D2u_*

d2u_* utilities

A set of **d2u_* utilities** is used for automatic update of D2000 System from FTP server.

Note: This state works starting the version d2u_client 2.2.0.6 (D2000 v7.02.002 patch).

The types of d2u_* utilities:

- d2u_service
- d2u_client
- d2u_reload
- d2u_snapshot

The fast mode (default) is available from the version **d2u_client 2.3.0.24**. It causes that the files and directories are not checked on the client, but it reads UUID (identifier) in a local file "content.xml" in *installdir* and *appldir*. If they are consistent with UUID in "content.xml" on FTP server, the client is considered to be full-synchronized.
If a local UUID in "content.xml" is different or missing, the full update is executed and UUID is written into local "content.xml", if any error has not occurred. The fast mode may be ignored by the parameter /F (see command line parameters).

To apply the fast mode you should use d2u_snapshot.exe (minimally the version since October 24, 2011) and regenerate the files *content.xml* on FTP server.

d2u_service utility

d2u_service utility runs as service under a user with the rights to write into installation and application directory. It enables updating even if the user without rights to write is logged on. The utility must be registered by a local administrator who starts "**d2u_service -i**" in the directory *D2000_INSTALL_DIR\bin*.

IPSAutoRun in d2u_service.exe

IPSAutorun enables to start the prepared package from defined folder or USB key. This package can update the system, install patch, perform some service, etc. It is digitally signed by an authorized subject and **d2u_service** starts this package only after successful verification of signature.

IPSAutoRun package consists of two files:

- <<name>>.7z
- <<name>>.7z.sig

while "<<name>>.7z" is .7zip archive and "<<name>>.7z.sig" is a file with the digital signature that guarantee the authentication of archive. The archive must contain the file "\autorun.cmd" and other files.

d2u_service.exe monitors the presence of IPSAutoRun packages:

- in defined folder (WatchedDir),
- in a root directory of connected USB key.

IPSAutoRun configuration

d2u_service.exe reads these parameters from registry:

(string) HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<<base>>\IPSAutoRun\PublicKey
- full path to the file with a public key of authorized subject (e.g. "c:\D2000\D2000.EXE\utils\ipsautorun\public_key.pem"). This file must be protected against deletion / writing by the access rights of OS against unauthorized users.

(string) HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<<base>>\IPSAutoRun\UnpackDir
- path to the work directory in which the autorun packages will be unpacked and from which they will be started (e.g. "c:\temp\ipsautorun\")

(string) HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<<base>>\IPSAutoRun\WatchedDir
- path to the work directory, which will be monitored for the presence of autorun packages. Due to the consistence of package and signature, at first the archive <name>.7z must be copied into the folder and then the signature <name>.7z.sig.

Signing the packages

The authorized subject generates both the private + public key by, for example, *openssl.exe*, which should be downloaded from a safe source (or found in the directory <D2000_installation_directory>/bin or /bin64).

Generating the keys in the Windows command line cmd.exe:

```
set RANDFILE=.\.rnd
openssl genrsa -out private_key.pem 4096
openssl rsa -pubout -in private_key.pem -out public_key.pem
```

In the current folder, there will be created two files "private_key.pem" - private key and "public_key.pem" - public key. Private key must be secured against unauthorized access. It will be used for signing the packages. Public key must be distributed on the workstations with the running d2u_service.

Preparing and signing the packages

Insert a batch file *autorun.cmd*, which contains the necessary functionality and other useful files, to the package (to the root folder). Pack the package into 7zip format (www.7-zip.org). Then sign the package by the private key by this way (once again use *openssl.exe*):

```
openssl dgst -sha256 -sign private_key.pem -keyform PEM -out .7z.sig .7z
```

.7z.sig file will be created, which together with .7z, forms the package prepared for use for d2u_service.

d2u_client utility

d2u_client utility updates the installation and application directory compared to FTP server and starts required process (typically HI.EXE).

The utility needs the following attributes in the system registers:

[HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\D2000V70\Update]
"FTPServer"=*ip_address_of_ftp_server* ; example: "192.168.103.10" [1]
"FTPMode"=*ftp_connection_mode* ; acceptable modes are: "active" and "passive", it depends on a configuration of firewall
"FTPApplDir"=*application_directory_on_ftp_server* ; example "/pub/D2000.APP", [2]
"FTPInstallRoot"=*installation_directory_on_ftp_server* ; example "/pub/D2000.EXE", [3]
"FTPUser"=*ftp_login* ; example "d2update"
"FTPPassword"=*ftp_password* ; example "w589pt7yegsf6hjx" [4]

1. D2Update can use more FTP servers. If it uses more FTP servers, their addresses must be separated by a comma. D2Update tries gradually a connection with all the servers and made an update with the first functioning server. Number of FTP servers is unlimited.
2. The structure of the application directory on FTP server:

   file /*application_directory_on_ftp_server*/content.xmld2group

   The file must be generated by d2u_snapshot.exe utility after each change of the application directory on FTP server:

   application directory /*installation_directory_on_ftp_server*/application1/
   application directory /*installation_directory_on_ftp_server*/application2/

   Above mentioned are the application directories of the specific applications.
3. The structure of installation directory on FTP server:

   file /*installation_directory_on_ftp_server*/update/content.xml

   The file must be generated by d2u_snapshot.exe utility after each change of the application directory on FTP server.

   directory /*installation_directory_on_ftp_server*/scripts/

   It contains the scripts which are started before and after update.

   /*installation_directory_on_ftp_server*/update/

   It contains the folders bin/, gtk/, log/ etc.
4. *Password* is decoded and readable.

Command line parameters:

/V	verbose
/S=scriptsfile	starts the scripts that are described in *scripstfile*
/Lsk	Slovak language
/Q	quiet
/A	updates only an application directory
/T	a test - it simulates the updating and does not write anything

/F	fullcheck - ignores a local *content.xml*
/Dtime	delay [time] in seconds

d2u_reload utility

This utility is used by both d2u_service and d2u_client utilities to update themselves.

d2u_snapshot utility

After starting this utility creates the file **content.xml** in current folder. This file is placed on FTP server and **d2u_client.exe** compares the files. If any file is missed the utility downloads it from FTP server.
A typical technique is to copy **d2u_snapshot.exe** into /*application_directory_on_ftp_server*/ and /*installation_directory_on_ftp_server/update*/.

Each change in the installation and application directory on FTP server requires activating the both utilities. They will generate the files /*application _directory_on_ftp_server*/content.xml and /*installation_directory_on_ftp_server*/update/content.xml. While the client is updated **d2u_client.exe** utility downloads these files and makes update according them.

> ⓘ **Related pages:**
>
> D2000 System utilities