

Inhalt

Worum geht es?	2
Installation eines Windows Server 2019 Server Core	2
Integration des Feature on Demand „AppCompatibility“	8
Vorbereitung der Generalisierung	11
Testlauf der Bereitstellung	14

Worum geht es?

Windows Server 2019 hat wie seine Vorgänger auch die Installationsvariante „Server Core“. Diese bietet einige Vorteile gegenüber der Version mit „Desktop Experience“ – also der grafischen Oberfläche:

- Es sind weniger Komponenten enthalten, die gehärtet werden müssen
- Durch die geringere Anzahl an Bestandteilen ist wahrscheinlich die Angriffsfläche kleiner
- Weniger Administratoren werden sich auf die Server Core mit RDP „verirren“ und somit auch weniger „Footprints“ für Hacker hinterlassen. (Mal ehrlich: jede Anmeldung hinterlässt per default ein Profil und Anmeldeinformationen im Cache!)

Das klingt nicht schlecht. Doch warum gibt es dann so wenige Server Core in den Infrastrukturen? Auch das ist einfach erklärt:

- Im Falle eines Troubleshooting fehlen die gewohnten Tools auf dem Betriebssystem
- Nicht alle Anwendungen unterstützen die Installation auf einem Server Core

Mit Windows Server 2012 konnte man noch die Verwaltungskomponenten auf einem Server Core installieren und somit zumindest das erste Problem etwas entschärfen. Doch leider hat Microsoft diese „Mini-Shell“-Variante mit Windows Server 2016 nicht fortgesetzt. Und so gilt: Einmal Windows Server 2016 Server Core – immer Windows Server 2016 Server Core!

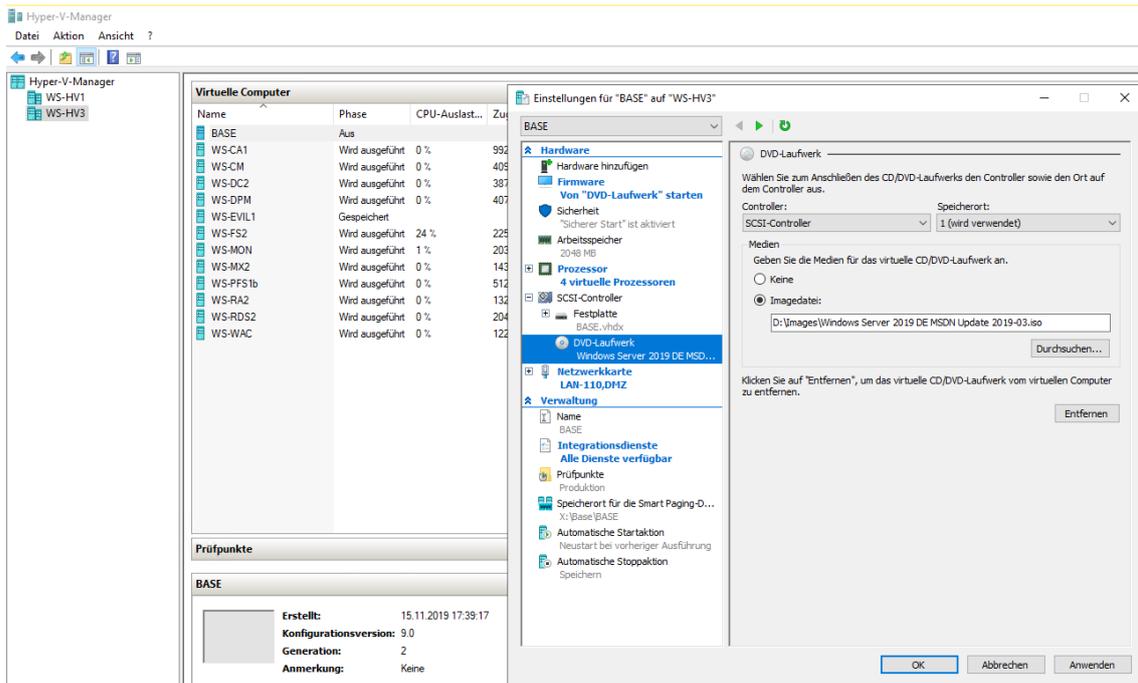
Mit Windows Server 2019 wurde eine Neuerung eingeführt: Die „ServerCore-AppCompatibility“. Mit diesem Feature On Demand werden wieder einige zusätzliche Werkzeuge auf dem Server Core platziert. Zusätzlich arbeitet Microsoft immer weiter am Anwendungssupport. So unterstützen beispielsweise Exchange Server 2019 und SQL Server 2017+ die Installation ohne grafische Oberfläche.

Und so wird es Zeit, diese Installationsvariante erneut zu evaluieren!

Im Folgenden zeige ich, wie eine Installation mit Server Core um die Zusatzfunktion erweitert wird und wie daraus ein Basis-Image für eine Installation im Hyper-V bereitgestellt wird.

Installation eines Windows Server 2019 Server Core

In meinem Hyper-V-Host erstelle ich eine passende VM für die Installation:



The screenshot shows the Hyper-V Manager interface. On the left, a list of virtual machines is visible, including 'BASE', 'WS-CA1', 'WS-CM', 'WS-DC2', 'WS-DPM', 'WS-EVIL1', 'WS-FS2', 'WS-MON', 'WS-MX2', 'WS-PFS1b', 'WS-RA2', 'WS-RDS2', and 'WS-WAC'. The 'Einstellungen für "BASE" auf "WS-HV3"' window is open, showing the configuration for the 'BASE' VM. The 'Hardware' tab is selected, showing 4 virtual processors and a DVD drive. The 'DVD-Laufwerk' settings are open, showing the ISO file path: D:\images\Windows Server 2019 DE MSDN Update 2019-03.iso.

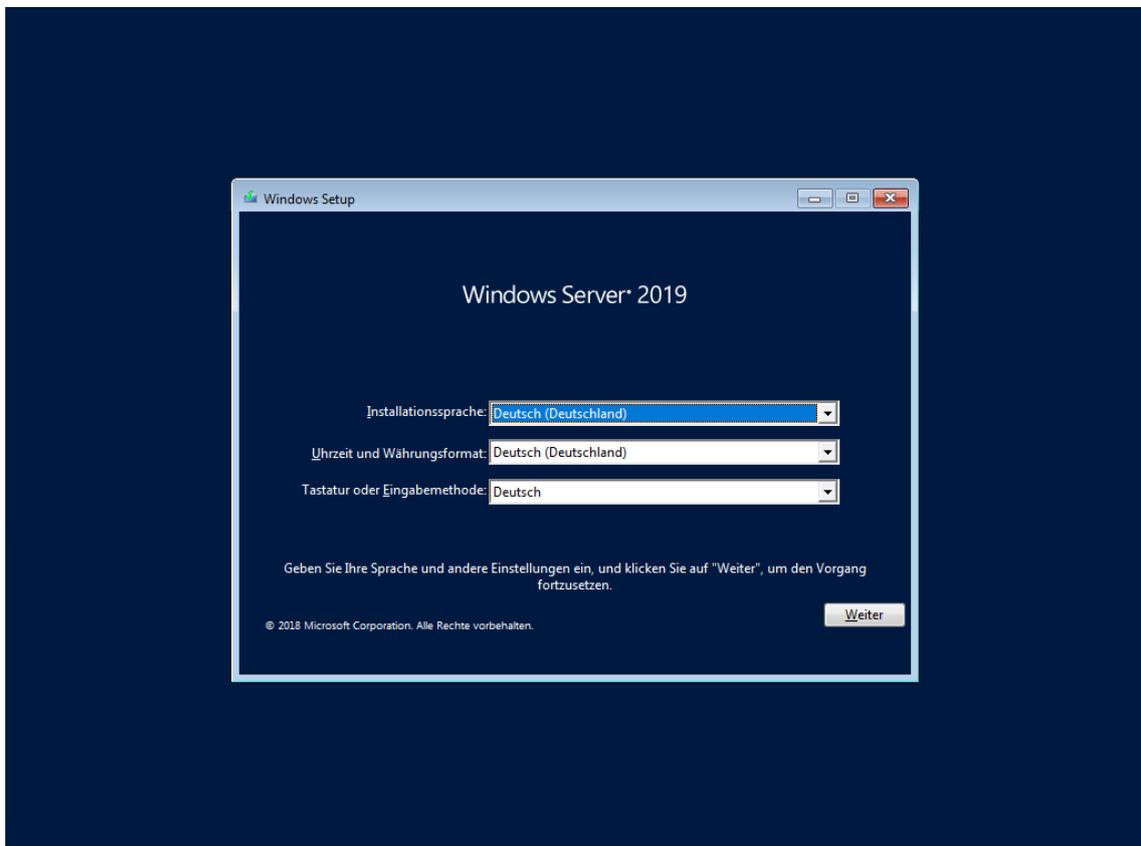
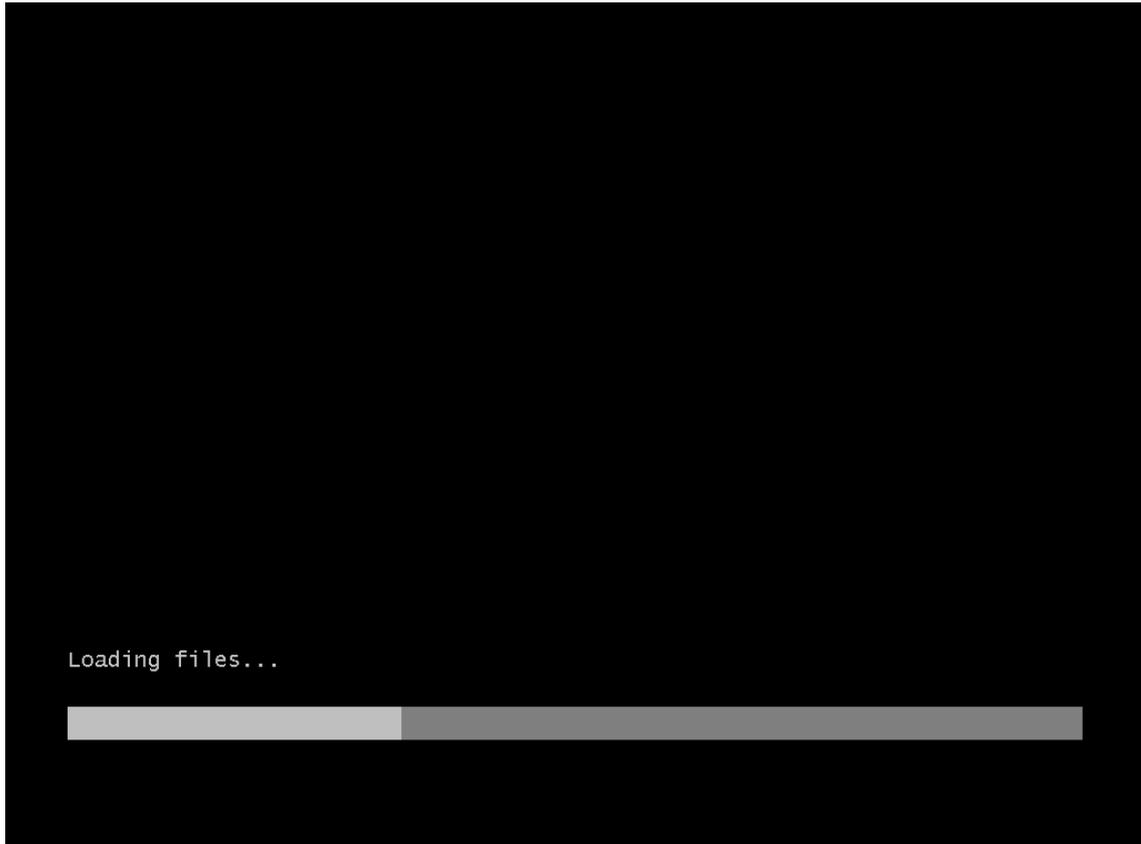
Name	Phase	CPU-Auslast...	Zu...
BASE	Aus		
WS-CA1	Wird ausgeführt	0 %	952
WS-CM	Wird ausgeführt	0 %	408
WS-DC2	Wird ausgeführt	0 %	387
WS-DPM	Wird ausgeführt	0 %	407
WS-EVIL1	Gespeichert		
WS-FS2	Wird ausgeführt	24 %	228
WS-MON	Wird ausgeführt	1 %	203
WS-MX2	Wird ausgeführt	0 %	143
WS-PFS1b	Wird ausgeführt	0 %	512
WS-RA2	Wird ausgeführt	0 %	132
WS-RDS2	Wird ausgeführt	0 %	204
WS-WAC	Wird ausgeführt	0 %	122

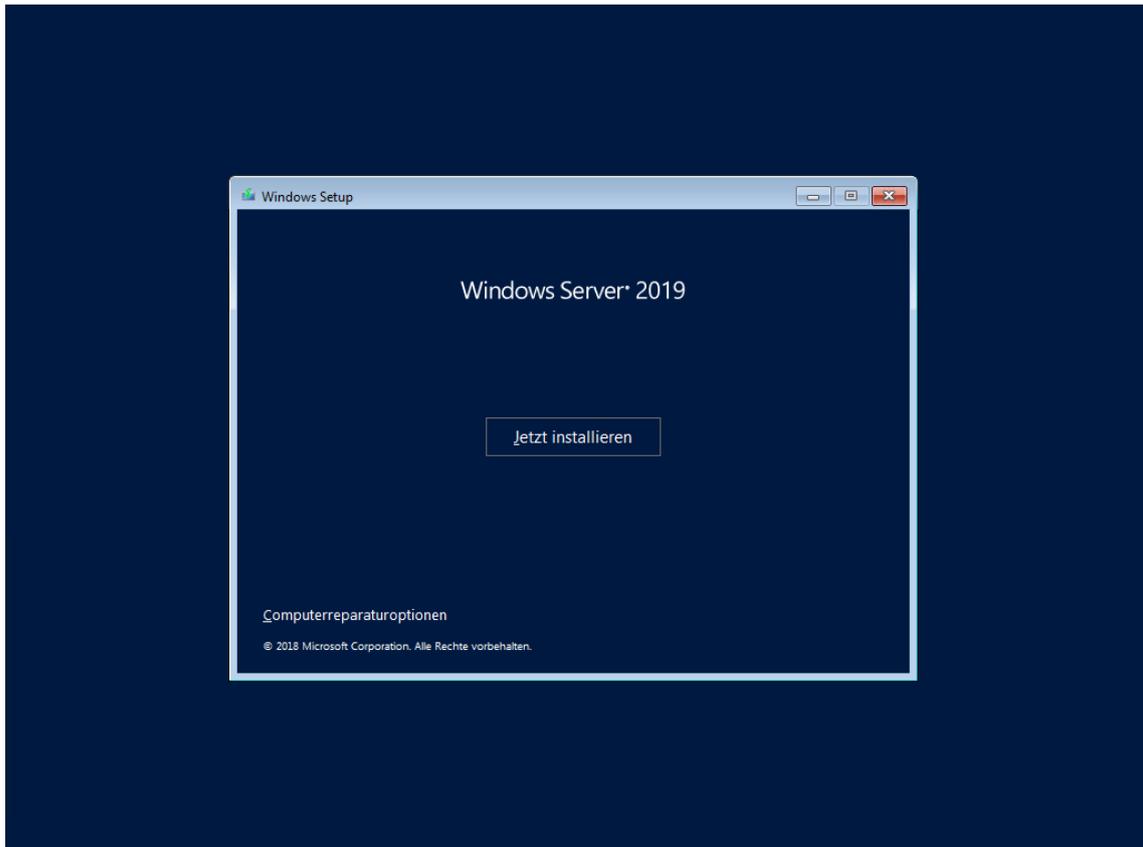
Prüfpunkte

BASE

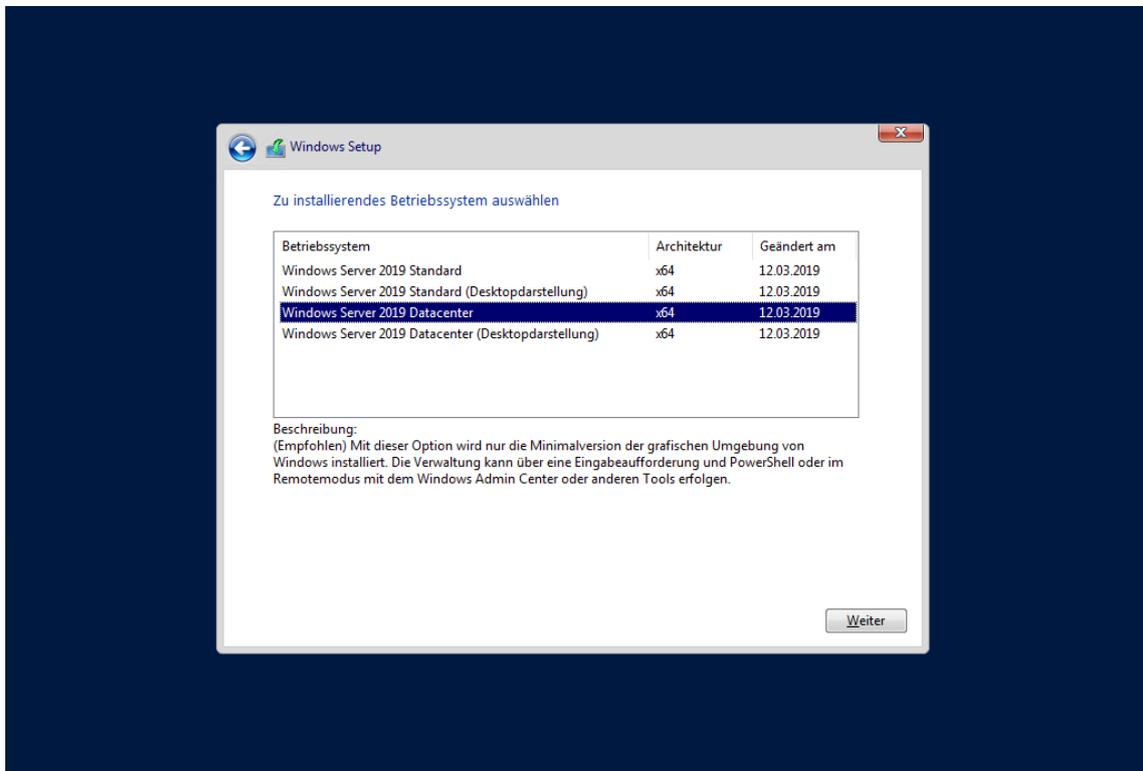
Erstellt: 15.11.2019 17:39:17
 Konfigurationsversion: 9.0
 Generation: 2
 Anmerkung: Keine

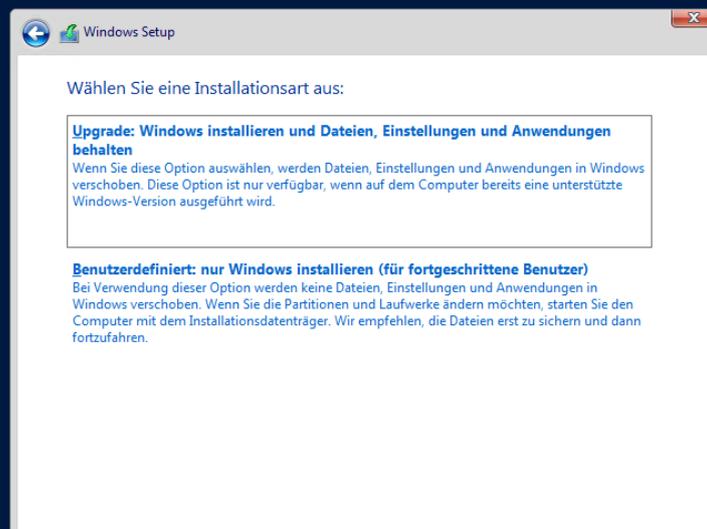
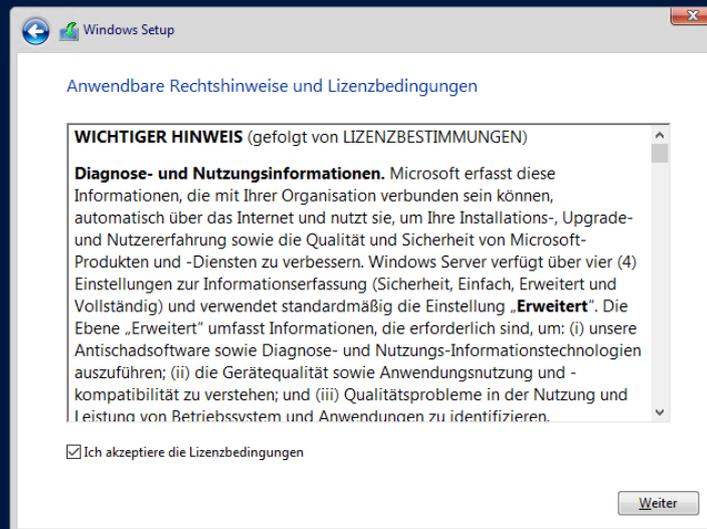
Das Setup ist seit Jahren vom Prozess her unverändert und stellt somit keine Herausforderung dar:

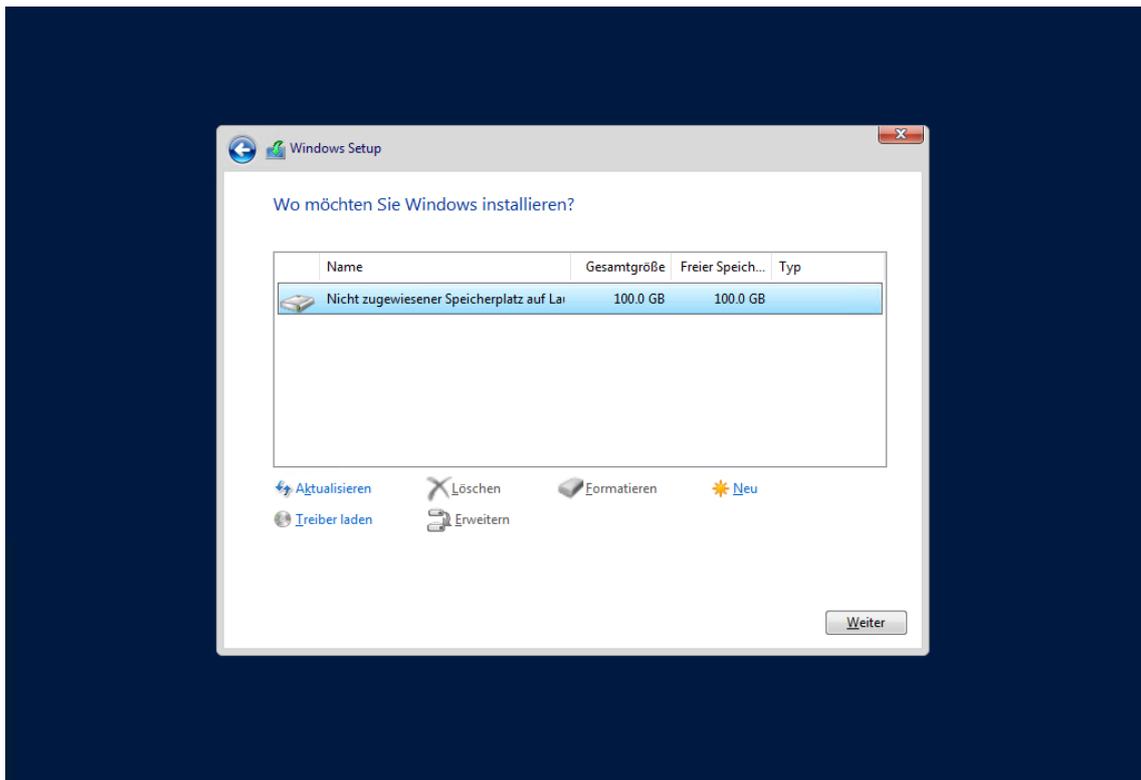




Der Standard für die Bereitstellung ist bereits Server Core – auch wenn das so nicht explizit genannt wird:

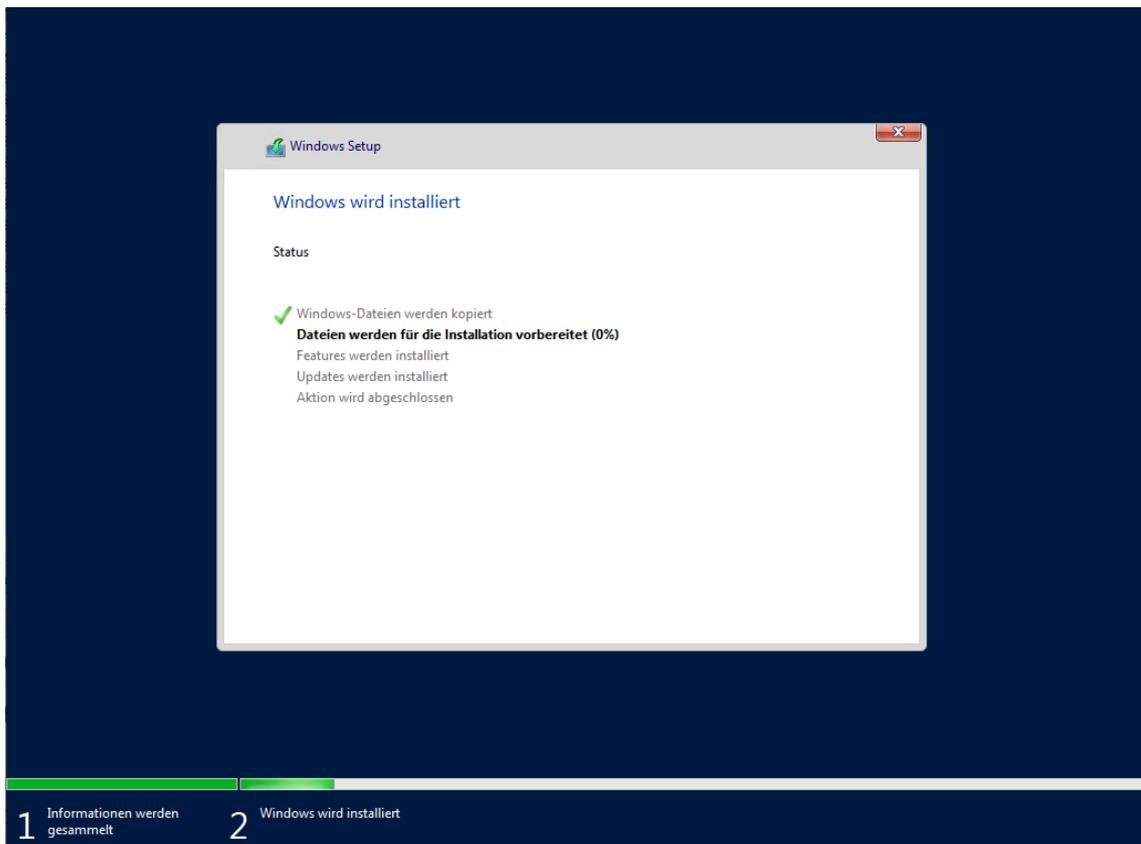






1 Informationen werden gesammelt

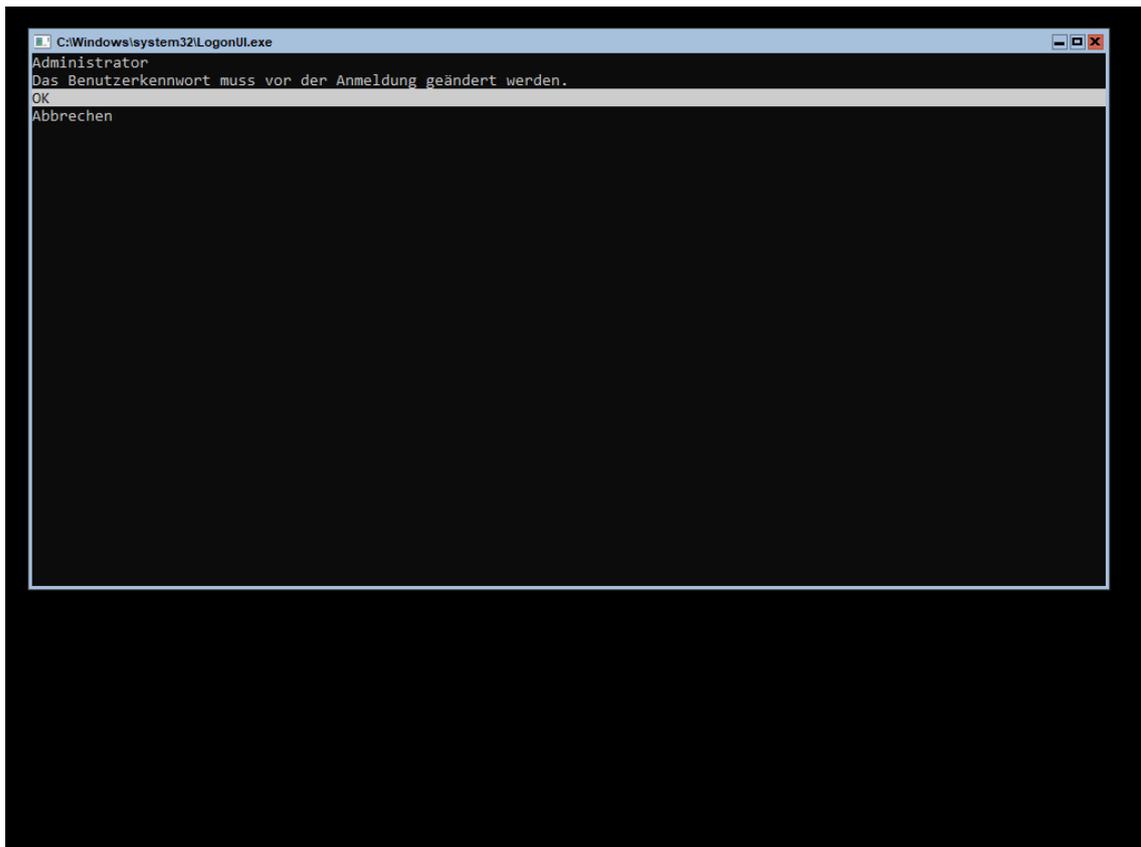
2 Windows wird installiert



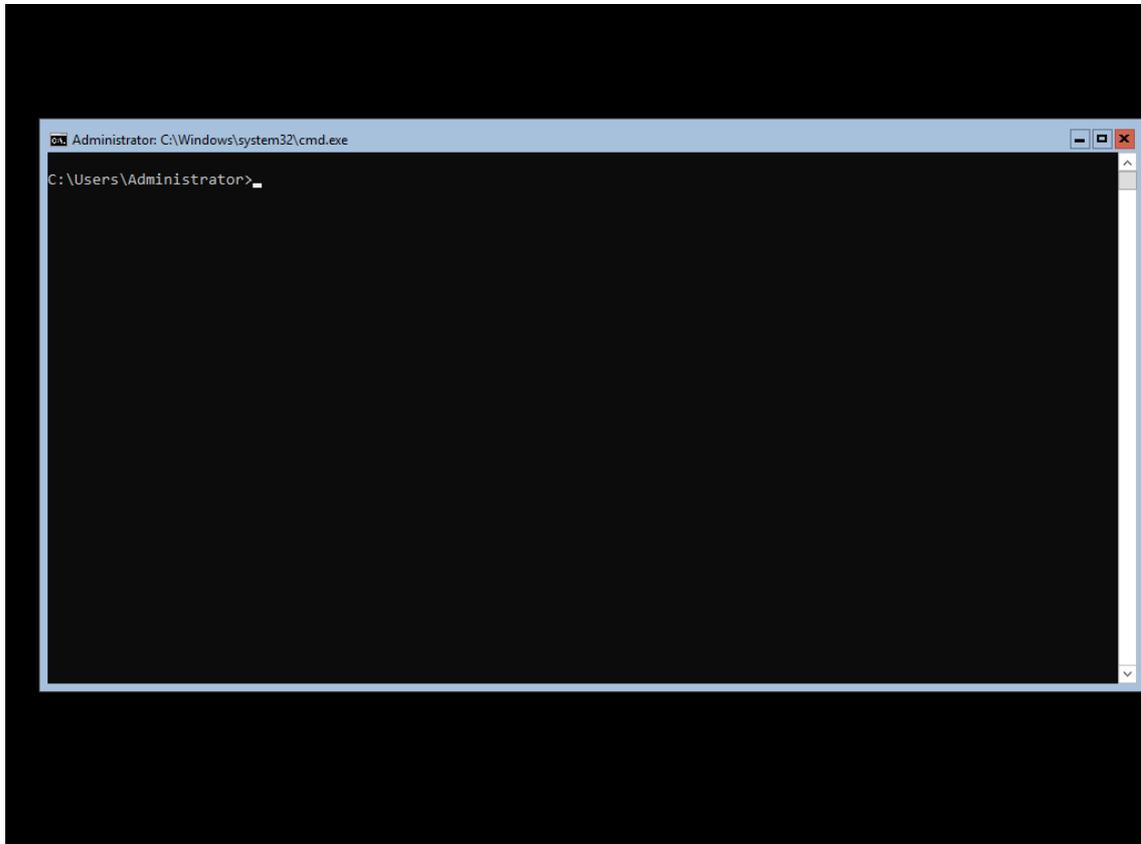
1 Informationen werden gesammelt

2 Windows wird installiert

Bis hier wars nichts Neues. Nach der Bereitstellung und der Treibererkennung kommt hier das verknappte Out-Of-Box-Experience (OOBE) des Server Core ums Eck. Dabei kann mit dem Tabulator navigiert werden:

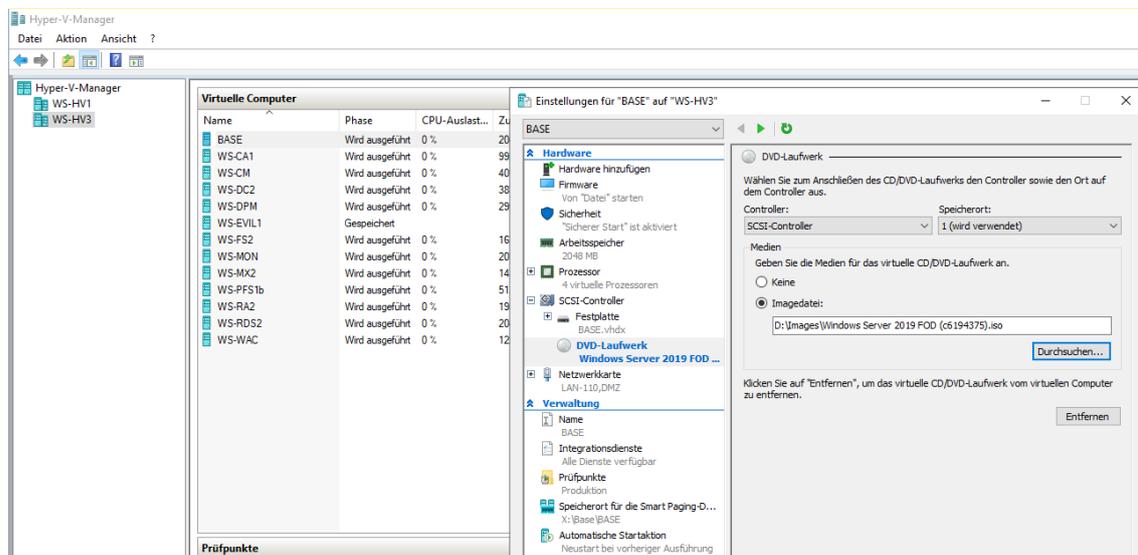


Und dann ist die Anmeldung auch schon erledigt. Wie gewohnt präsentiert sich der Prompt der cmd – dem Fenster ohne Windows Explorer:

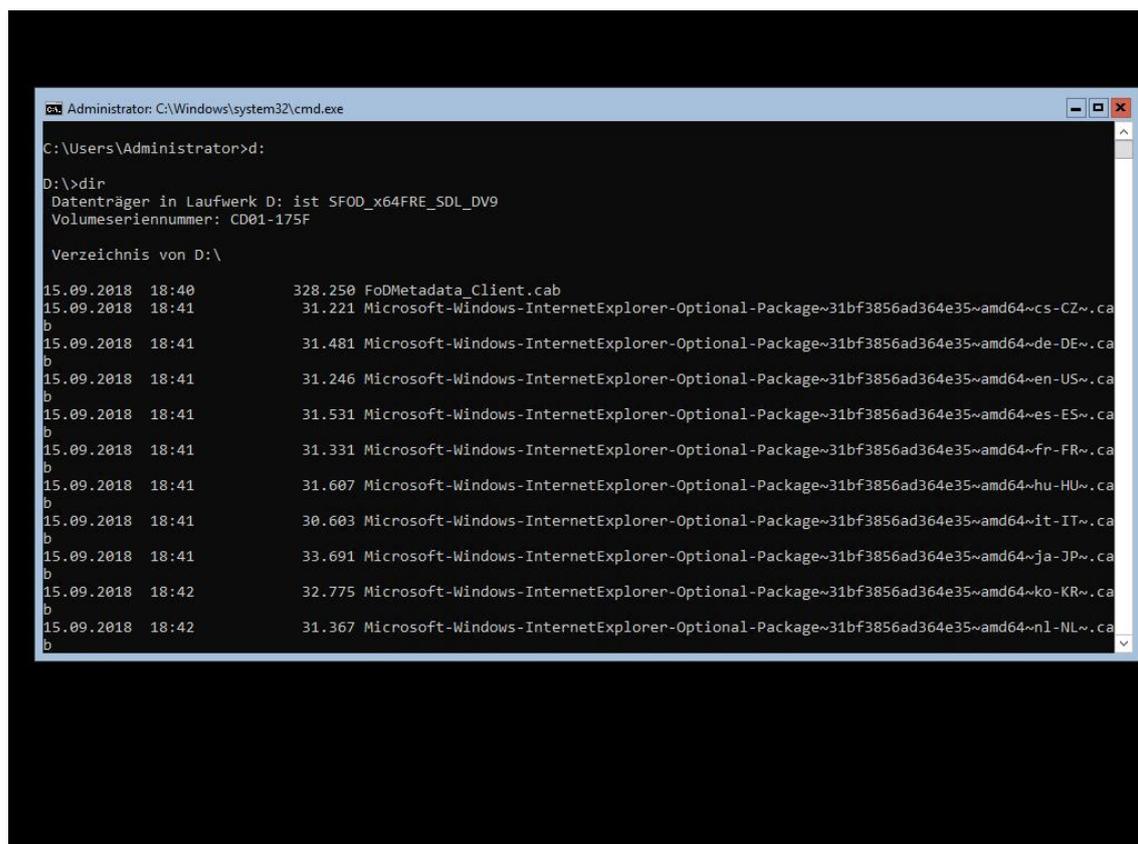


Integration des Feature on Demand „AppCompatibility“

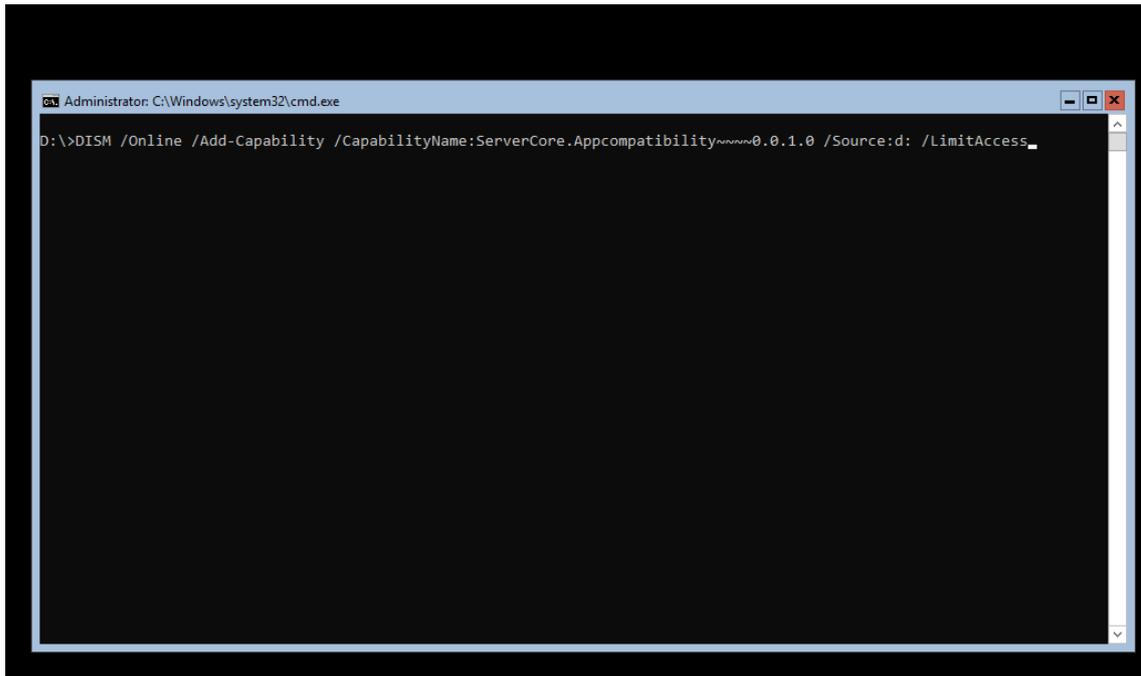
Nun installiere ich die Erweiterung. Diese stellt Microsoft in einer etwa 350MB großen ISO-Datei zur Verfügung. Die lege ich in das virtuelle DVD-Laufwerk ein:



Dort angekommen riskiere ich einen Blick über die cmd. Es befinden sich einige „Feature on Demand“ (FOD)-Ordner darauf:



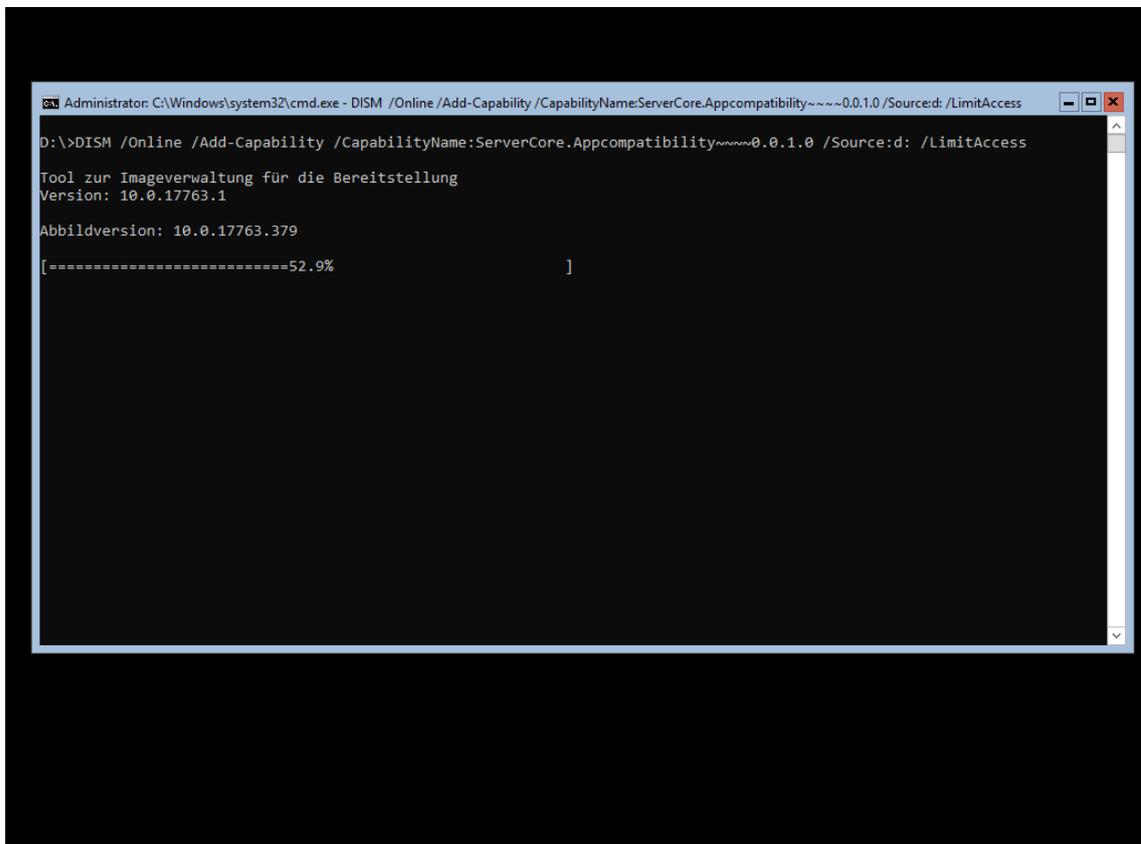
Die Integration wird mit DISM in das bereits laufende Betriebssystem gestartet. Damit wäre auch eine Offline-Erweiterung ohne Installation des Betriebssystems denkbar:



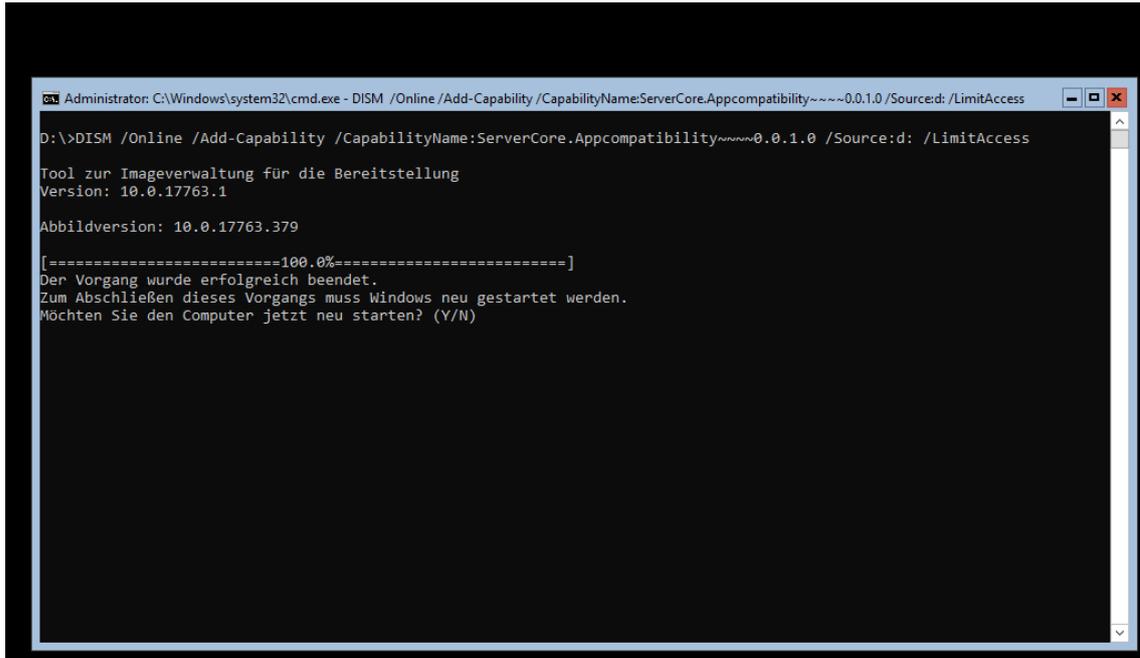
Der Befehl ist etwas gewöhnungsbedürftig:

```
DISM /Online /Add-Capability /CapabilityName:ServerCore.Appcompatibility~~~~0.0.1.0 /Source:d: /LimitAccess
```

Die Installation benötigt einige Minuten:



Und natürlich wird dieser tiefe Eingriff nicht ohne Neustart aktiv:



```
Administrator: C:\Windows\system32\cmd.exe - DISM /Online /Add-Capability /CapabilityName:ServerCore.Appcompatibility~~~~0.0.1.0 /Source:d: /LimitAccess

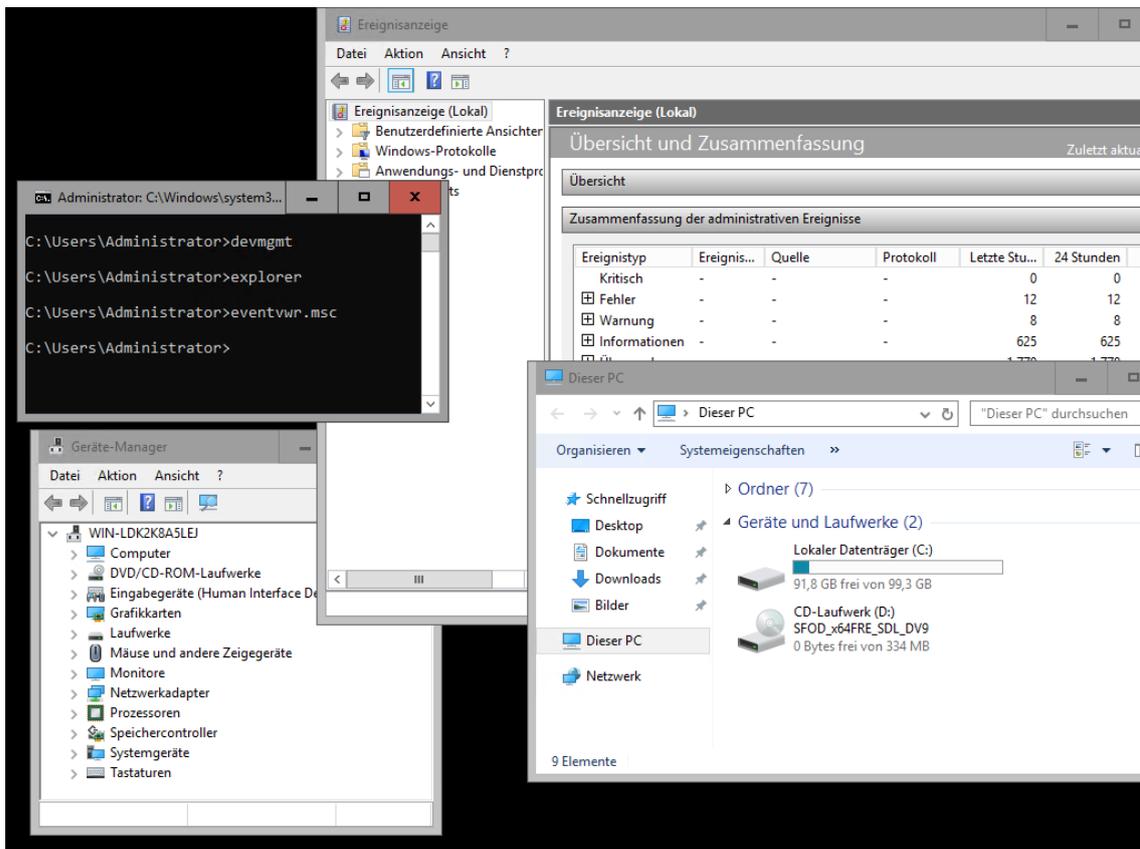
D:\>DISM /Online /Add-Capability /CapabilityName:ServerCore.Appcompatibility~~~~0.0.1.0 /Source:d: /LimitAccess

Tool zur Imageverwaltung für die Bereitstellung
Version: 10.0.17763.1

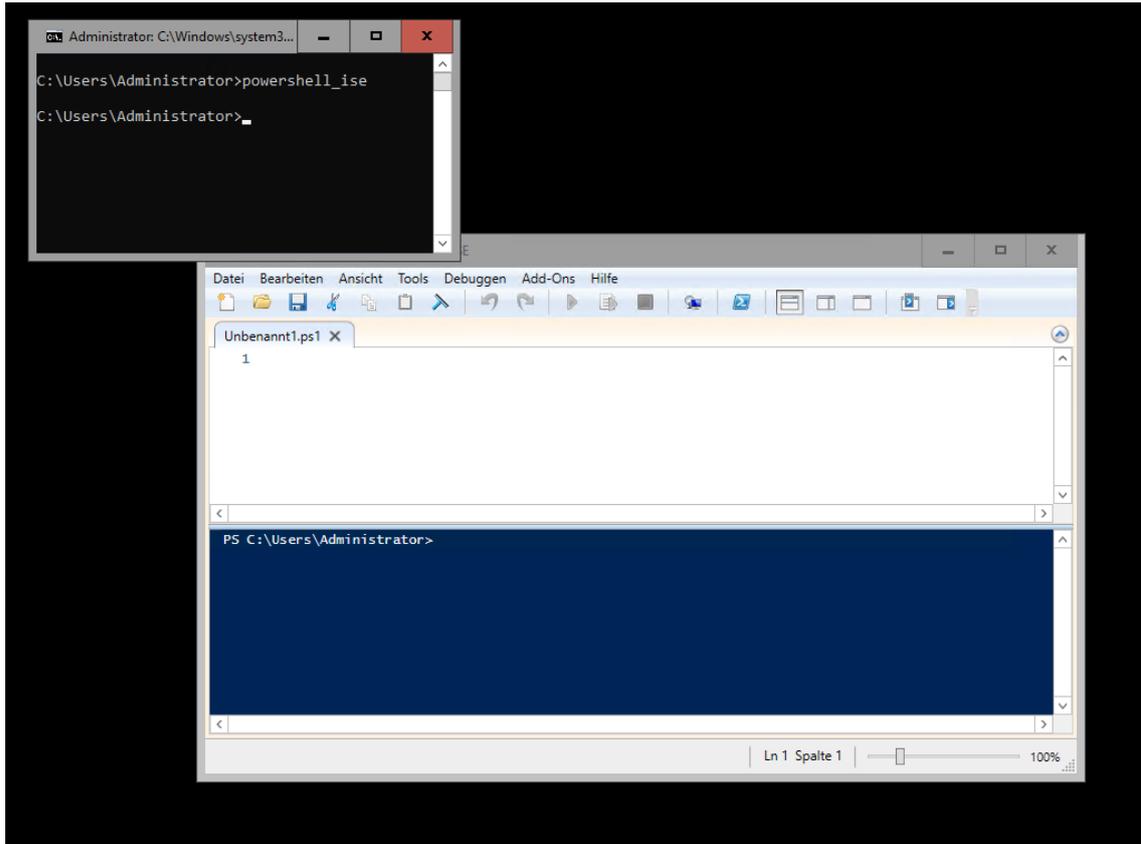
Abbildversion: 10.0.17763.379

[=====100.0%=====]
Der Vorgang wurde erfolgreich beendet.
Zum Abschließen dieses Vorgangs muss Windows neu gestartet werden.
Möchten Sie den Computer jetzt neu starten? (Y/N)
```

Nach dem Neustart und der Anmeldung teste ich einige Tools, indem ich diese über die cmd direkt aufrufe. Die für mich stets wichtige Ereignisanzeige (eventvwr.msc) ist mit dabei. Ebenso zeigt sich der Gerätemanager (devmgmt). Zu meiner Überraschung hat Microsoft auch einen Windows Explorer integriert. Damit entfällt der übliche Umweg über den OpenDialog des Notepads.



Sehr zu meiner Freude ist auch die PowerShell-ISE dabei! Damit lassen sich lokale Scripte bequem editieren und ausführen:



Vorbereitung der Generalisierung

Natürlich gibt es wie in den Vorgängerversionen das bekannte Script sconfig. Mit diesem lassen sich wichtige, administrative Aufgaben steuern, wie z.B. das Suchen und Installieren von Windows Updates:

```

Administrator: C:\Windows\system32\cmd.exe - sconfig
1) Domäne/Arbeitsgruppe:           Arbeitsgruppe:  WORKGROUP
2) Computername:                   WIN-LDK2K8A5LEJ
3) Lokalen Administrator hinzufügen
4) Remoteverwaltung konfigurieren   Aktiviert

5) Windows Update-Einstellungen:   Nur Downloads
6) Updates herunterladen u. installieren
7) Remotedesktop:                   Deaktiviert

8) Netzwerkeinstell.
9) Datum und Uhrzeit
10) Telemetrieinstellungen          Unbekannt
11) Windows-Aktivierung

12) Benutzer abmelden
13) Server neu starten
14) Server herunterfahren
15) Zur Befehlszeile wechseln

Geben Sie eine Zahl ein, um eine Option auszuwählen: 6

=====
                          Serverkonfiguration
=====
C:\Windows\System32\cscript.exe
1) Microsoft (R) Windows Script Host, Version 5.812
2) Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.
3)
4) Nach (a)llen oder nur nach (e)mpfohlenen Updates suchen? a
5) Alle geeigneten Updates werden gesucht...
6)
7)
8)
9)
10)
11) Windows-Aktivierung

12) Benutzer abmelden
13) Server neu starten
14) Server herunterfahren
15) Zur Befehlszeile wechseln

Geben Sie eine Zahl ein, um eine Option auszuwählen:
  
```

Damit mein Basis-Image möglichst aktuell ist, installiere ich alle möglichen Updates. Eigentlich ist so etwas heute nicht mehr nötig, da die Updates üblicherweise kumulativ sind: Es muss also abgesehen von Servicing Stack Updates nur das aktuellste installiert werden. Aber schaden kann es auch nicht:

```

Administrator: C:\Windows\system32\cmd.exe - sconfig
1) Domäne/Arbeitsgruppe:           Arbeitsgruppe:  WORKGROUP
2) Computername:                   WIN-LDK2K8A5LEJ
3) Lokalen Administrator hinzufügen
4) Remoteverwaltung konfigurieren   Aktiviert

5) Windows Update-Einstellungen:   Nur Downloads
6) Updates herunterladen u. installieren
7) Remotedesktop:                   Deaktiviert

8) Netzwerkeinstell.
9) Datum und Uhrzeit
10) Telemetrieinstellungen          Unbekannt
11) Windows-Aktivierung

12) Benutzer abmelden
13) Server neu starten
14) Server herunterfahren
15) Zur Befehlszeile wechseln

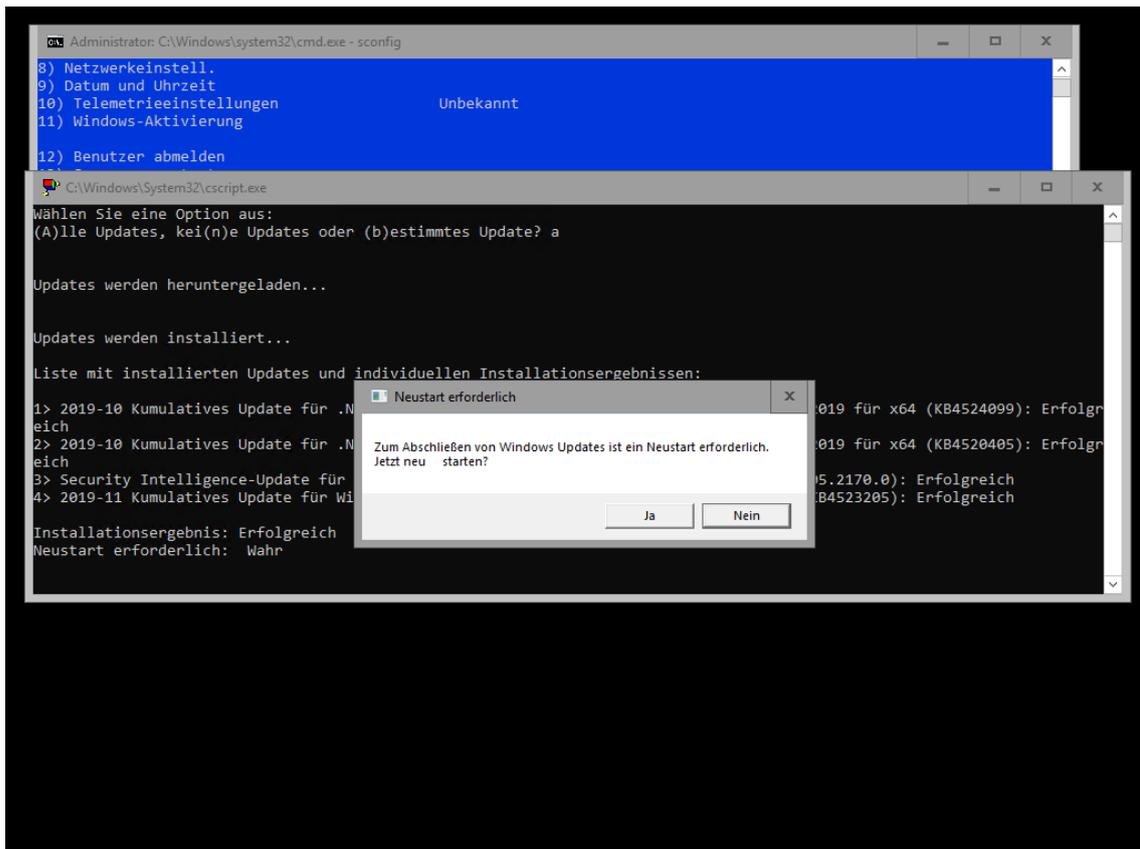
Geben Sie eine Zahl ein, um eine Option auszuwählen:

C:\Windows\System32\cscript.exe
1) Microsoft (R) Windows Script Host, Version 5.812
2) Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.
3)
4) Nach (a)llen oder nur nach (e)mpfohlenen Updates suchen? a
5) Alle geeigneten Updates werden gesucht...
6)
7)
8)
9)
10)
11) Liste geeigneter Elemente auf dem Computer:
12)
13)
14)
15)
Ge 1> 2019-10 Kumulatives Update für .NET Framework 3.5, 4.7.2 und 4.8 für Windows Server 2019 für x64 (KB4524099)
    2> 2019-10 Kumulatives Update für .NET Framework 3.5, 4.7.2 und 4.8 für Windows Server 2019 für x64 (KB4520405)
    3> Security Intelligence-Update für Windows Defender Antivirus - KB2267602 (Version 1.305.2170.0)
    4> 2019-11 Kumulatives Update für Windows Server 2019 (1809) für x64-basierte Systeme (KB4523205)
==
== Wählen Sie eine Option aus:
    (A)lle Updates, kei(n)e Updates oder (b)estimmtes Update? a
1)
2)
3) Updates werden heruntergeladen...
4)
5)
6) Updates herunterladen u. installieren
7) Remotedesktop:                   Deaktiviert

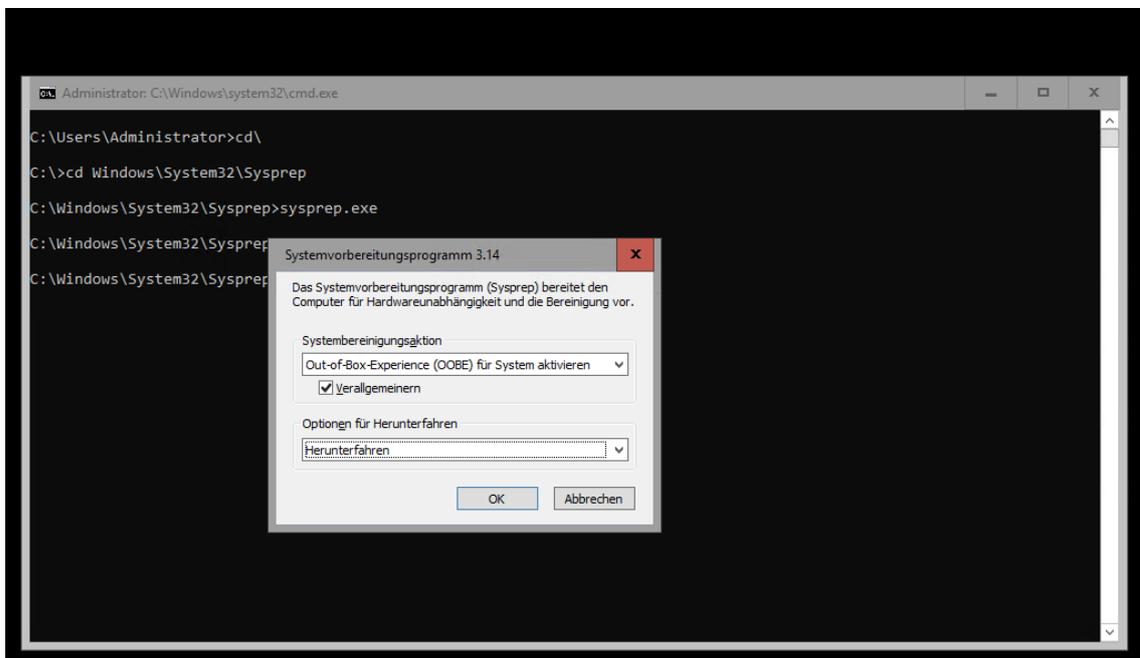
8) Netzwerkeinstell.
9) Datum und Uhrzeit
10) Telemetrieinstellungen          Unbekannt
11) Windows-Aktivierung

12) Benutzer abmelden
13) Server neu starten
14) Server herunterfahren
15) Zur Befehlszeile wechseln

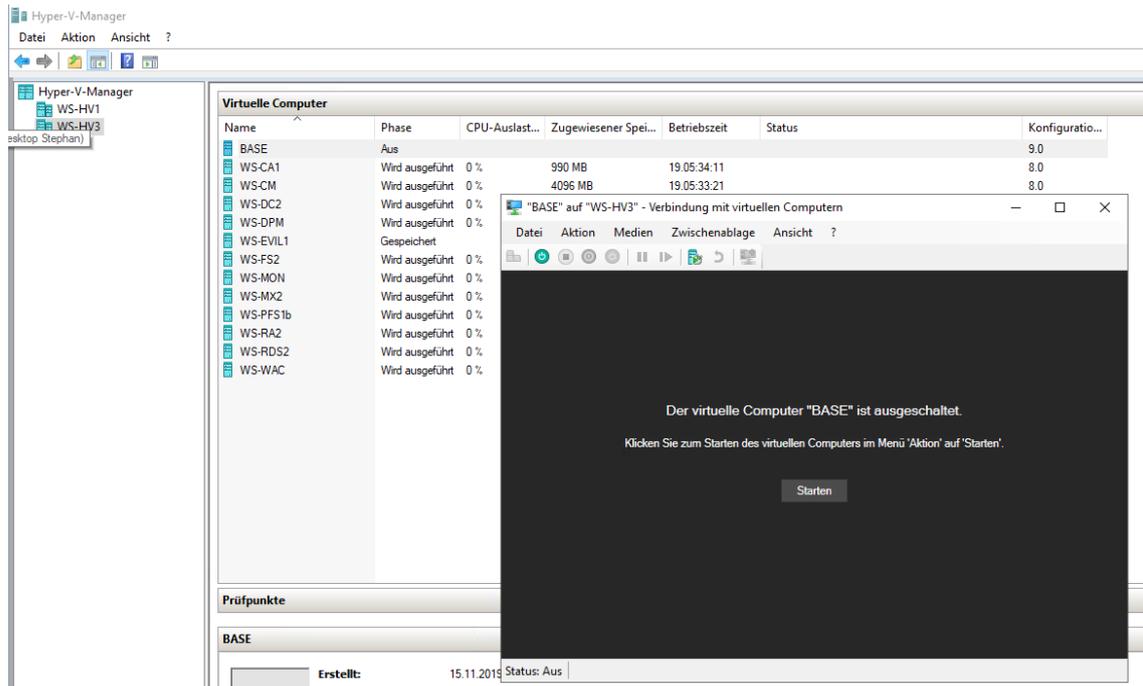
Geben Sie eine Zahl ein, um eine Option auszuwählen:
  
```



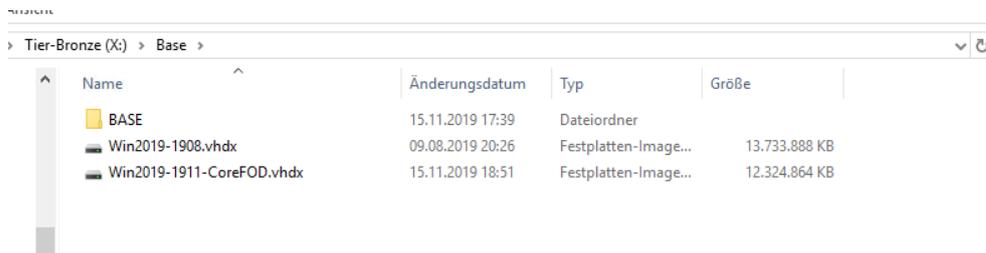
Nach dem Neustart und einer Anmeldung generalisiere ich nun den Server. Dabei werden alle individuellen Informationen entfernt und danach kann der Server vervielfältigt werden:



Ein paar Minuten später ist der Server wie gewünscht heruntergefahren:

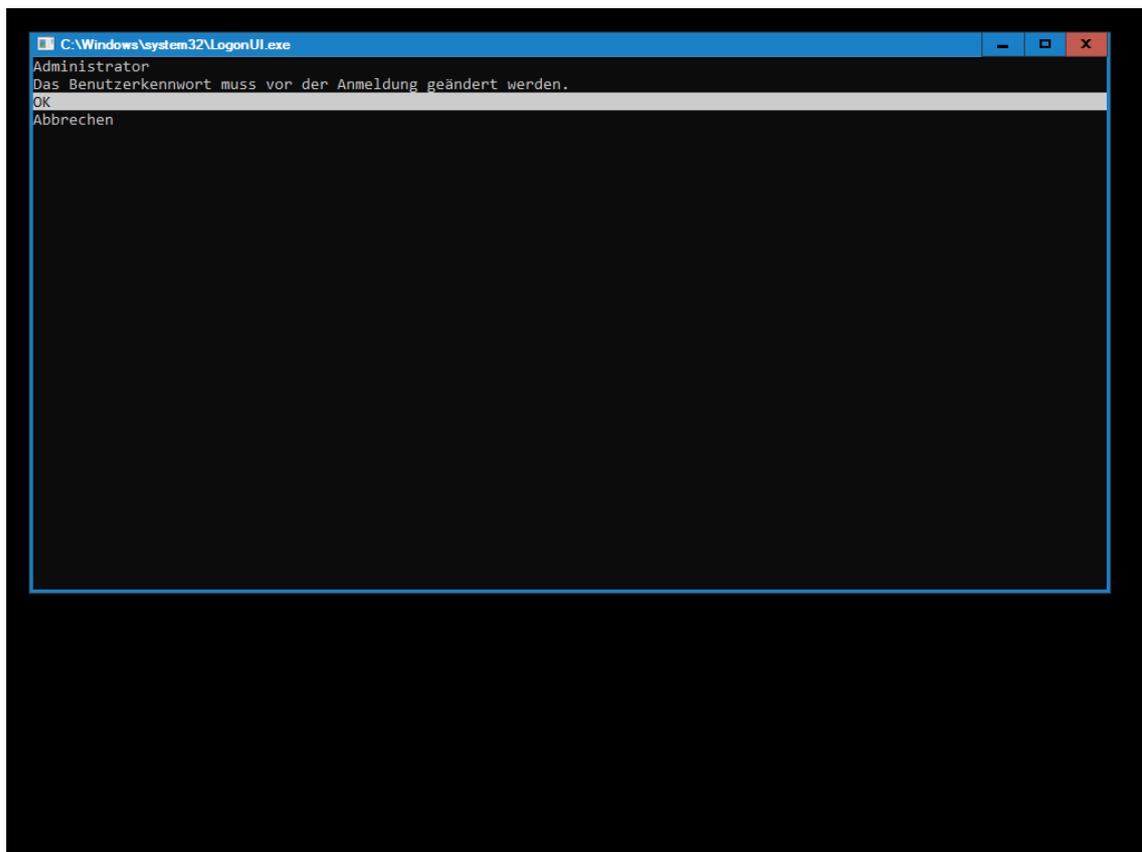
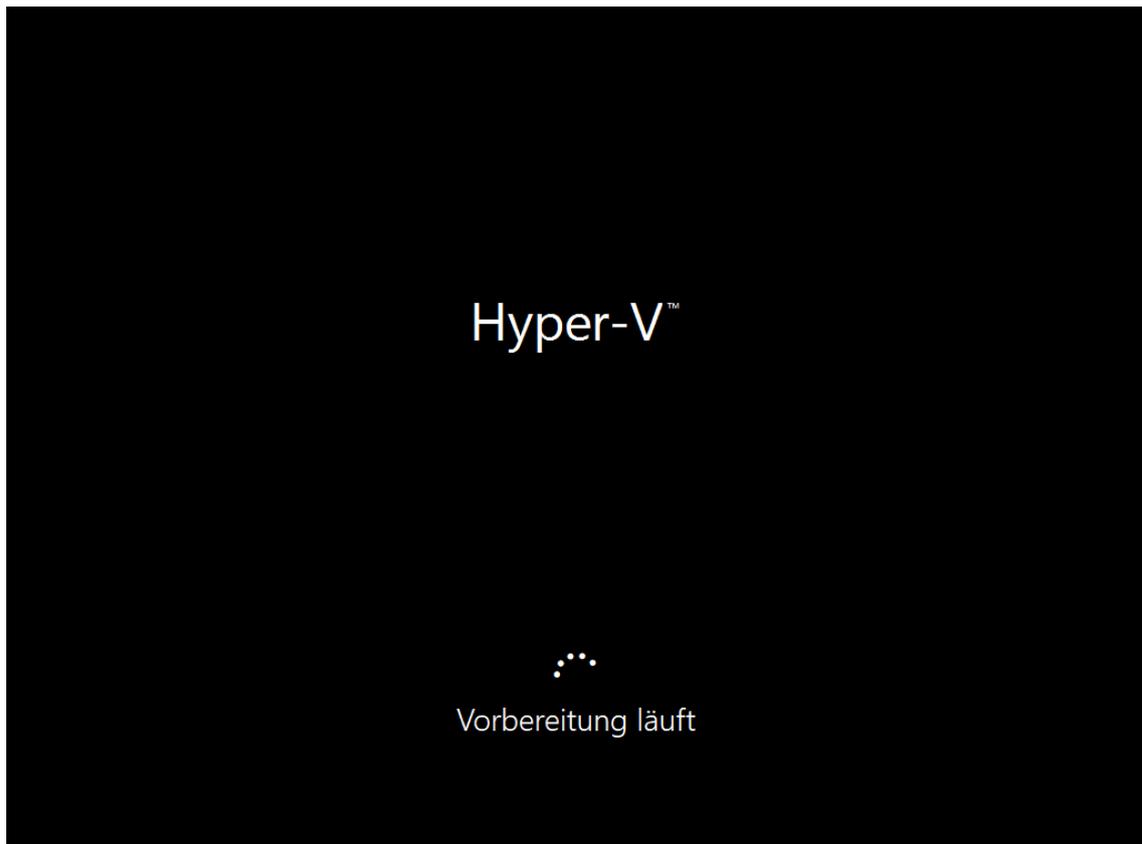


Die VHDX-Datei enthält jetzt eine Betriebssystemversion mit dem aktuellen Patchlevel eines Windows Server 2019 Server Core, in dem auch die Erweiterung „AppCompatibility“ installiert ist. Die Datei kopiere ich in mein Repository. Dabei sieht man, dass die Bereitstellung i Modell Thin-Provision nahezu genauso viel Speicher belegt, wie ein Server mit Desktop Experience. Das war in den früheren Versionen noch deutlich anders:

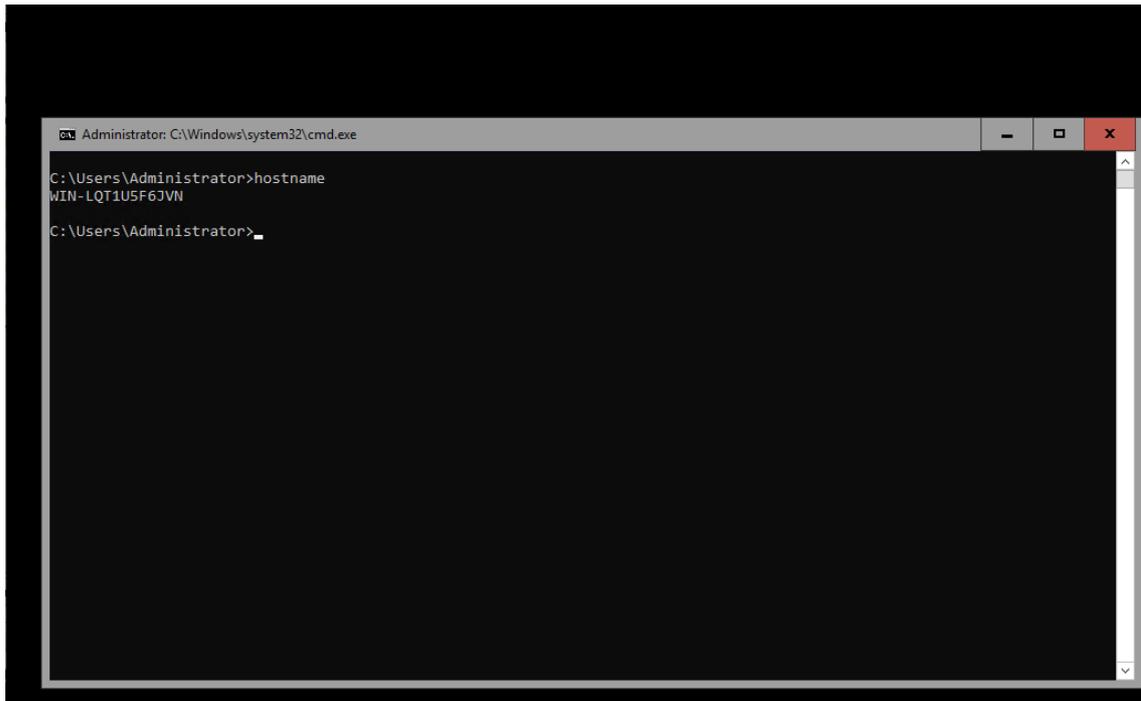


Testlauf der Bereitstellung

Für einen Test schalte ich die VM noch einmal ein. Dabei wird das „neue“ Betriebssystem eine Hardwareerkennung ausführen und die Ersteinrichtung (OOBE) vornehmen:



Nach dem Erstellen eines Adminpasswortes prüfe ich den Rechnernamen. Dieser wurde erwartungsgemäß neu vergeben:



```
Administrator: C:\Windows\system32\cmd.exe
C:\Users\Administrator>hostname
WIN-LQT1U5F6JVN
C:\Users\Administrator>
```

Die VM schalte ich aus und entferne sie aus meiner Hyper-V-Konfiguration. Mit dem neuen Basis-Image kann ich nun jederzeit weitere Server Core bereitstellen.