重み付けランダム早期検出の設定

- ネットワーク輻輳の回避 （1 ページ）
- テール ドロップ （1 ページ）
- 重み付けランダム早期検出 （2 ページ）
- WRED 設定の制限 （3 ページ）
- WRED 使用上の注意事項 （3 ページ）
- WRED の設定 （4 ページ）
- WRED の設定例 （8 ページ）
- 階層化 QoS を使用した WRED のサポート （8 ページ）
- WRED 設定の確認 （9 ページ）
- WRED 設定のベスト プラクティス （11 ページ）
- 重み付けランダム早期検出の機能履歴 （12 ページ）

ネットワーク輻輳の回避

異種ネットワークには、アプリケーションが使用する異なるプロトコルが含まれており、これにより、ファイル転送などの時間依存が比較的少ないアプリケーションのニーズに対処しながら、タイムクリティカルなアプリケーションに応えるためにトラフィックの優先順位を付ける必要が生じています。ネットワーク内のデバイス間で単一のデータ パスを共有するさまざまなタイプのトラフィックをサポートするようにネットワークが設定されている場合、輻輳回避メカニズムを実装することにより、さまざまなタイプのトラフィックを公平に処理し、共通のネットワーク ボトルネックでの輻輳を回避できます。輻輳回避メカニズムは、パケットのドロップにより実現します。

ランダム早期検出（RED）は、ネットワークで一般的に使用される輻輳回避メカニズムです。

テール ドロップ

テール ドロップでは、すべてのトラフィックを平等に扱い、サービス クラス内では差別化しません。出力キューが一杯でテール ドロップが有効な場合、輻輳が解消されてキューが一杯でなくなるまでパケットはドロップされます。

重み付けランダム早期検出

RED メカニズムは、TCP の輻輳制御メカニズムを利用します。輻輳が頻繁に発生する前にパケットがランダムにドロップされます。パケット送信元が TCP を使用する場合、送信元はすべてのパケットが宛先に届くようになるまで送信速度を下げます。これは輻輳が解消されたことを示します。RED を、TCP のパケットの転送速度を下げる方法として使用できます。TCP は停止するだけでなく、素早く再起動して、ネットワークがサポート可能なレートに伝送レートを対応させます。

WRED は、シスコが実装している RED です。RED アルゴリズムの機能と、IP プレシデンス、DiffServ コード ポイント（DSCP）、またはサービス クラス（CoS）の値を組み合わせています。

WRED の仕組み

WREDは、出力インターフェイスにネットワーク混雑の兆候が表れた際に、選択的にパケットをドロップしてテール ドロップの確率を減らします。WRED は、キューが一杯になるまで待機するのではなく、一部のパケットを早期にドロップします。そのため、一度に大量のパケットをドロップすることを防ぎ、TCP グローバル同期の可能性を最小限に抑えます。

Approximate Fair Drop（AFD）は、パケットのドロップ確率を決定するアクティブキュー管理（AQM）アルゴリズムです。パケットをドロップする確率は、入力時のフローの着信レート計算と現在のキュー長によって異なります。

AFD ベースの WRED は、有線ネットワークポートに実装されます。

AFD ベースの WRED は、WRED の優先的なドロップ動作をエミュレートします。この優先的なドロップ動作は、WRED の対応するドロップしきい値に基づいて AFD サブクラスの重みを変更することで実現します。物理キュー内では、重みが大きいトラフィクのドロップ確率は、重みの小さいトラフィックよりも低くなります。

- 各 WRED 対応キューには、上限と下限のしきい値があります。
- 優先度の高いサブクラスには 大きな AFD の重みが設定されます。
- サブクラスは、最も低い WRED minThreshold に基づいて昇順でソートされます。

WRED 重み計算

AFD の重みは、下限と上限のしきい値を使用して計算されます。AFD は、WRED の上限と WRED の下限のしきい値の平均を表す調整されたインデックスです。

パケットがインターフェイスに着信すると、次のイベントが発生します。

1. ドロップ確率が計算されます。AFD の重みが減少するほど、ドロップ確率は高くなります。つまり、下限と上限のしきい値の平均が小さいほど、ドロップ確率は高くなります。

2. WREDは、パケットのドロップを決定する前に、パケットフローのプライオリティとしきい値を検討します。CoS、DSCP、またはIP Precedenceの値は、指定されたしきい値にマッピングされます。これらのしきい値を超えると、これらのしきい値にマッピングされた設定値を持つパケットはドロップの対象になります。高いしきい値に割り当てられたCoS、DSCP、またはIP Precedence値を持つその他のパケットは、キューに入れられます。このプロセスにより、プライオリティの高いフローがそのまま維持され、パケット伝送の遅延が最小限に抑えられます。
3. パケットがWREDを使用してドロップされない場合、テールドロップされます。

WRED 設定の制限

- デフォルトでは、重み付きテール ドロップ（WTD）がすべてのキューでイネーブルになっています。
- WREDはキューごとに有効または無効にできます。WREDを無効にすると、WTDがターゲットキューに適用されます。WREDプロファイルを持つポリシーマップは出力ポリシーとして物理ポート上にのみ設定されます。
- WREDは、ネットワーク ポート キューのみでサポートされており、内部CPUキューとスタック キューではサポートされていません。
- 各WRED物理キューは、一意のWREDしきい値ペア設定を使用して3つのサブキューをサポートできます。
- WREDとともに、ポリシーマップで帯域幅または形状を設定することを確認します。
- すべてのWREDしきい値は必ずパーセンテージ モードで指定します。
- WREDしきい値ペアのマッピングは、対応する一致フィルタを使用してクラスマップフィルタをマッピングすることで行います。

  「any」一致フィルタが設定されたクラスマップをお勧めします。
- プライオリティトラフィックのWREDはサポートされていません。
- WREDとキュー制限は、同じポリシーではサポートされません。
- 有線ポートは最大で8つの物理キューをサポートします。そのうちの4つの物理キューでそれぞれが3つのしきい値ペアを持つWREDを設定できます。残りのキューは、WTDで設定されます。5つ以上のWREDキューを持つポリシーは拒否されます。

WRED 使用上の注意事項

AFDベースのWRED機能を設定するには、ポリシー マップを指定し、クラスを追加します。**random-detect** コマンドを使用し、ドロップ確率の計算にWREDが使用する方式を（dscp-based/cos-based/cos-based引数を使用して）指定します。

（注） ポリシーは作業中に変更できます。AFD の重みが自動的に再計算されます。

WRED は IPv4/IPv6、マルチキャストなどのどのような種類のトラフィックにも設定できます。WRED は、8 つのキューイングクラスすべてでサポートされます。

random-detect コマンドを使用して WRED を設定する場合は次の点を考慮してください。

- dscp-based 引数を使用する場合、WRED は DSCP 値を使用してドロップ確率を計算します。
- cos-based 引数を使用する場合、WRED は CoS 値を使用してドロップ確率を計算します。
- デフォルトでは、WRED はドロップ確率の計算に IP precedence 値を使用します。**precedence-based** 引数がデフォルトであり、CLI には表示されません。

（注） **show run policy-map** *policy-map* コマンドは、**random-detect** コマンドで precedence が設定されていても、「precedence」を表示しません。

- dscp-based 引数と precedence-based 引数は、相互に排他的です。
- 8 つの物理キューを、それぞれ異なる WRED プロファイルで設定できます。

WRED の設定

DSCP 値に基づく WRED の設定

DSCP 値に基づいて WRED プロファイルをパケット モードで設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**class-map** *match-criteria class-name* 例： `device(config)# class-map match-any CS`	クラスマップに一致基準を設定します。 推奨する一致基準は match-any です。
ステップ 2	**match** *class-map-name* 例： `device(config-cmap)#match dscp CS1`	クラスマップを照合します。

	コマンドまたはアクション	目的
ステップ 3	**policy-map** *name* 例： `device(config)#policy-map PWRED`	作成する WRED プロファイル ポリシーの名前を指定します。
ステップ 4	**class** *class-name* 例： `device(config-pmap)#class CS`	ポリシーに関連付けるクラスの名前を指定します。
ステップ 5	Use either **bandwidth** {*kbps*| **remaining** *percentage* | **percent** *percentage*} or **shape** { **average** | **peak** }*cir* 例： `device(config-pmap-c)#bandwidth percent 10`	ポリシーマップに属しているクラスに割り当てる帯域幅またはトラフィック シェーピングを指定します。
ステップ 6	**random-detect** *dscp-based* 例： `device(config-pmap-c)#random-detect dscp-based`	パケットのドロップ確率を計算する際には DSCP 値を使用するように WRED を設定します。
ステップ 7	**random-detect dscp** *dscp-value* **percent** *minThreshold maxThreshold* 例： `device(config-pmap-c)#random-detect dscp cs1 percent 10 20`	最小しきい値および最大しきい値をパーセンテージで指定します。
ステップ 8	**interface** *interface-name* 例： `device(config)#interface HundredGigE1/0/2`	インターフェイスコンフィギュレーション モードを開始します。
ステップ 9	**service-policy output** ポリシーマップ 例： `device(config-if)#service-policy output pwred`	ポリシー マップを出力インターフェイスに付加します。

サービス クラス値に基づく WRED の設定

サービスクラス（CoS）値に基づいてWREDプロファイルをパケットモードで設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**class-map** *match-criteria class-name* 例： `device(config)# class-map match-any CS`	クラスマップに一致基準を設定します。 推奨する一致基準は match-any です。
ステップ 2	**match** *class-map-name* 例： `device(config-cmap)#match cos 3`	クラスマップを照合します。
ステップ 3	**policy-map** *name* 例： `device(config)#policy-map PWRED`	作成する WRED プロファイル ポリシーの名前を指定します。
ステップ 4	**class** *class-name* 例： `device(config-pmap)#class CS`	ポリシーに関連付けるクラスの名前を指定します。
ステップ 5	**bandwidth** {*kbps*| **remaining** *percentage* | **percent** *percentage*} 例： `device(config-pmap-c)#bandwidth percent 10`	ポリシー マップに属しているクラスに割り当てる帯域幅を指定します。
ステップ 6	**random-detect** *cos-based* 例： `device(config-pmap-c)#random-detect cos-based`	パケットのドロップ確率を計算する際には CoS 値を使用するように WRED を設定します。
ステップ 7	**random-detect cos** *cos-value* **percent** *minThreshold maxThreshold* 例： `device(config-pmap-c)#random-detect cos 3 percent 10 20`	最小しきい値および最大しきい値をパーセンテージで指定します。
ステップ 8	**interface** *interface-name* 例： `device(config)# interface HundredGigE1/0/2`	インターフェイスコンフィギュレーション モードを開始します。
ステップ 9	**service-policy output** ポリシーマップ 例： `device(config-if)#service-policy output pwred`	ポリシー マップを出力インターフェイスに付加します。

IP プレシデンス値に基づく WRED の設定

IP プレシデンス値に基づいて WRED プロファイルをパケット モードで設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**class-map** *match-criteria class-name* 例： `device(config)# class-map match-any CS`	クラスマップに一致基準を設定します。 推奨する一致基準は match-any です。
ステップ 2	**match** *class-map-name* 例： `device(config-cmap)#match precedence 3`	クラスマップを照合します。
ステップ 3	**policy-map** *name* 例： `device(config)#policy-map pwred`	作成する WRED プロファイル ポリシーの名前を指定します。
ステップ 4	**class** *class-name* 例： `device(config-pmap)#class CS`	ポリシーに関連付けるクラスの名前を指定します。
ステップ 5	**bandwidth** {*kbps*| **remaining** *percentage* | **percent** *percentage*} 例： `device(config-pmap-c)#bandwidth percent 10`	ポリシー マップに属しているクラスに割り当てる帯域幅を指定します。
ステップ 6	**random-detect** *precedence-based* 例： `device(config-pmap-c)#random-detect precedence-based`	パケットのドロップ確率を計算する際には IP プレシデンス値を使用するように WRED を設定します。
ステップ 7	**random-detect precedence** *precedence-value* **percent** *minThreshold maxThreshold* 例： `device(config-pmap-c)#random-detect precedence 3 percent 10 20`	最小しきい値および最大しきい値をパーセンテージで指定します。
ステップ 8	**interface** *interface-name* 例：	インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`device(config)#interface HundredGigE1/0/2`	
ステップ 9	**service-policy output** ポリシーマップ **例：** `device(config-if)#service-policy output pwred`	ポリシー マップを出力インターフェイスに付加します。

WRED の設定例

次に、クラス CS の DSCP プロファイルを使用するように WRED をイネーブルにする例を示します。この例では、cs1、cs2、および cs3 という 3 つのサブクラスを WRED の最小しきい値および最大しきい値で設定し、最終的にはポリシーを 100 ギガビット イーサネット インターフェイス 8 に適用します。

```
Device(config)# class-map match-any CS
Device(config-cmap)# match dscp cs1
Device(config-cmap)# match dscp cs2
Device(config-cmap)# match dscp cs3
Device(config-cmap)# policy-map PWRED
Device(config-pmap)# class CS
Device(config-pmap-c)# bandwidth percent 10
Device(config-pmap-c)# random-detect dscp-based
Device(config-pmap-c)# random-detect dscp cs1 percent 10 20
Device(config-pmap-c)# random-detect dscp cs2 percent 20 30
Device(config-pmap-c)# random-detect dscp cs3 percent 34 44
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface HundredGigE1/0/8
Device(config-if)# service-policy output PWRED
```

階層化 QoS を使用した WRED のサポート

階層型 QoS では、トラフィック管理をより細かい粒度で実行する、複数のポリシー レベルで QoS 動作を指定できます。

HQoS の場合、子ポリシーでのみ WRED が許可され、親ポリシーでは許可されません。親ポリシーにシェーピングを、子ポリシーに WRED を設定できます。

次に、親ポリシー **pwred-parent** を帯域幅の 10 パーセントでシェーピングしたトラフィックで設定し、それを DSCP ベースの WRED に設定されたその子ポリシー **pwred-child** に適用する例を示します。

```
policy-map PWRED-CHILD
 class CWRED
    bandwidth percent 10
    random-detect dscp-based
    random-detect dscp 1 percent 10 20
    random-detect dscp 10 percent 20 30
```

```
policy-map PWRED-PARENT
  class class-default
  shape average percent 10
  service-policy PWRED-CHILD
```

次に、HQoS WRED 設定を確認する show コマンドを示します。

```
device# show policy-map PWRED-PARENT
  policy Map PWRED-PARENT
    class class-default
      average Rate Traffic Shaping
      cir 30%
      service-policy PWRED-CHILD
policy-map PWRED-CHILD
 class CWRED
  bandwidth percent 10
  random-detect dscp-based
  random-detect dscp 1 percent 10 20
  random-detect dscp 10 percent 20 30
policy-map PWRED-PARENT
 class class-default
   shape average percent 30
   service-policy PWRED-CHILD
```

WRED 設定の確認

次の show コマンドを使用して、WRED の設定を確認します。

手順

ステップ 1 **show policy-map** *policy-map-name*

WRED としきい値のラベルが表示されます。

例：

```
Device# show policy-map PWRED
  Policy Map PWRED
    Class CS
      bandwidth 10 (%)
       percent-based wred

      dscp    min-threshold    max-threshold
      ------------------------------------
      cs1 (8)      10               20
      cs2 (16)     20               30
      cs3 (24)     34               44
      default (0)   -
```

ステップ 2 **show policy-map interface** *interface-name*

WRED AFD の重み、WRED Enq（パケット数およびバイト数）、WRED ドロップ（パケット数およびバイト数）、しきい値ペアに対して設定された DSCP ラベルが表示されます。

（注）

トラフィックを開始した後にのみ、このコマンドを使用します。**show policy-map interface** は、トラフィックが送信された後にのみ、WRED 設定が更新されます。

例：

```
Device#show policy-map interface HundredGigE 1/0/2
 HundredGigE1/0/2

  Service-policy output: PWRED

    Class-map: CS (match-any)
      0 packets
      Match:  dscp cs1 (8)
      Match:  dscp cs2 (16)
      Match:  dscp cs3 (24)
      Queueing

      (total drops) 27374016
      (bytes output) 33459200081
      bandwidth 10% (1000000 kbps)

AFD WRED STATS BEGIN
Virtual Class   min/max        Transmit                     Random drop
 AFD Weight


   0          10 / 20        (Byte)33459183360             27374016
 12
                             (Pkts)522799759                  427716

             dscp : 8


   1          20 / 30          Byte)0                        0
 20
                               (Pkts)0                       0

             dscp : 16


   2          34 / 44        (Byte)16721                     0
 31
                               (Pkts)59                      0
             dscp : 24

     Total Drops(Bytes)   : 27374016

     Total Drops(Packets) : 427716
AFD WRED STATS END


    Class-map: class-default (match-any)
      0 packets
      Match: any


      (total drops) 0
      (bytes output) 192
```

WRED 設定のベスト プラクティス

- **3 つの WRED 設定ペアのサポート**

  各 WRED 物理キュー（AFD キュー）は、一意の WRED しきい値ペア設定を使用して 3 つの WRED 設定ペアをサポートできます。

```
Policy-map P1
    Class CS
        Random-detect dscp-based
        Random-detect dscp CS1 percent 10 20      // WRED pair 1
        Random-detect dscp CS2  percent 20 30      // WRED pair 2
        Random-detect dscp CS3 percent 30 40      // WRED pair 3
Class-map match-any CS
        match cs1
        match cs2
        match cs3
```

- **WRED 設定ペアの追加**

  重複するしきい値ペアを WRED 設定ペアに追加できます。

```
Policy-map P1
    Class CS
       Random-detect dscp-based
       Random-detect dscp CS1 percent 10 20       // WRED pair 1
       Random-detect dscp CS2 percent 20 30       // WRED pair 2
       Random-detect dscp CS3 percent 30 40       // WRED pair 3
       Random-detect dscp CS4 percent  30 40     ==> belongs to WRED pair 3
       Random-detect dscp CS5 percent 20 30     ==> belongs to WRED pair 2
Class-map match-any CS
       match cs1
       match cs2
       match cs3
       match cs4  >>
       match cs5  >>
```

- **デフォルトの WRED ペア**

  2 つ以下の WRED ペアが設定されている場合、WRED に参加しているどのクラスマップフィルタも最大しきい値（100, 100）でデフォルトの 3 番目の WRED ペアに割り当てられます。

```
Policy-map P1
    Class CS
        Random-detect dscp-based
        Random-detect dscp CS1 percent 10 20   // WRED pair 1
        Random-detect dscp CS2 percent 20 30   // WRED pair 2
Class-map match-any CS
        match CS1
        match CS2
        match CS3
        match CS4
```

  この場合は、CS3 と CS4 のクラスはしきい値（100, 100）で WRED ペア 3 にマッピングされます。

- **一致しない設定の拒否**

クラスマップ内に一致フィルタがない場合にrandom-detectを設定すると、ポリシーのインストールが拒否されます。

```
Class-map match-any CS
       match CS1
       match CS2
       match CS5
Policy-map P1
    Class CS
        Shape average percent 10
        Random-detect dscp-based
        Random-detect dscp CS1 percent 10 20   // WRED pair 1
        Random-detect dscp CS2 percent 20 30   // WRED pair 2
        Random-detect dscp CS3 percent 30 40   // WRED pair 3  ===> Mismatched
sub-class.
```

このポリシーを出力側のインターフェイスに適用すると、クラスマップ値が不正であるとして、インストール時にそのポリシーは失敗します。

```
device(config)# int Fo1/0/5
device(config-if)# service-policy output P1
device(config-if)#
*Feb 20 17:33:16.964: %IOSXE-5-PLATFORM: Switch 1 R0/0: fed: WRED POLICY INSTALL
FAILURE.Invalid WRED filter mark: 24 in class-map: CS
*Feb 20 17:33:16.965: %FED_QOS_ERRMSG-3-LABEL_2_QUEUE_MAPPING_HW_ERROR: Switch 1
R0/0: fed: Failed to detach queue-map for FortyGigabitEthernet1/0/5: code 2
```

重み付けランダム早期検出の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.2	重み付けランダム早期検出メカニズム	WRED は、ネットワーク内の輻輳を回避するメカニズムです。WREDは、出力インターフェイスにネットワーク混雑の兆候が表れた際に、選択的にパケットをドロップしてテール ドロップの確率を減らし、多数のパケットが一度にドロップされないようにします。次の値に基づいて動作するようにWRED を設定できます。 • DiffServ コードポイント • IP Precedence • サービスクラス

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。

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