Guide to Transfer and Restore API Functions - Check the Transfer Session

Introduction

A transfer session check ensures that you can create a system account on the remote server.

- For `root` sessions, analyze the transfer session and validate whether you can transfer a username to the remote server.
- For user transfer sessions, validate whether you can transfer a username to the remote server.

> ⚠️ **Note:**
>
> In this example, the following statements are true:
>
> - `remote.example.com` and `local.example.com` represent the remote server and the local server.
> - You logged in to the local server with the `cpsess##########` security token.
> - The `example` account owns `example.com`, and you want to transfer it from the remote server to the local server.
> - The `example.com` domain contains `123,456,789` bytes.
> - You want to transfer the `package1` package from the remote server to the local server.
> - `luggage12345` represents the `root` password for the remote server.

Analyze the transfer session

To confirm that the transfer session contains all of the items that you want to transfer, analyze the transfer session. To do this, use the WHM API 1 `retrieve_transfer_session_remote_analysis` function.

Select the tab below that corresponds to the method by which you wish to call the function:

```
https://hostname.example.com:2087/cpsess##########/json-api
/retrieve_transfer_session_remote_analysis?api.
version=1&transfer_session_id=exampleservercopya20140206192428
NtyW
```

The function call returns the following list:

- Transfer session settings.
- Local and remote groups and users.
- Databases and database users.
- Packages and accounts (transfer items) that the transfer session contains.

The function only returns metadata, which resembles the following example:

```
{   "data": {
      "options": {
        "skip_reseller_privs": null,
        "unrestricted": "0"
      },
      "transfer_session_id":
"vm5docscpanelnoroo201402251939519hmy",
      "config": {
        "shared_mysql_server": null
      },
      "local": {
        "groups": {
          "never": 1,
          "gonna": 1,
          "give": 1,
          "you": 1,
          "up": 1
        },
        "users": {
          "never": 1,
          "gonna": 1,
          "let": 1,
          "you": 1,
          "down": 1
        },
        "dbs": null,
        "available_ips": [

        ]
      },
      "remote": {
        "version": 0,
        "resellers": {

        },
        "has_disk_used": 0,
        "dbs": null,
        "host": null,
        "has_xfertool": 0,
        "has_owners": 0,
        "server_type": null
      },
      "items": {
        "packages": [

        ],
        "accounts": [

        ]
      }
    },
    "metadata": {
      "version": 1,
      "reason": "OK",
      "result": "1",
      "command": "retrieve_transfer_session_remote_analysis"
    }
}
```

The `items` and `dbs` containers are empty. These containers will hold accounts, packages, databases, and database users.

Validate the username

To ensure that the username is not currently in use, validate whether you can transfer that username to the remote server. To do this, use the WHM API 1 `validate_system_user` function.

Select the tab below that corresponds to the method by which you wish to call the function:

```
https://hostname.example.com:2087/cpsess##########/json-api
/validate_system_user?api.version=1&user=username
```

The output contains the following Boolean values which you can use to check the username:

```
{
  "data": {
    "valid_for_transfer": 1,
    "reserved": 1,
    "exists": 1,
    "valid_for_new": 1
  },
  "metadata": {
    "version": 1,
    "reason": "OK",
    "result": "1",
    "command": "validate_system_user"
  }
}
```