



## RADIUS サーバ ロード バランシング

RADIUS サーバ ロード バランシング機能は、認証、認可、およびアカウントティング（AAA）の認証トランザクションとアカウントティングトランザクションをサーバグループ内のRADIUSサーバに分配します。これらのサーバは、AAA トランザクションの負荷を共有することで、着信要求に迅速に応答できるようになります。

このモジュールでは、RADIUS サーバ ロード バランシング機能について説明します。

- [RADIUS サーバ ロード バランシングの前提条件](#) (1 ページ)
- [RADIUS サーバ ロード バランシングの制約事項](#) (1 ページ)
- [RADIUS サーバ ロード バランシングに関する情報](#) (2 ページ)
- [RADIUS サーバ ロード バランシングの設定方法](#) (5 ページ)
- [RADIUS サーバ ロード バランシングの設定例](#) (9 ページ)
- [RADIUS サーバ ロード バランシングのその他の参考資料](#) (15 ページ)
- [RADIUS サーバ ロード バランシングの機能履歴](#) (15 ページ)

### RADIUS サーバ ロード バランシングの前提条件

- 認証、認可、およびアカウントティング（AAA）を RADIUS サーバに設定する必要があります。
- AAA RADIUS サーバ グループを設定する必要があります。
- 認証、アカウントティング、スタティック ルート ダウンロードなどの機能用に RADIUS を設定する必要があります。

### RADIUS サーバ ロード バランシングの制約事項

- パケット オブ ディスコネクト（POD）要求などの着信 RADIUS 要求はサポートされていません。
- ロード バランシングは、プロキシ RADIUS サーバ上とプライベート サーバ グループではサポートされていません。

# RADIUS サーバロード バランシングに関する情報

## RADIUS サーバロード バランシングの概要

ロードバランシングは、トランザクションのバッチをサーバグループ内のRADIUSサーバに分配します。ロードバランシングにより、トランザクションの各バッチは、キュー内の未処理トランザクション数が最も少ないサーバに割り当てられます。トランザクションのバッチの割り当てプロセスは次のとおりです。

1. 最初のトランザクションが新しいバッチとして受信されます。
2. すべてのサーバ トランザクション キューがチェックされます。
3. 最小番号の未処理トランザクションを持つサーバが特定されます。
4. 特定されたサーバが、トランザクションの次のバッチに割り当てられます。

バッチサイズはユーザー設定のパラメータです。バッチサイズを変更すると、CPUの負荷やネットワークのスループットに影響する可能性があります。バッチサイズが大きくなるほど、CPUの負荷が減少し、ネットワークのスループットが増加します。ただし、バッチサイズが大きくても、使用可能なすべてのサーバリソースが使い果たされることはありません。バッチサイズが小さくなるほど、CPUの負荷が増加し、ネットワークのスループットが減少します。



---

(注) 大きなバッチサイズまたは小さなバッチサイズに関する設定数はありません。50を超えるトランザクションを含むバッチは大きいと見なされ、25より少ないトランザクションを含むバッチは、小さいと見なされます。

---



---

(注) サーバグループに10以上のサーバが含まれている場合、CPUの負荷を軽減するために高いバッチサイズを設定することを推奨します。

---

## RADIUS サーバグループ全体のトランザクションのロードバランシング

名前付きRADIUSサーバグループごとに、またはグローバルRADIUSサーバグループに対してロードバランシングを設定できます。ロードバランシングサーバグループは、認証、認可、およびアカウントリング(AAA)方式リストで「radius」として参照される必要があります。RADIUSサーバグループの一部であるすべてのパブリックサーバは、その後、ロードバランシングされます。

同じ RADIUS サーバを使用するか、または別のサーバを使用するように認証およびアカウントティングを設定できます。1 つのサーバをセッションの事前認証、認証、またはアカウントティング トランザクションに使用することもできます。内部設定であり、デフォルトとして設定される優先サーバが、サーバ コストに関係なく、セッションの開始レコードと終了レコードに対して同じサーバを使用するよう AAA に指示します。優先サーバ設定を使用する場合は、初期 トランザクション（認証など）に使用されるサーバ、つまり優先サーバが、以降の トランザクション（アカウントティングなど）に使用される他のサーバグループにも属するようにします。

優先サーバは、次のいずれかの条件が真である場合は使用されません。

- **load-balance method least-outstanding ignore-preferred-server** コマンドが使用されている。
- 優先サーバが停止中である。
- 優先サーバが隔離中である。
- 必要サーバ フラグがセットされている場合は、優先サーバ設定が無効になります。

内部設定である必要サーバ フラグは、サーバ コストに関係なく、マルチステージ トランザクションのすべてのステージに対して同じサーバを使用する必要がある場合に使用されます。必要サーバが使用できない場合は、トランザクションが失敗します。

次のいずれかの設定がある場合、**load-balance method least-outstanding ignore-preferred-server** コマンドを使用できます。

- 専用の認証サーバと別の専用のアカウントティング サーバ
- 開始レコードと終了レコード、および別のサーバに保存されたレコードなど、すべての通話レコード統計情報と通話レコード詳細を追跡可能なネットワーク

認証サーバをアカウントティング サーバのスーパーセットとして設定している場合、優先サーバは使用されません。

## RADIUS サーバ ステータスと自動テスト

RADIUS サーバ ロード バランシング機能では、バッチを割り当てるときにサーバ ステータスを考慮します。トランザクションのバッチは、稼働中のサーバのみに送信されます。あまり使用されていないサーバ（バックアップサーバなど）を含む、すべての RADIUS ロード バランシング サーバのステータスをテストすることを推奨します。

停止中としてマークされたサーバにはトランザクションが送信されません。隔離状態になったサーバは、タイマーが切れるまで停止中としてマークされます。RADIUS 自動テスト機能によって動作中であることが確認されるまでサーバは隔離中になります。

サーバが稼働中でトランザクションを処理できるかどうかを確認するために、RADIUS 自動テストは、テスト ユーザー ID で要求を定期的にサーバに送信します。サーバが **Access-Reject** メッセージを返した場合、サーバは稼働中です。それ以外の場合、サーバは停止中または隔離中です。

未応答のサーバーに送信されたトランザクションは、未応答のサーバーが停止中としてマークされる前に、次の使用可能なサーバーにフェールオーバーされます。失敗したトランザクションには再試行順序変更モードを使用することを推奨します。

RADIUS 自動テスターを使用する場合、認証、認可、およびアカウントिंग (AAA) サーバーが、ネットワーク アクセス サーバー (NAS) によって送信されるテスト パケットに応答していることを確認します。サーバーが正しく設定されていない場合は、パケットが破棄され、サーバーが誤って停止中としてマークされる可能性があります。



---

**注意** RADIUS サーバー上で定義されていないテスト ユーザーを RADIUS サーバー自動テストに使用して、テストユーザーが正しく設定されていない場合に発生するセキュリティ上の問題を解決することを推奨します。

---



---

(注) ロード バランシング トランザクションを確認するには、**test aaa group** コマンドを使用します。

---

## VRF 対応 RADIUS 自動テスト

RADIUS 自動テスター機能は、サーバーレベルの設定で動作します。機能に関連付けられたグループがありません。VRF はグループレベルの設定です。VRF および送信元インターフェイスの設定に関連するすべての情報は、グループ設定で維持されます。VRF および送信元インターフェイスの設定に関する情報がグローバル送信元インターフェイスで使用できる場合、自動テスターはこれにアクセスできます。情報がグローバル送信元インターフェイスまたはデフォルト VRF で使用できない場合、自動化されたテスターはサーバーをデッドサーバーとしてマークします。

Cisco IOS XE Bengaluru 17.4.1 以降では、VRF を認識するように自動テスターを設定できます。**automate-tester** コマンドで **vrf** キーワードを使用すると、デフォルト以外の VRF の自動テスト機能を有効化します。



---

(注) VRF 対応の自動化テスターを機能させるには、**global config ipv4/ipv6 source interface interface-namevrf vrf-name** コマンドを設定する必要があります。

---

# RADIUS サーバロードバランシングの設定方法

## 名前付き RADIUS サーバグループのロードバランシングの有効化

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> <b>enable</b>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	<b>configure terminal</b> 例： Device# <b>configure terminal</b>	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	<b>aaa group server radius group-name</b> 例： Device (config)# <b>aaa group server radius rad-sg</b>	サーバグループ コンフィギュレーションモードに入ります。
ステップ 4	<b>server ip-address [auth-port port-number] [acct-port port-number]</b> 例： Device (config-sg-radius)# <b>server 192.0.2.238 auth-port 2095 acct-port 2096</b>	グループサーバ用の RADIUS サーバの IP アドレスを設定します。
ステップ 5	<b>load-balance method least-outstanding [batch-size number] [ignore-preferred-server]</b> 例： Device (config-sg-radius)# <b>load-balance method least-outstanding batch-size 30</b>	名前付きサーバグループに対して最小未処理ロードバランシングを有効にします。
ステップ 6	<b>end</b> 例： Device (config-sg-radius)# <b>end</b>	サーバグループ コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

## RADIUS サーバロード バランシングのトラブルシューティング

RADIUS サーバロード バランシング機能を設定した後は、アイドルタイマー、デッドタイマー、ロードバランシングサーバの選択をモニターしたり、手動テストコマンドを使用してサーバステータスを確認したりできます。

### 手順

**ステップ1 debug aaa test** コマンドを使用して、アイドルタイマーやデッドタイマーが期限切れになった日時、テストパケットが送信された日時、およびサーバステータスを特定し、サーバの状態を確認します。

アイドルタイマーは、サーバステータスのチェックに使用され、着信要求の有無に関係なく更新されます。アイドルタイマーをモニターすると、未応答のサーバが存在するかどうかを判断し、RADIUSサーバのステータスを最新の状態に保つことができるため、利用可能なリソースを効率的に利用できます。たとえば、アイドルタイマーが更新されていれば、着信要求が動作中のサーバに送信されていることを簡単に確認できます。

デッドタイマーは、サーバが停止中であることを特定したり、停止中のサーバのステータスを適切に更新したりするために使用します。

サーバの選択をモニターすると、サーバの選択が変更される頻度を特定するのに役立ちます。サーバの選択は、ボトルネック、つまり、キュー内に大量の要求が存在するかどうかや、特定のサーバのみが着信要求を処理しているかどうかを分析するのに有効です。

**debug aaa test** コマンドの次のサンプル出力は、アイドルタイマーが期限切れになった日時を示しています。

例：

```
Device# debug aaa test

Jul 16 00:07:01: AAA/SG/TEST: Server (192.0.2.245:1700,1701) quarantined.
Jul 16 00:07:01: AAA/SG/TEST: Sending test request(s) to server (192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Sending 1 Access-Requests, 1 Accounting-Requests in current
batch.
Jul 16 00:07:01: AAA/SG/TEST(Req#: 1): Sending test AAA Access-Request.
Jul 16 00:07:01: AAA/SG/TEST(Req#: 1): Sending test AAA Accounting-Request.
Jul 16 00:07:01: AAA/SG/TEST: Obtained Test response from server (192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Obtained Test response from server (192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Necessary responses received from server
(192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Server (192.0.2.245:1700,1701) marked ALIVE. Idle timer
set for 60 sec(s).
Jul 16 00:07:01: AAA/SG/TEST: Server (192.0.2.245:1700,1701) removed from quarantine.
```

**ステップ2 debug aaa sg-server selection** コマンドを使用して、ロードバランシング用に選択されたサーバを特定します。

**debug aaa sg-server selection** コマンドの次のサンプル出力は、5つのアクセス要求がバッチサイズ3のサーバグループに送信されていることを示しています。

例：

```

Device# debug aaa sg-server selection

Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [3] transactions remaining in batch. Reusing
server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [2] transactions remaining in batch. Reusing
server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [1] transactions remaining in batch. Reusing
server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: No more transactions in batch. Obtaining a new
server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining a new least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Server[0] load: 3
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Server[1] load: 0
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Server[2] load: 0
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Selected Server[1] with load 0
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [3] transactions remaining in batch.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [2] transactions remaining in batch. Reusing
server.

```

**ステップ 3 test aaa group** コマンドを使用して、RADIUS ロードバランシング サーバのステータスを手動で確認します。

次のサンプル出力は、ユーザー名「test」がユーザー プロファイルと一致しない場合の動作中の RADIUS ロードバランシング サーバからの応答を示しています。**test aaa group** コマンドを使用して生成された認証、認可、およびアカウントिंग (AAA) パケットに対し、サーバが Access-Reject 応答を発行する場合、そのサーバは動作中であることが確認されます。

例：

```

Device# test aaa group SG1 test lab new-code

00:06:07: RADIUS/ENCODE(00000000):Orig. component type = INVALID
00:06:07: RADIUS/ENCODE(00000000): dropping service type, "radius-server attribute 6
on-for-login-auth" is off
00:06:07: RADIUS(00000000): Config NAS IP: 192.0.2.4
00:06:07: RADIUS(00000000): sending
00:06:07: RADIUS/ENCODE: Best Local IP-Address 192.0.2.141 for Radius-Server 192.0.2.176
00:06:07: RADIUS(00000000): Send Access-Request to 192.0.2.176:1645 id 1645/1, len 50
00:06:07: RADIUS: authenticator CA DB F4 9B 7B 66 C8 A9 - D1 99 4E 8E A4 46 99 B4
00:06:07: RADIUS: User-Password [2] 18 *
00:06:07: RADIUS: User-Name [1] 6 "test"
00:06:07: RADIUS: NAS-IP-Address [4] 6 192.0.2.141
00:06:07: RADIUS: Received from id 1645/1 192.0.2.176:1645, Access-Reject, len 44
00:06:07: RADIUS: authenticator 2F 69 84 3E F0 4E F1 62 - AB B8 75 5B 38 82 49 C3
00:06:07: RADIUS: Reply-Message [18] 24
00:06:07: RADIUS: 41 75 74 68 65 6E 74 69 63 61 74 69 6F 6E 20 66 [Authentication f]
00:06:07: RADIUS: 61 69 6C 75 72 65 [failure]
00:06:07: RADIUS(00000000): Received from id 1645/1
00:06:07: RADIUS/DECODE: Reply-Message fragments, 22, total 22 bytes

```

## VRF 対応 RADIUS 自動テストの有効化

デフォルト以外の VRF の RADIUS 自動テストを有効化するには、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	<b>enable</b> 例： Device> <b>enable</b>	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	<b>configure terminal</b> 例： Device# <b>configure terminal</b>	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	<b>radius server name</b> 例： Device(config)# <b>radius server myserver</b>	RADIUS サーバの名前を指定し、RADIUS サーバ コンフィギュレーション モードを開始します。
ステップ 4	<b>address</b> { <b>ipv4 ipv6</b> } { <i>ip-address host-name</i> } <b>auth-port</b> <i>port-number</i> <b>acct-port</b> <i>port-number</i> 例： Device(config-radius-server)# <b>address</b> <b>ipv4 192.0.2.1 auth-port 1812</b> <b>acct-port 1813</b>	RADIUS サーバのアカウントingおよび認証パラメータの IPv4 アドレスを設定します。
ステップ 5	<b>automate-tester username user</b> [ <b>ignore-auth-port</b> ] [ <b>ignore-acct-port</b> ] [ <b>idle-time minutes</b> ] <b>vrf vrf-name</b> または <b>automate-tester username user probe-on</b> <b>vrf vrf-name</b> 例： Device(config-radius-server)# <b>automate-tester username user1</b> <b>idle-time 2 vrf VRF1</b>  または Device(config-radius-server)# <b>automate-tester username user1 probe-on</b> <b>vrf VRF1</b>	デフォルト以外の VRF の RADIUS 自動テストを有効にします。
ステップ 6	<b>end</b> 例： Device(config-radius-server)# <b>end</b>	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。



# RADIUS サーバロードバランシングの設定例

## 例：名前付き RADIUS サーバグループのロードバランシングの有効化

次の例は、名前付き RADIUS サーバグループで有効化されたロードバランシングを示しています。これらの例は、RADIUS コマンド出力の現在の設定、デバッグ出力、認証、認可、およびアカウントिंग (AAA) サーバのステータス情報という 3 つの部分からなります。

次のサンプル出力は、関連する RADIUS 設定を示しています。

```
Device# show running-config
.
.
.
aaa group server radius server-group1
 server 192.0.2.238 auth-port 2095 acct-port 2096
 server 192.0.2.238 auth-port 2015 acct-port 2016
 load-balance method least-outstanding batch-size 5
!
aaa authentication ppp default group server-group1
aaa accounting network default start-stop group server-group1
.
.
.
Device(config-sg-radius)# load-balance method least-outstanding batch-size 30
```

上記の RADIUS コマンド出力のうち、現在の設定に関する行は次のように定義されます。

- **aaa group server radius** コマンドは、2 つのメンバー サーバからなるサーバグループの設定を表示します。
- **load-balance** コマンドは、バッチサイズが指定されたグローバル RADIUS サーバグループのロードバランシングを有効化します。
- **aaa authentication ppp** コマンドは、RADIUS を使用してすべての PPP ユーザーを認証します。
- **aaa accounting** コマンドは、クライアントが認証された後に **start-stop** キーワードを使用して切断されたときに、すべてのアカウントング要求を AAA サーバに送信できるようにします。

下の `show debug` サンプル出力は、前の設定に関する優先サーバの選択と要求の処理を示しています。

```
Device# show debug
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002C):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Server[0] load:0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Selected Server[0] with load 0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
```

## 例：名前付き RADIUS サーバグループのロードバランシングの有効化

```

*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002C):Server (192.0.2.238:2095,2096) now
being used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002D):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[4] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002D):Server (192.0.2.238:2095,2096) now
being used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002E):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[3] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002E):Server (192.0.2.238:2095,2096) now
being used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002F):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[2] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002F):Server (192.0.2.238:2095,2096) now
being used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(00000030):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[1] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(00000030):Server (192.0.2.238:2095,2096) now
being used as preferred server
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000031):No preferred server available.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new
server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Server[0] load:5
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Selected Server[1] with load 0
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000031):Server (192.0.2.238:2015,2016) now
being used as preferred server
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000032):No preferred server available.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:[4] transactions remaining in batch. Reusing
server.
.
.
.

```

**show aaa servers** コマンドの次のサンプル出力は、名前付き RADIUS サーバグループ設定に対する AAA サーバのステータスを示しています。

このサンプル出力は、2つの RADIUS サーバのステータスを示しています。両方のサーバが動作中ですが、カウンタが0分前にクリアされて以降は、どの要求も処理されていません。

```

Device# show aaa servers

RADIUS: id 3, priority 1, host 9:76:239::219, auth-port 1812, acct-port 1813, hostname
r6
State: current UP, duration 223000s, previous duration 301s
Dead: total time 682s, count 2
Platform State from SMD: current UP, duration 222972s, previous duration 258s
SMD Platform Dead: total time 702s, count 3
Platform State from WNCD (1) : current UP
Platform State from WNCD (2) : current UP
Platform State from WNCD (3) : current UP
Platform State from WNCD (4) : current UP
Platform State from WNCD (5) : current UP

```

```

Platform State from WNCD (6) : current UP
Platform State from WNCD (7) : current UP
Platform State from WNCD (8) : current UP, duration 2451264s, previous duration 258s
Platform Dead: total time 703s, count 3
Quarantined: No
Authen: request 68, timeouts 68, failover 0, retransmission 53

Sates defination:
State: current UP. ==> this is IOSD state
Platform State from SMD: current UP. ==> This is wired BINOS i.e SMD
Platform State from WNCD (1) : current UP ==> This is wireless BINOS i.e WNCD instance
1
Platform State from WNCD (2) : current UP. ==> This is wireless BINOS i.e WNCD instance
2
Platform State from WNCD (3) : current UP
Platform State from WNCD (4) : current UP
Platform State from WNCD (5) : current UP
Platform State from WNCD (6) : current UP
Platform State from WNCD (7) : current UP
Platform State from WNCD (8) : current UP. ==> This is wireless BINOS i.e WNCD instance
8

```

## 例：アイドルタイマーのモニタリング

次の例は、名前付き RADIUS サーバグループに対して有効にされたロードバランシングに関するアイドルタイマーと関連するサーバ状態を示しています。RADIUS コマンド出力と debug コマンド出力の現在の設定も表示されます。

次のサンプル出力は、関連する RADIUS 設定を示しています。

```

Device(config)# do show run aaa

aaa group server radius server-group1
radius server server1
address ipv4 192.0.2.1 auth-port 1812 acct-port 1813
automate-tester username user1 idle-time 2 vrf VRF1
radius-server load-balance method least-outstanding batch-size 5

```

上記の RADIUS コマンド出力のうち、現在の設定に関する行は次のように定義されます。

- **aaa group server radius** コマンドは、サーバグループの設定を表示します。
- **radius server** および **address** コマンドは、指定された認可ポートおよびアカウントングポートを使用して、RADIUS サーバ名と RADIUS サーバの IP アドレスを定義します。
- **radius-server load-balance** コマンドは、バッチサイズが指定された RADIUS サーバのロードバランシングを有効化します。

下の **show debug** サンプル出力は、サーバに送信されるテスト要求を示しています。サーバに送信されたテスト要求に対する応答が受信され、必要に応じて、隔離からサーバが除外され、サーバが動作中としてマークされてから、アイドルタイマーがリセットされます。

```

Device# show debug

*Feb 28 13:52:20.835:AAA/SG/TEST:Server (192.0.2.238:2015,2016) quarantined.
*Feb 28 13:52:20.835:AAA/SG/TEST:Sending test request(s) to server (192.0.2.238:2015,2016)
*Feb 28 13:52:20.835:AAA/SG/TEST:Sending 1 Access-Requests, 1 Accounting-Requests in

```

## 例：認証サーバと認可サーバが同じ優先サーバの設定

```

current batch.
*Feb 28 13:52:20.835:AAA/SG/TEST(Req#:1):Sending test AAA Access-Request.
*Feb 28 13:52:20.835:AAA/SG/TEST(Req#:1):Sending test AAA Accounting-Request.
*Feb 28 13:52:21.087:AAA/SG/TEST:Obtained Test response from server (192.0.2.238:2015,2016)
*Feb 28 13:52:22.651:AAA/SG/TEST:Obtained Test response from server (192.0.2.238:2015,2016)
*Feb 28 13:52:22.651:AAA/SG/TEST:Necessary responses received from server
(192.0.2.238:2015,2016)
*Feb 28 13:52:22.651:AAA/SG/TEST:Server (192.0.2.238:2015,2016) marked ALIVE. Idle timer
set for 60 secs(s).
*Feb 28 13:52:22.651:AAA/SG/TEST:Server (192.0.2.238:2015,2016) removed from quarantine.
.
.
.

```

## 例：認証サーバと認可サーバが同じ優先サーバの設定

次の例は、サーバーの 209.165.200.225 と 209.165.200.226 を共有する認証サーバー グループと認可サーバー グループを示しています。両方のサーバー グループで優先サーバー フラグが有効になっています。

```

Device> enable
Device# configure terminal
Device(config)# aaa group server radius authentication-group
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# exit
Device(config)# aaa group server radius accounting-group
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# end

```

あるセッションで優先サーバーが選択されると、そのセッションのすべてのトランザクションでオリジナルの優先サーバーの使用が継続されます。サーバーの 209.165.200.225 と 209.165.200.226 は、トランザクションではなく、セッションに基づいてロードバランシングされます。

## 例：認証サーバと認可サーバが別々の優先サーバの設定

次の例は、サーバーの 209.165.200.225 と 209.165.200.226 を使用する認証サーバー グループとサーバーの 209.165.201.1 と 209.165.201.2 を使用する認可サーバー グループを示しています。両方のサーバー グループで優先サーバー フラグが有効になっています。

```

Device> enable
Device# configure terminal
Device(config)# aaa group server radius authentication-group
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# exit
Device(config)# aaa group server radius accounting-group
Device(config-sg-radius)# server 209.165.201.1 key radkey3
Device(config-sg-radius)# server 209.165.201.2 key radkey4
Device(config-sg-radius)# end

```

認証サーバー グループとアカウントングサーバー グループはどの共通サーバーも共有しません。アカウントング トランザクションでは優先サーバーは検出されないため、認証サー

サーバとアカウントングサーバはトランザクションに基づいてロードバランシングされます。1つのセッションで開始レコードと終了レコードが同じサーバに送信されます。

## 例：認証サーバと認可サーバが重複している優先サーバの設定

次の例は、サーバの 209.165.200.225、209.165.200.226、および 209.165.201.1 を使用する認証サーバグループとサーバの 209.165.201.1 と 209.165.201.2 を使用する認可サーバグループを示しています。両方のサーバグループで優先サーバフラグが有効になっています。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius authentication-group
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# server 209.165.201.1 key radkey3
Device(config-sg-radius)# exit
Device(config)# aaa group server radius accounting-group
Device(config-sg-radius)# server 209.165.201.1 key radkey3
Device(config-sg-radius)# server 209.165.201.2 key radkey4
Device(config-sg-radius)# end
```

すべてのサーバのトランザクション処理能力が同じ場合は、すべての認証トランザクションの 1/3 がサーバの 209.165.201.1 に転送されます。したがって、すべてのアカウントングトランザクションの 1/3 もサーバの 209.165.201.1 に転送されます。アカウントングトランザクションの残りの 2/3 は、サーバの 209.165.201.1 と 209.165.201.2 の間で均等にロードバランシングされます。サーバの 209.165.201.1 に未処理のアカウントングトランザクションがあるため、サーバの 209.165.201.1 が受信する認証トランザクション数は減少します。

## 例：認証サーバが認可サーバのサブセットである優先サーバの設定

次の例は、サーバの 209.165.200.225 と 209.165.200.226 を使用する認証サーバグループと、サーバの 209.165.200.225、209.165.200.226、および 209.165.201.1 を使用する認可サーバグループを示しています。両方のサーバグループで優先サーバフラグが有効になっています。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius authentication-group
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# exit
Device(config)# aaa group server radius accounting-group
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# server 209.165.201.1 key radkey3
Device(config-sg-radius)# end
```

すべての認証トランザクションの半分がサーバの 209.165.200.225 に送信され、残りの半分がサーバの 209.165.200.226 に送信されます。サーバの 209.165.200.225 と 209.165.200.226 は、認証およびアカウントングトランザクションの優先サーバです。そのため、認証およびアカウントングトランザクションは、サーバの 209.165.200.225 と 209.165.200.226 に均等に分配されます。サーバの 209.165.201.1 は相対的に使用されません。

## 例：認証サーバが認可サーバのスーパーセットである優先サーバの設定

次の例は、サーバの 209.165.200.225、209.165.200.226、および 209.165.201.1 を使用する認証サーバグループとサーバの 209.165.200.225 と 209.165.200.226 を使用する認可サーバグループを示しています。両方のサーバグループで優先サーバフラグが有効になっています。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius authentication-groupp
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# server 209.165.201.1 key radkey3
Device(config-sg-radius)# exit
Device(config)# aaa group server radius accounting-group
Device(config-sg-radius)# server 209.165.200.225 key radkey1
Device(config-sg-radius)# server 209.165.200.226 key radkey2
Device(config-sg-radius)# end
```

最初に、認証トランザクションの 1/3 が認可サーバグループ内の各サーバに割り当てられます。追加のセッションに対してアカウントングトランザクションが生成されますが、優先サーバフラグがオンになっているため、アカウントングトランザクションはサーバの 209.165.200.225 と 209.165.200.226 に送信されます。サーバの 209.165.200.225 と 209.165.200.226 がトランザクションの処理を開始しますが、認証トランザクションはサーバの 209.165.201.1 に送信されます。サーバの 209.165.201.1 で認証されたトランザクション要求は、どの優先サーバ設定も含まず、サーバの 209.165.200.225 と 209.165.200.226 に分配されるため、優先サーバフラグの使用が無効になります。この設定は慎重に使用する必要があります。

## 例：VRF 対応 RADIUS 自動テストの有効化

次に、RADIUS サーバでデフォルト以外の VRF の自動テストを有効にする例を示します。

```
Device(config)# radius server myserver
Device(config-radius-server)# address ipv4 192.0.2.1 auth-port 1812 acct-port 1813
Device(config-radius-server)# automate-tester username user1 idle-time 2 vrf VRF1
Device(config-radius-server)# end

Device(config)# radius server myserver
Device(config-radius-server)# address ipv4 192.0.2.1 auth-port 1812 acct-port 1813
Device(config-radius-server)# automate-tester username user1 probe-on vrf VRF1
Device(config-radius-server)# end
```

## RADIUS サーバロードバランシングのその他の参考資料

### 関連資料

関連項目	マニュアルタイトル
RADIUS	『 <i>Security Configuration Guide</i> 』の「Configuring RADIUS」モジュール

### シスコのテクニカルサポート

説明	リンク
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンラインリソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。</p>	<a href="http://www.cisco.com/en/US/support/index.html">http://www.cisco.com/en/US/support/index.html</a>

## RADIUS サーバロードバランシングの機能履歴

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 Everest 16.11.1	RADIUS サーバロード バランシング	RADIUS サーバロード バランシング機能は、認証、認可、およびアカウントिंग (AAA) の認証トランザクションとアカウントングトランザクションをサーバグループ内のサーバに分配します。これらのサーバは、AAA トランザクションの負荷を共有することで、着信要求に迅速に応答できるようになります。
Cisco IOS XE Bengaluru 17.4.1	VRF 対応 RADIUS 自動テスト	デフォルト以外の VRF の RADIUS 自動テストを設定できます。

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。