



# **PLESK 7.5 RELOADED**

## **BACKUP/RESTORE UTILITIES MANUAL**

---

Distribution of this work or derivative of this work in any form is prohibited unless prior written permission is obtained from the copyright holder.

Plesk™ and Plesk 7 Reloaded™ are registered trademarks of SWsoft, Inc. All other trademarks and copyrights are the property of their respective owners.

13755 Sunrise Valley Drive, Suite 325, Herndon, VA, 20171 USA, Ph.: +1 (703) 815-5670, Fax.: +1 (703) 815-5675

---

---

## Table of Contents

1. Introduction .....	1
2. Backing up the configuration information and the data .....	3
Redirecting the output .....	3
Compression options .....	4
Selective backup .....	4
Other options .....	6
3. Restoring the backed up configuration information and data .....	9
Redirecting the input .....	9
Mapping .....	10
Selective restore .....	12
Setting the virtual hosts directory .....	13
Restoring the certificate .....	14
Other options .....	14
Notes on restoring at a non-clean Plesk installation .....	15
Restoring users' backups .....	17
4. Distributed backup and restore .....	19
Backing up from the remote computer .....	19
Restoring from the remote computer .....	19
Simultaneous backup/restore .....	20
5. Configuration file examples .....	21
Selective backup files .....	21
Mapping files .....	21



---

# Chapter 1. Introduction

The Plesk™ backup/restore utilities allow backing up the Plesk server configuration information as well as its current content (client accounts data, domains and so forth) as they are at a certain moment in time - the moment when the backup is carried out. They also allow restoring these data at the server. This proves useful when either you need to execute a system operation, which can influence such data (for instance, upgrade the OS that Plesk is running on, or upgrade Plesk itself), or simply for the safety purposes. The two utilities that provide these functions are:

**psadump** - this utility is used for backing up the configuration information and the data by storing it in the dump-file

**psarestore** - this utility is used for restoring the backed up configuration information and data at the server from the dump-file

Executing **psadump** with various options and keys generates the so-called *dump-file* – the file that contains all of the backed-up data. The dump-file is a text file, which is a MIME document. The first segment of this file is the description of the data structure in Plesk presented in the form of an XML document. The other segments contain data of various types: tar archive, mysqldump, file, etc. – all in UUENCODE. These latter segments contain the actual server content data – the descriptions and settings for client accounts and their domains, the user databases contents and so on. This file can then be used in order to restore all of the information it contains at the Plesk server. Note that event handlers are not backed up. Chapter Backing up the configuration information and the data presents the detailed instructions for executing the **psadump** utility, describes the available options and keys and their usage.

In order to restore the backed-up information at the Plesk server the **psarestore** utility is used. It is called with one required parameter – the name of the dump-file; there are also a number of options and keys that allow certain flexibility in the process of restoring the data. After the restore process is successfully completed, the Plesk server is configured the same way as the one, at which the back up was carried out, and contains all or some of the data, that was backed up. Chapter Restoring the backed up configuration information and data presents the detailed instructions for executing the **psarestore** utility, describes the available options and keys and their usage.



---

## Chapter 2. Backing up the configuration information and the data

```
psadump -f <dumpfile> [-z] [--no-internal-zip]
[--clients-logins=<client-logins-file> |
--clients-ids=<client-ids-file> |
--domains-names=<domain-names-file> |
--domains-ids=<domain-ids-file>] [--nostop] [--force]
[--tar-ignore-failed-read]

psadump [--version]

psadump [-h]
```

### Redirecting the output

The `-f` argument serves for specifying the file, to which the output of the `psadump` utility will be directed. The name of this file is specified by the `<dumpfile>` parameter.

#### IMPORTANT

The `-f` argument is a required argument.

There are four options available:

- `<dumpfile> = /fullpath/filename` - you can specify the exact name of the file, to which the output will be directed. In other words, you specify the full name of the future dump-file

For example, execute `psadump -f /root/psa_backup/my_psa_backup_file` to back up the information and store it in the file `/root/psa_backup/my_psa_backup_file`

- `<dumpfile> = /fullpath/` - you can specify only the name of the directory, at which the dump-file will be created. In this case, the default name will be used for the dump-file, which will locate in the specified directory

For example, execute `psadump -f /root/psa_backup/` to back up the information and store it in the directory `/root/psa_backup/` under the default name

**i NOTE**

The default dump-file name is generated automatically at the moment of executing `psadump`.

- `<dumpfile> = - -` you can direct the output to `stdout`. This can be very useful if you are executing the distributed backup/restore (backing up Plesk at one machine and restoring it at another machine) or if you wish to split the dump-file into chunks (piping the output to `split`), which can be used as a workaround for 2-GB limit on certain file systems.

For example, execute `psadump -f - | split -b1000m - dump.` to have the backed up information stored in files named `dump.aa`, `dump.ab`, `dump.ac`, etc., each 1000MB in size, except maybe for the last one, which will be less than or equal in size to the limit value.

For examples and instructions on distributed backup/restore see chapter Distributed backup and restore.

- To create a backup file on the FTP server, use the command `./psadump -f ftp://<login>:<password>@<server>/<dump_file_path>`

## Compression options

The `psadump` utility allows various options of compressing the resulting dump-file:

- Specifying the `-z` argument makes `psadump` compress the dump-file after it was generated with `gzip`.
- Specifying the `--no-internal-zip` argument tells `psadump` not to compress the separate segments of the dump-file, which they are by default.

## Selective backup

You can choose to back up only the data associated with certain clients, or certain domains. The `psadump` utility allows you to supply the list of clients or domains for executing the selective backup of data.

### Backing up selected client accounts

There are two ways of specifying the clients, whose account data you wish to have backed up. You may either specify the client logins or the identifiers assigned to the clients.



- To specify the client accounts by their login use the `--clients-logins` argument. The corresponding file of client logins must contain the logins of the clients, whose account data you wish to back up, one login per line, no other separators.

For example, execute `psadump -f /root/psa_backup/my_psa_backup_file --clients-logins=/root/psa_backup/my_client_logins_file` to execute backing up the data belonging to the clients listed by their logins in the file `/root/psa_backup/my_client_logins_file`. See chapter Configuration file examples section Client logins file for an example of the client logins file.

- To specify the client accounts by the client identifiers use the `--clients-ids` argument. The corresponding file of client identifiers must contain the identifiers of the clients, whose account data you wish to back up, one identifier per line, no other separators.

### NOTE

The client identifiers are the Plesk-related unique identifiers assigned to all clients. They can be found in the Plesk server database, table `clients` column `id`.

For example, execute `psadump -f /root/psa_backup/my_psa_backup_file --clients-ids=/root/psa_backup/my_client_ids_file` to execute backing up the data belonging to the clients listed by their identifiers in the file `/root/psa_backup/my_client_ids_file`. See chapter Configuration file examples section Client id's file for an example of the client identifiers file.

## Backing up selected domains

There are two ways of specifying the domains, whose content you wish to have backed up. You may either specify the domain names or the identifiers assigned to the domains.

- To specify the domains by their name use the `--domains-names` argument. The corresponding file of domain names must contain the names of the domains, whose content you wish to back up, one domain name per line, no other separators.

For example, execute `psadump -f /root/psa_backup/my_psa_backup_file --domains-names=/root/psa_backup/my_domain_names_file` to execute backing up the content of the domains listed by their names in the

file `/root/psa_backup/my_domain_names_file`. See chapter Configuration file examples section Domain names file for an example of the domain names file.

- To specify the domains by the domain identifiers use the `--domains-ids` argument. The corresponding file of domain identifiers must contain the identifiers of the domains, whose content you wish to back up, one identifier per line, no other separators.

### NOTE

The domain identifiers are the Plesk-related unique identifiers assigned to all domains. They can be found in the Plesk server database, table `domains` column `id`.

For example, execute `psadump -f /root/psa_backup/my_psa_backup_file --domains-ids=/root/psa_backup/my_domain_ids_file` to execute backing up the content of the domains listed by their identifiers in the file `/root/psa_backup/my_domain_ids_file`. See chapter Configuration file examples section Domain id's file for an example of the domain identifiers file.

## Other options

- Use the `--force` argument to overwrite the existing dump-file without prompting for confirmation.
- Use the `--tar-ignore-failed-read` argument to make `tar` not exit with non-zero status on unreadable files (see argument `--ignore-failed-read` for `tar`).
- If you do not want to stop all of the Plesk services when executing backup, include the `--nostop` argument.

 **IMPORTANT**

If during the process of backing up with this option some changes are made through the Plesk control panel the resulting dump may be generated incorrectly or the process itself may fail.

You are strongly advised to stop the admin Apache, so as to prevent the clients changing data through the control panel in the process of executing a backup.

For the RPM version execute: `service psa stop1`

For the standard version execute: `/usr/local/psa/rc.d/psa stop1`

- In order to prevent certain user data from being changed during the backup process when executing `psadump` with option `--nostop`, each domain is deactivated before backing up its data, and then reactivated again. If you wish to back up data without deactivating domains, use option `--nostop-domain` along with `--nostop`. Just note that in this case there is a possibility that some data can be modified through the control panel in the course of backup.
- The `--do-not-dump-logs` option allows to not back up the virtual host log files.
- To view the version of the Backup/Restore Utilities, execute `psadump` with the `--version` argument.
- Executing `psadump -h` will display a brief help to remind you of the available options.



---

# Chapter 3. Restoring the backed up configuration information and data

```
psarestore -f <dumpfile> [-L] [-t] -m <ip_map_file> -s  
<shells_map_file> [-c <clients_map_file>]  
[--clients-logins=<client-logins-file> |  
--domains-names=<domain-names-file>] [-V {conf | dump} |  
--vhosts-dir={conf | dump}] [--restore-bu-cert] [--force]  
  
psarestore --version  
  
psarestore -h
```

## Redirecting the input

The argument `-f` serves for specifying the dump-file. The `psarestore` utility uses this file to read the configuration information and server content data when restoring the Plesk server.

### IMPORTANT

The `-f` argument is a required argument.

There are three options available:

- `<dumpfile> = /fullpath/filename` - you can specify the exact name of the dump-file

For example, execute `psarestore -f /root/psa_backup/my_psa_backup_file` to restore the Plesk server based on the configuration and content information stored in the file `/root/psa_backup/my_psa_backup_file`

- `<dumpfile> = -` - you can specify stdin as the dump-file. This can be very useful if you are executing the distributed backup/restore (backing up Plesk at one machine and restoring it at another machine) or if you are restoring from the chunked dump-file.

For example, execute `cat dump.* | ./psarestore -f -` to direct the listing of the files of format `dump.*` as the input for `psarestore`.

For examples and instructions on distributed backup/restore see chapter Distributed backup and restore.

- To restore a backup file stored on an FTP server, use the command  
`./psarestore -f ftp://<login>:<password>@<server>/<dump_file_path>`

## Mapping

Usually, there can be found no two identical servers for restoring Plesk. The mapping mechanism serves for associating various entities between the backed up server and the restored server. The example of this is mapping the IP addresses: the set of available IP addresses at the restored server may differ from the like set at the backed up server, some of the IP addresses could be used on domains at the backed up server. In order to have such domains restored properly and functional, one needs to 'map' the corresponding IP addresses so that when Plesk is restored, the appropriate IP addresses are substituted in place of the ones, used at the backed up server. The `psarestore` utility allows mapping of the IP addresses, shells and client accounts.

### IMPORTANT

The `-m` and `-s` arguments, responsible for mapping the IP addresses and shells, are required arguments. Note that the further examples may not contain some of them, this is only done so as not to overload the perception of the important information with the details.

## Mapping IP addresses

The most important aspects of mapping IP addresses are:

- The type and belonging to a specific client's IP pool are significant. This must be taken into account when mapping IP addresses
- Each separate IP address should only be mapped to a separate IP address
- The types of the source IP address and destination IP address must coincide if the new IP address already exists in the system
- The accounts of clients - owners of the source IP address and destination IP address - must coincide if the new IP address belongs to some client's IP pool

To have `psarestore` map the IP addresses, use the `-m` argument. The corresponding file of mapped IP addresses must contain the set of strings that describe how to map the IP addresses.

 **IMPORTANT**

If a client, whose account was backed up and is being restored, has domains using certain IP addresses, these addresses are required to be mapped. Otherwise Plesk will not be restored.

For example, execute `psarestore -f /root/psa_backup/my_psa_backup_file -m /root/psa_backup/my_ip_map_file` to restore the Plesk server mapping the IP addresses according to the instructions contained in the file `/root/psa_backup/my_ip_map_file`. See chapter Configuration file examples section IP mapping file for an example of the IP addresses mapping file.

 **IMPORTANT**

When restoring at a non-clean Plesk installation there may appear the IP address collisions if an already used IP address appears in the IP address mapping file. Such situations can be resolved by either removing the troublemaking strings from the IP address mapping file or by modifying the conflicting IP address.

## Mapping shells

To have `psarestore` map the shells, use the `-s` argument. The corresponding file of mapped shells must contain the set of strings that describe how to map the shells.

For example, execute `psarestore -f /root/psa_backup/my_psa_backup_file -s /root/psa_backup/my_shells_map_file` to restore the Plesk server mapping the shells according to the instructions contained in the file `/root/psa_backup/my_ip_map_file`. See chapter Configuration file examples section Shell mapping file for an example of the shells mapping file.

## Mapping client accounts

This may come useful when restoring at a non-clean Plesk installation. It may happen so that the client accounts you are trying to restore already “exist” at the Plesk server. Such client collision means that a client with such login name or personal name already exists. Situations like this can be resolved either by mapping client accounts or by ignoring the colliding client accounts.

- To have `psarestore` map the client accounts, use the `-c` argument. The corresponding file of mapped clients must contain the set of strings that

describe how to map the client accounts.

For example, execute `psarestore -f /root/psa_backup/my_psa_backup_file -c /root/psa_backup/my_clients_map_file` to restore the Plesk server mapping the client accounts according to the instructions contained in the file `/root/psa_backup/my_clients_map_file`. See chapter Configuration file examples section Client mapping file for an example of the client accounts mapping file.

- Should none of the arguments be supplied and collisions encountered, `psarestore` will warn you of the fact and exit, generating a default client account mapping file named `clients.rej`. This file will contain the list of clients you need to map to avoid collisions. What you will need to do is to edit this file, providing where necessary new login names and/or personal names. After that you will have to try executing `psarestore` again specifying this file as the client account mapping file.

If a client account in the dump file and a client account that already exists at the server, where the restore is executed, have the same login name and personal name, the two such accounts are considered the same and the data are merged.

### **IMPORTANT**

If the limits of the latter account cannot be observed at the restore, the restore fails. In this case, you may want to restore this account separately, by mapping it.

If only the login name or only the personal name is the same, this situation is considered a collision and the restore will not be performed.

## Selective restore

You can choose to restore only the data associated with certain clients, or certain domains. The `psarestore` utility allows you to supply the list of clients or domains for executing the selective restore of data.

### Restoring selected client accounts

To specify the client accounts you wish to restore use the `--clients-logins` argument. The corresponding file of client logins must contain the logins of the clients, whose account data you wish to restore, one login per line, no other separators.



For example, execute `psarestore -f /root/psa_backup/my_psa_backup_file --clients-logins=/root/psa_backup/my_client_logins_file` to restore only the data belonging to the client accounts listed in the file `/root/psa_backup/my_client_logins_file`. See chapter Configuration file examples section Client logins file for an example of the client logins file.

## Restoring selected domains

To specify the domains you wish to restore use the `--domains-names` argument. The corresponding file of domain names must contain the logins of the domains that you wish to restore, one domain name per line, no other separators.

For example, execute `psarestore -f /root/psa_backup/my_psa_backup_file --domains-names=/root/psa_backup/my_domain_names_file` to restore only the content of the domains listed in the file `/root/psa_backup/my_domain_names_file`. See chapter Configuration file examples section Domain names file for an example of the domain names file.

### NOTE

During the domains restoring process the respective client accounts, these domains are related to, will be also restored.

## Setting the virtual hosts directory

The `psarestore` utility allows specifying the location for placing the restored virtual hosts directory at the moment of performing the restore operation. You can choose to either use the location specified in the `psa.conf` of the server, at which the restore is being performed, or the location used at the backed up server.

- To restore the virtual hosts directory at the location specified in the `psa.conf` file of the server, at which the restore is being performed, use the `-V conf` (or `--vhosts-dir=conf`) argument.
- To restore the virtual hosts directory at the location used at the backed up server use the `-V dump` (or `--vhosts-dir=dump`) argument.

**! IMPORTANT**

If the virtual hosts directory locations are not the same, the virtual hosts directory of the server, at which the restore is performed, will be relocated according to the information stored in the dump-file.

If you specify neither of these options the psarestore utility will request whether you agree to place the virtual hosts directory in the location specified in the dump-file.

**i NOTE**

When restoring at a non-clean Plesk installation it is *impossible* to restore the virtual hosts directory at a location other than the one specified in the `psa.conf` file of the server, at which the restore is being performed. All the virtual hosts will be restored in the virtual hosts directory of that server.

## Restoring the certificate

The backup/restore utilities are installed along with a special certificate, which is then used to sign the data backed up through the Plesk control panel by various users. This is done in order to ensure the security of the saved data.

**i NOTE**

See the *Plesk 7.5 Reloaded User's Manual* chapter *Managing Domains* for details on performing backup and restore of data through the Plesk control panel.

In order to enable users restore their data through the control panel at the restored server from the dumps they made before the full system backup, you should retrieve the previous certificate from the system dump. This can be done with the help of the `--restore-bu-cert` argument.

**! IMPORTANT**

If this action is not taken, the user data saved and signed by the previous certificate will be considered invalid and will not be restored.

## Other options

- Use the `--force` argument to use default options when requested by the

utility.

- To run `psarestore` in the test mode use the `-t` argument. The actual restore will not be performed but should any errors or collisions appear, you will be notified of them. This is a very useful means of setting up the restore process. It allows you to make necessary modifications in the mapping files before committing the restore.
- If you are restoring accounts of clients who used a specific language that is not currently supported by your Plesk (the respective language pack is not installed), employing the `-L <language_code>` option will allow to restore such client accounts associating it with any other supported language that you specify. To view a list of language packs currently installed in your Plesk, you may use the `-L list` option. The language code is a four letter notation that is comprised of hyphenated two-letter lower-case language code (ISO 639) and two-letter upper-case country code (ISO 3166), for example, `en-US`.
- To view the version of the Backup/Restore Utilities, execute `psarestore` with the `--version` argument.
- Executing `psarestore` with `-h` will display a brief help to remind you of the available options.

## Notes on restoring at a non-clean Plesk installation

The *non-clean Plesk installation* is the situation when the restore process is performed at the Plesk server, which is already in use, i.e. it has been configured and there are client accounts, domains, etc. created and functioning. This situation is very delicate and therefore requires a special mentioning.

### **IMPORTANT**

Two consequent `psarestore` runs even with the very same dump-file are always considered two completely independent restore attempts with completely different backup information.

## Server settings

The server settings and the Administrator information of the non-clean Plesk server are not changed in the process of restore.

## Possible collisions

All the restored client accounts and domains are considered different from

those at the non-clean Plesk server a priori. This can result in encountering collisions.

- Collisions in client accounts - these can be resolved in any of the two ways: by either mapping the colliding client accounts to other accounts or by ignoring the colliding client accounts. See section Mapping client accounts for detailed instructions.
- Collisions in IP addresses - these can also be resolved by adjusting the IP address mapping. See section Mapping IP addresses for detailed instructions.
- Collisions in domains, hosting accounts, web users, etc. - should such a collision occur, the corresponding object and all of the dependent objects will not be restored. For example, if there is a collision when restoring a domain, the domain itself as well as all of its databases, web users, protected directories, hosting accounts, etc. will not be restored.

### **IMPORTANT**

Use the argument `-t` to run `psarestore` in the test mode to make sure the backed up information can be restored completely and without any complications.

## Selective restore

The situations of restoring selected client accounts and domains are handled absolutely the same as they are for the clean Plesk installation.

### **IMPORTANT**

It is impossible to restore selected domains for the client, whose account already exists at the server. You may have to either restore such domains under a different client using mapping mechanism or remove the existing client account and restore it from the dump, in which case, all the existing client's data will be lost.

## psa.conf settings

No settings in `psa.conf` file can be changed in the process of restoring at a non-clean installation. This file is used in the 'read only' mode. This fact especially influences the process of restoring the virtual hosts directories. In this situation the restored virtual hosts are always placed at the location used by the server, at which the restore is being performed (the one specified in `psa.conf`).

## Restoring users' backups

Plesk allows clients and domain administrators to individually back up particular domain's data from the control panel. In order to ensure security of the system such dump-files are digitally signed. This prevents illegal modification of the dump-file contents and thus protects the system from possible misuses.

### **IMPORTANT**

The digitally signed user backups can be restored through the control panel only at the same server. This is due to the fact that a key, which is used for signing, is generated when the backup/restore utilities are installed on the server. User dump-files can be restored on different servers only by administrator using the command line tools.

See the *Plesk 7.5 Reloaded User's Manual* for instructions on creating and restoring domain data from the control panel.

## Restoring users' dump-files from the command line

The administrator can restore any dump-file at any server.

### **IMPORTANT**

When a dump-file is restored by the admin the digital signature is not verified. Restoring "insecure" dump-files can cause system security breaches.

The following options are to be used with the `psarestore` utility in order to restore a user's backup:

- `--single-domain-mode` - this option specifies that a user backup (as opposed to regular server backup) is being restored, user's dump-file contains data of a single domain
- `--domain-name` - the name of the domain where the data from the dump-file will be restored

### **NOTE**

Presently the name of the domain stored in the dump-file and the value of this parameter must be the same.

- `--domain-ip` - the IP address at which the domain from the dump-file will be restored

 **IMPORTANT**

The IP address must be available in the client IP pool

- `--client-login` - the login of the client for whom the domain from the client dump-file will be restored

 **NOTE**

If a domain with the same name already exists, either the data will be merged or the dump-file will not be restored due to some irresolvable collisions

For example, execute `psarestore -f /root/psa_backup/client_backup --single-domain-mode --domain-name=client.domain.com --domain-ip=1.2.3.4 --client-login=client` to restore a domain named 'client.domain.com' from the client dump-file `/root/psa_backup/client_backup` for the client with login name 'client' at the IP address '1.2.3.4'.

---

# Chapter 4. Distributed backup and restore

The distributed backup and restore are the option provided by the `psadump` and `psarestore` utilities. These utilities can be used in a fashion, which allows to back up a Plesk server creating the dump-file on the disk of a remote computer as well as to restore a Plesk server using the dump-file stored at the remote computer.

## Backing up from the remote computer

The `psadump` utility can be used the following way to for backing up the remote Plesk server information storing the dump-file at the local machine. This can be done, for instance, to provide the regular server backup.

```
ssh root@remotehost '/usr/local/psa/bin/psadump -f -  
--nostop' | split -b1000m - dump.
```

### NOTE

Please use caution when executing a backup with the `--nostop` parameter. See the details of using `--nostop` in order to avoid possible complications.

The backup of Plesk can take up a considerable amount of disk space at the server. Using this technique you are both: saving the disk space at the server and placing the dump-file at a different location, which enhances the security of the backed up information.

## Restoring from the remote computer

The `psarestore` utility much similar to the `psadump` utility allows executing remote restore operation. In this case the remote dump-file is used to restore Plesk at the local server.

```
ssh root@remotehost 'cat /root/psa_backup/dump.*' |  
/usr/local/psa/bin/psarestore -f - -m my_ip_map_file -s  
my_shells_map_file
```

This, like the above technique, is useful when you are unable or choose not to store the dump-file at the same machine with Plesk.

## Simultaneous backup/restore

Combining the two previously described techniques allows to execute the simultaneous backup/restore of a Plesk server. This could be very useful when you need to work (make an upgrade or a similar operation) on your machine that is currently employed under Plesk but you wish to decrease the idle time as much as possible. In such case you can simply enough migrate Plesk from one machine to another.

```
ssh root@remotehost '/usr/local/psa/bin/psadump -f -  
--nostop' | /usr/local/psa/bin/psarestore -f - -m  
my_ip_map_file -s my_shells_map_file
```



---

# Chapter 5. Configuration file examples

## Selective backup files

The following example files merely list the items that should be backed up. Place one item per line, use no other delimiters.

### Client logins file

```
client_1  
client_2  
client_x
```

### Client id's file

```
123  
321  
333
```

### Domain names file

```
domain1.someplace.com  
domain2.someplace.com  
domainx.someplace.com
```

### Domain id's file

```
213  
312  
222
```

## Mapping files

### IP mapping file

You can create the IP mapping file completely manually. But you can also have psarestore create the template of this file for you. To do that, execute

`psarestore` with `-m ip_map_file_template`, where `ip_map_file_template` is a nonexistent file. The `ip_map_file_template` file will be generated automatically. If you generate this file automatically, please take time to review it. You may need to make certain adjustments there by editing it manually.

This technique is quite useful in many respects. Basically, you will have a file with all the necessary IP addresses accounted, all you will have to do is to supply the necessary target IP addresses for mapping. Here is an example of such file:

```
# ATTENTION: all not existing TARGET ip addresses will be
added into your system.
# Write target ip addresses for mapping.
# for example:
#
# fxp0:
# 192.168.1.1 -> 195.173.23.200
# eth0:
# 192.168.1.2 -> 193.233.1.130
# 192.168.1.3 -> 193.233.1.131
#
# or simply:
# 192.168.1.1 -> fxp0:195.173.23.200
# 192.168.1.2 -> eth0:193.233.1.130 255.255.255.0
# 192.168.1.3 -> eth0:193.233.1.131 255.255.255.0
#
# 1) if ip address exists in the system then it will be
skipped on the old interface.
# 2) if netmask was not specified then used 255.255.255.0

# the FullHostName:
# host.someplace.com -> host.someplace.com # if this
option is not be specified then PSA will be configured
with this hostname.

#####
# The following ip addresses are used for domains:
#####

# used for domains: another.domain
10.1.15.114 -> lnc0 : 10.1.15.114 255.255.255.255   ###
exclusive

# used for domains: dom1.host.someplace.com
10.1.15.115 -> lnc0 : 10.1.15.115 255.255.255.255   ###
exclusive

# used for domains: domx.host.someplace.com
10.1.15.119 -> lnc0 : 10.1.15.119 255.255.255.255   ###
shared

#####
# The following ip addresses are not required:
```

```
#####
fxp0:
# included in ip pool for the clients: test1
10.1.208.2 -> 192.6.168.13 255.255.255.0 ### exclusive

# included in ip pool for the clients: test2
10.1.208.3 -> 192.7.168.12 255.255.255.0 ### exclusive

#####
# The following ip addresses are external and located on
other hosts:
#####
# used for domains: some.domain, some.other.domain,
yet.another.domain
# 10.1.15.116 ->
```

## Shell mapping file

All the above is true for the shell mapping file. Execute `psarestore` with `-s shell_map_file_template` to automatically generate a file, containing all shells that were mentioned in the backup. Following is an example of this file.

```
/bin/sh => /bin/sh
/usr/local/bin/rbash => /usr/local/bin/rbash
/usr/bin/false => /usr/bin/false
/bin/csh => /bin/csh
/usr/local/bin/bash => /bin/bash
```

## Client mapping file

The same way as with the IP address mapping file you can choose between creating such file manually or having it generated by `psarestore`. In this case, execute `psarestore` with `-c client_map_file_template`. Here too, if you generate this file automatically, please take time to review it. You may need to make certain adjustments there by editing it manually. Following is an example of the generated client mapping file template.

### NOTE

This file will be generated automatically only if there are client accounts collisions present. Otherwise, if you wish to make a custom mapping of restored clients you have to create this file manually.

If a client account in the dump file and a client account that already exists at the server, where the restore is executed, have the same login name and personal name, the two such accounts are considered the same and the data are merged.

** IMPORTANT**

If the limits of the latter account cannot be observed at the restore, the restore fails. In this case, you may want to restore this account separately, by mapping it.

If only the login name or only the personal name is the same, this situation is considered a collision and the restore will not be performed.

```
#The following clients presented in dump have a collision
with some clients in Plesk.
#Collision is detected for a client if he has login or
personal name which coincides
#login or personal name of some client in Plesk.
#
#
#
login:client_1 => new_client_1
source personal name:Client One
target personal name: New Client One
-----
login:client_2 => new_client_2
source personal name:Client Two
target personal name: New Client Two
-----
```

** NOTE**

Different clients must have unique login names and personal names. Therefore you may need to modify the login name for the mapped client and/or maybe the personal name too. Also, it is important to separate each set or client data with a delimiter, which consists of one or more dashes.