クライアントクラスとクライアントの管理

Cisco Prime Network レジストラークライアントとクライアント クラスの概念を使用して、共通のネットワークを介してユーザーに差別化されたサービスを提供します。管理基準に基づいてクライアントをグループ化し、各グループが適切なサービス クラス (COS) を受け取ることを確認できます。クライアント クラスの処理を行わない場合、DHCP サーバーはネットワーク上の場所のみに基づいてクライアント リースを提供します。

- クライアントクラスの設定 （1 ページ）
- クライアントクラスのトラブルシューティング （9 ページ）
- クライアントの設定 （10 ページ）
- オプション 82 を使用したサブスクライバの制限 （16 ページ）
- LDAP を使用するように Cisco Prime Network Registrar を設定する （21 ページ）

クライアントクラスの設定

クライアント サービスは、次の方法で区別できます。

- Cisco Prime Network レジストラーデータベース(このセクション)またはライトウェイト ディレクトリ アクセスLDAP を使用するように Cisco Prime Network Registrar を設定する （21 ページ） プロトコル(を参照)を使用してクライアントを登録します。
- アップストリーム クライアントをサービス クラス別に区別できるように、仲介デバイス (ケーブル モデムなど)を登録します。
- クライアント データの予知なしに、クライアント パケットの内容を使用します。
  - パケット内に存在する既知の DHCP オプション*(dhcp-user-class-id* DHCP オプション (77)、またはリレー エージェント情報DHCP オプションの*radius* 属性サブオプション外部ソースを含むクライアント データの処理 （7 ページ） (82 を参照)
  - クライアント クラス *lookup-ID* DHCP サーバー属性の式を使用して抽出するパケット内のその他クライアントクラスの計算とキーの作成 （17 ページ） のデータ (を参照)

- クライアントクラスを作成してクライアントを割り当て、次に特定のクライアントに対してクライアントルックアップ *ID*を設定する2サブスクライバ制限のための式処理（18ページ）段階のプロセスを使用します (「」を参照)。

クライアントクラス処理

DHCP サーバーのクライアント クラス処理を有効または無効にし、一連のプロパティをクライアントのグループに適用します。クライアントクラスが有効な場合、サーバーは、一致する DHCPv4 スコープまたは DHCPv6 プレフィックスからクライアントをアドレスに割り当てます。サーバーはパケット内のデータに従って動作します。クライアント クラスを構成するには、次の手順に従います。

1. DHCP サーバーのクライアントクラス処理を有効にします。
2. 選択タグ (条件) を含む、または除外するクライアント クラスを定義します。
3. 選択タグを特定のスコープまたはプレフィックス (またはそのテンプレート) に適用します。
4. これらのクラスにクライアントを割り当てます。

このプロセスは Cisco Primeネットワーク レジストラーを通じて設定されたクライアントに対するものです。外部ソースからのデータの影響を受ける処理外部ソースを含むクライアントデータの処理 （7 ページ）については、を参照してください。

クライアントクラスの定義

クライアント クラスは、サーバー レベルで有効にして定義します。

ローカル Web UI

ステップ 1 基本モードまたは詳細モードでクライアントクラスを有効にするには、次の手順を実行します。

a) **Deploy** メニューで、[DHCP] サブメニューから **DHCP Server** を選択し、[DHCPサーバーの管理（Manage DHCP Server）] ページを開きます。
b) [DHCP サーバー] ペインでサーバーを選択します。
c) [DHCP サーバーの編集] タブで、クライアント クラス属性を有効にします。
d) **Save** をクリックします。

ステップ 2 [**デザイン**] メニューの**[DHCP 設定] Classes**サブメニューの [**クライアント**] を選択して、[DHCP クライアント クラスの一覧/追加] ページを開きます。

ステップ 3 [クライアント クラス] ウィンドウの [**クライアント クラスの追加**] アイコンをクリックして、[DHCP クライアント クラスの追加] ダイアログ ボックスを開きます。

ステップ 4 クライアントクラス名を入力します。

ステップ 5 その他のクライアント クラス プロパティを設定します。ホスト名とドメイン名の属性は、DNS 更新の構成を使用していない場合、主にDNS 更新設定の作成DNS 更新に使用されます (を参照してください)。

ホスト名のプロパティについては、クライアントクラス ホスト名プロパティの定義 （5 ページ）で説明します。クライアント クラスに適したポリシーを選択することもできます。

ステップ 6 [**クライアント クラスの追加]**をクリックします。

ステップ 7 選択基準を定義します。

クライアントクラスを作成する際の重要なステップは、クライアントクラスを DHCPv4 スコープまたは DHCPv6 プレフィックスに関連付けることができるように、選択基準を定義することです。選択基準属性を使用します (表 1 : 使用する選択タグと基準属性 も参照)。

複数の選択タグをコンマで区切って入力できます。値は、目的のスコープまたはプレフィックスに設定された選択タグと一致するスコープとプレフィックスの選択タグの設定 （4 ページ）必要があります (を参照)。

ステップ 8 クライアント クラスに埋め込みポリシーを追加するにはクライアントと組み込みポリシーの編集 （12 ページ）、「」を参照してください。

ステップ 9 [保存（Save）] をクリックします。

ステップ 10 必要に応じてデバッグします。クライアント クラスのエラーをデバッグするには、[ローカル DHCP サーバー] ページの [ログ設定] セクションでクライアント基準処理属性を有効にします。

ステップ 11 クライアントクラスを削除するには、クライアントを選択し、左側の[クライアントクラス]ペインの[**クライアントクラスの削除**]アイコンをクリックして、削除を確認します。

CLI コマンド

クライアントクラスを有効にするには、**dhcp enable client-class** を使用します。クライアントクラスを作成するには**client-class** 、*name* **create**を使用します。名前は、その意図を明確に識別する必要があります。大文字と小文字は区別されません。クラスPCはクラスPCと同じです。

client-class name**set**属性=値を使用して、クライアント クラスのクライアントのプロパティを設定します。たとえば、**client-class** 名前 **set policy-name=**値を使用して、クライアント クラスに関連付ける必要のあるポリシーを設定します。**client-class** nameを**set**使用して、スコープをクライアント クラスに関連付**selection-criteria**けます。（スコープとプレフィックスの選択タグの設定 （4 ページ）を参照してください）。

client-class *name* [ ]**show**を使用して、作成したクライアント クラスのプロパティを表示します。作成されたすべてのクライアント クラスのプロパティを一覧表示したり、名前だけを一覧表示したりすることもできます。クライアント クラスの処理をデバッグするには、**dhcp set log-settings=client-criteria-processing**を使用します。クライアント クラスを削除するには **client-class** 、*name***delete**を使用します。

DHCPv6 クライアントクラスの設定

DHCPv6 クライアント クラス属性は次のように設定できます。

- *v6-client-lookup-id* : クライアントデータベースで（ローカルに、または LDAP を介して）DHCPv6 クライアントを検索するために使用するキーの値。文字列（または有効な文字列としての BLOB）に対して評価する式として指定されます。

- *v6-override-client-id*：着信パケットで client-identity 値を置き換える値。BLOB に対して評価する式として指定します。

ローカル アドバンスド Web UI

メニューから**Design**サブ**Clients**メニューの下を**DHCPSettings**選択して、[DHCP クライアントの一覧/追加] ページを開きます。既存のクライアントを選択して [DHCP クライアントの編集]**Add Clients**ページを開くか、[クライアント] ペインのアイコンをクリックして新しいクライアント クラスを追加し、設定された DHCPv6 属性DHCPv6 クライアントクラスの設定 （3 ページ）を含むクライアント クラスを選択します (を参照**Save**) 。

ヒント　DHCP サーバーの検証クライアント名-*mac*属性を無効にします。

CLI コマンド

既存**clientlist**の**client**クライアントを表示するには、または名前**show**を使用します。クライアントのクライアント クラス名を設定するには**client** 、*name* **set client-class-name=** *value*を使用します。また、DHCP サーバーに対して、検証クライアント名-*as-mac*属性が無効になっていることを確認します。

スコープとプレフィックスの選択タグの設定

クライアントを異なるアドレス プールに割り当てるには、クライアント クラスの選択基準で指定した選択タグを使用して、DHCPv4 スコープ (またはテンプレート) または DHCPv6 プレフィックス (またはテンプレート) を定義する必要があります。スコープまたはプレフィックスに追加のタグがある場合でも、クライアント クラスに含まれるすべての選択基準タグは、スコープまたはプレフィックスが持つタグと一致する必要があります。クライアント・クラスがすべての選択基準を省略した場合、スコープ選択または接頭部選択に制限は適用されません。

次に例を示します。

スコープ A にはタグ 1、タグ 2 があります

スコープ B にはタグ 3、タグ 4 があります

両方のスコープが同じネットワーク上にある場合、クライアントクラスのクライアントは次の情報を持ちます。

- Tag1、tag2、またはその両方がスコープ A からリースを取得します。
- Tag3、tag4、またはその両方がスコープ B からリースを取得します。
- 両方のスコープ (tag1 や tag3 など) の 1 つ以上のタグがどちらのスコープからもリースを取得しません。
- どちらのスコープからもリースを取得するタグはありません。

次の表に、Cisco Prime Network レジストラーがネットワーク オブジェクトの選択タグまたは選択基準を参照するために使用する属性を示します。

表 *1:* 使用する選択タグと基準属性

オブジェクト	属性（**Attribute**）
クライアント	*selection-criteria*
クライアントクラス	*selection-criteria*
範囲	*selection-tag-list*
スコープ テンプレート	*selection-tag-list*
[プレフィックス（Prefix）]	*selection-tags*
接頭語テンプレート	*selection-tags*
アドレス ブロック	*selection-tags*
サブネット（Subnets）	*selection-tags*

ローカル Web UI

範囲あるいはプレフィックスまたはそのテンプレートを作成または編集します。範囲あるいはプレフィックス（またはそのテンプレート）の[追加（Add）]ページまたは[編集（Edit）]ページで *selection-tags* 属性を見つけ、この範囲あるいはプレフィックス（またはそのテンプレート）に関連付けるクライアントクラスの *selection-criteria* 属性で作成した一連のカンマ区切りの選択タグを入力します。その後、必要に応じて変更を保存し、DHCPサーバーを再ロードします。

CLI コマンド

scope 名前 **set selection-tag-list**を使用します。スコープ テンプレートの場合は**scope-template**、*name***setselection-tag-list**を使用します。プレフィックスの場合は、 **prefix** *name* **set selection-tags** を使用します。プレフィックステンプレートの場合は**prefix-template**、 *name***setselection-tags**を使用します。

クライアントクラス ホスト名プロパティの定義

クライアントクラスのホスト名*(host-name)*属性を使用して、各クライアントが採用するホスト名を指定できます。これは、DHCPクライアント要求に含まれるものを上書きする絶対の有効な DNS 値、または次のいずれかです。

- ***@host-name-option***-サーバーは、クライアントが送信したホスト名オプションを使用します。
- ***@no*ホスト名*-option***-サーバーはクライアントが送信するホスト名を無視します。DNS 名の生成が有効な場合、サーバーは、動的 DNS 更新用にそのように設定されている場合、生成された名前を使用します。

- ***@use-macaddress***:サーバーはクライアントのMACアドレスからホスト名を合成し、オクテットをハイフネーション処理してから、**x**前面にaを追加します。たとえば、クライアント MAC アドレスが 1,6:00:d0:ba:d3:bd:3b の場合、合成されたホスト名は x1-6-00-d0-ba-d3-bd-3b になります。

値を省略すると、ホスト名は指定されません。DNS 更新の構成を使用してホスト名を合成することもできます (DNS 更新設定の作成を参照)。

クライアントクラスとその埋め込みポリシーの編集

クライアントクラスの編集には、クライアントクラスを作成するのと同じ属性が含まれます。また、クライアントクラスの埋め込みポリシーを追加および変更して、そのポリシーオプションを設定することもできます。埋め込みポリシーには、追加するまでプロパティやDHCPオプションは関連付けされません。(も参照組み込みポリシーの作成と編集してください)。クライアントクラスの埋め込みポリシー設定は、DHCP サーバーがポリシー選択で使用する 3 番目の優先度であり、クライアントDHCPv4 ポリシー階層自体に設定された後です (「」を参照)。

ローカル アドバンスド Web UI

ステップ 1 クライアント クラスを作成します。

ステップ 2 左側の [クライアント クラス] ペインでクライアント クラスを選択して、[DHCP クライアント クラスの編集] ページを開きます。

ステップ 3 必要に応じて属性設定を変更します。

ステップ 4 クライアント クラスに新しい埋め込みポリシーを追加**Create New Embedded Policy**するには、 をクリックします。編集する既存の埋め込みポリシーがある場合は、**Edit Existing Embedded Policy**をクリックします。(既存の埋め込みポリシーの設定を解除する**Unset**場合は、[DHCP クライアント クラスの編集]**Create New Embedded Policy**ページをクリックします。

a) このページのフィールド、オプション、属性を変更します。たとえば、[DHCPv4 オプション] の下で、ドロップダウン リストから**dhcp-lease-time [51]**[リース期間] を選択してクライアントリース期間を設定し、[値] フィールド**Add Option**にリース間隔の値を入力して、 をクリックします。必要に応じて、属性値を設定解除します。

ステップ 5 [保存（Save）] をクリックします。

CLI コマンド

クライアント クラスに既に設定されている埋め込みポリシー値があるかどうかを確認するには、 **client-class-policy** *client-class-name* **show**を使用します。組み込みポリシーの属性を設定するには**client-class-policy**、クライアント クラス名**set**属性=*value*を使用します。

DHCP オプションを設定するには、次のいずれかのコマンドを使用します。

```
nrcmd> client-class-policy client-class-name setOption {opt-name | id} value [-blob]
[-roundrobin]
nrcmd> client-class-policy client-class-name setV6Option {opt-name | id}[.instance] value
```

```
 [-blob] [-roundrobin]
nrcmd> client-class-policy client-class-name setVendorOption {opt-name | id} opt-set-name
 value [-blob]
nrcmd> client-class-policy client-class-name setV6VendorOption {opt-name | id} opt-set-name
 value [-blob]
```

リース時間を設定するには、クライアント**client-class-policy** クラス名 **setLeaseTime**の値を使用します。

外部ソースを含むクライアント データの処理

DHCP クライアントを実行しているネットワーク ホストとそのユーザーに関する情報は、複数の外部ソースから DHCP サーバーに到着できます。サーバーは、クライアント クラスの処理の一部としてこのデータを使用し、リースデータベースにキャプチャして、Cisco Prime Network レジストラー管理システムで使用できるようにします。

最近導入された外部要因は、クライアントの定義に影響を与える可能性があります。

- リレー エージェント情報DHCP オプション (82) のサブスクライバ *ID*サブオプションは、ネットワーク管理者がネットワークの加入者またはクライアントを定義し、このデータを DHCP サーバーに送信します。
- RADIUS 認証サーバーデータは、802.1x プロトコル導入の一部として使用され、RADIUS データは DHCP の意思決定に役立ちます。この場合、デバイスは、リレー エージェント情報DHCP オプション (82) の*radius* 属性サブオプション属性の一部としてデータを送信できます。

これらの外部オプションはどちらも DHCP オプション 82オプション 82 を使用したサブスクライバの制限 （16 ページ）を使用します (を参照)。RADIUS ソースは、次の属性を終了できます。

- クライアントユーザー名またはアカウント名（*user* 属性）
- 管理上定義されたクラス文字列（*class* 属性）
- ベンダー固有データ（*vendor-specific* 属性）
- セッションタイムアウト値（*session-timeout* 属性）
- クライアントに使用する IP アドレス プール（*framed-pool* 属性）
- クライアントに使用する IPv6 アドレスプール（*framed-ipv6-pool* 属性）

Cisco Prime Network Registrar は、 *subscriber-id* サブオプション、 RADIUS サブオプションの *user*、*class*、および*framed-pool* 属性の拡張サポートとすべてのサブオプションの式のサポートを提供します（式の使用方法 を参照）。さらに、DHCP サーバーには、RADIUS の *class* 属性と *framed-pool* 属性をサーバーが処理する方法を設定する属性設定もあります。Cisco Prime Network Registrar は、サーバー属性を使用して RADIUS 属性値を選択タグまたは client-class 名としてマッピングするか、あるいはクライアントデータベースで検出された選択タグに値を追加できます。次に例を示します。

```
nrcmd> dhcp set map-radius-class=append-to-tags
```

RADIUSなどの外部リソースから決定されたクライアントクラスと選択タグの場合、処理順序はでクライアントクラス処理 （2ページ）説明されているよりもやや複雑です。次のサブセクションを参照してください。クライアントクラス機能を使用するには、DHCPサーバークライアント クラス属性を有効にする必要があります。

クライアントクラスを判別する処理順序

DHCPサーバーがクライアントクラス名を決定するために使用可能なソースを使用する順序は次のとおりです。

1. 拡張環境ディクショナリでクライアント クラス名を使用します。
2. データベース内に実際のクライアントエントリが見つかった場合は、そのクライアントクラス名を使用します。(データベース内のクライアントを検索する必要がなくなると思われる場合は、スキップ クライアントルックアップDHCPサーバー属性を有効にすることで、データベース検索クライアントクラスのクライアント エントリのスキップ （14ページ）を回避できます。
3. RADIUS フレーム プール値をクライアント クラスにマップする場合(を使用**dhcp set map-radius-pool-name=map-as-class**) は、フレーム プールの値を使用します。
4. RADIUS クラスの値をクライアント クラスにマップする場合(**dhcp set map-radius-class=map-as-class**を使用) は、クラス値を使用します。
5. *dhcp-user-class-id* DHCP オプション (77) をクライアントクラスにマップする場合(を使用**dhcpsetmap-user-class-id=map-as-class**) は、オプション値を使用します。(このマッピングクライアントクラスの検索式の処理 （18ページ）の代わりにルックアップ ID 式を使用することもできます。
6. マッピングまたはユーザー クラス ID が見つからず、環境ディクショナリの default-client-class-name が使用されます。
7. クライアントエントリで構成されているデフォルトクライアントクラス名またはクライアントクラスが見つからない場合は、名前が付けられた**default**クライアントからクライアントクラス名を使用します(見つかった場合)。

選択タグを判別する処理順序

サーバーが選択タグを決定するために使用可能なソースを使用する順序(最初のnullでないソースを使用) は次のとおりです。

1. 拡張環境ディクショナリ内の選択タグ。
2. データベース内に実際のクライアントエントリが見つかった場合は、クライアントエントリ選択タグを使用します。(この不要なデータベースの読み取りを防ぐには、スキップ クライアント ルックアップクライアントクラスのクライアント エントリのスキップ （14ページ）DHCP サーバー属性を有効にします。
3. クライアント クラスの選択タグ。

4. 使用可能な RADIUS フレーム プール値をタグにマップする場合**dhcp set map-radius-pool-name=map-as-tag(**を使用) 、そのタグが使用されます。
5. 使用可能なRADIUSクラス値をタグにマップする場合**dhcp set map-radius-class=map-as-tag(**を使用) は、そのタグを使用します。
6. 使用可能な*dhcp-user-class-id* DHCP オプション (77) をタグ(**dhcp set map-user-class-id=map-as-tag**を使用) にマップする場合、そのタグが使用されます。

次に、サーバーは、次のいずれかを選択タグ (存在する場合) のリストに追加できます。

1. RADIUS フレーム プールの値が使用可能で、*map-radius-pool* DHCP 属性を（**dhcp set map-radius-pool=append-to-tags** を使用して）タグに追加するように設定されている場合は、サーバーがその属性を追加します。
2. RADIUS クラスの値が使用可能で、*map-radius-class* DHCP 属性を（**dhcp set map-radius-class=append-to-tags** を使用して）選択タグに追加するように設定されている場合は、サーバーがその属性を追加します。
3. dhcp-user-class-id が使用可能で、*map-user-class-id* DHCP 属性が（**dhcp set map-user-class-id=append-to-tags** を使用して）選択タグに追加するように設定されている場合は、サーバーがその属性を追加します。

クライアントクラスのトラブルシューティング

クライアント クラスのトラブルシューティングを行うには、Web UI の [DHCP サーバーの編集] ページの*log* 設定属性を使用してクライアント クラスのログ記録を有効にするか **dhcpsetlog-settings=**、CLI で設定してから DHCP サーバをリロードします(段階的 dhcp 編集モードの場合)。推奨設定値は以下のとおりです。

- **client-detail**- クライアントクラスのクライアント検索操作の最後に、1 行のログを記録します。この行には、クライアントに対して検出されたすべてのデータと、クライアントクラスで検出されたデータが表示されます。
- **client-criteria-processing**- サーバーが有効なリースを見つけたり、リースが既にリースを持っているクライアントに対して引き続き許容されるかどうかを判断するために、サーバーがスコープまたはプレフィックスを調べるたびにメッセージをログに記録します。
- **ldap-query-detail**DHCP サーバーが LDAP サーバーへのリース状態エントリの作成を開始した場合、LDAP サーバーからの応答を受信したとき、または LDAP サーバーから結果またはエラー メッセージを取得するたびにメッセージをログに記録します。
- 問題が LDAP サーバーに関連している可能性がある場合は、LDAPの照会可能設定も有効にします。

これらのログは、次の質問に答える上で役立ちます。

- サーバーは、予期されたデータベースからクライアントエントリを読み取っていますか。

サーバーは、LDAP または CNRDB（Cisco Prime Network Registrar 内部データベース）からクライアントエントリを読み取ることができます。クライアント詳細ログには、サーバーがクライアントエントリを読み取っている場所が示されます。

- クライアントクラスは有効になっていますか?

  有効にしても予期しない結果が得られる場合は、どのデータベースから Cisco Prime Network レジストラーサーバー読み取りクライアントであるか確認します。LDAP または CNRDB から読み取っていますか?*LDAP* クエリ詳細ログは、LDAP から読み取り中かどうかを示します。ない場合は、DHCP の ldap クライアントデータのプロパティを有効にします。

（注） LDAP を使用するには、照会用に LDAP サーバーを構成する必要があります。LDAPの照会可能属性を有効にします。また、クエリに LDAP を使用するように DHCP サーバーを構成する必要があります。

- サーバーがクライアントに適切なデータを提供していますが、そのデータから誤った結果が見られる (クライアントが予期した IP アドレスを受信していないなど)。

  ネットワーク上の明示的な関係を確認します。クライアント基準処理ログは、サーバーがアドレスを取得しているスコープまたはプレフィックスを示します。予期されるソースからアドレスを取得しない場合は、明示的な関係が正しく定義されていない可能性があります。2 次スコープであると考えたスコープは、そのように定義されていない可能性があります。

- エキスパートモードで、選択タグの包含基準と除外基準を適切に設定しましたか?

  一連の選択タグを定義して含める場合、スコープまたはプレフィックスのタグはクライアントのタグと一致する必要があります。エキスパートモードでは、クライアントクラスで選択基準除外属性を使用して、選択タグを除外することもできます。除外する系列を定義する場合、スコープまたはプレフィックスには、クライアントが構成パラメーターを取得できるように、これらのタグを定義する必要があります。選択タグの操作を開始するときに、複雑な包含および除外のシナリオを避けます。

クライアントの設定

DHCP クライアントのプロパティには、参加するクライアント クラスとクライアントに関連付けられたポリシー、実行するアクション、および選択タグの包含と除外の基準が含まれます。クライアントは、クライアント クラスからプロパティを継承します。

ローカル Web UI

ステップ 1 メニューから**DesignClients[DHCP 設定]**サブメニューの下で[DHCP クライアントの一覧/追加]ページを開きます。

ステップ 2 [クライアント] ウィンドウの [**クライアントの追加**] アイコンをクリックして [DHCP クライアントの追加] ダイアログ ボックスを開き、クライアント ID (通常は MAC アドレス) を入力しますが、DUID またはルックアップ キーを指定することもできます。(サーバー属性の検証-クライアント名-*as-mac*を有効にすることで、クライアント名を MAC アドレスとして検証するように DHCP サーバーを設定できます。

特定のクライアント構成を持たない**default**名前のクライアントを作成することもできます。たとえば、クライアントが常にそのホスト名に MAC アドレスを使用できます。

ステップ 3 必要に応じて、定義済みのクライアント クラスのドロップダウン リストからクライアント クラス名を選択します。

ステップ 4 **Add DHCP Client** をクリックします。[DHCP クライアントの編集] ページが開きます。

クライアントを作成する際の重要なステップは、スコープまたはプレフィックスにクライアントを関連付けることができるように、選択基準を定義することです(クライアントに関連付けられたクライアントクラスに対して選択基準が既に設定されている場合を除く)。

[属性] リストの下にある選択基準属性を使用表 1 : 使用する選択タグと基準属性 （5 ページ）します (参照)。複数の選択タグをコンマで区切って入力できます。値は、目的のスコープまたはプレフィックスに設定された選択タグと一致するスコープとプレフィックスの選択タグの設定 （4 ページ）必要があります (を参照)。

（注） クライアントにクライアントクラスを選択した場合、このページは表示されず、クライアント名はリスト/クライアントの追加ページに表示されます。

ステップ 5 必要に応じて、他の属性を設定します。次に例を示します。

- *host-name*属性を*@no*ホスト名オプションに設定して、不明なクライアントに仮のアドレスを提供します。
- 動的 DNS 更新を実行するときに使用するゾーンのドメイン名を設定します。
- クライアントのポリシーとアクションを設定します。*exclude*アクションを使用すると、サーバーはこのクライアントからのすべての通信を無視します（パケットは表示されません）。
- 認証の有効期限を示す時間単位 (秒、分、時間、日、週)、または UNIX スタイルの日付 (2002 年 3 月 24 日 12:00:00 など) を選択するか、または を使用します**forever**。

ステップ 6 ページの一番下にある **Save** をクリックします。

ステップ 7 必要に応じてデバッグします。クライアント エラーをデバッグするには、DHCP ログ設定 **client-criteria-processing**を に設定します。

ステップ 8 クライアントを削除するには、左側の [クライアント] ウィンドウの [**クライアントの削除**] アイコンをクリックし、削除を確認します。

CLI コマンド

クライアントを作成するには**client** 、*name* **create**を使用します。クライアント クラスをクライアントに関連付けるには**client** 、*name* **set client-class-name**=*value*を使用します。スコープまたはプレフィックスの選択基準を設定するには、 **client** *name* **set selection-criteria**を使用します。その他の属性を設定**client**するには、名前**set**属性=値を使用します。

クライアントのプロパティを表示するには**client** 、*name* [ ]**show**を使用します。すべてのクライアントのプロパティを表示するには、**client list**を使用**client listnames**するか、名前だけを一覧表示します。クライアントをデバッグするには、**dhcp set log-settings=client-detail**を使用します。クライアントを削除するには**client** 、*name* **delete**を使用します。

クライアントと組み込みポリシーの編集

クライアントの編集には、クライアントの作成と同じ属性が含まれます。また、クライアントの埋め込みポリシーを追加および変更して、ポリシー オプションを設定することもできます。埋め込みポリシーには、それを追加するまで、プロパティやDHCP オプションが関連付けられません。（組み込みポリシーの作成と編集も参照してください）。クライアントの埋め込みポリシー設定は、DHCP サーバーがポリシー選択で使用するDHCPv4 ポリシー階層最優先の優先順位です (を参照) 。

ローカル Web UI

ステップ 1 クライアントを作成します。

ステップ 2 [DHCP クライアントの一覧/追加] ページの [クライアント] ペインからクライアントを選択し、[DHCP クライアントの編集] ページを開きます。

ステップ 3 必要に応じて属性設定を変更します。

ステップ 4 クライアント クラスに新しい埋め込みポリシーを追加**Create New Embedded Policy**するには、 をクリックします。編集する既存の埋め込みポリシーがある場合は、**Edit Existing Embedded Policy**をクリックします。どちらの操作でも、[クライアントの DHCP 埋め込みポリシーの編集] ページが開きます。(このページは、[クライアント クラスの DHCP 埋め込みポリシーの編集] ページとほぼ同じです)。

a) [クライアント用 DHCP 組み込みポリシーの編集] ページのフィールド、オプション、および属性を変更します。たとえば、[DHCPv4 オプション] の下で、ドロップダウン リストから**dhcp-lease-time [51]**[リース期間] を選択してクライアントリース期間を設定し、[値] フィールド**Add Option**にリース間隔の値を入力して、 をクリックします。必要に応じて、属性値を設定解除します。

既存の埋め込みポリシーを設定解除する場合**Unset**は、[DHCP クライアントの編集] ページをクリックします。これにより、ボタンが に**Create New Embedded Policy**リセットされます。

ステップ 5 [保存（Save）] をクリックします。

CLI コマンド

クライアントに対して既に設定されている埋め込みポリシー値があるかどうかを確認**client-policy**するには、 *client-name***show**を使用します。埋め込みポリシーを作成**client-policy**するには、クライアント名**set**属性=値を使用します。

これらの DHCP オプションを設定するには、次のコマンドのいずれかを使用します。

```
nrcmd> client-policy client-name setOption <opt-name | id> value [-blob] [-roundrobin]
nrcmd> client-policy client-name setV6Option <opt-name | id>[.instance] value [-blob]
[-roundrobin]
nrcmd> client-policy client-name setVendorOption <opt-name | id> opt-set-name value
[-blob]
nrcmd> client-policy client-name setV6VendorOption <opt-name | id> opt-set-name value
[-blob]
```

リース時間を設定するには**client-policy**、クライアント名**setLeaseTime**の値を使用します。

DHCPv6 クライアントの設定

DHCPv6 クライアントを構成できます。

ローカル アドバンスド Web UI

メニューから**Design**サブ**Clients**メニューの下を**DHCPSettings**選択して、[DHCP クライアントの一覧/追加] ページを開きます。既存のクライアントを選択して [DHCP クライアントの編集]**Add Clients**ページを開くか、[クライアント] ペインのアイコンをクリックして新しいクライアント クラスを追加し、設定された DHCPv6 属性DHCPv6 クライアントクラスの設定 （3 ページ）を含むクライアント クラスを選択します (を参照**Save)** 。

ヒント　DHCP サーバーの検証クライアント名*-mac*属性を無効にします。

CLI コマンド

既存**clientlist**の**client**クライアントを表示するには、または名前**show**を使用します。クライアントのクライアント クラス名を設定するには**client** 、*name* **set client-class-name=** *value*を使用します。また、DHCP サーバーに対して、検証クライアント名*-as-mac*属性が無効になっていることを確認します。

Windows クライアント のプロパティの設定

Windows クライアントは、クラスベースのプロビジョニングをサポートします。クライアント クラスの処理に関連する特定のプロパティを設定できます。次のものがあります。

- クライアント・クラス処理のデフォルト・クライアントを判別するために、クライアント・エントリーを検索します。
- ユーザー・クラス ID をクライアント・クラスまたは選択タグにマップします。

- 選択タグ名にクラス ID を追加するかどうかを設定します。

Windows クライアントの設定

Windows クライアント ホストで、**ipconfig /setclassid**クラス ID を設定します。このクライアント ID をクライアント クラスまたは選択タグにマップする場合は、同じ名前を持つ必要があります。次に、**ipconfig /showclassid**を使用して確認します。次に例を示します。

```
DOS> ipconfig /setclassid adapter engineering

DOS> ipconfig /showclassid adapter
```

DHCP サーバーの設定

DHCP サーバーで Windows クライアントのプロパティを設定する必要があります。

ローカル・クラスターのWeb **dhcp set** UI または CLI のコマンド属性でDHCP サーバー属性を使用して、サーバーの Windows クライアント・プロパティーを設定します。スキップ クライアントルックアップ属性を true に設定した場合 (デフォルトは false)、DHCP サーバーはクライアントクラス処理のためにクライアントエントリをスキップします。（クライアントクラスのクライアント エントリのスキップ （14 ページ） を参照）。マップ ユーザークラス *ID*属性設定のいずれかを使用します。

- **0**- ユーザー クラス ID を無視します(デフォルト)。
- **1**- ユーザークラス ID を選択タグにマップします。
- **2**- ユーザー クラス ID をクライアント クラスにマップします。
- **3**- 選択タグのリストにユーザー クラス ID を追加します。

クライアントクラスのクライアント エントリのスキップ

不要なデータベースの読み取りを防ぐために、クライアント クラスのクライアント エントリを優先する必要はありません。これを実現するには、スキップクライアントルックアップDHCP サーバー属性**dhcp enable skip-client-lookup** (CLI) を有効にします。

クライアント認証の制限

デフォルトでは、クライアントエントリは無制限の認証を取得します。*authenticate-until*属性を使用すると、有効期限を指定してクライアント・エントリーの認証を制限できます。

クライアント エントリが認証されなくなった場合、DHCP サーバーは、この DHCP 要求の応答に使用するクライアント クラス エントリの名前に、認証されていないクライアント クラス名属性値を使用します。この属性が設定されていない場合、またはクライアント クラスのエントリが存在しない場合、DHCP サーバーは要求を無視します。

有効なクライアント認証値は次のとおりです。

- —num が 10 進数で、単位が秒、分、時間、日、週の場合は、以降の時間です。 **+num unit** たとえば、"+3w" は 3 週間後です。

- **date**—月、日、24 時間、2 桁または 4 桁の年。たとえば、「2002年6月30日20:00:00」とします。ローカル プロセス時間を入力します。サーバーが別のタイム ゾーンで実行されている場合は、タイム ゾーンを無視して、代わりにローカル時刻を使用します。
- **forever**— このクライアントの認証を期限切れにしません。

認証対象のクライアントと認証されていないクライアントを区別するために、*authenticate-until* 属性を使用する例を次に示します。認証の期限が切れ、クライアントが別のアドレスを要求すると、DHCPサーバーはクライアントに認証されていないスコープ範囲のアドレスを割り当てます。

ステップ 1 認証済みおよび認証されていないクライアント クラスを作成します。必要に応じて、それぞれの選択基準を設定します。

ステップ 2 クライアントを作成し、認証期限の有効期限を含めます。必要に応じて、クライアント クラス名属性と認証されていないクライアント クラス名属性を設定します。

ステップ 3 認証されたスコープと認証されていないスコープを作成し、アドレス範囲を定義し、それぞれの選択タグに結び付けます。

ステップ 4 サーバーのクライアントクラス処理を有効にします。

ステップ 5 必要に応じて、DHCP サーバーをリロードします。

クライアントのキャッシュ パラメータの設定

DHCPサーバーからのアドレスに対するクライアントからの最初の要求は、多くの場合、DHCPディスカバー-DHCPオファー-DHCP要求-DHCPACK サイクルを通過します。このプロセスでは、サーバーがクライアント データの要求ごとにデータベースを 2 回調べなければなりません。クライアント キャッシュ パラメータが設定されている場合、DHCP サーバーはクライアントデータをメモリにキャッシュして、データベースを1回だけ参照する必要があります。クライアント・キャッシングを使用すると、クライアント情報をLDAPに保管するシステムのパフォーマンスが大幅に向上します。適用可能な属性を設定解除しない限り、クライアントキャッシュは既定で有効になっています。

クライアント要求の予想レートに基づいて、最大キャッシュ数と存続時間(TTL)パラメータを調整できます。要求の猛攻撃が予想される場合は、使用可能なメモリに基づいてキャッシュ数を上限まで増やしたい場合があります。要求サイクルが長くなると予想される場合は、TTLを増やしてください。目的は、要求サイクル中にサーバーがクライアントキャッシュを1回参照するようにすることです。

サーバーがクライアントキャッシュに保持するエントリ数の制限を設定するには、[DHCPサーバーの編集] ページ**dhcpsetclient-cache-count**または CLI でクライアントキャッシュカウント属性を使用します。デフォルトでは、キャッシュする最大数は 1000 クライアントです。キャッシュを無効にするには、属性を 0 に設定します。

通常、クライアント キャッシュはキャッシュ TTL と呼ばれる 10 秒間だけ有効です。TTL の有効期限が切れると、サーバーは必要に応じてデータベースからクライアント情報を読み取りま

す。TTL は、[DHCP サーバーの編集] ページ**dhcpsetclient-cache-ttl**または CLIのクライアントキャッシュ *ttl*属性を使用して調整できます。

クライアント キャッシュ数が指定された最大値に達すると、クライアント エントリ TTL が期限切れになるまで、サーバーはクライアントをキャッシュできません。

DHCPサーバーは、デフォルトでは、DISCOVERメッセージの処理中にのみクライアントデータをキャッシュします。REQUEST（更新またはリバインド）メッセージ中にクライアントデータをキャッシュする場合は、*cache-client-for-requests* 属性を true に設定する必要があります。この属性は、[DHCP サーバーの編集] ページで設定するか、または CLI で DHCP セット の**キャッシュ クライアントの要求を**使用して設定できます。この属性は、2 つの REQUEST (リニューアルまたは再バインド) メッセージ間の存続期間がキャッシュ TTL より短い場合にのみ true に設定する必要があります。

オプション 82 を使用したサブスクライバの制限

多くの場合、サービス プロバイダは、DHCP サーバーが顧客の設置型のデバイスに提供するIPアドレスの数を制限します。これらのデバイスは、DHCPサーバーが提供する"実アドレス"を持ち、その数を制限することを望んでいます。1つの方法は、クライアントクラスを使用して各顧客デバイスを登録(またはプロビジョニング) して、サーバーがクライアント/エントリ データベースに登録されているデバイスにのみ IP アドレスを発行するようにすることです。このアプローチの主な欠点は、MAC アドレスを知る必要があるすべての顧客デバイスを登録する必要があります。サービスプロバイダは、各デバイスについて知りたいとは思わないが、顧客ごとにデバイスの数が多すぎるという点が多すぎるという点が多い。

別のアプローチは、DHCP リレー エージェントが DHCPDISCOVER メッセージで送信するリレー エージェント情報DHCP オプション (RFC 3046 で説明されているオプション 82) の値に基づいて、加入者ごとに顧客デバイスを制限することです。このオプションには、お客様のデバイスが接続されているスイッチのポートに関するデータが含まれます。ケーブルモデムシナリオでは、オプション 82 サブオプションの 1 つに、通常、DHCP 要求がケーブル モデムの外に接続されたデバイスから来る場合、ケーブル モデムの MAC アドレスが含まれています。一般に、オプション 82 データを生成する多くのデバイスは、サブオプションに値を置き、その値が同じアップストリームデバイス上のサブスクライバごとに変化します。場合によっては、この値は、すべての可能なサブスクライバ(ケーブル モデムの MAC アドレスなど)で一意です。その他の場合は、スイッチ上のポートになることができ、そのスイッチに接続されている他のサブスクライバ全体で固有のポートになります。ただし、スイッチ上のすべてのサブスクライバで一意であるとはいえない場合があります。

この方法を使用すると、ネットワーク管理者は、他のDHCPサーバーの機能に重大な影響を与えることなく、DHCP割り当てアドレスの加入者の使用に関する制限を構成できます。多くの環境では、ネットワーク管理者は、デバイスのクラスによってはオプション82制限を使用し、他のクラスには使用しない場合があります。このサポートの重要な側面は、ネットワーク管理者がオプション 82 制限を使用するデバイスと使用しないデバイスを分離できるようにすることです。

サブスクライバ制限への全般的なアプローチ

クライアント処理の現在のアプローチは、クライアントエントリデータベース内のすべてのクライアントを検索することです。オプション 82 制限の目標の 1 つは、クライアント・エントリー・データベース (CNRDB または LDAP のいずれか) 内のすべての顧客デバイスを明示的に登録(プロビジョニング)する必要性を取り除く方法です。ただし、サブスクライバーが制限されている特定の番号を構成し、すべての未登録のサブスクライバーに与えられた既定の番号を上書きする必要があります。

（注） DHCPv6 クライアントでは、制限処理は現在利用できません。

大まかに言えば、サーバーが各着信パケットについて評価し、クライアントを行くクライアントクラスの名前を返す式を作成することによって、加入者制限を設定できます (式の使用式の使用方法の詳細については、「」を参照)。各クライアントクラスは、制限識別子(ID)、サーバーが着信パケットから決定し、実際にデバイスの数を制限するために後の処理で使用するキーの指定を可能にします。サーバーは、同じ制限 ID (制限 *id*プロパティ) を持つすべてのデバイスが同じサブスクライバーから取得されるとみなします。

一般的な制限シナリオ

たとえば、着信パケットは次のように評価されます。

1. オプション 82 の*remote-id*サブオプションがクライアントのハードウェア アドレス (*chaddr*) と一致する**cm-client-class**場合、サブスクライバはケーブル モデムであり、 .
2. *dhcp* クラス識別子オプションの最初の 6 バイトが文字列**docsis**に一致する場合、サブスクライバは DOCSIS モデム**docsis-cm-client-class**であり、 .
3. ユーザークラスオプションの値が文字列**alternative-class**と一致する場合は、サブスクライバー**alternative-cm-client-class**が に含まれる必要があります。

クライアントクラスの計算とキーの作成

DHCP サーバーのクライアント クラス-*lookup-id*属性、または**dhcpsetclient-class-lookup-id=**CLI の式のクライアント クラスを決定する式を設定します。属性定義で参照されるファイルに、属性定義に単純式を含めるか、より複雑な式式の使用方法を含める (を参照)。

クライアントとクライアントクラスでは、クライアントまたはクライアントクラスに対して制限 *ID*値を指定することもできます。サーバーはこの ID 値を使用して、同じネットワークまたは LAN セグメント上で同一 ID を持つデバイスの数に対するアドレス制限を設定します。要求側のクライアントがその ID に対して使用可能なアドレスの制限を超える場合、サーバーはそれを制限超過クライアント・クラス名(設定されている場合)に割り当てます。それ以外の場合は、パケットをドロップします。制限 *ID*は、実質的に、サブスクライバーを定義します。

クライアントクラスの検索式の処理

最初のクライアントクラスルックアップでは、クライアントが何らかの制限に参加するかどうかを決定できます。クライアント クラス検索 *ID*属性を使用して、式サーバー全体を構成します。サーバーは、パケットのクライアントクラスを決定することを目的として、すべての着信パケットに対してこの式を実行します。

この式は、パケットのクライアントクラス名である文字列、またはクライアント要求に対してクライアント クラスの値が考慮されなくなったことを示す識別文字列 <none> を返す必要があります。<none> 文字列を返すことは、クライアント クラスルックアップ *ID*値を構成しないことと同じであり、クライアント クラスの処理は行われません。式が null を返すか、クライアントクラスルックアップ*ID*を評価するエラーが発生した場合、サーバーはパケットを (付随するログ メッセージとともに) ドロップします。

制限の処理

DHCP サーバーは、同じネットワークまたは LAN セグメント内で同じ制限 *ID*値を持つ DHCP クライアントに割り当てられる IP アドレスの数を制限します。サーバーがクライアントに別のアドレスを割り当てることで制限を超える場合、クライアント パケットはオーバーフロー クライアントクラスに配置されます(指定されている場合)。これにより、構成された制限を超えるクライアントに対して特別な処理が可能になります。これらのクライアントを何らかの自己プロビジョニング方法で処理することは、ハードウェアではなくDHCPサーバーの制限を使用する利点の 1 つです (サポートされている場合もあります)。

クライアントクラスに制限超過がない場合、サーバーはパケットをドロップし、そのパケットのアドレス割り当てがその制限*id*の制限カウントを超える可能性があります。サーバーは、単一のネットワークまたはLANセグメントでのみ制限を適用します。ネットワーク マネージャは、一度に1つのLANセグメントを介して接続している1つの加入者を見る傾向があるため、これは制限ではありません。

DHCP ポリシーで、制限数を同一の制限 *ID*で設定します。制限コードは、他のポリシー アイテムと同様に、ポリシー階層の制限数を検索します。つまり、クライアントクラスの埋め込みポリシーまたは名前付きポリシー、スコープの埋め込みまたは名前付きポリシー、またはシステム*system_default_policy*で制限カウントを構成できます。

クライアントクラスで制限 *ID*を設定すると、クライアントクラスの制限処理を追及するように合図されます。制限 *ID*を設定しない場合は、それを追求しないように信号を送ります。式を実行して制限 *id*を判別する場合、式が null を返す場合、このシグナルは、制限処理が行われ、リース状態データベースに保存されている制限 *id*を使用する必要があります。

サブスクライバ制限のための式処理

式は、制限処理の複数の場所に存在します。各式は、null または文字列 (通常はクライアントクラスを検索するときにクライアント クラス名を決定する) または制限 *id*を作成するときに一連のバイト (BLOB) に評価されます。式は、次の場所で使用できます。

- クライアントクラスの検索

- 同じサブスクライバのクライアントを制限するキーの作成 (制限 *id)*
- クライアント・エントリー・データベース (クライアント・ルックアップ *ID)*で検索するキーを作成する。

オプション 82 制限の設定

ステップ 1 クライアントを明示的に登録しない場合は、オプション 82 データを使用する場合は、DHCP サーバー プロパティとしてクライアント クラスを有効にしないでください。

ステップ 2 クライアントの数を制限し、他のクライアントを制限しないかを決定します。一部のクライアントを制限する場合は、次の手順を実行します。

a) 各クラスのクライアントからの DHCP 要求に含まれる値に基づいて、これらのクライアントを他のクライアントと区別する方法を見つけます。

b) 制限のないクライアントを配置するクライアント クラスの名前と、これらの無制限のクライアントに使用する選択タグとスコープを決定します。

ステップ 3 制限超過のクライアントを別のクライアント クラスに配置するか、単にパケットをドロップするかを決定します。クライアント クラスを制限超過にする場合は、クライアント クラス名と、超過クライアントを配置する範囲と選択タグとスコープを決定します。

ステップ 4 制限するクライアントを配置するクライアント クラスと、これらのクライアントに使用する選択タグとスコープを決定します。

ステップ 5 これらすべての選択タグ、クライアントクラス、およびスコープを作成します。

ステップ 6 ポリシー内の制限カウント(クライアントクラスに関連付けられた名前付きポリシー) を構成して、クライアントが制限する。

ステップ 7 入力するクライアントを制限するクライアントと制限されないクライアントに分離する式を記述します。クライアント クラス検索 *ID*属性を設定して、DHCP サーバー上で構成します。

ステップ 8 制限するデバイスの制限 ID を決定する式を記述し、クライアント クラスで制限id を設定して制限するようにクライアント クラスで構成します。

オプション 82 制限のリース更新処理

DHCP クライアントがブロードキャストするパケットのみが、オプション 82 データが付加されたサーバーに到着します。BOOTP または DHCP リレー エージェントは、クライアント デバイスから最初のアップストリーム ルータにオプション 82 データを追加します。DHCPRENEW パケットはサーバーにユニキャストされ、オプション 82 データなしで到着します。これにより、サブスクライバの制限をサーバーに構成するときに問題が発生する可能性があります。

更新を処理する場合、一般的に 2 つの方法があります。

- オプション 82 データを持たないパケットはすべて、関連する選択タグのないクライアントクラスに配置します。これはワイルドカード選択と同等であり、オプション 82 データのないパケットは受け入れられることを意味します。

- オプション 82 データを持つパケットを配置し、その制限 *id*を null と評価する場合と同じクライアント クラスに DHCPRENEW を配置します。これは、制限をチェックする際に、パケットから 1 つではなく、以前に保存された制限 *ID*を DHCP サーバーが使用する必要があるというシグナルです。

どちらのアプローチも機能します。2つ目の方が安全ですが、実際には最初のものよりはるかに優れているわけではありません。これは、DHCP サーバーが DHCPRENEW に応答するために IP アドレスを使用する必要があり、ほとんどのクライアントはサーバーの状態の一部を失わない限り、このアドレスを使用しないためです。この場合、クライアントにアドレスを与える必要があります。悪意のあるクライアントの場合、サーバーをクライアントにアドレスを渡すためにアドレスを使用する必要があり、それによってこのケースの公開を制限します。

オプション 82 制限の管理

制限 *id*を持つクライアント クラスに含まれるクライアントが制限に関与している場合は常に、クライアント データ ログが発生するたびに、使用される制限 ID が DHCP ログ ファイルに表示されます。LID: *nnn* :*nnn* :*nnn* .. データは、現在制限カウントの 1 つを占有しているアクティブなリースを持つクライアントに対してのみ記録されます。

サブネット内の制限 *ID*を使用して、すべてのクライアントを決定できます。[DHCP サーバーの管理]ページで、[コマンド]列の [**実行**] アイコンをクリックして、[DHCP サーバーコマンド]ページを開きます。[IP アドレス] フィールドに、現在アクティブなリースの IP アドレスを入力してから、[**実行**]アイコンをクリックします。また、*limitation-id* 自体を *nn*:*nn*:*nn* の形式で入力するか、または文字列（"*nnnn*"）として入力することもできます。この場合、IP アドレスが検索対象のネットワークになります。CLI で、次**dhcp limitationList**を使用します。

```
nrcmd> dhcp limitationList ipaddr [limitation-id] show
```

*ipaddr*と制限 *id*の両方を指定すると、サーバーは、サブネットを決定するために、*giaddr*と同じようにそれを使用します。ネットワークのスコープ(プライマリまたはセカンダリ)に表示される可能性のある任意の IP アドレスを使用して、サブネットを指定できます。*ipaddr*だけを指定する場合は、DHCP サーバーが提供するアドレスを指定する必要があり、コマンドは、すべてのクライアントと、そのクライアントが使用するリースを返します。

制限カウントのオーバーフローによりクライアントがサービスを拒否された場合、DHCP サーバーのログ ファイルに次のようなメッセージが表示されます。

```
Warning Server 0 05646 Could not add Client MAC: '1,6,01:02:03:04:0c:03'  with
limitation-id: 01:02:03 using Lease: 10.0.0.23, already 3 Clients with that id.
No over-limit client class specified! Dropping packet!
```

制限**dhcp**カウント**limitationList**を超えて拡張されるクライアントを特定できます。コマンドの *ipaddr*値は「リースを使用する:」値にし、制限 id はログ ファイル内の "制限 id:" 値にする必要があります。ログ・ファイルの例を使用すると、コマンドは次のようになります。

```
nrcmd> dhcp limitationList 10.0.0.23 01:02:03 show
```

オプション 82 制限のトラブルシューティング

制限サポートをデバッグする方法はいくつかあります。最初に、DHCPサーバーのデバッグ値を**VX=1**(または**dhcp setDebug VX=1**を使用して)に設定して、パケット トレースを有効にする必要がある場合があります。(デバッグ**VX=0**値はパケットトレースを無効にします。次に、クライアント基準処理とクライアント詳細をログ設定に追加して、クライアントクラスのデバッグを有効にする必要があります。

サーバー全体の式トレース レベル、式トレース レベルもあり、さまざまなレベルに設定できます。6 に設定すると、式の評価の詳細なトレースが表示されます。この処理は、ログに少しのスペースを要し、サーバーの速度も大幅に低下しますが、式の評価に慣れる過程で非常に貴重です。デバッグ式を参照してください。

問題が変わったように見える場合や、ログファイルを送信して問題を報告する場合は、DHCPサーバーのデバッグ値を**QR57=9**(**dhcpsetDebugQR57=9**または を使用して) 設定して、追加のトレースを有効にすることが重要です。(デバッグ**QR57=0**値はこのトレースを無効にします)。Q と R はどちらも大文字であることに注意してください。Q はクライアントクラスのデバッグで、R は応答デバッグです(ログ内の制御フローをクリアするために必要)。5 は式処理であり、7 はクライアント・クラス・ルックアップ処理です。これにより、パケットごとに 1 ページほどの出力が生成され、サーバー内で何が起こっているのかを理解するのに役立ちます。

式の例

式を使用して、サブスクライバーにリースされるIPアドレスを制限する を参照してください。

LDAP を使用するように Cisco Prime Network Registrar を設定する

ライトウェイト ディレクトリ アクセス プロトコル(LDAP)は、Cisco Prime Network レジストラークライアントとリース情報を統合するためのディレクトリ サービスを提供します。LDAP ディレクトリに格納されているオブジェクトの既存の標準スキーマを構築することで、DHCP クライアントエントリに関する情報を処理できます。したがって、DHCPサーバーデータベース内のクライアント情報を維持する代わりに、Cisco Prime Network レジストラー DHCP サーバーに対して、DHCP クライアント要求に応答するデータのクエリを 1 つ以上の LDAP サーバーに発行してもらうか、リース データを LDAP サーバーに書き込むことができます。

Cisco Prime Network Registrar は、Linux で使用可能な OpenLDAP クライアントを使用します。

LDAP ディレクトリ サーバーについて

LDAP ディレクトリ サーバーは、属性/値ペアのコレクションに名前を付け、管理し、アクセスする方法を提供します。Cisco Prime Network レジストラーは特定の LDAP オブジェクト クラスまたはスキーマに依存しないため、LDAP サーバーに情報をいくつでも入力できます。

- DHCPクライアント情報は、使用されていない属性に格納できます。たとえば、指定された名前属性を使用して、DHCP クライアントクラス名の値を保持できます。
- LDAP スキーマ検査を無効にした場合、LDAP スキーマを変更せずに、オブジェクト・クラスに新しい属性を追加できます。たとえば、組織の人物オブジェクトクラスにクライアント クラス名属性を追加できます。
- 新しいオブジェクトクラスを作成し、適切な属性を定義できます。たとえば、DHCPクライアント オブジェクト クラスを作成し、使用するクライアント属性を定義できます。

LDAP から読み取るように DHCP サーバーを構成すると、照会辞書は照会する LDAP 属性をサーバーに指示します。サーバーは、結果のデータをDHCPクライアントデータ属性に変換します。

ヒント　LDAPサーバーが応答を停止したり、DHCPサーバーからの要求に応答を再開したりしたときに SNMP トラップを生成するように Cisco Prime Network レジストラーを設定できます。

LDAP リモート サーバーの追加と編集

LDAP サービスの使用を開始するには、リモート LDAP サーバーを追加する必要があります。

ローカル アドバンスド Web UI

メニューから**DeployLDAP[DHCP]**サブメニューの下で[LDAPリモートサーバーのリスト/追加]ページを開きます。**Add LDAP** [LDAP] ペインのアイコンをクリックして、[DHCP LDAP サーバーの追加] ダイアログ ボックスを開きます。リモート サーバーを編集するには、[LDAP] ペインで LDAP を選択し、[LDAP リモート サーバーの編集] ページを開きます。

このページでは、LDAP サーバーの名前と完全修飾ドメイン名または IP アドレス (IPv4 または IPv6) を少なくとも指定する必要があります。操作を正常に実行するには、ユーザー名とパスワードが必要です。

（注）　クエリ設定と作成設定は、ローカルで DHCP リースの作成に使用されます。

CLI コマンド

ldap *name* **create** *domain-name* を使用します。次に例を示します。

```
nrcmd> ldap ldap-1 create ldap.example.com
```

IP アドレス**ldap server**(IPv4 または IPv6) を使用することもできます。次に例を示します。

```
nrcmd> ldap ldap-1 create 192.0.2.1
nrcmd> ldap ldap-1 create 2001:DB8:1::1
```

LDAP での DHCP クライアント クエリの設定

LDAP クライアントエントリでは、DHCP クライアントクエリの設定とプロビジョニング解除、および組み込みポリシーの設定ができます。

DHCP サーバーから LDAP へのクライアント クエリの設定

DHCP サーバーがクライアント データを LDAP サーバーに照会できるようにするには、次の手順を実行します。ローカル クライアント エントリと同様に、LDAP クライアント エントリはクライアントの MAC アドレスによってキー設定されます。

（注） LDAP サーバーに接続する場合は、ユーザーの識別名(*dn*) を使用します。LDAP スキーマ内のオブジェクトを一意に識別し、データベース内の一意キーまたはファイルの完全修飾パス名に似ています。たとえば、人の dn は dn: cn=ベス・ジョーンズ、ou=マーケティング、o=サンプル・コーポレーションです。この会社には、ベスという名前の人やジョーンズという名前の人がたくさんいるかもしれませんが、ベス・ジョーンズという名前の人は他に例のコーポレーションでマーケティングで働いていません。

ステップ 1 LDAP サーバーのホスト名を指定します。[LDAP リモートサーバーの追加（Add LDAP Remote Server）] ページで、[名前（name）]フィールドに値を入力します。ローカルCLIで、次のコマンドを使用します。

```
nrcmd> ldap ldap-1 create ldap.example.com
```

後でサーバーを削除する必要がある場合は**ldap** 、*server* を **delete**使用します。

ステップ 2 接続の認証情報を設定します。ユーザーに識別名（*dn*）を使用します。[ユーザー名（username）]フィールドに値を入力します。CLI で、次のコマンドを使用します。

```
nrcmd> ldap ldap=1 set username="cn=joe,o=Example Corp,c=US" password=access
```

ステップ 3 検索パス (および必要に応じて検索範囲) を設定します。パスは、検索を開始するディレクトリ内のポイントです。検索範囲が次の場合:

- SUBTREE を使用すると、サーバーは検索パスのすべての子を検索します。
- ONELEVEL を指定すると、サーバーは基本オブジェクトの直接の子のみを検索します。
- BASE の場合、サーバーはベース オブジェクト自体だけを検索します。

この例では、検索のベースを組織 Example Corp と国 US に設定し、サブツリー検索範囲を設定します。[検索パス（search-path）] フィールドに値を入力します。CLI で、次のようなコマンドを使用します。

```
nrcmd> ldap ldap-1 set search-path="o=Example Corp,c=US" search-scope=SUBTREE
```

ステップ 4 検索フィルタを、DHCP がクライアントの MAC アドレス（DHCPv4 の場合）または DUID（DHCPv6 の場合）に置き換える属性に設定します。この例では、属性は共通名 (*cn*) です。[検索フィルタ（search-filter）] フィールドに値を入力します。CLI で、次のようなコマンドを使用します。

```
nrcmd> ldap ldap-1 set search-filter=(cn=%s)
```

ステップ 5 LDAP と DHCP のマッピングをすべて含むクエリ ディクショナリを設定します。これらのマッピングを設定するには、**ldap** サーバー名 **setEntry**を使用します。

1. *sn* LDAP 属性から DHCP 姓を取得します。

   ```
   nrcmd> ldap ldap-1 setEntry query-dictionary sn=host-name
   ```

2. 最初の名前 LDAP 属性からクライアント・クラス名を取得します。

   ```
   nrcmd> ldap ldap-1 setEntry query-dictionary givenname=client-class-name
   ```

3. ローカルの *LDAP*属性からドメイン名を取得します。

   ```
   nrcmd> ldap ldap-1 setEntry query-dictionary localityname=domain-name
   ```

4. いずれかのエントリを設定解除する必要がある場合は、**ldap** サーバー **unsetEntry** 属性キーを使用します。**また、ldap**サーバーの**getEntry**属性キーを使用して、任意の設定を確認することもできます。

ステップ 6 LDAP サーバーに対する照会を使用可能にします。この例では、*myserver*のクエリを有効にします。*can-query* 属性を enabledに設定します。CLI で、次のコマンドを使用します。

```
nrcmd> ldap ldap-1 enable can-query
```

ステップ 7 DHCP サーバーのクライアントクラス処理を有効にします。[DHCP サーバーの編集（Edit DHCP Server）] ページで、 *client-class* 属性を enabled に設定します。CLI で、次のコマンドを使用します。

```
nrcmd> dhcp enable client-class
```

ステップ 8 DHCP サーバーがクライアント エントリクエリに LDAP を使用できるようにします。[DHCP サーバーの管理] ページで、クライアント クラス属性を有効に設定します。CLI で、次のコマンドを使用します。

```
nrcmd> dhcp enable use-ldap-client-data
```

ステップ 9 複数の LDAP サーバーを構成している場合は、ラウンドロビンモードまたはフェイルオーバーモードで動作するように設定することもできます。

- **ラウンドロビン**-LDAP サーバーのプリファレンス値は無視され、クライアント クエリを処理し、リース状態の更新を受け入れるように構成されているすべてのサーバーが等しく処理されます。
- **フェールオーバー**:DHCP サーバーは、最も優先度の高い(最も低い設定番号)のアクティブ LDAP サーバーを使用します。優先サーバーが接続を失ったり、失敗したりすると、DHCP サーバーは次の低い優先順位の LDAP サーバーを使用します (優先順位が高くなります)。設定値が同じ (または設定されていない) 場合、DHCP はこれらのサーバーとのラウンドロビン モードに戻ります。

[DHCP サーバーの編集] ページで LDAP モードを設定して、LDAP サーバー モードを設定します。LDAP フェールオーバー モードは、実際には優先的なロード バランシングを実行します。DHCP サーバーは、LDAP 接続とエラー状態、および LDAP サーバーの応答速度を評価します。最適な状態では、DHCP サーバーは、最も高い優先順位 (最も低い優先順位番号) を割り当てた LDAP サーバーを使用します。最適ではない状態では、DHCP サーバーは、次の低い優先順位の LDAP サーバーを使用します (優先順位の数が増加します)。設定値が同じ (または設定されていない) 場合、DHCP サーバーはラウンドロビン モードに戻ります。

CLI で、**dhcp set ldap モード**を使用してモードを設定し、**ldap** サーバーが**設定**設定してサーバーの基本設定を設定します。例えば：

```
nrcmd> dhcp set ldap-mode=failover
nrcmd> ldap ldap-1 set preference=1
nrcmd> ldap ldap-2 set preference=2
```

また、DHCP サーバーと LDAP サーバー間の接続属性 (を参照LDAP の推奨値 （35 ページ）) を使用して設定した、開いているスレッドの数によっては、DHCP サーバーは、クエリ タイムアウトが切れる前に、できるだけ多くのスレッドを開くだけであることに注意してください。LDAP サーバーがこれらのスレッドを処理している可能性がありますが、フェイルオーバー・サーバーが引き継いだため、要求を処理していません。

ステップ 10 DHCP サーバーがクライアント エントリクエリに LDAP を使用できるようにします。[DHCP サーバーの管理（Manage DHCP Server）] ページで、*client-class* 属性を enabled に設定します。CLI で、次のコマンドを使用します。

```
nrcmd> dhcp enable use-ldap-client-data
```

ステップ 11 LDAP 構成を表示または一覧表示します。[LDAP リモート サーバーの一覧/追加] ページに移動します。CLI で、次のコマンドを使用します。

```
nrcmd> ldap ldap-1
nrcmd> ldap list
nrcmd> ldap listnames
```

ステップ 12 DHCP サーバーをリロードします。

（注） DHCP サーバーは通常、**%s** をクライアントの MAC アドレス（DHCPv4 の場合）または DUID（DHCPv6 の場合）に置き換えます。ただし、他のクライアント指定子を使用できます。他のクライアント指定子（拡張によって生成されるなど）を使用する場合は、文字列を使用して LDAP インジェクションを実行できないようにしてください。これは、クライアントから送信されたデータによって次の文字が挿入されないようにするため、または、可能であれば次の文字列が適切にエスケープされるようにする必要があるためです。

カンマ（,）、バックスラッシュ文字（\）、ポンド（ハッシュ）記号（#）、プラス記号（+）、小なり記号（>）、セミコロン（;）、二重引用符（"）、等号記号（=）、および先頭または末尾のスペース

場合によっては、他の文字が問題になることもあります（LDAP サーバーまたは RFC 4514 で確認してください）。受信パケットのデータを使用する場合、問題になることがあります。DHCP サーバーでは、指定された文字列は変更されません。提供された文字列がそのまま安全に使用できることを前提としています。

クライアント エントリのプロビジョニング解除

LDAP クライアント情報が LDAP に残るように LDAP クライアントエントリをアンプロビジョニングできますが、DHCP サーバーはクライアントをその情報が存在しないものとして扱います。DHCP サーバーは、クライアントにデフォルトの動作を提供します。LDAP サーバーが値を持つ指定DHCP サーバーから LDAP へのクライアント クエリの設定 （23 ページ） された属

性を含むクライアントエントリを返さないように、前のセクションのステップ4で検索フィルタ セットを設定します。

LDAP エントリ *givenname* のプロビジョニングを解除する場合は、それに応じた検索フィルタを設定します。次に例を示します。

```
nrcmd> ldap ldap-1 set search-filter=(&(cn=%s)(!(givenname=unprovision)))
```

LDAP クライアント エントリの指定された名前属性が"準備解除"文字列に設定されている場合、LDAP サーバーはクライアント エントリを DHCP サーバーに返しません。つまり、DHCP サーバーは、クライアントを LDAP クライアント エントリがないかのように扱います。この手順では、DHCP サーバーまたは LDAP サーバーに対してパフォーマンスに対する測定可能な影響はありません。

LDAP での埋め込みポリシーの設定

ステップ 1 たとえば、LDAP サーバーを構成し、そのサーバーに my サーバーという名前を付けます。

ステップ 2 DHCP サーバーが組み込みポリシーとして解釈する LDAP 属性を、内部組み込みポリシー プロパティにマップします。この例では、ビジネス カテゴリ LDAP 属性をマップします。

```
nrcmd> ldap myserver setEntry query-dictionary businessCategory=embedded-policy
```

ステップ 3 DHCP サーバーが組み込みポリシーとして解釈できる LDAP 属性に文字列を追加します。この文字列の外観を決定する最も実用的な方法は、Cisco Prime Network レジストラーデータベースにダミー クライアントを作成し、クライアントの組み込みポリシー設定からデータを抽出することです。このダミー クライアントは、LDAP を使用しているため、使用されることはないので、後で削除できます。必要なオプションデータタイプを埋め込みポリシーに含めます。

1. たとえば、ダミー クライアント 1,6,00:d0:ba:d3:bd:3b 用の組み込みクライアント ポリシーを作成します。応答オプションと、IP アドレスデータタイプの複数値オプション (ルーター) を追加します。

   ```
   nrcmd> client 1,6,00:d0:ba:d3:bd:3b create
   nrcmd> client-policy 1,6,00:d0:ba:d3:bd:3b set v4-reply-options=routers
   nrcmd> client-policy 1,6,00:d0:ba:d3:bd:3b setOption routers 1.2.3.4,5.6.7.8
   nrcmd> save
   ```

2. 値を表示できるように、クライアントの埋め込みポリシー データを取得します。

   ```
   nrcmd> client 1,6,00:d0:ba:d3:bd:3b get embedded-policy
   100 Ok
   embedded-policy="((ClassName Policy)(name client-policy:00:d0:ba:d3:bd:3b)(option-list [((ClassName
    Option)(number 3)(option-definition-set-name dhcp-config)(value
   01:02:03:04:05:06:07:08))])(v4-reply-options [routers ])"
   ```

3. 前のサブステップのクライアント出力の引用符の間にある内容をコピーし、それを businessCategory LDAP 属性の定義に貼り付けます。

   ```
   businessCategory:((ClassName Policy)(name client-policy:00:d0:ba:d3:bd:3b)(option-list [((ClassName
    Option)(number 3)(option-definition-set-name dhcp-config)(value
   01:02:03:04:05:06:07:08))])(v4-reply-options [routers ])
   ```

4. LDAP の新しい組み込みポリシー エントリごとに、構文をモデルとして使用します。LDAP 文字列内の他のオプション データ型がどのように表示されるか確認するには、これらのオプションをクライア

ントに追加するか、またはクライアントと共にさらにダミー クライアントを作成します。データを抽出したら、ダミークライアントを削除できます。

```
nrcmd> client 1,6,00:d0:ba:d3:bd:3b delete
nrcmd> save
```

LDAP での組み込みポリシーの設定 (複数のオプション定義を使用)

複数のオプション定義を持つ別の例を次に示します。

ステップ 1 ダミー・クライアント 1,6,00:d0:ba:d3:bd:3b およびそのクライアントにアタッチされた埋め込みポリシーを作成します。

```
3 routers 10.1.1.1,10.2.1.1
66 tftp-server tftp-server.com
67 bootfile device-boot-file.txt
```

ステップ 2 埋め込みポリシーへの変更を保存し、クライアントを保存してから、次の出力文字列を LDAP クライアント構成に抽出します。

```
nrcmd> client 1,6,00:d0:ba:d3:bd:3b get embedded-policy
100 Ok
embedded-policy="((ClassName Policy)(name client-policy:00:d0:ba:d3:bd:3b)(option-list [((ClassName
 Option)(number 3)(option-definition-set-name dhcp-config)(value 0a:01:01:01:0a:02:01:01))((ClassName
 Option)(number 66)(option-definition-set-name dhcp-config)(value
74:66:74:70:2d:73:65:72:76:65:72:2e:63:6f:6d))((ClassName Option)(number 67)(option-definition-set-name
 dhcp-config)(value 64:65:76:69:63:65:2d:62:6f:6f:74:2d:66:69:6c:65:2e:74:78:74))])"
```

DHCP LDAP 更新とサービスの作成の設定

Cisco プライム ネットワークレジストラー DHCP サーバーを設定して、リースおよびクライアント データを LDAP サーバーに書き込むことができます。DHCP サーバーは、クエリ構成を使用して、DHCP クライアント要求に応答するときにクライアント データを使用できます。LDAP サーバーのクライアント・オブジェクトの属性にリース状態データをコピーするように DHCP LDAP サービスを構成できます。DHCP サーバーは、リース状態データを文字列形式に変換し、更新ディクショナリを使用して DHCP データ値を LDAP 属性にマップします。

リース状態が変更されるたびに、DHCP サーバーはデータを格納するように構成した LDAP サーバーに変更を書き込みます。DHCP サーバーが LDAP に書き込むリース データは、リース状態データベース内の権限のあるデータのコピーであるという「書き込み専用」です。

リース状態属性

LDAP サーバーにリース状態情報に関する以下の属性を保存できます。

- *address*：このリースの IP アドレス。

- *client-dns-name*：DHCP サーバーがこのクライアントの DNS サーバーに入力しようとした名前。
- *client-domain-name*：クライアント名を配置するドメイン。
- *client-flags*：クライアントに関連するさまざまなフラグ。
- *client-host-name*：クライアントが DNS サーバーに配置するように DHCP サーバーに対して要求した DNS 名。
- *client-id*：クライアントによって指定されたクライアント ID。またはこのクライアントの DHCP サーバーによって合成されたクライアント ID。
- *client-mac-addr*：クライアントが DHCP サーバーに提示した MAC アドレス。

（注） LDAP の MAC アドレスは、ローカル クライアント エントリを作成するときに Cisco Prime Network レジストラーによってフォーマットされるとおりにフォーマットする必要がありますが、それらは個別のインスタンスであり、リース データに固有です。

- *expiration*：リースの有効期限が切れる時刻。
- *flags*：リースのフラグ（reserved や deactivated）。
- *lease-renewal-time*：クライアントがリースの更新を発行する予定の最も早い時刻。Cisco プライムネットワーク レジストラーを使用**dhcp enable save-lease-renewal-time**して、リース状態の一部として保存できます(デフォルトでは保存されません)。
- *start-time-of-state*：状態が現在の値に最後に変更された時刻。
- *state*：次のようなリース状態があります。
  - 利用可能 (1)
  - Deferred (2)
  - リース (3)
  - Expired (4)
  - Unavailable (5)
  - リリース済み (6)
  - Other_available (7)
  - Disconnected (8)
  - 削除済み (9)
- *vendor-class-identifier*：ベンダー固有の情報を交換するためにクライアントとサーバーが使用するベンダーの名前。

すべてのリースにこれらすべての属性があるわけではありません。クライアントがリースを解放するか、Cisco Prime *Network* レジストラー*IP Express*を通じて強制的に利用可能にされる場

合、クライアント-mac-addrおよびクライアントのリース状態属性は存在しません。また、DHCP を使用してリース更新の保存時プロパティが無効になっている場合、リース更新時間属性が存在しない場合があります。同様に、ベンダクラス識別子プロパティは、DHCP を使用して *SAVE-Vendor-class-id*プロパティが無効になっている場合は、CLI を使用して存在しない場合があります。

LDAP にリース状態を書き込むための DHCP の設定

DHCP 書き込みリース状態を LDAP に更新するには、次の手順を実行します。

ステップ 1 LDAP リース状態更新スキームを選択します。

ステップ 2 ディレクトリにエントリを追加するか、リース状態情報を格納する既存のエントリを変更します。属性またはカスタム オブジェクト クラスを追加してエントリを拡張する必要がある場合があります。

ステップ 3 更新を実行するには、Cisco プライムネットワーク レジストラーを設定します。

ディレクトリの柔軟性を考えると、ディレクトリにリース状態属性のコピーを格納する方法はさまざまです。たとえば、リース状態データを既存のエントリの一部として格納するか、リース状態データを個別に保存することができます。

既存のエントリの一部としてリース状態データを保存

リース状態データは、既存のエントリの一部として格納できます。クライアントエントリ、リース状態、従業員データを同じエントリに格納することも可能です。このメソッドのセットアップの一部として、リース データ属性の格納方法を決定する必要があります。データ属性は、次の方法で格納できます。

- エントリから属性をマップする
- エントリに属性を追加する
- 新しいオブジェクト クラスを作成してエントリを拡張する

利点は、リース データが他のクライアント情報と共に直接格納されるということです。欠点は、クライアント クラスや予約に関連するシナリオが存在する可能性があり、サーバーがクライアントをリースから移動するときに、ディレクトリ内に古いデータが短時間存在する可能性があることです。

（注） 更新される状態のリースにクライアントがない場合、関連付けられたMACアドレスは存在しません。この状況は、クライアントがリースを取得し、クライアント クラスの処理によってリースから移動された場合に発生します。また、クライアントが既存のリースを持ち、同じ LAN セグメント内の別のリースの予約を行う場合にも発生します。予約済みリースが使用可能な場合、サーバーはクライアントを既存のリースから予約に移動します。これらの転送の両方は、クライアントMACアドレスなしで古いリースのLDAP更新になります。新しいリース(関連MACアドレスを持つ)の更新が行われる必要があるため、これは一般的に問題ではありません。

また、この方法では、リース情報を書き込むために2つのLDAP対話が必要です。DHCP LDAPサービスは、エントリを更新する際にエントリを見つける方法を知るだけでは不十分であるため、リース状態情報を更新する場合、ディレクトリに2回接続します。具体的には、エントリの*dn*を知っている必要があります。

DHCP LDAP サービスは、まず、選択したリース状態属性 (できれば MAC アドレス) を検索条件として使用して、ディレクトリ内の適切なエントリを検索します。これは、リース状態属性のいずれもエントリの*dn*の一部ではないため、必要です。DHCP LDAP サービスがエントリを見つけると、*dn*が返されます。DHCP LDAP サービスは、適切な情報を使用して同じエントリを更新します。このメソッドの使用例については、「」を参照してくださいLDAP状態の更新の設定 （31 ページ）。

リース状態データを個別に保存

IPアドレスによってリース状態データを独自のエントリに格納できます。この方法は、ディレクトリ内のサーバー リース データベースのコピーとなり、データベースを構成する最も簡単な方法です。この方法のセットアップの一部として、サーバーがサービスを提供できる各 IP アドレスに対して新しいエントリを作成します。この方法の利点は、ディレクトリ内のリース状態データが古くなるシナリオが存在しない点です。欠点は、リースデータが他の関連するクライアント情報と直接格納されないことです。

リース状態情報を更新するには、DHCP LDAP サービスがディレクトリ サービスに 1 回接続します。更新を実行すると、サービスは IP アドレスを使用して*dn*を構築します。

LDAP 更新の使用

LDAP 更新機能を使用するには、次の 2 つの方法があります。

- LDAP クライアント・エントリー情報を使用するクライアントを追跡し、その LDAP ホストの属性の一部をリース状態属性に関連付けます。
- IP アドレスで見つけることができるオブジェクトを作成および更新します。Cisco Prime Network レジストラーがこれらのオブジェクトを作成する場合、DHCP サーバーのリース状態に一致する(またはある)LDAP オブジェクトのレベルを作成できます。

Cisco プライムネットワーク レジストラーを使用する場合は、次の点に注意する必要があります。

- DHCPサーバーは、単一のオブジェクトからの読み取りと書き込みのみを行います。クライアントエントリデータの読み取りとリース状態の日付を保持するために別々のオブジェクトを使用できますが、Cisco Prime Network レジストラーは、あるオブジェクトと別のオブジェクトから属性を読み取ることはできません。
- すべてのデータベースアクセスと同様に、LDAPクエリのパフォーマンスは、インデックス付き属性によって異なります。クエリ フィルターで使用するように構成した属性にインデックスを付けていない場合は、パフォーマンスが低下します。
- LDAP 属性は、サーバーのインストール時にLDAP スキーマで事前設定されるか、またはCisco Prime Networkレジストラー以外の他の方法で作成する必要があります。

LDAP 状態の更新の設定

LDAP サーバーに対してリース状態更新を実行するには、次の 2 つのオプションを使用できます。

- **更新検索パス**: DHCP サーバーは、まず更新の*dn*を検索するためにクエリを実行します。
- ***dn-format***—サーバーには、更新用の*dn*が提供されます。つまり、DHCPは更新前にクエリを実行しなくても直接更新を実行します。

オプション 1: update-search-path オプションの使用

次の例は、最初のオプションである更新検索パスを示しています。LDAPオブジェクトの識別名 (*dn*) をリース状態で使用可能なデータから構築できない場合の処理を示します。DHCP サーバーは、更新検索 *xxx*情報に基づいて LDAP クエリを作成し、LDAP オブジェクトを検索し、その *dn*を使用して LDAP 更新を発行します。

次の表に示す例では、標準 LDAP 組織の個人オブジェクト クラス属性を使用して、リース更新データを保持していることを前提としています。

表 *2 : LDAP* と *DHCP* のマッピングの例

属性（Attribute）	DHCP リースエントリマッピング
uid	アドレス（IP アドレス）
カーライセンス	状態 (リース状態)

ステップ 1 LDAP 構成でサーバーのホスト名を指定して、LDAP サーバーについて DHCP に伝えます。

ステップ 2 LDAP サーバーに接続するときに使用するログイン情報を設定します 。この CLI の例では、管理者に joe と、アクセスするパスワードを設定します。ユーザーに識別名（*dn*）を使用します。

```
nrcmd> ldap myserver set username="cn=joe,o=Example Corporation,c=US" password=access
```

ステップ 3 DHCPサーバーが更新するオブジェクトのディレクトリ内の開始点である更新検索パス属性を構成します。また、更新検索の範囲も設定できます。この CLI の例では、組織単位 (ou) IT、組織のサンプル コーポレーション、および国USから開始する検索パスを設定します。更新検索範囲は、サブツリーに設定されます。

```
nrcmd> ldap myserver set update-search-path="ou=IT,o=Example Corp,c=US"
update-search-scope=SUBTREE
```

ステップ 4 更新する LDAP オブジェクトの検索に使用する属性の ID を設定します。次の CLI の例では、検索属性をクライアント MAC アドレスに設定します。

```
nrcmd> ldap myserver set update-search-attribute=client-mac-addr
```

ステップ 5 更新検索属性の書式を設定するフィルタ式を構成します。この式には、検索属性データを置換する場所を示す「%s」を含める必要があります。次は CLI の例です。

```
nrcmd> ldap myserver set update-search-filter=(cn=%s)
```

ステップ 6 *update-dictionary* 属性を設定すると、対応するリース状態属性の値を使用して設定する LDAP 属性を識別できます。この例では、LDAP UID を更新して IP アドレスを含め、カーライセンス属性を更新して DHCP リース状態情報を含める必要があることを指定します。CLI の使用：

```
nrcmd> ldap myserver setEntry update-dictionary uid=address carlicense=state
```

ステップ 7 新しい LDAP サーバーの更新を有効にします。次は CLI の例です。

```
nrcmd> ldap myserver enable can-update
```

ステップ 8 DHCP サーバーをリロードします。

オプション 2: dn-format オプションの使用

この例では、2 番目のオプション*dn-format*を使用する方法を示します。

ステップ 1 LDAP 構成でサーバーのホスト名を指定して、LDAP サーバーについて DHCP に伝えます。

ステップ 2 LDAP サーバーに接続するときに使用するログイン情報を設定します 。この CLI の例では、管理者に joe と、アクセスするパスワードを設定します。ユーザーの*dn*を使用します。

```
nrcmd> ldap myserver_option2 set username="cn=joe,o=Example Corporation,c=US"
password=access
```

ステップ 3 *dn-format* 文字列を使用して、更新の検索を開始する LDAP サーバーのデータベース階層内の場所を指定します。次は CLI の例です。

```
nrcmd> ldap myserver_option2 set dn-format="cn=\"%s\",ou=IT,o=Example Corp,c=US"
```

ステップ 4 *dn-format* 文字列が参照する *dn-attribute* 属性を設定します。次の CLI の例では、*dn* 属性をクライアント MAC アドレスに設定します。

```
nrcmd> ldap myserver_option2 set dn-attribute=client-mac-addr
```

ステップ 5 更新するエントリを指定します。CLI の使用：

```
nrcmd> ldap myserver_option2 setEntry update-dictionary uid=address carlicense=state
```

ステップ 6 *can-update* 属性を有効にします。次は CLI の例です。

```
nrcmd> ldap myserver_option2 enable can-update
```

ステップ 7 DHCP サーバーをリロードします。

LDAP エントリ作成の設定

このセクションでは、LDAP エントリについて説明します。LDAP エントリの作成機能を使用すると、エントリを検索し、現在のリース情報で更新することができます。エントリが作成されるのは、エントリが見つからないために状態更新操作が失敗した場合だけです。

前の例の手順を実行した後、CLI の次の手順を実行します。

ステップ 1 *client-mac-addr* フィールドなどのリースオブジェクト属性の LDAP サーバーに対して *dn-attribute* プロパティを設定し、 *dn-format* 文字列を設定します。次に CLI の例を示します。

```
nrcmd> ldap myserver set dn-attribute=client-mac-addr dn-format="cn=\"%s\",ou=IT,o=Example Corp,c=US"
```

この手順は、更新検索パスオプションを使用してリース状態の更新を構成する場合にのみ必要です。（オプション 1: update-search-path オプションの使用 （31 ページ） を参照）。*dn* フォーマット文字列を使用してリース状態の更新を構成する場合は、この手順をスキップします。（オプション2: dn-format オプションの使用 （32 ページ） を参照）。

ステップ 2 既存の *dn-attribute* プロパティと組み合わせるときに作成するエントリの *dn* を指定します。次は CLI の例です。

```
nrcmd> ldap myserver set dn-create-format="cn=\"%s\",ou=IT,o=Example Corp,c=US"
```

The Cisco Prime Network Registrar *client-mac-addr* フィールドでは、 1,6:*xx*:*xx*:*xx*:*xx*:*xx*:*xx* 形式を使用します。コンマ文字は LDAP の特殊な区切り文字であるため\"、*dn*を引用符で囲むには、その文字を使用する必要があります。

ステップ 3 ディクショナリ作成プロパティーを使用して、一連の名前と値のペアを入力して、LDAP 属性とリース状態属性の間のマッピングを確立します。LDAP 属性は、対応するリース状態の属性の値に設定されたエントリ属性を示します。CLI：

```
nrcmd> ldap myserver setEntry create-dictionary sn=client-host-name

nrcmd> ldap myserver setEntry create-dictionary givenname=client-class-name

nrcmd> ldap myserver setEntry create-dictionary localityname=client-domain-name
```

ステップ 4 *create-object-classes* プロパティを使用して、エントリを作成するときに使用するオブジェクトクラスを指定します。次は CLI の例です。

```
nrcmd> ldap myserver set create-object-classes="top,person,organizationalPerson,inetorgperson"
```

ステップ 5 LDAP サーバーの myserver のエントリ作成を有効にします。次は CLI の例です。

```
nrcmd> ldap myserver enable can-create
```

（注） 属性を作成できる属性を有効にする前に、更新可能属性を有効にします。例については、LDAP 状態の更新の設定 （31 ページ）を参照してください。

ステップ 6 DHCP サーバーをリロードします。

ステップ 7 作成、クエリ、および更新が正常に行われたかどうかを確認するには、LDAP ログの設定を表示します。

LDAP のトラブルシューティング

以下のセクションでは、LDAP サーバーの障害の微調整と検出に関するアドバイスを示します。

LDAP 接続の最適化

個別に微調整が可能な読み取りオブジェクトと書き込みオブジェクトを使用して、LDAP 接続を最適化できます。この CLI の例では、書き込み (作成および更新) 操作を調整し、より長いサーバー処理を必要とします。

```
nrcmd> ldap LDAP-Write create csrc-ldap password=changeme port=389 preference=1

nrcmd> ldap LDAP-Write setEntry query-dictionary csrcclientclasas=client-class-name

nrcmd> ldap LDAP-Write set
search-filter=(&(macaddress=%s)(|(crscclassname=Computer)(csrcclassname=Modem)))

nrcmd> ldap LDAP-Write set search-path=csrcprogramname=csrc,o=NetscapeRoot

nrcmd> ldap LDAP-Write set
username=uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeRoot

nrcmd> ldap LDAP-Write disable can-query

nrcmd> ldap LDAP-Write enable can-create

nrcmd> ldap LDAP-Write enable can-update

nrcmd> ldap LDAP-Write enable limit-requests

nrcmd> ldap LDAP-Write set connections=2 max-requests=8 timeout=10s
```

次の CLI の例では、読み取り (クエリ) 操作を調整します。

```
nrcmd> ldap LDAP-Read create csrc-ldap password=changeme port=389 preference=1

nrcmd> ldap LDAP-Read setEntry query-dictionary csrcclientclasas=client-class-name

nrcmd> ldap LDAP-Read set
search-filter=(&(macaddress=%s)(|(crscclassname=Computer)(csrcclassname=Modem)))

nrcmd> ldap LDAP-Read set search-path=csrcprogramname=csrc,o=NetscapeRoot

nrcmd> ldap LDAP-Read set
username=uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeRoot

nrcmd> ldap LDAP-Read enable can-query

nrcmd> ldap LDAP-Read disable can-create

nrcmd> ldap LDAP-Read disable can-update
```

```
nrcmd> ldap LDAP-Read enable limit-requests

nrcmd> ldap LDAP-Read set connections=3 max-requests=12 timeout=4s
```

LDAP の推奨値

以下の表は、いくつかの重要な LDAP 属性の推奨値を示しています。

表 *3 : LDAP* 属性の推奨値

属性と値	説明
connections =5 to 25	サーバーが LDAP サーバーに対して行う必要がある接続の数。これは、主にパフォーマンス・チューニング・パラメーターです。デフォルト値は 1 接続です。場合によっては、複数の接続によって全体的なスループットが向上することがあります。この量は、LDAP サーバーの負荷によって異なります。LDAP を使用するアプリケーションが多数ある場合は、5 つの接続が適切です。LDAP を使用した Cisco プライムネットワーク レジストラーだけで、25 が適切です。
threadwaittime =2	LDAP クライアント接続が結果をポーリングする間隔 (ミリ秒単位)。
query-timeout =3	Cisco プライムネットワーク レジストラー DHCP サーバーは、フェールオーバーとクエリが設定されている場合は、クエリ タイムアウト間隔でフェールオーバーします。デフォルト設定は 3 秒で、推奨されます (DHCP サーバーのデフォルトの 4 秒のドロップ・オールド・パケット値よりも小さいため、接続が非アクティブで LDAP サーバーが「異常」と見なされます)。
timeout =10	LDAP 要求が接続キューに残っている秒数で、失効とタイムアウトが宣言されます。クライアントのタイムアウト期間の後に DHCP クライアントが受信した応答は、古くなっています。デフォルトは 10 秒で、推奨されます。Cisco Prime Network レジストラー DHCP サーバーは、フェールオーバーと更新可能または作成が有効な場合にタイムアウト間隔でフェールオーバーします。

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。