

複数のスパニングツリー移行のベストプラクティス

目的

このドキュメントの目的は、マルチスパニングツリー(MSTP)に移行する際のベストプラクティスを提供することです。他のスパニングツリーバリエーションに対してMSTPを使用すると、ネットワークの効率と信頼性が向上します。

要件

- 混合ハードウェア環境でレイヤ2を最適化する必要性
 - Cisco Small Businessスイッチ
 - Sx250シリーズ([管理ガイド](#))
 - Sx300シリーズ([管理ガイド](#))
 - Sx350シリーズ([管理ガイド](#))
 - SG350Xシリーズ([管理ガイド](#))
 - Sx550Xシリーズ([管理ガイド](#))
 - Cisco Catalystスイッチ
- スパニングツリーに関する実務的な理解([詳細はこちら](#))
- Wireshark (オプション)

目次

1. [MSTPの用語](#)
2. [ベストプラクティス#1:MSTPへの移行の必要性を検証する](#)
3. [ベストプラクティス#2 – 移行の戦略](#)
4. [ベストプラクティス#3 – ベストプラクティス#3 - PortFastを使用するためのポイントツーポイントポートの有効化](#)
5. [ベストプラクティス#4 : エッジポートでBPDUガードを有効にする](#)
6. [ベストプラクティス#5:VLANをIST\(MST0\)ではなくMSTIにマッピングする](#)
7. [ベストプラクティス#6 : すべてのMSTP対応スイッチを同じ領域内に配置する](#)
8. [ベストプラクティス#7 – プライマリMST領域内にCISTのルートブリッジをネストする](#)
9. [移行の検証 – これはオンですか？](#)
10. [結論](#)

このガイドの構成

このガイドでは、SSH経由でデバイスにログインする手順や、管理インターフェイスを省略します。代わりに、コアコマンドを強調表示します。各ベストプラクティスには、シスコのハードウェア (エンタープライズおよびSMB) の混在に関する適切な手順を説明するサブタスクが含まれます。設定ガイドについては、次の2つのリンクを参照してください。

- [SMBスイッチでのMSTPの設定](#)
- [CatalystスイッチでのMSTPの設定](#)

MSTPの用語

このセクションは、プレイ中のプロトコルのアクセス可能なメンタルモデルを提供することを目的としています。定義は、MSTPプロトコルの連動コンポーネントです。詳細については、サブ箇条書きの項目を参照してください。

BPDU - Bridge Protocol Data Unit(BPDU) : スイッチの動作を継続するために必要なすべての情報を含むマルチキャストフレームです。

注 : インスタンスのマッピング自体がBPDU内にはないことを示しています。

Region - (MSTPに固有) : ある地域では、VLANごとに1つのBPDUを送信する他のSTPプレーヤーによって発生する問題が解決されます。Per Vlan Spanning Tree(PVST)と同様に、大量のBPDUを送信すると、CPUの負荷が高くなり、ネットワークのパフォーマンスが低下します。代わりに、MSTPを使用すると、すべてのVLANが1つのリージョンにマッピングされます。

インスタンス : インスタンスは、特定の領域に対するVLANまたは多数のVLANの論理テーブルのテーブルです。このインスタンスはゾーンにマッピングされます。移行の一部として、次の手順を実行します。

デフォルトインスタンス0 (ゼロ) は、次の用語MST0、*Internal Spanning Tree(IST)*と同義です。

ユーザが作成したインスタンスは、マルチスパンニングツリーインスタンス(MSTI)として参照されます。

ネットワークのVLANに関する適切なドキュメントを作成すれば、頭痛の種を減らすことができます。

- インスタンスが失敗しても、他のインスタンスには影響しません。

MSTI - Multiple Spanning Tree Instances – 管理上で作成されたインスタンスが含まれます。これらのマッピングは、「MRecord」と呼ばれるものに含まれており、Wiresharkを介して表示できます。レコードには、インスタンスのトポロジを管理するために必要な詳細が含まれます。

IST : 内部スパンニングツリー:MSTPゾーンに参加しているスイッチのレコードです。ゾーン内に含まれるスイッチ (数にかかわらず) は、ゾーン外のエリアに対して1つのスイッチとして表されます。

- **CST (共通スパンニングツリー)** は、独自の従来のスパンニングツリーを実行するMSTP領域で構成されます。CSTは、MSTPゾーンの境界でスイッチ間のリンクを使用します。

CIST : 共通および内部スパンニングツリー:VLANとインスタンスの共有マッピングに基づいて複数のインスタンスを通過するCSTとISTの両方で構成されます。

共通スパニングツリーと内部スパニングツリーは共通スパニングツリーではありません。

この記事の対象者と関連する定義を確認できたので、ベストプラクティスを見てみましょう。

ベストプラクティス#1:MSTPへの移行の必要性を検証する

最初のベストプラクティスは、MSTPへの移行の必要性を確認することです。この決定では、ネットワークの既存のスパニングツリーパフォーマンスを理解することが重要です。MSTPへの移行は、いくつかの理由から最適なオプションです。ロードシェアリングが導入され、ネットワークの効率に最大の影響を与えます。レイヤ2トラフィックが予測よりも先に増加している場合は、MSTPに移行することで、パフォーマンスが向上し、機器の有用性と寿命が向上します。その他の考慮事項：

既存のSTPパフォーマンスが不十分である：コンバージェンス時間が短いか、BPDUが送信される量が問題の原因である

セグメントスパニングツリー:MSTPリージョンに含まれるスイッチのリソース負荷を軽減します。

混合ハードウェア環境:MSTPはオープンスタンダードであるため、マルチベンダー環境に適しています。広く支持されている。

注：一般的な誤解として、多重スパニングツリーに移行する場合は、インスタンスごとに1つのVLANをマッピングする必要があります。

スパニングツリーの種類が増え、以前のバージョンではバリエーションやツイストが発生しています。Per VLAN Spanning Tree(PVST+)と比較すると、MSTPは、スパニングツリーのインスタンスまたはスパニングツリーの論理バージョンを維持することで、使用するリソース (BPDU、CPUサイクル、送信時間) を削減します。VLANトラフィックは、ネットワークのレイヤ2セグメントを通過するように有効になります。1つのポート (およびVLAN) の転送は、別のVLANの転送もブロックできます。さらに、あるインスタンスにループが形成されても、他のインスタンスには影響しません。

ベストプラクティス#2 – 移行の戦略

移行の必要性を検証したら、理想的には、移行は最小限のダウンタイムで実現され、既存の接続は維持されます。移行に取り組むちょっとした戦略は、スムーズな展開を実現するために長い道のりを歩むでしょう。このプロセスを支援するために、次の戦術的な手順を推奨します。

1. **ドキュメント、ドキュメント、ドキュメント**：詳細なメモを保持することで、移行時間が短縮され、エラーが発生する可能性が低くなります。

すべてのポイントツーポイントポート、または別のスイッチまたはルータにつながるポートを特定して文書化します。

すべてのエッジポート、またはPCやプリンタなどのエンドポイントにつながるポートを特定してドキュメント化します。

移行に参加しているVLANを定義します

インターンは本当にこのステップが得意です！

ネットワークの動作の順序を決定します。

あるスイッチでの変更が別のVLANにどのように影響するかを確認してください。

ネットワークのダウンタイムをスケジュールするか、週末に移行します。

ネットワークのコアで移行を開始し、ディストリビューション層、アクセス層へと移行します。

ベストプラクティス#3：ポイントツーポイントポートでPortFastを使用できるようにする

このベストプラクティスと次のベストプラクティスは、すべてのポートのドキュメントを有効に活用します。管理者は、PortFast機能を使用してエッジポートにオプションのパラメータを定義します。PortFastは、スパニングツリーがそのポートで実行されることを防止します。スイッチとギアの向かい合うポートには、サーバ、ワークステーション、ルータが含まれます。このポートは、ネットワークを別のオープンポートのセットにブリッジしないようにすることを目的としています。スイッチが上位のBPDUを受信すると、ループが発生する可能性があります。ネットワークにオンラインになるポートでは、ポートでSTP計算が行われます。ブロッキングステータスを事前に割り当てることで、時間とCPU負荷を節約できます。ポートがBPDU送信 - 転送状態に迅速に移行できます。ステータスが事前に割り当てられているため。

注：スイッチのポートが全二重伝送用に設定されていることを確認します。

次の手順は、SMBスイッチ(CLI + GUI)とEnterprise Catalystスイッチ(CLI)で分けられます。

CatalystスイッチでのPortFastの有効化：CLI

CLIコマンドの構文が最初に示され、その後にライブコマンドの例が示されます。#の後に余分なスペースが追加され、コピーのハイライト表示>貼り付けを少し簡単に行うことができます。青色で強調表示されたテキストは、ネットワークからコンテキストの詳細に置き換えられる変数を示します。また、簡略化のために、MSTP設定で使用するprivilege elevationコマンドは唯一ありません。

```
Catalyst(config)# interface [range(optional)] [port-id]
Catalyst(config-if)# spanning-tree portfast [auto]
```

```
Catalyst(config)# interface range fa0/1 - 24
Catalyst(config-if)# spanning-tree portfast auto
```

SMBスイッチでのPortFastの有効化：CLI

```
SMBswitch(config)# interface [range(optional)] [port-id]
SMBswitch(config-if)# spanning-tree portfast
```

```
SMBswitch(config)# interface range gi1-15
SMBswitch(config-if)# spanning-tree portfast
```

SMBスイッチでのPortFastの有効化：GUI

注意すべき点の1つは、SMBスイッチのGUIではPortFastという同義語を使用していることです。これはFast Linkと呼ばれています。

ステップ1:[スパンニングツリー]>[STPインターフェイス設定]をクリックします。

ステップ2：インターフェイスを選択し、[Edit]ボタンをクリックします。

ステップ3:[Enable Fast Link]をクリックします。

注：変更を適用し、スタートアップコンフィギュレーションに実行コンフィギュレーションを書き込むことを忘れないでください。

ベストプラクティス#4：エッジポートでBPDUガードを有効にする

このベストプラクティスは、前のベストプラクティスの延長です。BPDUガードが有効なポートで上位のトポロジ変更BPDUを受信しているポートが検出された場合、そのポートはerr-disableステータスを使用してただちにシャットダウンします。そのため、スイッチにアクセスして状況を解決する必要があります。

注：これは、スキップできるベストプラクティスの1つに見えるかもしれませんが。逃げられるか？たぶん、未来の自分のために、そう作りなさい。1台の誤ったスイッチがネットワークに接続され、誤ったBPDUが送出されると、ネットワークが破壊される可能性があります。

CatalystスイッチでのBPDUガードの有効化：CLI

```
Catalyst(config)# interface [range(optional)] [port-id]
Catalyst(config-if)# spanning-tree bpduguard enable
```

```
Catalyst(config)# interface range fa0/1 - 24
Catalyst(config-if)# spanning-tree bpduguard enable
```

SMBスイッチでのBPDUガードの有効化：CLI

```
SMBswitch(config)# interface [range(optional)] [port-id]
SMBswitch(config-if)# spanning-tree bpduguard enable
```

```
SMBswitch(config)# interface range fa0/1 - 24
SMBswitch(config-if)# spanning-tree bpduguard enable
```

SMBスイッチでのBPDUガードの有効化 – GUI

ステップ1:Web設定ユーティリティにログインし、[スパンニングツリー]>[STPインターフェイス設定]を選択します。[STP Interface Settings]ページが開きます。

ステップ2:[インターフェイスタイプ]ドロップダウンリストから、編集するインターフェイスのタイプを選択します。

ステップ3:[Go]をクリックし、ページにポートまたはLAGのみを表示します。

ステップ4：他のスイッチに接続されているポートまたはLAGのオプションボタンをクリックし、[Edit]をクリックします。[Edit STP Interface]ウィンドウが表示されます。

ステップ5:[Interface]フィールドで、目的のインターフェイスタイプに対応する[BPDU Guard Enable]チェックボックスをオンにします。

ベストプラクティス#5:VLANをIST(MST0)ではなくMSTIにマッピングする

ポートは適切な役割を認識し、インスタンスのマッピングに進みます。最適な結果を得るには、作成するインスタンスの量を制限します。ニュアンスが存在することに注意してください。これはベストプラクティスに反し、エンジニアをソリューションとしてMSTPから除外する可能性があります。複数のインスタンスに対して有効なネットワーク設計の考慮事項がある場合がありますが、ベストプラクティスは1つのインスタンスを使用することであることを注意してください。インスタンスにマッピングするVLANを決定します。次に、ネットワーク内のすべてのスイッチに共通する設定名とリビジョン番号を選択します。

注：MSTI VLANマッピングを編集すると、MSTPが再起動します。

CatalystスイッチでのVLANのマッピング – CLI

```
Catalyst(config)# spanning-tree mst configuration
Catalyst(config-mst)# instance [instance-id] vlan [vlan-range]
```

```
Catalyst(config)# spanning-tree mst configuration
Catalyst(config-mst)# instance 1 vlan 1-11
```

SMBスイッチでのVLANのマッピング – CLI

```
SMBswitch(config)# spanning-tree mst configuration
SMBswitch(config-mst)# instance [instance-id] vlan [vlan-range]
```

```
SMBswitch(config)# spanning-tree mst configuration
SMBswitch(config-mst)# instance 1 vlan 1-11
```

VLANのMSTIへのマッピング：GUI

ステップ1:[スパンニングツリー]>[VLANからMSTPインスタンス]をクリックします。

[VLANからMSTPインスタンス]ページには、次のフィールドがあります。

- *MST*インスタンスID：すべてのMSTPインスタンスが表示されます。
- *VLAN:MST*インスタンスに属するすべてのVLANが表示されます。

ステップ2:MSTPインスタンスにVLANを追加するには、MSTPインスタンスを選択して、[Edit]をクリックします。

- [MSTPインスタンスID]:MSTPインスタンスを選択します。
- *VLAN*：このMSTPインスタンスにマッピングされるVLANを定義します。
- *アクション*：VLANをMSTPインスタンスに追加（マッピング）するか、削除するかを定義します。

ステップ3：パラメータを入力します。

ステップ4:[Apply]をクリックします。この時点で、MSTP VLANマッピングが確立されます。

ベストプラクティス#6：すべてのMSTP対応スイッチを同じ領域

内に配置する

ベストプラクティスは、できるだけ多くのスイッチを1つの領域に配置することです。ネットワークを複数の地域にセグメント化する利点はありません。他のルーティングプロトコルと同様に、プロトコルへのメンバーシップを確認する方法が必要です。送信されたBPDUにより、スイッチは自身を特定の領域のメンバとして認識できます。ブリッジが特定のリージョンのメンバーシップを理解するには、次の設定を共有する必要があります。

1. 地域名
2. リビジョン番号
3. VLANからインスタンスへのマッピングから計算されたダイジェスト

Catalystスイッチのリージョン内のブリッジのホーム帰還 – CLI

```
Catalyst(config)# spanning-tree mst [instance-id] root primary
```

```
Catalyst(config)# spanning-tree mst 5 root primary
```

SMBスイッチのリージョン内のブリッジのホーム帰還 – CLI

```
SMBswitch(config)# spanning-tree mst configuration  
SMBswitch(config-mst)# instance [instance-id] vlan [vlan-range]  
SMBswitch(config-mst)# name [region-name]  
SMBswitch(config-mst)# revision [revision-id]
```

```
SMBswitch(config)# spanning-tree mst configuration  
SMBswitch(config-mst)# instance 1 vlan 10-20  
SMBswitch(config-mst)# name region1  
SMBswitch(config-mst)# revision 1
```

SMBスイッチのリージョン内のブリッジのホーム帰還 – GUI

MSTPの[Properties]ページは、スイッチが存在する領域を定義するために使用されます。デバイスが同じリージョンにある場合は、同じリージョン名とリビジョン値が必要です。

ステップ1: メニューから[Spanning Tree] > [MSTP Properties]を選択します。

ステップ2:[リージョン名]フィールドにMSTPリージョンの名前を入力します。領域名は、ネットワークの論理境界を定義します。MSTPリージョン内のすべてのスイッチは、同じ設定済みリージョン名を持っている必要があります。

ステップ3: 「改訂」フィールドに改訂番号を入力します。これは、MSTP設定のリビジョンを示す論理番号です。MSTPリージョン内のすべてのスイッチのリビジョン番号は同じである必要があります。

ステップ4:[Max Hops]フィールドにホップの最大数を入力します。Max Hopsは、ホップカウントにおけるBPDUのライフタイムを指定します。ブリッジはBPDUを受信すると、ホップカウントを1ずつ減らし、新しいホップカウントでBPDUを再送信します。ブリッジがホップカウントがゼロのBPDUを受信すると、BPDUは廃棄されます。

注: [IST Active]フィールドには、リージョンのアクティブスイッチのブリッジプライオリティとMACアドレスが表示されます。詳細については、[用語集を参照してください](#)。

ステップ5:[Apply]をクリックします。

ベストプラクティス#7 – プライマリMST領域内にCISTのルートブリッジをネストする

このベストプラクティスは、移行全体を同時に保持するためのリンチの一部です。これは、MSTPトポロジのルートブリッジをプライマリMSTPリージョン内に配置することを目的としています。すべてのVLANを同じ領域内に配置するというベストプラクティスを踏まえ、ルート選択はすべてのVLANに対して有効です。これは、ルートガードと呼ばれる機能によって実現されます。ルートガードは、ユーザが作成したルート配置を適用します。ルートガードがアクティブなポートで上位のBPDUをブリッジが受信すると、そのポートはroot-inconsistent STPステートを介してただちにリスニングモードになります。これにより、不良BPDUの転送が防止され、その結果、リージョンのルートブリッジの指定ポートが保持されます。これにより、リージョンのルートブリッジの指定ポートが保持されます。

注：各インスタンスのルートとバックアップルートを慎重に選択します。

CatalystスイッチのCISTへのルートブリッジの配置：CLI

```
Catalyst(config)# spanning-tree mst [instance-id] root {primary | secondary} [diameter dia  
[hello-time hello-time]
```

```
Catalyst(config)# spanning-tree mst 1 root primary 7
```

トラブルシューティング：Catalyst

次のコマンドが返されます。次のコマンドは、不整合とマークされたポートを返します。ただし、このコマンドはSMBスイッチでは使用できません。

```
Catalyst# show spanning-tree inconsistentports
```

SMBスイッチのCISTへのルートブリッジの配置：CLI

```
SMBswitch(config)# interface [interface-id]  
SMBswitch(config-if)# spanning-tree guard root
```

```
SMBswitch(config)# interface gi1/1/1  
SMBswitch(config-if)# spanning-tree guard root
```

SMBスイッチのCISTへのルートブリッジの配置：GUI

ステップ1: Web設定ユーティリティにログインし、[スパンニングツリー] > [STPインターフェイス設定]を選択します。

ステップ2: [インターフェイスタイプ] ドロップダウンリストからインターフェイスを選択します。

ステップ3: [Go] をクリックして、インターフェイス上のポートまたはLAGのリストを表示します。

ステップ4: 変更するポートまたはLAGのラジオボタンをクリックし、[Edit] をクリックします。[Edit STP Interface Setting] ウィンドウが表示されます。

ステップ5: [Interface] フィールドで、目的のインターフェイスに対応するオプションボタンをクリックします。

- [Port]: [Port] ドロップダウンリストから、設定するポートを選択します。これは、選択した1つのポートにのみ影響します。
- LAG: [LAG] ドロップダウンリストから、設定するLAGを選択します。これは、LAG設定で定義されたポートのグループに影響します。

ステップ6：インターフェイスでSTPを有効にするには、[STP]フィールドで[Enable]にチェックマークが入っていることを確認します。

ステップ7:[Root Guard]フィールドの[Enable]をオンにして、インターフェイスでRoot Guardを有効にします。このオプションは、ネットワーク内のルートブリッジの配置を強制する方法を提供します。ルートガードは、新しく接続されたデバイスがルートブリッジとして引き継がれることを防止するために使用されます。

移行の検証 – これはオンですか？

この時点で、MSTPの実装とネットワークはパーリングしている必要があります。信頼はあるが検証の群集では、フレームキャプチャを実行してMSTPステータスを確認できます。次に、結果を予想されるドキュメントと比較します。

Wiresharkを介してパケットキャプチャを実行すると、インスタンスIDを含むMrecordsが表示されます。次に、拡張の前のMrecordのスクリーンショットを示します。

```
▼ Spanning Tree Protocol
  Protocol Identifier: Spanning Tree Protocol (0x0000)
  Protocol Version Identifier: Multiple Spanning Tree (3)
  BPDU Type: Rapid/Multiple Spanning Tree (0x02)
  ▶ BPDU flags: 0x7c, Agreement, Forwarding, Learning, Port Role: Designated
  ▶ Root Identifier: 24576 / 0 / 24:e9:b3:78:fe:80
  Root Path Cost: 0
  ▶ Bridge Identifier: 24576 / 0 / 24:e9:b3:78:fe:80
  Port identifier: 0x8018
  Message Age: 0
  Max Age: 20
  Hello Time: 2
  Forward Delay: 15
  Version 1 Length: 0
  Version 3 Length: 96
  ▼ MST Extension
    MST Config ID format selector: 0
    MST Config name: Cisco
    MST Config revision: 1
    MST Config digest: 2a5477095c475f337a69c797b32cd60a
    CIST Internal Root Path Cost: 0
    ▶ CIST Bridge Identifier: 24576 / 0 / 24:e9:b3:78:fe:80
    CIST Remaining hops: 20
    ▶ MSTID 1, Regional Root Identifier 24576 / 24:e9:b3:78:fe:80
    ▶ MSTID 2, Regional Root Identifier 24576 / 24:e9:b3:79:06:00
```

MRecords

Mrecordの拡張MSTPに関する詳細なデータを表示できます。以下を含む：

- ポートのロール
- MST ID
- 地域ルート
- 内部パスコスト
- ブリッジIDプライオリティ
- ポートIDプライオリティ
- 残りのホップ

```
▼ MSTID 1, Regional Root Identifier 24576 / 24:e9:b3:78:fe:80
  ▶ MSTI flags: 0x7c, Agreement, Forwarding, Learning, Port Role: Designated
  0110 .... = Priority: 0x6
  .... 0000 0000 0001 = MSTID: 1
  Regional Root: Cisco_78:fe:80 (24:e9:b3:78:fe:80)
  Internal root path cost: 0
  Bridge Identifier Priority: 6
```

クイック検証コマンド : SMB CLI

コマンドラインから確認する場合は、次のコマンドを試してください。

```
SMBswitch# show spanning-tree mst-configuration

SMBswitch(config)# spanning-tree mst-configuration
SMBswitch(config-mst)# show pending
```

```
MST
[region1]
Revision 1
2
Vlan
--
0 1-9,21-4094
1 10 20
-
```

```
SMBswitch# show spanning-tree mst-configuration
[]
Revision 0
VLAN
--
0 1-4094
-
```

注 : showコマンドのCatalystバージョンでは、mstと設定の間の - は除外されます。例 : 「show spanning-tree mst configuration」

同じネットワーク上に存在するPVST+とMSTPについて知っておくべきこと

PVST+を実行するレガシースイッチのサポートを継続する必要がある場合は、ポート単位で処理します。これらのスイッチのいずれかがVLANトランクとして動作している場合は、MSTPスイッチがトランクに割り当てられたすべてのVLANのルートであることを確認します。さらに、MSTPはPVST+ BPDUのデコードを試みますが、このシミュレーションは不完全です。境界ポートの概念を詳しく説明する必要があります。

MSTP境界ポートの役割と状態は、外部トポロジと相互作用する内部スパニングツリーによって決まります。これは、ポートがISTでブロッキングモードの場合、MSTPのすべてのインスタンスでブロッキングモードであることを意味します。この影響はPVST+の実装に移り、VLANの機能に影響を与えます。ポートがフォワーディング、ラーニングなどを行っている場合も同様です。ご想像の通り、これは問題になる可能性があります。これは、あるVLANに対して転送する必要があるポートが、別のVLANのニーズのためにブロックされているのに対して、問題が解決しない可能性があります。PVST+シミュレーションでは、ISTからの情報を利用してVLAN BPDUごとに作成します。これにより、ネットワーク全体で、MSTP領域がすべてのVLANに対して1つのスイッチとして認識されるという「幻想」が生じます。スイッチをスタックする方法と同じようには、半不良ではありません。境界ポートの位置からは、シミュレートされた各VLANに対して個別のBPDUを送信する必要が生じることが問題です。BPDU間に不整合があると、シミュレーション全体がエラーで破棄される可能性があります。一貫性のあるBPDUを受信するだけで、シミュレーション自体がバックアップされます。

結論として、この状況は、境界ポートで受信されるBPDUが同一である必要がある理由です。[このトピックに関するその他の情報については、このコミュニティスレッドを参照してください。](#)

ネットワークハードウェアに関する知識はありますか。完全にシスコじゃないのか？

MSTPは下位互換性があります。シスコ以外のハードウェアが高速スパニングツリーをサポートしている限り、問題ありません。問題が発生した場合は、[スイッチングコミュニティに問い合わせてください](#)。

結論

このガイドをご覧くださいありがとうございます。これらのベストプラクティスを使用して、レイヤ2ネットワークのパフォーマンスを向上させる必要があります。

注目に値する、スパニングツリーはユーザにとってスリリングとは言えないかもしれませんが、ロードシェアリングの利点は、ネットワークの効率性を維持するための努力に値します。スパニングツリーの創造者であるRadia Perlmanは、母親が今まで愛していたことと同じくらい大好きです。彼女はそれについて詩をさえた。