Windows Inventory

Inventory of Windows Systems

TITLE	Windows Inventory
AUTHOR	Docusnap Consulting
DATE	10/18/2022
VERSION	1.0 | valid from 10/6/2022

CONTENTS

1. Introduction

Docusnap inventories your Windows systems. Besides hardware information (manufacturer, serial number, processor, RAM, monitors, etc.) you also get installed software products, server roles and updates as well as local users, security center information, services and much more (WSUS configuration, drive information with BitLocker status).

The Windows inventory requires a WMI connection. During the inventory, PowerShell queries are performed in addition to WMI queries. In detail used ports and necessary permissions are listed in the whitepaper for the inventories.

In addition to the remote inventory, a script variant can also be used for the Windows inventory. This topic is described in a separate HowTo: Inventory - Docusnap Script for Windows.

In chapter 2 you will first find the prerequisites for Windows inventory.

Chapter 3 describes the two remote inventories.

In chapter 4 you will find information regarding error analysis / explanation of possible error messages.

In chapter 5 you will find information about the file and software search.

Chapter 6 describes the use of the additional programs.

2. Prerequisites

The Windows inventory requires a transparent firewall configuration in the first step. In domain networks, we recommend setting the firewall configuration using group policy. Here you can use the preconfigured firewall rules:

- File and Printer Sharing (ICMP echo request - ICMPv6-In) and
- File and Printer Sharing (ICMP Echo Request - ICMPv4-In)
- Windows Management Instrumentation (WMI) - WMI incoming

The same settings are available for local firewall configuration of workgroup systems.

In the second step, a local administrator is needed to establish the WMI connection.

More detailed information about used ports, protocols and required permissions can be found in the whitepaper for inventories.

3. Windows inventory

Two inventory variants are available for remote Windows inventory:

Windows (AD) - captures domain-integrated Windows systems.

Windows (IP) - scans systems based on an IP range. You can use this scan wizard if you or your customer has only one workgroup and therefore no domain available.

3.1 Windows AD Inventory

For the network inventory of Windows AD systems start the corresponding wizard:

- Discovery - All Wizards - Windows (AD)
- Inventory - All Wizards - Windows (AD)
- All Jobs - All Wizards - Windows (AD)

In step 1, select your company or your customer's company.

In step 2, select the discovery service through which you want to perform the inventory.

For a scheduled inventory, select Docusnap Server Discovery or a Discovery Service you have configured here.

Furthermore, select or enter the domain and an appropriately authorized user with their password. Use the NetBios notation for the user (domain\user).

In step 3 you can now set the necessary properties for the inventory:

Using the filter criterion Start search at - Set OU, it is possible to limit the inventory to specific organizational units.

The Start search button now collects all ACTIVE computer accounts from the Active Directory.

These computer accounts will now be listed for you. You now have the possibility to set filters - for example on the name, type or the last inventory.

You can activate or deactivate the filter with a right click in the data area.

Via Select components you can narrow down the components to be inventoried.

The option Update selected systems: ... is crucial if you set up a time-controlled job and should be active!

This option causes the scheduled inventory to be dynamic. New computer accounts are included in the inventory and deactivated / deleted computer accounts are no longer included.

3.1.1 Best Practice - Windows (AD) Inventory

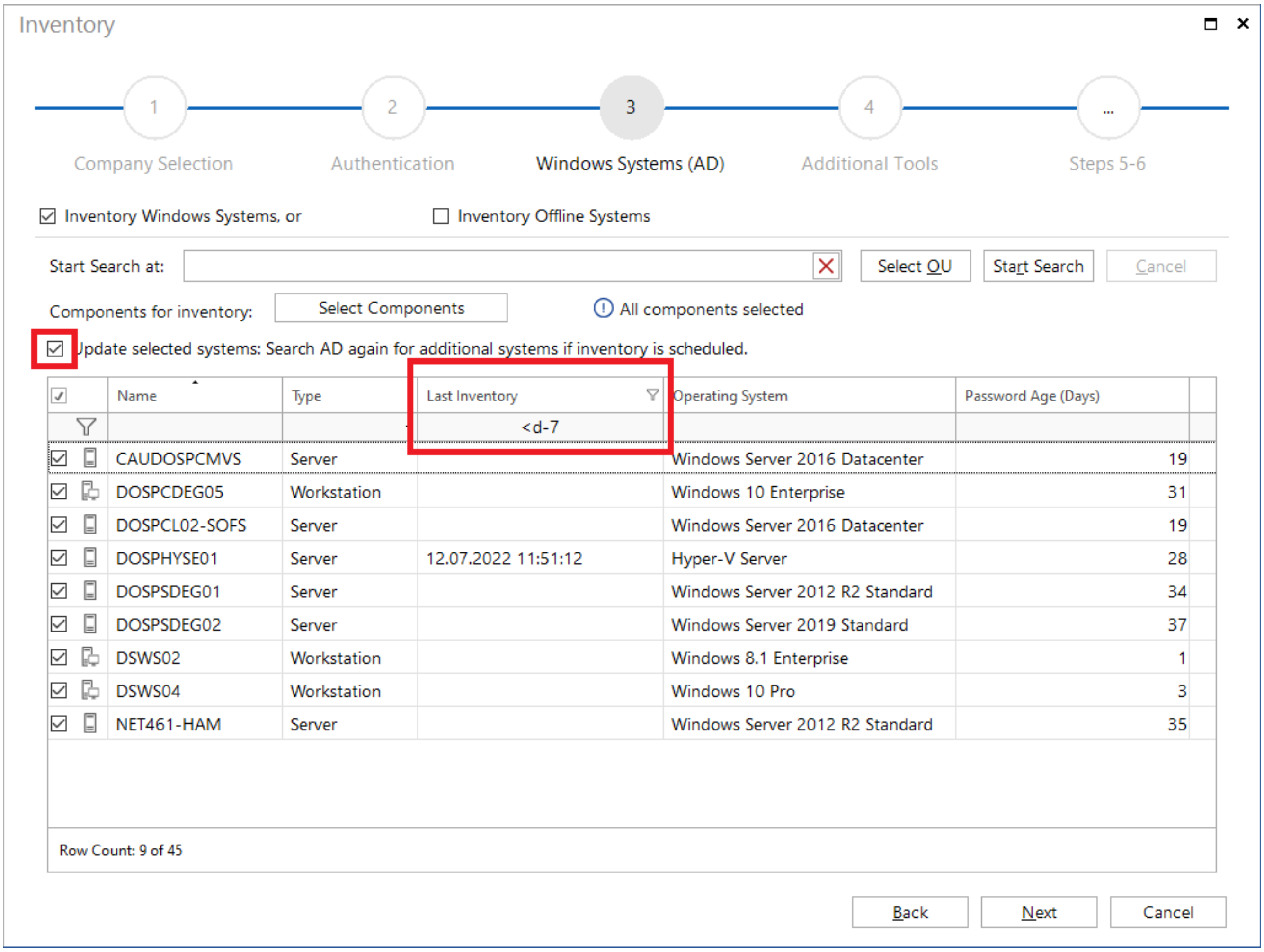
We recommend setting up the Windows (AD) inventory as a recurring job!

Step 2: Use Server Discovery or a discovery configured by you.

Step 3: Enable the option: Update selected systems: ....
Filtering: Last inventory: <d-7

Step 5: Schedule type:	Repeated
Occurrence:	Weekly
Repeats Every:	1 week on: Monday through Friday
Frequency per day	
All:	3 hours between 6 am and 6 pm

The filter set up in step 3 causes only systems whose last inventory is older than 7 days to be considered during the inventory runs. Furthermore, systems that have never been inventoried at the current time are considered. In this way, existing systems are regularly updated and new systems are inventoried as quickly as possible.



☑ Inventory Windows Systems, or ☐ Inventory Offline Systems

Start Search at:

Components for inventory: All components selected

☑ Update selected systems: Search AD again for additional systems if inventory is scheduled.

☑	Name	Type	Last Inventory	Operating System	Password Age (Days)
			<d-7		
☑	CAUDOSPCMVS	Server		Windows Server 2016 Datacenter	19
☑	DOSPCDEG05	Workstation		Windows 10 Enterprise	31
☑	DOSPCL02-SOFS	Server		Windows Server 2016 Datacenter	19
☑	DOSPHYSE01	Server	12.07.2022 11:51:12	Hyper-V Server	28
☑	DOSPSDEG01	Server		Windows Server 2012 R2 Standard	34
☑	DOSPSDEG02	Server		Windows Server 2019 Standard	37
☑	DSWS02	Workstation		Windows 8.1 Enterprise	1
☑	DSWS04	Workstation		Windows 10 Pro	3
☑	NET461-HAM	Server		Windows Server 2012 R2 Standard	35

Row Count: 9 of 45

Figure 1 - Best Practice Windows (AD) inventory

Depending on the size (750 Windows systems upwards) of your environment, it may be beneficial to set up more than one scheduled Windows (AD) job. Here you can use the various filtering options - for example

- Job with filter on the type: Server + Domain Controller
- Job with filter on the type: Workstations
- Job with filter on the name: Location abbreviation*, e.g. FFM* / MUC*
- Job with filter on specific ADDS containers

In multi-site environments, inventory can be split using Docusnap Discovery Services (DDS). The DDS are used to inventory decentralized networks (e.g. sites and compartmentalized VLANs).

3.1.2 Best Practice – DocusnapScript.exe

In addition to the best practice approach described above, another approach is the use of DocusnapScript.exe, the script-based inventory of Windows systems. Its execution can be automated and efficiently performed using group policy or software distribution tools.

The use of DocusnapScript.exe can / should be in addition to the remote inventory described above. Parallel execution is recommended especially in larger environments (750 Windows workstations and up).

3.2 Windows (IP) Inventory

For the network inventory of the Windows IP systems start the corresponding wizard:

- Discovery - All Wizards - Windows (IP)
- Inventory - All Wizards - Windows (IP)
- All jobs - All wizards - Windows (IP)

In **step 1**, select your company or your customer's company.

In **step 2**, select the discovery service through which you want to perform the inventory.

> For a scheduled inventory, select Docusnap Server Discovery or a discovery service you have configured here.
>
> Furthermore, select or specify a / the domain under which the systems to be inventoried are to be sorted.

In **step 3**, you can now set the necessary properties for the inventory:

> Via **Select components** you can narrow down the components to be inventoried.
>
> You can now define one or more **IP range(s)** in which the systems to be inventoried can be found.
>
> > In addition to the **IP range**, a local administrator and password are required. This local administrator and password must be available across all systems. Enter the user as follows: .\LocalAdministrator
> >
> > Now start the search. In the window below you will find the Windows IP systems found.
> >
> > You can also enter the Windows systems manually in the Systems section.
> >
> > Here you can enter either the host name or the IP address.
> >
> > Again, a local administrator (.\LocalAdministrator) and its password are required for the inventory.
>
> In the advanced options you still have the following configuration options:
>
> > Limit pings executed in parallel during the search.
> >
> > > Adjustments can be made if the firewall alerts during the search
> >
> > Discard systems already found when searching again in the wizard
> >
> > > With this option, the systems found are first discarded during each search. Useful if a new IP range is to be stored as a time-controlled job.
> >
> > Update selected systems: ...
> >
> > > This option causes the time controlled inventory to be dynamic. The deposited IP range is searched again.

4. Error analysis / error messages

A scheduled Windows inventory is completed as erroneous in 99% of cases. This should not be a cause for concern. If a system is not successfully inventoried, for example because it is switched off, then the error will occur.

You should periodically check the error messages for each Windows system in the Discovery or All Jobs area.

Select the corresponding scheduled job and then select Summary.

The following error messages in the course of Windows occur frequently:

Error	Description
No ping	System unavailable - turned off or no longer active
Access denied	User does not have local admin permissions
RPC server is not available	Firewall not configured for WMI access
Timeout	Busy system / bad connection - increase timeout
Port blocked	Firewall not configured for WMI access
DNS failed	System has no DNS entry
DNS misconfiguration	DNS forward and reverse lookup are different

For more error sources and analysis, see the inventory whitepaper.

5. Software and file search

The software and file search in Docusnap is used to search for specific files on the file system of Linux, Mac, and Windows systems. File names are defined, which Docusnap then searches for in the course of the inventory on the file system.

The files found are made available for evaluation in Docusnap in different ways depending on the categorization (file search Linux, Mac and Windows or software search Windows).

The software search here refers to applications that have been "installed" on the target system without registration. If the defined files are found, a corresponding entry is created in the list of installed software products on the system. Thereupon this software can be analyzed also in the range of the license management.

The file search designates here any files, which you would like to search, for example due to a safety gap (log4j). The files are then also listed with the path in which they were found. A new object within the summary as well as a new predefined Docusnap Connect package provide the possibility for cross-system analysis.

The setup of the software and file search is described in a separate HowTo: Inventory - Software and File Search.

6. Additional programs

With the help of additional programs you can extend the Windows inventory with certain properties, so that not only WMI classes are queried, but also command line programs, such as systeminfo.exe, values from the registry or even PowerShell queries. The results are written to the Docusnap database below the respective system.

6.1 Limitations

When it comes to Additional Tools, there are also few limitations that you should be aware of.

- In order to run an additional program remotely, access to the C$ share is required
- Calling a script from a network share is not possible. The program to be executed must exist either on the Docusnap system that performs the inventory or on the target system that is currently being queried. Storage in the form of a UNC path (\\hostname\share\example script.exe) is not possible.
- Command line commands that are executed exclusively in the context of the 64-bit shell cannot be stored in the additional tools. Only commands that can be executed in the 32-bit and 64-bit context will work.
    - For example, it is not possible to access 64bit paths using PowerShell or RegQuery - following the example of the Team Viewer ID on a 64bit installation: Computer\HKEY_LOCAL_MACHINE\SOFTWARE\TeamViewer
    - Another example is that the command: netsh mbn show interface using additional programs does not return any results because the call is made in the 32bit context

6.2 Administration

In the Docusnap administration you will find the administration of the additional tools in the Inventory area. Within this, you can create new additional tools programs or modify existing ones. The following fields are available as properties of the additional tools:

- Name
- Program path

Path / name of the executable program that will be used. Command line programs / PowerShell can be specified without path. For remote execution, the program must be available on the target system.

- Parameters

Here the parameters are passed to the executable program.

For PowerShell queries, the actual command is output here. The PowerShell command must be enclosed in brackets! Pipes and quotation marks must be masked. Quotation marks with \ (\"\) and pipes with ^ (^|^).

- Results file

Docusnap stores the results in BLOB format in the database, which needs a name. The name of the result file is again freely definable, but the format must be specified, e.g. whether it is a .TXT file or a .DOCX file, etc.

- Open with

With this field you can define which program is used to open the previously created result file (e.g. notepad.exe, WinWord.exe, etc.).

- Type

If Stream is selected, the results from the standard output are immediately written to the database. However, the cmd.exe will be visible for a short time.
We recommend the use of the stream.
If File is selected, a temporary file is created on the hard disk during the inventory, in which the results are saved - this file is then read into the database and saved. The File selection requires the parameter > %targetfile%.

- Run additional programs remotely

Here it is defined whether the program is executed on the system where the inventory process takes place or remotely at the particular system that is being inventoried.

If an additional program is not executed remotely, further specific parameters are available. These can be found in the configuration manual - search word add-on programs - F1 key in Docusnap in the context of add-on programs.

6.3 Activate additional programs for the inventory

In order to be able to use the additional programs in the context of the Windows AD and/or IP inventory, this step must first be activated.

To do this, switch to the Docusnap options via the gear wheel in the upper right corner. In the Inventory section, you can activate the Additional programs option.

6.4 Examples

The following example serves to illustrate the additional programs. The firewall status is already part of the standard Windows inventory.

Query firewall status via PowerShell	
Name:	Windows Firewall Status
Program Path:	PowerShell.exe
Parameters:	(Get-NetFirewallProfile ^|^ Select-Object -Property Name, Enabled)
Result file:	FirewallStatus.txt
Open with:	Notepad.exe
Type:	Stream
Timeout:	10000
Run additional program remotely:	Ja

Zusatzprogramm für die TeamViewer ID	
Name:	TeamViewerID
Program Path:	%systemroot%\System32\WindowsPowerShell\v1.0\powershell.exe
Parameters:	32bit (Get-ItemProperty -Path "HKLM:\SOFTWARE\Wow6432Node\TeamViewer").ClientID 64bit: Not supported!
Result file:	TeamViewerID.txt
Open with:	Notepad.exe
Type:	Stream
Timeout:	10000
Run additional program remotely:	Ja

6.5 Evaluation options

6.5.1 Output in the tree structure

The first evaluation option is to view the results via the tree structure. You can find the additional programs below

- Company - Infrastructure - Domain - Workstations/Servers - System - Snapshot - Add-on programs.

6.5.2 Output in data sheets

The second evaluation option is possible in the form of data sheets. In the wizard for creating the data sheets, you have the possibility to set the option Output results of additional tools. This causes the results to be output as a corresponding file in your documentation path.

You will find detailed instructions on the data sheets in the Data sheets section of our user manual.

6.5.3 Output via Docusnap Connect

With the help of Docusnap Connect you can "quickly and easily" create your own queries. In this case, you can create a query about the results of the add-on programs.

A detailed description regarding the creation of Connect packages can be found in the following HowTo: Docusnap 11 - Docusnap Connect, CSV Import.

You can create the Connect package to output add-on programs with the following selection.

Docusnap objects:

- Workstations
  - Additional programs

As columns use for example these:

Element	Column
Workstation	Name
Additional Tools	Description
Additional Tools	ToolResultVarchar

Now you can view the results of the additional programs via the workstations. In the tree structure, you can find this in the Inventory - Your Company - Connect section.

6.5.4 Ausgabe per View

As a final evaluation option, you can also use a more advanced variant in the form of an SQL query. For example, a new node can be created in the inventory tree, which displays the results of the additional programs and the most important system information.

```
select tHosts.HostID, tHosts.Hostname, SiteID, OS,
(select ToolResultVarChar
from tToolResult where DocuID = (select DocuID from tDocu where tDocu.HostID = tHosts.HostID and Archiv
= 0)
and Filename like '%Example.exe%' -- the name of the result file must be entered here
) AS BuildNumber
from tHosts, tDocu, tDocuWindows
where tHosts.HostID = tDocu.HostID
and tDocu.DocuID = tDocuWindows.DocuID
and HostTypeID = 1
and Archiv = 0
```

The SQL queries can also be customized to start from a certain character. For this you need the substring function in SQL. In this example, everything up to the 84th digit is truncated and then the following 100 characters are output.

```
SUBSTRING(convert(varchar(max), cast(ToolResult as varbinary(max))),84,100)
```

For more information on creating views, see the following HowTo: Customizing - Creating a data view.

LIST OF FIGURES

VERSION HISTORY

Date	Description
09/30/2022	HowTo created