

Unicorn ID Manager

外部API 仕様書



OSSTech

オープンソース・ソリューション・テクノロジー(株)

文書番号: XXXXXXXXXXXX

作成者: 田村 広平

作成日: 2012年 2月 28日

更新日: 2012年 2月 28日

リビジョン: 1.0

目次

1. Unicorn ID Manager の外部 API 仕様概要	1
2. コマンドラインインターフェース	1
2.1 概要.....	1
2.2 詳細.....	1
2.2.1 ユーザー一括登録.....	1
2.2.2 ユーザー一括更新.....	3
2.2.3 ユーザー一括削除.....	4
2.2.4 ユーザー一括有効化.....	5
2.2.5 ユーザー一括無効化.....	6
2.2.6 パスワード変更.....	7
2.2.7 ユーザー一覧取得.....	8
2.2.8 ユーザー情報取得.....	8
2.2.9 グループ一括登録.....	8
2.2.10 グループ一括更新.....	9
2.2.11 グループ一括削除.....	11
2.2.12 スキーマ拡張.....	12
2.3 Unicorn IDM の CSV 属性仕様.....	13
3. REST API	15
3.1 ログイン	15
3.2 ログアウト	15
3.3 ユーザー情報検索	16
3.4 ユーザー登録	16
3.5 ユーザー更新	17
3.6 ユーザー削除	17
3.7 ユーザー一覧検索	18
3.8 ユーザー一括登録	18
3.9 ユーザー一括更新	19
3.10 ユーザー一括削除	20
3.11 グループ情報検索	21
3.12 グループ登録	22
3.13 グループ更新	22
3.14 グループ削除	23

3.15 グループ一覧検索	23
3.16 グループ一括登録	24
3.17 グループ一括更新	25
3.18 グループ一括削除	26

4. 改版履歴	28
----------------	-----------

1. Unicorn ID Manager の外部 API 仕様概要

本文書は Unicorn ID Manager が提供している外部 API の仕様について記載します。外部 API には大きく分けて、次の 2 つがあります。

- コマンドラインインターフェイス
- REST API

2. コマンドラインインターフェース

2.1 概要

Unicorn ID Manager は、外部API としてコマンドベースのバッチ処理機能を提供しています。

コマンドベースで提供している機能は、次のとおりです。

機能	コマンド	説明
ユーザー一括登録	unicornidm-account-csvhandle	CSV ファイルに指定したユーザーを登録
ユーザー一括更新	unicornidm-account-csvhandle	CSV ファイルに指定したユーザーを更新
ユーザー一括削除	unicornidm-account-csvhandle	CSV ファイルに指定したユーザーを削除
ユーザー一括有効化	unicornidm-account-csvhandle	CSV ファイルに指定したユーザーを有効化
ユーザー一括無効化	unicornidm-account-csvhandle	CSV ファイルに指定したユーザーを無効化
パスワード変更	unicornidm-passwd	指定したユーザーのパスワードを変更
ユーザー一覧取得	unicornidm-account-userlist	ユーザー一覧をファイルに取得
ユーザー情報取得	unicornidm-account-userinfo	ユーザー情報をファイルに取得
グループ一括登録	unicornidm-group-csvhandle	CSV ファイルに指定したグループを登録
グループ一括更新	unicornidm-group-csvhandle	CSV ファイルに指定したグループを更新
グループ一括削除	unicornidm-group-csvhandle	CSV ファイルに指定したグループを削除
スキーマ拡張	unicornidm-schema-extend	指定したオブジェクトクラスおよびその属性を追加・削除

登録・更新・削除は同一コマンドですが、引数により動作を変えます。

各コマンドは root ユーザー権限で実行する必要があります。

ユーザーの一括登録などの操作の結果は、結果ファイルに記録され、Unicorn ID Manager の Web 画面から確認することが可能です。

2.2 詳細

2.2.1 ユーザー一括登録

CSV ファイルに指定したユーザーの一括登録を行います。

CSV ファイルの書式は、UnicornIDM の設定に依存しますが、基本的には次の形式となります。

username	sn	givenName	password	description	...
takeda	武田	保真	secret123	技術部	
suzuki	鈴木	一郎	secret234	営業部	
tanaka	田中	竜也	secret345	人事部	

- CSV ファイルは UTF-8 で作成します。
- 1 行目はヘッダです。LDAP/AD に登録する属性の属性名を指定します。

コマンド書式

```
unicornidm-account-csvhandle -t <組織名> -f <ファイル名> -o useradd
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-account-csvhandle -t example -f add.csv -o useradd
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_useradd_example-ldap.result
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_useradd_example-ad.result
```

返り値

- 0 : バッチ処理の実行に成功
- 1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2 つの結果ファイル名が表示されています。

結果ファイルの内容は、「ユーザー名,結果,追加情報」からなるテキストファイルです。

登録処理に成功したユーザーの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
takeda,User already exists,
suzuki,OK,
tanaka,OK,
```

「追加情報」は、パスワードの自動発行を行った場合に、発行されたパスワード情報が記録されます。パスワードを CSV で指定した場合は、空となります。

2.2.2 ユーザー一括更新

CSV ファイルに指定したユーザーの一括更新を行います。

CSV ファイルの書式は、UnicornIDM の設定に依存しますが、基本的には次の形式となります。

username	password	description	...
takeda	new1234		
suzuki		技術部	
tanaka	new1134		

- 更新するエントリのみが必要です。
- CSV ファイルは UTF-8 で作成します。
- 1 行目はヘッダです。LDAP/AD で更新する属性の属性名を指定します。

コマンド書式

```
unicornidm-account-csvhandle -t <組織名> -f <ファイル名> -o usermod
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-account-csvhandle -t example -f add.csv -o usermod
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_usermod_example-ldap.result
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_usermod_example-ad.result
```

返り値

- 0 : バッチ処理の実行に成功
- 1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2 つの結果ファイル名が表示されています。

結果ファイルの内容は、「ユーザー名, 結果, 追加情報」からなるテキストファイルです。

更新処理に成功したユーザーの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
takeda,OK  
suzuki,OK,  
tanaka,OK,
```

|| 2.2.3 ユーザー一括削除

CSV ファイルに指定したユーザーの一括削除を行います。

CSV ファイルの書式は、次の形式です。

username
takeda
suzuki
tanaka

- 1 行目はヘッダです。 username を指定します。
- username 以外の列が存在しても無視されます。

コマンド書式

```
unicornidm-account-csvhandle -t <組織名> -f <ファイル名> -o userdel
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-account-csvhandle -t example -f add.csv -o userdel  
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_userdel_example-ldap.result  
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_userdel_example-ad.result
```

返り値

- 0 : バッチ処理の実行に成功
- 1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2つの結果ファイル名が表示されています。

結果ファイルの内容は、「ユーザー名,結果,追加情報」からなるテキストファイルです。

更新処理に成功したユーザーの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
takeda,OK
suzuki,OK,
tanaka,No such user,
```

2.2.4 ユーザー一括有効化

CSV ファイルに指定したユーザーの一括有効化を行います。

CSV ファイルの書式は、次の形式です。

username
takeda
suzuki
tanaka

- 1行目はヘッダです。 username を指定します。
- username 以外の列が存在しても無視されます。

コマンド書式

```
unicornidm-account-csvhandle -t <組織名> -f <ファイル名> -o userenable
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-account-csvhandle -t example -f enablelist.csv -o userenable
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_userenable_example-ldap.result
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_userenable_example-ad.result
```

返り値

0 : バッチ処理の実行に成功

1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2つの結果ファイル名が表示されています。

結果ファイルの内容は、「ユーザー名,結果,追加情報」からなるテキストファイルです。

更新処理に成功したユーザーの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
takeda,OK
suzuki,OK,
tanaka,No such user,
```

|| 2.2.5 ユーザー一括無効化

CSV ファイルに指定したユーザーの一括無効化を行います。

CSV ファイルの書式は、次の形式です。

username
takeda
suzuki
tanaka

- 1行目はヘッダです。 username を指定します。
- username 以外の列が存在しても無視されます。

コマンド書式

```
unicornidm-account-csvhandle -t <組織名> -f <ファイル名> -o userdisable
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-account-csvhandle -t example -f enablelist.csv -o userdisable
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_userdisable_example-ldap.result
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_userdisable_example-ad.result
```

返り値

0 : バッチ処理の実行に成功

1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2つの結果ファイル名が表示されています。

結果ファイルの内容は、「ユーザー名,結果,追加情報」からなるテキストファイルです。

更新処理に成功したユーザーの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
takeda,OK
suzuki,OK,
tanaka,No such user,
```

|| 2.2.6 パスワード変更

任意のユーザーのパスワードを強制的にリセットすることができます。

コマンド書式

```
unicornidm-passwd -t <組織名> -u <ユーザー名> [-s]
```

-s オプションを指定しない場合、パスワードはプロンプトから入力します。

-s オプションを指定する場合、パスワードは標準入力から入力します。

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-passwd -t example takeda
(New) password:
Retype (New) password:
```

返り値

0: パスワードの変更に成功

1: パスワードの変更に失敗

パスワードの変更に失敗した場合、標準エラー出力にエラー原因が出力されます。

2.2.7 ユーザー一覧取得

ユーザー一覧を CSV 形式で取得することができます。

コマンド書式

```
unicornidm-account-userlist -t <組織名> -f <ファイル名>
```

- 取得した CSV ファイルはシフト JIS 形式です。
- パスワードは含まれません。

2.2.8 ユーザー情報取得

指定したユーザーのユーザー情報を CSV 形式で取得することができます。

コマンド書式

```
unicornidm-account-userinfo -t <組織名> -u <ユーザー名> -f <ファイル名>
```

- 取得した CSV ファイルはシフト JIS 形式です。
- パスワードは含まれません。

2.2.9 グループ一括登録

CSV ファイルに指定したグループの一括登録を行います。

CSV ファイルの書式は、UnicornIDM の設定に依存しますが、基本的には次の形式となります。

groupname	gidNumber	description	addMembers	...	
strudents	1001	学生	[user01,user02]		
teachers	1002	教師	[user11]		
other	1003	その他			

- CSV ファイルは UTF-8 で作成します。
- 1 行目はヘッダです。LDAP/AD に登録する属性の属性名を指定します。
- addMembers に対応する値に対して、[] の中にユーザー名をカンマ区切りで指定することで、グループにユーザーを追加できます。

コマンド書式

```
unicornidm-group-csvhandle -t <組織名> -f <ファイル名> -o groupadd
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-group-csvhandle -t example -f add.csv -o useradd  
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_groupadd_example-ldap.result  
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_groupadd_example-ad.result
```

返り値

0 : バッチ処理の実行に成功

1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2 つの結果ファイル名が表示されています。

結果ファイルの内容は、「グループ名, 結果, 追加情報」からなるテキストファイルです。

登録処理に成功したグループの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
strudents,OK,  
teachers,OK,  
others,Group already exists,,
```

「追加情報」は、パスワードの自動発行を行った場合に、発行されたパスワード情報が記録されます。パスワードを CSV で指定した場合は、空となります。

2.2.10 グループ一括更新

CSV ファイルに指定したグループの一括更新を行います。

CSV ファイルの書式は、UnicornIDM の設定に依存しますが、基本的には次の形式となります。

groupname	gidNumber	description	addMembers	delMembers	...
strudents	1001	学生	[user03,user04]	[user01,user02]	
teachers	1002	教師	[user11]		
others	1003	その他			

- 更新するエントリのみが必要です。
- CSV ファイルは UTF-8 で作成します。
- 1 行目はヘッダです。LDAP/AD で更新する属性の属性名を指定します。
- addMembers に対応する値に対して、[] の中にユーザー名をカンマ区切りで指定することで、グループにユーザーを追加できます。
- delMembers に対応する値に対して、[] の中にユーザー名をカンマ区切りで指定することで、グループからユーザーを削除できます。

コマンド書式

```
unicornidm-group-csvhandle -t <組織名> -f <ファイル名> -o groupmod
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-group-csvhandle -t example -f add.csv -o groupmod
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_groupmod_example-ldap.result
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_groupmod_example-ad.result
```

返り値

- 0 : バッチ処理の実行に成功
- 1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2 つの結果ファイル名が表示されています。

結果ファイルの内容は、「グループ名,結果,追加情報」からなるテキストファイルです。

更新処理に成功したグループの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
students,OK,  
teachers,OK,  
others,OK,
```

|| 2.2.11 グループ一括削除

CSV ファイルに指定したグループの一括削除を行います。

CSV ファイルの書式は、次の形式です。

groupname
students
teachers
others

- 1 行目はヘッダです。 groupname を指定します。
- groupname 以外の列が存在しても無視されます。

コマンド書式

```
unicornidm-group-csvhandle -t <組織名> -f <ファイル名> -o groupdel
```

コマンド実行例

```
# /opt/osstech/sbin/unicornidm-group-csvhandle -t example -f add.csv -o groupdel  
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_groupdel_example-ldap.result  
/opt/osstech/var/lib/unicornIDM/results/example/20111019-182819_groupdel_example-ad.result
```

返り値

- 0 : バッチ処理の実行に成功
- 1 : CSV ファイルが読めないなどで、バッチ処理の実行に失敗

また、コマンドの終了時に、各レポジトリに対する結果ファイルのファイル名が標準出力に表示されます。

例えば、上記の例では、LDAP サーバーと Active Directory に連携している設定のため、2つの結果

ファイル名が表示されています。

結果ファイルの内容は、「グループ名,結果,追加情報」からなるテキストファイルです。

更新処理に成功したグループの「結果」は、「OK」となります。エラーの場合は、「結果」の部分に、エラー理由などの情報が記録されます。

```
students,OK,
teachers,OK,
others,OK,
```

2.2.12 スキーマ拡張

Unicorn ID Manager に対するスキーマ拡張を Web 画面を使わずに行うためのコマンドインターフェースを用意します。

次のコマンドインターフェースを用意します。

- /opt/osstech/sbin/unicornidm-schema-extend
 - スキーマ拡張のための Unicorn ID Manager の設定を行います。

いずれのコマンドも、次の引数をとります。

引数	内容
-t <target>	設定を適用する「組織」の名称を指定します。
-o <optype>	add:スキーマ追加、del:スキーマ削除
-c <objectClass>	オブジェクトクラスの名称を指定します。
-a <attributenames>	オブジェクトクラスの全属性名をカンマ区切りで指定します。 スキーマ削除の場合、このオプションは不要です。

コマンドの実行は root ユーザーで行います。

実行例

追加の場合

```
# /opt/osstech/sbin/unicornidm-schema-extend -t example-ldap -o add -c extPerson -a
companyCd,mobilephoneNo
```

削除の場合

```
# /opt/osstech/sbin/unicornidm-schema-extend -t example-ldap -o del -c extPerson
```

設定の更新に成功した場合、コマンドの返り値は0を返します。

設定の更新に失敗した場合、コマンドの返り値は1を返します。

追加されたオブジェクトクラスの属性は全て MAY(任意)属性となります。

追加した内容に不備があり、更新したい場合はいったん削除してから、再度追加します。

組織に対して同一のオブジェクトクラスを複数追加しようとするとエラーとなります。

2.3 Unicorn IDM の CSV 属性仕様

ユーザー登録時に値を入力する際の注意事項です。

- username
 - 登録するユーザーのユーザー名を指定するカラムです。
 - LDAP において
 - 明示的に cn が指定されていない場合、cn にも同じ値が設定されます。
 - 明示的に uid が指定されていない場合、uid にも同じ値が設定されます。
 - Active Directory において
 - cn にも同じ値が設定されます
 - sAMAccountName にも同じ値が設定されます
 - userPrincipalName には、「username@ドメイン名」が設定されます
- password
 - LDAP、Active Directory のパスワードとして設定されます

- LdapPrimaryGroup
 - gidNumber に登録する番号を、gidNumber 属性の代わりにグループ名で指定したいときに利用します。
- AdPrimaryGroup
 - Active Directory のプライマリグループを「Domain Users」以外に指定したい場合、グループ名を指定します。
- AdGroups
 - Active Directory のプライマリグループ以外のグループに所属させたいとき、グループ名を指定してください。「,」で区切って、複数のグループ名を指定することもできます。（事前に Active Directory 上にグループを作成してください）
- LdapGroups
 - OpenLDAP のプライマリグループ以外のグループに所属させたいときは、LdapGroups 属性に、グループ名を指定してください。「,」で区切って、複数のグループ名を指定することもできます。事前に OpenLDAP にグループを作成してください。
- unixHomeDirectory
 - UNIX 用のホームディレクトリの属性は、「homeDirectory」ではなく、「unixHomeDirectory」にホームディレクトリのパスを指定してください。（「homeDirecotry」は Active Directory でのホームディレクトリ属性となります）
- homeDirectory
 - Active Directory 用のホームディレクトリの属性は「homeDirectory」で指定してください。
- LdapUserSuffix
 - デフォルトのユーザー登録場所以外の OU にユーザーを登録する場合に指定します。
- AdUserSuffix
 - デフォルトのユーザー登録場所以外の OU にユーザーを登録する場合に指定します。

上記で指定した属性名以外は、LDAP、Active Directory のそれぞれの属性として、そのまま扱われます。

3. REST API

Unicorn ID Manager における REST API の仕様について説明します。

3.1 ログイン

以下の URL に GET メソッドでリクエストを送信することでログインします。

```
http://<UnicornIDM_Host>/unicornIDM/rest/login?username=<uname>&password=<passwd>
```

このリクエストにより、正常に認証が完了すると次のレスポンスを返します。

```
rest_id=FZ4FafLUod (※英数字混在の 10 桁程度のランダムな文字列)
```

以降のユーザー登録などの REST リクエストのクエリストリングに、この rest_id を付加することでユーザーを識別します。

- 認証に失敗すると、エラーコード 401(エラーメッセージ「Authentication failed.」)の HTTP レスポンスを返します。
- rest_id が誤っていると、エラーコード 401(エラーメッセージ「rest_id is incorrect.」)の HTTP レスポンスを返します。
- URL の形式が不正な場合は、エラーコード 400(エラーメッセージ「Bad Request.」)の HTTP レスポンスを返します。
- サーバ内部でシステムエラーが発生した場合は、エラーコード 500(エラーメッセージ「System error occurred.」)の HTTP レスポンスを返します。

3.2 ログアウト

以下の URL に GET メソッドでリクエストを送信することでログアウトする。

```
http://<UnicornIDM_Host>/unicornIDM/rest/logout?rest_id=FZ4FafLUod
```

このリクエストにより、正常にログアウトが完了すると次のレスポンスを返す。

```
Logout successful.
```

3.3 ユーザー情報検索

以下の URL をリクエストをすることで、ユーザー名が ichiro のユーザーの属性情報を取得します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/userinfo?target=example&username=ichiro&rest_id=m7bGI9ei16
```

取得した結果は以下のような CSV 形式となります。

```
username,sn,givenname,gidNumber,description,mail,profilePath,LdapGroups,AdGroups,GoogleGroup
ichiro,,"suzuki","ichiro","1000","","ichiro@osstech.co.jp","","",""
```

先頭の username はユーザー名を意味するカラムです。uid または cn のことです。username を除く各カラムはユーザー属性を表します。

ユーザーが存在しない場合は、エラーコード 404(エラーメッセージ「Not Found.」)の HTTP レスポンスを返します。

3.4 ユーザー登録

以下の URL をリクエストをすることで、ユーザー名が ichiro のユーザーを登録します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/useradd/?
rest_id=FZ4FafLUod&username=ichiro&password=CMnYOMPIub&sn=%E9%88%B4%E6%9C%A8&givenName=%E4%B8%80%E6%9C%97&target=example
```

必須属性はクエリストリングに必ず含める必要があります。任意属性は含めても含めなくてもどちらでも構いません。

「sn=%E9%88%B4%E6%9C%A8」は sn が「鈴木」、「givenName=%E4%B8%80%E6%9C%97」は givenName が「一郎」をそれぞれ URL エンコードした文字列です。

日本語のように URL として許容されていない文字列は、URL エンコードする必要があります。

パスワードの自動生成を行う場合は、「password=%23random%23」を指定することで、パスワードポリシーが適用されたランダムパスワードが生成されます。

※「%23」は URL エンコードされた「#」。

このリクエストにより、正常に登録が完了すると、組織ごとに次のレスポンスを返します。

```
'組織', '結果(OK)またはエラーメッセージ', '生成されたパスワード'
```

以下に、実行例を表します。

```
'example-ldap', 'OK', 'CMnYOMPIub'  
'example-ad', 'OK', 'CMnYOMPIub'  
'example-google', 'OK', 'CMnYOMPIub'
```

エラーがあった場合は、次のようなレスポンスを返します。

ユーザーが既に存在している場合などの特定の除き、LDAP や Google Apps がエラーメッセージを返した場合は、受け取ったエラー内容をそのままエラーメッセージに含めます。

```
'example-ldap', "object class 'posixAccount' requires attribute 'uidNumber'", 'CMnYOMPIub'  
'example-ad', 'OK', 'CMnYOMPIub'  
'example-google', 'DomainUserLimitExceeded(1200)', 'CMnYOMPIub'
```

3.5 ユーザー更新

以下の URL をリクエストをすることで、ユーザー名が ichiro のユーザーを更新します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/usermod/?  
rest_id=FZ4FafLUod&username=ichiro&password=CMnYOMPIub&sn=%E9%88%B4%E6%9C%A8&givenName=%E4%B8%80%E6%9C%  
%97&target=example
```

「sn=%E9%88%B4%E6%9C%A8」は sn が「鈴木」、「givenName=%E4%B8%80%E6%9C%97」は givenName が「一郎」をそれぞれ URL エンコードした文字列です。

日本語のように URL として許容されていない文字列は、URL エンコードする必要があります。

パスワードの自動生成を行う場合は、「password=%23random%23」を指定することで、パスワードポリシーが適用されたランダムパスワードが生成されます。

※ 「%23」は URL エンコードされた「#」。

結果はユーザー登録と同様の書式で返されます。

```
'example-ldap', 'No such user.', 'CMnYOMPIub'  
'example-ad', 'OK', 'CMnYOMPIub'  
'example-google', 'EntityDoesNotExist(1301)', 'CMnYOMPIub'
```

3.6 ユーザー削除

以下の URL をリクエストをすることで、ユーザー名が ichiro のユーザーを削除します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/userdel/?rest_id=FZ4FafLUod&username=ichiro&target=example
```

結果はユーザー登録と同様の書式で返されます。

```
'example-ldap', 'No such user.'  
'example-ad', 'OK'  
'example-google', 'EntityDoesNotExist(1301)'
```

3.7 ユーザー一覧検索

以下の URL をリクエストをすることで、ユーザー名が「ichiro」のユーザーの情報を取得します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/userinfo/?rest_id=FZ4FafLUod&username=ichiro&target=example
```

取得した結果は以下のような CSV 形式となります。

```
username,sn,givenName,description,mail,GoogleGroup,example-ldap,example-ad,example-google  
ichiro,"鈴木","一朗","","","group1","True","False",""
```

ファイルの最後にあるカラム(example-ldap,example-ad,example-google)は、ユーザーが登録されているサーバーにおけるユーザーの状態を表します。Trueの場合はログイン可、Falseの場合はログイン不可を意味します。TrueでもFalseでもない場合はユーザー自体が対象サーバー上に存在しません。上の CSV の場合、サーバーexample-ldap にログイン可能なユーザーが、サーバーexample-ad にログイン不可能なユーザーが存在し、サーバーexample-google にはユーザー自体が存在しないことを意味します。

3.8 ユーザー一括登録

以下の URL に対してパラメータをセットし、POST でリクエストをすることでユーザーを一括登録します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/useradd/?rest_id=FZ4FafLUod&target=example
```

POST するパラメータは以下のように GET メソッドでクエリストリングと同様です。

※「password=(値無し)」とすることでランダムなパスワードが設定されます。

```
&username=ichiro&password=&sn=suzuki&givenName=ichiro  
&username=hideki&password=matsui55&sn=matsui&givenName=hideki
```

結果は以下のような形式となります。

```
#yyyyMMdd-hhssmm (管理対象組織名)  
[summary]  
実行件数  
成功件数  
失敗件数  
[result]  
ユーザー名,OK(またはエラーメッセージ),パスワード
```

実際の実行例を表します。

```
#20111101-105345 example-ldap  
[summary]  
2 user entries are handled.  
Success to add:1 user entries  
Fail to add:1 user entries  
[result]  
ichiro,OK,nVeeEk  
hideki,object class 'posixAccount' requires attribute 'uidNumber',matsui55  
  
#20111101-105345 example-ad  
[summary]  
2 user entries are handled.  
Success to add:2 user entries  
Fail to add:0 user entries  
[result]  
ichiro,OK,nVeeEk  
hideki,OK,matsui55
```

3.9 ユーザー一括更新

以下の URL に対してパラメータをセットし、POST でリクエストをすることでユーザーを一括更新しま

す。

```
http://<UnicornIDM_Host>/unicornIDM/rest/usermod/?rest_id=FZ4FafLUod&target=example
```

POST するパラメータは以下のように GET メソッドでクエリストリングと同様です。

※ 「password=%23random%23」 とすることでランダムなパスワードが設定されます。

```
&username=ichiro&password=%23random%23&sn=suzuki&givenName=ichiro  
&username=hideki&password=matsui55&sn=matsui&givenName=hideki
```

結果は、ユーザー一括登録と同様に、次の形式となります。

```
#20111101-105401 example-ldap  
[summary]  
2 user entries are handled.  
Success to modify:0 user entries  
Fail to modify:2 user entries  
[result]  
ichiro,No such user.,8UNEY1  
hideki,No such user.,matsui55  
  
#20111101-105401 example-ad  
[summary]  
2 user entries are handled.  
Success to modify:2 user entries  
Fail to modify:0 user entries  
[result]  
ichiro,OK,8UNEY1  
hideki,OK,matsui55
```

3.10 ユーザー一括削除

以下のURLに対してパラメータをセットし、POSTでリクエストをすることでユーザーを一括削除します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/userdel/?rest_id=FZ4FafLUod&target=example
```

POSTするパラメータは以下のようにGETメソッドでクエリストリングと同様です。

```
&username=ichiro  
&username=hideki
```

結果は以下のような形式となる。

```
#20111101-105405 example-ldap
[summary]
2 user entries are handled.
Success to delete:0 user entries
Fail to delete:2 user entries
[result]
ichiro,No such user.,
hideki,No such user.,

#20111101-105405 example-ad
[summary]
2 user entries are handled.
Success to delete:2 user entries
Fail to delete:0 user entries
[result]
ichiro,OK,
hideki,OK,
```

3.11 グループ情報検索

以下の URL をリクエストをすることで、グループ名が `students` のグループの属性情報を取得します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupinfo?
target=example&groupname=students&rest_id=m7bGI9ei16
```

取得した結果は以下のような CSV 形式となります。

```
groupname,gidNumber,description,members,onlyldap
"students","1021","学生","['user01', 'user02']", "True"
```

先頭の `groupname` はグループ名を意味するカラムです。 `groupname` を除く各カラムはグループ属性を表します。

グループが存在しない場合は、エラーコード 404(エラーメッセージ「Not Found.」)の HTTP レスポンスを返します。

3.12 グループ登録

以下の URL をリクエストをすることで、グループ名が students のグループを登録します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupadd/?  
rest_id=m7bGI9ei16&target=example&groupname=students&gidNumber=1021&description=%8aw  
%90%b6&addMembers=[user01,user02]
```

必須属性はクエリストリングに必ず含める必要があります。任意属性は含めても含めなくてもどちらでも構いません。

description の「%8aw%90%b6」は「学生」を URL エンコードした文字列です。

日本語のように URL として許容されていない文字列は、URL エンコードする必要があります。

addMembers に対応する値に対して、[] の中にユーザー名をカンマ区切りで指定することで、グループにユーザーを追加できます。

このリクエストにより、正常に登録が完了すると、組織ごとに次のレスポンスを返します。

```
'組織', '結果(OK)またはエラーメッセージ', '生成されたパスワード'
```

以下に、実行例を表します。

```
'example-ldap', 'OK'  
'example-ad', 'OK'  
'example-google', 'OK'
```

エラーがあった場合は、次のようなレスポンスを返します。

グループが既に存在している場合などの特定の場合を除き、LDAP や Google Apps がエラーメッセージを返した場合は、受け取ったエラー内容をそのままエラーメッセージに含めます。

```
('example-ldap', 'Failed to add user to [{\'info\': \"object class \'posixGroup\' requires  
attribute \'gidNumber\'\", \'desc\': \'Object class violation\'}].')  
(\'example-ad', \"Failed to add user to [{\'info\': '0000207B: UpdErr: DSID-030511CF, problem 6002  
(OBJ_CLASS_VIOLATION), data 0\\n', \'desc\': 'Object class violation'}].\")  
(\'example-google', 'EntityExists(1300)')
```

3.13 グループ更新

以下の URL をリクエストをすることで、グループ名が students のグループを更新します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupmod/?
```

```
rest_id=FZ4FafLUod&groupname=students&password=CMnYOMPIub&sn=%E9%88%B4%E6%9C%A8&givenName=%E4%B8%80%E6%9C%97&target=example
```

description の「%8aw%90%b6」は「学生」を URL エンコードした文字列です。

日本語のように URL として許容されていない文字列は、URL エンコードする必要があります。

結果はグループ登録と同様の書式で返されます。

```
('example-ldap', 'No such group [students].')  
( 'example-ad', 'No such group [students].')  
( 'example-google', 'OK')
```

3.14 グループ削除

以下の URL をリクエストをすることで、グループ名が students のグループを削除します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupdel/?  
rest_id=FZ4FafLUod&groupname=students&target=example
```

結果はグループ登録と同様の書式で返されます。

```
('example-ldap', 'No such group [students].')  
( 'example-ad', 'No such group [students].')  
( 'example-google', 'OK')
```

3.15 グループ一覧検索

以下の URL をリクエストをすることで、グループ名が「students」のグループの情報を取得します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupinfo/?  
rest_id=FZ4FafLUod&groupname=students&target=example
```

取得した結果は以下のような CSV 形式となります。

```
groupname,description,gidNumber,example-ldap,example-ad,example-google  
"students","学生","1021","True","True"," "
```

ファイルの最後にあるカラム(example-ldap,example-ad,example-google)は、グループの登録状況を表します。Trueの場合は登録済み、空の場合は未登録を意味します。上のCSVの場合、サーバーexample-ldapとexample-ad にグループが存在し、サーバーexample-google にはグループが存在しないことを意味します。

3.16 グループ一括登録

以下の URL に対してパラメータをセットし、POST でリクエストをすることでグループを一括登録します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupadd/?rest_id=FZ4FafLUod&target=example
```

POST するパラメータは以下のように GET メソッドでクエリストリングと同様です。

```
&groupname=students&gidNumber=1271&description=%8aw%90%b6&addMembers=[user01,user02]  
&groupname=teachers&gidNumber=1272&description=%8b%b3%8et&addMembers=[user03,user04]
```

description の「%8aw%90%b6」、「%8b%b3%8et」はそれぞれ「学生」、「教師」を URL エンコードした文字列です。

日本語のように URL として許容されていない文字列は、URL エンコードする必要があります。

結果は以下のような形式となります。

```
#yyyyMMdd-hhssmm (管理対象組織名)  
[summary]  
実行件数  
成功件数  
失敗件数  
[result]  
グループ名,OK(またはエラーメッセージ),パスワード
```

実際の実行例を表します。

```
#20111101-105345 example-ldap  
[summary]  
2 group entries are handled.  
Success to add:2 group entries
```

```
Fail to add:0 group entries
[result]
students,OK
teachers,OK

#20111101-105345 example-ad
[summary]
2 group entries are handled.
Success to add:2 group entries
Fail to add:0 group entries
[result]
students,OK
teachers,OK
```

3.17 グループ一括更新

以下の URL に対してパラメータをセットし、POST でリクエストをすることでグループを一括更新します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupmod/?rest_id=FZ4FafLUod&target=example
```

POST するパラメータは以下のように GET メソッドでクエリストリングと同様です。

```
&groupname=students&gidNumber=1271&description=%8a%90%b6&addMembers=[user01,user02]
&groupname=teachers&gidNumber=1272&description=%8b%b3%8e&addMembers=[user03,user04]
```

description の「%8a%90%b6」、「%8b%b3%8e」はそれぞれ「学生」、「教師」を URL エンコードした文字列です。

日本語のように URL として許容されていない文字列は、URL エンコードする必要があります。

結果は、グループ一括登録と同様に、次の形式となります。

```
#20111101-105345 example-ldap
[summary]
2 group entries are handled.
Success to modify:2 group entries
Fail to modify:0 group entries
[result]
students,OK
```

```
teachers,OK

#20111101-105345 example-ad
[summary]
2 group entries are handled.
Success to modify:2 group entries
Fail to modify:0 group entries
[result]
students,OK
teachers,OK
```

3.18 グループ一括削除

以下のURLに対してパラメータをセットし、POSTでリクエストをすることでグループを一括削除します。

```
http://<UnicornIDM_Host>/unicornIDM/rest/groupdel/?rest_id=FZ4FafLUod&target=example
```

POSTするパラメータは以下のようにGETメソッドでクエリストリングと同様です。

```
&groupname=students
&groupname=teachers
```

結果は以下のような形式となる。

```
#20111101-105345 example-ldap
[summary]
2 group entries are handled.
Success to modify:2 group entries
Fail to modify:0 group entries
[result]
students,OK
teachers,OK

#20111101-105345 example-ad
[summary]
2 group entries are handled.
Success to modify:2 group entries
Fail to modify:0 group entries
[result]
students,OK
```

teachers,OK

4. 改版履歴

- 2012 年 02 月 28 日 田村 広平
 - 初版