

---

## NAME

IdPAccountManager::Tools - Command line client to the Test Account manager

## SYNOPSIS

```
./bin/account-manager-client.pl --list_authentication_tokens
```

## DESCRIPTION

The Test Account manager instantiates test accounts associated to a SAML Identity Provider. This script provides a command-line interface for most functions.

## EXAMPLES

```
account-manager-client.pl --add_test_account --account_profile=student1
--sp_entityid=https://test.federation.renater.fr/test/ressource
```

Adds a new test account.

```
account-manager-client.pl --list_test_accounts
--sp_entityid=https://test.federation.renater.fr/test/ressource
--account_profile=student1
```

List all test accounts. Criterias can be added to filter test accounts.

```
account-manager-client.pl --list_test_accounts --filter_expired
```

List all expired test accounts.

```
account-manager-client.pl --list_test_accounts --filter_expired --delete
```

Remove all expired test accounts from DB.

```
account-manager-client.pl --parse_federation_metadata
```

Parses the SAML metadata file, as defined by the  
federation\_metadata\_file\_path configuration parameter.

```
account-manager-client.pl --list_authentication_tokens
--sp_entityid=https://test.federation.renater.fr/test/ressource
--token=dhj67sjJ
```

List all authentication tokens. Criterias can be added to filter tokens.

```
account-manager-client.pl --list_authentication_tokens --filter_expired
```

List all expired authentication tokens.

```
account-manager-client.pl --list_authentication_tokens --filter_expired
--delete
```

Remove all expired authentication tokens from DB.

```
account-manager-client.pl --get_authentication_token --token=dhj67sjJ
```

Get informations on a token.

```
account-manager-client.pl --add_authentication_token
--email_address=john@my.fqdn
--sp_entityid=https://test.federation.renater.fr/test/ressource
```

Adds a new test account.

```
account-manager-client.pl --send_notice --email_address=john@my.fqdn
```

Sends a mail notice to the specified email address.

```
account-manager-client.pl --add_service_provider
--sp_entityid='https://test.federation.renater.fr/test/ressource
--displayname='Test SP' --contacts=email1@dom,email2@dom
```

---

Adds a new Service provider

**AUTHOR**

Olivier Salaun (olivier.salaun@renater.fr)

---

## NAME

IdPAccountManager::TestAccount - Manage test user accounts for the Test Identity Provider

## SYNOPSIS

```
my $test_account = new IdPAccountManager::TestAccount(account_profile =>
'student1',
                                                    sp_entityid =>
'https://test.federation.renater.fr/test/ressource');
unless (defined $test_account) {
    die "Failed to create test account";
}

unless ($test_account->save()) {
    die "Failed to create test account";
}

printf "Account created:\n\tuserid: user%d\n\tpassword: %s\n",
$test_account->get('id'), $test_account->get('user_password');
```

## DESCRIPTION

The Test Account manager instanciates test accounts associated to a SAML Identity Provider. This module allows to manage the test accounts.

## SUBROUTINES/METHODS

new ARGS

Class method. Create a new IdPAccountManager::TestAccount object. Example:

```
my $test_account = new
IdPAccountManager::TestAccount(account_profile => 'student1',

sp_entityid =>
'https://test.federation.renater.fr/test/ressource');
```

Supported arguments include:

account\_profile

ID of the account profile to be used.

sp\_entityid

EntityID (SAML ID) of the Service Provider associated to the test account.

create\_test\_accounts\_for\_sp ARGS

Class method. Create test accounts for supported account profiles.

Supported arguments include:

sp\_entityid

EntityID (SAML ID) of the Service Provider associated to the test account.

delete

Deletes the test account in the database.

get ATTR\_NAME

Returns the value of the specified ATTR\_NAME attribute of the test account.

---

`list_test_accounts` ARGS

Class method. List all test accounts in database.

Supported arguments include:

`sp_entityid`

Entityid of a SAML Service Provider to list only test accounts linked to this Service Provider.

`account_profile`

Test account profile to list only test accounts linked based on this profile.

`print` FD

Dumps the content of the test account to the specified FD file handler (default to STDOUT)

`save`

Save the test account in the database.

## **AUTHOR**

Olivier Salaun (olivier.salaun@renater.fr)

---

## NAME

SAMLMetadata - loading SAML federation metadata

## SYNOPSIS

```
my $federation_metadata = new IdPAccountManager::SAMLMetadata;
unless ($federation_metadata->load(federation_metadata_file_path =>
'/tmp/edugain-saml-metadata.xml')) {
    die;
}

my %args;
if ($options{'sp_entityid'}) {
    $args{'filter_entity_id'} = $options{'sp_entityid'};
}

unless ($federation_metadata->parse(sp_entityid =>
'https://test.federation.renater.fr/test/ressource')) {
    die;
}

## List SAML entities
printf "Hashref representing the metadata:\n";

&IdPAccountManager::Tools::dump_var($federation_metadata->{'federation_meta
data_as_hashref'}, 0, \*STDOUT);
```

## DESCRIPTION

The Test Account manager instantiates test accounts associated to a SAML Identity Provider. This module parses a SAML2 metadata file.

## SUBROUTINES/METHODS

new ARGS

Class method. Create a new IdPAccountManager::SAMLMetadata object. Example:

```
my $federation_metadata = new
IdPAccountManager::SAMLMetadata;
```

load ARGS

Loads the SAML metadata file.

Supported arguments include:

federation\_metadata\_file\_path

Path of the SAML metadata file.

parse ARGS

Parse the SAML metadata file.

Supported arguments include:

filter\_entity\_id

EntityID of SAML entities to filter.

print FD

Dumps the content of the SAML metadata to the specified FD file handler (default to STDOUT)

---

**AUTHOR**

Olivier Salaun (olivier.salaun@renater.fr)

---

## NAME

IdPAccountManager::Tools - Set of subroutines usefull for the Test Account manager

## DESCRIPTION

The Test Account manager instanciates test accounts associated to a SAML Identity Provider. This module gathers a set of usefull subroutines.

## SUBROUTINES/METHODS

`dump_var ($var, $level, $fd)`

Dumps a complex perl data structure. \$var is a reference to the variable to dump. \$level should be set to 0 (subroutine called recursively). \$fd is the file descriptor for the output (default is STDOUT).

`do_log ($level, $message)`

Write \$message to the log file. \$level sets the log level (debug, info, trace, notice, error).

`encode_utf8 ($string)`

Return a UTF8 encoded version of \$string.

`escape_xml ($string)`

Escape XML chars in \$string.

`generate_password`

Returns a random password following some security guidelines.

`mail_notice (ARGS)`

Send a mail notice.

Supported arguments include:

`template`

TT2 mail template to parse.

`to`

Destination email address.

`data`

A hashref with parameters used to parse the mail template.

`qencode ($string)`

Retunrs a Q-encoded version of \$string.

`sha256_hash ($string)`

Returns a SHA256 hash for \$string.

`update_ssp_authsources`

Update simpleSAMLphp authsources.php configuration file with the currently valid test accounts.

## AUTHOR

Olivier Salaun (olivier.salaun@renater.fr)

---

**NAME**

IdPAccountManager::ServiceProvider - Manage Service Providers for which test accounts have been requested for the Test Identity Provider

**SYNOPSIS****DESCRIPTION****SUBROUTINES/METHODS****AUTHOR**

Olivier Salaun (olivier.salaun@renater.fr)



---

## NAME

IdPAccountManager::AuthenticationToken - Manage Authentication tokens used to validate test account creation requests

## SYNOPSIS

```
my $authentication_token = new
IdPAccountManager::AuthenticationToken(token => 'sdfkl4fslkj44');

unless ($authentication_token->load()) {
    die "No corresponding token found in DB\n";
}

$authentication_token->print();
```

## DESCRIPTION

The Test Account manager instantiates test accounts associated to a SAML Identity Provider. This module allows to manage authentication tokens to validate requestor identity.

## SUBROUTINES/METHODS

new ARGS

Class method. Create a new IdPAccountManager::AuthenticationToken object.

Example:

```
my $authentication_token = new
IdPAccountManager::AuthenticationToken(token =>
'sdfkl4fslkj44');
```

Supported arguments include:

token

ID of the token.

sp\_entityid

EntityID (SAML ID) of the Service Provider associated to the authentication token.

email\_address

Email address of the user associated to the authentication token.

delete

Deletes the token in the database.

get ATTR\_NAME

Returns the value of the specified ATTR\_NAME attribute of the token.

list\_authentication\_tokens ARGS

Class method. List all tokens in database.

Supported arguments include:

sp\_entityid

Entityid of a SAML Service Provider to list only tokens linked to this Service Provider.

token

ID of the tokens to list only those tokens.

---

load

Loads the token from the database.

print FD

Dumps the content of the authentication token to the specified FD file handler (default to STDOUT)

set ARGS

Sets token attributes in ARGS.

save

Save the token in the database.

## **AUTHOR**

Olivier Salaun (olivier.salaun@renater.fr)

---

## NAME

Conf - Configuration of the Test Account manager

## SYNOPSIS

```
use Conf;  
printf "Version: %s\n", $Conf::global{'version'};
```

## DESCRIPTION

The Test Account manager instanciates test accounts associated to a SAML Identity Provider. This module is the main configuration file for the software.

## CONFIGURATION OPTIONS

See the code for the list of supported configuration options

## AUTHOR

Olivier Salaun (olivier.salaun@renater.fr)

---

**NAME**

Conf - Configuration of the eduGAIN Access Check

**SYNOPSIS**

```
use Conf;  
printf "Version: %s\n", $Conf::global{'version'};
```

**DESCRIPTION**

The Test Account manager instanciates test accounts associated to a SAML Identity Provider. This module is the main configuration file for the software.

**CONFIGURATION OPTIONS**

See the code for the list of supported configuration options

**AUTHOR**

Olivier Salaun (olivier.salaun@renater.fr)

