





## CONTENTS

1. Introduction	4
2. Basics	5
2.1 Filing location	5
2.2 Functionality	5
2.3 Automation	5
2.4 Required permissions	5
3. Running the script	6
3.1 Prerequisites	6
3.2 Copying the script to the target system	7
3.3 Running the script manually	8
3.3.1 Remote login	8
3.3.2 Making the script executable	9
3.3.3 RUNNING THE SCRIPT	9
3.4 Automatic execution	10
4. Importing the inventory data	11

## 1. Introduction

For the remote inventory of a Linux system with Docusnap SSH must be enabled on the Linux system and access must be possible via root user or sudo. In some Linux distributions (such as Ubuntu), remote access for the root user is blocked by default. In this case, Docusnap will not have the required permissions to scan this Linux system. If it is not possible to enable remote access via a root login, you can use the DSLinux script for this purpose.

IGEL Thin Clients based on Linux can be inventoried with this script as well.

Corresponding scripts are also available for Windows or Exchange Server systems. They are covered in separate support documents:

- Docusnap Script Windows
- Inventorying Microsoft Exchange

Finally, we will demonstrate how to import the resulting information into the Docusnap database.

## 2. Basics

### 2.1 Filing location

The Docusnap installer stores the DSLinux script in the **Bin** sub-directory of the Docusnap installation directory. The script is available in a 32-bit and a 64-bit version.

### 2.2 Functionality

When you run the DSLinux script, it creates XML output containing all inventory data of the local system. Using a redirection operator, you can write this output to an XML file. Then, this XML file can be imported into Docusnap.

For further details, please refer to the [IMPORTING THE INVENTORY DATA](#) section.

### 2.3 Automation

It is also possible to run the DSLinux script in an automated way. By specifying corresponding parameters, you can store the XML files created by the script in a central network share and then import it automatically at the time specified in the schedule.

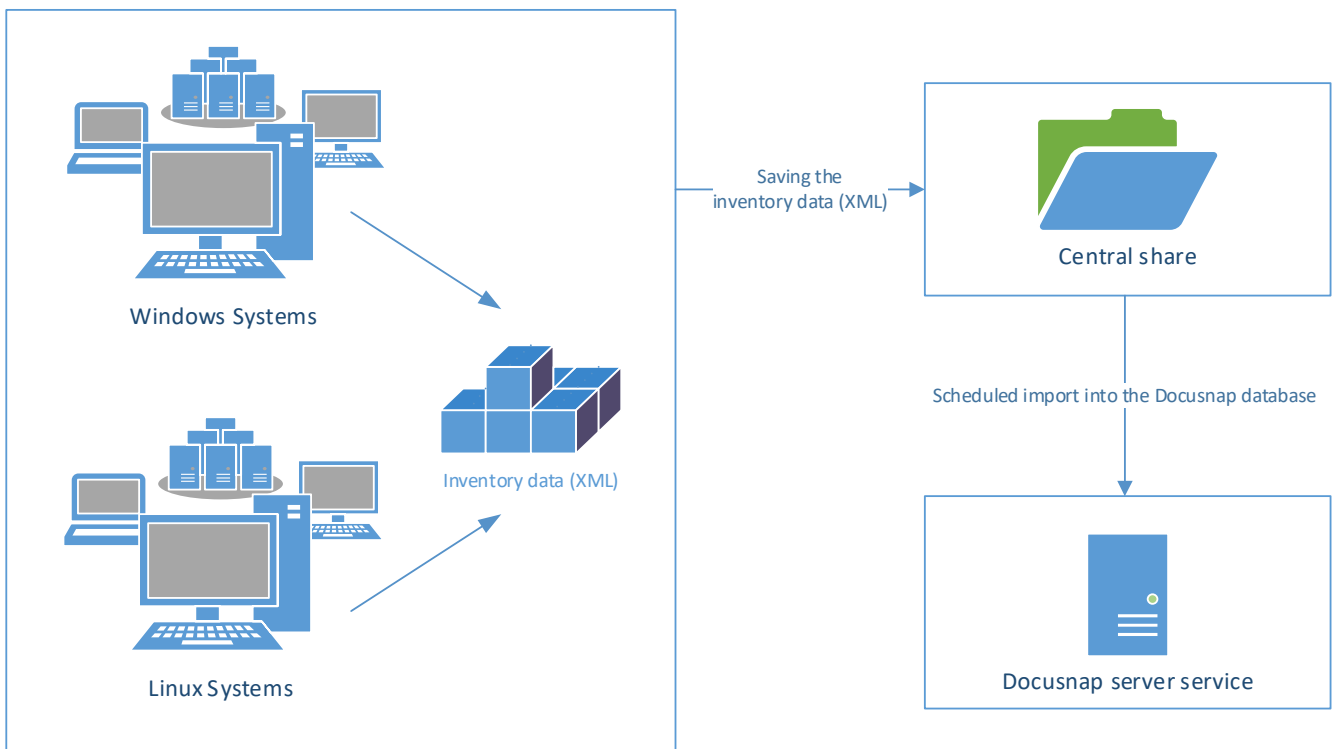


Figure 1 - Workflow of the automatic import

### 2.4 Required permissions

In order to properly run the DSLinux script, root permissions are required on the corresponding Linux system.

## 3. Running the script

### 3.1 Prerequisites

Before running the DSLinux script, check whether the Linux system to be scanned is a 32-bit or a 64-bit system. Select the matching script file (DSLinux\_x32 or DSLinux\_x64) for that system.

In most Linux systems, you can use the *uname -a* command to determine the system architecture.

This document describes how to access the Linux system using SSH.

For remote access to the Linux system to be scanned from a Windows system, the following free tools are available:

- WinSCP
- PuTTY

What is more, the script can be run directly from the console of the Linux system. The commands used in the shell are the same as for remote access.

In both cases, the DSLinux file must be stored either directly on the Linux system or in a directory that can be reached from the Linux system.

## 3.2 Copying the script to the target system

This section describes how to copy the DSLinux script to the target system using WinSCP.

This step is only required if you cannot access the DSLinux script on the Linux system via a network share. If it is not possible to access the DSLinux script, you can copy the required version to the network share and then continue with the RUNNING THE SCRIPT MANUALLY section. The paths shown in the example must be adapted accordingly.

Once you start WinSCP, the Login dialog opens where you can connect to the target system. To do so, enter the host name, the username and the associated password.

After successful login to the Linux system, the left panel in WinSCP shows the Windows file system, the right panel the file system of the Linux system.

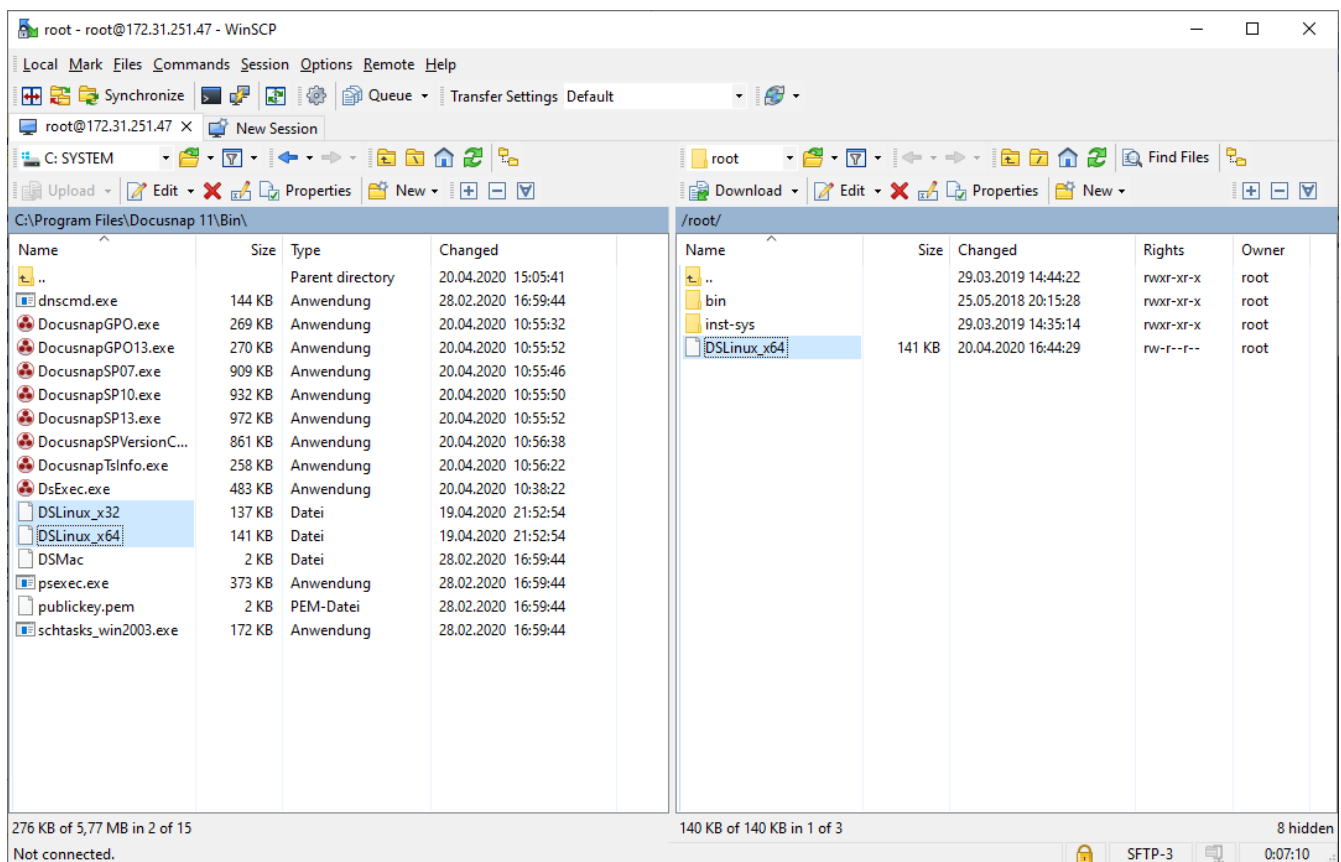


Figure 2 - WinSCP directory view

In the file system of the Windows computer, browse to the "Bin" directory below the Docusnap program directory. For the Linux script, the following two versions are available:

- DSLinux\_x32
- DSLinux\_x64

Depending on the Linux architecture used on the computer (32-bit or 64-bit), now copy the matching file to any directory on the Linux target system.

## 3.3 Running the script manually

### 3.3.1 Remote login

After you have copied the DSLinux file successfully to the Linux system, you need to establish an SSH connection. For this purpose, use the PuTTY tool.

In WinSCP, a button is available that opens the PuTTY tool and establishes a connection to the target system.

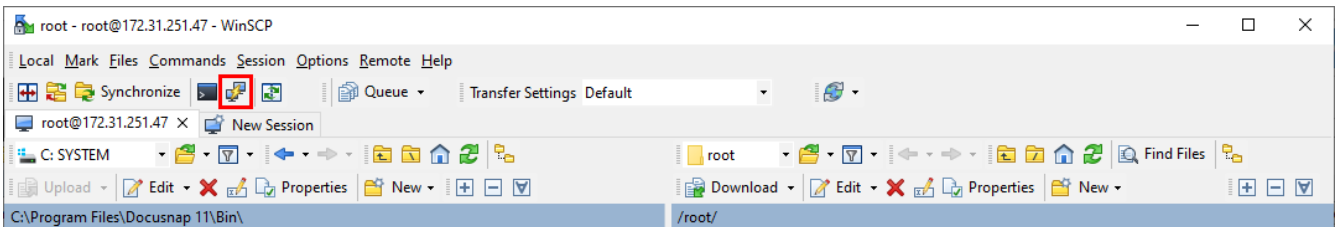


Figure 3 - WinSCP PuTTY button

Then, you need to authenticate on the target system.

After successful authentication, the connection via SSH to the target system is established. There, you can execute commands or browse the file system.

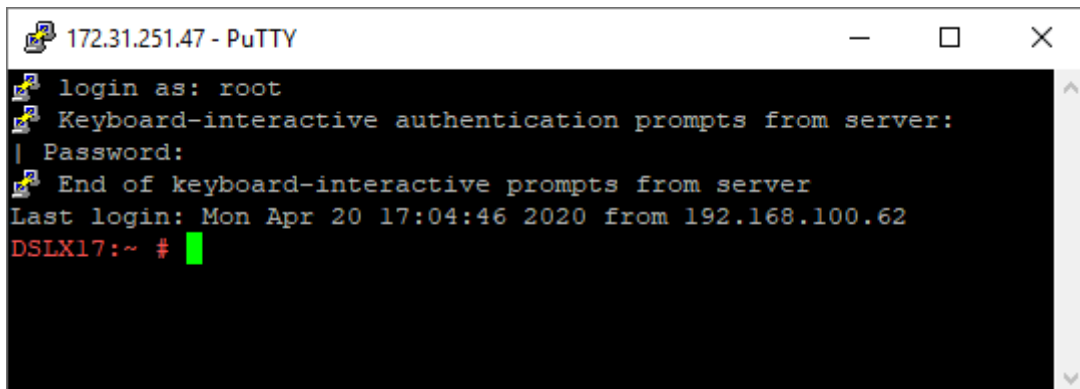


Figure 4 - Successful authentication



### 3.3.2 Making the script executable

Now, navigate to the folder in the Linux file system where the DSLinux script file is stored.

Enter the following command to make it executable:

```
chmod +x DSLinux_x32
```

or

```
chmod +x DSLinux_x64
```

### 3.3.3 RUNNING THE SCRIPT

**CAUTION:** For this and all further steps, root permissions are required. So first enter the *su* command to obtain root permissions.

```
./DSLinux_x32 > outputfile.xml
```

or

```
./DSLinux_x64 > outputfile.xml
```

By entering the redirection operator (>), you create the output file. You can use any desired name for it. Entering an output file name is mandatory. If you omit this name, the retrieved information will not be written to a file, but only displayed on the console.

It is possible to specify not only a file name, but also the directory where you want to store the output file. To store the output file in the */home/docusnap/DSLinux* directory, enter the following command:

```
./DSLinux_x64 > /home/docusnap/DSLinux/outputfile.xml
```

Once the command has been executed successfully, the */home/docusnap/DSLinux* directory contains an .XML file named *outputfile.xml*.

```
./DSLinux_x64 > /home/DSUser/DSLinux/outputfile.xml
cd /home/docusnap/DSLinux
ls
outputfile.xml
```

We recommend to redirect the output file directly to a network share that can be accessed both from Linux and Windows systems.

If the message "line 1104: -n command not found" is displayed while you run the script, ignore it because has no effect on the output file.

### 3.4 Automatic execution

To automate the execution of DSLinux, we recommend that you create a *bash* script and have it started automatically upon login to the Linux system.

The "*bash*" script can be structured like this:

```
#!/bin/bash

HostName='hostname'
ResultFile="$HostName.xml"

if [ "$1" = 64 ]
then
    chmod +x DSLinux_x64
    ./DSLinux_x64 > $ResultFile
else
    chmod +x DSLinux_x32
    ./DSLinux_x32 > $ResultFile
fi
```

This bash script retrieves the host name and uses it as the file name for the output file. In addition, you need to pass the parameter 32 or 64 to the script, depending on the Linux operating system architecture. Before actually running DSLinux, it ensures that the DSLinux is executable by entering the *chmod +x* command.

Before executing the bash script, it must also be made executable by using the *chmod +x* command.

```
chmod +x Scriptname
```

Enter the following to run the script:

```
./Scriptname 32
```

or

```
./Scriptname 64
```

After execution, you will find an .XML file whose file name is the same as the host name in the folder where the DSLinux file resides. As explained above, it is possible to store the output file in any folder you like. To do so, simply precede the *\$ResultFile* variable with the desired directory name.

```
./DSLinux_x64 > /home/docusnap/DSLlinux/$ResultFile
```

In addition, you can use the script for automatic rule-based execution via a *crontab*.







## LIST OF FIGURES

FIGURE 1 - WORKFLOW OF THE AUTOMATIC IMPORT .....	5
FIGURE 2 - WINSXP DIRECTORY VIEW .....	7
FIGURE 3 - WINSXP PUTTY BUTTON .....	8
FIGURE 4 - SUCCESSFUL AUTHENTICATION .....	8
FIGURE 5 - STARTING THE IMPORT WIZARD .....	11
FIGURE 6 - DEFINE XML IMPORT PATH.....	12
FIGURE 7 - SEARCH XML IMPORT FOR FILES .....	13

## VERSION HISTORY

---

<b>Date</b>	<b>Description</b>
April 27, 2017	Version 1.0 – First Draft
October 24, 2018	Version 1.1 – Replacement of outdated screenshots, links updated
February 07, 2020	Version 1.2 – Correction of the bash script
May 23, 2020	Version 2.0 - Revision of the HowTos for Docusnap 11

---

