CHAPTER 15

要求の認証と許可の設定

要求の認証および許可は、社員のインターネット使用を管理し、オンライン コンテンツへのアクセスを制限する手段です。

ここでは、中央管理型 ACNS ネットワークの Content Engine から、コンテンツ サービスの要求認証と許可を設定する方法について説明します。この章の構成は、次のとおりです。

（注） ACNS ネットワークでは、Content Engine がオリジン サーバからコンテンツを取得した際、自身をクライアントとして認証し、またコンテンツ サービスに備えて他も認証する準備をする必要があります。コンテンツ取得の認証は、第 6 章「ACNS ネットワークのコンテンツ取得設定」で説明します（「認証サポート」[p.6-44] を参照）。

（注） コンテンツ要求の認証と許可は、ログイン（ユーザ）の認証と許可とは関係ありません。ログインの認証と許可については、第 12 章「ログイン認証の設定および許可とアカウンティングの設定」を参照してください。

Content Engine への要求の認証と許可の概要

要求の認証と許可は、Content Engine を使用したコンテンツ サービスを制御します。Content Engine はコンテンツをサービスする要求を受信すると、その要求を許可するか拒否するかを決定する必要があります。要求の許可または拒否は、Content Engine によって、または外部認証サービスによって判断されます。要求が許可された場合、Content Engine はそのコンテンツのサービスを引き受けます。

プロキシ認証およびパススルー認証モードの概要

コンテンツ要求認証の決定については、ACNS ソフトウェアは、プロキシ認証およびパススルー認証の 2 つの動作モードをサポートしています。プロキシ認証モードでは、Content Engine はクライアントに証明書を要求し、その応答を受信します。次にその情報を他の認証サービスに提供し、クライアントの要求を許可するか拒否するかを判断します。認証サービスは、設定されたユーザ データベースにその証明書が有効かどうかを問い合わせます。プロキシ認証は、HTTP およびネイティブ FTP（ファイル転送プロトコル）要求で使用できます。

プロキシ認証ヘッダーの 401 および 407 応答メッセージの設定については、「プロキシ サーバ モードの HTTP 要求認証の概要」（p.15-10）および「透過モードの HTTP 要求認証の概要」（p.15-11）を参照してください。

（注）　この章の中では、*HTTP* 要求は HTTP、FTP-over-HTTP、および HTTPS-over-HTTP 要求全般を意味します（「HTTP 要求の認証および許可の概要」[p.15-6] を参照）。

（注）　ネイティブ FTP 要求は、FTP-over-HTTP ではなく FTP プロトコル（FTP-over-FTP）を使用します。ネイティブ FTP はプロキシ認証モードのみで使用できます（「ネイティブ FTP 要求の認証の概要」[p.15-13] を参照）。

パススルー認証モードでは、オリジン サーバがクライアントに証明書を要求し、Content Engine がクライアントにその認証要求を渡します。オリジン サーバが HTTP の基本認証要求を発行した場合、クライアントはエンド ユーザに証明書を要求します。オリジン サーバが Windows NT LAN Manager（NTLM）要求を発行した場合、クライアントは自動的にネットワークのログイン証明書で応答します。次に Content Engine はクライアント要求の応答をアップストリームのオリジン サーバに転送します。転送されたオリジン サーバはその証明書をユーザ データベース（または他のサービス）で検証します。Content Engine は、オリジン サーバの応答コードに基づいた最終的な判断（許可または拒否）を得ます。Content Engine は証明書の収集、評価、検証には関わっていません。

場合によっては、Web サイト（オリジン サーバ）がクライアントの IP アドレスの認証をクライアントに要求することがあります。これは古い Web サーバで一般的に使用される方法ですが、クライアント認証にとって望ましい解決方法ではありません。このような古い Web サイトの場合、Content Engine がクライアントと Web サーバ間を「すり抜け」てクライアントを認証する可能性もあります。このような場合、認証トラフィックのバイパス機能が使用されます。このトピックに関する詳細については、「認証トラフィックのバイパス設定」（p.4-24）を参照してください。

表 15-1 に、プロトコル、要求認証機構、応答認証機構、認証モード、Content Engine のサーバとしての認証決定サービスの関連性を示します。パススルー認証モードでは、サポートされているそれぞれの認証要求機構でのみ照合認証サービスを搭載した認証サーバと通信できます。プロキシ認証の場合、HTTP 要求の基本認証を他のサポートされている決定サービスで使用できます。HTTP の NTLM 認証は、NTLM サービスに対してのみプロキシできます。Microsoft Media Server（MMS）または Real Time Streaming Protocol（RTSP）のストリーミング プロトコルは両方ともプロキシ認証をサポートしていません。ネイティブ FTP 要求は Lightweight Directory Access Protocol（LDAP）、Remote Authentication Dial-In User Service（RADIUS）、および Terminal Access Controller Access Control System Plus（TACAS+）の決定サービスをプロキシできます。

表 15-1　コンテンツ配信の認証 — サーバとしての Content Engine

プロトコル	パススルー認証モード		プロキシ認証モード			
	認証機構	決定サービス	NTLM 決定サービスに対して	LDAP 決定サービスに対して	RADIUS 決定サービスに対して	TACACS+ 決定サービスに対して
HTTP[1]	基本	基本に対して	可能	可能	可能	可能
	NTLM	NTLM に対して	可能	不可能	不可能	不可能
ネイティブ FTP	—	—	不可能	可能	可能	可能
MMS	基本	基本に対して	不可能	不可能	不可能	不可能
	NTLM	NTLM に対して	不可能	不可能	不可能	不可能
RTSP	基本	基本に対して	不可能	不可能	不可能	不可能

1.HTTP は、HTTP、FTP-over-HTTP、および HTTPS-over-HTTP 全般を指しています。

サポートされているコンテンツ配信プロトコル

ACNS ソフトウェアの認証では、4 つの異なるコンテンツ配信プロトコル（HTTP、ネイティブ FTP、MMS、RTSP）がサポートされます。これらのプロトコルは、次の要求および応答認証機構と互換性があります。

- HTTP および MMS は基本および NTLM 認証機構の両方と互換性があります（「NTLM 認証の概要」[p.15-4] を参照）。
- RTSP は、基本認証のみと互換性があります（「基本認証の概要」[p.15-4] を参照）。
- ネイティブ FTP は、基本 または NTLM 要求と応答認証機構の両方に対応していません。

サポートされるプロトコルについては、次のリリース情報を参照してください。

- ACNS 5.2 以降のソフトウェア リリースでは、HTTP 要求認証がサポートされます。この章では、*HTTP 要求*は HTTP、FTP-over-HTTP、および HTTPS-over-HTTP 要求全般を意味します（詳細については、「HTTP 要求の認証および許可の概要」[p.15-6] を参照してください）。
- ACNS 5.4 以降のソフトウェア リリースでは、非透過性のネイティブ FTP 要求のプロキシ認証がサポートされます（Reflection X クライアントや FTP プロキシとして動作する Content Engine の UNIX コマンド ライン プログラムのような FTP クライアントからの FTP-over-FTP 要求認証）。（詳細については、「ネイティブ FTP 要求の認証の概要」[p.15-13] を参照してください）。
- ACNS 5.4 以降のソフトウェア リリースでは、IP Access Control List（ACL; アクセス制御リスト）を使用して、FTP ネイティブ要求のアクセスを制御できます（たとえば、FTP プロキシとして動作している Content Engine を有効にして、Reflection X クライアントや UNIX コマンド ライン プログラムの FTP クライアントから着信する FTP 接続のアクセス要求を許可したり拒否したりできます）。詳細については、19-19 ページの「Using IP ACLs to Control FTP Access」の項を参照してください。

認証方式のサポート

プロキシ認証モードの Content Engine は、基本および NTLM 認証要求をサポートします。

基本認証の概要

基本認証は、簡単な要求と応答の認証機構です。基本認証が使用されると、証明書（ユーザ名およびパスワードなど）がクリア テキスト形式でオリジン サーバに送信されます。そのため、場合によっては、基本認証はセキュリティにリスクを伴います。

NTLM 認証の概要

NTLM は、Microsoft のブラウザ（Internet Explorer）、プロキシ、Web サーバ（IIS）で使用される要求および応答の認証機構です。NTLM を使用している場合、クライアントはサーバ要求をパスワード ハッシュで暗号化し、検証するサーバに応答を送信します。HTTP 要求認証に NTLM を使用する主な利点は、NTLM がサーバに暗号化形式でパスワードを送信することにあります。これは、基本認証のようにクリア テキスト形式でインターネットに送信するよりも非常に安全です。

Content Engine の NTLM サポートには、次の 3 つのサポート タイプがあります。（1）NTLM から NTLM のパススルー認証サポート、（2）HTTP 要求の NTLM プロキシ認証、（3）認証目的の NTLM グループ情報照会（表 15-1 を参照）。

Content Engine が NTLM 要求応答機構を使用する設定にした場合、Content Engine が要求をサービスする前に、事前に設定されている Primary Domain Controller（PDC; プライマリ ドメイン コントローラ）がユーザのドメイン、ユーザ名、およびパスワードを検証します。ACNS 5.2 以降のソフトウェア リリースでは、Content Engine に NTLM 認証を実行させるドメイン リストを指定できます。

Content Engine の NTLM バージョン 1 および 2 のサポート

ACNS 5.4 ソフトウェアは、HTTP 要求のパススルー認証およびプロキシ認証で、NTLM バージョン 1 およびバージョン 2 の両方をサポートします（それ以前の ACNS リリースでは NTLM バージョン 1 のみのサポートです）。

NTLM バージョン 2 は NTLM バージョン 1 の更新プログラムで、NTLM バージョン 1 よりも高いセキュリティ機能を有しています。NTLM バージョン 2 が要求を暗号化するためのアルゴリズムは、NTLM バージョン 1 よりもさらに複雑になっているだけでなく、NTLM バージョン 2 の暗号化応答には、応答時の攻撃を回避するためのタイムスタンプが追加されています。

NTLM バージョン 1 とは異なり、NTLM バージョン 2 はクライアントとサーバ（またはプロキシ）間でネゴシエーションを行いませんが、レジストリの値を変更することで個々の Windows ベースのクライアントとサーバで設定されています（Microsoft Windows クライアントのレジストリ、または Windows サーバのレジストリを変更する方法については、ご利用の Microsoft Windows のマニュアルを参照してください）。

クライアントに対する HTTP 応答で、サーバ（またはプロキシ）に認証が必要で、NTLM 証明書が承認されていることが示されている場合、クライアントは設定に基づいて適切な NTLM 応答を生成します。

- NTLM バージョン 1 がクライアントに設定されている場合、サーバ要求を受信した際、クライアントは NTLM バージョン 1 の応答を生成します。
- NTLM バージョン 2 がクライアントに設定されている場合、サーバ要求を受信した際、クライアントは NTLM バージョン 2 の応答を生成します。

ACNS ネットワークでは、クライアント、プロキシ、サーバ間で NTLM バージョン設定の同期がとれることが重要です。デフォルトでは、NTLM バージョン 2 は Content Engine で無効になっているため、バージョン 2 を有効にしないかぎり、Content Engine は自動的に NTLM バージョン 1 を使用します。

Content Engine で NTLM バージョン 2 を使用して指定の NTLM サーバと通信させるには、**ntlm server host** グローバル コンフィギュレーション コマンドの **v2** オプションを使用するか、Content Distribution Manager GUI で **v2** オプションを選択します。**v2** コマンド オプションを入力すると、Content Engine は特定の NTLM サーバとの要求認証に NTLM バージョン 2 を使用します。

基本認証中、Content Engine は NTLM 応答を生成して NTLM サーバと直接通信します。そのため、指定の NTLM ホスト サーバと通信する際、Content Engine に NTLM バージョン 2 を使用するように指定することは、基本認証にも非常に重要です。

次に、Content Engine を設定して 8 つの NTLM サーバから構成されるホストのリストを使用させる例を示します。

```
ContentEngine(config)# ntlm server host 172.16.10.10
ContentEngine(config)# ntlm server host 172.16.10.12
ContentEngine(config)# ntlm server host 172.16.10.14
ContentEngine(config)# ntlm server host 172.16.10.16
ContentEngine(config)# ntlm server host 172.16.10.18
ContentEngine(config)# ntlm server host 172.16.10.20
ContentEngine(config)# ntlm server host 172.16.10.22
ContentEngine(config)# ntlm server host 172.16.10.24
```

（注）　NTLM バージョン 2 をサポートしていないソフトウェア リリースでは、クライアントが NTLM バージョン 2 を使おうとしても、Content Engine がクライアント認証に失敗します。

（注）　ACNS 5.4 以降のソフトウェア リリースでは、NTLM バージョン 2 が Content Engine で有効になっているときに、ドメイン コントローラが NTLM バージョン 1 応答を認証した場合、クライアントが NTLM バージョン 1 を使用していれば Content Engine はクライアントを適切に認証します。

NTLM 対応ブラウザにおける NTLM バージョン 2 パススルー認証のサポート

Content Engine が ACNS 5.4 以降のソフトウェア リリースを使用しており、そのクライアント ブラウザが NTLM バージョン 2 に設定されている場合、パススルー認証とプロキシ認証の両方がサポートされます。

パススルー認証の場合、次の処理が発生します。

1. クライアントのブラウザは Content Engine から NTLM 要求を受信したあと、NTLM バージョン 2 応答を生成して Content Engine に送信します。
2. Content Engine はパススルー認証を実行し、NTLM バージョン 2 応答を Content Engine に認証要求を送信したドメイン コントローラに送信します。
3. Content Engine はドメイン コントローラの決定に基づいて、クライアントの要求コンテンツへのアクセスを許可（または拒否）します。

プロキシ認証の場合、ドメイン コントローラによってユーザが検証されたあと、Content Engine はクライアントのユーザ名およびドメイン名情報を認証キャッシュに保存します。Content Engine は、同じ IP アドレスからの以降の要求はそのまま認証します（「Content Engine 認証キャッシュの使用」[p.15-9] を参照）。

非 NTLM 対応ブラウザにおける NTLM バージョン 2 プロキシ認証のサポート

非 NTLM 対応のブラウザ（基本認証を使用するブラウザ）で Content Engine を設定し、NTLM バージョン 2 を使用したドメイン コントローラと通信できます。これにより、Content Engine がクリア テキストを使用したパスワードをネットワークに送信することを防ぐことができます。

Content Engine に NTLM バージョン 2 を使用させてドメイン コントローラと通信するには、次の条件をすべて満たす必要があります。

- NTLM バージョン 2 を使用するように Content Engine が設定されている。
- Content Engine は ACNS 5.4 以降のソフトウェア リリースを使用している。
- Content Engine 上で基本認証が無効になっている。

これらの条件を設定したうえで、クライアント ブラウザが Content Engine からクライアント認証が必要だという HTTP 応答を受信した場合、クライアント ブラウザは NTLM 認証の代わりに基本認証を実行します。クライアント ブラウザは、クリア テキストを使用してユーザの証明書をネットワーク上の Content Engine に送信します。Content Engine はドメイン コントローラからの要求を要請し、パスワードを使用して NTLM バージョン 2 応答を生成します。次にその NTLM バージョン 2 応答を検証のためにドメイン コントローラに送信します（「Content Engine の NTLM バージョン 1 および 2 のサポート」[p.15-4] を参照）。

（注）　クライアント ブラウザが Content Engine にパスワードをクリア テキストで送信しても、Content Engine はクリア テキスト形式のパスワードをネットワーク上のドメイン コントローラに送信しません。

Windows Media 要求の認証の概要

ACNS 5.2 以降のソフトウェア リリースでは、Windows Media クライアントからの MMS 要求のパススルー基本認証および NTLM 認証がサポートされています。ACNS 5.3 以降のソフトウェア リリースでは、Windows Media 9 プレーヤーからの Windows Media RTSP 要求のパススルー基本認証がサポートされています。このサポートにより、Content Engine はクライアントとオリジン サーバ間にトンネルを確立し、オリジン サーバにクライアントを認証させます。Content Engine はパススルーは実行しますが、プロキシ認証サーバとしての機能はありません。

Content Engine が直接プロキシ ルーティング モードで動作している場合、Content Engine の Windows Media 9 サーバは WMT 要求のパススルー認証に対して次の 2 つの認証機構をサポートします。

- 匿名の認証
- ネットワーク認証
  - プラグイン（NTLM または Kerberos）認証のネゴシエーション
  - プラグイン認証のダイジェスト

HTTP 要求の認証および許可の概要

企業はオンライン コンテンツへのアクセスを制限する方法として、HTTP 要求認証を使用できます。HTTP 認証を Content Engine に設定すると、Content Engine はリモート データベース（たとえば、RADIUS、TACACS+、LDAP、または NTLM データベース）でユーザ パスワードの認証をチェックして、ユーザが要求したコンテンツへのアクセスを受け入れるか、または拒否する必要があるのかを決定します。

たとえば、BigCorp 社が、社員にはインターネットのアクセスを制限し、パートには Web アクセスを禁止しようとした場合、Content Engine をインターネット ゲートウェイに設定すると、強制的に、社員にインターネット アクセス ポリシーを適用できます。Content Engine が、自身がサービスするコンテンツに対してクライアントのアクセス要求を受信した場合、次のような事項が発生します。

1. Content Engine は認証要求をクライアントに送信し、クライアントにユーザ名やパスワードなどの認証情報を入力するよう求める。
2. Content Engine は Authentication, Authorization, and Accounting（AAA; 認証、許可、アカウンティング）サーバと通信し、提供された認証情報が有効かどうかを判断する。
3. AAA サーバからの応答に基づいて、次の事項が発生します。
   a. AAA サーバがユーザを認証した場合、Content Engine はその要求を許可します（クライアントは要求したコンテンツにアクセスできます）。
   b. AAA サーバがユーザを認証しなかった場合、Content Engine はその要求を拒否し、クライアントに認証失敗のメッセージを送信します。

ACL を使用したグループベースの許可

より細かいレベルで制御するために、HTTP 認証に加えて、グループベースの許可を NTLM ユーザおよび LDAP ユーザに対して使用できます。ACNS 5.x ソフトウェアでは、ACL を作成することで、インターネット アクセスを許可するユーザ グループと許可しないグループを指定できます。

グループベースの許可について Content Engine を設定する場合、Content Engine は AAA サーバに、要求しているクライアントについてだけでなく、その所属するグループも問い合わせます。次に Content Engine は ACL を確認し、ユーザが所属するグループの要求したコンテンツへのアクセスを許可するかどうかを判断します。

ルールを使用したグループベースの許可

Content Engine は、グループベース許可に対するグループベースのルールもサポートしています。この機能を使用すると、HTTP 要求を作成しているエンド ユーザを外部の AAA サーバで認証させ、設定した Rules Template で許可させることができます。

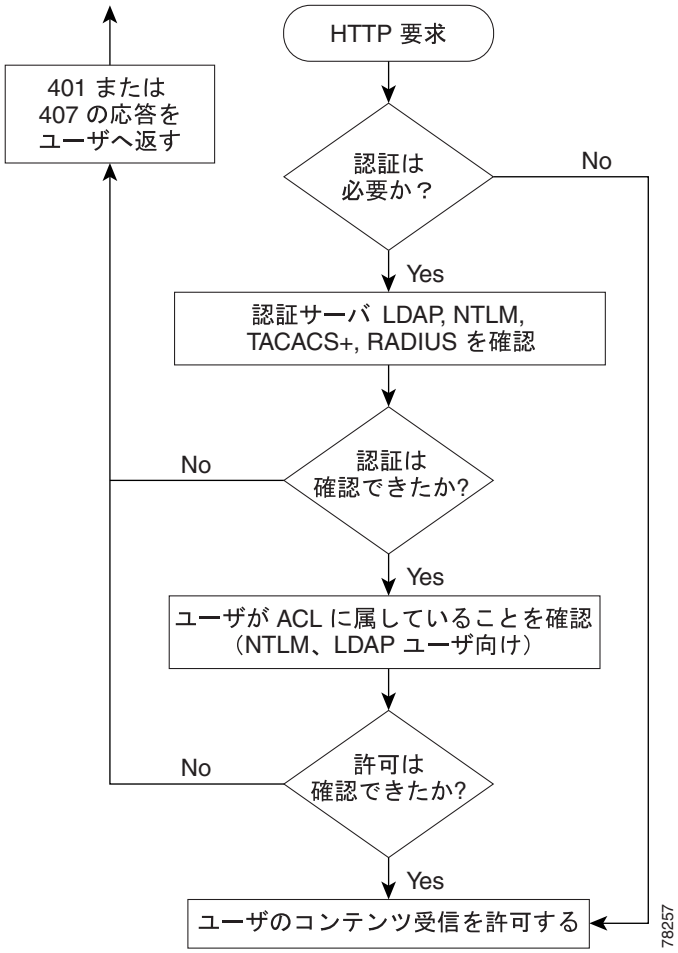
（注）　グループ ベースの許可は、HTTP 要求認証が発生したあとにのみ発生します。

Rules Template 機能を使用すると、ルールのセットを設定できます。各ルールは、HTTP、HTTPS、MMS、および RTSP 要求にフィルタをかけるために Content Engine が使用する動作やパターン、またはパターン グループで明確に識別できます。Rules Template 機能を Content Engine で有効にして Content Engine のルールを設定した場合、Content Engine はすべての着信クライアント要求を確認し、ルール パターンと一致する要求コンテンツがあるかどうかを判断します。それらの要求とルール パターンが一致した場合、Content Engine は指定の動作（またはポリシー）を実行して着信コンテンツをフィルタリングします。

ACNS 5.2 ソフトウェア リリースでは、3 つの新しいルール パターンが追加されました（**groupname**、**username**、**groupname-regex**）。これらの新ルール パターンは、認証 NTLM と LDAP ユーザのグループ名およびユーザ名に基づいたアクセス制御ポリシーをサポートします。グループ名に基づいたルールは、NTLM および LDAP で認証されたユーザに適用されます。ユーザ名に基づいたルールは、認証にユーザ名を含む要求認証方式である LDAP、NTLM、RADIUS、および TACACS+ で認証されたユーザに適用されます。

図 15-1 に、コンテンツへのアクセス制御機構としての HTTP 要求認証およびグループベースの許可の使用方法を示します。

図 15-1　HTTP 要求認証とグループベースの許可



グループ検索における Microsoft Active Directory の使用

グループベースの許可を使用した機能である Microsoft Active Directory は、Windows 2000 サーバで動作するソフトウェア アプリケーションです。Active Directory（AD）データベースは、Microsoft Active Directory プログラムを実行している Windows 2000 サーバにあるユーザ データベースです。Content Engine の LDAP クライアントは、Active Directory グループ検索の LDAP をサポートします。

（注）　Microsoft Active Directory は、LDAP バージョン 3 のみサポートします。Content Engine のデフォルトは LDAP バージョン 2 です。そのため、LDAP Active Directory 機能を Content Engine で有効にする前に、LDAP バージョン 3 を使用できるように Content Engine を設定する必要があります。

ACNS ソフトウェアのリリース 5.1 以降では、LDAP の再帰的検索を使用することで、ネストされたグループ名を抽出し、ネストされたグループに設定されたアクセス リストをすべて適用できます。Active Directory サーバでネストされたグループを使用する場合、親グループに設定されたポリシーが自動的にサブグループのメンバーにも適用されます。

次の項目は、Active Directory のグループ検索のトリガーとして使用されます。

- 設定されたグループ名ベースのアクセス リスト
- Rules Template で設定されたグループベースのルール
- 認証されたグループ ヘッダーを追加するために設定された Internet Content Adaptation Protocol（ICAP）
- Content Engine で有効になっている SmartFilter

Content Engine 認証キャッシュの使用

他のアクセス制御エリアは、認証コンテンツのキャッシングです。つまり、クライアントにオブジェクトを提供する前に Web サイトがクライアント認証を要求したため、そのオブジェクトが Content Engine にキャッシュされた場合、Content Engine は、オブジェクトを他のクライアントに配信する前にクライアントを認証します。

（注）　また、クライアント認証が必要なオブジェクトをキャッシュした場合（つまり、認証コンテンツをキャッシュした場合）、Content Engine は、クライアントがキャッシュ コンテンツへのアクセスを認証されないかぎり、キャッシュした認証コンテンツを他のクライアントに配信しないことを確実にする必要があります。

Content Engine は HTTP を照会して、ユーザからの証明書 1 セット（ユーザ ID およびパスワード）を入手し、これらを認証サーバ データベース内の証明書と照合します。Content Engine が認証サーバを介してユーザを認証すると、その認証の記録が、Content Engine の RAM（認証キャッシュ）内にローカルに保存されます。認証エントリが保存されているかぎり、制限されたインターネット コンテンツに同一ユーザがこれ以降アクセスを行っても、認証サーバによる検索は必要ありません。

使用されている Content Engine のルーティング方法は、認証コンテンツが認証キャッシュに入力されインデックスされる方法に影響します。HTTP 要求認証でサポートされているルーティング方法は、直接プロキシ ルーティングと透過的な WCCP 対応ルータのリダイレクションです。Content Engine は、次のように、使用されるルーティング方法によって、認証レコードを別々に保存（インデックス）しています。

- Content Engine が直接プロキシ ルーティング モードで動作している場合、クライアントのユーザ ID が認証キャッシュの鍵として使用されます。

  LDAP、RADIUS、および TACACS+ を直接プロキシ ルーティング モードで使用する場合、認証キャッシュ内に保持されている認証レコードには、入力されたユーザ名とパスワードがインデックスとして使用されます。ユーザ名は各クライアントの GET 要求で入力されます。
- Content Engine が透過的な WCCP 対応ルータのリダイレクション モードで動作している場合、クライアントの IP アドレスが認証キャッシュの鍵として使用されます。

  LDAP、RADIUS、および TACACS+ を 透過的な WCCP 対応ルータのリダイレクション モードで使用する場合、認証レコードに付けられるインデックスは、透過モードで要求を送信する Content Engine の IP アドレスです。クライアントの証明書は、要求時にのみ提出されます（毎回ではありません）。

（注）　NTLM サーバを直接プロキシ ルーティング モード、または透過的な WCCP 対応ルータのリダイレクション モードのいずれかで使用する場合も、すべての認証レコードのインデックスとして、要求側 Content Engine の IP アドレスが使用されます。

認証キャッシュの調整

認証キャッシュのサイズが、すべての認証済みユーザを同時に収容できるほど大きくない場合は、Content Engine は、まだ有効期限切れになっていない認証エントリの古いものから削除します。ユーザが最後にインターネット アクセスしたあと、認証キャッシュからユーザのエントリを削除するまでのデフォルト間隔は、480 分です。

CLI（コマンドライン インターフェイス）または Content Distribution Manager GUI から **http authentication cache timeout** グローバル コンフィギュレーション コマンドを使用して、認証キャッシュのタイムアウト値を調整できます。最小の間隔は 1 分、最大は 1440 分（24 時間）です。この間隔の有効期限が切れると、Content Engine は 認証サーバに対する再認証を要求します。

透過モードで HTTP 要求認証を使用している場合は、認証キャッシュのタイムアウト間隔を短くすることを推奨します。IP アドレスの再割り当てができるか、または、すでに認証されたデバイス（PC、ワーク ステーションなど）を介して、別のユーザがインターネットへアクセスできます。タイムアウト値を短くすると、個々のユーザが以前認証されたデバイスを使用してアクセスできる可能性を低くすることができます（ただし Content Engine がプロキシ モードで動作していると、非認証ユーザが簡単にインターネット アクセスできないため、エンド ユーザは有効なユーザ ID およびパスワードを提供する必要があります）。

ACNS 5.2 ソフトウェア リリースでは、**http authentication cache ttl** グローバル コンフィギュレーション コマンドにより、絶対タイムアウト設定オプションが導入されました。絶対タイムアウトを設定すると、過去に使用した認証ブラウザからアクセスする第三者の可能性を減らすことができます。詳細については、「再認証間隔の指定」（p.8-21）を参照してください。

プロキシ サーバ モードの HTTP 要求認証の概要

Content Engine がプロキシ サーバ モードで動作しており、HTTP 要求認証の設定がされている場合、次の 2 つの条件のうち 1 つが該当すれば、以下のイベントが発生します。（1）Content Engine がプロキシスタイルの要求をクライアントから直接受信する、または（2）Content Engine が WCCP リダイレクト要求を受信し、その **http authentication header** グローバル コンフィギュレーション コマンドのオプションがアップストリーム プロキシのため 407（Proxy Authorization Required）に設定されている。

1. Content Engine は、ユーザ情報を検出するためにクライアント要求の HTTP ヘッダーを検証します（Proxy-Authorization ヘッダー）。
2. ユーザ情報が提供されていない場合、Content Engine はクライアントに 407 メッセージ（Proxy Authorization Required）を返します。
3. クライアントはユーザ情報を含め、要求を再送信します。
4. Content Engine は認証キャッシュを検索し（ユーザ ID およびパスワードに基づく）、クライアントが以前認証されているかどうかを確認します。
5. 検索で一致するものが見つかった場合、その要求は通常どおりサービスされます。
6. 検索で一致しなかった場合、Content Engine は認証サーバに要求を送信してこのクライアントのエントリを探します。
7. 認証サーバが一致する情報を見つけた場合、Content Engine はその要求に対し通常どおりサービスし、クライアントのユーザ ID およびパスワードを認証キャッシュに保存します。
8. サーバ側で一致する情報が見つからなかった場合、Content Engine はクライアントに 407 メッセージを返します。

場合によっては、Content Engine（CE1）が支店でプロキシ モードに設定され、他のプロキシ モードの Content Engine（CE2）または他の HTTP 互換プロキシ デバイスが、ログイン認証に Content Engine またはプロキシ デバイスの両方を利用できる TACACS+、RADIUS、NTLM、または LDAP サーバと一緒にアップストリームに設定されていることもあります。

（注）　**http append proxy-auth-header** グローバル コンフィギュレーション コマンドは、ダウンストリームの Content Engine で設定される必要があります。そうすることで、アップストリームの Content Engine で必要なプロキシ認証情報が、ダウンストリームの Content Engine によって HTTP 要求から削除されずに済みます。各ダウンストリームの Content Engine では、最大 8 つのアップストリーム IP アドレスを設定できます。

支店のユーザ 1 がインターネットにアクセスし、CE1 にコンテンツがキャッシュされている場合、このコンテンツはユーザが認証されないかぎり、他の支店のオフィス ユーザにはサービスされません。CE1 はローカル ユーザを認証する必要があります。

CE1 および CE2 の両方がサーバに接続し、ユーザを認証する場合、支店のオフィス ユーザ 2 が、はじめにインターネット コンテンツを要求すると、CE1 はその要求に認証失敗（プロキシ モードの場合 HTTP 407 エラー、透過モードの場合 HTTP 401 エラー）の応答を返します。ユーザ 2 はユーザ ID およびパスワードを入力し、元の要求を証明書を追加して繰り返します。CE1 は HTTP 要求認証サーバに接続し、ユーザ 2 を認証します。

認証に成功して、キャッシュ ミスになった場合、証明書と一緒に要求は CE2 に送信されます。CE2 も、認証ユーザ 2 に関して認証サーバと通信します。認証が成功している場合、CE2 は自身のキャッシュからその要求のサービスを実行するか、オリジン サーバへ要求を転送します（この証明書の転送機能はデフォルト設定ではありません。証明書を転送する場合、**http append proxy-auth-header host** *CE2ipaddress* グローバル コンフィギュレーション コマンドでそれを設定する必要があります）。

これでユーザ 2 の認証情報が CE1 および CE2 の両方の認証キャッシュに保存されました。以降、ユーザ 2 のエントリ保有期間が切れ、認証キャッシュから削除されるまで CE1 と CE2 は両方ともユーザ 2 の要求に関して認証サーバに問い合わせる必要がなくなりました。

このシナリオでは、CE1 および CE2 はユーザの認証に同じ方法を使用していることを前提としています。特に、Content Engine には、両方とも同じ方法でエンコードされたユーザ証明書（ユーザ ID およびパスワード）が必要であると仮定します。

ヒント　認証がダウンストリームで実行されたあと、アップストリームでの Content Engine の認証を回避するには、**rule no-auth** グローバル コンフィギュレーション コマンドを使用して、ダウンストリームの Content Engine の IP アドレスを除外できます。

透過モードの HTTP 要求認証の概要

Content Engine が透過モード（WCCP 対応ルータのリダイレクションを使用）で動作している場合、ユーザの IP アドレスが認証キャッシュの鍵として使用されます。そのため、Content Engine は常に X-Forwarded-For ヘッダーとその送信元 IP アドレスを確認します。

一連のプロキシ（第 1 レベルの CE1 [クライアントに 1 番近い Content Engine] および第 2 レベルの CE2）の Content Engine に 2 つのレベルがある場合、CE1 および CE2 は両方とも **http append x-forwarded-for-header multiple-ip-address** グローバル コンフィギュレーション コマンドでそれらを設定する必要があります。設定後に、次の事項が発生します。

1. クライアントから要求を受信したあと、CE1 はデフォルトのクライアントの IP アドレスを X-Forwarded-For ヘッダーに追加し、その要求を CE2 に転送します。たとえば、クライアントの IP アドレスが 10.1.1.20 の場合、X-Forwarded-For ヘッダー は、「X-Forwarded-for: 10.1.1.20」のようになります。

2. CE2 が CE1 からの要求を受信したあと、CE2 は X-Forwarded-For ヘッダーに CE1 の IP アドレスを追加します。結果的に、X-Forwarded-For ヘッダーにクライアントの IP アドレス（ヘッダーに記入済み）と CE1 の IP アドレスがカンマ区切りで記述されることになります。たとえば、CE1 の IP アドレスが 10.40.1.40 の場合、X-Forwarded-For ヘッダー は、「X-Forwarded-for: 10.1.1.20, 10.40.1.40」のようになります。

ACNS 5.4.1 以降のソフトウェア リリースでは、X-Forwarded-For ヘッダーの複数の IP アドレスがサポートされています。**http append x-forwarded-for-header multiple-ip-address** グローバル コンフィギュレーション コマンドを入力して、X-Forwarded-For ヘッダーの複数の IP アドレス追加サポートを有効にしてください。このコマンドを指定し、さらに CE1 から CE2 に X-Forward-For ヘッダーで要求が到着した場合、CE1 の IP アドレスが、X-Forwarded-For ヘッダーの既存のクライアントの IP アドレのあとにカンマ区切りで追加されます。

（注）　CE1 が X-Forwarded-For ヘッダーを作成しない場合（たとえば、Cisco Content Engine ではなく、このヘッダーをサポートしていない場合など）、CE2 の認証は機能しません。

2 つの Content Engine を持つトポロジーでは、CE1 は透過モードで動作し、CE2 はプロキシ モードで動作します。すべてのユーザのブラウザはプロキシとしての CE2 を参照します。

ブラウザはプロキシに要求を送信するため、証明書が必要な場合、HTTP 407 メッセージ（Proxy Authorization Required）が CE1 から各ユーザに送信されます。この 407 メッセージを使用することで、送信元 IP アドレスに関連する認証問題を回避できます。ユーザ名およびパスワードがその代わりに使用されます。

このモードは、HTTP 401 メッセージを使用するよりも安全です。Content Engine はアドレス形式を確認し、アップストリーム プロキシがあるかどうかを判断します。アップストリームにプロキシが存在する場合、透過モードで動作している場合でも、Content Engine は HTTP 407 メッセージを使用してユーザに証明書を要求します。

Content Engine が透過モードで動作しており、HTTP 要求認証の設定がされている場合、次の 2 つの条件のうち 1 つが該当すれば、以下のイベントが発生します。（1）Content Engine がリダイレクトした要求をクライアントから受信する、または（2）アップストリームにプロキシが存在しないため、**http authentication header** グローバル コンフィギュレーション コマンドのオプションが 401（Unauthorized）に設定されている。

1. Content Engine は認証キャッシュを検索し、ユーザの IP アドレスが以前認証されているかどうかを確認します。
2. 検索で一致するものが見つかった場合、Content Engine はその要求を通常どおりサービスします。
3. 最初の段階で一致するものが見つからなかった場合、Content Engine は、ユーザ情報を検出するために HTTP ヘッダーを検証します（Authorization ヘッダーに含まれています）。
4. ユーザ情報が提供されていない場合、Content Engine はクライアントに 401 メッセージ（Unauthorized）を返します。
5. クライアントはユーザ情報を含め、要求を再送信します。
6. Content Engine は認証サーバに要求を送信してこのユーザのエントリを探します。
7. 認証サーバが一致する情報を見つけた場合、Content Engine はその要求に対し通常どおりサービスし、クライアントの IP アドレスを認証キャッシュに保存します。
8. サーバ側で一致する情報が見つからなかった場合、Content Engine はクライアントに 401（Unauthorized）メッセージを再度返します。

透過モードでは、Content Engine はクライアントの IP アドレスを Content Engine の認証キャッシュの鍵として使用します。

ACNS 5.4.1 以降のソフトウェア リリースでは、X-Forwarded-For ヘッダーの複数の IP アドレスがサポートされています。

（注）　Content Distribution Manager GUI を使用して、HTTP ヘッダーを作成する方法については、「認証済み HTTP キャッシュの設定」（p.8-16）を参照してください。

次の認証および許可サービスのうちいずれか 1 つを設定し、Content Engine の HTTP 要求認証を制御できます。

- LDAP サーバの設定（p.15-15）
- NTLM サーバの設定（p.15-21）
- RADIUS サーバの設定（p.15-30）
- TACACS+ サーバの設定（p.15-32）

（注）　Content Engine の NTLM サポートには、3 つのサポート タイプがあります（NTLM パススルー認証サポート、HTTP 要求の NTLM 認証、認証目的の NTLM グループ情報照会）。

ネイティブ FTP 要求の認証の概要

ACNS 5.4 以降のソフトウェア リリースでは、FTP クライアントと FTP プロキシ間の非透過接続のプロキシ認証がサポートされます。FTP プロキシは Content Engine の認証デーモンと通信し、RADIUS、TACACS、LDAP、NTLM プロトコルによるプロキシ認証サービスを提供します。

（注）　NTLM の場合、FTP プロキシは、Content Engine に設定されたデフォルトの NTLM ドメイン名を使用します。FTP クライアントは基本認証を使用して FTP サーバ（オリジン サーバ）で認証を実行します。NTLM 認証証明書がサーバに渡されることはありません。

次に、NTLM プロキシ認証を有効にした FTP クライアント セッションの例を示します。

```
bash-2.04$ ftp -d 10.19.228.108 8021
Connected to 10.19.228.108.
220 Welcome to FTP-proxy.  Login to the proxy using username and password.
Name (10.19.228.108:cisco_user): cnbu1\cisco_user
---> USER cnbu1\cisco_user
331 Password required for cnbu1\cisco_user.
Password:
---> PASS XXXX
220-Welcome to FTP-proxy.
220-Login to origin server using the 'USER username@server-hostname' command, or
220 Login to origin server using the 'SITE server-hostname' followed by the 'USER
username' command.
ftp>
ftp> user admin@22.9.192.10
---> USER admin@22.9.192.10
331 Password required for admin.
Password:
---> PASS XXXX
230 User admin logged in.  Access restrictions apply.
ftp> pwd
---> PWD
257 "/" is current directory.
ftp>
```

ネイティブ FTP プロキシ認証設定時の考慮事項

Content Engine の非透過性のネイティブ FTP 要求に対する要求認証を設定する場合、次の内容を考慮する必要があります。

- セキュリティ上の理由から、ネイティブ FTP プロキシ認証は Content Engine で無効になっています（デフォルト設定）。FTP プロトコルは基本的にセキュリティが高いものではありません。そのため、セキュア チャネル（HTTP プロキシ認証使用時など）で提供されたユーザ証明書でも、ネイティブ FTP プロキシ認証使用時に外部にさらされる可能性があります。
- ネイティブ FTP プロキシ認証を設定するとき、Content Engine にまだ認証サービス（RADIUS、LDAP、TACACS+、NTLM など）を設定していない場合、警告メッセージが表示されます。このメッセージが表示された場合、FTP プロキシ認証機能を有効にする前に、Content Engine に認証サービスを設定してください。
- 非透過性のネイティブ FTP 要求の要求認証の場合、ACNS は認証サービスとして TACACS+、RADIUS、および LDAP をサポートします。ネイティブ FTP 要求を送信している FTP クライアントは、サポートされている認証サービス（TACACS+、RADIUS、LDAP）が Content Engine で有効の場合にのみ Content Engine によってプロキシ認証の問い合わせを受けます。

（注）　NTLM が Content Engine で設定されている場合、Content Engine は FTP クライアント（ネイティブ FTP 要求を送信した）にプロキシ認証を問い合わせません。

ACNS 5.4 以降のソフトウェア リリースでは、HTTP 要求および非透過性の FTP ネイティブ要求の認証サービスを有効に設定する手順は同じです（たとえば、RADIUS を HTTP 要求および非透過性のネイティブ FTP 要求の認証サービスとして有効に設定する場合の手順も同じです）。

ただし、ネイティブ FTP キャッシングのサポートを適用する場合、次の制限事項が発生します。

- FTP 要求のプロキシ ルールはサポートされません
- URL フィルタリング方式（good リスト、bad リスト、N2H2、Websense、SmartFilter）はサポートされません

- ACNS 5.4 以降のソフトウェア リリースでは、IP ACL を使用して、ユーザ（FTP クライアント）に対して FTP プロキシとして稼働している Content Engine に送信する非透過性および透過性の FTP ネイティブ要求のアクセスを制御できます（詳細については、第 17 章「IP ACL の作成と管理」を参照してください）。
- ACNS 5.4 以降のソフトウェア リリースでは、着信プロキシ モード接続の応答として Content Engine が FTP クライアントに送信するメッセージをカスタマイズできます（詳細については、「ネイティブ FTP カスタム メッセージの設定」[p.8-70] を参照してください）。

中央管理 Content Engine の要求認証の設定

Content Engine 上に要求認証を設定するには、次の作業を実行する必要があります。

1. コンテンツ要求の認証時、対応する外部認証サーバのうち、どのデバイスを使うのかを決定します。
2. Content Engine 上で使用する認証サーバの設定を行います（次の「認証サーバの設定」を参照）。
3. 要求認証を処理するために、デバイスがどの認証データベースを確認する必要があるかを指定します。（「要求認証の認証方式の設定」[p.15-35] を参照）。

認証サーバの設定

Cisco ACNS ネットワーク ユーザの一部では、外部アクセス サーバを中央ロケーションのように使用し、ユーザに対するアカウントとアクティビティの認証、許可、およびアカウンティングを制御しています。外部認証サーバには、プロトコルとアプリケーション レベルで TACACS+、RADIUS、LDAP、および NTLM が実装されています。

要求に対する認証は、一度に 1 つのタイプのみ有効にできます。たとえば、LDAP 認証と NTLM 認証を同時に有効にすることはできません。

次に、Content Engine に対して LDAP、NTLM、RADIUS、および TACACS+ サーバを設定する方法を説明します。

LDAP サーバの設定

システム管理者は、Content Engine を使用してユーザのインターネット アクセスを制限できます。その際には、認証用に LDAP サーバを使用します。ACNS 5.x ソフトウェアは、LDAP バージョン 2 およびバージョン 3 をサポートし、Secure Authentication and Security Layer（SASL）以外のすべての LDAP 機能をサポートします。

Content Distribution Manager GUI を使用して LDAP サーバを設定する手順は、次のとおりです。

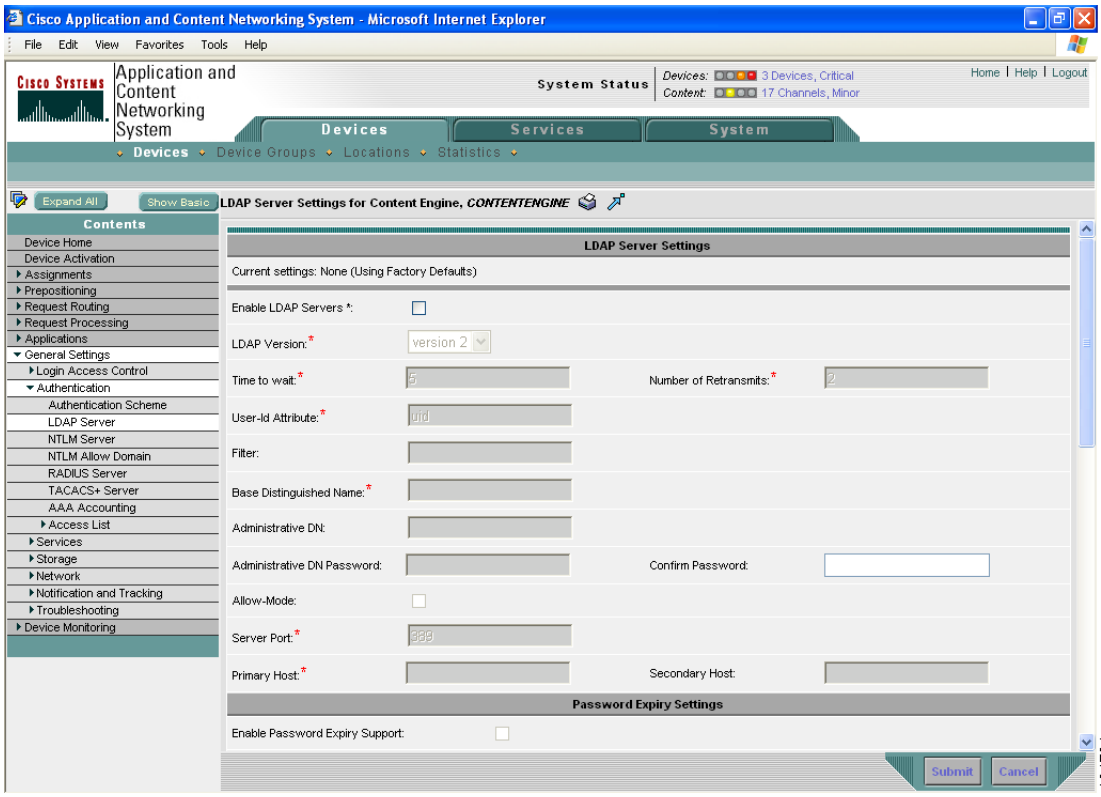
ステップ 1　Content Distribution Manager GUI から、**Devices > Devices** の順に選択します。

ステップ 2　設定する Content Engine の名前の横にある **Edit** アイコンをクリックします。Contents ペインが左側に表示されます。

ステップ 3　コンテンツのテーブル全体を表示するために、Contents ペインの上にある **Show All** ボタンをクリックします。

ステップ 4　Contents ペインから、**General Settings > Authentication > LDAP Server** の順に選択します。LDAP Server Settings ウィンドウが表示されます（図 15-2 を参照）。表 15-2 では、この図に表示されるフィールドを説明しています。

図 15-2　LDAP Server Settings ウィンドウ



ステップ 5　LDAP 要求認証を有効にするには、**Enable LDAP Servers** チェックボックスにチェックマークを付けます。

ステップ 6　LDAP Version ドロップダウン リストで、使用する LDAP プロトコル バージョンを選択します。

ステップ 7　Time to wait フィールドで、タイムアウトになるまで Content Engine が待機する秒数を指定します。

ステップ 8　Number of Retransmits フィールドで、Content Engine が LDAP サーバへの接続を試みている間にタイムアウト値を超えた場合、Content Engine が LDAP サーバへの接続を再度確定するために試行できる回数を設定します。送信試行の回数は 1 ～ 3 回で設定します。デフォルトでは 2 回です。

ステップ 9　User-id Attribute フィールドに、ユーザ ID 属性を入力します。

ステップ 10　Filter フィールドに、LDAP サーバが使用するフィルタ ストリングを入力します。

ステップ 11　Base Distinguished Name フィールドに、LDAP サーバ内の検索用のベース識別名ストリングを入力します。

ステップ 12　Administrative DN フィールドに、管理識別名を入力します。

ステップ 13　Administrative DN password フィールドに、管理識別名のパスワードを入力します。

ステップ 14　LDAP サーバが使用できないときにユーザへのアクセスを有効にするには、**Allow-Mode** チェックボックスにチェックマークを付けます。

ステップ 15 Windows Active Directory グループを使用できるようにするには、**Active Directory Groups** チェックボックスにチェックマークを付けます。

ステップ 16 Server Port フィールドに、LDAP サーバ認証用の TCP ポート番号を指定します。デフォルト ポート値 389 の使用を推奨します。

ステップ 17 Primary Host フィールドに、プライマリ LDAP サーバの IP アドレスを入力します。

ステップ 18 Secondary Host フィールドに、セカンダリ LDAP サーバの IP アドレスを入力します。

ステップ 19 この設定を保存するには、**Submit** をクリックします。

表 15-2　LDAP サーバの設定

GUI パラメータ	機能	CLI コマンド
Enable LDAP Servers	LDAP サーバを使用して HTTP 認証を有効にします。	**ldap server enable**
LDAP Version	使用する LDAP プロトコル バージョン（バージョン 2 またはバージョン 3）。	**ldap server version**
Time to wait	特定の LDAP サーバに接続するときタイムアウトが発生するまでの Content Engine が応答が送られてくるまで待つ秒数。デフォルト値は 5 秒です。	**ldap server timeout**
Number of Retransmits	LDAP サーバへの接続試行回数。デフォルト値は 2 回です。	**ldap server retransmit**
User-id Attribute	LDAP サーバ上のユーザ ID 属性の名称。デフォルトは「uid」です。	**ldap server userid-attribute**
Filter	LDAP フィルタ ストリング。デフォルト値はありません。	**ldap server filter**
Base Distinguished Name	LDAP サーバの検索の開始点となるベース識別名。これにより、ドメイン「com」など特定のストリングを対象とした検索が可能です。	**ldap server base**
Administrative DN	管理識別名。ベース識別名と関連する特定のユーザを検索することが可能です。	**ldap server administrative-dn**
Administrative DN Password	管理識別名のパスワード。	**ldap server administrative-passwd**
Allow-Mode	LDAP サーバが使用できないときにユーザへのアクセスを可能にします。	**ldap server allow-mode**
Active Directory Group	Windows Active Directory グループへのアクセスを可能にします。	**ldap server active-directory-group**
Server Port	LDAP サーバが受信するポート番号。デフォルト ポート番号は 389 です。	**ldap server port**
Primary Host	プライマリ LDAP サーバの IP アドレス。	**ldap server host**
Secondary Host	セカンダリ LDAP サーバの IP アドレス。	**ldap server host**

ACNS 5.4 ソフトウェアでは、LDAP サーバのパスワードの期限、ポリシーのリダイレクション、およびグループ設定を Content Distribution Manager GUI から設定できます。

これらを設定する手順は、次のとおりです。

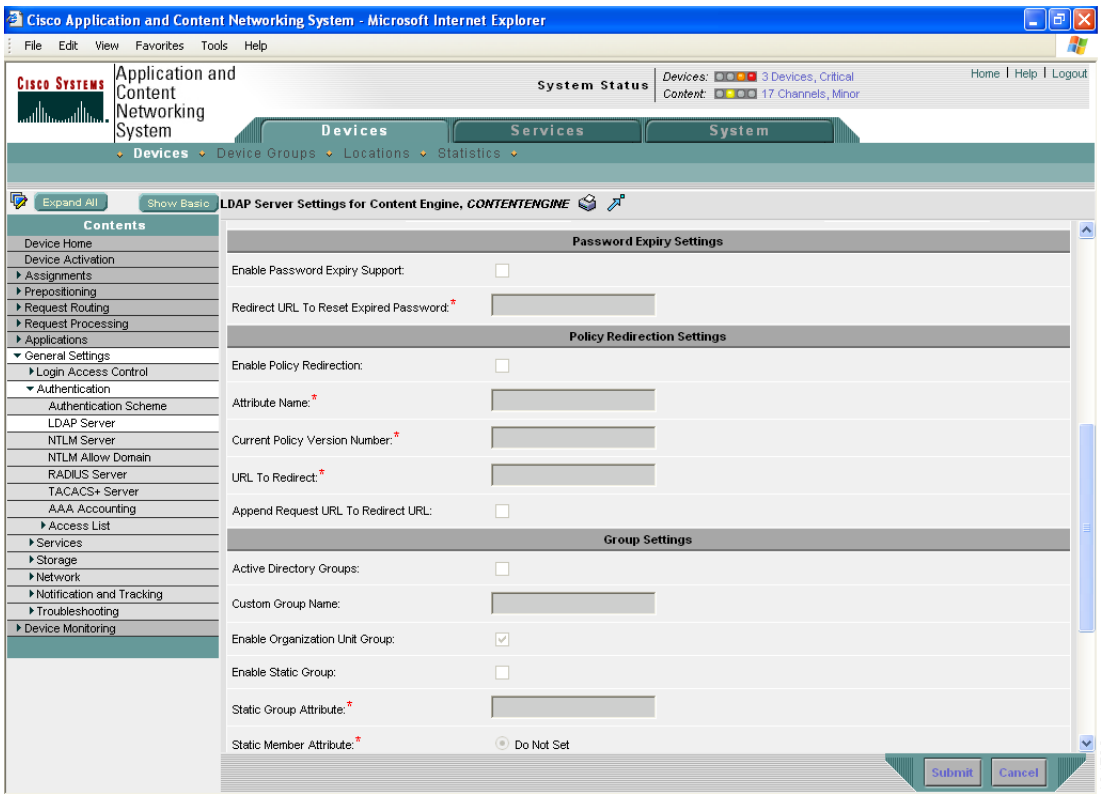
ステップ 1　Content Distribution Manager GUI から、**Devices > Devices** の順に選択します。

ステップ 2　設定する Content Engine の名前の横にある **Edit** アイコンをクリックします。Contents ペインが左側に表示されます。

ステップ 3　コンテンツのテーブル全体を表示するために、Contents ペインの上にある **Show All** ボタンをクリックします。

ステップ 4　Contents ペインから、**General Settings > Authentication > LDAP Server** の順に選択します。LDAP Server Settings ウィンドウが表示されます（図 15-3 を参照）。表 15-3 では、この図に表示されるフィールドを説明します。

図 15-3　LDAP サーバの追加設定



ステップ 5　LDAP Password Expiry Settings 見出しの下で、次の項目を設定します。

a. 認証パスワードの期限のサポートを有効にするには、**Enable Password Expiry Support** チェックボックスにチェックマークを付けます。

b. 期限切れのパスワードをリセットする URL へ直接移動するには、Reset Expired Password フィールドの Redirect URL に URL を入力します。

ステップ 6　Policy Redirection Settings 見出しの下で、次の項目を設定します。

a. ポリシー リダイレクションを有効にするには、**Enable Policy Redirection** チェックボックスにチェックマークを付けます。

b. 格納されているポリシーのバージョン番号で LDAP 属性を定義するには、Attribute Name フィールドに名前を入力します。

c. ポリシー バージョン番号を定義するには、Current Policy Version Number フィールドに 1 ～ 99999999 の値を入力します。

d. Usage Policy acceptance Web ページへユーザをリダイレクトさせるには、Redirect フィールドにその Web ページの URL を入力します。

e. Usage Policy の認証で、要求の URL に直接移動する場合、**Append Request URL to Redirect URL** チェック ボックスにチェック マークを付けます。

ステップ 7　Group Settings 見出しの下で、次の項目を設定します。

a. ユーザ アカウントの *memberOf* 属性からグループ名を取得し、その属性に応じたグループ メンバーシップを有効にするには、Active Directory Groups チェック ボックスにチェック マークを付けます。このチェック ボックスは、LDAP Version ドロップダウン リストから LDAP バージョンに 3 を選択した場合にのみチェックできます（図 15-2 を参照）。

b. ユーザ アカウントの *custom* 属性からグループ名を取得し、その属性に応じたグループ メンバーシップを有効にするには、Custom Group Name フィールドに名前を入力します。入力できる最大文字数は 256 文字です。

c. ユーザ アカウントの *organizationUnit* 属性からグループ名を取得し、その属性に応じたグループ メンバーシップを有効にするには、Enable Organization Unit Group チェック ボックスにチェック マークを付けます。

d. グループ メンバーシップのスタティック グループ クエリを有効にするには、**Enable Static Group** チェックボックスにチェックマークを付けます。

e. スタティック グループの属性名を設定するには、Static Group Attribute フィールドに名前を入力します。

f. オプション ボタンをクリックして、スタティック グループのメンバー属性を次の中から 1 つ選択します。

- **Do Not Set**
- **Member**
- **Unique Member**
- **Custom Member**

  Custom Member を選択した場合、テキスト フィールドにスタティック グループのカスタム メンバー名を入力します。

g. ユーザ メンバーシップのネストされたスタティック グループ クエリを有効にするには、**Enable Nested Static Group** チェックボックスにチェックマークを付けます。

h. ネストされたスタティック グループ クエリのレベルを設定する場合、Level of Nested Static Group フィールドに 1 ～ 100 の数値を入力します。

ステップ 8　この設定を保存するには、**Submit** をクリックします。

表 15-3　LDAP サーバの追加設定

GUI パラメータ	機能	CLI コマンド
LDAP パスワードの期限設定		
Enable Password Expiry Support	認証パスワードの期限設定のサポートを有効にします。	**ldap server password-expiry enable**
Redirect URL to Reset Expired Password	期限切れのパスワードをリセットできる URL に直接移動します。このパラメータには、次のような有効な形式であれば、すべての URL を使用できます。 http://www.someserver.com/file.html	**ldap server password-expiry redirect-url** *url*
ポリシー リダイレクションの設定		
Enable Policy Redirection	ポリシー リダイレクションのサポートを有効にします。	**ldap server policy-redirect enable**
Attribute Name	格納されているポリシー バージョン番号のもとで、LDAP 属性を定義します。	**ldap server policy-redirect attribute** *name*
Current Policy Version Number	ポリシー バージョン番号を定義します。設定できる範囲は 1 ～ 99999999 です。	**ldap server policy-redirect version-number** *number*
URL to Redirect	ユーザを Usage Policy acceptance Web ページにリダイレクトします。	**ldap server policy-redirect redirect-url** *url*
Append Request URL to Redirect URL	Usage Policy を承認することで要求の URL に直接移動します。	**ldap server policy-redirect append-request-url**
グループ設定		
Active Directory Groups	ユーザ アカウントの *memberOf* 属性を問い合わせし、その属性に対応したグループ メンバーシップを有効にします。 Active Directory の問い合わせを有効にするには、LDAP バージョンをバージョン 3 に設定する必要があります。	**ldap server group active-directory enable**
Custom Group Name	ユーザ アカウントの *custom* 属性を問い合わせし、その属性に対応したグループ メンバーシップを有効にします。カスタム グループ名の最大文字数は 256 文字です。	**ldap server group custom** *name* **enable**
Enable Organization Unit Group	ユーザ アカウントの *organizationUnit* 属性を問い合わせし、その属性に対応したグループ メンバーシップを有効にします。	**ldap server group organizationUnit enable**
Enable Static Group	グループ メンバーシップのスタティック グループの問い合わせを有効にします。	**ldap server group static enable**
Static Group Attribute	スタティック グループの属性名を設定します。スタティック グループ属性の最大文字数は 256 文字です。Active Directory グループが有効の場合は、設定できません。	**ldap server group static group-attribute** *name*

表 15-3　LDAP サーバの追加設定（続き）

GUI パラメータ	機能	CLI コマンド
グループ設定		
Static Member Attribute: Do Not Set Member Unique Member Custom Member	スタティック グループのメンバー属性を設定します。カスタム スタティック メンバー属性の最大文字数は 256 文字です。Active Directory グループが有効の場合、または LDAP バージョンが 3 以外の場合は、設定できません。	**ldap server group static member-attribute {custom-member** *name* **| member | uniquemember}**
Enable Nested Static Group	スタティック グループ メンバーシップのネストされた問い合わせを有効にします。	**ldap server group static nested enable**
Level of Nested Static Group	問い合わせに対するネストされたスタティック グループのレベルを設定します。レベルには 1 ～ 100 の数値が入ります。デフォルトは 1 です。	**ldap server group static nested level** *number*

NTLM サーバの設定

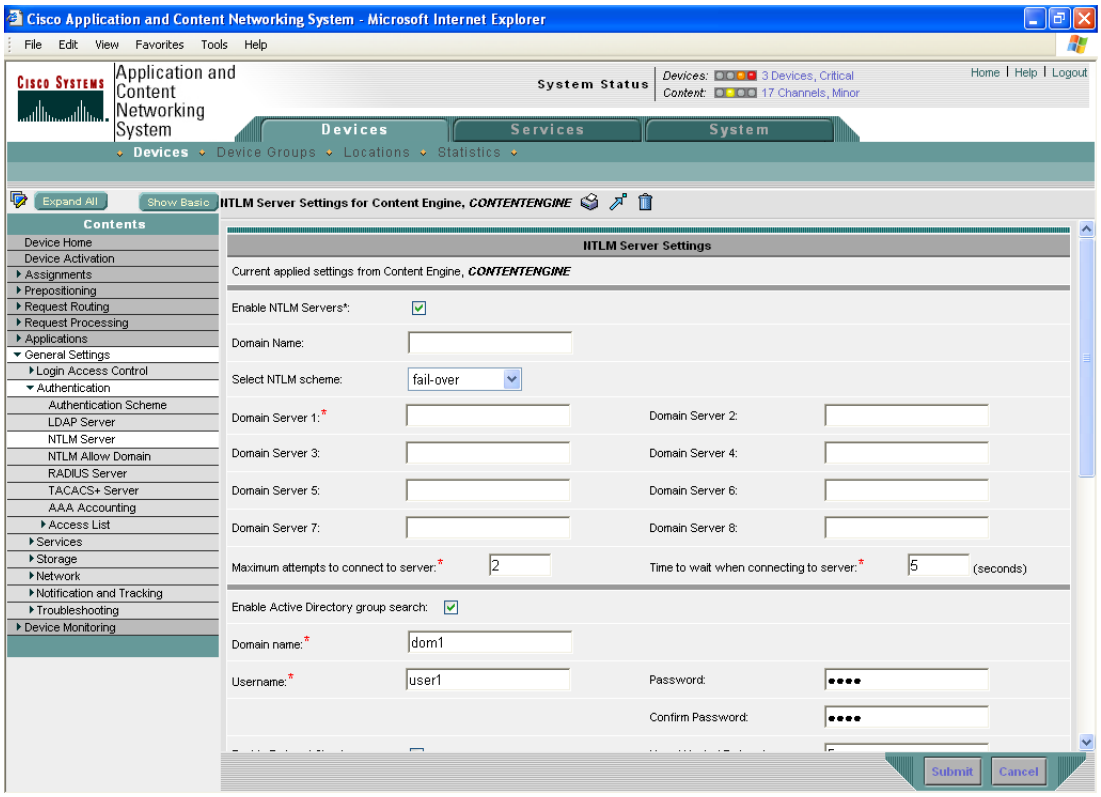
Content Distribution Manager GUI を使用して NTLM サーバを設定する手順は、次のとおりです。

ステップ 1　Content Distribution Manager GUI から、**Devices > Devices** の順に選択します。

ステップ 2　設定する Content Engine の名前の横にある **Edit** アイコンをクリックします。Contents ペインが左側に表示されます。

ステップ 3　Contents ペインから、**General Settings > Authentication > NTLM Server** の順に選択します。NTLM Server Settings ウィンドウが表示されます（図 15-4 を参照）。表 15-4 では、このウィンドウに表示されるフィールドを説明します。

図 15-4　NTLM Server Settings ウィンドウ — 上部



ステップ 4　NTLM 認証を有効にし、NTLM サーバ ドメイン名、NT プライマリ ドメイン コントローラ名または IP アドレスを設定し、必要に応じて、ホスト名またはアドレスをプライマリまたはセカンダリとして設定するには、**Enable NTLM Servers** チェックボックスにチェックマークを付けます。

NTLM 認証要求をする前に、必ず、次の条件が満たされていることを確認してください。

- NTLM プライマリ ドメイン コントローラに、NetBIOS で指定されたコンピュータ アカウントと一致する Domain Name System（DNS）へのエントリがある。
- プライマリ ドメイン コントローラは、正引きおよび逆引きの両方の DNS 解決ができる。
- Content Engine 上で設定されたドメイン名は、プライマリ ドメイン コントローラをその一部としたドメインと一致する。このドメインは、PDC をホストとするドメインか、または PDC が他の PDC との通信によって認証できるトラステッド（信頼できる）ドメインかのいずれかです。

（注）　このドメインは、PDC をホストとするドメインか、または PDC が他の PDC との通信によって認証できるトラステッド（信頼できる）ドメインかのいずれかです。

ステップ 5　Domain Name フィールドで、認証を受ける必要があるユーザが存在するドメイン名を指定します（「No Default NTLM Domain のサポート」[p.15-28] を参照）。

ステップ 6　Select NTLM scheme ドロップダウン リストから、HTTP 要求認証用の NTLM サーバのためのスキームを指定するオプションを選択します。このオプションによって、設定した NTLM サーバで使用される方式（負荷分散およびフェールオーバー）を指定できます。デフォルトの方式は **fail-over** で、サポートされるもう 1 つの方式は **load-balanced** です。

負荷分散スキームが使用可能になると、最初の要求のみが最初の設定済みサーバへ送信され、その後複数の設定済みサーバのなかでラウンド ロビンが使われます（「HTTP 要求認証用 NTLM 負荷分散について」[p.15-29] を参照）。フェールオーバー方式が使用可能になると、Content Engine はすべての要求を最初の設定済みサーバへ送信します（「HTTP 要求認証用 NTLM フェールオーバーについて」[p.15-29] を参照）。

ステップ 7　Domain Server1 フィールドに、ドメイン サーバのホスト名または IP アドレスを入力します。複数のサーバがフェールオーバーするように設定されている場合、Server1 と設定されたサーバがプライマリ NTLM ドメイン サーバになります。

（注）　フェールオーバー方式が有効になっている場合にのみ、設定済みの最初のサーバは「プライマリ」サーバで、すべての要求が最初に送られます。負荷分散方式が有効になっている場合、「プライマリ」サーバの定義は適用できません。

ステップ 8　Domain Server 2 ～ 8 フィールドに、Content Engine が HTTP 要求認証に使う各 NTLM サーバのホスト名、または IP アドレスを入力します。設定済みの NTLM サーバのリストは、「ホスト リスト」として参照されます。

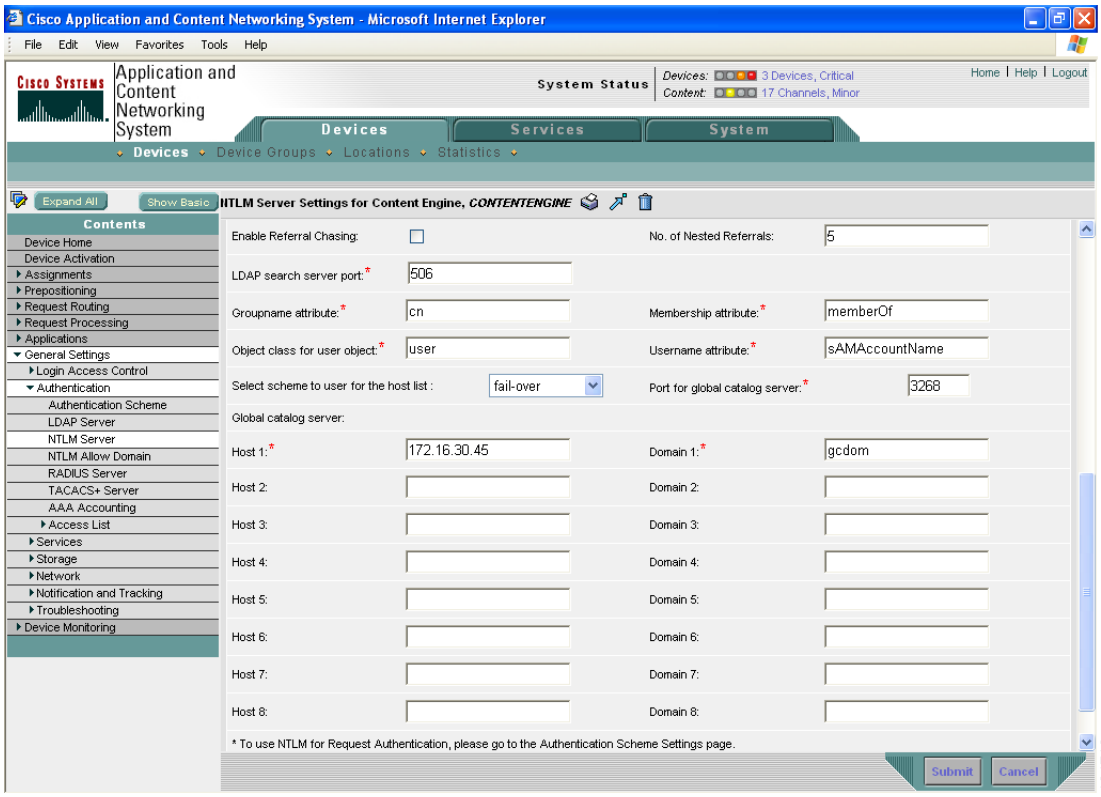
ACNS ソフトウェア内で、Content Engine が NTLM サーバを 8 台まで HTTP 要求認証に使うよう設定できます。サーバ設定の順番によって、負荷分散あるいはフェールオーバーの順番が決まります。

ステップ 9　接続試行回数を設定するには、Maximum attempts to connect to server フィールドに数を入力します。

ステップ 10　サーバ接続が失敗するまでの待機時間を設定するには、Time to wait when connecting to server フィールドに時間（秒数）を入力します。

ステップ 11　Active Directory group サーチのサポートを有効にするには、**Enable Active Directory group search** チェックボックスにチェックマークを付けます（図 15-5 を参照）。

図 15-5　NTLM Server Settings ウィンドウ — 下部



Active Directory group サーチが有効になっていると、ACNS ソフトウェアは NTLM グループ認証を使用する Microsoft Active Directory データベースと相互運用できます。

ステップ 12　Domain Name フィールドに、列挙するユーザのドメインを入力します。

（注）　列挙ユーザは、Content Engine 上に定義されたアカウントです。このアカウントを使用すると、Content Engine は Microsoft Active Directory データベース内をサーチできます。この列挙ユーザには、ディレクトリ全体に読み取り権限が必要です。

ステップ 13　Username フィールドに、列挙ユーザ名を入力します。

ステップ 14　Password フィールドに、列挙ユーザのパスワードを入力します。

ステップ 15　Confirm Password フィールドに、列挙ユーザのパスワードを再度入力します。

ステップ 16　Microsoft Active Directory データベース内をサーチできるようにするには、**Enable Referral Chasing** チェックボックスにチェックマークを付けます。

ステップ 17　No. of Nested Referrals フィールドに、サーチを行うディレクトリ レベルを入力します。

ステップ 18　LDAP search server port フィールドに、LDAP サーバのポート番号を入力します。デフォルトは 389 です。

ステップ 19 Groupname attribute フィールドに、Microsoft Active Directory データベース内のグループ名属性を入力します。デフォルトは cn です。

ステップ 20 Membership attribute フィールドに、Microsoft Active Directory データベース内の構成員属性を入力します。デフォルトは MemberOf です。

ステップ 21 Object class for user object フィールドに、Microsoft Active Directory データベース内のユーザ オブジェクトのオブジェクト クラスを入力します。デフォルトは user です。

ステップ 22 Username attribute フィールドに、Microsoft Active Directory データベース内のユーザ名属性を入力します。デフォルトは sAMAccountName です。

ステップ 23 **Select scheme to use for the host list** ドロップダウン リストから、方式を選択します。

- ホスト間でフェールオーバーを行うようにする場合、**failover** を選択する。
- ホスト間でラウンドロビンで負荷分散を行うようにする場合、**load-balanced** を選択する。

ステップ 24 Port for global catalog server フィールドに、グローバル カタログ サーバのポート番号を入力します。デフォルト ポート番号は 3268 です。

ステップ 25 Host フィールドに、グローバル カタログ サーバのホスト名または IP アドレスを入力します。グローバル カタログ サーバのホスト名を 8 つまで追加できます。

ステップ 26 ホスト名に対応する Domain フィールドに、グローバル カタログ サーバのドメイン名を入力します。グローバル カタログ サーバのドメイン名を 8 つまで入力できます。

ステップ 27 この設定値を保存するには、**Submit** をクリックします。

表 15-4　NTLM サーバの設定値

GUI パラメータ	機能	CLI コマンド
Domain Name	認証を受けるユーザが存在するドメイン名。	**ntlm server domain** *name*
Select NTLM Scheme	HTTP 要求認証用 NTLM サーバのための方式。	**ntlm server scheme** {**fail-over** | **load-balanced**}
Domain Server1	プライマリ ホストとして機能する NTLM サーバ。	**ntlm server host** {*hostname* | *ipaddress*} **primary**
Domain Server2–8	バックアップホストとしての役目をする NTLM サーバ。	**ntlm server host** {*hostname* | *ipaddress*} **secondary**
Maximum attempts to connect to server	サーバに接続するための最大試行回数（1 ～ 3）。デフォルトは 2 回です。	**ntlm server connection-retry** *number*
Time to wait when connecting to server	サーバに接続するために待機する秒数（1 ～ 20）。デフォルト値は 5 秒です。	**ntlm server connection-timeout** *seconds*
Enable Active Directory group search	ACNS ソフトウェアは NTLM グループ認証を使用する Microsoft Active Directory データベースと相互運用できるようにします。	**ntlm server ad-group-search enable**
Domain Name	列挙ユーザのドメイン。	**ntlm server ad-group-search enum-user domain** *domainname*

表 15-4　NTLM サーバの設定値（続き）

GUI パラメータ	機能	CLI コマンド
Username	列挙ユーザのユーザ名。	**ntlm server ad-group-search enum-user username** *username*
Password/Confirm Password	列挙ユーザのパスワード。	**ntlm server ad-group-search enum-user password** *password*
Enable Referral Chasing	LDAP 照会を有効にします。デフォルトは、無効になっています	**ntlm server ad-group-search ldap-referral enable**
No. of Nested Referrals	サーチを行うディレクトリ レベル。	**ntlm server ad-group-search ldap-referral limit** *number*
LDAP search server port	LDAP サーチ サーバのポート番号。	**ntlm server ad-group-search ldap-search-port** *portnum*
Groupname attribute	Active Directory データベース内のグループ名属性。デフォルト属性は *cn* です。	**ntlm server ad-group-search groupname-attribute** *attribute*
Membership attribute	Active Directory データベース内のメンバーシップ属性。デフォルトは *memberOf* です。	**ntlm server ad-group-search membership-attribute** *attribute*
Object class for user object	ユーザ オブジェクトのオブジェクト クラス。デフォルトは *user* です。	**ntlm server ad-group-search user-objectclass** *class*
Username attribute	Active Directory データベース内のユーザ名属性。デフォルト属性は *sAMAccountName* です。	**ntlm server ad-group-search username-attribute** *attribute*
Select scheme to use for the host list	グローバル カタログ ホスト リストに使用する方式。オプションは、fail-over または load-balanced です。	**ntlm server ad-group-search gc-server scheme** {**fail-over** | **load-balanced**}
Port for global catalog server	グローバル カタログ サーバのポート。デフォルトのポートは 3268 です。	**ntlm server ad-group-search gc-server port** *portnum*
Host	グローバル カタログ サーバの IP アドレスまたはホスト名。	**ntlm server ad-group-search gc-server host** {*hostname* | *ipaddress*}
Domain	グローバル カタログ サーバが設定されている場合のホストのドメイン名。	**ntlm server ad-group-search gc-server host** {*hostname* | *ipaddress*} **domain** *domain*

NTLM サーバの LDAP メモリ キャッシュの設定

ACNS 5.4 ソフトウェア リリースでは、ネストされたグループ検索に対して LDAP メモリのキャッシュをサポートします。この機能を使用すると、Content Engine はネストされたグループ検索の結果を LDAP キャッシュにローカルで保存できます。この機能は、デフォルトで有効になっています。

以前の ACNS のリリースでは、Content Engine が NTLM グループ検索を実行する場合、LDAP Global Catalog サーバと通信してすべての要求に対してグループ情報を取得する必要がありました。ユーザが複数のレベルでネストされたグループに所属している場合、Content Engine から Active Directory サーバに送信される問い合わせでユーザのすべてのグループ情報を取得する前に、グループ内の各親レベルを検索して、LDAP Global Catalog サーバに数回通信しなければなりませんでした。

ネストされたグループ検索のパフォーマンスを向上させるために、ACNS 5.4 ソフトウェアでは Content Engine に LDAP メモリ キャッシュを使用して、LDAP の問い合わせの結果をキャッシュできるようになりました。同じ検索要求が発行された場合、その結果は Content Engine の LDAP メモリ キャッシュから取得されるため、Content Engine が LDAP Global Catalog サーバへ接続する回数を減らすことができます。

NTLM サーバの LDAP メモリ キャッシュを設定する手順は、次のとおりです。

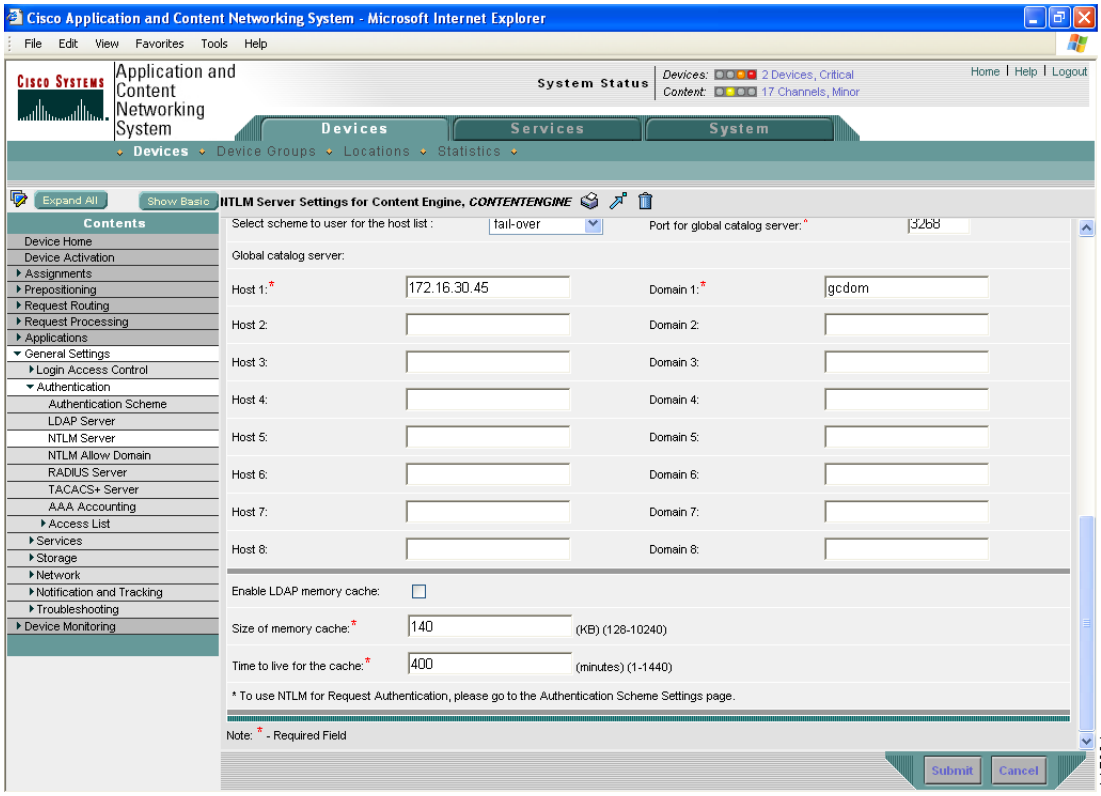
ステップ 1　Content Distribution Manager GUI から、**Devices > Devices** の順に選択します。

ステップ 2　設定する Content Engine の名前の横にある **Edit** アイコンをクリックします。

ステップ 3　Contents ペインから、**General Settings > Authentication > NTLM Server** の順に選択します。NTLM Server Settings for Content Engine ウィンドウが表示されます。

ステップ 4　LDAP メモリ キャッシュの設定場所へ移動するには、ウィンドウのスクロール ボタンを使用します（図 15-6 を参照）。

図 15-6　Content Engine に対する NTLM サーバの LDAP メモリ キャッシュの設定



（表 15-5 に、このウィンドウ内のフィールドについての説明と、対応する CLI コマンドを示します。）

ステップ 5　ネストされたグループの問い合わせ結果をキャッシュするには、**Enable LDAP memory cache** チェックボックスにチェックマークを付けます。

ステップ 6　LDAP メモリ キャッシュのサイズを設定するには、Size of memory cache フィールドに値（キロバイト）を入力します。

ステップ 7　メモリ内のエントリのキャッシュ期間を設定するには、Time to live for the cache フィールドに値（分）を入力します。

ステップ 8　この設定を保存するには、**Submit** をクリックします。

（注）　NTLM Server Settings for Content Engine ウィンドウで必要な他のフィールドもすべて設定してください。設定されていない場合、Submit のクリック時にエラー メッセージが表示されます（「NTLM サーバの設定」[p.15-21] を参照）。

表 15-5　Content Engine に対する NTLM サーバの LDAP メモリ キャッシュの設定

GUI パラメータ	機能	CLI コマンド
Enable LDAP memory cache	ContentEngine 上で LDAP のメモリ キャッシュを有効にします。デフォルトで有効になっています。	**[no] ntlm server ad-group-search mem-cache enable**
Size of memory cache	LDAP の問い合わせ結果をキャッシュするキャッシュ メモリに割り当てるサイズ（キロバイト）。範囲は、128 ～ 10240 KB です。デフォルトは 140 KB です。	**ntlm server ad-group-search mem-cache size** *size*
Time to live for the cache	LDAP メモリのキャッシュ内のエントリの有効期間（分）。範囲は、1 ～ 1440 分です。デフォルトは 400 分です。	**ntlm server ad-group-search mem-cache max-ttl** *max-ttl*

No Default NTLM Domain のサポート

ユーザが要求認証証明書のドメイン名を出さず、Content Engine 上に設定済みのデフォルト ドメインがない場合、ACNS ソフトウェアは「no domain configuration」エラー メッセージをクライアントに送信します。このエラー メッセージにはエラーの理由を示すテキストが含まれています。

（注）　no domain configuration 機能は NTLM をサポートしないブラウザ（たとえば、Netscape ブラウザなど）でのみサポートされます。Netscape ブラウザの場合、Content Engine の NTLM デフォルト ドメインが設定されていない場合、ユーザはドメインを指定する必要があります。指定しないと、クライアントはエラーを受信します。Netscape ブラウザの場合、ドメインは「domain\username」の形式でユーザ名の部分としてのみ供給できます。Internet Explorer などの NTLM をサポートしないブラウザでは、証明書を指定したユーザやデスクトップへのログオンに使用したドメインからの認証証明書に、常にドメイン名が含まれています。

HTTP 要求認証用 NTLM 負荷分散について

Content Engine を介してすべてのネットワークのトラフィックが行き来する大規模なネットワークでは、たとえ Content Engine 認証キャッシュがドメイン コントローラの負荷を削減する役割を担っているとしても、やはりエンド ユーザからのすべての認証の照会を単体のドメイン コントローラで処理するのは非実用的です。ACNS 5.4 ソフトウェアでは、負荷分散とフェールオーバーを目的として、サーバ（ドメイン コントローラ）を 8 台まで設定可能としました。

認証方式に **load-balanced** を選択すると、要求はドメイン コントローラ間のラウンドロビン処理に従います。ドメイン コントローラ（サーバ）が設定された順番で、負荷分散の順番が決まります。たとえば、サーバが *n* 台あるとすると、最初の要求はサーバ 0 へ送信され、2 番目の要求はサーバ 1 へ、そして n 番目の要求はサーバ n － 1 へ、n ＋ 1 番目の要求はサーバ 0 へと送信されます。サーバ 0 が失敗した場合には、Content Engine は要求を次の活動しているサーバ（この場合、サーバ 1）へ送信を試みます。ただし、次の活動しているサーバへのフェールオーバーは、1 回しか行われません。サーバ 1 が要求 1 の処理中にダウンした場合、要求 1 は再度フェールオーバーしません。

負荷分散が使用可能で、サーバ情報が実行時に変更された場合、その変更はサービスを中断することなく、実行時に引き取られます。

HTTP 要求認証用 NTLM フェールオーバーについて

ACNS 5.4 ソフトウェアは、ドメイン コントローラ間のフェールオーバーをサポートします。ドメイン コントローラ（サーバ）が設定された順序に基づき、フェールオーバーの順序が決まります。最初に設定されたサーバ（サーバ 0）が、最初に交信されます。最後に設定されたサーバ（サーバ 7）が、最後に交信されます。

1 回の接続動作のためのタイムアウト期間を超えると、Content Engine は接続を停止し、同じサーバと再度接続する動作を行います。ホスト リスト上で次に設定されたサーバへの接続を試行する前に、Content Engine は設定された再試行回数に達するまで（**ntlm server connection-retry** グローバル コンフィギュレーション コマンド）接続を再試行します。

HTTP 要求認証の NTLM 許可ドメインの設定

NTLM プロトコルは､インターネットへのユーザ アクセスを認証するためおよびブロックするために使用できます。ユーザが Windows NT または Windows 2000 のドメインにログインし、ブラウザを始動すると、認証情報はブラウザにより保存され、インターネットにアクセスするための NTLM 証明書として後で使用されます。ブラウザは、ドメイン名と一緒に NTLM 証明書を Content Engine に送信します。このキャッシュは、順に要求を Windows NT ドメイン コントローラに送信して、そのドメインにおけるユーザの妥当性をチェックします。ユーザがドメイン内の有効なユーザではない場合、インターネットへのアクセス要求は拒否されます。認証が成功すると、送信元 IP アドレスは Content Engine の認証キャッシュに入力できます。この IP アドレス以後の要求は、認証キャッシュ エントリが有効期限切れになるか、クリアされるまで、身元証明を要求されることはありません。

Content Engine で HTTP 要求認証の NTLM 許可ドメインを設定する手順は、次のとおりです。

ステップ 1　**Devices > Device Groups** の順に選択します。Device Groups ウィンドウが表示されます。

ステップ 2　NTLM HTTP 要求認証を使用するドメインを設定するデバイス グループの横にある **Edit** アイコンをクリックします。Modifying Device Group ウィンドウが表示されます。

ステップ 3　Contents ペインから、**General Settings > Authentication > NTLM Allow Domain** の順に選択します。NTLM Allow Domain Settings ウィンドウが表示されます。

ステップ 4　Content Engine 上で NTLM HTTP 要求認証を有効にするには、**Enable Use of the List of Domains Allowed to Authenticate** チェックボックスにチェックマークを付けます。

ステップ 5　Domain フィールドに、NTLM HTTP 要求認証をサポートするドメインの名前を入力します。

ステップ 6　NTLM HTTP 要求認証が許可されているドメインのリストにドメイン名を追加するには、**Add** をクリックします。このリストには、32 ドメインまで追加できます。新規に追加されたドメイン名がウィンドウに表示されます。

ステップ 7　ドメイン名を編集するには、NTLM HTTP 認証で許可されているドメインのリストとして表示されているドメイン名の横にある **Edit** アイコンをクリックします。

ステップ 8　ドメイン名フィールドを編集してから、**Edit** をクリックして、変更内容を保存します。

ステップ 9　NTLM HTTP 認証をサポートしているドメイン名をドメイン リストから削除するには、ドメイン名の横にある **Delete** アイコンをクリックします。

ステップ 10　この変更内容を保存するには、**Submit** をクリックします。

CLI から NTLM HTTP 要求認証を有効にするには、**ntlm allow-domain** {**enable** | **domain** *domainname*} グローバル コンフィギュレーション コマンドを使用します。

RADIUS サーバの設定

RADIUS 認証クライアントは、ACNS 5.x ソフトウェアを実行しているデバイス上にあります。RADIUS 認証クライアントが有効になっていると、ユーザ認証情報とネットワーク サービス アクセス情報が保存されている中央の RADIUS サーバに認証要求を送信します。

ヒント　Content Distribution Manager は、ユーザ認証情報をキャッシュしません。したがって、ユーザは RADIUS サーバに毎回要求を送って、再度認証を受けます。多数の認証要求によるパフォーマンスの低下を防止するために、Content Distribution Manager は、RADIUS サーバと同じロケーションにインストールしておくか、できるだけ RADIUS サーバの近くにインストールして、認証要求ができるだけ迅速に行われるようにしてください。

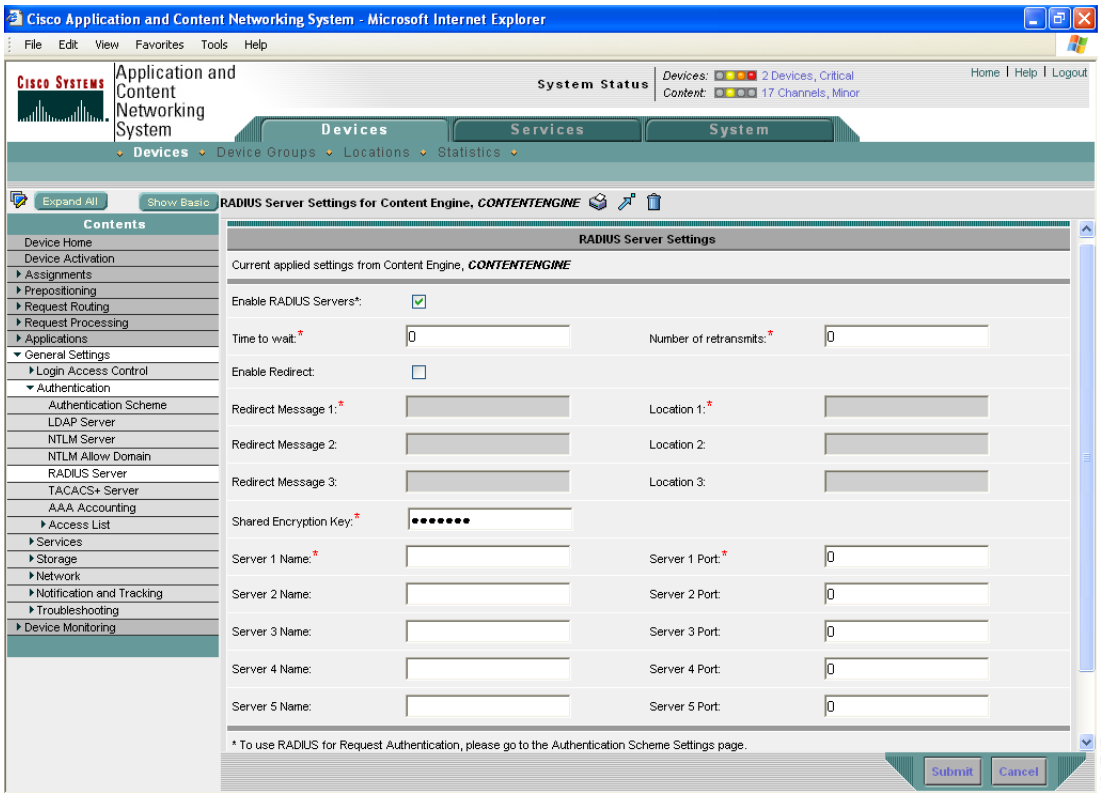
Content Distribution Manager GUI を使用して RADIUS サーバを設定する手順は、次のとおりです。

ステップ 1　Content Distribution Manager GUI から、**Devices > Devices** の順に選択します。

ステップ 2　設定する Content Engine の名前の横にある **Edit** アイコンをクリックします。Contents ペインが左側に表示されます。

ステップ 3　Contents ペインから、**General Settings > Authentication > RADIUS Server** の順に選択します。RADIUS Server Settings ウィンドウが表示されます（図 15-7 を参照）。表 15-6 では、このウィンドウ内のフィールドについて説明します。

図 15-7　RADIUS Server Settings ウィンドウ



ステップ 4　RADIUS 認証を有効にするには、**Enable RADIUS Servers** チェックボックスにチェックマークを付けます。

ステップ 5　Time to wait フィールドに、タイムアウトになるまで Content Engine が待機する時間を指定します。デフォルト値は 5 秒です。

ステップ 6　Number of Retransmits フィールドに、RADIUS サーバへの接続に成功するまでの試行を許可する回数を指定します。

ステップ 7　RADIUS リダイレクションを有効にするには、**Enable Redirect** チェックボックスにチェックマークを付けます。

ステップ 8　Redirect Message フィールドに、ユーザへのリダイレクト メッセージを入力します。3 つのリダイレクト メッセージを指定できます。

ステップ 9　Location フィールドに、リダイレクト メッセージの送信先を入力します。3 つの異なる送信先を指定できます。

ステップ 10　Shared Encryption Key フィールドに、RADIUS サーバとの通信に使用する秘密鍵を入力します。

ステップ 11　Server Name フィールドに、IP アドレスまたはホスト名を入力します。5 つの異なるホストを指定できます。

ステップ 12　Server Port フィールドに、RADIUS サーバが待ち受けるポート番号を入力します。5 つの異なるポートを指定できます。

ステップ 13 この設定を保存するには、**Submit** をクリックします。

表 15-6　RADIUS サーバの設定

GUI パラメータ	機能	CLI コマンド
Enable RADIUS Servers	RADIUS サーバを使用して HTTP 認証を有効にします。	**radius-server enable**
Time to wait	特定の RADIUS サーバに接続するときに、タイムアウトするまで応答を待つ秒数。範囲は 1 ～ 20 秒です。デフォルト値は 5 秒です。	**radius-server timeout**
Number of retransmits	RADIUS サーバへの接続を許可する回数。デフォルト値は 2 回です。	**radius-server retransmit**
Enable redirect	RADIUS サーバを使用した認証要求が失敗した場合に、認証応答を別の認証サーバにリダイレクトします。	**radius-server redirect enable**
Redirect Message	リダイレクトが行われたときにユーザに送られるメッセージ。	**radius-server redirect message**
Location	HTML ページ ロケーションを設定します。これは認証が失敗した場合に送信されるリダイレクト メッセージの URL の宛先です。	**radius-server redirect message** *reply* **location** *url*
Shared Encryption Key	RADIUS サーバと共有する暗号鍵。	**radius-server key** *keyword*
Server Name	RADIUS サーバの IP アドレスまたはホスト名。	**radius-server host** {*hostname* | *ipaddress*}
Server Port	RADIUS サーバが受信するポート番号。	**radius-server host auth-port** *port*

TACACS+ サーバの設定

TACACS+ データベースは、ユーザのアクセスが Content Engine に届く前にユーザを確認します。TACACS+ は、RFC 1492 を起源とし、非特権モードおよび特権モードのアクセスに対する追加制御方式としてシスコシステムズが使用しています。ACNS 5.x ソフトウェアは、TACACS+ のみをサポートし、TACACS および Extended TACACS をサポートしていません。

TACACS+ を有効にし HTTP 要求認証を実行するには、**tacacs enable** グローバル コンフィギュレーション コマンドを使用します。

ヒント Content Distribution Manager は、ユーザ認証情報をキャッシュしません。したがって、要求が発生するごとに TACACS+ サーバに対してユーザは再認証されることになります。多数の認証要求によるパフォーマンスの低下を防止するために、Content Distribution Manager は、TACACS+ サーバと同じロケーションにインストールしておくか、できるだけ TACACS+ サーバの近くにインストールして、認証要求ができるだけ迅速に行われるようにしてください。

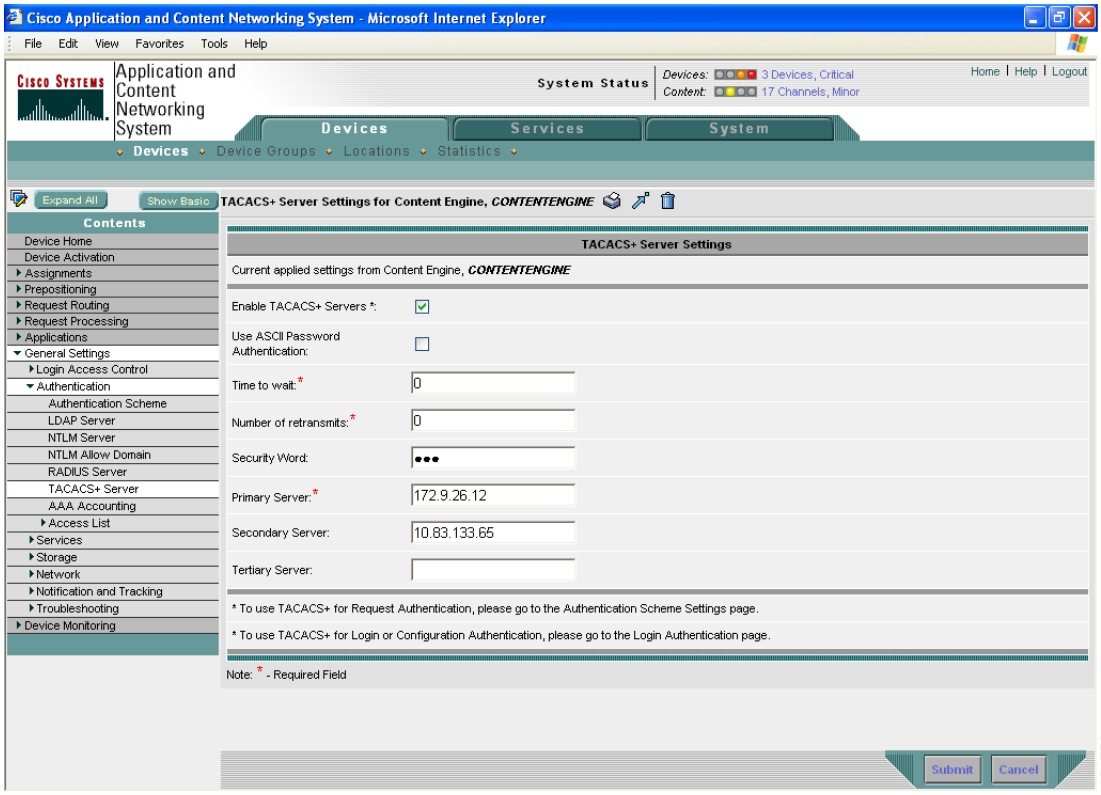
Content Distribution Manager GUI を使用して TACACS+ サーバを設定する手順は、次のとおりです。

ステップ 1　Content Distribution Manager GUI から、**Devices > Devices** の順に選択します。

ステップ 2　設定する Content Engine の名前の横にある **Edit** アイコンをクリックします。Contents ペインが左側に表示されます。

ステップ 3　Contents ペインから、**General Settings > Authentication > TACACS+ Server** の順に選択します。TACACS+ Server Settings ウィンドウが表示されます（図 15-8 を参照）。表 15-7 では、このウィンドウ内のフィールドについて説明します。

図 15-8　TACACS+ Server Settings ウィンドウ



ステップ 4　TACACS+ 認証を有効にするには、**Enable TACACS+ Servers** チェックボックスにチェックマークを付けます。

ステップ 5　認証に ASCII パスワード タイプを使用するには、**Use ASCII Password Authentication** チェックボックスにチェックマークを付けます。デフォルトのパスワード タイプは、パスワード認証プロトコル（PAP）です。ただし、認証パケットが ASCII クリアテキスト形式で送信される場合は、パスワード タイプを ASCII に変更できます。

ステップ 6　Time to wait フィールドに、タイムアウトになるまで Content Engine が待機する時間を指定します。デフォルト値は 5 秒です。

ステップ 7　Number of Retransmits フィールドに TACACS+ サーバへの接続に成功するまでの試行を許可する回数を指定します。デフォルト値は 2 です。

ステップ 8　Security Word フィールドに、TACACS+ サーバとの通信に使用する秘密鍵を入力します。

ステップ 9　Primary Server フィールドに、プライマリ TACACS+ サーバの IP アドレスまたはホスト名を入力します。

ステップ 10　Secondary Server フィールドに、セカンダリ TACACS+ サーバの IP アドレスまたはホスト名を入力します。

ステップ 11　Tertiary Server フィールドに、第 3 の TACACS+ サーバの IP アドレスまたはホスト名を入力します。

ステップ 12　この設定を保存するには、**Submit** をクリックします。

表 15-7　TACACS+ サーバの主要パラメータ

GUI パラメータ	機能	CLI コマンド
Enable TACACS+ Servers	TACACS+ 認証を有効にします。	**tacacs enable**
Use ASCII Password Authentication	デフォルトのパスワード タイプを PAP から ASCII クリアテキスト形式に変更します。	**tacacs password ascii**
Time to wait	特定の TACACS+ サーバに接続するときにタイムアウトが発生するまで応答を待つ秒数。範囲は 1 ～ 20 秒です。デフォルト値は 5 秒です。	**tacacs timeout**
Number of retransmits	TACACS+ サーバへの接続試行を許可する回数。デフォルト値は 2 回です。	**tacacs retransmit**
Security Word	TACACS+ サーバと共有する暗号鍵。	**tacacs key**
Primary Server	プライマリ TACACS+ サーバの IP アドレスまたはホスト名。	**tacacs host** {*hostname* | *ipaddress*} [**primary**]
Secondary Server Tertiary Server	バックアップ TACACS+ サーバの IP アドレスまたはホスト名。バックアップ サーバを 2 台まで使用できます。	**tacacs host** {*hostname* | *ipaddress*}

要求認証の認証方式の設定

要求認証の認証方式を有効にする手順は、次のとおりです。

ステップ 1　Content Distribution Manager GUI から、**Devices > Devices** の順に選択します。

ステップ 2　設定する Content Engine の名前の横にある **Edit** アイコンをクリックします。Contents ペインが左側に表示されます。

ステップ 3　Contents ペインから、**General Settings > Authentication > Authentication Scheme** の順に選択します

ステップ 4　Authentication Scheme ドロップダウン リストから、次の認証方式のいずれかを選択します。

- Disable Authentication
- RADIUS
- LDAP
- NTLM
- TACACS+

（注）　このウィンドウから認証方式の設定を保存するには、事前に認証サーバを設定し、有効にしておく必要があります。

ステップ 5　この設定を保存するには、**Submit** をクリックします。