Red Hat Directory Server 12

サーバーおよびデータベースアクティビティーの監視

Red Hat Directory Server アクティビティー、レプリケーショントポロジー、および
データベースアクティビティーの監視

Last Updated: 2024-08-02

Red Hat Directory Server 12 サーバーおよびデータベースアクティビティーの監視

Red Hat Directory Server アクティビティー、レプリケーショントポロジー、およびデータベースアクティビティーの監視

法律上の通知

概要

Web コンソールと dsconf ユーティリティーを使用して、Directory Server とレプリケーショントポロジーを監視できます。ディレクトリーアクティビティーのトラブルシューティング、モニター、および分析に使用できるログファイルにイベントを記録するようにディレクトリーサーバーを設定できます。

目次

RED HAT DIRECTORY SERVER に関するフィードバックの提供

Red Hat のドキュメントおよび製品に関するご意見をお待ちしております。ドキュメントの改善点があればお知らせください。これを行うには、以下を行います。

- Jira を通じて Red Hat Directory Server ドキュメントに関するフィードバックを送信する場合 (アカウントが必要):
  1. Red Hat Issue Tracker にアクセスしてください。
  2. **Summary** フィールドにわかりやすいタイトルを入力します。
  3. **Description** フィールドに、ドキュメントの改善に関するご意見を記入してください。ドキュメントの該当部分へのリンクも追加してください。
  4. ダイアログの下部にある **Create** をクリックします。
- Jira を通じて Red Hat Directory Server 製品に関するフィードバックを送信する場合 (アカウントが必要):
  1. Red Hat Issue Tracker にアクセスしてください。
  2. **Create Issue** ページで、**Next** をクリックします。
  3. **Summary** フィールドに入力します。
  4. **Component** フィールドでコンポーネントを選択します。
  5. **Description** フィールドに以下の内容を入力します。
     a. 選択したコンポーネントのバージョン番号。
     b. 問題を再現するための手順、または改善のための提案。
  6. **Create** をクリックします。

第1章 ディレクトリーサーバーアクティビティーのモニタリング

Red Hat Directory Server は、パフォーマンスカウンターとログを使用してパフォーマンスデータを追跡および記録します。

- パフォーマンスカウンターは、ディレクトリーサーバーのパフォーマンスの測定値を提供します。パフォーマンスカウンターは、サーバーのディレクトリーサーバー、設定されたデータベース、データベースリンク (データベースのチェーン) の操作および情報に集中します。
- ログファイルには、サーバーアクティビティー中に発生したイベントが記録されます。パフォーマンスを監視するには、次のログを使用できます。
  - アクセスログ
  - エラーログ
  - 監査ログ
  - 監査失敗ログ
  - セキュリティーログ
    ログファイルの詳細は、Directory Server ログファイルの概要 を参照してください。

現在のディレクトリーサーバーアクティビティーに関する情報は、Web コンソールまたはコマンドラインを使用して入手できます。すべてのデータベースのキャッシュアクティビティーをモニターすることもできます。

注記

アクセスログはバッファリングされ、負荷の高いサーバーでも完全なアクセスロギングが可能になります。ただし、サーバーでイベントが発生した時刻と、イベントがログに記録された時刻には不一致があります。

1.1. コマンドラインを使用したディレクトリーサーバーのモニタリング

dsconf コマンドを使用すると、ディスクの使用状況、ディレクトリーに格納されているサーバー統計情報のクエリー、およびその他のメトリックをモニターして、パフォーマンスを追跡できます。

前提条件

- **dconf** ユーティリティーを使用するには、サーバーが実行中であることを確認する。

手順

- コマンドラインを使用してサーバーのパフォーマンスを監視するには、次を実行します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com monitor server
```

  ここでは、以下のようになります。
  - **-D** は、LDAP ディレクトリーに接続するための **bindDN** 引数です。
  - **cn=Directory Manager** は、LDAP 認証の **bindDN** 値です。

- **ldap** は、指定された Lightweight Directory Access Protocol (LDAP) URL を使用してサーバー統計情報を収集します。

1.2. サーバーモニタリング属性

dsconf コマンドは、ディレクトリーサーバーのモニタリング時に次の属性を返します。

表1.1 サーバーモニタリング属性

属性	説明
version	現在のディレクトリーのバージョン番号を識別します。
threads	リクエストを処理しているアクティブなスレッドの現在の数。レプリケーションやチェーンなどの内部サーバータスクは、必要に応じて追加のスレッドを作成できます。
connection	ディレクトリーマネージャーとしてディレクトリーにバインドすると、開いている接続ごとに次の概要情報が提供されます。 **fd**: 接続のファイル記述子。 **opentime**: 接続を開く時間。 **opscompleted**: 完了した操作の数。 **binddn**: ディレクトリーに接続するための識別名。 **rw**: 読み取り権限または書き込み権限でブロックされた接続。 デフォルトでは、この情報は Directory Manager で利用できます。ただし、ディレクトリーエントリーのアクセス制御命令 (ACI) 属性を編集し、情報へのアクセス権限で追加のユーザーを設定できます。
currentconnections	ディレクトリーによって現在サービス中の接続の数を識別します。
totalconnections	サーバーが始動後に処理する接続の数を識別します。
currentconnectionsatmaxthreads	現在 **max thread** 状態にある接続を表示します。
maxthreadsperconnhits	接続が **max thread** 状態に達した回数を表示します。

属性	説明
dtablesize	ディレクトリーで利用可能なファイル記述子の数を示します。各接続には、開いているインデックス、ログファイル管理、および **ns-slapd** ごとに 1 つのファイル記述子が必要です。基本的に、この値は、ディレクトリーがサービスを提供できる追加の同時接続の数を示します。ファイル記述子の詳細は、オペレーティングシステムのドキュメントを参照してください。
readwaiters	クライアントからデータの読み取りを待機するスレッドの数を特定します。
opsinitiated	サーバーが始動後に開始する操作の数を識別します。
opscompleted	サーバーが完了した操作の数を識別します。
entriessent	サーバーの始動後にクライアントに送信されたエントリーの数を識別します。
bytessent	サーバーの起動後にクライアントに送信されたバイト数を識別します。
currenttime	サーバーのスナップショット時刻を識別します。時間表示は UTC 形式のグリニッジ標準時 (GMT) です。
starttime	サーバーが起動した時刻を識別します。時間表示は UTC 形式のグリニッジ標準時 (GMT) です。
nbackends	サーバーサービスのバックエンド (データベース) の数を識別します。

1.3. WEB コンソールを使用したディレクトリーサーバーのモニタリング

Web コンソールは、ユーザーが管理タスクを実行できるブラウザーベースのグラフィカルユーザーインターフェイス (GUI) です。ディレクトリーサーバーパッケージは、Web コンソールのディレクトリーサーバーユーザーインターフェイスを自動的にインストールします。

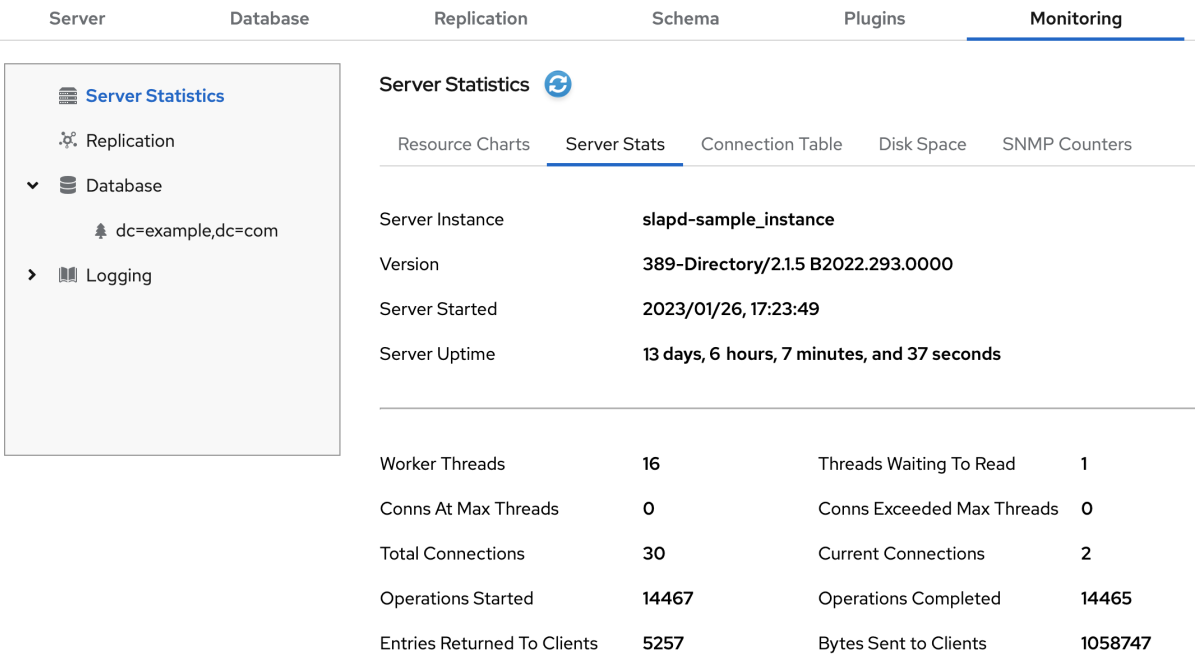
手順

1. Web コンソールでディレクトリーサーバーを開くには、ディレクトリーサーバーホストのポート 9090 で実行されている Web コンソールに接続します。

   ```
   https://server.example.com:9090
   ```

2. **root** ユーザーまたは **sudo** 権限でログインします。

3. **Monitoring** タブで、**Server Statistics** → **Server Stats** を選択します。



関連情報

- Web コンソールを使用したディレクトリーサーバーへのログイン

1.4. サーバー情報

ディレクトリーサーバーは、**Server Information** メニューの下に次のフィールドを表示します。

表1.2 サーバー情報

フィールド	説明
Server Instance	ディレクトリーサーバーインスタンスの名前を表示します。
バージョン	現在のサーバーバージョンを識別します。
Server Started	サーバーが稼働している日時。
Server Uptime	インスタンスが稼働している時間の測定。
Worker Threads	リクエストを処理するアクティブなスレッドの現在の数。レプリケーションやチェーンなどの内部サーバータスクは、必要に応じて追加のスレッドを作成できます。

フィールド	説明
Threads Waiting To Read	クライアントからの読み取りを待つスレッドの合計数。サーバーがクライアントから新しい要求を受信し、要求の送信を停止した場合、スレッドはすぐに読み取られない場合があります。一般に、待機中のスレッドは、低速のネットワークまたは低速のクライアントを示します。
Conns At Max Threads	現在、**max thread** 状態にあるすべての接続を表示します。
Conns Exceeded Max Threads	接続が **max thread** 状態に達した回数を表示します。
Total Connections	このディレクトリーサーバーインスタンスに確立された接続の総数。
Current Connections	オープン接続の合計数。各接続は複数の操作を開始できるため、複数のスレッドを開始できます。
Operations Started	接続によって開始された操作の数。
Operations Completed	すべての接続のためにサーバーが完了した操作の数。
Entries Returned to Clients	サーバーの起動後にクライアントに送信されるエントリーの数。

第2章 ヘルスチェックを使用した問題の特定

ヘルスチェックを実行して、潜在的な問題がないか Directory Server インスタンスを分析し、推奨されるソリューションを取得できます。

2.1. ディレクトリーサーバーヘルスチェックの実行

dsctl healthcheck コマンドを使用してヘルスチェックを実行します。

手順

- ヘルスチェックを実行するには、以下を入力します。

```
# dsctl instance_name healthcheck
Beginning lint report, this could take a while ...
Checking Backends ...
Checking Config ...
Checking Encryption ...
Checking FSChecks ...
Checking ReferentialIntegrityPlugin ...
Checking MonitorDiskSpace ...
Checking Replica ...
Checking Changelog ...
Checking NssSsl ...
Healthcheck complete.
1 Issue found!  Generating report ...
```

JSON 形式で出力を表示するには、**--json** パラメーターをコマンドに渡します。

```
# dsctl --json instance_name healthcheck
```

例2.1 ヘルスチェックの予想されるレポート

```
[1] DS Lint Error: DSELE0001
------------------------------------------------------------------------------
Severity: MEDIUM
Affects:
 -- cn=encryption,cn=config

Details:
-----------
This Directory Server may not be using strong TLS protocol versions. TLS1.0 is known to
have a number of issues with the protocol. Please see:

https://tools.ietf.org/html/rfc7457

It is advised you set this value to the maximum possible.

Resolution:
-----------
There are two options for setting the TLS minimum version allowed.  You,
can set "sslVersionMin" in "cn=encryption,cn=config" to a version greater than "TLS1.0"
You can also use 'dsconf' to set this value.  Here is an example:
```

```
    # dsconf slapd-instance_name security set --tls-protocol-min=TLS1.2

You must restart the Directory Server for this change to take effect.

Or, you can set the system wide crypto policy to FUTURE which will use a higher TLS minimum version, but doing this affects the entire system:

    # update-crypto-policies --set FUTURE


===== End Of Report (1 Issue found) =====
```

例2.2 JSON 形式のヘルスチェックの可能なレポート

```
[
    {
        "dsle": "DSELE0001",
        "severity": "MEDIUM",
        "items": [
            "cn=encryption,cn=config"
        ],
        "detail": "This Directory Server may not be using strong TLS protocol versions.
TLS1.0 is known to\nhave a number of issues with the protocol. Please
see:\n\nhttps://tools.ietf.org/html/rfc7457\n\nIt is advised you set this value to the
maximum possible.",
        "fix": "There are two options for setting the TLS minimum version allowed.  You,\ncan
set \"sslVersionMin\" in \"cn=encryption,cn=config\" to a version greater than
\"TLS1.0\"\nYou can also use 'dsconf' to set this value.  Here is an example:\n\n    #
dsconf slapd-instance_name security set --tls-protocol-min=TLS1.2\n\nYou must restart
the Directory Server for this change to take effect.\n\nOr, you can set the system wide
crypto policy to FUTURE which will use a higher TLS\nminimum version, but doing this
affects the entire system:\n\n    # update-crypto-policies --set FUTURE"
    }
]
```

関連情報

- ヘルスチェックの概要

2.2. ヘルスチェックの概要

dsctl healthcheck コマンドは次のチェックを実行します。

表2.1 ヘルスチェックの概要

コンポーネント	重大度	結果コード	説明
バックエンド	Low	DSBLE0003	データベースは初期化されませんでした。データベースは作成されていますが、空です。

コンポーネント	重大度	結果コード	説明
バックエンド	Medium	DSBLE0001	バックエンドのマッピングツリーエントリーが設定にありません。
コンフィグ	低	DSCLE0001	高解像度のタイムスタンプが無効になります。
コンフィグ	高	DSVIRTLE0001	仮想属性が誤ってインデックス化されています。ロールまたは CoS (Class of Service) 定義で使用されるインデックス化された属性により、検索結果が破損する可能性があります。
オペレーティングシステム	中	DSPERMLE0001	**/etc/resolve.conf** ファイルに設定されているパーミッションは、**0644** とは異なります。
オペレーティングシステム	高	DSDSLE0001	ディスク領域不足
オペレーティングシステム	高	DSPERMLE0002	**/etc/dirsrv/slapd-instance_name/pin.txt** および **/etc/dirsrv/slapd-instance_name/pwdfile.txt** ファイルに設定された権限は、**0400** とは異なります。
プラグイン	低	DSRILE0001	更新の遅延は Referential Integrity プラグインに設定されます。これにより、レプリケーションの問題が発生する可能性があります。
プラグイン	高	DSRILE0002	Referential Integrity プラグインにはインデックスがありません。プラグインは、インデックス化されていない場合にすべての削除操作に対して特定の属性をクエリーします。これにより、インデックスのない検索結果が検出されにくくなったり、CPU 使用率が高くなったりします。
レプリケーション	低	DSREPLLE0002	競合エントリーがデータベースに存在します。
レプリケーション	低	DSSKEWLE0001	レプリケーションタイムのずれが 6 時間より大きく、12 時間より小さくなっています。
レプリケーション	中	DSCLLE0001	changelog のトリミングは無効になっています。この場合、changelog は制限なしで増加します。
レプリケーション	中	DSREPLLE0004	ヘルスチェックは、レプリケーションのステータスを取得できませんでした。

コンポーネント	重大度	結果コード	説明
レプリケーション	中	DSREPLLE0003	トポロジーは同期されていませんが、レプリケーションは機能しています。
レプリケーション	中	DSREPLLE0005	リモートレプリカには到達できません。
レプリケーション	中	DSSKEWLE0002	レプリケーションタイムのずれが 12 時間より大きく、24 時間より小さくなっています。
レプリケーション	高	DSREPLLE0001	トポロジーが同期されておらず、レプリケーションが機能していません。
レプリケーション	高	DSSKEWLE0003	レプリケーションタイムのずれが 24 時間より大きい。レプリケーションセッションが破損する可能性がありました。
セキュリティー	中	DSELE0001	最小の TLS バージョンは TLS 1.2 未満の値に設定されます。
セキュリティー	高	DSCLE0002	パスワードストレージスキームが弱い。
Server	高	DSBLE0002	ヘルスチェックは、バックエンドのクエリーに失敗しました。
Transparent Huge Pages (THP)	Medium	DSTHPLE0001	THP が有効になっているため、Directory Server のパフォーマンスに影響する可能性があります。
TLS 証明書	中	DSCERTLE0001	サーバー証明書は次の 30 日以内に有効期限が切れます。
TLS 証明書	高	DSCERTLE0002	サーバー証明書の有効期限が切れている。

第3章 ログファイルの設定

ディレクトリーアクティビティーのトラブルシューティング、モニター、および分析に使用できるログファイルにイベントを記録するようにディレクトリーサーバーを設定できます。イベントをログファイルに記録するようにディレクトリーサーバーを設定することは、既存の問題を解決したり、障害やパフォーマンスの低下につながる可能性のある潜在的な問題を予測したりするために不可欠です。

3.1. ディレクトリーサーバーログファイルの概要

ディレクトリーサーバーは、4 種類のログファイルを **/var/log/dirsrv/slapd-instance_name/** ディレクトリーに保存します。

アクセスログ (access)

クライアント接続および Directory Server インスタンスへの接続試行に関する情報が含まれます。このログタイプは、デフォルトで有効になります。アクセスログはバッファーリングされるため、サーバー上でイベントが発生した時刻とログにイベントが記録された時刻との間に不一致が生じる場合があることに注意してください。

エラーログ (error)

通常の操作中にディレクトリーで発生するエラーとイベントの詳細なメッセージが含まれます。このログタイプは、デフォルトで有効になります。

警告

ディレクトリーサーバーが **error** ログファイルへのメッセージの書き込みに失敗した場合、サーバーはエラーメッセージを **syslog** サービスに送信して終了します。

監査ログ (audit)

各データベースおよびサーバー設定に対して行われた変更を記録します。このログタイプは、デフォルトでは有効になって **いません**。監査ログを有効にすると、Directory Server は成功した操作のみを **audit** ログファイルに記録します。ただし、監査失敗ログを有効にすると、失敗した操作を別のファイルに記録できます。

監査失敗ログ (audit-failure)

失敗した変更操作を記録します。このログタイプは、デフォルトでは有効になって **いません**。

セキュリティーログ (security)

認証イベント、認可の問題、DoS/TCP 攻撃、その他のセキュリティーイベントを記録します。

Directory Server ログファイルの詳細は、ログファイルリファレンス を参照してください。

3.2. ログファイルの表示

コマンドラインおよび Web コンソールを使用して Directory Server ログファイルを表示できます。

3.2.1. コマンドラインを使用したログファイルの表示

コマンドラインを使用してログファイルを表示するには、**less**、**more**、**cat** などの、Red Hat Enterprise Linux に含まれるユーティリティーを使用します。

手順

- たとえば、**access** ログファイルを表示するには、次のコマンドを使用します。

```
# less /var/log/dirsrv/slapd-instance_name/access
```

- ログファイルの場所を表示するには、次のコマンドを使用します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config get nsslapd-accesslog nsslapd-errorlog nsslapd-auditlog nsslapd-auditfaillog nsslapd-securitylog

nsslapd-accesslog: /var/log/dirsrv/slapd-instance_name/access
nsslapd-errorlog: /var/log/dirsrv/slapd-instance_name/errors
nsslapd-auditlog: /var/log/dirsrv/slapd-instance_name/audit
nsslapd-auditfaillog: /var/log/dirsrv/slapd-instance_name/audit-failure
nsslapd-securitylog: /var/log/dirsrv/slapd-instance_name/security
```

注記

指定したログタイプのログを有効にしていない場合、対応するログファイルは存在しません。

3.2.2. Web コンソールを使用したログファイルの表示

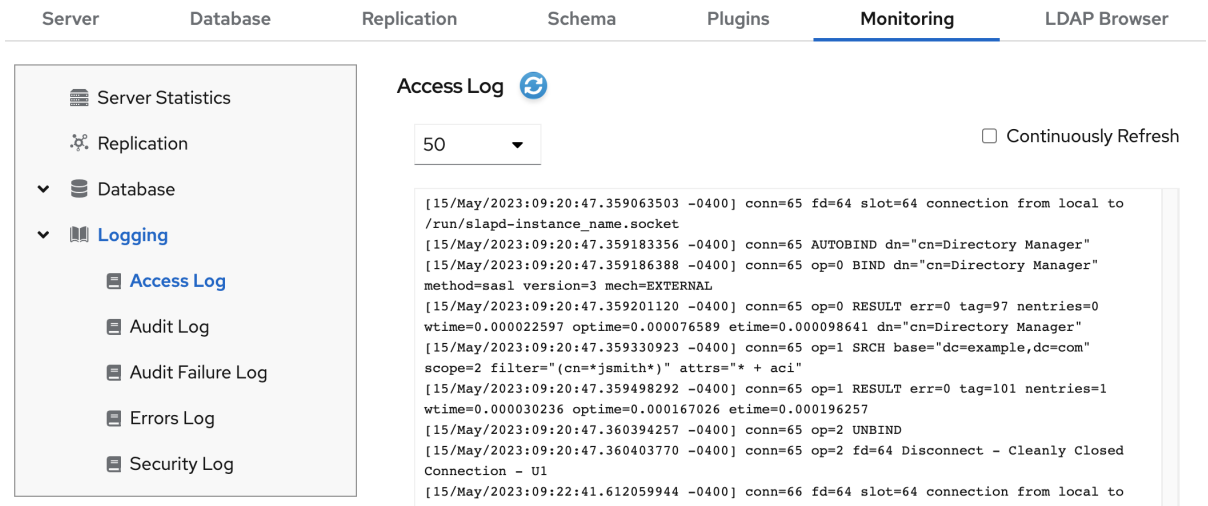
Web コンソールを使用して、ディレクトリーサーバーのログファイルを表示できます。

前提条件

- Web コンソールにログインしている。

手順

1. インスタンスを選択します。
2. **Monitoring → Logging** に移動します。
3. **Access Log** など、表示するログファイルを選択します。



4. 必要に応じて、以下の設定をログファイルビューアーに適用することができます。

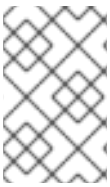
a. 表示するレコード数を設定します。

b. **Continuously Refresh** を選択して、新しいログエントリーを自動的に表示できるようにします。

5. **Refresh** ボタンをクリックして変更を適用します。

3.3. ログの有効化または無効化

デフォルトでは、ディレクトリーサーバーはアクセスとエラーのログを有効にし、監査と監査失敗のログを無効にします。



注記

アクセスログを無効にすると、ディレクトリーへの 2000 回のアクセスごとに約 1 MB のログファイルが増加するため、特定のシナリオで有用です。ただし、アクセスログをオフにする前に、この情報で問題のトラブルシューティングを行うことができます。

3.3.1. コマンドラインを使用したログの有効化または無効化

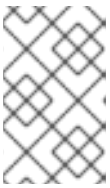
dsconf config replace コマンドを使用して、ディレクトリーサーバーのログ機能を管理する **cn=config** DN エントリーの次の属性を変更します。

- **nsslapd-accesslog-logging-enabled** (access log)
- **nsslapd-errorlog-logging-enabled** (error log)
- **nsslapd-auditlog-logging-enabled** (audit log)
- **nsslapd-auditfaillog-logging-enabled** (audit fail log)
- **nsslapd-securitylog-logging-enabled** (security log)

手順

- たとえば、アクセスログを有効にするには、次のコマンドで **nsslapd-accesslog-logging-enabled** 属性値を **on** に設定します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-accesslog-logging-enabled=on
```



注記

nsslapd-accesslog 属性に、ログファイルの有効なパスとファイル名が含まれていることを確認してください。そうしないと、アクセスログを有効にできません。

- たとえば、エラーログを無効にするには、次のコマンドで **nsslapd-errorlog-logging-enabled** 属性値を **off** に設定します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-errorlog-logging-enabled=off
```

関連情報

関連情報

- ログを有効にする属性の詳細は、コアサーバー設定属性の説明 で対応するセクションを参照してください。

3.3.2. Web コンソールを使用したログの有効化または無効化

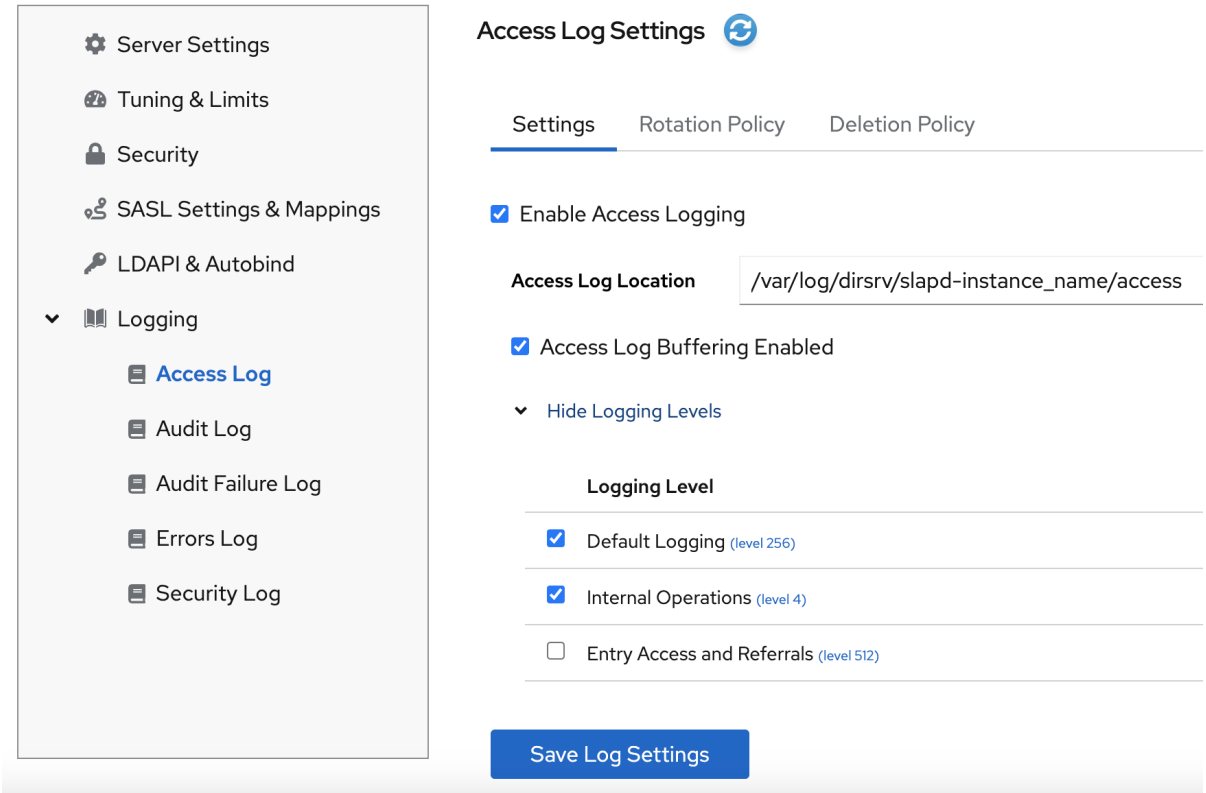
ディレクトリーサーバー Web コンソールを使用して、インスタンスのログを有効または無効にすることができます。

前提条件

- Web コンソールにログインしている。

手順

1. インスタンスを選択します。
2. **Server → Logging** に移動します。
3. 設定するログのタイプを選択します (例: **Access Log**)。
4. 選択したログタイプのログを有効または無効にします。
5. 必要に応じて、ログレベル、ログローテーションポリシー、ログバッファリングなどの追加設定を設定します。



6. **Save Log Settings** ボタンをクリックして、変更を適用します。

検証

- **Monitoring → Logging** に移動し、ディレクトリーサーバーがイベントをログに記録するようになったかどうかを確認します。

3.4. ログローテーションポリシーの定義

Directory Server は定期的に現在のログファイルをローテーションし、新しいログファイルを作成します。ただし、コマンドラインまたは Web コンソールを使用してローテーションポリシーを設定することもできます。管理できるローテーション設定は次のとおりです。

ログの最大数

保持するログファイルの最大数を設定します。ファイル数に達すると、Directory Server は新しいログファイルを作成する前に、最も古いログファイルを削除します。デフォルトでは、アクセスログの場合は **10**、その他のログの場合は **1** です。

最大ログサイズ (MB)

ログファイルがローテーションされるまでの最大サイズをメガバイト単位で設定します。デフォルトでは、すべてのログのサイズは **100** MB です。

新しいログを作成する間隔

ログファイルの最大期間を設定します。デフォルトでは、Directory Server はすべてのログを毎週ローテーションします。

時刻

ログファイルをローテーションする時刻を設定します。この設定は、デフォルトですべてのログに対して有効になっているわけではありません。

アクセスモード

アクセスモードでは、新規に作成されたログファイルにファイル権限を設定します。デフォルトでは、すべてのログで **600** です。

3.4.1. コマンドラインを使用したログローテーションポリシーの設定

dsconf config replace コマンドを使用して、ローテーションポリシーを管理する次の属性を変更できます。

	access log	error log	audit log	audit fail log	security log
ログの最大数	nsslapd-accesslog-maxlogsperdir	nsslapd-errorlog-maxlogsperdir	nsslapd-auditlog-maxlogsperdir	nsslapd-auditfaillog-maxlogsperdir	nsslapd-securitylog-maxlogsperdir
最大ログサイズ (MB)	nsslapd-accesslog-maxlogsize	nsslapd-errorlog-maxlogsize	nsslapd-auditlog-maxlogsize	nsslapd-auditfaillog-maxlogsize	nsslapd-securitylog-maxlogsize
新しいログを作成する間隔	nsslapd-accesslog-logrotationtime、nsslapd-accesslog-logrotationtimeunit	nsslapd-errorlog-logrotationtime、nsslapd-errorlog-logrotationtimeunit	nsslapd-auditlog-logrotationtime、nsslapd-auditlog-logrotationtimeunit	nsslapd-auditfaillog-logrotationtime、nsslapd-auditfaillog-logrotationtimeunit	nsslapd-securitylog-logrotationtime、nsslapd-securitylog-logrotationtimeunit

	access log	error log	audit log	audit fail log	security log
時刻	nsslapd-accesslog-logrotationsynchour、nsslapd-accesslog-logrotationsyncmin	nsslapd-errorlog-logrotationsynchour、nsslapd-errorlog-logrotationsyncmin	nsslapd-auditlog-logrotationsynchour、nsslapd-auditlog-logrotationsyncmin	nsslapd-auditfaillog-logrotationsynchour、nsslapd-auditfaillog-logrotationsyncmin	nsslapd-securitylog-logrotationsynchour、nsslapd-securitylog-logrotationsyncmin
アクセスモード	nsslapd-accesslog-mode	nsslapd-errorlog-mode	nsslapd-auditlog-mode	nsslapd-auditfaillog-mode	nsslapd-securitylog-mode

手順

1. たとえば、アクセスモード 600 を使用し、最大 2 つのログを保持し、ログファイルを 100 MB または 5 日ごとにローテーションするようにエラーログを設定するには、次のように入力します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-errorlog-mode=600 nsslapd-errorlog-maxlogsperdir=2 nsslapd-errorlog-maxlogsize=100 nsslapd-errorlog-logrotationtime=5 nsslapd-errorlog-logrotationtimeunit=day
```

関連情報

- ローテーションポリシー属性の詳細は、**設定およびスキーマリファレンス** ドキュメントの cn=config セクションで該当するセクションを参照してください。

3.4.2. Web コンソールを使用したログローテーションポリシーの設定

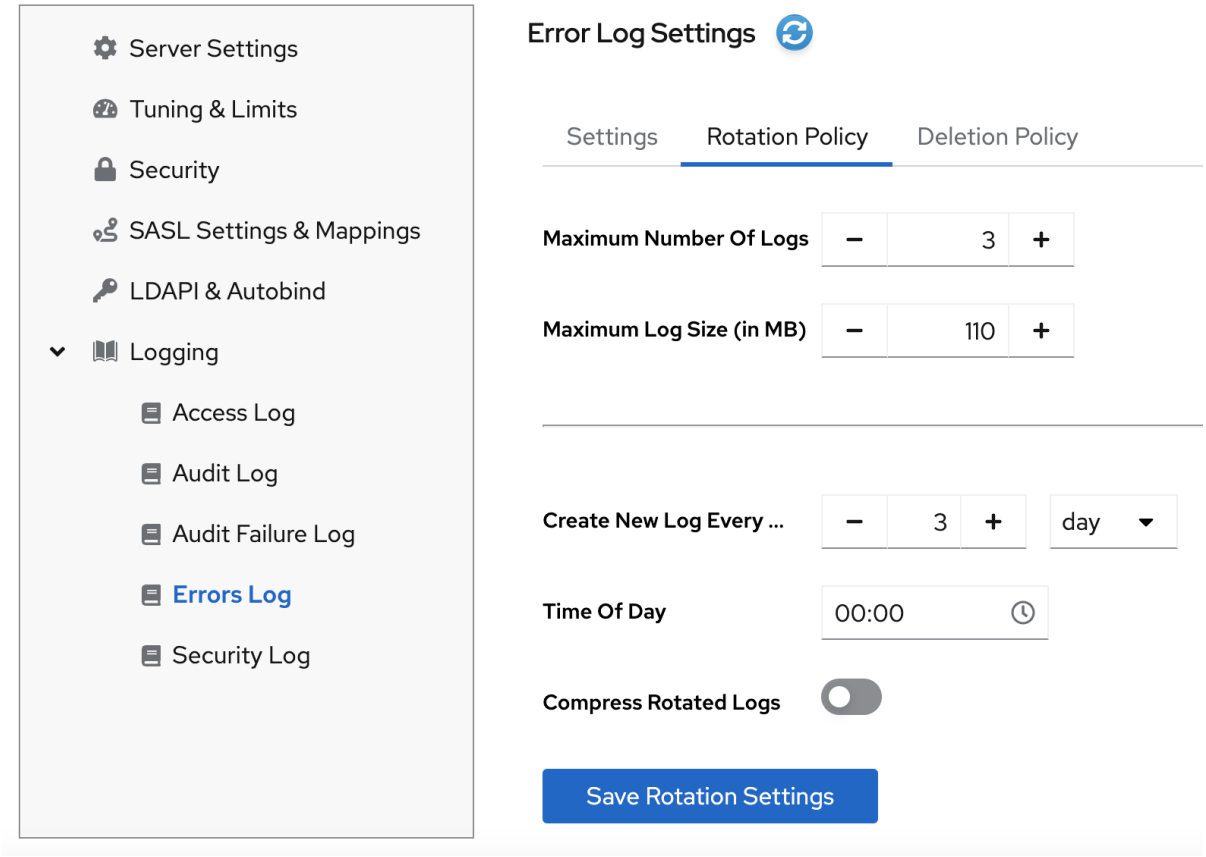
Web コンソールを使用してログローテーションポリシーを管理できます。

前提条件

- Web コンソールにログインしている。

手順

1. インスタンスを選択します。
2. **Server → Logging** に移動し、ログの種類 (**Error Log** など) を選択します。**Error Log Settings** ページが開きます。
3. **Rotation Policy** タブをクリックします。
4. ローテーションポリシーパラメーターを設定します。たとえば、最大 3 つのログファイル、最大 110 MB のログサイズ、および 3 日ごとに新しいログファイルの作成を設定します。



5. **Save Rotation Setting** ボタンをクリックして、変更を適用します。

関連情報

- ログ削除ポリシーの設定

3.5. ログ削除ポリシーの定義

ディレクトリーサーバーは、削除ポリシーを設定すると、アーカイブされた古いログファイルを自動的に削除します。

注記

ログファイルのローテーションポリシーが設定されている場合に限り、ログファイルの削除ポリシーを設定できます。ディレクトリーサーバーは、ログローテーション時に削除ポリシーを適用します。

次の属性を設定して、ログファイルの削除ポリシーを管理できます。

ログアーカイブの超過 (MB)

あるタイプのログファイルのサイズが設定値を超えると、その最も古いログファイルが自動的に削除されます。

空きディスク容量 (MB)

空きディスク容量がこの値に達すると、最も古いアーカイブファイルが自動的に削除されます。

ログファイルが次の日付より古い

ログファイルが設定された時間よりも古い場合は、これが自動的に削除されます。

3.5.1. コマンドラインを使用したログ削除ポリシーの設定

dsconf config replace コマンドを使用して、削除ポリシーを管理する次の属性を変更できます。

	access log	error log	audit log	audit fail log	security log
ログアーカイブの超過 (MB)	nsslapd-accesslog-logmaxdiskspace	nsslapd-errorlog-logmaxdiskspace	nsslapd-auditlog-logmaxdiskspace	nsslapd-auditfaillog-logmaxdiskspace	nsslapd-securitylog-logmaxdiskspace
空きディスク容量 (MB)	nsslapd-accesslog-logminfreediskspace	nsslapd-errorlog-logminfreediskspace	nsslapd-auditlog-logminfreediskspace	nsslapd-auditfaillog-logminfreediskspace	nsslapd-securitylog-logminfreediskspace
ログファイルが次の日付より古い	nsslapd-accesslog-logexpirationtime、nsslapd-accesslog-logexpirationtimeunit	nsslapd-errorlog-logminfreediskspace、nsslapd-errorlog-logexpirationtimeunit	nsslapd-auditlog-logminfreediskspace、nsslapd-auditlog-logexpirationtimeunit	nsslapd-auditfaillog-logminfreediskspace、nsslapd-auditfaillog-logexpirationtimeunit	nsslapd-securitylog-logminfreediskspace、nsslapd-securitylog-logexpirationtimeunit

手順

1. たとえば、すべてのアクセスログファイルの合計サイズが 500 MB を超える場合に、最も古いアクセスログファイルを自動的に削除するには、次のように実行します。

```
dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-accesslog-logmaxdiskspace=500
```

関連情報

- 削除ポリシー属性の詳細は、**設定およびスキーマリファレンス** ドキュメントの cn=config セクションで該当するセクションを参照してください。

3.5.2. Web コンソールを使用したログ削除ポリシーの設定

Web コンソールを使用してログ削除ポリシーを設定できます。

前提条件

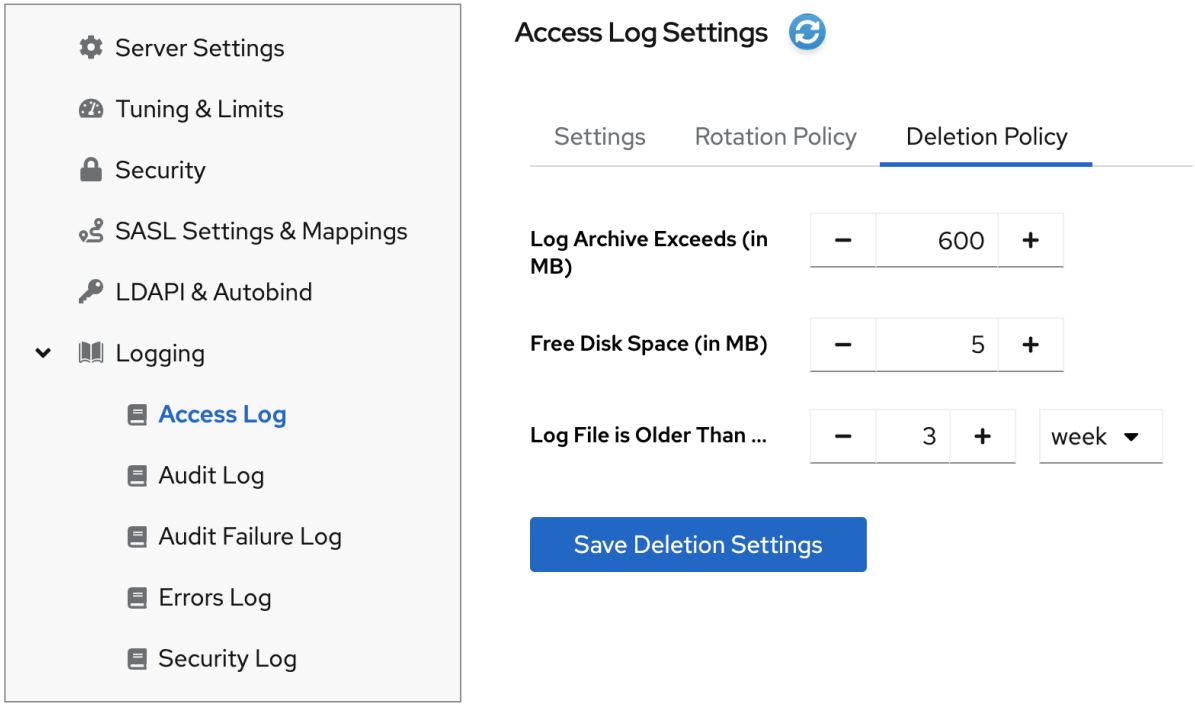
- Web コンソールにログインしている。

手順

1. インスタンスを選択します。
2. **Server → Logging** に移動し、ログのタイプ (**Access Log** など) を選択します。**Access Log Settings** ページが開きます。

3. **Deletion Policy** タブをクリックします。

4. 削除ポリシーパラメーターを設定します。たとえば、アーカイブの最大サイズを 600 MB に設定し、ログファイルの保存期間を 3 週間に設定します。



5. **Save Deletion Setting** ボタンをクリックして、変更を適用します。

関連情報

- ログローテーションポリシーの設定

3.6. 手動ログファイルローテーション

自動ログファイルローテーションまたは削除ポリシーを設定していない場合にのみ、ログファイルを手動でローテーションできます。

手順

1. インスタンスを停止します。

```
# dsctl instance_name stop
```

2. ログファイルディレクトリーに移動します。デフォルトでは、Directory Server はアクセスログ、エラーログ、監査ログ、監査失敗ログ、およびセキュリティーファイルを **/var/log/dirsrv/slapd-instance/** ディレクトリーに保存します。

3. ローテーションするログファイルを移動するか名前を変更して、後で参照できるようにします。

4. インスタンスを起動します。

```
# dsctl instance_name restart
```

関連情報

- ログローテーションポリシーの設定
- ログ削除ポリシーの設定

3.7. ログレベルの設定

ログがいかに詳細か、つまりログに記録される情報の量を管理するために、アクセスログとエラーログのログレベルを指定できます。

注記

デフォルトのログレベルを変更すると、ログファイルが非常に大きくなる可能性があります。Red Hat は、Red Hat テクニカルサポートからの依頼がない限り、デフォルトのログ値を変更 **しない** ことを推奨します。

3.7.1. コマンドラインを使用したログレベルの設定

次の設定属性を設定することで、ログレベルを調整できます。

- アクセスログの **nsslapd-accesslog-level**
- エラーログの **nsslapd-errorlog-level**

dsconf config replace コマンドを使用して、ログレベル属性を変更します。属性値は加算されます。たとえば、ログレベル値を 12 に設定する場合、レベル 8 と 4 が含まれます。

前提条件

- アクセスログとエラーログを有効にした。

手順

- たとえば、アクセスログの **Logging internal access operations** (4) と **Logging for connections, operations, and results** (256) を有効にするには、次のコマンドで **nsslapd-accesslog-level** 属性を 260 (4 + 256) に設定します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-accesslog-level=260
```

- たとえば、エラーログの **Search filter logging** (32) と **Config file processing** (64) のログレベルを有効にするには、次のコマンドで **nsslapd-errorlog-level** 属性を 96 (32 + 64) に設定します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-errorlog-level=96
```

検証

アクセスログレベルを **Logging internal access operations (4)** に設定した場合は、次の手順を実行して、ディレクトリーサーバーが内部アクセスイベントのログを記録し始めたかどうかを確認します。

1. インスタンスを再起動して、コマンドで内部イベントをトリガーします。

```
# dsctl instance_name restart
Instance "instance_name" has been restarted
```

2. アクセスログファイルを表示し、内部操作記録を見つけます。

```
# cat /var/log/dirsrv/slapd-instance_name/access
...
[08/Nov/2022:16:29:05.556977401 -0500] conn=2 (Internal) op=1(1)(1) SRCH
base="cn=config,cn=WritersData,cn=ldbm database,cn=plugins,cn=config" scope=1
filter="objectclass=vlvsearch" attrs=ALL
[08/Nov/2022:16:29:05.557250374 -0500] conn=2 (Internal) op=1(1)(1) RESULT err=0
tag=48 nentries=0 wtime=0.000016828 optime=0.000274854 etime=0.000288952
...
```

関連情報

- アクセスログレベル属性の説明
- エラーログレベル属性の説明

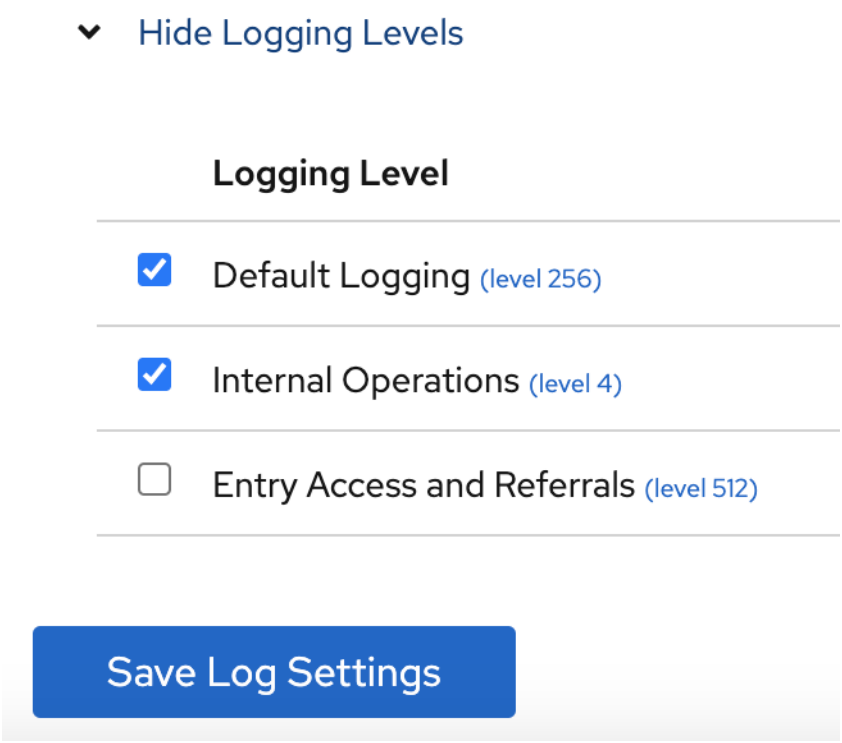
3.7.2. Web コンソールを使用したログレベルの設定

前提条件

- Web コンソールにログインしている。
- アクセスログとエラーログを有効にした。

手順

1. インスタンスを選択します。
2. **Server → Logging** に移動します。
3. **Access Log** などのログタイプを選択します。
4. **Show Logging Levels** ボタンをクリックして、ログタイプで使用可能なすべてのログレベルを表示します。



5. ログレベルを選択します。たとえば、**Default Logging** レベルや **Internal Operations** レベルなどです。

6. **Save Log Setting** ボタンをクリックして、変更を適用します。

検証

ディレクトリーサーバーが内部アクセスイベントの記録を開始したかどうかを確認するには、次の手順を実行します。

1. **Action** ボタンをクリックし、**Restart Instance** を選択してインスタンスを再起動します。ディレクトリーサーバーはインスタンスを再起動し、内部イベントを生成します。

2. **Monitoring → Logging → Access Log** に移動します。

3. アクセスログを更新し、記録された内部イベントを表示します。

```
[08/Nov/2022:17:04:17.035502206 -0500] conn=6 (Internal) op=1(2)(1) SRCH
base="cn=config,cn=Example database,cn=ldbm database,cn=plugins,cn=config" scope=1
filter="objectclass=vlvsearch" attrs=ALL
[08/Nov/2022:17:04:17.035579829 -0500] conn=6 (Internal) op=1(2)(1) RESULT err=0
tag=48 nentries=0 wtime=0.000004563 optime=0.000078000 etime=0.000081911
```

関連情報

- アクセスログレベル属性の説明
- エラーログレベル属性の説明

3.8. プラグインのログの設定

デフォルトでは、ディレクトリーサーバーはプラグインが開始する内部イベントをログに記録 **しません**。プラグイン操作をデバッグするには、すべてのプラグインまたは特定のプラグインに対してアクセスログおよび監査ログを有効にします。

3.8.1. すべてのプラグインのログの設定

nsslapd-plugin-logging 属性を使用して、すべてのプラグインのログを設定します。

手順

- すべてのプラグインのアクセスログおよび監査ログを有効にするには、次のコマンドを使用します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-plugin-logging=on
```

関連情報

nsslapd-plugin-logging 属性の詳細は、説明セクションを参照してください。

- nsslapd-plugin-logging

3.8.2. 特定のプラグインのログ設定

nsslapd-logAccess および **nsslapd-logAudit** 属性を使用して、プラグインのログを設定します。

前提条件

- **nsslapd-accesslog** 属性には、アクセスログファイルの有効なパスとファイル名が含まれる。
- **nsslapd-auditlog** 属性には、監査ログファイルの有効なパスとファイル名が含まれる。

手順

- 特定のプラグインのアクセスおよび監査ログを有効にするには、LDAP インターフェイスを使用して **nsslapd-logAccess** および **nsslapd-logAudit** 属性を変更します。

```
# ldapmodify -D "cn=Directory Manager" -W -x -H ldap://server.example.com:389

dn: cn=MemberOf Plugin,cn=plugins,cn=config
changetype: modify
replace: nsslapd-logAccess
nsslapd-logAccess: on

dn: cn=MemberOf Plugin,cn=plugins,cn=config
changetype: modify
replace: nsslapd-logAudit
nsslapd-logAudit: on
```

関連情報

属性の詳細については、説明のセクションを参照してください。

- nsslapd-logAccess

- nsslapd-logAudit

3.9. 検索操作ごとの統計情報をログに記録する

検索操作中、特に **(cn=user*)** などのフィルターを使用する場合、サーバーがタスクを受信してから結果を送り返すまでにかかる時間 (**etime**) が非常に長くなることがあります。

検索操作中に使用されたインデックスに関連する情報によってアクセスログを拡張すると、**etime** の値が大きい理由を診断するのに役立ちます。

nsslapd-statlog-level 属性を使用すると、サーバーへの影響を最小限に抑えながら、インデックス検索 (データベース読み取り操作) の数や各検索操作のインデックス検索の全体的な時間などの統計情報を収集できます。

前提条件

- アクセスログを有効にした。

手順

1. 検索操作メトリクスを有効にします。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace
nsslapd-statlog-level=1
```

2. インスタンスを再起動します。

```
# dsctl instance_name restart
```

検証

1. 検索操作を実行します。

```
# ldapsearch -D "cn=Directory Manager" -H ldap://server.example.com -b
"dc=example,dc=com" -s sub -x "cn=user*"
```

2. アクセスログファイルを表示し、検索統計情報のレコードを見つけます。

```
# cat /var/log/dirsrv/slapd-instance_name/access
...
[16/Nov/2022:11:34:11.834135997 +0100] conn=1 op=73 SRCH
base="dc=example,dc=com" scope=2 filter="(cn=user)"* attrs=ALL
[16/Nov/2022:11:34:11.835750508 +0100] conn=1 op=73 STAT read index:
attribute=objectclass key(eq)=referral --> count 0
[16/Nov/2022:11:34:11.836648697 +0100] conn=1 op=73 STAT read index: attribute=cn
key(sub)=er_ --> count 25
[16/Nov/2022:11:34:11.837538489 +0100] conn=1 op=73 STAT read index: attribute=cn
key(sub)=ser --> count 25
[16/Nov/2022:11:34:11.838814948 +0100] conn=1 op=73 STAT read index: attribute=cn
key(sub)=use --> count 25
[16/Nov/2022:11:34:11.841241531 +0100] conn=1 op=73 STAT read index: attribute=cn
key(sub)=^us --> count 25
[16/Nov/2022:11:34:11.842230318 +0100] conn=1 op=73 STAT read index: duration
0.000010276
```

```
[16/Nov/2022:11:34:11.843185322 +0100] conn=1 op=73 RESULT err=0 tag=101
nentries=24 wtime=0.000078414 optime=0.001614101 etime=0.001690742
...
```

関連情報

- nsslapd-statlog-level の説明へのリンク (追加予定)

3.10. ログファイルの圧縮

ディスク領域を節約するには、アーカイブされたログを **.gzip** ファイルに圧縮するログファイル圧縮を有効にします。

dsconf config replace コマンドを使用して、ログファイルの圧縮を管理する次の属性を変更します。

- **nsslapd-accesslog-compress** (アクセスログ)
- **nsslapd-errorlog-compress** (エラーログ)
- **nsslapd-auditlog-compress** (監査ログ)
- **nsslapd-auditfaillog-compress** (監査失敗ログ)
- **nsslapd-securitylog-compress** (セキュリティーログ)

デフォルトでは、Directory Server はアーカイブされたセキュリティーログファイルのみを圧縮します。

手順

- ログファイルの圧縮を有効にするには、次のコマンドを実行します。

  ```
  # dsconf -D "cn=Directory Manager" ldap://server.example.com config replace
  nsslapd-accesslog-compress=on nsslapd-errorlog-compress=on
  ```

  このコマンドは、アクセスログとエラーログの圧縮を有効にします。

- ログファイルの圧縮を無効にするには、次のコマンドを実行します。

  ```
  # dsconf -D "cn=Directory Manager" ldap://server.example.com config replace
  nsslapd-accesslog-compress=off
  ```

  このコマンドは、アクセスログの圧縮を無効にします。

検証

- ログファイルディレクトリーに圧縮されたログファイルが含まれていることを確認します。
  # ls /var/log/dirsrv/slapd-instance_name/

関連情報

- nsslapd-accesslog-compress 属性の説明
- nsslapd-errorlog-compress 属性の説明

- nsslapd-auditlog-compress 属性の説明
- nsslapd-auditfaillog-compress 属性の説明
- nsslapd-securitylog-compress 属性の説明

3.11. デバッグ目的でのアクセスログバッファーの無効化

デバッグの目的で、デフォルトで有効になっているアクセスログバッファーを無効にできます。アクセスログのバッファーが無効になっていると、Directory Server はログエントリーをディスクに直接書き込みます。

警告

通常の操作環境では、アクセスログを無効にしないでください。バッファーを無効にすると、特に負荷が大きい場合に、Directory Server のパフォーマンスが低下します。

3.11.1. コマンドラインを使用したアクセスログバッファーの無効化

アクセスログのバッファリングを無効にすると、ディレクトリーサーバーはログエントリーをディスクに直接書き込みます。

手順

1. アクセスログのバッファリングを無効にするには、次のように入力します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace
nsslapd-accesslog-logbuffering=off
```

検証

1. 連続モードでアクセスログを表示します。

```
# tail -f /var/log/dirsrv/slapd-instance_name/access
```

2. 検索など、ディレクトリー内でアクションを実行します。
3. アクセスログをモニターします。ユーザーがディレクトリーでアクションを実行すると、ログエントリーが遅延なく表示されます。

3.11.2. Web コンソールを使用したアクセスログバッファーの無効化

アクセスログのバッファリングを無効にすると、ディレクトリーサーバーはログエントリーをディスクに直接書き込みます。

手順

1. **Server → Logging → Access Log → Settings** に移動します。

2. **Access Log Buffering Enabled** の選択を解除します。
3. **Save Log Settings** をクリックします。

検証

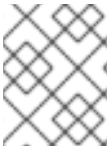
1. **Monitoring → Logging → Access Log** に移動します。
2. **Continuously Refresh** を選択します。
3. 検索など、ディレクトリー内でアクションを実行します。
4. アクセスログをモニターします。ユーザーがディレクトリーでアクションを実行すると、ログエントリーが遅延なく表示されます。

3.12. 高解像度のログタイムスタンプの無効化

デフォルトでは、ディレクトリーサーバーはエントリーをナノ秒の精度でログに記録します。

```
[29/Jun/2022:09:10:04.300970708 -0400] conn=81 op=13 SRCH
base="cn=dc\3Dexample\2Cdc\3Dcom,cn=mapping tree,cn=config" scope=0 filter="(objectClass=*)"
attrs="cn"
[29/Jun/2022:09:10:04.301010337 -0400] conn=81 op=13 RESULT err=0 tag=101 nentries=1
wtime=0.000038066 optime=0.000040347 etime=0.000077742
```

dsconf config replace コマンドを使用して、ログのタイムスタンプを担う属性を変更します。



注記

Red Hat は、高解像度ログのタイムスタンプを無効にするオプションを非推奨とし、今後のリリースで削除する予定です。

手順

- コマンドラインで高解像度ログのタイムスタンプを無効にするには、次のコマンドを入力します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-logging-hr-
timestamps-enabled=off
```

検証

- 新しいログレコードの精度が 2 番目であることを確認します。たとえば、次のコマンドでアクセスログファイルを開きます。

```
# less /var/log/dirsrv/slapd-instance_name/access
```

第4章 コマンドラインを使用したレプリケーショントポロジーの監視

サプライヤー、コンシューマー、およびハブの間のディレクトリーデータレプリケーションの状態を監視するには、レプリケーションの進行状況、レプリカ ID、変更の数、およびその他のパラメーターに関する情報を提供するレプリケーショントポロジーレポートを使用できます。レポートをより速く生成して読みやすくするために、独自の認証情報およびエイリアスを設定できます。

4.1. コマンドラインを使用したレプリケーショントポロジーレポートの表示

レプリケーショントポロジー内の各契約のレプリケーションステータスに関する全体的な情報を表示するには、レプリケーショントポロジーレポートを表示できます。これを行うには、**dsconf replication monitor** コマンドを使用します。

前提条件

- ホストがレプリケーショントポロジーのメンバーである。
- コンシューマーを初期化している。

手順

- レプリケーショントポロジーレポートを表示するには、次のように入力します。

```
# dsconf -D "cn=Directory Manager" ldap://supplier.example.com replication monitor
```

dsconf ユーティリティーは、トポロジー内の各インスタンスの認証情報を要求します。

```
Enter password for cn=Directory Manager on ldap://supplier.example.com: password
Enter a bind DN for consumer.example.com:389: cn=Directory Manager
Enter a password for cn=Directory Manager on consumer.example.com:389: password

Supplier: server.example.com:389
--------------------------------
Replica Root: dc=example,dc=com
Replica ID: 1
Replica Status: Online
Max CSN: 5e3acb77001d00010000

Status For Agreement: "example-agreement" (consumer.example.com:1389)
Replica Enabled: on
Update In Progress: FALSE
Last Update Start: 20211209122116Z
Last Update End: 20211209122116Z
Number Of Changes Sent: 1:21/0
Number Of Changes Skipped: None
Last Update Status: Error (0) Replica acquired successfully: Incremental update succeeded
Last Init Start: 20211209122111Z
Last Init End: 20211209122114Z
Last Init Status: Error (0) Total update succeeded
Reap Active: 0
Replication Status: In Synchronization
Replication Lag Time: 00:00:00
```

```
Supplier: consumer.example.com:1389
--------------------------------
Replica Root: dc=example,dc=com
Replica ID: 65535
Replica Status: Online
Max CSN: 00000000000000000000
```

関連情報

- .dsrc ファイルでのレプリケーション監視の認証情報の設定
- レプリケーショントポロジーモニタリング出力でのエイリアスの使用
- Web コンソールを使用したレプリケーショントポロジーレポートの表示

4.2. .DSRC ファイルでのレプリケーション監視の認証情報の設定

デフォルトでは、**dsconf replication monitor** コマンドは、リモートインスタンスに対して認証する際に、バインド DN およびパスワードを要求します。将来、レポートをより速く簡単に生成するために、ユーザーの **~/.dsrc** ファイルで、トポロジー内のサーバーごとにバインド DN と任意でパスワードを設定できます。

前提条件

- ホストがレプリケーショントポロジーのメンバーである。
- コンシューマーを初期化している。

手順

1. オプション: **~/.dsrc** ファイルを作成します。
2. **~/.dsrc** ファイルで、バインド DN およびパスワードを設定します。以下に例を示します。

```
[repl-monitor-connections]
connection1 = server1.example.com:389:cn=Directory Manager:*
connection2 = server2.example.com:389:cn=Directory Manager:[~/pwd.txt]
connection3 = hub1.example.com:389:cn=Directory Manager:S3cret
```

この例では、connection1 から connection3 を各エントリーのキーとして使用します。ただし、任意の一意のキーを使用できます。

dsconf replication monitor コマンドを実行すると、**dsconf** ユーティリティーはインスタンスのレプリカ合意で設定されたすべてのサーバーに接続します。このユーティリティーが **~/.dsrc** でホスト名を見つけると、定義された認証情報を使用してリモートサーバーに対して認証されます。上記の例では、**dsconf** はサーバーへの接続時に以下の認証情報を使用します。

Hostname	バインド DN	パスワードの設定方法
server1.example.com	cn=Directory Manager	パスワードを要求する
server2.example.com	cn=Directory Manager	**~/pwd.txt** からパスワードを読み取る

Hostname	バインド DN	パスワードの設定方法
hub1.example.com	cn=Directory Manager	S3cret

検証

- **dsconf replication monitor** コマンドを実行して、**dsconf** ユーティリティーが ~/**.dsrc** ファイルで設定された認証情報を使用するかどうかを確認します。詳細は以下を参照してください。

コマンドラインを使用したレプリケーショントポロジーレポートの表示

関連情報

- Web コンソールを使用したレプリケーション監視の認証情報の設定

4.3. レプリケーショントポロジーモニタリング出力でのエイリアスの使用

レポートを読みやすくするために、レポート出力に表示される独自のエイリアスを設定できます。デフォルトでは、レプリケーション監視レポートにはリモートサーバーのホスト名が含まれています。

前提条件

- ホストがレプリケーショントポロジーのメンバーである。
- コンシューマーを初期化している。

手順

レポートにエイリアスを表示する場合は、次のいずれかの方法を使用します。

- ~/**.dsrc** ファイルでエイリアスを定義します。

```
[repl-monitor-aliases]
M1 = server1.example.com:389
M2 = server2.example.com:389
```

- **-a alias=host_name:port** パラメーターを **dsconf replication monitor** コマンドに渡して、エイリアスを定義します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com replication monitor -a
M1=server1.example.com:389 M2=server2.example.com:389
```

どちらの場合も、**dsconf replication monitor** コマンドは出力にエイリアスを表示します。

```
...
Supplier: M1 (server1.example.com:389)
--------------------------------
Replica Root: dc=example,dc=com

...
```

```
Supplier: M2 (server2.example.com:389)
--------------------------------
Replica Root: dc=example,dc=com
```

関連情報

- Web コンソールを使用したレプリケーション命名エイリアスの設定

第5章 WEB コンソールを使用したレプリケーショントポロジーの監視

サプライヤー、コンシューマー、およびハブの間のディレクトリーデータレプリケーションの状態を監視するには、レプリケーションの進行状況、レプリカ ID、変更の数、およびその他のパラメーターに関する情報を提供するレプリケーショントポロジーレポートを使用できます。レポートをより速く生成して読みやすくするために、独自の認証情報およびエイリアスを設定できます。

5.1. WEB コンソールを使用したレプリケーショントポロジーレポートの表示

レプリケーショントポロジー内の各契約のレプリケーションステータスに関する全体的な情報を表示するには、レプリケーショントポロジーレポートを表示できます。

前提条件

- ホストがレプリケーショントポロジーのメンバーである。
- コンシューマーを初期化している。
- Web コンソールにログインしている。

手順

1. **Monitoring → Replication** に移動します。**Replication Monitoring** ページが開きます。
2. **Generate Report** をクリックします。
3. リモートインスタンスにログインするためのパスワードを入力し、**Confirm Credentials Input** をクリックします。Directory Server は、既存のレプリケーションアグリーメントのバインド DN 値を使用します。
   レプリケーショントポロジーレポートは、**Report Result** タブで生成されます。

注記

別のレプリケーショントポロジーレポートを生成するには、**Prepare Report** タブに移動します。

関連情報

- .dsrc ファイルでのレプリケーション監視の認証情報の設定
- レプリケーショントポロジーモニタリング出力でのエイリアスの使用
- Web コンソールを使用したレプリケーショントポロジーレポートの表示

5.2. WEB コンソールを使用したレプリケーション監視の認証情報の設定

レプリケーショントポロジーレポートをより迅速かつ簡単に生成するために、認証用のトポロジー内のサーバーごとに、独自のバインド DN と任意でパスワードを設定できます。この場合、レプリケーショントポロジーレポートを生成するたびにレプリケーション認証情報を確認する必要はありません。デフォルトでは、Directory Server は既存のレプリケーションアグリーメントからこれらの認証情報を取得します。

前提条件

- ホストがレプリケーショントポロジーのメンバーである。
- コンシューマーを初期化している。
- Web コンソールにログインしている。

手順

1. **Monitoring → Replication** に移動します。**Replication Monitoring** ページが開きます。
2. **Add Credentials** をクリックします。
3. リモートインスタンスへの認証に使用するレプリケーションのログイン認証情報を入力します。
   - **Hostname**。サーバーが認証するリモートインスタンスのホスト名。
   - **Port**。リモートインスタンスポート。
   - **Bind DN**。認証に使用される DN をリモートインスタンスにバインドします。
   - **Password**。認証に使用されるパスワード。
   - **Interactive Input**。オンにすると、レプリケーショントポロジーレポートを生成するたびに Directory Server がパスワードを要求します。
4. **Save** をクリックします。

検証

レプリケーショントポロジーレポートを生成して、レポートが認証情報を要求するかどうかを確認します。詳細は以下を参照してください。

Web コンソールを使用したレプリケーショントポロジーレポートの表示

5.3. WEB コンソールを使用したレプリケーション命名エイリアスの設定

レポートを読みやすくするために、レポート出力に表示される独自のエイリアスを設定できます。デフォルトでは、レプリケーション監視レポートにはサーバーのホスト名が含まれています。

前提条件

- ホストがレプリケーショントポロジーのメンバーである。
- コンシューマーを初期化している。
- Web コンソールにログインしている。

手順

1. **Monitoring → Replication** に移動します。**Replication Monitoring** ページが開きます。
2. **Add Alias** をクリックします。
3. エイリアスの詳細を入力します。

- **Alias**。レプリケーショントポロジーレポートに表示されるエイリアス。
- **Hostname**。インスタンスのホスト名。
- **Port**。インスタンスポート。

4. **Save** をクリックします。

検証

- レプリケーショントポロジーレポートを生成して、レポートが新しいエイリアスを使用しているかどうかを確認します。詳細は以下を参照してください。

Web コンソールを使用したレプリケーショントポロジーレポートの表示

第6章 プラグイン開始更新のバインド DN の追跡

Directory Server では、プラグインがエントリーを更新するアクションを実行するユーザーを追跡できます。追跡が有効になり、ユーザーが実行するアクションの結果としてプラグインがエントリーを変更した場合は、更新されたエントリーの **modifiersname** 属性でユーザーの名前を確認できます。

6.1. プラグインによって実行されるエントリー変更のユーザー情報の追跡

ユーザーがエントリーを変更するアクションを実行すると、ディレクトリーツリー全体で他の自動変更をトリガーできます。デフォルトでは、Directory Server は、データ変更を開始したアクションを実行したユーザー名を追跡しません。ユーザー情報を追跡するには、**nsslapd-plugin-binddn-tracking** パラメーターを使用できます。

たとえば、管理者がユーザーを削除すると、Referential Integrity Postoperation プラグインはすべてのグループからユーザーを自動的に削除します。サーバーにバインドされているユーザーアカウントによって実行されたエントリーの初期アクションを確認できます。ただし、デフォルトで、関連するすべての更新はプラグインによって実行されているように表示され、どのユーザーが更新を開始したかの情報はありません。

2 つ目の例は、MemberOf プラグインを使用してグループメンバーシップでユーザーエントリーを更新する場合があります。グループアカウントの更新はバインドされたユーザーが実行済みとして表示されますが、ユーザーエントリーの編集は MemberOf プラグインによって実行されると表示されます。

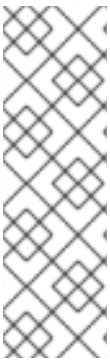
```
dn: cn=example_group,ou=groups,dc=example,dc=com
modifiersname: uid=example,ou=people,dc=example,dc=com

dn: uid=example,ou=people,dc=example,dc=com
modifiersname: cn=MemberOf Plugin,cn=plugins,cn=config
```

nsslapd-plugin-binddn-tracking パラメーターにより、サーバーは、どのユーザーが更新操作を開始したか、また操作を実際に実行した内部プラグインを追跡できます。バインドされたユーザーは **modifiersname** および **creatorsname** の運用属性に表示されますが、更新を実行したプラグインは **internalModifiersname** および **internalCreatorsname** 操作属性に表示されます。以下に例を示します。

```
dn: uid=example,ou=people,dc=example,dc=com
modifiersname: uid=admin,ou=people,dc=example,dc=com
internalModifiersname: cn=MemberOf Plugin,cn=plugins,cn=config
```

nsslapd-plugin-binddn-tracking パラメーターは、バインドされたユーザーと、その接続に対して実行される更新の関係を追跡し、維持します。



注記

internalModifiersname 属性および **internalCreatorsname** 属性は、常にプラグインをアイデンティティーとして表示します。属性の値は次のとおりです。

- **cn=ldbm database,cn=plugins,cn=config**(コアの Directory Server が変更を実行する際に)
- プラグインがエントリーを変更した場合の **cn=the DN of the plug-in,cn=plugins,cn=config**

6.2. コマンドラインで開始した更新のバインド DN の追跡の有効化

プラグインによって開始されるデータ更新の場合は、更新の原因となったアクションを実行したユーザーを把握しておく必要があります。コマンドラインで、**nsslapd-plugin-binddn-tracking** パラメーターを設定して、このようなユーザー情報を追跡します。

手順

- **nsslapd-plugin-binddn-tracking** パラメーターを **on** に設定します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace
nsslapd-plugin-binddn-tracking=on
```

検証

- プラグインによって変更されたエントリーの **modifiersname** および **internalModifiersname** 属性を表示します。たとえば **memberOf** 属性が有効な場合は、ユーザーをグループに追加した後にユーザーの属性を表示します。

```
# ldapsearch -D "cn=Directory Manager" -W -H ldap://server.example.com -x -b "uid=example-
user,ou=People,dc=example,dc=com" -s base -x internalModifiersname -x modifiersname
dn: uid=example-user,ou=people,dc=example,dc=com
modifiersname: uid=admin,ou=people,dc=example,dc=com
internalModifiersname: cn=MemberOf Plugin,cn=plugins,cn=config
```

関連情報

- プラグインによって実行されるエントリー変更のユーザー情報の追跡

6.3. WEB コンソールを使用したプラグイン開始の更新のバインド DN の追跡の有効化

プラグインによって開始されるデータ更新の場合は、更新の原因となったアクションを実行したユーザーを把握しておく必要があります。Web コンソールを使用して、ユーザー情報の追跡を有効にできます。

前提条件

- Web コンソールで Directory Server インスタンスにログインしている。

手順

1. **Server → Server Settings** メニューを開きます。
2. **Advanced Settings** タブで **Enable Plugin Bind DN Tracking** を選択します。
3. **Save** をクリックします。

検証

- プラグインによって変更されたエントリーの **modifiersname** および **internalModifiersname** 属性を表示します。たとえば **memberOf** 属性が有効な場合は、ユーザーをグループに追加した後にユーザーの属性を表示します。

```
# ldapsearch -D "cn=Directory Manager" -W -H ldap://server.example.com -x -b "uid=example-user,ou=People,dc=example,dc=com" -s base -x internalModifiersname -x modifiersname
dn: uid=example-user,ou=people,dc=example,dc=com
modifiersname: uid=admin,ou=people,dc=example,dc=com
internalModifiersname: cn=MemberOf Plugin,cn=plugins,cn=config
```

関連情報

- プラグインによって実行されるエントリー変更のユーザー情報の追跡

第7章 データベースアクティビティーの監視

管理者は、データベースアクティビティーを監視して、キャッシュなどのチューニング設定が適切に設定されていることを確認する必要があります。

7.1. コマンドラインを使用したデータベースアクティビティーの監視

コマンドラインを使用して監視アクティビティーを表示するには、**cn=monitor,cn=database_name,cn=ldbm database,cn=plugins,cn=config** に格納されている動的に更新された読み取り専用属性を表示します。

手順

- データベースの現在のアクティビティーを表示するには、次のように入力します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com monitor backend userRoot
```

このコマンドは、**userRoot** データベースのアクティビティーを表示します。

関連情報

- データベース監視属性

7.2. WEB コンソールを使用したデータベースアクティビティーの監視

Web コンソールでは、Directory Server は、監視タブの **cn=monitor,cn=database_name,cn=ldbm database,cn=plugins,cn=config** から動的に更新された読み取り専用監視属性の値を表示します。

手順

1. **Monitoring → Database → database name** に移動します。
2. **Entry Cache** タブと **DN Cache** タブにキャッシュ値を表示します。

関連情報

- データベース監視属性

7.3. データベース監視属性

表7.1 継承の設定

属性	説明
readonly	データベースが読み取り専用モード (**1**) であるか、読み取り/書き込みモード (**0**) であるかを示します。
entrycachehits	成功したエントリーキャッシュルックアップの合計数。この値は、サーバーがデータベースからエントリーをリロードせずにエントリーキャッシュからエントリーを取得できた合計回数です。

属性	説明
entrycachetries	インスタンスを開始してからのエントリーキャッシュルックアップの総数。値は総数です。インスタンスが開始されたため、Directory Server はエントリーキャッシュからエントリーを取得しようとしました。
entrycachehitratio	エントリーキャッシュの数は、エントリーキャッシュのルックアップを成功させようとします。この数は、インスタンスを最後に開始してからのルックアップとヒットの合計に基づいています。エントリーキャッシュのヒット率が 100% に近いほど、優れています。 操作がエントリーキャッシュに存在しないエントリーを見つけようとするときはいつでも、サーバーはエントリーを取得するためにデータベースにアクセスする必要があります。したがって、この比率がゼロに近づくと、ディスクアクセスの数が増え、ディレクトリー検索のパフォーマンスが低下します。この比率を改善するには、データベースのエントリーキャッシュのサイズを増やします。 この比率を改善するには、データベースの **cn=database_name,cn=ldbm database,cn=plugins,cn=config** エントリーの **nsslapd-cachememsize** 属性の値を増やして、エントリーキャッシュのサイズを増やします。
currententrycachesize	エントリーキャッシュに現在存在するディレクトリーエントリーの合計サイズ (バイト単位)。 キャッシュに存在するエントリーのサイズを増やすには、データベースの **cn=database_name,cn=ldbm database,cn=plugins,cn=config** エントリーの **nsslapd-cachememsize** 属性の値を増やします。
maxentrycachesize	Directory Server がエントリーキャッシュに保持できるディレクトリーエントリーの最大サイズ (バイト単位)。 キャッシュに存在するエントリーのサイズを増やすには、データベースの **cn=database_name,cn=ldbm database,cn=plugins,cn=config** エントリーの **nsslapd-cachememsize** 属性の値を増やします。
currententrycachecount	特定のバックエンドのエントリーキャッシュに保存されているエントリーの現在の数。
maxentrycachecount	データベースのエントリーキャッシュに保存されるエントリーの最大数。 この値を調整するには、**cn=database_name,cn=ldbm database,cn=plugins,cn=config** エントリーの **nsslapd-cachesize** 属性の値を増やします。
dncachehits	サーバーが、再度正規化するのではなく、DN キャッシュから正規化された識別名 (DN) を取得することにより、要求を処理できた回数。

属性	説明
dncachetries	インスタンスを開始してからの DN キャッシュアクセスの総数。
dncachehitratio	キャッシュの比率は、DN キャッシュヒットの成功を試みます。この値が 100% に近いほど、優れています。
currentdncachesize	DN キャッシュに現在存在する DN の合計サイズ (バイト単位)。 DN キャッシュに存在するエントリーのサイズを増やすには、データベースの **cn=database_name,cn=ldbm database,cn=plugins,cn=config** エントリーの **nsslapd-dncachememsize** 属性の値を増やします。
maxdncachesize	Directory Server が DN キャッシュに保持できる DN の最大サイズ (バイト単位)。 キャッシュに存在するエントリーのサイズを増やすには、データベースの **cn=database_name,cn=ldbm database,cn=plugins,cn=config** エントリーの **nsslapd-dncachememsize** 属性の値を増やします。
currentdncachecount	DN キャッシュに現在存在する DN の数。
maxdncachecount	DN キャッシュで許可される DN の最大数。

第8章 コマンドラインを使用したディレクトリーサーバーアクセスログの取得

logconv.pl コマンドは、ディレクトリーサーバーのアクセスログを分析し、統計情報の使用状況を抽出し、コマンドラインで指定された重要なイベントの発生回数を数えます。**logconv.pl** コマンドは、総操作数、総接続数、各操作タイプごとのカウント、永続的な検索などの拡張操作のカウント、およびバインド情報のリストを表示します。

logconv.pl コマンドの構文は次のとおりです。

```
logconv.pl/path/to/accesslog
```

複数のアクセスログファイルを分析するには、次の形式とアスタリスク (*) を使用します。

```
logconv.pl /var/log/dirsrv/slapd-instance_name/access*
```

logconv.pl コマンドは、ディレクトリーサーバーのモニタリングとディレクトリーサーバー設定の最適化に役立つ、次の 3 種類の統計情報を生成します。

- 実行された合計バインド数や合計検索数などのイベント数。
- LDAP 要求で最も頻繁に発生するパラメーターのリスト。たとえば、**logconv.pl** コマンドは、返された上位 10 個のバインド DN、ベース DN、フィルター文字列、および属性のリストを生成します。
- **ldap.h** で定義されているようなエラーコードの発生回数。

8.1. コマンドラインを使用したディレクトリーサーバーのアクセスログの分析

logconv.pl コマンドは、ディレクトリーサーバーのアクセスログを分析し、統計情報の使用状況を抽出し、重要なイベントの発生をカウントします。

logconv.pl には、次のオプションを使用します。

- **-S**: ログファイルの分析を開始する時刻を指定します。
- **-E**: ログファイルの分析を停止する時間を指定します。
- **-bc**: サーバーへの接続に使用される DN の数と、サーバーが返す合計接続コードに基づいてレポートを生成します。
- **-m**: 1 秒あたりの出力データ (**-m**) を指定された CSV 出力ファイルに生成します。
- **-M**: 1 分あたりの出力データ (**-M**) を指定された CSV 出力ファイルに生成します。

手順

- 簡単なアクセスログの概要を生成するには、次のコマンドを実行します。

```
# logconv.pl /var/log/dirsrv/slapd-instance_name/access

Access Log Analyzer 8.2
Command: logconv.pl /var/log/dirsrv/slapd-instance_name/access
```

```
Processing 1 Access Log(s)...

[001] /var/log/dirsrv/slapd-instance_name/access size(bytes):77532

Total Log Lines Analysed:  527
Start of Logs:    14/Oct/2017:16:15:22.452909568
End of Logs:      14/Oct/2017:16:39:50.157790196

Processed Log Time:  0 Hours, 24 Minutes, 27.704877056 Seconds

Restarts:                    10
Secure Protocol Versions:
 - TLS1.2 client bound as uid=user_name,ou=people,o=example.com (11 connections)
 - TLS1.2 128-bit AES; client CN=CA Subsystem,O=example.com; issuer CN=Certificate
Authority,O=example.com (11 connections)
 - TLS1.2 128-bit AES-GCM (2 connections)
 - TLS1.2 128-bit AES (3 connections)

Peak Concurrent Connections:  38
Total Operations:            4771
Total Results:               4653
Overall Performance:         97.5%

Total Connections:           249         (0.17/sec)  (10.18/min)
 - LDAP Connections:         107          (0.07/sec)  (4.37/min)
 - LDAPI Connections:        128          (0.09/sec)  (5.23/min)
 - LDAPS Connections:        14           (0.01/sec)  (0.57/min)
 - StartTLS Extended Ops:    2            (0.00/sec)  (0.08/min)

Searches:                    2963        (2.02/sec)  (121.13/min)
Modifications:               649         (0.44/sec)  (26.53/min)
Adds:                        785         (0.53/sec)  (32.09/min)
Deletes:                     10          (0.01/sec)  (0.41/min)
Mod RDNs:                    6           (0.00/sec)  (0.25/min)
Compares:                    0           (0.00/sec)  (0.00/min)
Binds:                       324         (0.22/sec)  (13.25/min)

Proxied Auth Operations:     0
Persistent Searches:         17
Internal Operations:         0
Entry Operations:            0
Extended Operations:         4
Abandoned Requests:          0
Smart Referrals Received:    0

VLV Operations:              30
VLV Unindexed Searches:      0
VLV Unindexed Components:    20
SORT Operations:             22

Entire Search Base Queries:  12
Paged Searches:              2
Unindexed Searches:          0
Unindexed Components:        149

FDs Taken:                   249
```

```
FDs Returned:                212
Highest FD Taken:            107

Broken Pipes:                0
Connections Reset By Peer:   0
Resource Unavailable:        0
Max BER Size Exceeded:       0

Binds:                       324
Unbinds:                     155
--------------------------------
 - LDAP v2 Binds:            41
 - LDAP v3 Binds:            180
 - AUTOBINDs(LDAPI):         103
 - SSL Client Binds:         0
 - Failed SSL Client Binds:  0
 - SASL Binds:               134
   - EXTERNAL: 114
   - GSSAPI: 20
 - Directory Manager Binds:  10
 - Anonymous Binds:          1

Cleaning up temp files...
Done.
```

logconv.pl スクリプトは、総操作数、総接続数、各操作タイプごとのカウント、永続的な検索などの拡張操作のカウント、およびバインド情報のリストを表示します。

- オプション: サーバーへの接続に使用される DN の数 (**b**) やサーバーが返す合計接続コード (**c**) など、単一のオプションとして渡される追加の接続サマリーを有効にする必要がある場合は、次のように **-bc** オプションを指定します。

```
# lotgconv.pl -bc /var/log/dirsrv/slapd-instance_name/access

----- Total Connection Codes -----
U1              3    Cleanly Closed Connections
B1              1    Bad Ber Tag Encountered

----- Top 20 Bind DN's -----
Number of Unique Bind DN's: 212
1801             cn=Directory Manager
1297             Anonymous Binds
311              uid=jsmith,ou=people...
87               uid=bjensen,ou=peopl...
85               uid=mreynolds,ou=peo...
69               uid=jrockford,ou=peo...
55               uid=sspencer,ou=peop...
```

- オプション: 特定の開始時間 (**-S**) および終了時間 (**-E**) または特定の範囲内でデータ出力を有効にする必要がある場合は、次のコマンドを実行します。

```
# logconv.pl -S "[01/Jul/2022:16:11:47.000000000 -0400]" -E "
[01/Jul/2022:17:23:08.999999999 -0400]" /var/log/dirsrv/slapd-instance_name/access
...
----------- Access Log Output ------------
```

```
Start of Logs:    01/Jul/2022:16:11:47
End of Logs:      01/Jul/2022:17:23:08
```

開始時間と終了時間が設定されると、**logconv.pl** コマンドが最初に指定の時間範囲を出力し、次にその期間の概要を出力します。

- オプション:1分あたり (**-M**) または1秒あたり (**-m**) の回数でデータ出力を有効にする必要がある場合は、次のコマンドを実行します。

```
# logconv.pl -m|-M outputFile accessLogFile
```