



## CHAPTER 2

# CFS の設定

---

この章では、コンフィギュレーションの変更を含むデータをネットワークのすべての Cisco NX-OS デバイスに配信するシスコ独自の機能、Cisco Fabric Services (CFS) の使い方を説明します。

ここでは、次の内容を説明します。

- 「CFS の概要」 (P.2-1)
- 「CFS のライセンス要件」 (P.2-4)
- 「CFS の前提条件」 (P.2-4)
- 「注意事項および制約事項」 (P.2-4)
- 「CFS 配信の設定」 (P.2-5)
- 「CFS コンフィギュレーションの確認」 (P.2-23)
- 「デフォルト設定」 (P.2-23)
- 「その他の関連資料」 (P.2-24)
- 「CFS 機能の履歴」 (P.2-24)

## CFS の概要

CFS over IP (CFS over IP; IP を介した CFS) を使用して、1 台のシスコ デバイスまたはネットワークの他のすべてのシスコ デバイスにコンフィギュレーションを配信し、同期させることができます。CFS over IP を使用すると、ネットワークの設定と動作に一貫性をもたせ、ほとんどの場合、同じ設定と動作を維持できます。

ここでは、次の内容について説明します。

- 「アプリケーションデータベースのマージ」 (P.2-2)
- 「CFS を使用してコンフィギュレーションの変更を配信するアプリケーション」 (P.2-2)
- 「CFS の配信」 (P.2-2)
- 「ハイ アベイラビリティ」 (P.2-3)
- 「仮想化サポート」 (P.2-4)

## アプリケーション データベースのマージ

ネットワークで新しいデバイスが検出されると、CFS は他のデバイスの設定とのマージ、つまり同期を管理します。また、リージョンごとおよびアプリケーションごとに 1 台のデバイスでマージを管理するよう指定して、マージ数を調整し、最小限に抑えます。他のデバイスは、マージプロセスには関与しません。

2 つのネットワークのマージ中、指定されたマネージャどうしでコンフィギュレーション データベースが交換されます。一方のアプリケーションによりデータベースがマージされ、マージが成功したかどうか判断され、他のすべてのデバイスに通知されます。

マージが成功すると、マージされたデータベースは統合されたファブリックの全デバイスに配信され、新しいファブリック全体が一貫性のある状態で表示されます。マージが失敗した場合は、新しいファブリックのいずれかのデバイスから配信を開始して回復できます。この配信により、ファブリック内のすべてのピアは同じコンフィギュレーション データベースに復元されます。

## CFS を使用してコンフィギュレーションの変更を配信するアプリケーション

CFS は、表 2-2 に示すアプリケーションのコンフィギュレーションの変更を配信します。

表 2-1 CFS がサポートするアプリケーション

アプリケーション	デフォルトの状態
RADIUS	ディセーブル
TACACS+	ディセーブル
ユーザおよび管理者のロール	ディセーブル
Call Home	ディセーブル

## CFS の配信

CFS は、定義されたリージョンまたはネットワーク全体に存在する複数のデバイスにコンフィギュレーションの変更を配信します。

次に、CFS がアプリケーションのコンフィギュレーションを配信する手順の概要を示します。

1. CFS をイネーブルにして、Call Home などのアプリケーションのコンフィギュレーションを配信できるようにします。
2. Call Home などの CFS アプリケーションのコンフィギュレーションを変更するコマンドを入力します。
3. CFS は、アクティブなファブリック ロックをチェックして、このアプリケーションに対してコンフィギュレーションの変更がすでに進行していないかどうかを確認します。



(注) 1 つのアプリケーションに対してアクティブにできる CFS セッションは 1 に限られます。CFS は、ロックを使用してこの制限を強制します。ファブリック内の任意の場所でアプリケーションに対してロックが適用されている場合、配信を開始できません。

4. 次のどちらかが発生します。
  - このアプリケーションに対してアクティブなファブリック ロックが存在する場合、CFS はコマンドを拒否します。既存のファブリック ロックが解放されるまで、変更は許可されません。
  - このアプリケーションに対してアクティブなファブリック ロックがない場合、CFS はセッションを開始し、このアプリケーションに対してファブリックをロックします。
5. アプリケーションの残りのコンフィギュレーション コマンドを入力します。
6. コンフィギュレーションは、**commit** コマンドを使用してコミットします。
7. CFS はコンフィギュレーションを配信し、ロックを解放します。

## CFS リージョン

CFS リージョンとは、ある機能またはアプリケーションに対してユーザが定義したデバイスのサブセットです。通常、互いに近くに存在するデバイスに基づいて配信をローカライズまたは制限する場合にリージョンを定義します。

多数の地域で別々の管理者がデバイスのサブセットを担当するネットワークの場合、CFS リージョンを設定して、アプリケーションの物理的な範囲を管理できます。

CFS リージョンは数字の 0 ~ 200 で識別されます。リージョン 0 はデフォルトリージョンです。リージョン 1 ~ 200 を設定できます。



(注)

機能を移動して、新しいリージョンに割り当てる場合、その範囲はそのリージョンに制限され、他のすべてのリージョンは配信やマージの対象外となります。

複数の機能について CFS リージョンを設定し、コンフィギュレーションを配信できます。ただし、1 台のデバイスで、特定の機能のコンフィギュレーションを配信するために設定できる CFS リージョンは一度に 1 つに限られます。CFS リージョンに 1 つの機能を割り当てると、そのコンフィギュレーションは、他の CFS リージョン内で配信できなくなります。



(注)

デフォルトリージョンはファブリック内のすべてのデバイスに変更を配信するために使用されます。リージョン 0 はデフォルトリージョンとして予約され、ファブリック内のすべてのデバイスが含まれます。リージョンからアプリケーションを削除し、それを別のリージョンに割り当てない場合、アプリケーションはデフォルトリージョン（リージョン 0）に追加されます。

## ハイ アベイラビリティ

CFS のステートレス リスタートがサポートされています。リブート後またはスーパーバイザ スイッチオーバー後に、実行コンフィギュレーションが適用されます。ハイ アベイラビリティの詳細については、『Cisco Nexus 7000 Series NX-OS High Availability and Redundancy Guide, Release 4.1』を参照してください。

## 仮想化サポート

CFS は、VDC ごとに設定されます。

Cisco NX-OS にアクセスすると、別の VDC を指定しない限り、デフォルト VDC が使用されます。VDC の詳細については、次の URL にアクセスして、『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide, Release 4.1』を参照してください。

[http://www.cisco.com/en/US/docs/switches/datacenter/sw/4\\_1/nx-os/virtual\\_device\\_context/configuration/guide/vdc\\_nx-os\\_book.html](http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_1/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html)

## CFS のライセンス要件

CFS にはライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『Cisco Nexus 7000 Series NX-OS Licensing Guide, Release 4.1』を参照してください。

## CFS の前提条件

CFS の前提条件は、次のとおりです。

- CFS はデフォルトでイネーブルです。ファブリック内のすべてのデバイスで CFS をイネーブルにしておかないと配信は受信されません。
- アプリケーションに対して CFS がディセーブルになっていると、そのアプリケーションからコンフィギュレーションは配信されず、ファブリック内の他のデバイスからの配信も受け取られません。

## 注意事項および制約事項

CFS に関するコンフィギュレーションの注意事項および制約事項は、次のとおりです。

- CFS は、同じデバイスから IPv4 と IPv6 両方を介して配信できません。
- デバイスで Virtual Port Channel (vPC; 仮想ポート チャネル) 機能がイネーブルになっている場合、イーサネットを介して CFS をディセーブルにしないでください。



**注意**

vPC 機能を動作させるには、イーサネットを介した CFS をイネーブルにする必要があります。

- アプリケーションデータの CFS 配信では、ダイレクト ユニキャストを使用します。
- 同様のマルチキャスト アドレスをもつ IP を介した CFS 対応デバイスはすべて、1 つの IP を介した CFS ファブリックを形成します。
- 設定するアプリケーションについて CFS がイネーブルになっていることを確認してください。詳細については、「アプリケーションの CFS 配信のイネーブル」の手順 (P.2-5) を参照してください。
- ファブリックをロックすると、リスタートとスイッチオーバーにわたってユーザ名が記憶されません。

- ファブリックをロックすると、他のユーザによるコンフィギュレーションの変更の試みは拒否されます。
- ファブリックのロック中、アプリケーションでは実行コンフィギュレーションではなく保留データベースまたは一時ストレージにコンフィギュレーションの変更の作業コピーが保持されます。
- コミットされていないコンフィギュレーションの変更（作業コピーとして保存）は実行コンフィギュレーションには含まれないため、**show** コマンドの出力にも表示されません。
- 変更をコミットすると、実行コンフィギュレーションが作業コピーによって上書きされます。
- ファブリック ロックが必要な CFS セッションを開始し、セッションを終了するのを忘れた場合は、管理者がセッションをクリアできます。詳細については、「[ロックされたセッションのクリア](#)」の手順 (P.2-19) を参照してください。
- 以前コンフィギュレーションの変更が行われていない場合は、空のコミットを行えます。この場合、**commit** コマンドにより、ロックを取得し、現在のデータベースを配信するセッションが作成されます。
- **commit** コマンドは、ファブリック ロックが取得されたデバイスだけで使用できます。

## CFS 配信の設定

この項では、CFS の設定方法について説明します。次のトピックで構成されています。

- 「[アプリケーションの CFS 配信のイネーブル](#)」 (P.2-5)
- 「[CFS 配信モードの指定](#)」 (P.2-9)
- 「[IP を介した CFS の IP マルチキャストアドレスの設定](#)」 (P.2-10)
- 「[CFS リージョンの設定](#)」 (P.2-12)
- 「[CFS 設定の作成と配信](#)」 (P.2-18)
- 「[設定の破棄](#)」 (P.2-20)
- 「[CFS 配信のグローバルなディセーブル](#)」 (P.2-21)

## アプリケーションの CFS 配信のイネーブル

ここでは、次の内容について説明します。

- 「[CFS をイネーブルにして Call Home コンフィギュレーションを配信する](#)」 (P.2-5)
- 「[CFS をイネーブルにして RADIUS コンフィギュレーションを配信する](#)」 (P.2-6)
- 「[CFS をイネーブルにして TACACS+ コンフィギュレーションを配信する](#)」 (P.2-7)
- 「[CFS をイネーブルにしてロールのコンフィギュレーションを配信する](#)」 (P.2-8)

## CFS をイネーブルにして Call Home コンフィギュレーションを配信する

CFS をイネーブルにして、Call Home コンフィギュレーションを配信できます。

### 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

## 手順概要

1. **config t**
2. **callhome**
3. **distribute**
4. **copy running-config startup-config**

## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: switch# config t switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# <b>callhome</b>  例: switch(config)# callhome switch(config-callhome)#	グローバル コンフィギュレーションモードを開始します。
ステップ 3	switch(config)# <b>distribute</b>  例: switch(config-callhome)# distribute switch(config-callhome)#	CFS をイネーブルにして、Call Home コンフィギュレーションのアップデートを配信します。
ステップ 4	<b>show application_name status</b>  例: switch(config-callhome)# show callhome status	(任意) 指定されたアプリケーションについて、CFS 配信ステータスを表示します。
ステップ 5	<b>copy running-config startup-config</b>  例: switch(config)# copy running-config startup-config	(任意) リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

CFS をイネーブルにして、Call Home コンフィギュレーションを配信する例を示します。

```
switch(config)# callhome
switch(config-callhome)# distribute
switch(config-callhome)# show callhome status
Distribution : Enabled
switch(config-callhome)# copy running-config startup-config
[#####] 100%
```

## CFS をイネーブルにして RADIUS コンフィギュレーションを配信する

CFS をイネーブルにして、RADIUS コンフィギュレーションを配信できます。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

## 手順概要

1. `config t`
2. `radius distribute`
3. `show radius status`
4. `copy running-config startup-config`

## 手順詳細

	コマンド	目的
ステップ 1	<code>config t</code>  例: <code>switch# config t</code> <code>switch(config)#</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>switch(config)# radius distribute</code>  例: <code>switch(config)# radius distribute</code>	指定されたアプリケーションについて、デバイスが CFS から配信されたコンフィギュレーションのアップデートを受け取れるようにします。
ステップ 3	<code>show radius status</code>  例: <code>switch(config)# show radius status</code>	(任意) 指定されたアプリケーションについて、CFS 配信ステータスを表示します。
ステップ 4	<code>copy running-config startup-config</code>  例: <code>switch(config)# copy running-config startup-config</code>	(任意) リポートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

CFS をイネーブルにして、RADIUS コンフィギュレーションを配信する例を示します。

```
switch(config)# radius distribute
switch(config)# show radius status
Distribution : Enabled
switch(config)# copy running-config startup-config
[#####] 100%
```

## CFS をイネーブルにして TACACS+ コンフィギュレーションを配信する

CFS をイネーブルにして、TACACS+ コンフィギュレーションを配信できます。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は `switchto vdc` コマンドを使用します。

## 手順概要

1. `config t`
2. `tacacs+ distribute`
3. `show tacacs+ status`
4. `copy running-config startup-config`

## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: switch# config t switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# <b>tacacs+ distribute</b>  例: switch(config)# tacacs+ distribute	CFS をイネーブルにして、TACACS+ のコンフィギュレーションのアップデートを配信します。
ステップ 3	<b>show tacacs+ status</b>  例: switch(config)# show tacacs+ status	(任意) TACACS+ の CFS 配信ステータスを表示します。
ステップ 4	<b>copy running-config startup-config</b>  例: switch(config)# copy running-config startup-config	(任意) リポートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

CFS をイネーブルにして、TACACS+ コンフィギュレーションを配信する例を示します。

```
switch(config)# callhome distribute
switch(config)# show callhome status
Distribution : Enabled
Last operational state: No session
switch(config)# copy running-config startup-config
[#####] 100%
```

## CFS をイネーブルにしてロールのコンフィギュレーションを配信する

CFS をイネーブルにして、ロールのコンフィギュレーションを配信できます。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

## 手順概要

1. **config t**
2. **role distribute**
3. **show role status**
4. **copy running-config startup-config**



## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: switch# config t switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# <b>role distribute</b>  例: switch(config)# role distribute	CFS をイネーブルにしてロールのコンフィギュレーションを配信します。
ステップ 3	<b>show role status</b>  例: switch(config)# show role status	(任意) CFS 配信ステータスを表示します。
ステップ 4	<b>copy running-config startup-config</b>  例: switch(config)# copy running-config startup-config	(任意) リポートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

CFS をイネーブルにして、Call Home コンフィギュレーションを配信する例を示します。

```
switch(config)# role distribute
switch(config)# show role status
Distribution : Enabled
switch(config)# copy running-config startup-config
[#####] 100%
```

## CFS 配信モードの指定

CFS 配信モード（イーサネット、IPv4 または IPv6）を指定し、イネーブルにできます。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

## 手順概要

1. **config t**
2. **cfs [eth | ipv4 | ipv6] distribute**
3. **show cfs status**
4. **copy running-config startup-config**

## 手順詳細

	コマンド	目的
ステップ 1	<pre>config t</pre> <p>例:</p> <pre>switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<pre>cfs [eth   ipv4   ipv6] distribute</pre> <p>例:</p> <pre>switch(config)# cfs ipv4 distribute switch(config)#</pre>	<p>デバイス上の次のアプリケーションの 1 つまたはすべてに対して CFS 配信をグローバルにイネーブルにします。</p> <ul style="list-style-type: none"> <li>イーサネット</li> <li>IPv4</li> <li>IPv6</li> </ul> <p>この例では、CFS 配信は、IPv4 を介してイネーブルになります。</p>
ステップ 3	<pre>show cfs status</pre> <p>例:</p> <pre>switch(config)# show cfs status Distribution : Enabled Distribution over IP : Enabled - mode IPv4 IPv4 multicast address : 239.255.70.83 IPv6 multicast address : ff15::efff:4653 switch(config)#</pre>	<p>配信モードを含む CFS の現在の状態を表示します。</p> <p>この例では、CFS は IPv4 から配信されることが示されています。</p>
ステップ 4	<pre>copy running-config startup-config</pre> <p>例:</p> <pre>switch(config)# copy running-config startup-config</pre>	(任意) リポートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

## IP を介した CFS の IP マルチキャスト アドレスの設定

ネットワーク トポロジの変更を検出するキープアライブ メカニズムなど、CFS プロトコル固有の配信では、IP マルチキャスト アドレスを使用して情報を送受信します。

次のいずれかについて、IP を介した CFS の配信に使用する IP マルチキャスト アドレスを設定できます。

- IPv4 : デフォルトの IPv4 マルチキャスト アドレスは 239.255.70.83 です。
- IPv6 : デフォルトの IPv6 マルチキャスト アドレスは ff13:7743:4653 です。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。マルチキャスト アドレスを変更する前に、CFS IP 配信をディセーブルにする必要があります。

## 手順概要

1. `config t`
2. `no cfs [ipv4 | ipv6] distribute`
3. `cfs [ipv4 | ipv6] mcast-address ip_address`
4. `show cfs status`
5. `copy running-config startup-config`

## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: <pre>switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>no cfs [ipv4   ipv6] distribute</b>  例: <pre>switch(config)# no cfs ipv4 distribute This will prevent CFS from distributing over IPv4 network. Are you sure? (y/n) [n] y switch(config)#</pre>	デバイス上のすべてのアプリケーションに対して IP を介した CFS 配信をグローバルにディセーブルにします。  <b>(注)</b> マルチキャスト アドレスを変更する前に、IP を介した CFS をディセーブルにする必要があります。
ステップ 3	<b>cfs [ipv4   ipv6] mcast-address ip_address</b>  例: <pre>switch(config)# cfs ipv4 mcast-address 239.255.1.1 Distribution over this IP type will be affected Change multicast address for CFS-IP ? Are you sure? (y/n) [n] y</pre>	次の 1 つを使用した CFS 配信のマルチキャスト アドレスを設定します。 <ul style="list-style-type: none"> <li>• <b>IPv4</b>                有効な IPv4 アドレスの範囲は 239.255.0.0 ~ 239.255.255.255 と 239.192/16 ~ 239.251/16 です。                デフォルトの IPv4 アドレスは 239.255.70.83 です。</li> <li>• <b>IPv6</b>                有効な IPv6 アドレスの範囲は ff15::/16 (ff15::0000:0000 ~ ff15::ffff:ffff) ff18::/16 (ff18::0000:0000 ~ ff18::ffff:ffff) です。                デフォルトの IPv6 アドレスは ff13:7743:4653 です。</li> </ul>

	コマンド	目的
ステップ 4	<b>show cfs status</b>  例: <pre>switch(config)# show cfs status Distribution : Enabled Distribution over IP : Enabled - mode IPv4 IPv4 multicast address : 239.255.1.1 IPv6 multicast address : ff15::efff:4653 switch(config)#</pre>	イネーブルかどうか、IP モード、マルチキャストアドレスなど、CFS の現在の状態を表示します。  この例では、CFS は 239.255.1.1 で IPv4 を介して配信されることが示されています。
ステップ 5	<b>copy running-config startup-config</b>  例: <pre>switch(config)# copy running-config startup-config</pre>	(任意) リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

## CFS リージョンの設定

この項では、CFS リージョンの作成および設定方法について説明します。次のトピックで構成されています。

- 「CFS リージョンの作成」 (P.2-12)
- 「異なるリージョンへのアプリケーションの移動」 (P.2-13)
- 「リージョンからのアプリケーションの削除」 (P.2-15)
- 「CFS リージョンの削除」 (P.2-16)

## CFS リージョンの作成

CFS リージョンを作成し、Call Home などのアプリケーションを追加できます。

### 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

### 手順概要

1. **config t**
2. **cfs region *region\_number***
3. ***application\_name***
4. **show cfs region brief**
5. **copy running-config startup-config**

## 手順詳細

	コマンド	目的
ステップ 1	<code>config t</code>  例: switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>cfs region region_number</code>  例: switch(config)# cfs region 4 switch(config-cfs-region)#	リージョンを作成し、指定されたリージョンのコンフィギュレーション モードを開始します。 この例では、リージョン 4 が作成されます。
ステップ 3	<code>application_name</code>  例: switch(config-cfs-region)# callhome switch(config-cfs-region)#	指定されたリージョンに、指定されたアプリケーションを追加します。
ステップ 4	<code>show cfs region brief</code>  例: switch(config-cfs-region)# show cfs region brief  ----- Region            Application    Enabled ----- 4            callhome        yes  switch(config-cfs-region)#	(任意) 設定されたすべてのリージョンとアプリケーションを表示します (ピアは表示されません)。 この例では、リージョン 4 に Call Home アプリケーションが表示されます。
ステップ 5	<code>copy running-config startup-config</code>  例: switch(config)# copy running-config startup-config	(任意) リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

## 異なるリージョンへのアプリケーションの移動

アプリケーションを異なるリージョンに移動できます。たとえば、NTP をリージョン 1 からリージョン 2 に移動できます。



(注)

アプリケーションを移動すると、その範囲は新しいリージョンに制限され、他のすべてのリージョンは配信やマージの対象外となります。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は `switchto vdc` コマンドを使用します。

## 手順概要

1. **config t**
2. **cfs region region\_number**
3. **application\_name**
4. **show cfs region**

## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>cfs region region_number</b>  例: switch(config)# cfs region 2 switch(config-cfs-region)#	ターゲット リージョンと宛先リージョンでコンフィギュレーション モードを開始します。
ステップ 3	<b>application_name</b>  例: switch(config-cfs-region)# callhome switch(config-cfs-region)# radius	移動するアプリケーションを指定します。  この例では、リージョン 2 に Call Home アプリケーションが移されます。
ステップ 4	<b>show cfs region name application_name</b>  例: switch(config-cfs-region)# show cfs region name callhome	指定されたアプリケーションのピアおよびリージョン情報を表示します。
ステップ 5	<b>copy running-config startup-config</b>  例: switch(config)# copy running-config startup-config	(任意) リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Call Home アプリケーションを CFS リージョン 2 に移動する例を示します。

```
switch# config t
switch(config)# cfs region 2
switch(config-cfs-region)# callhome
switch(config-cfs-region)# show cfs region name callhome

Region-ID : 2
Application: callhome
Scope : Physical-fc-ip
-----
Switch WWN          IP Address
-----
20:00:00:22:55:79:a4:c1 172.28.230.85          [Local]
                        switch

Total number of entries = 1

switch(config-cfs-region)#
```

## リージョンからのアプリケーションの削除

アプリケーションをリージョンから削除できます。アプリケーションをリージョンから削除することは、アプリケーションをデフォルトリージョンに戻すことと同じです。デフォルトリージョンは通常リージョン 0 です。これにより、ファブリック全体がアプリケーションの配信範囲になります。

### 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

### 手順概要

1. **config t**
2. **cfs region region\_number**
3. **no application\_name**
4. このリージョンから削除するアプリケーションごとにステップ 3 を繰り返します。
5. **show cfs region brief**
6. **copy running-config startup-config**

### 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>cfs region region_number</b>  例: switch(config)# cfs region 2 switch(config-cfs-region)#	指定されたリージョンでコンフィギュレーション モードを開始します。
ステップ 3	<b>no ntp</b>  例: switch(config-cfs-region)# no ntp	指定されたアプリケーションをリージョンから削除します。
ステップ 4	(任意) このリージョンから削除するアプリケーションごとにステップ 3 を繰り返します。	—

コマンド	目的
<p><b>ステップ 5</b> <code>show cfs region brief</code></p> <p>例:</p> <pre>switch(config-cfs-region)# show cfs region brief</pre> <pre>----- Region          Application  Enabled ----- 4               tacacs+     yes 6               radius      yes</pre> <pre>switch(config-cfs-region)#</pre>	<p>設定されたすべてのリージョンとアプリケーションを表示します（ピアは表示されません）。</p>
<p><b>ステップ 6</b> <code>copy running-config startup-config</code></p> <p>例:</p> <pre>switch(config)# copy running-config startup-config</pre>	<p>(任意) リポートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。</p>

## CFS リージョンの削除

リージョンを削除し、リージョン内のすべてのアプリケーションをデフォルト リージョンに戻すことができます。

### 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は `switchto vdc` コマンドを使用します。

### 手順概要

1. `config t`
2. `no cfs region region_number`
3. `show cfs region brief`
4. `show cfs application name application-name`
5. `copy running-config startup-config`



## 手順詳細

	コマンド	目的
ステップ1	<b>config t</b>  例: <pre>switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ2	<b>no cfs region region_number</b>  例: <pre>switch(config)# no cfs region 4 WARNING: All applications in the region will be moved to default region. Are you sure? (y/n) [n] switch(config)#</pre>	この操作によってリージョン内のすべてのアプリケーションがデフォルトリージョンに移されることを伝える警告が表示されたあと、指定されたリージョンが削除されます。  リージョンの削除後、グローバル コンフィギュレーション モードに戻ります。
ステップ3	<b>show cfs region brief</b>  例: <pre>switch(config)# show cfs region brief</pre> <pre>----- Region      Application  Enabled ----- 6           radius      no switch(config)#</pre>	設定されたすべてのリージョンとアプリケーションを表示します (ピアは表示されません)。  この例では、リージョン4がありません。
ステップ4	<b>show cfs application name application-name</b>  例: <pre>switch# show cfs application name callhome</pre> <pre>Enabled      : Yes Timeout      : 20s Merge Capable : Yes Scope        : Physical-fc-ip Region       : Default switch#</pre>	ローカル アプリケーション情報を名前別に表示します。  この例では、Call Home アプリケーションがデフォルトリージョンに属していることが示されています。
ステップ5	<b>copy running-config startup-config</b>  例: <pre>switch(config)# copy running-config startup-config</pre>	(任意) リポートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

## CFS 設定の作成と配信

アプリケーションのコンフィギュレーションの変更を作成し、その変更をアプリケーション ピアに配信できます。



**注意**

変更をコミットしない場合、変更は配信されず、アプリケーション ピア デバイスの実行コンフィギュレーションに保存されません。



**注意**

配信されたアプリケーション ピア デバイスごとに変更をスタートアップ コンフィギュレーションに保存しない場合、変更は実行コンフィギュレーションだけに保持されます。

### 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

### 手順概要

1. **config t**
2. *application\_name*
3. *application\_command*
4. 作成するコンフィギュレーション コマンドごとにステップ 3 を繰り返します。
5. **show application\_name status**
6. **commit**
7. **copy running-config startup-config fabric**

### 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: switch# config t switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<i>application_name</i>  例: switch(config)# callhome switch(config-callhome)#	CFS は指定されたアプリケーション名のセッションを開始し、ファブリックをロックします。
ステップ 3	<i>application_command</i>  例: switch(config-callhome)# email-contact admin@Mycompany.com	コンフィギュレーションの変更は作業コピーとして保存され、 <b>commit</b> コマンドを入力するまで実行コンフィギュレーションには保存されません。
ステップ 4	(任意) 作成するコンフィギュレーション コマンドごとにステップ 3 を繰り返します。	—

	コマンド	目的
ステップ 5	<b>show application_name status</b>  例: <pre>switch(config-callhome)# show callhome status Distribution : Enabled switch(config-callhome)#</pre>	(任意) 指定されたアプリケーションについて、CFS 配信ステータスを表示します。  この例の出力では、Call Home に対して配信がイネーブルになっていることを示しています。
ステップ 6	<b>commit</b>  例: <pre>switch(config-callhome)# commit</pre>	CFS は、コンフィギュレーションの変更を各アプリケーション ピア デバイスの実行コンフィギュレーションに配信します。  1 台以上の外部デバイスが成功のステータスを報告すると、ソフトウェアは実行コンフィギュレーションを CFS 作業コピーからの変更で上書きし、ファブリック ロックを解放します。  成功のステータスを報告する外部デバイスがない場合、変更は行われず、ファブリック ロックがそのまま適用されます。
ステップ 7	<b>copy running-config startup-config fabric</b>  例: <pre>switch(config)# copy running-config startup-config fabric</pre>	ファブリック内のすべてのデバイスで実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、リポートとリスタートを通して変更を継続的に保存します。

Call Home のコンタクト情報を設定し、配信する例を示します。

```
switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server contact personname@companyname.com
switch(config)# callhome
switch(config-callhome)# email-contact admin@Mycompany.com
switch(config-callhome)# phone-contact +1-800-123-4567
switch(config-callhome)# street-address 123 Anystreet st. Anytown,AnyWhere
switch(config-callhome)# commit
switch(config-callhome)# copy running-config startup-config fabric
[#####] 100%
switch(config-callhome)#
```

## ロックされたセッションのクリア

ファブリック内のデバイスから、アプリケーションによって設定されたロックをクリアできます。ロックを解放するには、Admin 権限が必要です。



### 注意

ファブリックのロックをクリアすると、そのファブリック内のデバイスで保留されているコンフィギュレーションは破棄されます。

### 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

## 手順概要

1. **config t**
2. **show application\_name status**
3. **clear application\_name session**
4. **show application\_name status**

## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: switch# config t switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>show application_name status</b>  switch(config)# show ntp status Distribution : Enabled Last operational state: Fabric Locked switch(config)#	現在のアプリケーションの状態を表示します。  この例では、NTP がロックされていることが示されています。
ステップ 3	<b>clear application_name session</b>  例: switch# clear ntp session switch#	アプリケーション コンフィギュレーション セッションをクリアし、ファブリックのロックをクリアします。  保留中の変更はすべて破棄されます。
ステップ 4	<b>show application_name status</b>  例: switch# show ntp status Distribution : Enabled Last operational state: No session switch#	現在のアプリケーションの状態を表示します。  この例では、NTP アプリケーションからロックが削除されることが示されています。

## 設定の破棄

コンフィギュレーションの変更を破棄して、ロックを解放できます。



## 注意

コンフィギュレーションの変更を破棄すると、保留データベースが削除され、ファブリックのリリースが解放されます。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は **switchto vdc** コマンドを使用します。

## 手順概要

1. `config t`
2. `application_name abort`
3. `show application_name session status`

## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: <pre>switch# config t switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<pre>switch(config)# application_name abort Y</pre> 例: <pre>switch(config)# no cfs distribute This will prevent CFS from distributing the configuration to other switches. Are you sure? (y/n) [n] y</pre>	確認の要求後、アプリケーションの設定を中止します。  この場合、NTP のコンフィギュレーションは中止され、コンフィギュレーションの変更が破棄されて CFS セッションは終了し、ファブリックロックが解放されます。  <b>(注)</b> <code>abort</code> コマンドは、ファブリック ロックを取得したデバイスだけでサポートされます。
ステップ 3	<b>show application_name session status</b>  例: <pre>switch(config)# show ntp session status Last Action Time Stamp      : Wed Nov 12 16:07:25 2008 Last Action                  : Abort Last Action Result           : Success Last Action Failure Reason  : none switch(config)#</pre>	(任意) 指定されたアプリケーションについて、CFS セッションのステータスを表示します。  この例の出力では、CFS セッションが中止されたことが示されています。

## CFS 配信のグローバルなディセーブル

デバイスに対する CFS 配信をディセーブルにして、物理的な接続を維持しながら、CFS を使用してファブリック全体の配信からアプリケーションを分離できます。

デバイスで CFS をグローバルにディセーブルにすると、デバイスに対する CFS の操作が制限され、すべての CFS コマンドは、デバイスが物理的に分離されているかのように機能し続けます。

## 操作の前に

正しい VDC を使用していることを確認します。VDC の変更は `switchto vdc` コマンドを使用します。

## 手順概要

1. `config t`
2. `no cfs distribute`
3. `show cfs status`
4. `copy running-config startup-config`

## 手順詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例: <pre>switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<pre>switch(config)# no cfs distribute</pre> 例: <pre>switch(config)# no cfs distribute This will prevent CFS from distributing the configuration to other switches. Are you sure? (y/n) [n] y switch(config)#</pre>	デバイス上のすべてのアプリケーションに対して CFS 配信をグローバルにディセーブルにします。  <b>(注)</b> vPC 機能をイネーブルにすると、IP 配信だけがディセーブルになります。CFS 配信をディセーブルにする前に、vPC をディセーブルにする必要があります。
ステップ 3	<b>show cfs status</b>  例: <pre>switch(config)# show cfs status Distribution : Enabled Distribution over IP : Disabled IPv4 multicast address : 239.255.70.83 IPv6 multicast address : ff15::efff:4653 Distribution over Ethernet : Disabled switch(config)#</pre>	(任意) デバイスに対するグローバルな CFS 配信ステータス (イネーブルまたはディセーブル) を表示します。
ステップ 4	<b>copy running-config startup-config</b>  例: <pre>switch(config)# copy running-config startup-config</pre>	(任意) リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

## CFS コンフィギュレーションの確認

CFS のコンフィギュレーション情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
<code>show cfs application</code>	現在 CFS がイネーブルになっているアプリケーションを表示します。
<code>show cfs application name</code>	特定のアプリケーションについて、イネーブルまたはディセーブルの状態、CFS で登録されたタイムアウト、CFS でマージサポートが登録されている場合のマージ機能、配信範囲、配信リージョンなどの詳細を表示します。
<code>show application_name session status</code>	最終アクション、結果、失敗した場合はその理由を含む、コンフィギュレーションセッションステータスを表示します。
<code>show cfs internal</code>	メモリの統計情報、イベント履歴など、CFS 内部の情報を表示します。
<code>show cfs lock</code>	すべてのアクティブロックを表示します。
<code>show cfs merge status name</code>	特定のアプリケーションのマージステータスを表示します。
<code>show cfs peers</code>	物理ファブリックのすべてのピアを表示します。
<code>show cfs regions</code>	ピアおよびリージョンの情報とともにすべてのアプリケーションを表示します。
<code>show cfs static</code>	すべてのスタティックピアのステータスを表示します。
<code>show cfs status</code>	デバイスでの CFS 配信のステータスと IP 配信情報を表示します。
<code>show logging level cfs</code>	CFS ログギングのコンフィギュレーションを表示します。
<code>show tech-support cfs</code>	CFS の問題解決時にテクニカルサポートが必要とする CFS コンフィギュレーションに関する情報を表示します。

## デフォルト設定

表 2-2 に、CFS パラメータのデフォルト設定を示します。

表 2-2 デフォルトの CFS パラメータ

パラメータ	デフォルト
デバイスでの CFS 配信	イネーブル
IP を介した CFS	ディセーブル
IPv4 マルチキャストアドレス	239.255.70.83
IPv6 マルチキャストアドレス	ff15::efff:4653

## その他の関連資料

詳細については、次の項を参照してください。

- 「[関連資料](#)」 (P.2-24)
- 「[MIB](#)」 (P.2-24)

## 関連資料

関連項目	マニュアル名
CFS CLI コマンド	『Cisco Nexus 7000 Series NX-OS System Management Command Reference, Release 4.1』
Call Home の CFS コンフィギュレーション	「 <a href="#">Smart Call Home の設定</a> 」 (P.6-1)
VDC および VRF	『Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide, Release 4.1』
TACACS+ の CFS コンフィギュレーション	『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 4.1』
RADIUS の CFS コンフィギュレーション	『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 4.1』
ロールの CFS コンフィギュレーション	『Cisco Nexus 7000 Series NX-OS Security Configuration Guide, Release 4.1』

## MIB

MIB	MIB のリンク
• CISCO-CFS-MIB	<a href="#">Cisco NX-OS MIB サポート</a>

## CFS 機能の履歴

この項では CFS のリリース履歴を示します。

機能名	リリース	機能情報
CFS プロトコル	4.1(2)	この機能が導入されました。