

# Effective Automated Windows Lab Deployment

Fons Mijnen Vincent van Dongen

February 6, 2017

# Problem part 1

- IT professionals, students, and researchers use test labs for a variety of reasons.
- Many products and techniques exist to automatically deploy Windows systems. However, these tools only deploy Windows systems and don't configure a realistic test environment
- Therefore:
  - Manual configuration is required to create a useful testlab.
  - Technical knowledge is required to build a testlab.
  - These deployment tools require a lot of user input

## Problem part 2

In order to create a realistic testlab, traces of users and systems have to be added to the testlab. Examples of traces of user and systems are:

- Groups and user account located in the user-database.
- Random files located in user folders.
- Mailboxes with email included.
- Client applications.
- Log and event files.

## Main question:

- *Is it possible to automate a fast and easy rollout of a realistic Windows test environment with minimal user interaction?*

## Sub question:

- *What kind of techniques and methods exist to deploy and configure a testlab?*
- *What is the most suitable option to automate the deployment and configuration of a testlab?*
- *What kind of techniques and methods exists to simulate system and user behavior on machines?*
- *What is the most suitable option to automate the simulation of system and user behavior on windows machines?*

# Defining the testlab

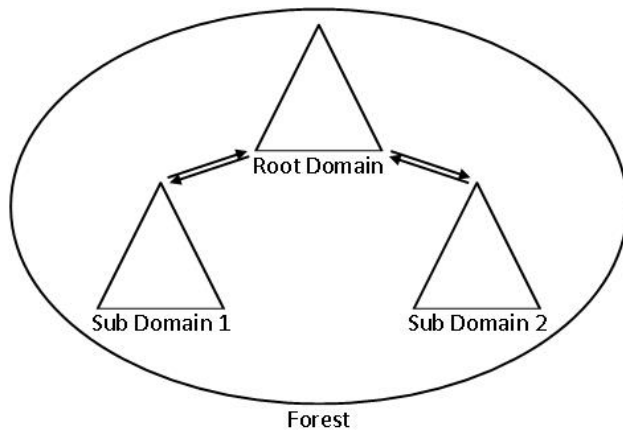


Figure: Overview of the domain structure of the testlab

## Defining the testlab with system requirements

- Multiple Active Directory Domain Controllers.
- Active Directory for a user database with users, groups and Organization Units.
- Email Server, Domain Name Server, Web server, DHCP server, SMB share.
- Client computer systems with internet access.
- Traces of user and system behavior.

## Defining the testlab with functional requirements

- Relatively fast deployment (less than 12 hours).
- Minimal user interaction.
- Definable parameters such as domain names, IP-addresses and users/groups.
- The total costs should be as low as possible.
- The total amount of disk space should be as low as possible.
- Functionality to automatically update Windows servers and clients.

## Current techniques to deploy and configure a testlab

- 1 Configuration management
- 2 Image deployment
- 3 Virtual Machine Snapshot
- 4 Templates



## Current techniques to simulate user and system behavior:

- 1 Groups and user account located in the user-database.
- 2 Random files located in user folders.
- 3 Mailboxes with email included.
- 4 Client applications.
- 5 Log and event files.

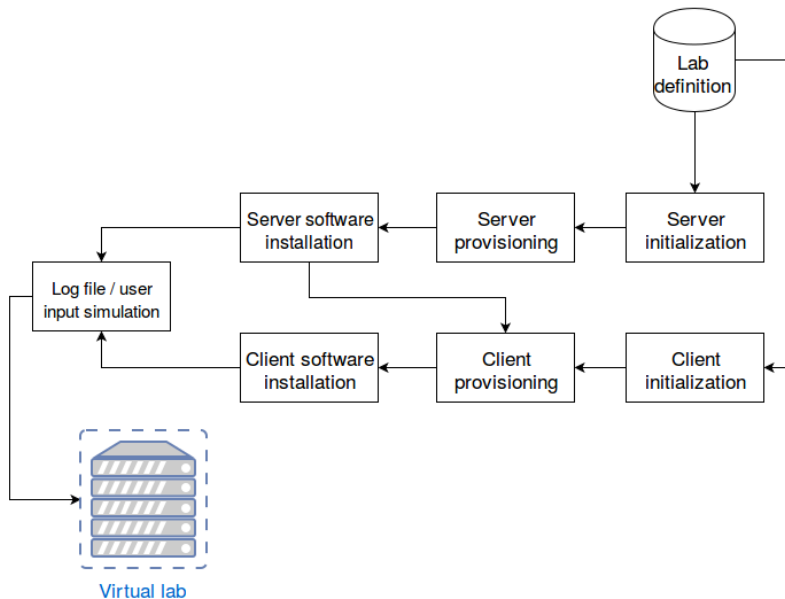
## Conclusion:

By using one these techniques and tools its only partially possible to automatically deploy a realistic testlab. Therefore, some requirements cannot be fulfilled

## Model specification:

- 1 Deployment of server and OS installation.
- 2 Server provisioning.
- 3 Software installation and configuration on servers.
- 4 Deployment of clients and OS installation.
- 5 Client provisioning.
- 6 Software installation and configuration on client.
- 7 Log file and user behavior emulation.

# New Windows testlab deployment mode



- Build a prototype conform to the model and the lab specification
- Prototype build on a Windows server.
  - Intel(R) Xeon(R) CPU E3-1240L v5 @ 2.10GHz 4 cores
  - 16GB RAM
  - 100GB disk
  - Windows server 2012R2 OS

# Prototype: underlying architecture

- Prototype build with Powershell, Windows native scripting
  - Includes many native functions for windows configuration that can be used in the prototype
  - Now windows main focus instead of GUI making sure the prototype is viable in the future [3] [2]
- Hyper-v used as Hypervisor
- All remote invocations and commands are called from the Hypervisor server

---

<sup>3</sup><http://searchwindowsserver.techtarget.com/tip/>

How-and-why-Microsoft-is-killing-the-GUI-on-Windows-Server

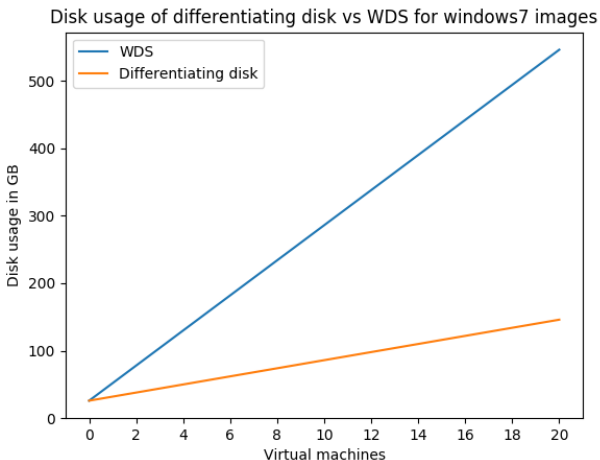
<sup>2</sup><https://mva.microsoft.com/en-us/training-courses/getting-started-with-powershell-3-0-jump-start-8276>

# Phase 0: lab definition

- The lab should be defined before deployment.
- The lab is defined in a XML file.
- IP range, AD domain(s) and a lab name is defined.
- For each machine in the lab:
  - A computer name
  - Programs to be installed
  - Windows OS and version
  - Machine Domain

# Phase 1: Deployment of server and OS installation 1/3


- Differencing disks are used to keep disk usage to a minimum.
- Uses a parent-child relationship where changes are written to the child disk. The parent disk is read only.





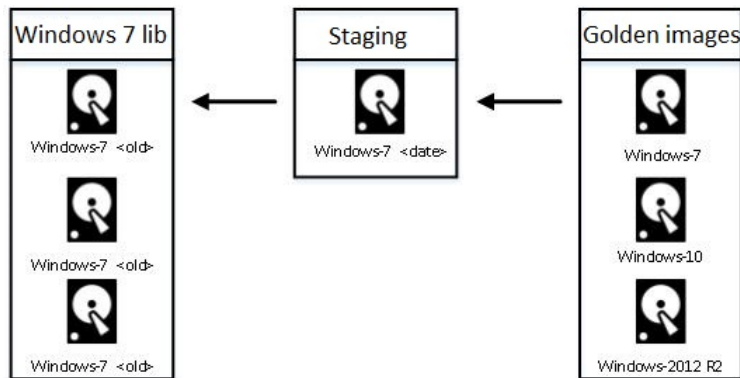
- Sysprep is used to prepare the image for deployment.[1]
- *unattend.xml* is used to automate the final installation steps.
- Sysprep and differencing disks allow for a high level of automation.

---

<sup>1</sup><https://technet.microsoft.com/en-us/library/hh824938.aspx> 

# Phase 1: Deployment of server and OS installation 3/3

Images can be automatically added to the library of *sysprepped* disks



## Phase 2: Server provisioning

- Servers get a static IP, computer names and new passwords
- Servers are contacted through APIPA address
  - APIPA is a IP range windows uses if a NIC fails to get a DHCP address
  - In the 169.254.0.0/16 range
  - Polls for a DHCP server every 5 minutes
- Hypervisor server has a 10.0.x.1 address and 169.254.1.x address on the virtual switch

## Phase 3: Software installation and configuration on servers

- Windows features can be directly installed with `Install-WindowsFeature` cmdlet
- EXE, MSI and other installers can be transferred to the server and then called with `Invoke-Command` feature
  - Undocumented sandboxing and environment issues make installing some software difficult
  - Environment can be broken out of by scheduling cronjob like tasks to do software installation
- ISO, IMG and other image files can be mounted to the virtual DVD drive

## Phase 4: Deployment of client and OS installation

- Virtually identical to server deployment
- In order to spread disk I/O load it is done during DC installation

## Phase 5: Client provisioning

- Clients are given DHCP addresses instead of static addresses
- The clients are then given a new name and joined to a domain

## Phase 6: Software installation on client

- Installers and versions can be stored in a library.
- Transferred and installed with Powershell.
- Allows for multiple clients with different versions of software in a single Lab.

## Phase 7: Log file and user behavior emulation 1/4

- Traces of use are generated by adding random folder, files and file extension to certain places
  - Windows user space (My documents, Desktop, Downloads)
  - SMB shares
  - Home folder
- Mails in the exchange server are also sent with random content at a random time
- Log manipulation is very hard in windows, log files are not flat text files
  - Some tools are around for altering logs in older Windows versions [4]
  - Log files will have to be generated in some other way

---

<sup>4</sup><http://www.securityfocus.com/tools/1726>

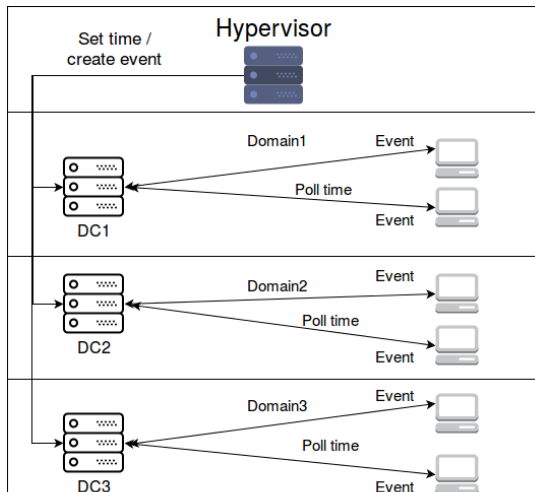


## Phase 7: Log file and user behavior emulation 2/4

- Log files are generated by manipulating the time server
- Sync time with a clients / servers DC every 5 seconds
- DC's jump time every 5 seconds by some amount of minutes in lock step
- Allow for a dynamic acceleration and of time and time it takes

# Phase 7: Log file and user behavior emulation 3/4

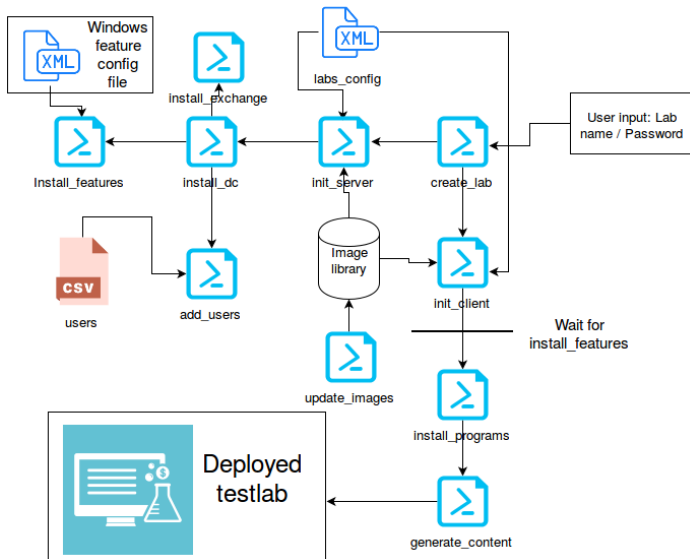
- Scripts are offloaded to clients to sync with the system
- Hypervisor generates events for the servers, clients generate random events from offloaded scripts



## Phase 7: Log file and user behavior emulation 4/4

- System works but not perfect
- AD depends on Kerberos and thus on time being within a certain skew
- Log entries for the time jumps are present on the system
- Not every user action can be created trough powershell and scripting

# Final overview of the prototype



## Findings and evaluation

- 1 Its possible to automatically deploy a realistic testlab.
- 2 Powershell was designed for maintenance.
- 3 Exchange has no function for remote installation.
- 4 Powershell isn't able to alter timestamps in logfiles
- 5 The average installation time is approximately 5,5 hours.
- 6 The average disk space on the hypervisor is approximately 160 GB.
- 7 Replication time between DCs is 15 minutes.

# Results and Comparison

## Comparison table part 1

	CM	WDS	Snapshots
Email-, DNS-, Web-, DHCP-, File-server		X	
Configured DC with replications, sites and trust			X
User database with users, groups, OUs	X		X
Clients, Mail and File-server connected to AD	X		X
Automatically update windows servers and clients	X		
Definable parameters such as DC, IP-addresses and users	X		
Deployment in less than 7 hours		X	X
Minimal user interaction		X	X
Total costs should be less than 2000,-	X	X	X
Total amount of disk space should be less then 250 GB	X		X

# Results and Comparison

## Comparison table part 2


	Templates	Our solution
Email- , DNS-, Web-, DHCP- , File-server	X	X
Configured DC with replications, sites and trust	X	X
User database with users, groups, OUs	X	X
Clients, Mail and File-server connected to AD	X	X
Automatically update windows servers and clients	X	X
Definable parameters such as DC, IP-addresses and users	X	X
Deployment in less than 7 hours	X	X
Minimal user interaction		X
Total costs should be less than 2000,-		X
Total amount of disk space should be less then 250 GB		X

This research has shown that it is possible to automate a fast and easy roll-out of a realistic Windows test environment with minimal user interaction by using methods and techniques specified in our prototype.



# Acknowledgements

*We would like to express our gratitude and appreciation towards our supervisors Marc Smeets and Mark Bergman. Their assistance and guidance during the whole research period have been really valuable to us.*

 [Sysprep \(generalize\) a windows installation.](#)

[https:](https://technet.microsoft.com/en-us/library/hh824938.aspx)

[//technet.microsoft.com/en-us/library/hh824938.aspx.](https://technet.microsoft.com/en-us/library/hh824938.aspx)

 [Jason Helmick Jeffrey Snover.](#)

Mva: Getting started with microsoft powershell, 2013.

 [Don Jones.](#)

How (and why) microsoft is killing the gui on windows server, 2011.

 [Arne Vidstrom.](#)

Winzipper, a windows 2000 log alteration tool.

[http://www.securityfocus.com/tools/1726.](http://www.securityfocus.com/tools/1726)