.1151.2.1.7.9.1.1.EntryOID.ScenarioSortIndex
固定値

EntryOID ：テーブル内エントリの番号です。表 4 の順序通りに 1 から並んでいます。長さは 1 です。

pfGsxScenarioStatisticsScenarioSortIndex	1
pfGsxScenarioStatisticsScenarioName	2
pfGsxScenarioStatisticsScenarioType	3
pfGsxScenarioStatisticsRxOctets	4
pfGsxScenarioStatisticsRxPackets	5
pfGsxScenarioStatisticsTxOctets	6
pfGsxScenarioStatisticsTxPackets	7
pfGsxScenarioStatisticsDiscardOctets	8
pfGsxScenarioStatisticsDiscardPackets	9
pfGsxScenarioStatisticsHCRxOctets	10
pfGsxScenarioStatisticsHCRxPackets	11
pfGsxScenarioStatisticsHCTxOctets	12
pfGsxScenarioStatisticsHCTxPackets	13
pfGsxScenarioStatisticsHCDiscardOctets	14
pfGsxScenarioStatisticsHCDiscardPackets	15
pfGsxScenarioStatisticsDefaultQueRxOctets	16
pfGsxScenarioStatisticsDefaultQueRxPackets	17
pfGsxScenarioStatisticsDefaultQueTxOctets	18
pfGsxScenarioStatisticsDefaultQueTxPackets	19
pfGsxScenarioStatisticsDefaultQueDiscardOctets	20
pfGsxScenarioStatisticsDefaultQueDiscardPackets	21
pfGsxScenarioStatisticsDefaultQueHCRxOctets	22
pfGsxScenarioStatisticsDefaultQueHCRxPackets	23
pfGsxScenarioStatisticsDefaultQueHCTxOctets	24
pfGsxScenarioStatisticsDefaultQueHCTxPackets	25
pfGsxScenarioStatisticsDefaultQueHCDiscardOctets	26
pfGsxScenarioStatisticsDefaultQueHCDiscardPackets	27

ScenarioSortIndex ：シナリオのソート番号です。長さは 16 です。ソート番号はシナリオツリーの並び順に対応した番号を表し，シナリオ登録／削除時に自動割り当てされます。シナリオ登録／削除の度に再割り当てされますので，シナリオ構成を変えるとソート番号も変化します。特定シナリオのソート番号を求めるには，シナリオ構成が決定された状態で pfGsxScenarioStatisticsTable を get next で全取得し，シナリオ名をキーにして求めるエントリを探してください。

pfGsxScenarioInformationTable の場合
pfGsxScenarioInformationEntry の OID は次のようになります。
1.3.6.1.4.1.1151.2.1.7.10.1.1.EntryOID.ScenarioSortIndex
固定値

EntryOID ： テーブル内エントリの番号です。番号は連続していませんので注意してください。長さは 1 です。

pfGsxScenarioInformationScenarioSortIndex	1
pfGsxScenarioInformationScenarioName	2
pfGsxScenarioInformationScenarioType	3
pfGsxScenarioInformationDefFlowNum	4
pfGsxScenarioInformationMaxBuffScenarioID	14
pfGsxScenarioInformationMaxBuffRaito	15
pfGsxScenarioInformationMaxBuff	16
pfGsxScenarioInformationMinBuffScenarioID	17
pfGsxScenarioInformationMinBuffRaito	18
pfGsxScenarioInformationMinBuff	19
pfGsxScenarioInformationAveBuffRatio	20
pfGsxScenarioInformationAveBuff	21
pfGsxScenarioInformationPeakScenarioID	22
pfGsxScenarioInformationPeakBuffRatio	23
pfGsxScenarioInformationPeakBuff	24
pfGsxScenarioInformationDefBuffRatio	25
pfGsxScenarioInformationDefBuff	26
pfGsxScenarioInformationDefPeakBuffRatio	27
pfGsxScenarioInformationDefPeakBuff	28
pfGsxScenarioInformationTxPeakRateBps	29
pfGsxScenarioInformationTxAveRateBps	31
pfGsxScenarioInformationIndQueNum	33

ScenarioSortIndex ： シナリオのソート番号です。長さは 16 です。求め方は
pfGsxScenarioStatisticsTable のソート番号と同様です。

pfGsxScenarioStatByScIdTable の場合
pfGsxScenarioStatByScIdEntry の OID は次のようになります。
1.3.6.1.4.1.1151.2.1.7.11.1.1.EntryOID.ScenarioId
固定値

EntryOID ：テーブル内エントリの番号です。表 4 の順序通りに 1 から並んでいます。長さは 1 です。

pfGsxScenarioStatByScIdScenarioId	1
pfGsxScenarioStatByScIdScenarioName	2
pfGsxScenarioStatByScIdScenarioType	3
pfGsxScenarioStatByScIdRxOctets	4
pfGsxScenarioStatByScIdRxPackets	5
pfGsxScenarioStatByScIdTxOctets	6
pfGsxScenarioStatByScIdTxPackets	7
pfGsxScenarioStatByScIdDiscardOctets	8
pfGsxScenarioStatByScIdDiscardPackets	9
pfGsxScenarioStatByScIdHCRxOctets	10
pfGsxScenarioStatByScIdHCRxPackets	11
pfGsxScenarioStatByScIdHCTxOctets	12
pfGsxScenarioStatByScIdHCTxPackets	13
pfGsxScenarioStatByScIdHCDiscardOctets	14
pfGsxScenarioStatByScIdHCDiscardPackets	15
pfGsxScenarioStatByScIdDefaultQueRxOctets	16
pfGsxScenarioStatByScIdDefaultQueRxPackets	17
pfGsxScenarioStatByScIdDefaultQueTxOctets	18
pfGsxScenarioStatByScIdDefaultQueTxPackets	19
pfGsxScenarioStatByScIdDefaultQueDiscardOctets	20
pfGsxScenarioStatByScIdDefaultQueDiscardPackets	21
pfGsxScenarioStatByScIdDefaultQueHCRxOctets	22
pfGsxScenarioStatByScIdDefaultQueHCRxPackets	23
pfGsxScenarioStatByScIdDefaultQueHCTxOctets	24
pfGsxScenarioStatByScIdDefaultQueHCTxPackets	25
pfGsxScenarioStatByScIdDefaultQueHCDiscardOctets	26
pfGsxScenarioStatByScIdDefaultQueHCDiscardPackets	27

ScenarioId ： シナリオのシナリオ ID です。長さは 1 です。シナリオ登録時に指定したシナリオ ID です。
シナリオ登録時にシナリオ ID の指定を省略した場合，シナリオ ID は自動割り当てされます。この場合，show scenario name コマンドで割り当てられたシナリオ ID を確認してください。
ポートシナリオのシナリオ ID はポート 1 では 40001，ポート 2 では 40002 が割り当てられます。

pfGsxScenarioInfoByScIdTable の場合
pfGsxScenarioInfoByScIdEntry の OID は次のようになります。
1.3.6.1.4.1.1151.2.1.7.12.1.1.EntryOID.ScenarioId
固定値

EntryOID ： テーブル内エントリの番号です。番号は連続していませんので注意してください。長さは 1 です。

pfGsxScenarioInfoByScIdScenarioId	1
pfGsxScenarioInfoByScIdScenarioName	2
pfGsxScenarioInfoByScIdScenarioType	3
pfGsxScenarioInfoByScIdDefFlowNum	4
pfGsxScenarioInfoByScIdMaxBuffScenarioID	14
pfGsxScenarioInfoByScIdMaxBuffRatio	15
pfGsxScenarioInfoByScIdMaxBuff	16
pfGsxScenarioInfoByScIdMinBuffScenarioID	17
pfGsxScenarioInfoByScIdMinBuffRatio	18
pfGsxScenarioInfoByScIdMinBuff	19
pfGsxScenarioInfoByScIdAveBuffRatio	20
pfGsxScenarioInfoByScIdAveBuff	21
pfGsxScenarioInfoByScIdPeakBuffScenarioID	22
pfGsxScenarioInfoByScIdPeakBuffRatio	23
pfGsxScenarioInfoByScIdPeakBuff	24
pfGsxScenarioInfoByScIdDefBuffRatio	25
pfGsxScenarioInfoByScIdDefBuff	26
pfGsxScenarioInfoByScIdDefPeakBuffRatio	27
pfGsxScenarioInfoByScIdDefPeakBuff	28
pfGsxScenarioInfoByScIdTxPeakRateBps	29
pfGsxScenarioInfoByScIdTxAveRateBps	31
pfGsxScenarioInfoByScIdIndQueNum	33

ScenarioSortIndex ： シナリオのシナリオ ID です。長さは 1 です。求め方は
pfGsxScenarioStatByScIdTable のシナリオ ID と同様です。

付録E　JSON の記述方法

JSON（JavaScript Object Notation:RFC4627）による記述方法を示します。

JSON は RFC4627 に規定されるテキストベースの簡易なデータ記述言語です。
JSON には 4 つの型と 2 つの構造体があります。本装置の WebAPI では string 型と object 構造体のみを使用します。

	種　別	記述例	説　明
型	string	"PureFlow"	文字列
	number	123	数値
	boolean	true	真（true）または偽（false）を示します。
	null	null	値なしを示します。
構造体	object	{name:value}	0 個以上の名前と値のペアを並べたものです。
	array	[value, value]	0 個以上の値を並べたものです。

以下，「付録 F　WebAPI 詳細」の下記シナリオ追加 API を例にして記述方法を示します。

API	キー	値	相当する CLI コマンドとパラメータ
シナリオ追加（Discard）	"command"（必須）	"add scenario"	add scenario
	"scenario_name"（必須）	シナリオ名	scenario <scenario_name>
	"action"（必須）	"discard"	action discard
	"scenario_id"（省略可）	シナリオ ID	[scenario <scenario_id>]

キーと値を":"コロンでペアにします。

```
"command":"add scenario"
"scenario_name":"/port1/North"
"action":"discard"
"scenario_id":"1"
```

シナリオ ID の指定が不要な場合は省略できます。

```
"command":"add scenario"
"scenario_name":"/port1/North"
"action":"discard"
```

これら 3 つのパラメータを", "カンマでつなげます。最後のパラメータにはカンマを加えません。

```
"command":"add scenario", "scenario_name":"/port1/North", "action":"discard"
```

最後に波括弧"{"と"}"で囲って object 構造体にします。

```
{"command":"add scenario", "scenario_name":"/port1/North", "action":"discard"}
```

記述を見やすくするために, 波括弧, コロン, カンマの前後には半角スペース, Tab, 改行を加えることができます。

```
{
      "command" : "add scenario",
      "scenario_name" : "/port1/North",
      "action" : "discard"
}
```

なお, 本装置の WebAPI ではパラメータの順序は順不同です。「付録 F　WebAPI 詳細」の順序に合わせる必要はありません。

```
{
      "action" : "discard",
      "scenario_name" : "/port1/North",
      "command" : "add scenario"
}
```

付録F　WebAPI 詳細

PureFlow GSX の WebAPI 詳細を示します。

WebAPI では以下の URL に対して JSON データを与えます。

http://システムインタフェースの IP アドレス/shapermng/json

HTTPS（Hypertext Transfer Secure）を利用する場合は，URL の先頭を"https"にしてください。

https://システムインタフェースの IP アドレス/shapermng/json

キーと値はすべて文字列で指定します。省略可能なパラメータは指定が不要な場合は記述不要です。キーにスペルミスがある場合，そのパラメータは無視されます。指定必須パラメータのスペルミスはエラーとなりますが，省略可能なパラメータのスペルミスや，未定義のパラメータはエラーとならないことに注意してください。

指定する値の詳細は「PureFlow GSX トラフィックシェーパー NF7101C コマンドリファレンス」を参照してください。

API	キー	値	相当する CLI コマンドとパラメータ
シナリオ追加（Discard）	"command" （必須）	"add scenario"	add scenario
	"scenario_name" （必須）	シナリオ名	scenario <scenario_name>
	"action" （必須）	"discard"	action discard
	"scenario_id" （省略可）	シナリオ ID	[scenario <scenario_id>]
シナリオ追加（Aggregate）	"command" （必須）	"add scenario"	add scenario
	"scenario_name" （必須）	シナリオ名	scenario <scenario_name>
	"action" （必須）	"aggregate"	action aggregate
	"min_bandwidth" （省略可）	最小帯域	[min_bw <min_bandwidth>]
	"peak_bandwidth" （省略可）	最大帯域	[peak_bw <peak_bandwidth>]
	"class " （省略可）	クラス	[class <class>]
	"bufsize" （省略可）	バッファサイズ	[bufsize <bufsize>]
	"scenario_id" （省略可）	シナリオ ID	[scenario <scenario_id>]

API	キー	値	相当する CLI コマンドと パラメータ
シナリオ追加 (Individual)	"command" (必須)	"add scenario"	add scenario
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"action" (必須)	"individual"	action individual
	"min_bandwidth" (省略可)	最小帯域	[min_bw <min_bandwidth>]
	"peak_bandwidth" (省略可)	最大帯域	[peak_bw <peak_bandwidth>]
	"class " (省略可)	クラス	[class <class>]
	"bufsize" (省略可)	バッファサイズ	[bufsize <bufsize>]
	"scenario_id" (省略可)	シナリオ ID	[scenario <scenario_id>]
	"maxqnum" (省略可)	個別キュー最大数	[maxquenum <quenum>]
	"quedivision" (省略可)	個別キュー分割対象	[quedivision <field>]
	"failaction" (省略可)	個別キュー数超過時の動作	[failaction <discard | forwardbesteffort | forwardattribute>]
	"fail_min_bw" (省略可)	個別キュー数超過時の最小帯域	[fail_min_bw <min_bandwidth>]
	"fail_peak_bw" (省略可)	個別キュー数超過時の最大帯域	[fail_peak_bw <peak_bandwidth>]
	"fail_class" (省略可)	個別キュー数超過時のクラス	[fail_class <class>]
シナリオ更新 (Aggregate)	"command" (必須)	"update scenario"	update scenario
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"action" (必須)	"aggregate"	action aggregate
	"min_bandwidth" (省略可)	最小帯域	[min_bw <min_bandwidth>]
	"peak_bandwidth" (省略可)	最大帯域	[peak_bw <peak_bandwidth>]
	"class " (省略可)	クラス	[class <class>]
	"bufsize" (省略可)	バッファサイズ	[bufsize <bufsize>]

API	キー	値	相当する CLI コマンドとパラメータ
シナリオ更新（Individual）	"command" （必須）	"update scenario"	update scenario
	"scenario_name" （必須）	シナリオ名	scenario <scenario_name>
	"action" （必須）	"individual"	action individual
	"min_bandwidth" （省略可）	最小帯域	[min_bw <min_bandwidth>]
	"peak_bandwidth" （省略可）	最大帯域	[peak_bw <peak_bandwidth>]
	"class " （省略可）	クラス	[class <class>]
	"bufsize" （省略可）	バッファサイズ	[bufsize <bufsize>]
	"maxqnum" （省略可）	個別キュー最大数	[maxquenum <quenum>]
	"quedivision" （省略可）	個別キュー分割対象	[quedivision <field>]
	"failaction" （省略可）	個別キュー数超過時の動作	[failaction <discard | forwardbesteffort | forwardattribute>]
	"fail_min_bw" （省略可）	個別キュー数超過時の最小帯域	[fail_min_bw <min_bandwidth>]
	"fail_peak_bw" （省略可）	個別キュー数超過時の最大帯域	[fail_peak_bw <peak_bandwidth>]
	"fail_class" （省略可）	個別キュー数超過時のクラス	[fail_class <class>]
シナリオ削除（全指定）	"command" （必須）	"delete scenario"	delete scenario
	"scenario_name" （必須）	"all" すべての登録済フィルタも削除します。	All
シナリオ削除（シナリオ指定）	"command" （必須）	"delete scenario"	delete scenario
	"scenario_name" （必須）	シナリオ名 対象シナリオに登録済のフィルタも削除します。	scenario <scenario_name>
	"recursive" （省略可）	"recursive"	[recursive]

API	キー	値	相当する CLI コマンドと パラメータ
シナリオ情報 取得	"command" （必須）	"show scenario"	show scenario
	"scenario_name" （必須）	シナリオ名	name <scenario_name>
	"search_type" （省略可）	取得方法 "exact"：指定したシナリオの情報を取得します。 "next"：指定したシナリオの次のシナリオ情報を取得します。 省略時・値のスペルミス時は"exact"を適用します。	なし

シナリオ情報取得 API について

シナリオ情報取得 API では取得方法を指定する"search_type"パラメータがあります。"search_type"には値として"exact"か"next"を指定します。

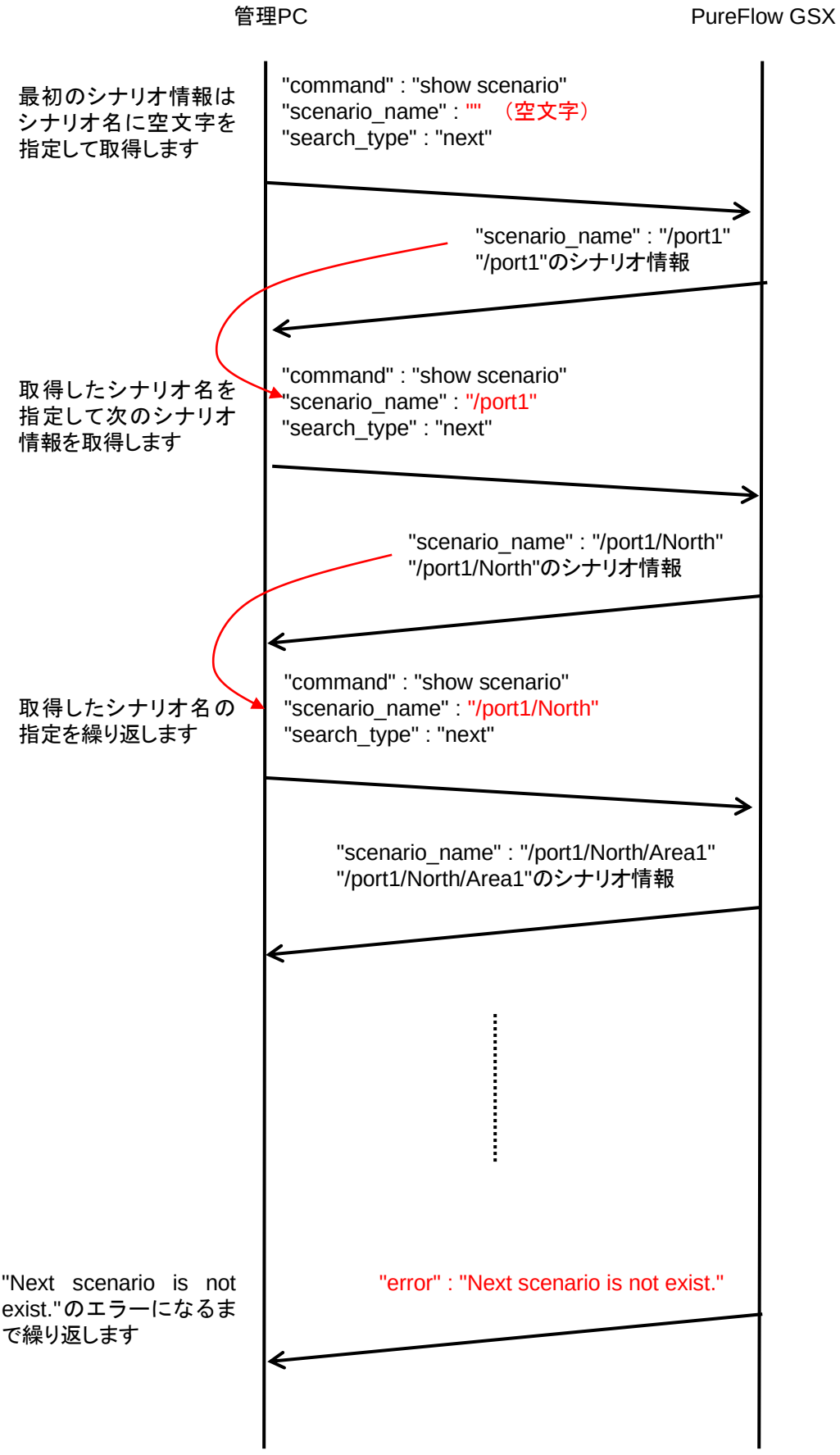
"exact"　"scenario_name"で指定したシナリオの情報を取得します。
"next"　"scenario_name"で指定したシナリオの次のシナリオの情報を取得します。
取得する順序は"show scenario"CLI コマンドと同様にシナリオツリー順です。

"search_type"を省略した場合は，"exact"を適用します。
特定のシナリオ情報を取得したい場合は，そのシナリオ名を指定して"exact"で取得してください。
CLI コマンドの"show scenario all"のようにすべてのシナリオ情報を取得したい場合は，"next"を使用して下記の手順で取得してください。

最初のシナリオ情報取得は"scenario_name"に空文字を指定します。
"scenario_name"："" （空文字）
"search_type"："next"

シナリオツリーの先頭のシナリオ"/port1"の情報を取得できます。
続いて，"scenario_name"に取得したシナリオ名を指定します。
"scenario_name"："/port1"
"search_type"："next"

シナリオツリーで"/port1"の次に位置するシナリオの情報を取得できます。
このように，取得したシナリオ名を指定して"next"による取得を繰り返します。シナリオツリーの最後尾を指定して"next"による取得を行うと"Next scenario is not exist."のエラーになります。


管理PC
PureFlow GSX
最初のシナリオ情報はシナリオ名に空文字を指定して取得します
"command" : "show scenario"
"scenario_name" : "" （空文字）
"search_type" : "next"
"scenario_name" : "/port1"
"/port1"のシナリオ情報
取得したシナリオ名を指定して次のシナリオ情報を取得します
"command" : "show scenario"
"scenario_name" : "/port1"
"search_type" : "next"
"scenario_name" : "/port1/North"
"/port1/North"のシナリオ情報
取得したシナリオ名の指定を繰り返します
"command" : "show scenario"
"scenario_name" : "/port1/North"
"search_type" : "next"
"scenario_name" : "/port1/North/Area1"
"/port1/North/Area1"のシナリオ情報
"Next scenario is not exist."のエラーになるまで繰り返します
"error" : "Next scenario is not exist."

API	キー	値	相当する CLI コマンドと パラメータ
フィルタ追加 (Bridge-ctrl)	"command" (必須)	"add filter"	add filter
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"filter_name" (必須)	フィルタ名	filter <scenario_name>
	"type" (必須)	"bridge-ctrl"	bridge-ctrl
	"priority" (省略可)	フィルタ優先度	[priority <filter_pri>]
フィルタ追加 (Ethernet)	"command" (必須)	"add filter"	add filter
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"filter_name" (必須)	フィルタ名	filter <scenario_name>
	"type" (必須)	"ethernet"	ethernet
	"vid" (省略可)	VLAN ID	[vid {<VID> | none}]
	"cos" (省略可)	CoS	[cos <user_priority>]
	"inner-vid" (省略可)	Inner-VLAN ID	[inner-vid {<VID> | none}]
	"inner-cos" (省略可)	Inner-CoS	[inner-cos <user_priority>]
	"ethertype" (省略可)	Ethernet Type/Length	[ethertype <type>]
	"priority" (省略可)	フィルタ優先度	[priority <filter_pri>]
フィルタ追加 (IPv4)	"command" (必須)	"add filter"	add filter
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"filter_name" (必須)	フィルタ名	filter <scenario_name>
	"type" (必須)	"ipv4"	ipv4
	"vid" (省略可)	VLAN ID	[vid {<VID> | none}]
	"cos" (省略可)	CoS	[cos <user_priority>]
	"inner-vid" (省略可)	Inner-VLAN ID	[inner-vid {<VID> | none}]

API	キー	値	相当する CLI コマンドと パラメータ
フィルタ追加 (IPv4) (続き)	"inner-cos" (省略可)	Inner-CoS	[inner-cos <user_priority>]
	"sip" または "sip list" (省略可)	送信元 IPv4 アドレス または ルールリスト名	[sip [list] {<src_IP_address> | <list_name>}]
	"dip" または "dip list" (省略可)	宛先 IPv4 アドレス または ルールリスト名	[dip [list] {<dst_IP_address> | <list_name>}]
	"tos" (省略可)	ToS	[tos <type_of_service>]
	"proto" (省略可)	プロトコル番号	[proto <protocol>]
	"sport" または "sport list" (省略可)	送信元ポート番号 または ルールリスト名	[sport [list] {<sport> | <list_name>}]
	"dport" または "dport list" (省略可)	宛先ポート番号 または ルールリスト名	[dport [list] {<dport> | <list_name>}]
	"priority" (省略可)	フィルタ優先度	[priority <filter_pri>]
フィルタ追加 (IPv6)	"command" (必須)	"add filter"	add filter
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"filter_name" (必須)	フィルタ名	filter <scenario_name>
	"type" (必須)	"ipv6"	ipv6
	"vid" (省略可)	VLAN ID	[vid {<VID> | none}]
	"cos" (省略可)	CoS	[cos <user_priority>]
	"inner-vid" (省略可)	Inner-VLAN ID	[inner-vid {<VID> | none}]
	"inner-cos" (省略可)	Inner-CoS	[inner-cos <user_priority>]
	"sip" または "sip list" (省略可)	送信元 IPv6 アドレス または ルールリスト名	[sip [list] {<src_IP_address> | <list_name>}]
	"dip" または "dip list" (省略可)	宛先 IPv6 アドレス または ルールリスト名	[dip [list] {<dst_IP_address> | <list_name>}]
	"tos" (省略可)	ToS	[tos <type_of_service>]

API	キー	値	相当する CLI コマンドと パラメータ
フィルタ追加 (IPv6) (続き)	"proto" (省略可)	プロトコル番号	[proto <protocol>]
	"sport" または "sport list" (省略可)	送信元ポート番号 または ルールリスト名	[sport [list] {<sport> | <list_name>}]
	"dport" または "dport list" (省略可)	宛先ポート番号 または ルールリスト名	[dport [list] {<dport> | <list_name>}]
	"priority" (省略可)	フィルタ優先度	[priority <filter_pri>]
フィルタ削除 (全指定)	"command" (必須)	"delete filter"	delete filter
	"scenario_name" (必須)	"all"	all
フィルタ削除 (シナリオ指定)	"command" (必須)	"delete filter"	delete filter
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
フィルタ削除 (フィルタ指定)	"command" (必須)	"delete filter"	delete filter
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"filter_name" (必須)	フィルタ名	filter <scenario_name>
フィルタ情報取得	"command" (必須)	"show filter"	show filter
	"scenario_name" (必須)	シナリオ名	scenario <scenario_name>
	"filter_name" (必須)	フィルタ名	filter <scenario_name>
	"search_type" (省略可)	取得方法 "exact":指定したフィルタの情報を取得します。 "next": 指定したフィルタの次のフィルタ情報を取得します。 省略時・値のスペルミス時は"exact"を適用します。	なし

フィルタ情報取得 API について

フィルタ情報取得 API では取得方法を指定する"search_type"パラメータがあります。"search_type"には値として"exact"か"next"を指定します。

"exact"　"scenario_name"および"filter_name"で指定したフィルタの情報を取得します。
"next"　"scenario_name"および"filter_name"で指定したフィルタの次のフィルタの情報を取得します。取得する順序は"show filter" CLI コマンドと同様にフィルタ名のアルファベット順です。シナリオの最後尾のフィルタを指定した場合は，次のシナリオの先頭のフィルタ情報を取得します。

特定のフィルタ情報を取得したい場合は，そのシナリオ名およびフィルタ名を指定して"exact"で取得してください。
CLI コマンドの"show filter all"のようにすべてのシナリオのすべてのフィルタ情報を取得したい場合は，"next"を使用します。"next"での取得手順はシナリオ取得 API と同様です。

API	キー	値	相当する CLI コマンドとパラメータ
ルールリストグループ追加	"command" （必須）	"add rulelist group"	add rulelist group
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	ルールリスト種別	{ipv4 | ipv6 | l4port}
ルールリストグループ削除（全指定）	"command" （必須）	"delete rulelist group"	delete rulelist group
	"list_name" （必須）	"all" すべてのルールリストエントリも削除します。	all
ルールリストグループ削除（グループ指定）	"command" （必須）	"delete rulelist group"	delete rulelist group
	"list_name" （必須）	ルールリスト名 対象ルールリストに登録済のルールリストエントリも削除します。	<list_name>
ルールリストエントリ追加（IPv4）	"command" （必須）	"add rulelist entry"	add rulelist entry
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	"ipv4"	ipv4
	"IP_address" （必須）	IPv4 アドレス	<IP_address>

API	キー	値	相当する CLI コマンドと パラメータ
ルールリストエントリ追加（IPv6）	"command" （必須）	"add rulelist entry"	add rulelist entry
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	"ipv6"	ipv6
	"IP_address" （必須）	IPv6 アドレス	<IP_address>
ルールリストエントリ追加（L4Port）	"command" （必須）	"add rulelist entry"	add rulelist entry
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	"l4port"	l4port
	"port" （必須）	L4 ポート番号	<port>
ルールリストエントリ削除（全指定）	"command" （必須）	"delete rulelist entry"	delete rulelist entry
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	"all"	all
ルールリストエントリ削除（IPv4）	"command" （必須）	"delete rulelist entry"	delete rulelist entry
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	"ipv4"	ipv4
	"IP_address" （必須）	IPv4 アドレス	<IP_address>
ルールリストエントリ削除（IPv6）	"command" （必須）	"delete rulelist entry"	delete rulelist entry
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	"ipv6"	ipv6
	"IP_address" （必須）	IPv6 アドレス	<IP_address>

API	キー	値	相当する CLI コマンドと パラメータ
ルールリストエントリ削除（L4Port）	"command" （必須）	"delete rulelist entry"	delete rulelist entry
	"list_name" （必須）	ルールリスト名	<list_name>
	"type" （必須）	"l4port"	l4port
	"port" （必須）	L4 ポート番号	<port>
ルールリスト情報取得	"command" （必須）	"show rulelist"	show rulelist
	"list_name" （必須）	ルールリスト名	[<list_name>]
	"rules" （必須）	ルールリストエントリ	なし
	"search_type" （省略可）	取得方法 "exact"：指定したルールリストエントリを取得します。 "next"：指定したルールリストエントリの次のルールリストエントリを取得します。 省略時・値のスペルミス時は"exact"を適用します。	なし

ルールリスト情報取得 API について

ルールリスト情報取得 API では"show rulelist" CLI コマンドにはない"rules"パラメータがあります。

"rules"には値としてルールリストエントリ（IP アドレスまたは L4 ポート番号）を指定します。ルールリストエントリは単一の値であっても常にハイフンを使用した範囲指定で指定してください。

IPv4 アドレス　　192.168.1.1-192.168.1.1
IPv6 アドレス　　FE80::0001-FE80::0001
L4 ポート番号　　1000-1000

なお，ルールリストエントリが設定されていないルールリストでは"none"が取得されます。

取得方法を指定する"search_type"には値として"exact"か"next"を指定します。

"exact"　"list_name"および"rules"で指定したルールリストエントリを取得します。
"next"　"list_name"および"rules"で指定したルールリストエントリの次のルールリストエントリを取得します。取得する順序は"show rulelist" CLI コマンドと同様です。ルールリストの最後尾のルールリストエントリを指定した場合は，次のルールリストの先頭のルールリストエントリを取得します。

特定のルールリストエントリを取得したい場合は，そのルールリスト名およびルールリストエントリを指定して"exact"で取得してください。
CLI コマンドの"show rulelist all"のようにすべてのルールリストのすべてのルールリストエントリを取得したい場合は"next"を使用します。"next"での取得手順はシナリオ取得 API と同様です。

API	キー	値	相当する CLI コマンドとパラメータ
コンフィギュレーション保存	"command" （必須）	"save config"	save config
コンフィギュレーション保存実行状態取得	"command" （必須）	"show save status"	なし

コンフィギュレーション保存 API について

コンフィギュレーション保存 API は保存の完了を待たずに終了します。コンフィギュレーション保存はバックグラウンドで実行されます。保存が実行されている最中にさらに本 API でコンフィギュレーションの保存を指示した場合，"configuration save is in progress"のエラーメッセージを返します。コンフィギュレーション保存の所要時間については「第 3 章　設定の基本」を参照してください。

コンフィギュレーション保存実行状態取得 API について

相当する CLI コマンドはありません。本 API はコンフィギュレーション保存の実行状態を取得します。

"configuration save is in progress"　　:コンフィギュレーション保存が実行中です。
"configuration save is not in progress"　:コンフィギュレーション保存は完了しています。

付録G　WebAPI サンプルプログラム

WebAPI で広く使用されているプログラミング言語に Python があります。Python には標準で HTTP および JSON のライブラリが含まれており，PureFlow GSX の WebAPI 利用にも適しています。

付録 F の各 WebAPI について Python バージョン 2.7.2 によるサンプルプログラムを示します。

設定系

設定の追加で add 系を，設定の更新で update 系を，設定の削除で delete 系の API を使用します。
add，update，および delete 系の API では，コマンドとパラメータを送信してレスポンスを確認する手順になります。いずれの API においても同様ですので，"add scenario"の例を示します。

①　単一の設定を行うサンプルプログラム。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義  PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義  PureFlowGSX WebAPI の URL HTTPS
#url       = 'https://192.168.1.1/shapermng/json'

#  パラメータ定義
params = {
                'command': 'add scenario',
                'scenario_name' : '/port1/North',
                'action' : 'aggregate',
                'min_bandwidth' : '100M',
                'peak_bandwidth' : '1G',
                'bufsize' : '1M'
                }
json_data = json.dumps(params)

# POST リクエスト
response = urllib2.urlopen(url, json_data)

#  レスポンスの表示
print 'RESPONSE:', response
print 'URL         :', response.geturl()

data = response.read()
print 'LENGTH    :', len(data)
print 'DATA       :'
print '----------'
print data
print
```

Python の urlopen は，HTTP リクエスト完了時に終了し，バックグラウンドで PureFlow GSX とのセッション終了処理を行います。このため，複数の urlopen を使用するとき，次の urlopen 使用時に，PureFlow GSX 側では前回のセッションが終了していない場合があります。この操作を繰り返すと，PureFlow GSX 側のセッションリソースが枯渇し，WebAPI が一時的に利用できなくなります。

複数の API を続けて実行する場合，HTTP の持続的接続を利用し，HTTP 接続を維持したまま複数の API を発行するようにプログラミングを行ってください。以下に持続的接続を利用したサンプルプログラムを示します。

② 接続を維持したまま複数の設定を行うサンプルプログラム。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json
import httplib

# URL 定義 PureFlowGSX WebAPI の IP アドレスとファイル名
ip      = '192.168.1.1'
file    = 'shapermng/json'

# コネクションを生成 HTTP
conn = httplib.HTTPConnection(ip)
# コネクションを生成 HTTPS
#conn = httplib.HTTPSConnection(ip)

# パラメータ定義
params = {
                'command': 'add scenario',
                'scenario_name' : '/port1/North',
                'action' : 'aggregate',
                'min_bandwidth' : '100M',
                'peak_bandwidth' : '1G',
                'bufsize' : '1M'
                }
json_data = json.dumps(params)

# POST リクエスト
conn.request("POST", '/'+file, json_data)
response = conn.getresponse()

# レスポンスの表示
print 'RESPONSE:', response

data = response.read()
print 'LENGTH   :', len(data)
print 'DATA     :'
print '---------'
print data
print

# コネクションを開放
conn.close()
```

コンフィギュレーション保存

一連の設定変更が完了したら，コンフィギュレーション保存 API によってコンフィギュレーションの変更を保存してください。

コンフィギュレーション保存 API ではコマンドを送信してレスポンスを確認する手順になります。

コンフィギュレーション保存 API は保存の完了を待たずにレスポンスを返し，バックグラウンドでコンフィギュレーション保存を実行します。保存が実行されている最中にさらに本 API でコンフィギュレーションの保存を指示した場合，"configuration save is in progress"のエラーメッセージを返しますので，レスポンス内容にこのエラーメッセージが表示された場合は時間を空けてからもう一度実行してください。コンフィギュレーション保存の所要時間については「第 3 章　設定の基本」を参照してください。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義  PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義  PureFlowGSX WebAPI の URL HTTPS
#url       = 'https://192.168.1.1/shapermng/json'

#  パラメータ定義
params = {
                'command': 'save config'
                }
json_data = json.dumps(params)

# POST リクエスト
response = urllib2.urlopen(url, json_data)

#  レスポンスの表示
print 'RESPONSE:', response
print 'URL         :', response.geturl()

data = response.read()
print 'LENGTH    :', len(data)
print 'DATA       :'
print '----------'
print data
print
```

コンフィギュレーション保存実行状態取得

コンフィギュレーション保存が実行中かどうかを本 API によって取得できます。
本 API はレスポンスに下記のメッセージを返します。

"configuration save is in progress"　　:コンフィギュレーション保存が実行中です。
"configuration save is not in progress"　:コンフィギュレーション保存は完了しています。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義  PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義  PureFlowGSX WebAPI の URL HTTPS
#url       = 'https://192.168.1.1/shapermng/json'

#  パラメータ定義
params = {
                'command': 'show save status'
                }

# URL エンコードする
params_url       = urllib.urlencode(params)

# GET リクエスト
response = urllib2.urlopen(url+'?'+params_url)

#  レスポンスの表示
print 'RESPONSE:', response
print 'URL         :', response.geturl()

data = response.read()
print 'LENGTH    :', len(data)
print 'DATA       :'
print '----------'
print data
print
```

表示系

設定内容を確認したい場合は show 系の API を使用します。
show 系 API では，コマンドとパラメータを送信してレスポンスの確認とデータを表示する手順になります。1 エントリのみの取得と，全エントリの取得ではプログラミング方法が異なります。各 API についてそれぞれのサンプルコードを示します。

(1) シナリオ情報取得(シナリオ指定)

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義 PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義 PureFlowGSX WebAPI の URL HTTPS
#url      = 'https://192.168.1.1/shapermng/json'

# パラメータ定義
# search_type は exact を指定する
params = {
                'command': 'show scenario',
                'scenario_name': '/port1/North',
                'search_type': 'exact'
                }

# URL エンコードする
params_url       = urllib.urlencode(params)

# GET リクエスト
response = urllib2.urlopen(url+'?'+params_url)

# レスポンスの表示
print 'RESPONSE:', response
print 'URL         :', response.geturl()

data = response.read()
print 'LENGTH   :', len(data)
print 'DATA       :'
print '----------'
print data
print
```

(2) シナリオ情報取得(全取得)

① 接続を維持したまま全シナリオの取得を行うサンプルプログラム。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json
import httplib

# URL 定義  PureFlowGSX WebAPI の IP アドレスとファイル名
ip      = '192.168.1.1'
file    = 'shapermng/json'

# コネクションを生成  HTTP
conn = httplib.HTTPConnection(ip)
# コネクションを生成  HTTPS
#conn = httplib.HTTPSConnection(ip)

# パラメータ定義
# シナリオ全表示のときは，最初のシナリオ名を0文字指定する
# search_type  は  next  を指定する
params = {
                'command': 'show scenario',
                'scenario_name': '',
                'search_type': 'next'
                }

while 1:
            # URL エンコードする
            params_url      = urllib.urlencode(params)

            # GET リクエスト
            conn.request("GET", '/'+file+'?'+params_url)
            response = conn.getresponse()

            # レスポンスの表示
            print 'RESPONSE:', response

            data = response.read()
            print 'LENGTH   :', len(data)
            print 'DATA     :'
            print '----------'
            print data
            print
```

（続き）

```
            # レスポンスのデータ部（JSON 形式の文字列）から
            # Python dictionary  データを取得する
            json_data = json.loads(data)

            # JSON  キーにシナリオ名が存在しない場合は終了
            if json_data.has_key("scenario_name")==False:
                        break

            # シナリオ名を取得する
            scenario_name = json_data['scenario_name']

            # シナリオ名を取得したものに更新して続行
            params['scenario_name'] = scenario_name

# コネクションを開放
conn.close()
```

② 1シナリオの取得ごとに接続と切断を行うサンプルプログラム。

下記サンプルプログラムを使用した場合，実行端末の処理性能によっては，PureFlow GSX 側のセッションリソースが枯渇し，urlopen がエラー終了する場合があります。エラーが発生する場合は，①の接続を維持したまま全シナリオの取得を行うサンプルプログラムを使用してください。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義 PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義 PureFlowGSX WebAPI の URL HTTPS
#url      = 'https://192.168.1.1/shapermng/json'

# パラメータ定義
# シナリオ全表示のときは，最初のシナリオ名を0文字指定する
# search_type は next を指定する
params = {
            'command': 'show scenario',
            'scenario_name': '',
            'search_type': 'next'
            }

while 1:
        # URL エンコードする
        params_url      = urllib.urlencode(params)

        # GET リクエスト
        response = urllib2.urlopen(url+'?'+params_url)

        # レスポンスの表示
        print 'RESPONSE:', response
        print 'URL         :', response.geturl()

        data = response.read()
        print 'LENGTH    :', len(data)
        print 'DATA       :'
        print '----------'
        print data
        print

        # レスポンスのデータ部（JSON 形式の文字列）から
        # Python dictionary データを取得する
        json_data = json.loads(data)

        # JSON キーにシナリオ名が存在しない場合は終了
        if json_data.has_key("scenario_name")==False:
                break

        # シナリオ名を取得する
        scenario_name = json_data['scenario_name']
```

（続き）

```
        # シナリオ名を取得したものに更新して続行
        params['scenario_name'] = scenario_name
```

(3) フィルタ情報取得(フィルタ指定)

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義  PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義  PureFlowGSX WebAPI の URL HTTPS
#url       = 'https://192.168.1.1/shapermng/json'

#  パラメータ定義
# search_type  は  exact  を指定する
params = {
                'command': 'show filter',
                'scenario_name' : '/port1/North',
                'filter_name' : 'filter1',
                'search_type' : 'exact'
                }

# URL エンコードする
params_url       = urllib.urlencode(params)

# GET リクエスト
response = urllib2.urlopen(url+'?'+params_url)

#  レスポンスの表示
print 'RESPONSE:', response
print 'URL         :', response.geturl()

data = response.read()
print 'LENGTH    :', len(data)
print 'DATA       :'
print '----------'
print data
print
```

(4) フィルタ情報取得(全取得)

① 接続を維持したまま全フィルタの取得を行うサンプルプログラム。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json
import httplib

# URL 定義  PureFlowGSX WebAPI の IP アドレスとファイル名
ip       = '192.168.1.1'
file    = 'shapermng/json'

# コネクションを生成  HTTP
conn = httplib.HTTPConnection(ip)
# コネクションを生成  HTTPS
#conn = httplib.HTTPSConnection(ip)

# パラメータ定義
# フィルタ全表示のときは，最初のシナリオ名とフィルタ名を0文字指定する
# search_type  は  next  を指定する
params = {
                'command': 'show filter',
                'scenario_name' : '',
                'filter_name' : '',
                'search_type' : 'next'
                }

while 1:
            # URL エンコードする
            params_url       = urllib.urlencode(params)

            # GET リクエスト
            conn.request("GET", '/'+file+'?'+params_url)
            response = conn.getresponse()
```

（続き）

```
            # レスポンスの表示
            print 'RESPONSE:', response

            data = response.read()
            print 'LENGTH   :', len(data)
            print 'DATA     :'
            print '----------'
            print data
            print

            # レスポンスのデータ部（JSON 形式の文字列）から
            # Python dictionary データを取得する
            json_data = json.loads(data)

            # JSON キーにシナリオ名が存在しない場合は終了
            if json_data.has_key("scenario_name")==False:
                    break

            # JSON キーにフィルタ名が存在しない場合は終了
            if json_data.has_key("filter_name")==False:
                    break

            # シナリオ名を取得する
            scenario_name = json_data['scenario_name']

            # フィルタ名を取得する
            filter_name = json_data['filter_name']

            # シナリオ名とフィルタ名を取得したものに更新して続行
            params['scenario_name'] = scenario_name
            params['filter_name'] = filter_name

# コネクションを開放
conn.close()
```

② 1フィルタの取得ごとに接続と切断を行うサンプルプログラム。

下記サンプルプログラムを使用した場合，実行端末の処理性能によっては，PureFlow GSX 側のセッションリソースが枯渇し，urlopen がエラー終了する場合があります。エラーが発生する場合は，①の接続を維持したまま全シナリオの取得を行うサンプルプログラムを使用してください。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義 PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義 PureFlowGSX WebAPI の URL HTTPS
#url       = 'https://192.168.1.1/shapermng/json'

# パラメータ定義
# フィルタ全表示のときは，最初のシナリオ名とフィルタ名を0文字指定する
# search_type は next を指定する
params = {
                'command': 'show filter',
                'scenario_name' : '',
                'filter_name' : '',
                'search_type' : 'next'
                }

while 1:
        # URL エンコードする
        params_url       = urllib.urlencode(params)

        # GET リクエスト
        response = urllib2.urlopen(url+'?'+params_url)

        # レスポンスの表示
        print 'RESPONSE:', response
        print 'URL         :', response.geturl()

        data = response.read()
        print 'LENGTH    :', len(data)
        print 'DATA       :'
        print '----------'
        print data
        print

        # レスポンスのデータ部（JSON 形式の文字列）から
        # Python dictionary データを取得する
        json_data = json.loads(data)

        # JSON キーにシナリオ名が存在しない場合は終了
        if json_data.has_key("scenario_name")==False:
                break
```

（続き）

```
# JSON キーにフィルタ名が存在しない場合は終了
if json_data.has_key("filter_name")==False:
                    break


# シナリオ名を取得する
scenario_name = json_data['scenario_name']


# フィルタ名を取得する
filter_name = json_data['filter_name']

# シナリオ名とフィルタ名を取得したものに更新して続行
params['scenario_name'] = scenario_name
params['filter_name'] = filter_name
```

(5) ルールリスト情報取得(ルールリストエントリ指定)

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義  PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義  PureFlowGSX WebAPI の URL HTTPS
#url       = 'https://192.168.1.1/shapermng/json'

#  パラメータ定義
# search_type  は  exact  を指定する
params = {
                'command': 'show rulelist',
                'list_name' : 'v4servers',
                'rules' : '192.168.10.1-192.168.10.1',
                'search_type': 'exact'
                }

# URL エンコードする
params_url       = urllib.urlencode(params)

# GET リクエスト
response = urllib2.urlopen(url+'?'+params_url)

#  レスポンスの表示
print 'RESPONSE:', response
print 'URL         :', response.geturl()

data = response.read()
print 'LENGTH    :', len(data)
print 'DATA       :'
print '----------'
print data
print
```

(6) ルールリスト情報取得(全取得)

① 接続を維持したまま全ルールリストの取得を行うサンプルプログラム。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json
import httplib

# URL 定義  PureFlowGSX WebAPI の IP アドレスとファイル名
ip       = '192.168.1.1'
file    = 'shapermng/json'

# コネクションを生成  HTTP
conn = httplib.HTTPConnection(ip)
# コネクションを生成  HTTPS
#conn = httplib.HTTPSConnection(ip)

# パラメータ定義
# ルールリスト全表示のときは，ルールリスト名とルールリストエントリを
# 0文字指定する
# search_type  は  next  を指定する
params = {
                'command': 'show rulelist',
                'list_name' : '',
                'rules' : '',
                'search_type': 'next'
                }

while 1:
            # URL エンコードする
            params_url       = urllib.urlencode(params)

            # GET リクエスト
            conn.request("GET", '/'+file+'?'+params_url)
            response = conn.getresponse()
```

（続き）

```
        # レスポンスの表示
        print 'RESPONSE:', response

        data = response.read()
        print 'LENGTH   :', len(data)
        print 'DATA     :'
        print '----------'
        print data
        print

        # レスポンスのデータ部（JSON 形式の文字列）から
        # Python dictionary データを取得する
        json_data = json.loads(data)

        # JSON キーにルールリスト名が存在しない場合は終了
        if json_data.has_key("list_name")==False:
                break

        # JSON キーにルールリストエントリが存在しない場合は終了
        if json_data.has_key("rules")==False:
                break

        # ルールリスト名を取得する
        list_name = json_data['list_name']

        # ルールリストエントリを取得する
        rules = json_data['rules']

        # ルールリスト名とルールリストエントリを取得したものに更新して続行
        # none の場合は次のルールリストエントリがないことを示し、
        # 次のルールリストを取得するために rules を 0 文字指定する
        params['list_name'] = list_name
        if rules == 'none':
                params['rules'] = ''
        else:
                params['rules'] = rules

# コネクションを開放
conn.close()
```

③　1 ルールリストの取得ごとに接続と切断を行うサンプルプログラム。

下記サンプルプログラムを使用した場合，実行端末の処理性能によっては，PureFlow GSX 側のセッションリソースが枯渇し，urlopen がエラー終了する場合があります。エラーが発生する場合は，①の接続を維持したまま全シナリオの取得を行うサンプルプログラムを使用してください。

```
# -*- coding: utf-8 -*-
import urllib
import urllib2
import json

# URL 定義 PureFlowGSX WebAPI の URL HTTP
url       = 'http://192.168.1.1/shapermng/json'
# URL 定義 PureFlowGSX WebAPI の URL HTTPS
#url       = 'https://192.168.1.1/shapermng/json'

# パラメータ定義
# ルールリスト全表示のときは，ルールリスト名とルールリストエントリを
# 0文字指定する
# search_type は next を指定する
params = {
                'command': 'show rulelist',
                'list_name' : '',
                'rules' : '',
                'search_type': 'next'
                }

while 1:
            # URL エンコードする
            params_url       = urllib.urlencode(params)

            # GET リクエスト
            response = urllib2.urlopen(url+'?'+params_url)

            # レスポンスの表示
            print 'RESPONSE:', response
            print 'URL         :', response.geturl()

            data = response.read()
            print 'LENGTH    :', len(data)
            print 'DATA       :'
            print '----------'
            print data
            print

            # レスポンスのデータ部（JSON 形式の文字列）から
            # Python dictionary データを取得する
            json_data = json.loads(data)

            # JSON キーにルールリスト名が存在しない場合は終了
            if json_data.has_key("list_name")==False:
                        break
```

（続き）

```
# JSON キーにルールリストエントリが存在しない場合は終了
if json_data.has_key(″rules″)==False:
        break


#  ルールリスト名を取得する
list_name = json_data['list_name']

#  ルールリストエントリを取得する
rules = json_data['rules']

#  ルールリスト名とルールリストエントリを取得したものに更新して続行
# none の場合は次のルールリストエントリがないことを示し、
#  次のルールリストを取得するために rules を 0 文字指定する
params['list_name'] = list_name
if rules == 'none':
        params['rules'] = ''
else:
        params['rules'] = rules
```

Anritsu

アンリツ株式会社

管理番号：
NF7101-W008J
Printed in Japan