Inhalt

Hintergrundinformation

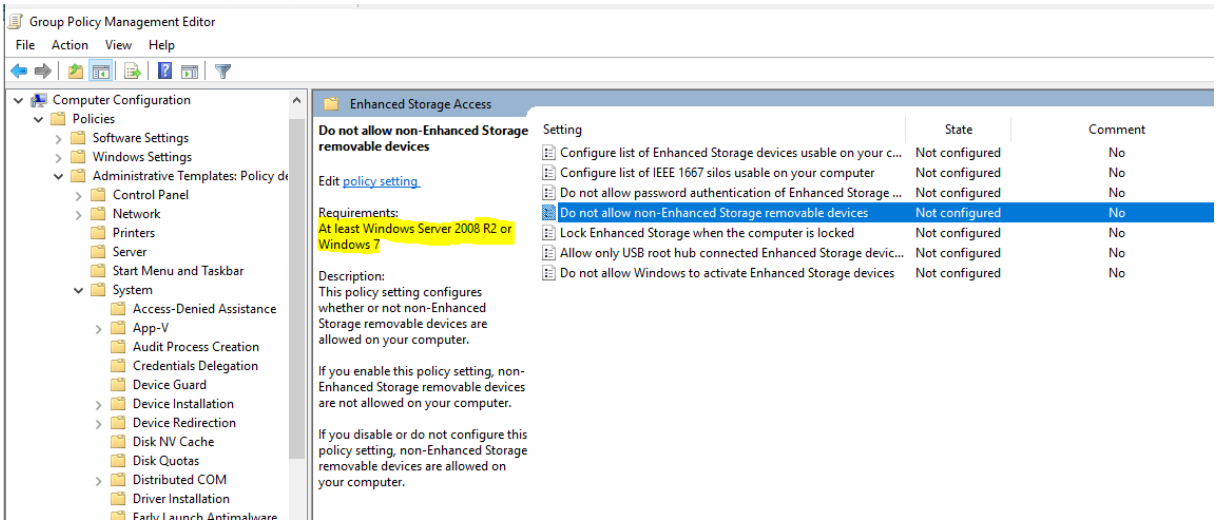
Ausgangssituation

Windows Administratoren können im Active Directory mit Gruppenrichtlinien zentral Einstellungen und Konfigurationen an eine Vielzahl von Clients und Benutzern automatisiert verteilen. Das Werkzeug ist mächtig und es wird seit Generationen von Betriebssystemen verwendet.

Wenn Microsoft ein neues Betriebssystem auf den Markt bringt, dann enthält dieses eigentlich immer Funktionen, welche die Vorgänger nicht hatten. Damit auch diese über Gruppenrichtlinien gesteuert werden können, gibt es mit jedem neuen Betriebssystem auch neue Versionen der Gruppenrichtlinien-Vorlagen. Diese sind als **ADMX** bekannt: **ADM**inistrative templates ne**X**t generation. Die Vorgänger waren ADM-Dateien. Es besteht eine gewisse Analogie zum Wechsel von *.doc auf *.docx.

Jede neue Generation von ADMX-Dateien sollte das aktuelle Betriebssystem und alle Vorgänger steuern können. Wir Administratoren mussten also nur die neusten Vorlagen in den richtigen Speicherplatz kopieren und schon konnten wir in der Management-Konsole für die Gruppenrichtlinien loslegen.

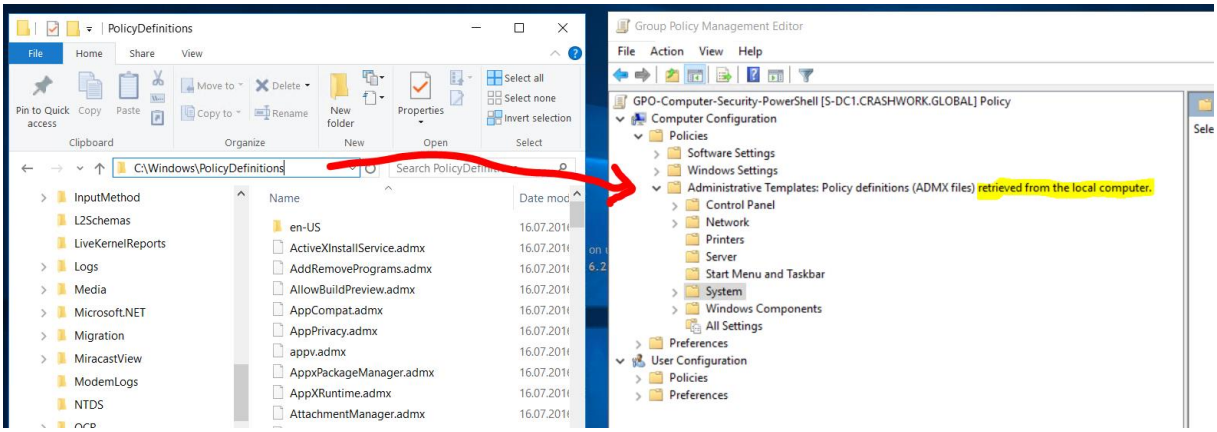
Beim Konfigurieren von Einstellungen mussten wir nur auf dieses Feld achten und alles wurde gut:



Das Prinzip wurde auch als „SuperSet“ bezeichnet. Ein schöner Gedanke an die vergangenen Tage!

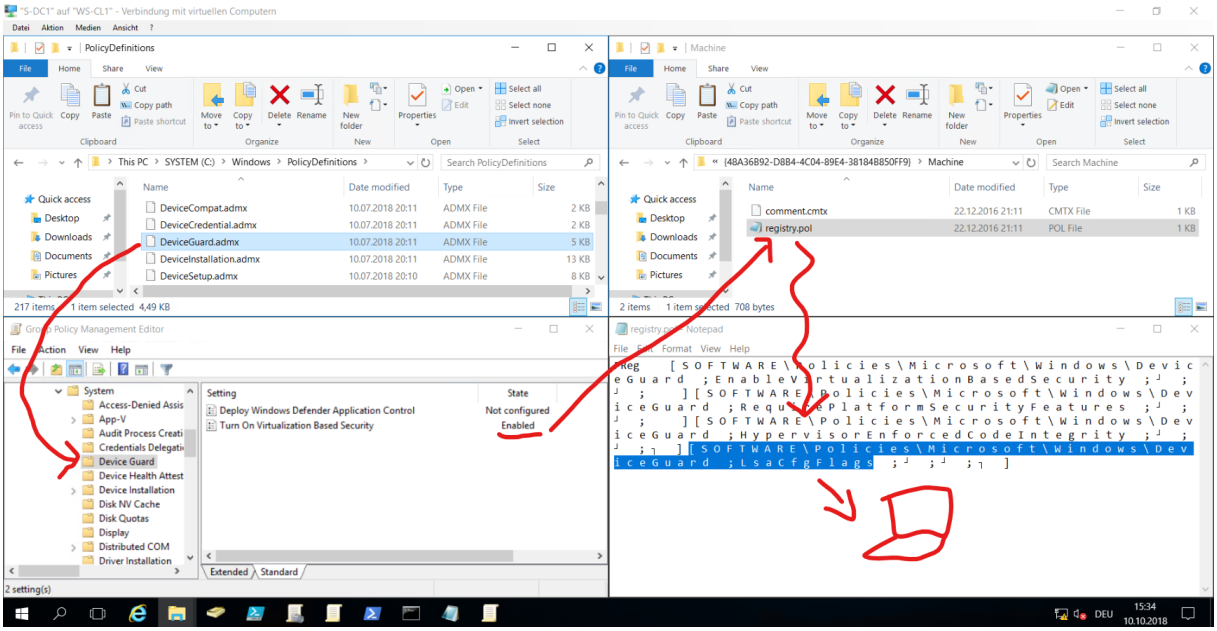
ADMX-Dateien und der Central Store

Wie funktionieren die ADMX-Vorlagen? Dazu habe ich in einem Gruppenrichtlinien-Workshop eine Zeichnung erstellt:



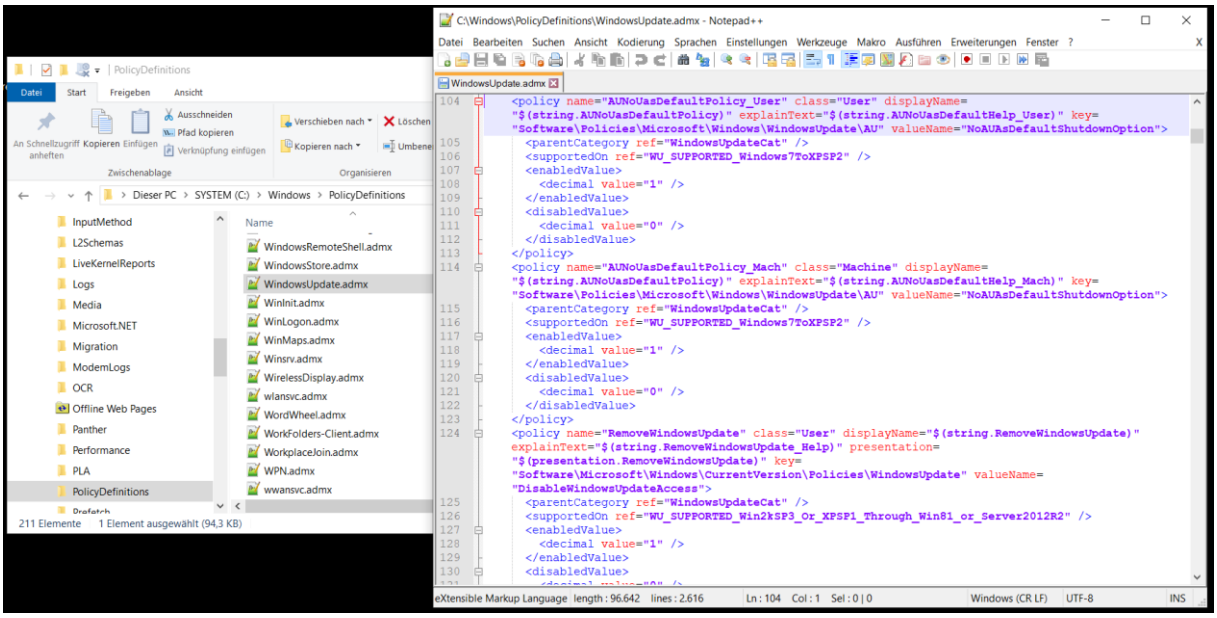
Die Vorlagen liegen immer im Verzeichnis C:\Windows\PolicyDefinitions. Öffnet man einen Editor für eine Gruppenrichtlinie (auch aus der Gruppenrichtlinienverwaltung für das Active Directory), dann werden die Vorlagen geladen und unter Computer|Benutzer\Policies\Administrative Templates eingebunden. In den ADMX sind dann die Anweisungen

enthalten, wie die Formulare im Editor auszusehen haben und welche Werte dann in die GPO geschrieben werden sollen. Das könnte so aussehen:

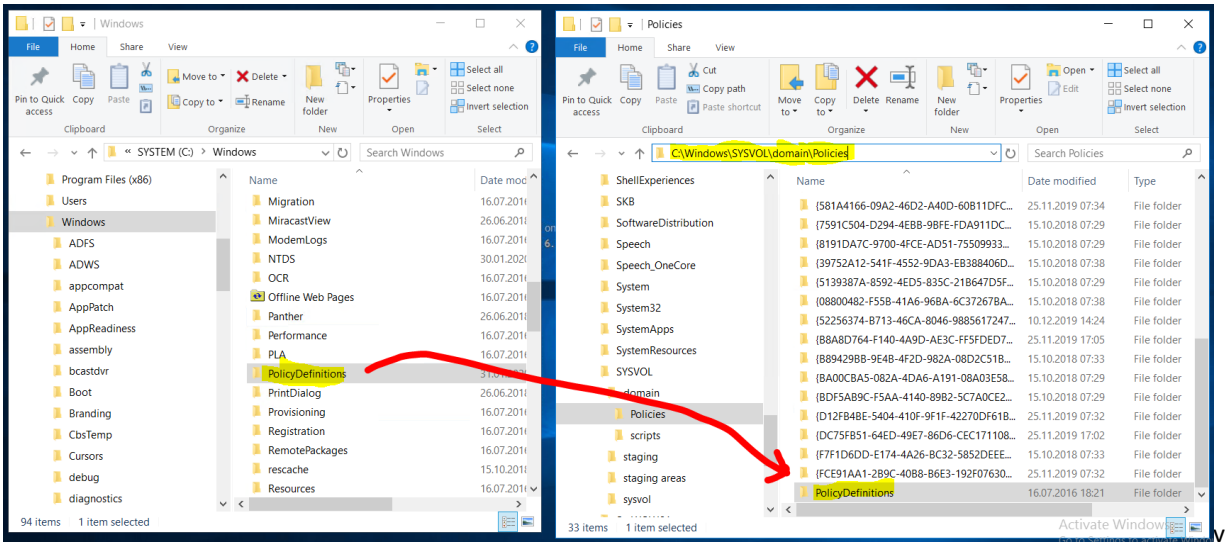


Ein Client läd nun die für ihn definierten Gruppenrichtlinien aus der SYSVOL-Freigabe als Dateien herunter und wendet sie lokal an.

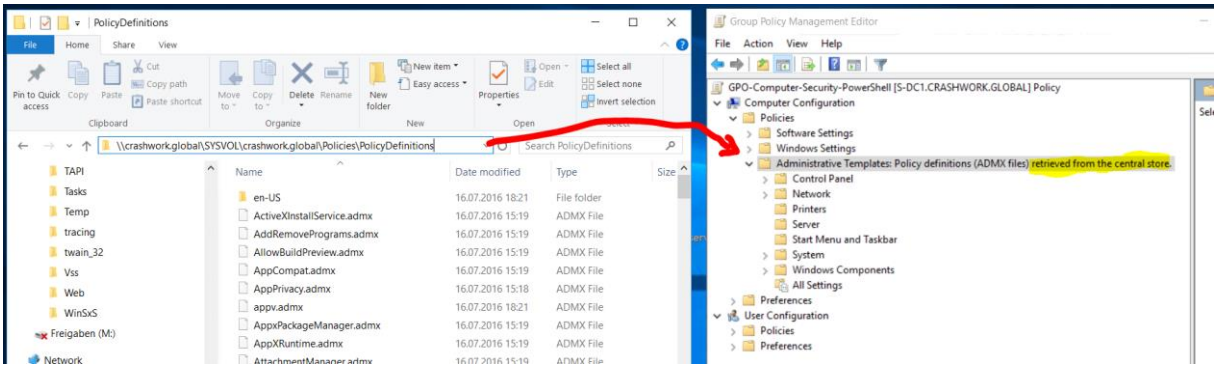
Die ADMX-Dateien sind für uns lesbar: Sie sind in einem XML-Format strukturiert:



Mit „Superset“ musste man immer nur die neusten ADMX-Vorlagen verwenden. Damit das in größeren Infrastrukturen kein Problem wird (es kann hier durchaus mehr als einen Editorrechner geben), kann man diese Dateien in ein spezielles Verzeichnis kopieren: Den Central Store. Dieser Speicherort kann im gleichen Verzeichnis wie die fertigen Gruppenrichtlinien erstellt werden:



Ab diesem Moment bezieht sich der Gruppenrichtlinieneditor immer auf den Central Store und verwendet die Vorlagen aus diesem Verzeichnis. Der Pfad wird dabei über die SYSVOL-Freigabe der Domain angesprochen:



Damit war das SuperSet sehr einfach umsetzbar.

Das Problem: Anzahl möglicher Betriebssysteme im Active Directory

In allen Windows Server Kursen von Microsoft und in fast allen Internet-Ressourcen gehen wir von diesem Idealfall des „SuperSet“ aus. Aber mit jedem weiteren Betriebssystem wurde es wohl für Microsoft komplizierter, diese Abwärtskompatibilität zu gewährleisten. **Die Entwickler konzentrierten sich meist nur noch auf das neue System**.

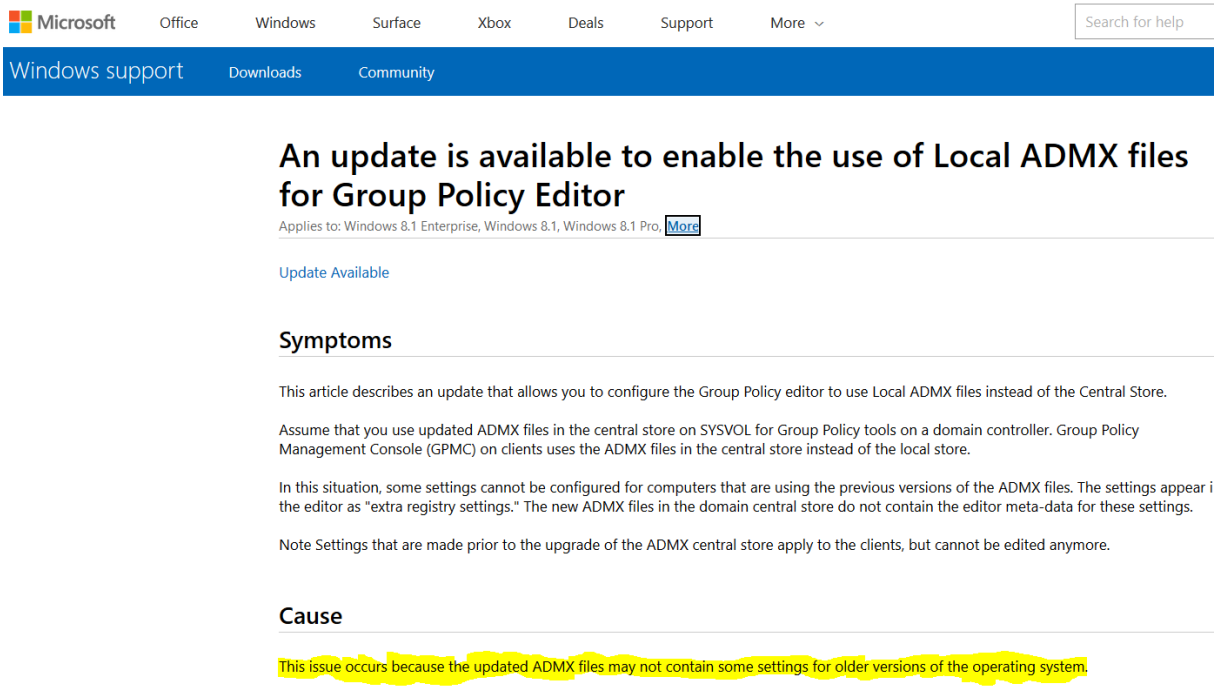
Naja, so oft kommen doch keine neuen Betriebssysteme raus, oder? Falsch: Jedes Jahr veröffentlicht Microsoft zwei neue Releases durch den Semi Annual Channel! Und damit wird das Problem immer größer. Mittlerweile haben wir unter Berücksichtigung des Extended Security Support folgende mögliche Versionen mit Support (**Stand 31.01.2020**):

Betriebssystem	Support von	Support bis	Support-Hinweis
Windows 7	22.10.2009	14.01.2020 +3a	max. 3 Jahre Extended Security
Windows 8.1	18.10.2013	10.01.2023	
Windows 10 1709	17.10.2017	14.04.2020	nur mit Enterprise Edition
Windows 10 1803	30.04.2018	10.11.2020	
Windows 10 1809	13.11.2018	11.05.2021	nur mit Enterprise Edition
Windows 10 1903	21.05.2019	08.12.2020	
Windows 10 1909	12.11.2019	10.05.2022	nur mit Enterprise Edition
Windows 10 LTSB 2015	29.07.2015	14.10.2025	nur mit Enterprise Edition
Windows 10 LTSB 2016	02.08.2016	13.10.2026	nur mit Enterprise Edition
Windows 10 LTSB 2019	13.11.2018	09.01.2029	nur mit Enterprise Edition
Windows Server 2008R2		14.01.2020 +3a	max. 3 Jahre Extended Security
Windows Server 2012	30.10.2012	10.10.2023	
Windows Server 2012R2	25.11.2013	10.10.2023	

Windows Server 2016	15.10.2016	12.01.2027	
Windows Server 2019	13.11.2018	09.01.2029	

In der Liste habe ich mal die Windows Server mit Semi Annual Channel ausgespart.

Die spannende Frage lautete für mich: „Ab wann war die Abwärtskompatibilität der Gruppenrichtlinien nicht mehr garantiert?“. Dazu habe ich einen Artikel bei Microsoft gefunden: **Das Problem mit den Vorlagen ohne komplette Abwärtskompatibilität existiert seit Windows Server 2012 und Windows 8:**



https://support.microsoft.com/en-us/help/2917033/an-update-is-available-to-enable-the-use-of-local-admx-files-for-group

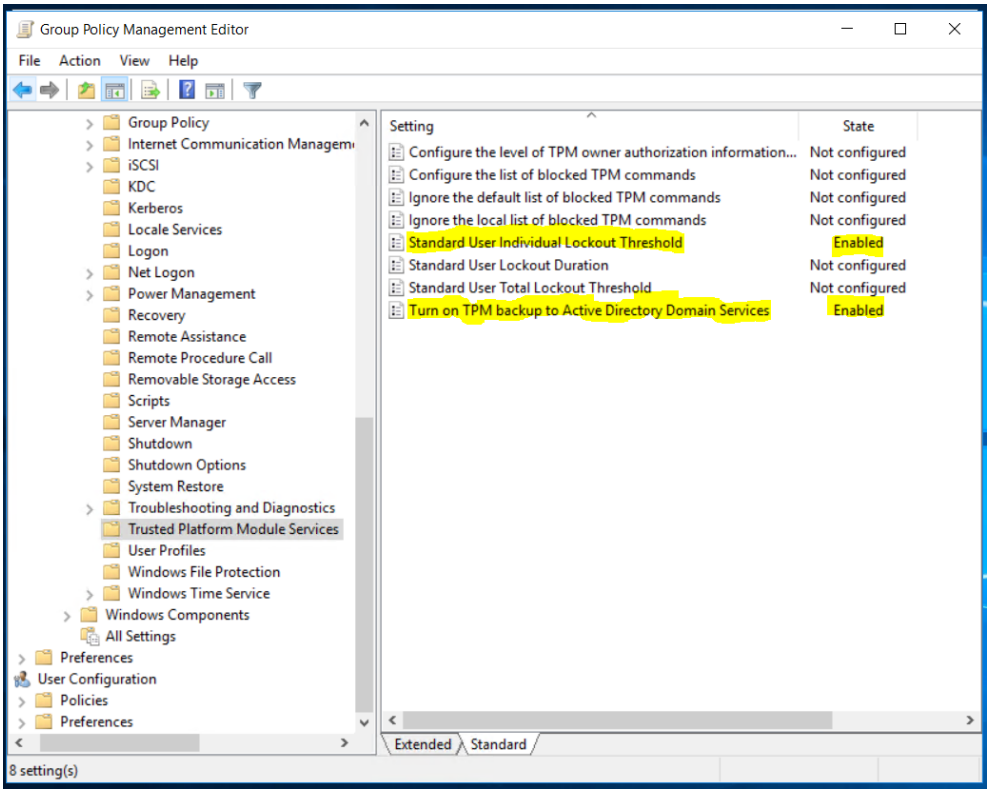
Damals war ein Hotfix für einen Workaround verfügbar, dessen Funktion mittlerweile als Standard in allen Betriebssystemen enthalten ist. Diesen Workaround erläutere ich weiter unten.

OK, dann gibt es einige Einstellungen nicht mehr. Was soll schon passieren, fragt ihr euch? Da gibt es einige Szenarien…

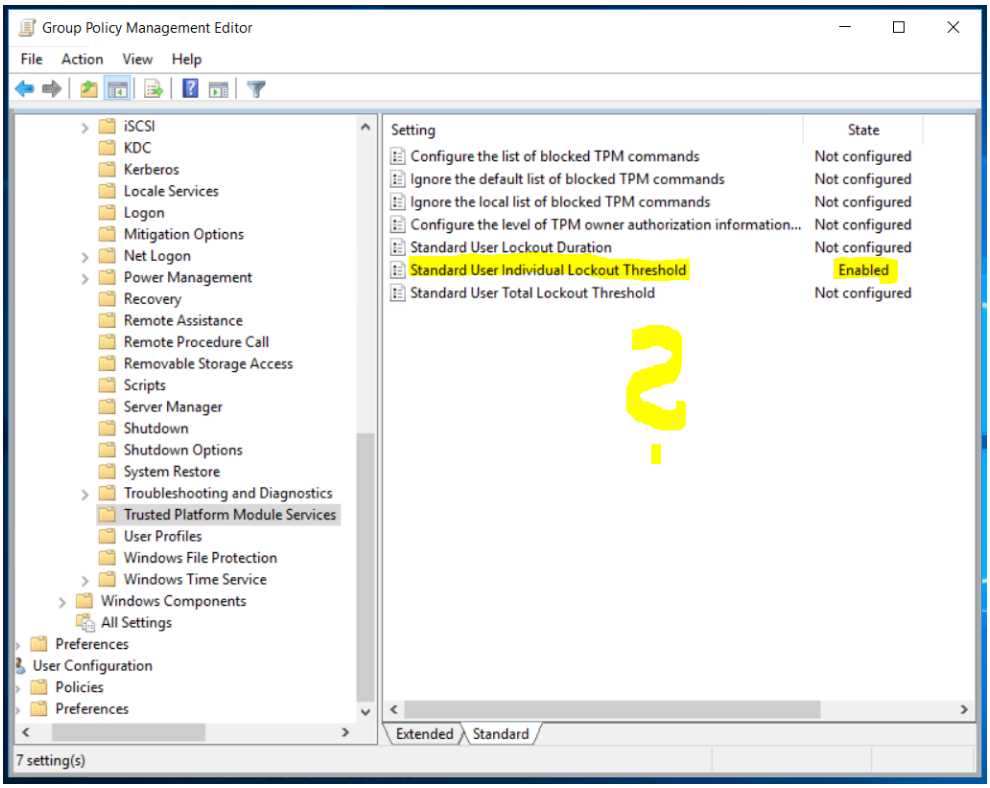
Problemszenario 1: entfernte Konfigurationsoption

Microsoft ersetzt eine ADMX-Datei durch eine neue Version. Wenn ihr die alte Version vorher für eine bestehende GPO verwendet habt, dann kommt ihr mit der neuen Vorlage eventuell nicht mehr an die konfigurierten Einstellungen heran. Man erkennt solche Elemente an den „Extra Registry Settings“ im Report der GPO.

Hier sieht man eine GPO, die ich mit den ADMX-Dateien von Windows Server 2008R2 erstellt habe. Die gleichen Einstellungen erreiche ich auch mit den Vorlagen vom Windows Server 2012R2:

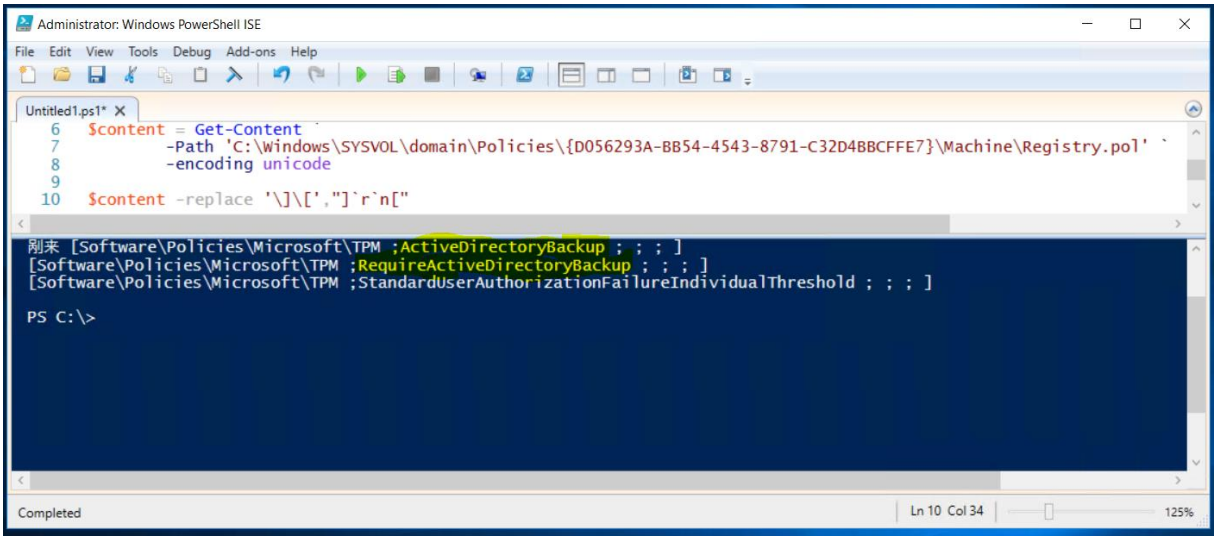


Mit den Vorlagen eines Windows Server 2016 und 2019 kann ich auf eine Einstellungen nicht mehr zugreifen:



Vielleicht wird diese nicht mehr ab Windows Server 2016 benötigt. Aber wenn ich noch „alte“ Windows Server 2012R2 im Einsatz habe, wie kann ich die Einstellung nachträglich anpassen?

Die gesetzte Einstellung in der registry.pol-Datei ist dabei natürlich immer noch vorhanden:



Das bedeutet, kompatible Clients werden die Einstellung immer noch anwenden. Man kommt nur nicht mehr an die Schalter im Frontend heran. Im Vergleich der beiden ADMX-Dateien sieht man schön den Unterschied. Hier ist die Einstellung bis Windows Server 2012R2 vorhanden:

```
<?xml version="1.0" encoding="utf-8"?>
<!--  (c) 2006 Microsoft Corporation  -->
<policyDefinitions xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSch
  <policyNamespaces>
    <target prefix="tpm" namespace="Microsoft.Policies.TrustedPlatformModule" />
    <using prefix="windows" namespace="Microsoft.Policies.Windows" />
  </policyNamespaces>
  <resources minRequiredRevision="1.0" />
  <categories>
    <category name="TPMCategory" displayName="$(string.TPMCategory)">
      <parentCategory ref="windows:System" />
    </category>
  </categories>
  <policies>
    <policy name="ActiveDirectoryBackup_Name" class="Machine" displayName="$(string.ActiveDirectoryBacku
    <policy name="OSManagedAuth_Name" class="Machine" displayName="$(string.OSManagedAuth_Name)" explair
    <policy name="BlockedCommandsList_Name" class="Machine" displayName="$(string.BlockedCommandsList_Na
    <policy name="IgnoreDefaultList_Name" class="Machine" displayName="$(string.IgnoreDefaultList_Name)"
    <policy name="IgnoreLocalList_Name" class="Machine" displayName="$(string.IgnoreLocalList_Name)" exp
      <policy name="StandardUserAuthorizationFailureDuration_Name" class="Machine" displayName="$(string
    <policy name="StandardUserAuthorizationFailureIndividualThreshold_Name" class="Machine" displayName=
    <policy name="StandardUserAuthorizationFailureTotalThreshold_Name" class="Machine" displayName="$(st
  </policies>
</policyDefinitions>
```

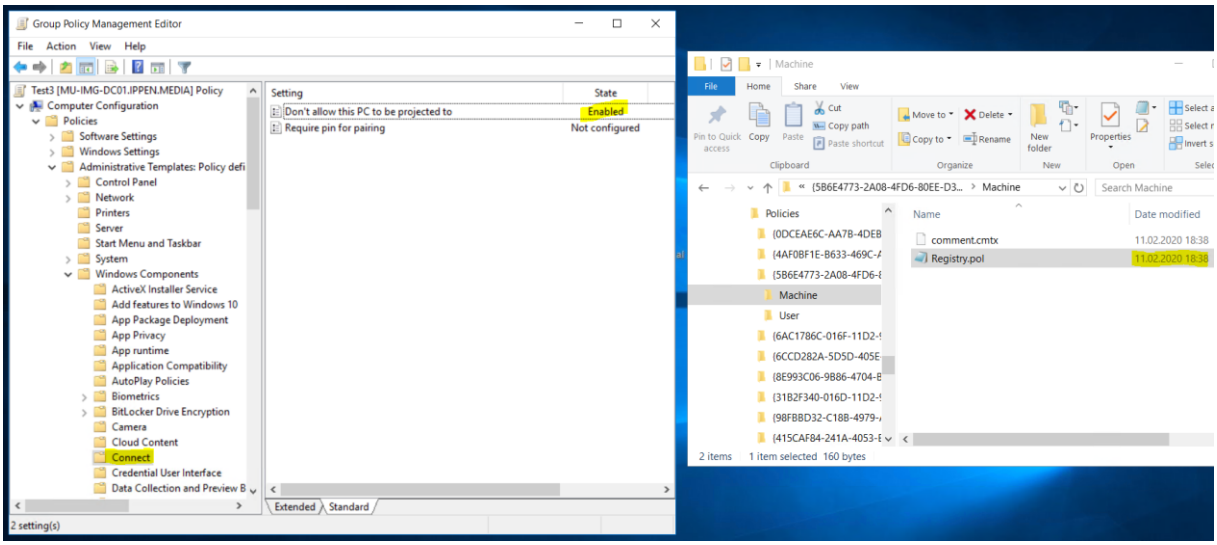
Die neue Vorlage ab Windows Server 2016 sieht dagegen so aus:

```
<?xml version="1.0" encoding="utf-8"?>
<!--  (c) 2006 Microsoft Corporation  -->
<policyDefinitions xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSch
  <policyNamespaces>
    <target prefix="tpm" namespace="Microsoft.Policies.TrustedPlatformModule" />
    <using prefix="windows" namespace="Microsoft.Policies.Windows" />
  </policyNamespaces>
  <resources minRequiredRevision="1.0" />
  <categories>
    <category name="TPMCategory" displayName="$(string.TPMCategory)">
      <parentCategory ref="windows:System" />
    </category>
  </categories>
  <policies>
    <policy name="OSManagedAuth_Name" class="Machine" displayName="$(string.OSManagedAuth_Name)" explair
    <policy name="BlockedCommandsList_Name" class="Machine" displayName="$(string.BlockedCommandsList_Na
    <policy name="IgnoreDefaultList_Name" class="Machine" displayName="$(string.IgnoreDefaultList_Name)"
    <policy name="IgnoreLocalList_Name" class="Machine" displayName="$(string.IgnoreLocalList_Name)" exp
      <policy name="StandardUserAuthorizationFailureDuration_Name" class="Machine" displayName="$(string
    <policy name="StandardUserAuthorizationFailureIndividualThreshold_Name" class="Machine" displayName=
    <policy name="StandardUserAuthorizationFailureTotalThreshold_Name" class="Machine" displayName="$(st
  </policies>
</policyDefinitions>
```

So wird eben nichts mehr angezeigt! Viel Spass beim TroubleShooten!

Problemszenario 2: Veränderung einer Einstellung #1

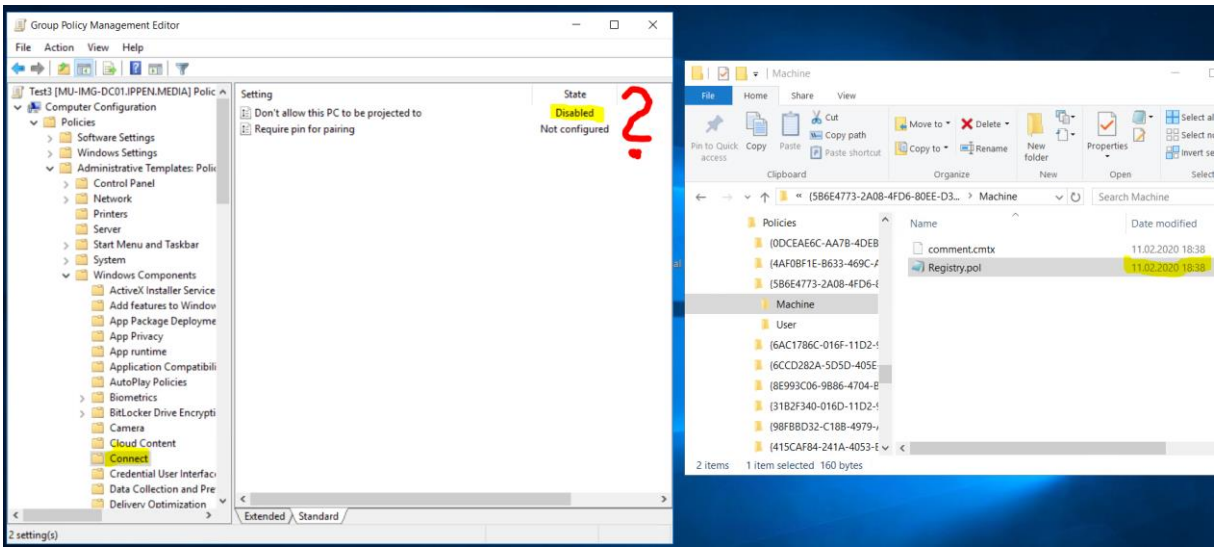
Jedes Editor-Formular wird mit den Definitionen einer ADMX aufgebaut. Setzt man darin eine Einstellung auf z.B. „aktiviert“, dann wird im Hintergrund in einer Datei im Sysvol ein Wert in einer Datei eingetragen. Hier seht ihr ein einfaches Beispiel:



Den Wert bestimmt der Inhalt der dazugehörigen ADMX-Datei. Sie gehört einem Windows Server 2016:

```
<policy name="AllowProjectionToPC" class="Machine" displayName=
"$(string.AllowProjectionToPC)" explainText=
"$(string.AllowProjectionToPC_help)" key=
"Software\Policies\Microsoft\Windows\Connect" valueName="AllowProjectionToPC">
  <parentCategory ref="Connect" />
  <supportedOn ref="windows:SUPPORTED_Windows_10_0_NOSERVER" />
  <enabledValue>
    <decimal value="0" />
  </enabledValue>
  <disabledValue>
    <decimal value="1" />
  </disabledValue>
</policy>
```

Es kommt selten vor: Microsoft ändert auch die hinterlegten Werte in den ADMX! Die gleiche GPO-Einstellung von eben öffne ich mit einem GPO-Editor, der auf die ADMX-Dateien eines Windows Server 2019 zugreift:



Was ist denn hier passiert?? Ganz einfach: in der neuen ADMX steht der Wert für „aktiviert“ mit einem anderen Wert:

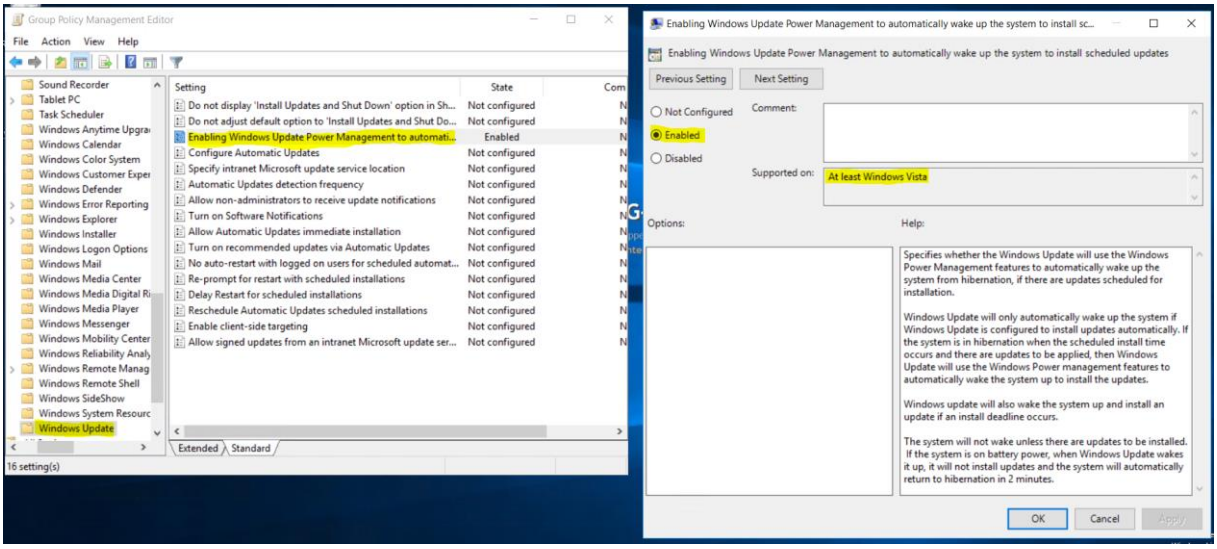
```
<policy name="AllowProjectionToPC" class="Machine" displayName=
"$(string.AllowProjectionToPC)" explainText=
"$(string.AllowProjectionToPC_help)" key=
"Software\Policies\Microsoft\Windows\Connect" valueName="AllowProjectionToPC">
  <parentCategory ref="Connect" />
  <supportedOn ref="windows:SUPPORTED_Windows_10_0_NOSERVER" />
  <enabledValue>
    <decimal value="1" />
  </enabledValue>
  <disabledValue>
    <decimal value="0" />
  </disabledValue>
</policy>
```

Öffnet man nun den Editor mit der neuen ADMX-Datei, dann ließt er die bereits konfigurierte Einstellung und zeigt in der grafischen Oberfläche den korrespondierenden Wert lauf der XML-Information an. Aus einem „aktiviert“ wird ein „deaktiviert“.

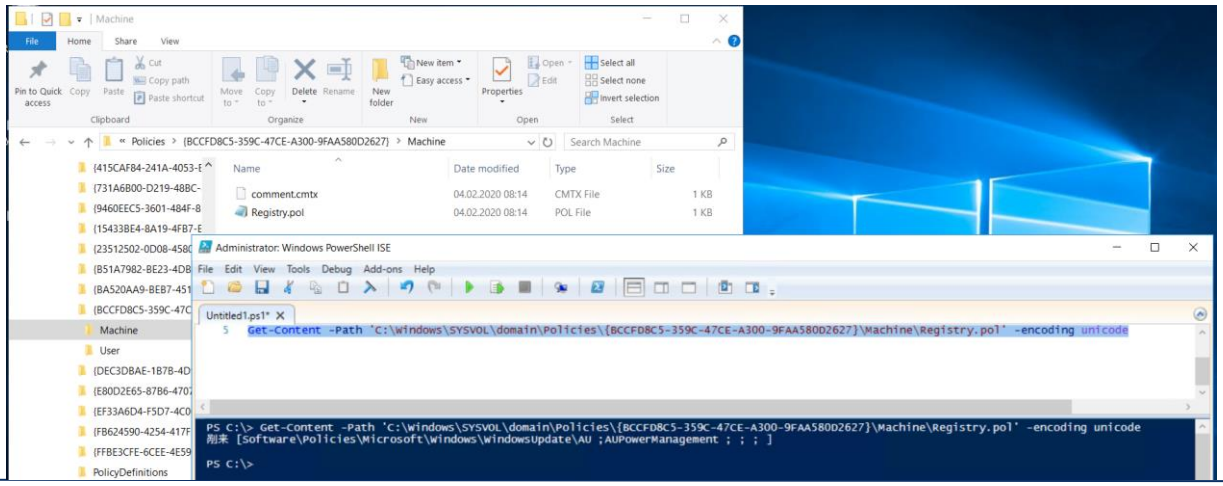
Ein Betriebssystem, dass mit genau dieser ADMX ausgeliefert wurde, wird die Einstellung lokal bestimmt richtig interpretieren. Was ist aber mit den anderen Betriebssystemen? Und was passiert, wenn ihr diese Einstellung auf einem Windows Server 2016 auf „aktiviert“ (also mit dem Wert 0(konfiguriert und durch ein Update auf einmal eure Clients mit Windows 10 1809+ laufen? Die Clients sehen in der registry.pol-Datei im SYSVOL nur eine 0 – und interpretieren sie als „deaktiviert“… Viel Spass beim TroubleShooting!

Problemszenario 3: Veränderung einer Einstellung #2

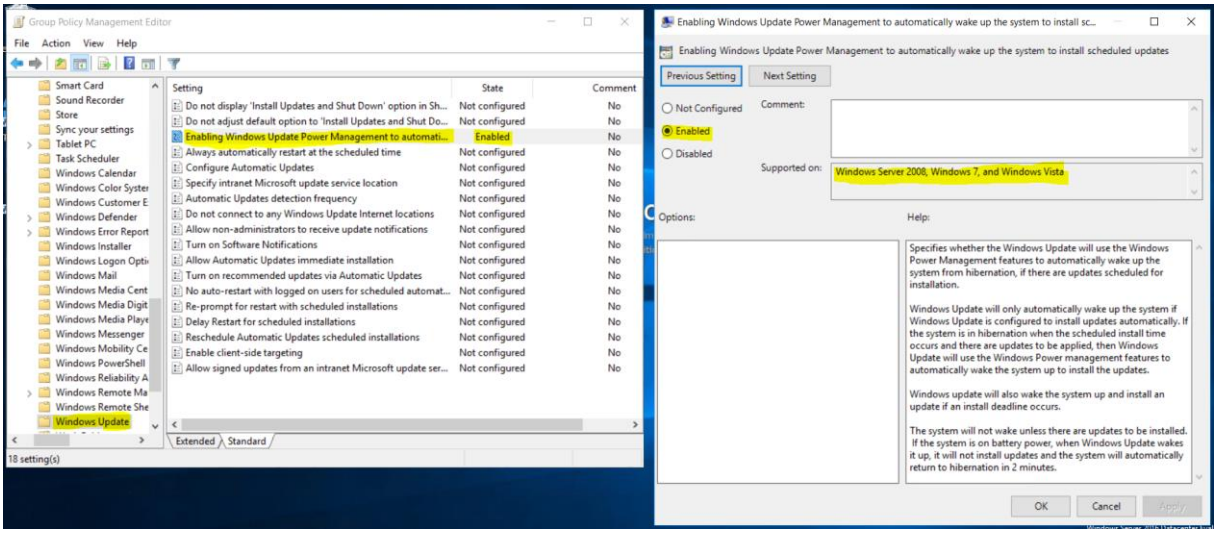
Diese Einstellung habe ich unter „Computer\Policies\Administrative Templates\Windows Components\Windows Update“ gefunden. Die ADMX-Dateien stammen von einem Windows Server 2008R2. Damals stand die Gültigkeit ab Windows Vista beschrieben:



Im Hintergrund wird die Einstellung in einer registry.pol-Datei abgelegt. Diese kann ich mit der PowerShell anzeigen:

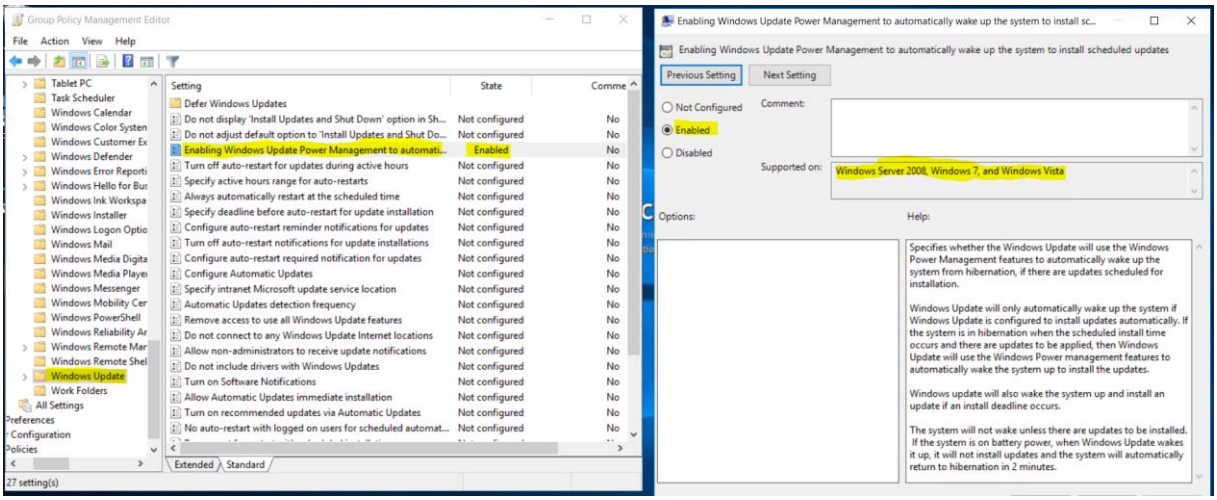


Wenn ich nun die ADMX-Dateien auf Windows Server 2012R2 aktualisiere und die gleiche GPO öffne, dann wird mir folgendes Bild im Editor angezeigt:



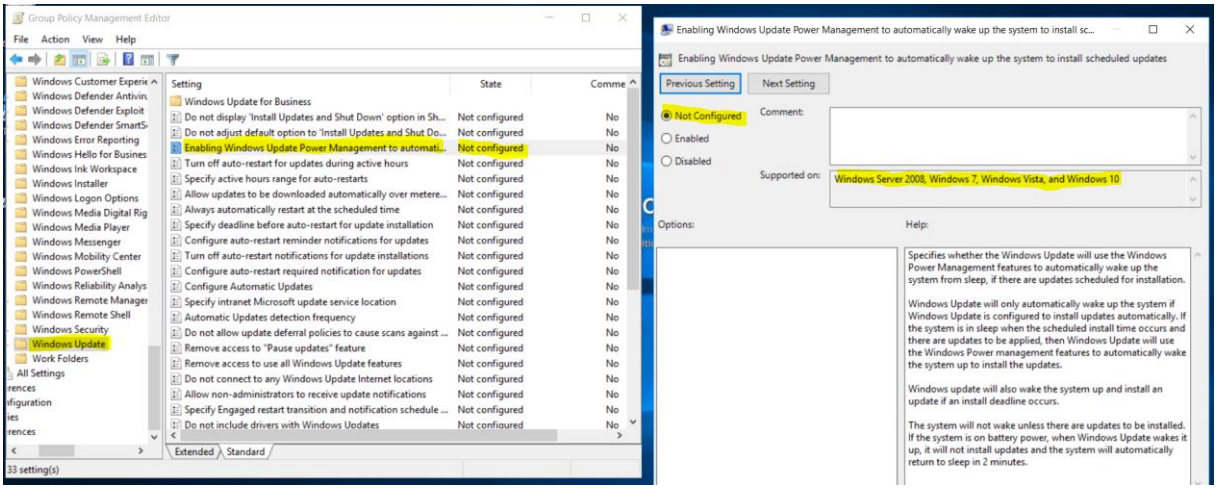
Die Einstellung ist natürlich weiter aktiv, da die zugehörige registry.pol-Datei im SYSVOL zwischenzeitlich nicht verändert wurde. Aber offensichtlich gilt die Regel nicht länger für die derzeit modernen Betriebssysteme Windows 8 und Windows 8.1.

Wie schaut es aus, wenn ich die ADMX-Dateien auf Windows Server 2016 aktualisiere?



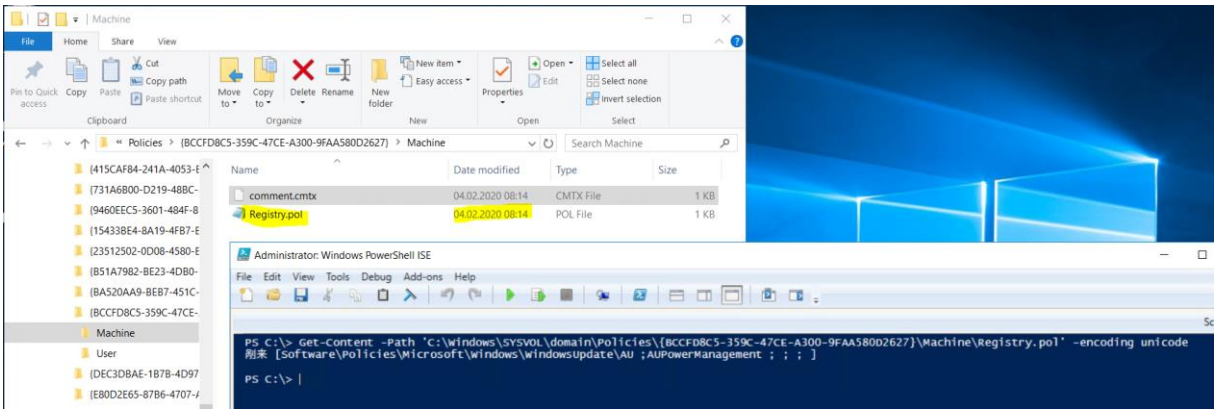
Das sieht nach dem Idealfall aus. So war das Prinzip „SuperSet“ gedacht. Die noch neuere Version hält die Legacy-Einstellung weiter vor und deren Wirkungsbereich wurde nicht verändert.

So wird es mit den ADMX-Dateien unter Windows Server 2019 weitergehen, oder?

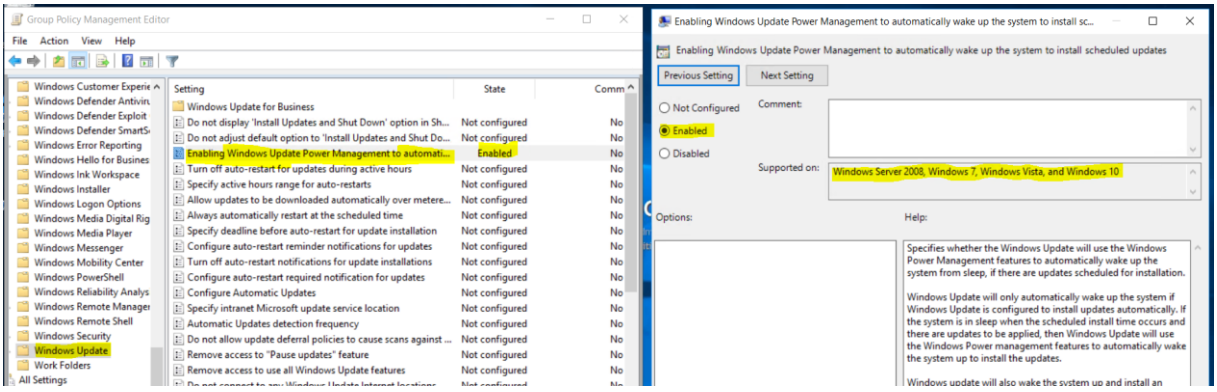


Nanu? Die Einstellung ist auf einmal nicht mehr konfiguriert?? Und plötzlich kann auch Windows 10 wieder damit umgehen? Aber Windows 10 gab es doch schon mit Windows Server 2016? Sind vielleicht nicht alle Versionen gemeint? Was ist da los?

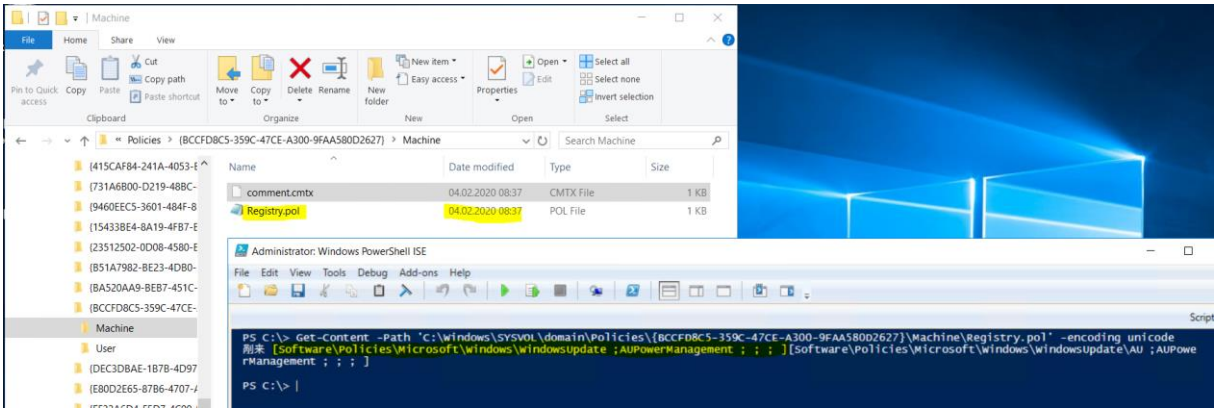
Zur Info: Die Einstellung ist im Hintergrund immer noch vorhanden und alle AMDX-Versionen der Windows Updates bis Windows Server 2016 erkennen die Einstellung auch:



Das bedeutet, die neue Windows Server 2019 ADMX erkennt beim Laden der GPO den Registry-Pfad in der Registry.pol nicht. Aber was passiert dann, wenn ich den Schalter wieder aktiviere?



Das hat funktioniert. Und was steht nun in der Registry.pol-Datei?



Seht ihr den fast identischen, doppelten Eintrag in der registry.pol-Datei? Das darf nicht wahr sein: Microsoft hat in der ADMX-Datei einfach den Pfad des Registry-Keys geändert. Daher wurde die alte Einstellung auch nicht mehr erkannt! Hier sieht man die Konfiguration der ADMX-Datei von Windows Server 2016:

```
    </policy>
    <policy name="AUPowerManagement_Title" class="Machine" displayName=
    "$(string.AUPowerManagement_Title)" explainText=
    "$(string.AUPowerManagement_Help)" key=
    "Software\Policies\Microsoft\Windows\WindowsUpdate\AU" valueName=
    "AUPowerManagement">
      <parentCategory ref="WindowsUpdateCat" />
      <supportedOn ref="windows:SUPPORTED_Windows7ToVista" />
      <enabledValue>
        <decimal value="1" />
      </enabledValue>
      <disabledValue>
        <decimal value="0" />
      </disabledValue>
    </policy>
```

Und hier die gleiche Einstellung in der ADMX-Datei eines Windows Server 2019:

```
    </policy>
    <policy name="AUPowerManagement_Title"
            class="Machine"
            displayName="$(string.AUPowerManagement_Title)"
            explainText="$(string.AUPowerManagement_Help)"
            key="Software\Policies\Microsoft\Windows\WindowsUpdate"
            valueName="AUPowerManagement">
      <parentCategory ref="WindowsUpdateCat" />
      <supportedOn ref="windows:SUPPORTED_Windows7ToVistaAndWindows10" />
      <enabledValue>
        <decimal value="1" />
      </enabledValue>
      <disabledValue>
        <decimal value="0" />
      </disabledValue>
    </policy>
```

Nur wissen denn das jetzt auch die alten Betriebssysteme? Die müssen ja zur richtigen Zeit an der richtigen Stelle in der Registry nach der Konfiguration suchen. Wenn die dazugehörige GPO aber den falschen Pfad abspeichert, weil die entsprechende ADMX-Datei nicht für das Zielbetriebssystem gebaut wurde, dann wird das nichts! Viel Spass beim TroubleShooting!

Problemszenario 4: Clientseitige Veränderungen

Ebenso kann die Interpretation von Einstellungen im Client problematisch werden. Ich habe einen Bekannten, der seine Clients über einen WSUS aktualisiert. Die Konfiguration dazu lief immer problemlos über eine GPO: Die Clients besuchten den WSUS-Server und installierten brav die dort genehmigten Updates. Die Featureupdates wurden aber für interne Tests zurückgehalten. Ein Standard in Firmenumgebungen.

Eines Tages installierten alle Clients fast gleichzeitig ihre Betriebsversion auf das neuste Release von Windows 10! Dabei war dieses Update nicht freigegeben! Was war passiert? Die Clients wurden durch ein vorheriges Featureupdate bewusst aktualisiert. Die neue Version war aber nicht mit der alten GPO kompatibel und die Systeme ignorierten die Einstellung für Windows Updates komplett. Danach arbeiteten sie im Standardmodus und gingen einfach zum Windows Update Service

im Internet. Mit dessen Updates hielten sie sich aktuell. Bis das nächste Featureupdate anstand – das intern nicht genehmigt war. Und die nicht genehmigte Installation startete.

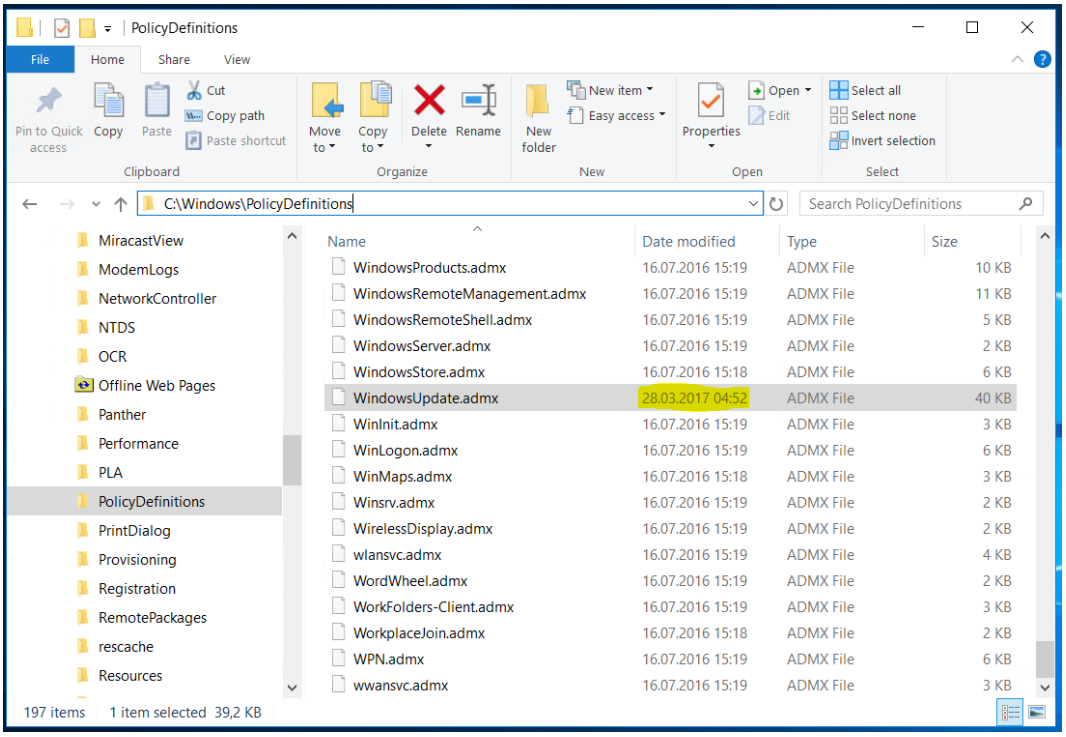
Und meine eigene Infrastruktur hatte das Problem auch schon, dass eine funktionierende GPO auf einmal nicht mehr wirkte: https://www.ws-its.de/wshowto-wsus-und-clients-melden-100-aber-einige-updates-fehlen/

Problemszenario 5: neue ADMX-Dateien durch Windows Updates

Administratoren legen gerne die passenden ADMX-Dateien im Central Store im Active Directory SYSVOL-Share ab. Die Dateien bekommt man aus dem Verzeichnis C:\Windows\PolicyDefinitions des Zielbetriebssystems.

Nach dem Prinzip „SuperSet“ sollten hier die ADMX-Dateien des neusten Betriebssystems abgelebt werden. Man installiert also ein Referenz-System und hat Zugriff auf die Dateien. Bis zum nächsten FeatureUpdate oder bis zur nächsten Version des Betriebssystems gibt es keine Veränderungen mehr.

Leider stimmt auch diese Annahme nicht (mehr): Microsoft passt auch bereits veröffentlichte Betriebssysteme an. Hier sieht man schön die unterschiedlichen Datumswerte auf einem Windows Server 2016:



Dies geschieht üblicherweise durch die monatlichen Updates. Da sind eben nicht nur Sicherheitspatches enthalten. Wird eine lokale Komponente des Betriebssystems angepasst, dann ziehen die lokalen ADMX-Dateien mit. Aber wer denkt schon jeden Monat daran, diese auch in den Central Store zu kopieren?

Die Folge kann man sich wieder gut vorstellen: Man erstellt mit den zentralen, veralteten Vorlagen die Richtlinien für aktualisierte Systeme. Und irgendwie funktioniert da etwas wieder nicht… Viel Spass beim TroubleShooting!

Die administrative Lösung bzw. der WorkAround

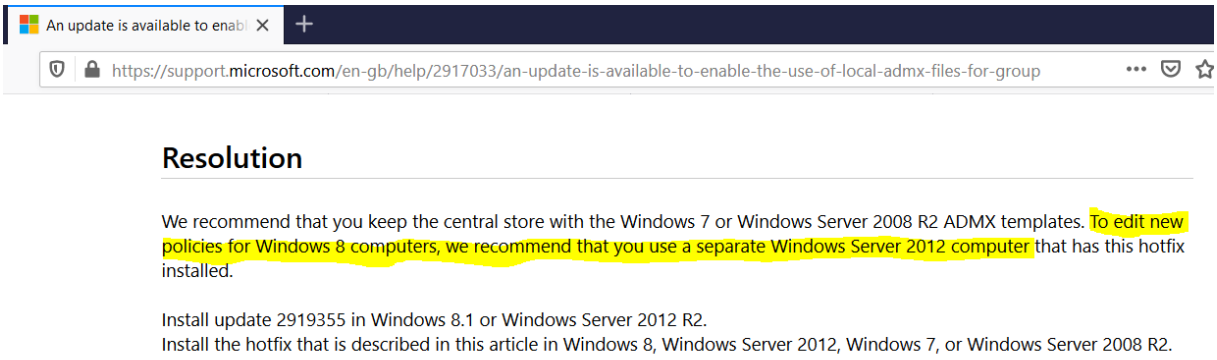
Die eingefahrene Vorgehensweise der Ablage aller ADMX-Vorlagen im Central Store stellt uns vor ein Problem: In diesem Verzeichnis kann ein Dateiname nur einmal verwendet werden. Und Microsoft versioniert die Vorlagen im Inhalt der ADMX-Dateien und leider nicht im Dateinamen. So kann also immer nur eine Betriebssystemversion zu einer Zeit für die Gruppenrichtlinien-Editierung verwendet werden!

Ich habe immer wieder Firmen gesehen, die unter dem Central Store „Versionsordner“ vorhalten. Bevor eine GPO editiert wird, ersetzt man einfach die Vorlagen durch den Inhalt des Versionsordners. Das ist keine praktikable Lösung:

- Zum einen kann man so immer nur ein Betriebssystem zu einer Zeit editieren.

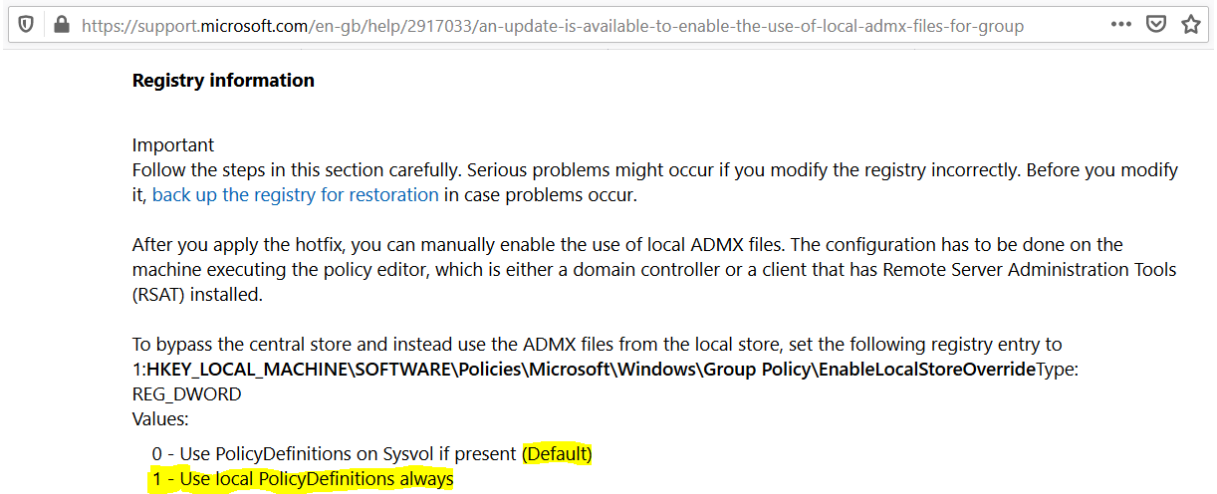
- Dazu werden Veränderungen im SYSVOL immer auf alle Domain Controller repliziert. Da gibt es dann auch Replikationsverzögerungen. Denkt man da immer dran?
- In den GPO-Editoren sehe ich nicht, welche Vorlagenversion gerade geladen ist. Administration ist da eher Glückssache.

Microsoft hat das Problem selber erkannt und im gleichen Dokument wie oben gezeigt einen Workaround vorgeschlagen:



Ab Windows 8 und Windows Server 2012 sollen also die Gruppenrichtlinien von einem passenden Betriebssystem editiert werden!

Aber mit dem Central Store ist es doch egal, auf welchem Rechner ich die GPO bearbeite, da die Editoren immer die zentral bereitgestellten Vorlagen verwenden… Hier kommt der Hotfix zum Einsatz. Dieser erweitert die Steuerung, aus welcher Quelle der Editor die ADMX-Dateien bezieht. **Mit dem Hotfix kann ein Registry-Key erstellt werden, der den Client immer seine eigenen Vorlagen verwenden lässt!**



Der Hotfix war für die Übergangszeit erhältlich. Mittlerweile ist er ein Bestandteil der Betriebssysteme. Daher wurde diese Seite nicht mehr aktualisiert. Die Empfehlung, Clientrichtlinien von einem Windows Server aus zu editieren, lässt sich daher nicht auf Windows 10 übertragen. Für dessen GPO ist ein Windows Server 2016 oder 2019 nur bedingt geeignet, da seit Windows 10 öfter neue Versionen erscheinen als auf der Serverseite (sehen wir mal von den SAC der Serverbetriebssysteme ab, die keine grafische Oberfläche haben und daher nicht für die Editierung von Gruppenrichtlinien geeignet sind). Die Server haben daher nicht die passenden ADMX-Dateien dabei!

Daher gelten seit Windows 10 und Windows Server 2016 folgende Regeln für die Bearbeitung von Gruppenrichtlinien:

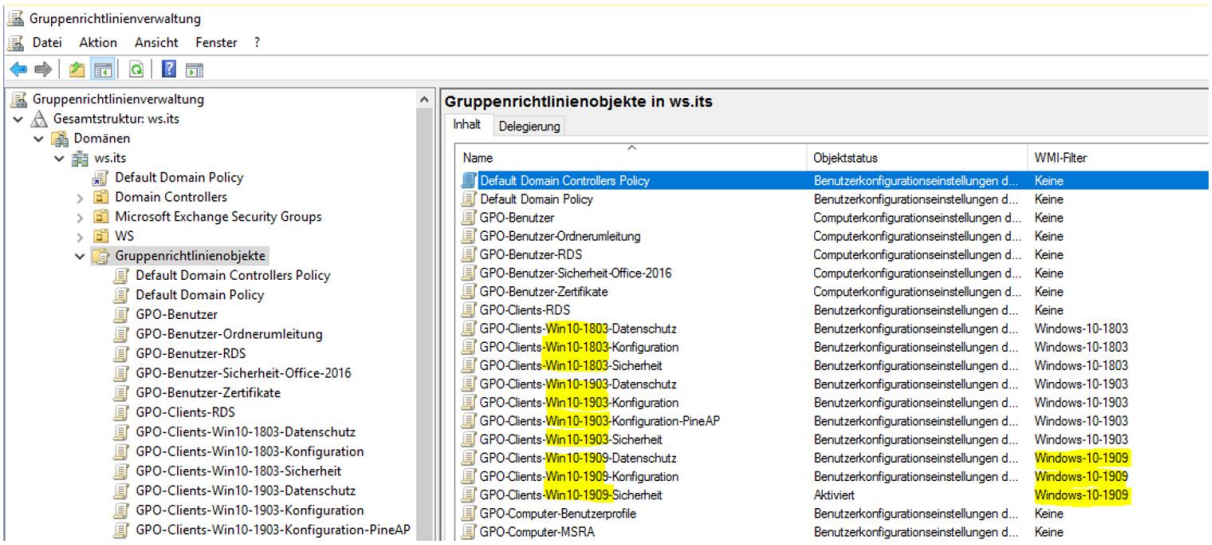
- Für jedes Betriebssystem muss ein eigener Editor-Rechner mit dem Remote Server Administration Tool (RSAT) für die Gruppenrichtlinienverwaltung und dem RegistryKey bereitgestellt werden. Nur von diesem System dürfen die GPO dieser Betriebssystemversion editiert werden.

- Für jede Version der Clients und Server werden separate GPO benötigt! Nur dann passt die Bearbeitung mit den richtigen ADMX-Dateien und die Verarbeitung der fertigen Richtlinien auf dem dazu kompatiblen Zielsystem zusammen!
- Für Windows 10 bietet sich der Einsatz von WMI-Filtern an, da durch die Feature-Updates entsprechende Versionswechsel zur Normalität werden.

Falls euch mit den weiter oben genannten Problemen eine andere Lösung einfällt: Ich bin neugierig!

Mit diesem Wissen kann man eigentlich auch auf den Einsatz des Central Stores verzichten. Das erspart das Setzen der Registry-Keys auf den Editor-Systemen.

Insgesamt wird unsere GPO-Landschaft bald so aussehen müssen:



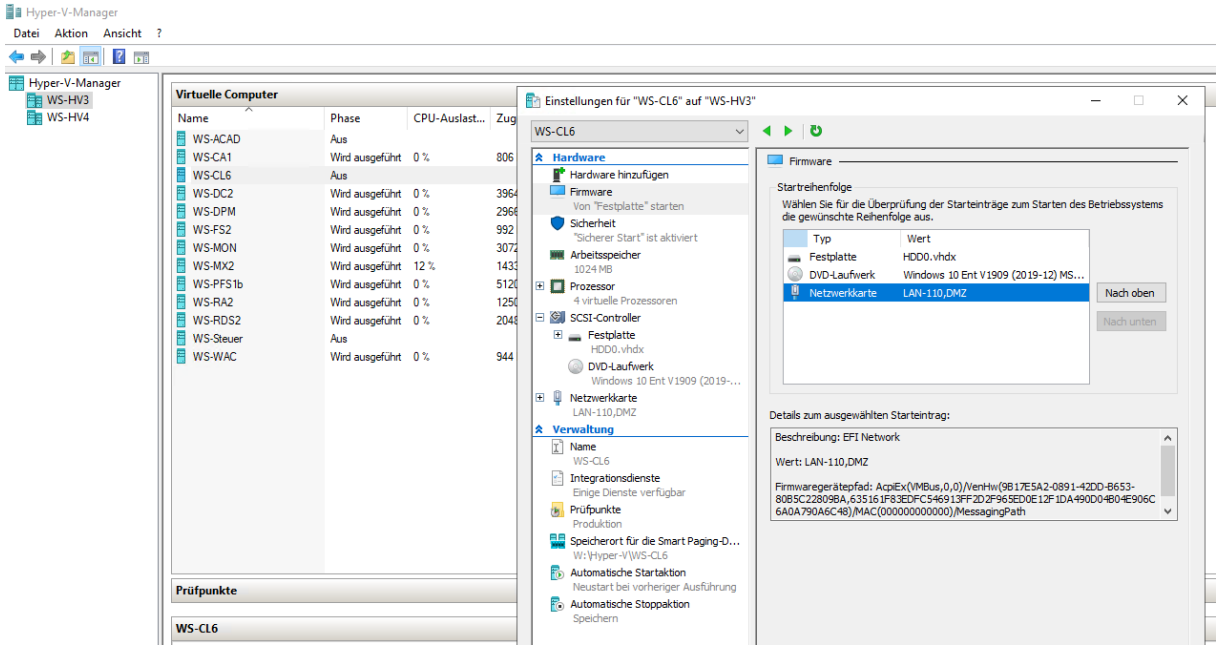
Der Aufwand scheint enorm: jedes Jahr müssen GPO auf‘s Neue erstellt werden! Das klingt dramatisch, ist es aber nicht. Ich habe mir ein System erarbeitet, mit dem die Erweiterung auf ein neues System schnell, unkompliziert und vor allem einfach beim TroubleShooting ist. Denn meine GPO passen einfach immer auf das Zielsystem!

Bereitstellung von GPO für ein neues Betriebssystem

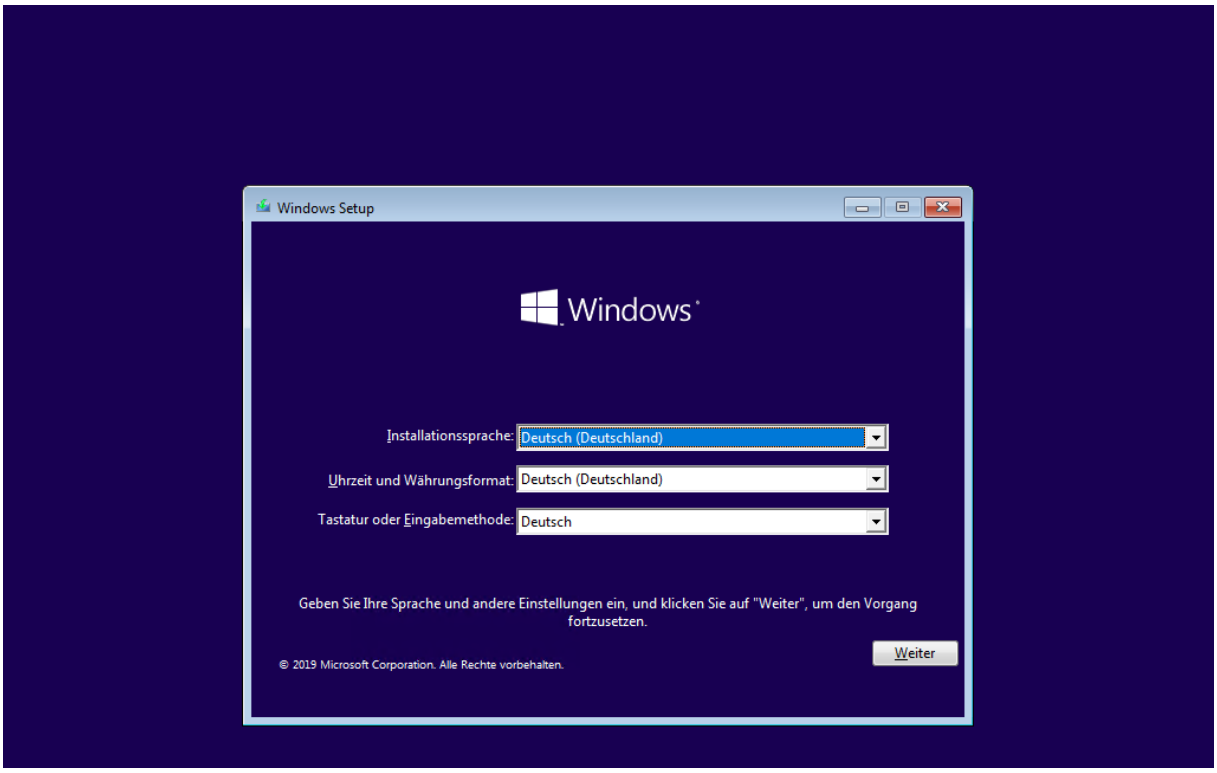
Referenzsystem erstellen

Zuerst benötige ich ein Referenzsystem mit dem gewünschten Betriebssystem. Dieses bringt die erforderlichen ADMX-Dateien mit. In diesem Beispiel möchte ich Windows 10 in der Version 1909 in meiner Infrastruktur bereitstellen. Aktuell habe ich Gruppenrichtlinien bis zur Version 1903 im Einsatz.

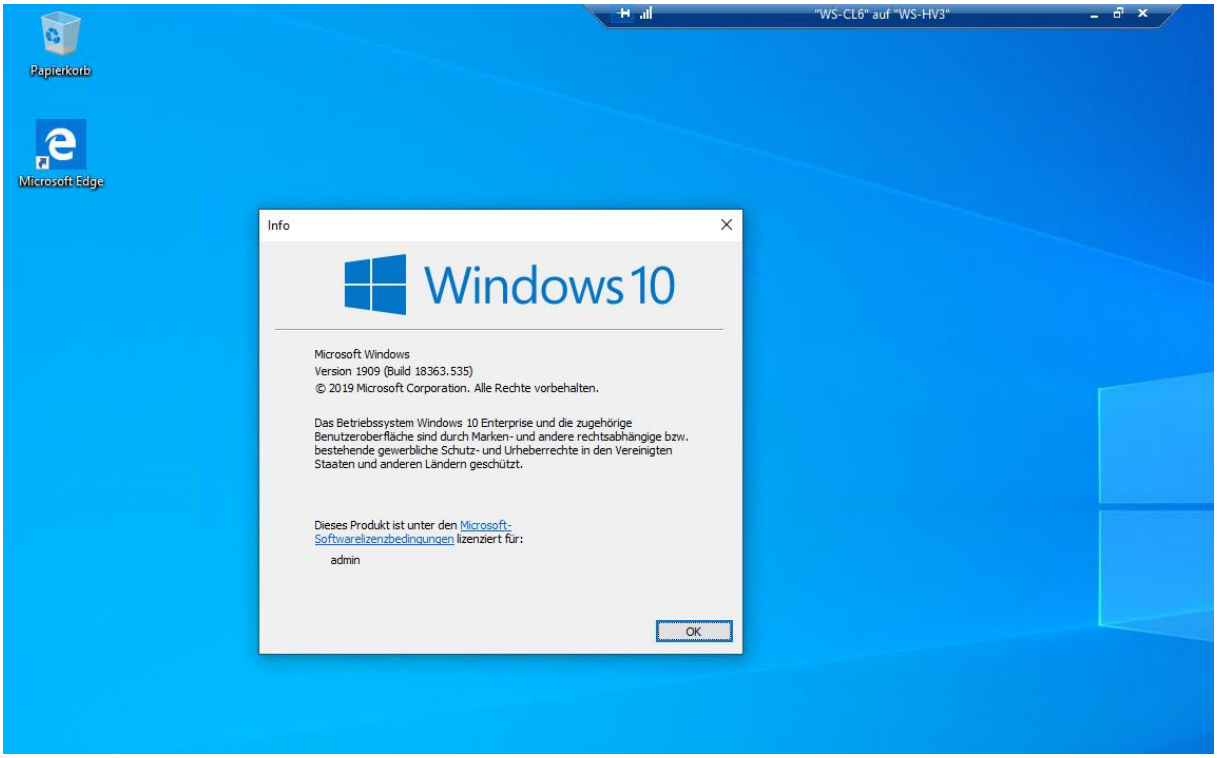
Das neue System wird mein GPO-Editorsystem. Ich benötige also hardwareunabhängigen Zugriff. Daher installiere ich den neuen Client in einer virtuellen Maschine auf einem meiner Hyper-V-Hosts. Diese VM erstelle ich mit dem Assistenten:



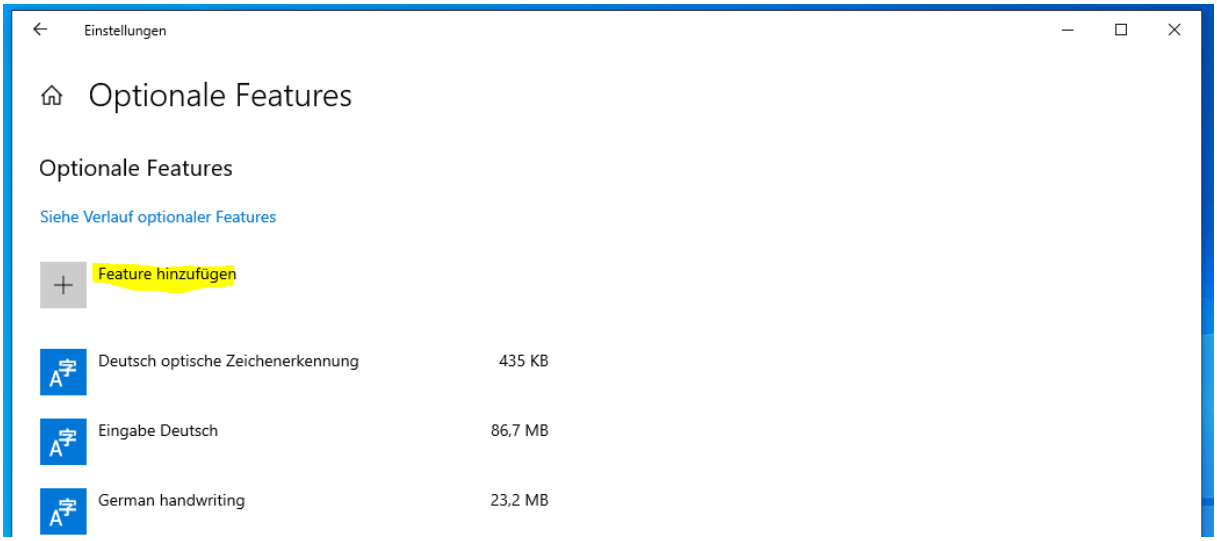
Danach starte ich die VM und installiere das System:

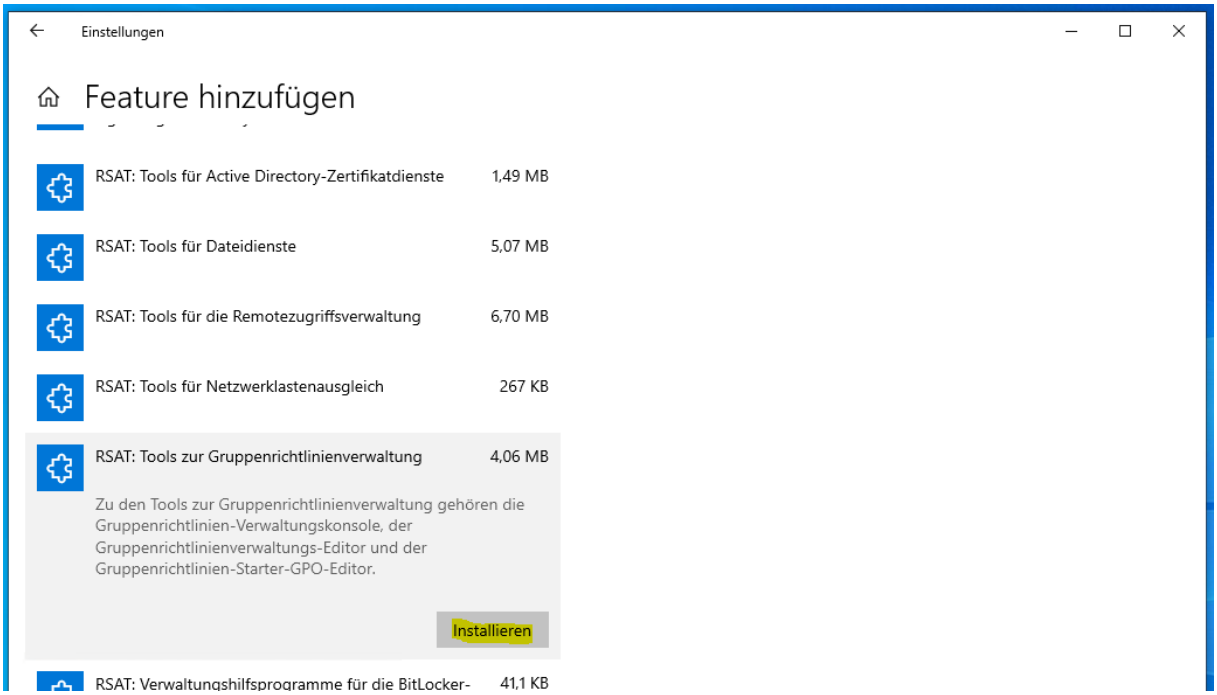


Der Prozess ist seit Jahren nahezu unverändert. Kurze Zeit später bin ich auf dem Client angemeldet:

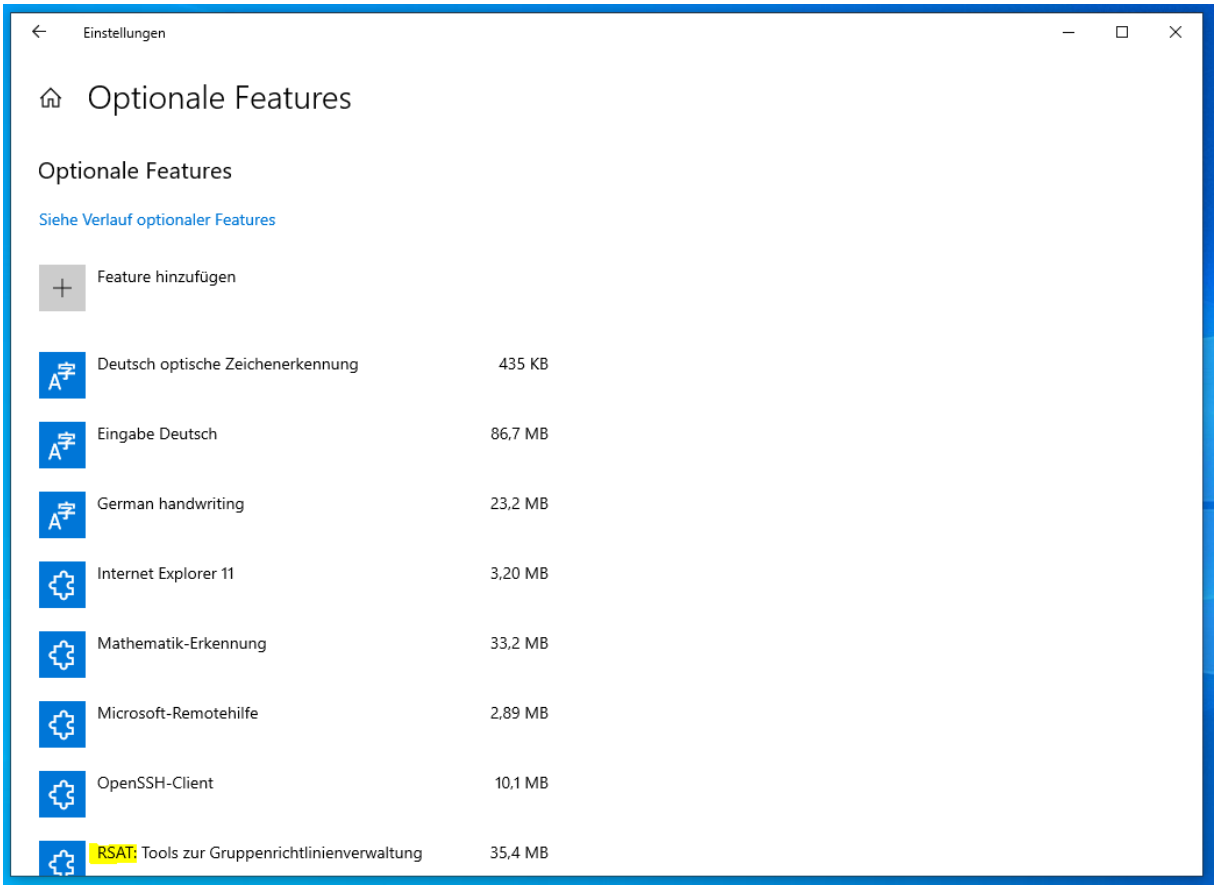


Damit ich von diesem Rechner die Richtlinien meiner Domain editieren kann, benötige ich die Remote Server Administrative Tools (RSAT) für die Gruppenrichtlinienverwaltung. Diese werden bei Servern über den Server Manager installiert. Bei Clients sind sie seit einigen Versionen in den Optionalen Features zu suchen:

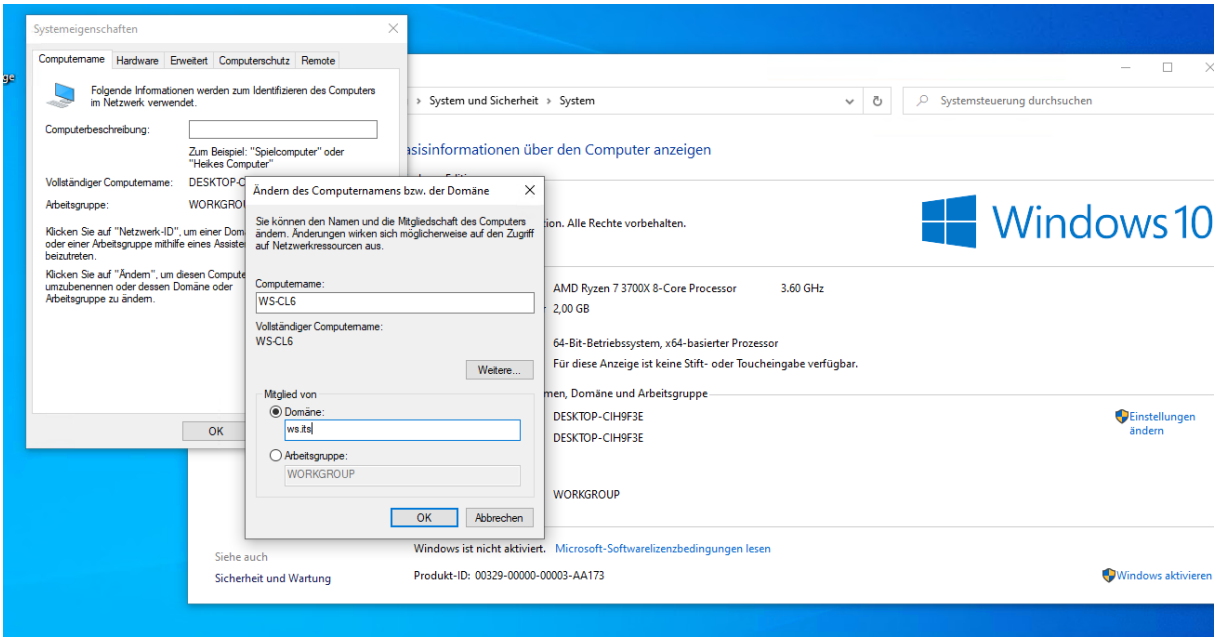




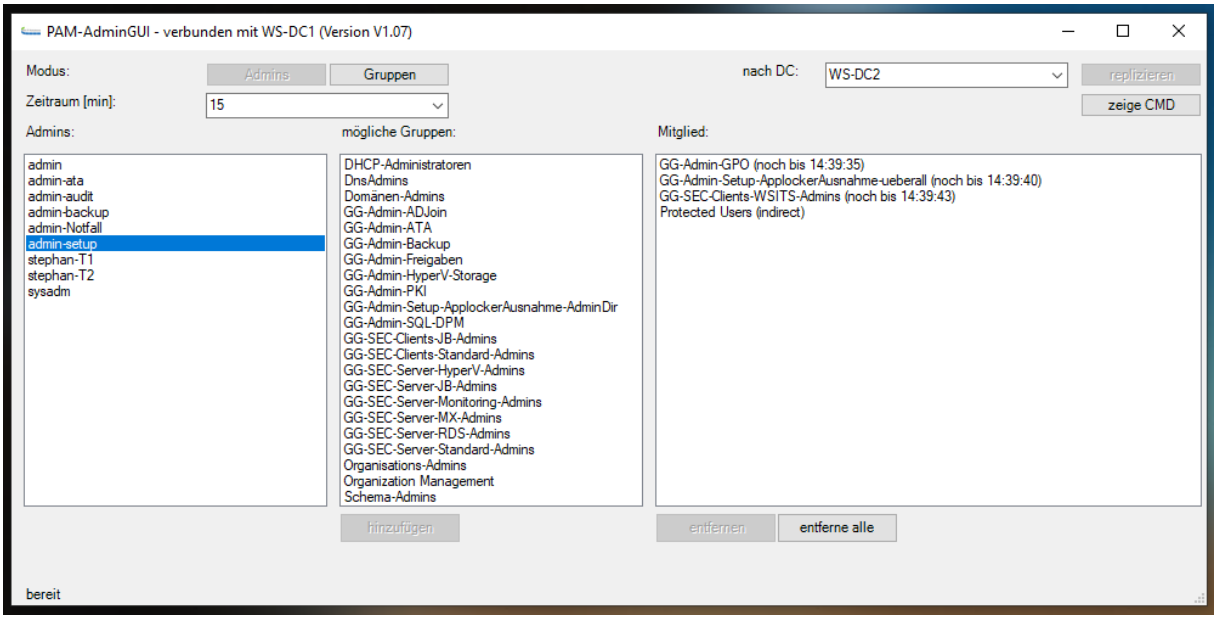
Die Installation benötigt administrative Rechte und eine Internetverbindung. Sie dauert nur wenige Sekunden:



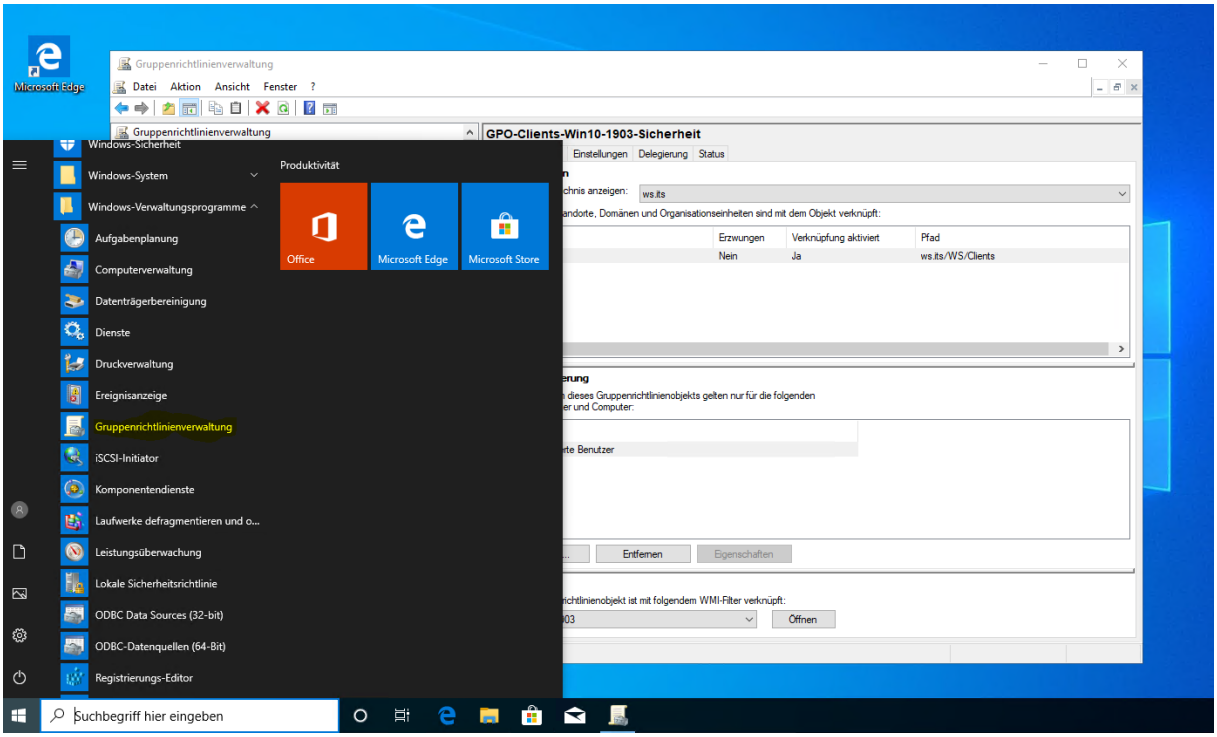
Für die Richtlinien-Bearbeitung ist ein Domain Join erforderlich. Ich nehme den Client in meine Domain auf und benenne ihn dabei um. Das AD-Computerobjekt platziere ich in der richtigen Organisationseinheit:



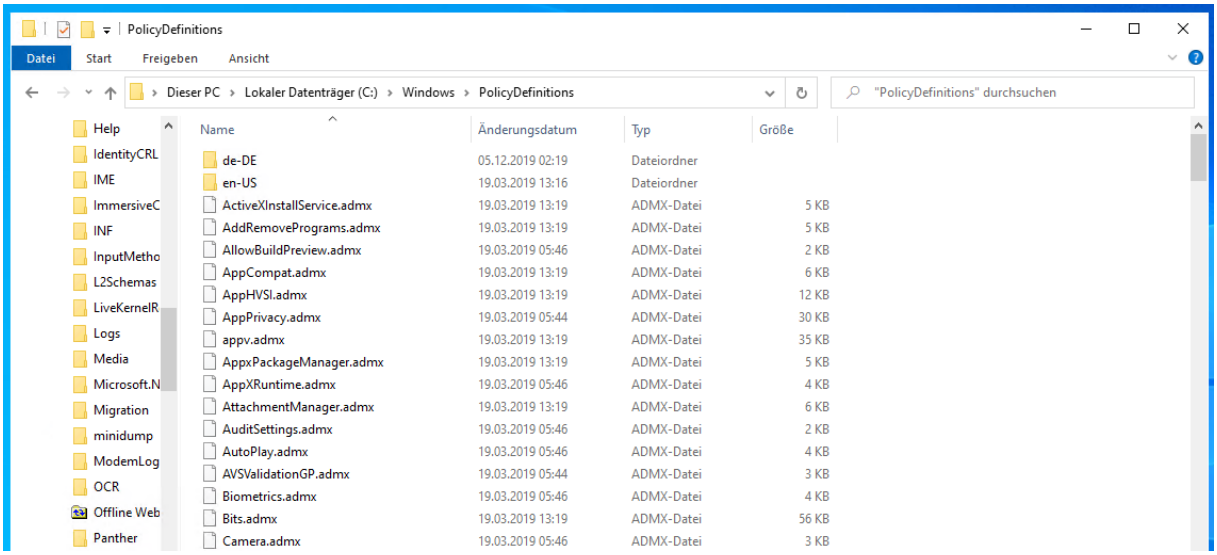
Durch mein Tier-Management und mein Privileged Access Management haben meine Admin-Accounts keine Rechte. Ich muss die erforderlichen Berechtigungen durch eine temporäre Gruppenmitgliedschaft zuweisen. Mein Account bekommt die Rechte zur Bearbeitung aller GPO (Das Recht habe ich an die Gruppe GG-Admin-GPO delegiert). Zusätzlich bekommt der Administrationsrechte auf der OU, in welcher der neue Client platziert wird. Darin sind auch die Logon-Rechte enthalten (diese würden ausreichen). Die Gruppenmitgliedschaften editiere ich mit meinem PowerShell-GUI-Script „PAM-Admin“:



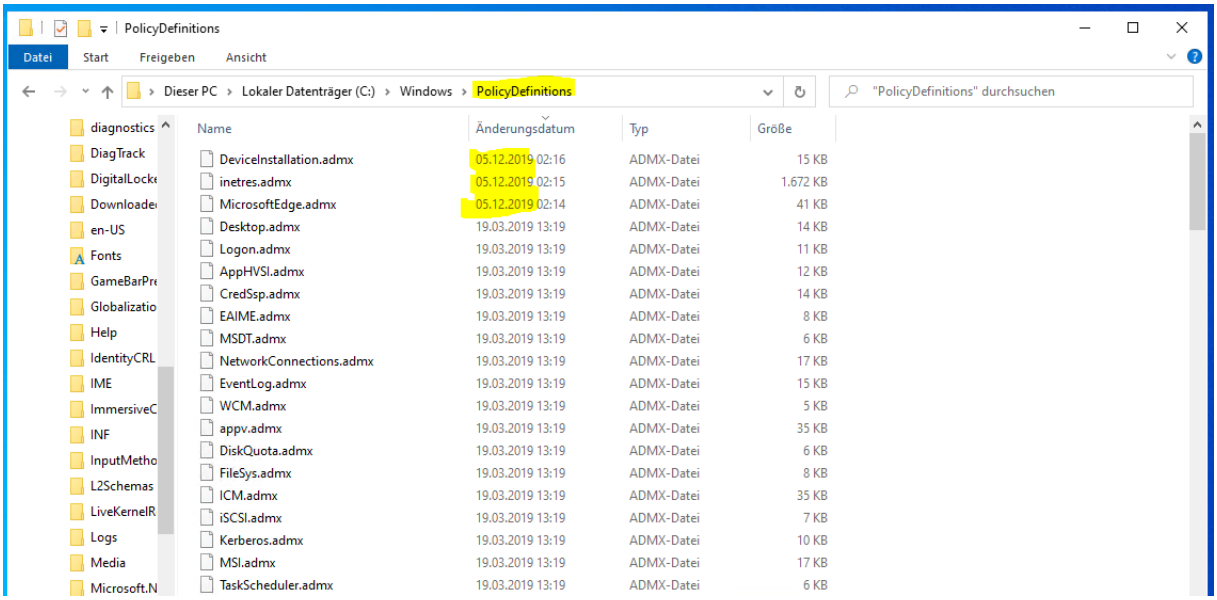
Mit diesen Rechten melde ich mich als Admin am Client an. Weiter geht es in der Gruppenrichtlinien-Verwaltungskonsole:



Die lokalen ADMX-Vorlagendateien des Clients wurden (wenn überhaupt) für Windows 10 Version 1909 getestet:

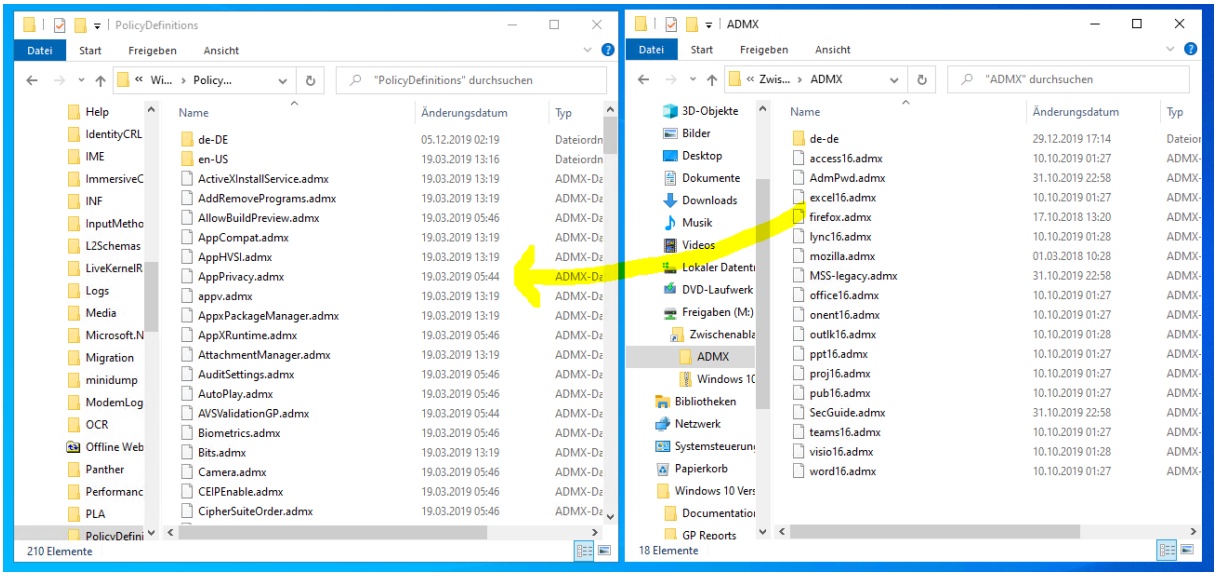


Je mehr sich diese Dateien im Vergleich zum Vorgänger verändert haben, je mehr Arbeit ist zu erwarten. Das Änderungsdatum finde ich interessant. Es entspricht dem des Windows 10 Version 1903. Gibt es überhaupt Veränderungen?



Das ist ein guter Tag! Die Vorlagen haben sich zwischen Windows 10 Version 1903 und 1909 fast nicht verändert! Dennoch möchte ich hier meinen üblichen Zyklus darstellen. Bei einem Wechsel von 1809 auf 1909 wäre das definitiv erforderlich!

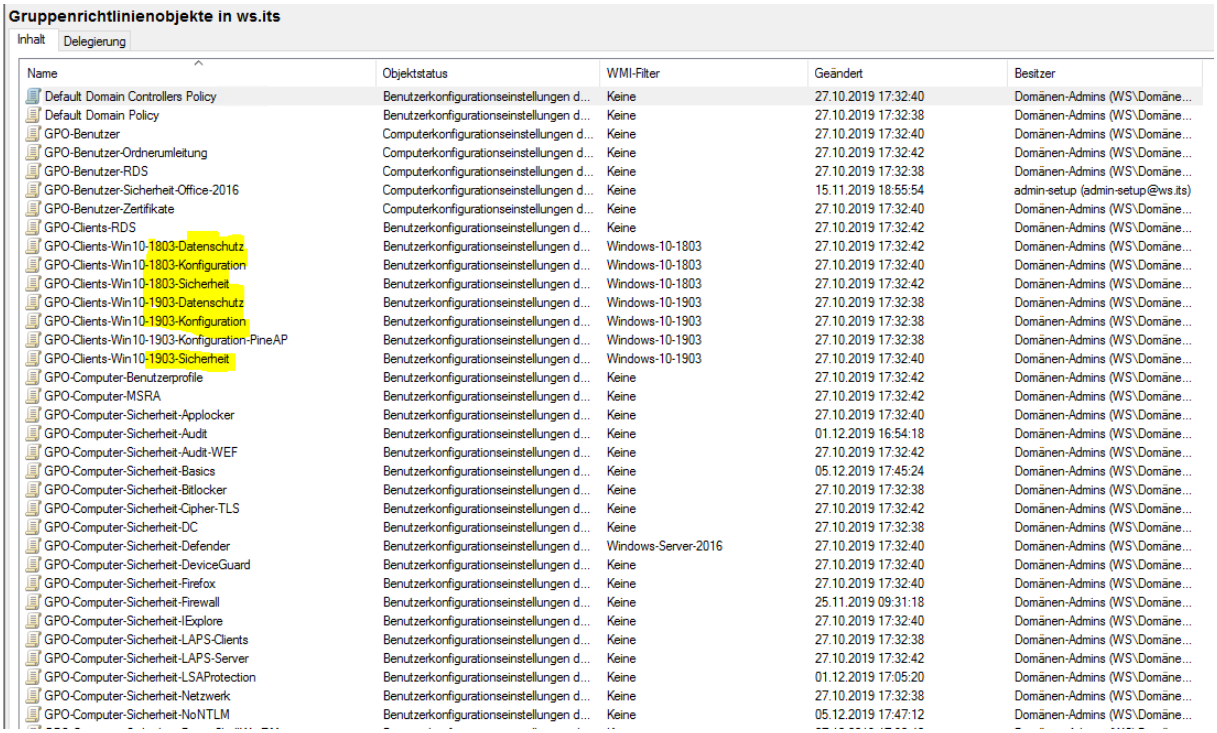
Ich verwende auf ADMX-Dateien für andere Anwendungen. Diese gehören nicht zum Betriebssystem-Standard. Damit ich von dem neuen Editor-Rechner die dazugehörigen Einstellungen verwalten kann, importiere ich die erforderlichen Dateien in das PolicyDefinitions-Verzeichnis. Hier kommen die Office-Templates, die SCT-SecurityTemplates und etwas Mozilla:



Der Editor-PC ist einsatzbereit.

GPO bereitstellen

Ich praktiziere das hier vorgestellte Modell seit einigen Versionen. Daher kann ich bereits auf bestehende Richtlinien zurückgreifen:


Gruppenrichtlinienobjekte in ws.its

Inhalt | Delegierung

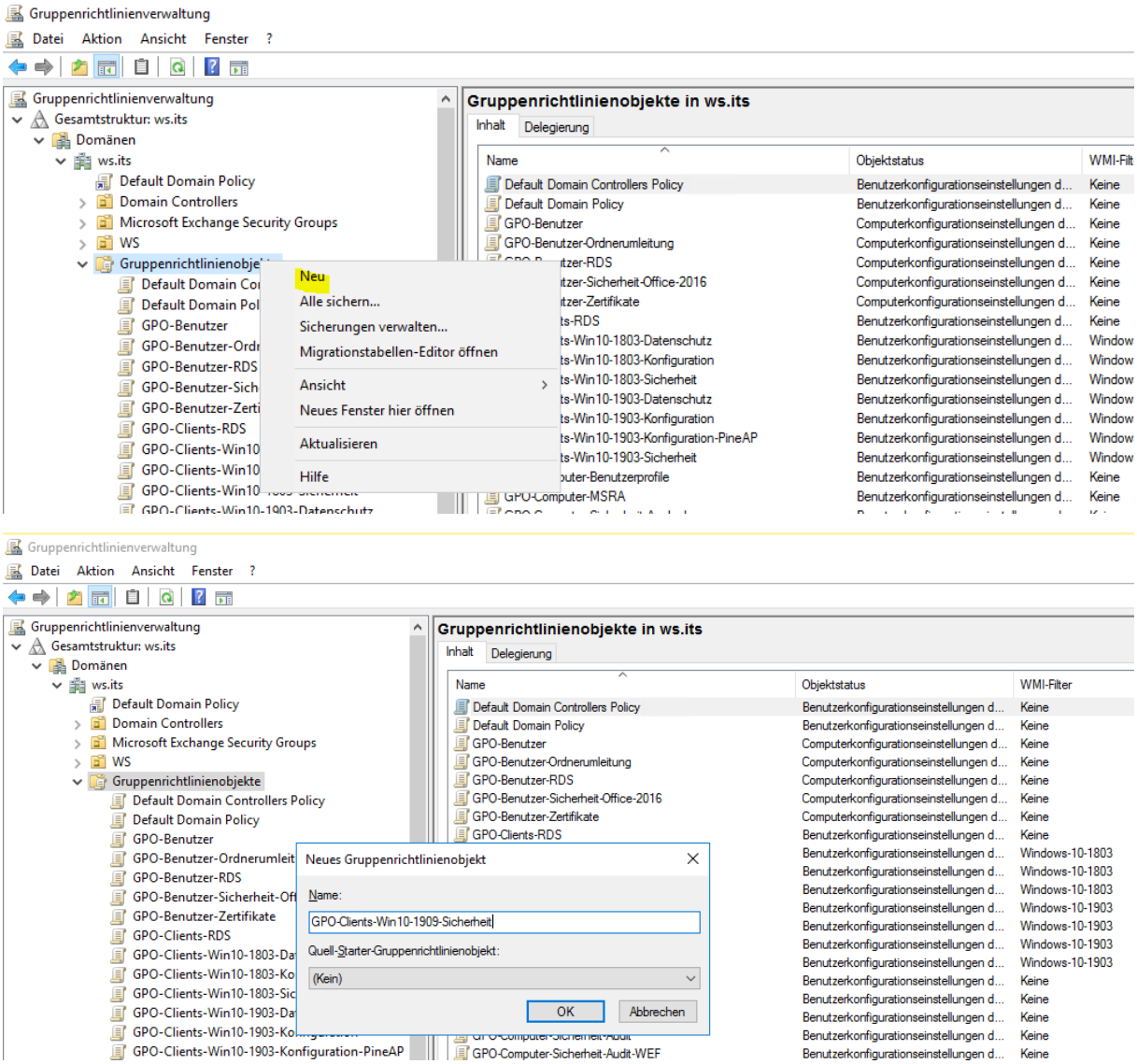
Name	Objektstatus	WMI-Filter	Geändert	Besitzer
Default Domain Controllers Policy	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
Default Domain Policy	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Benutzer	Computerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Benutzer-Ordnerumleitung	Computerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Benutzer-RDS	Computerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Benutzer-Sicherheit-Office-2016	Computerkonfigurationseinstellungen d...	Keine	15.11.2019 18:55:54	admin-setup (admin-setup@ws.its)
GPO-Benutzer-Zertifikate	Computerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Clients-RDS	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Clients-Win10-1803-Datenschutz	Benutzerkonfigurationseinstellungen d...	Windows-10-1803	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Clients-Win10-1803-Konfiguration	Benutzerkonfigurationseinstellungen d...	Windows-10-1803	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Clients-Win10-1803-Sicherheit	Benutzerkonfigurationseinstellungen d...	Windows-10-1803	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Clients-Win10-1903-Datenschutz	Benutzerkonfigurationseinstellungen d...	Windows-10-1903	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Clients-Win10-1903-Konfiguration	Benutzerkonfigurationseinstellungen d...	Windows-10-1903	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Clients-Win10-1903-Konfiguration-PineAP	Benutzerkonfigurationseinstellungen d...	Windows-10-1903	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Clients-Win10-1903-Sicherheit	Benutzerkonfigurationseinstellungen d...	Windows-10-1903	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Computer-Benutzerprofile	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Computer-MSRA	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Applocker	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Audit	Benutzerkonfigurationseinstellungen d...	Keine	01.12.2019 16:54:18	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Audit-WEF	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Basics	Benutzerkonfigurationseinstellungen d...	Keine	05.12.2019 17:45:24	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Bitlocker	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Cipher-TLS	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-DC	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Defender	Benutzerkonfigurationseinstellungen d...	Windows-Server-2016	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-DeviceGuard	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Firefox	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Firewall	Benutzerkonfigurationseinstellungen d...	Keine	25.11.2019 09:31:18	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-IExplore	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:40	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-LAPS-Clients	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-LAPS-Server	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:42	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-LSAProtection	Benutzerkonfigurationseinstellungen d...	Keine	01.12.2019 17:05:20	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-Netzwerk	Benutzerkonfigurationseinstellungen d...	Keine	27.10.2019 17:32:38	Domänen-Admins (WS\Domäne...
GPO-Computer-Sicherheit-NoNTLM	Benutzerkonfigurationseinstellungen d...	Keine	05.12.2019 17:47:12	Domänen-Admins (WS\Domäne...

Ich habe für jede Betriebssystemversion 3 Gruppenrichtlinien für die Basiskonfiguration:

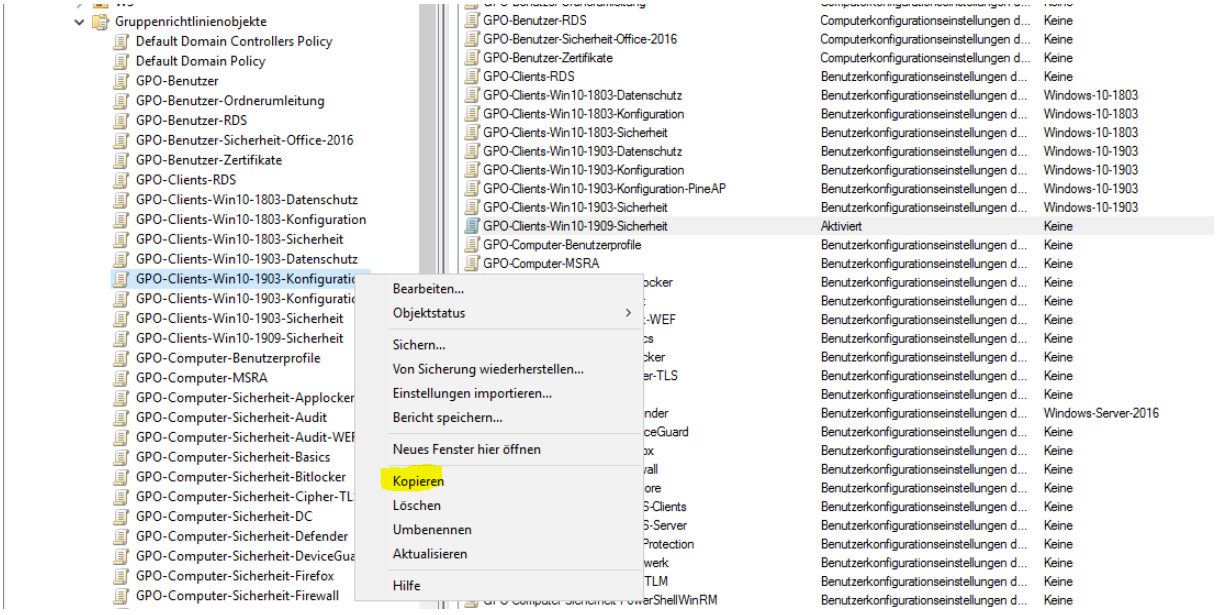
- **GPO-<OSVersion>-Sicherheit**
  - Diese GPO ist ein 1:1 Import der von Microsoft empfohlenen Security Baseline des Security Compliance Toolkits (SCT). Diese werden für jedes Betriebssystem herausgegeben.
  - Ich werde in dieser GPO keine Anpassungen vornehmen! Denn mit einem Wechsel auf ein neues Betriebssystem müsste ich herausfinden, welche Veränderungen ich vorgenommen hatte… Viel Spass dabei! Anpassungen kommen in die dritte GPO.
- **GPO-<OSVersion>-Datenschutz**
  - Die Security-Empfehlungen von Microsoft entsprechen nicht meinen Datenschutzanforderungen. Damit ich diese für Anforderungen aus der DSGVO und ähnlichen Regelwerken separat ausweisen kann, konfiguriere ich diese Einstellungen in einer separaten GPO.
- **GPO-<OSVersion>-Konfiguration**
  - Hier editiere ich alle funktionalen Einstellungen, wie z.B. die WSUS-Konfiguration
  - Auch Anpassungen des Betriebssystems gehören hier rein
  - Sollte eine Sicherheitseinstellung aus der SCT-Baseline zu streng oder zu schwach sein, dann platziere ich die Korrektur in dieser GPO

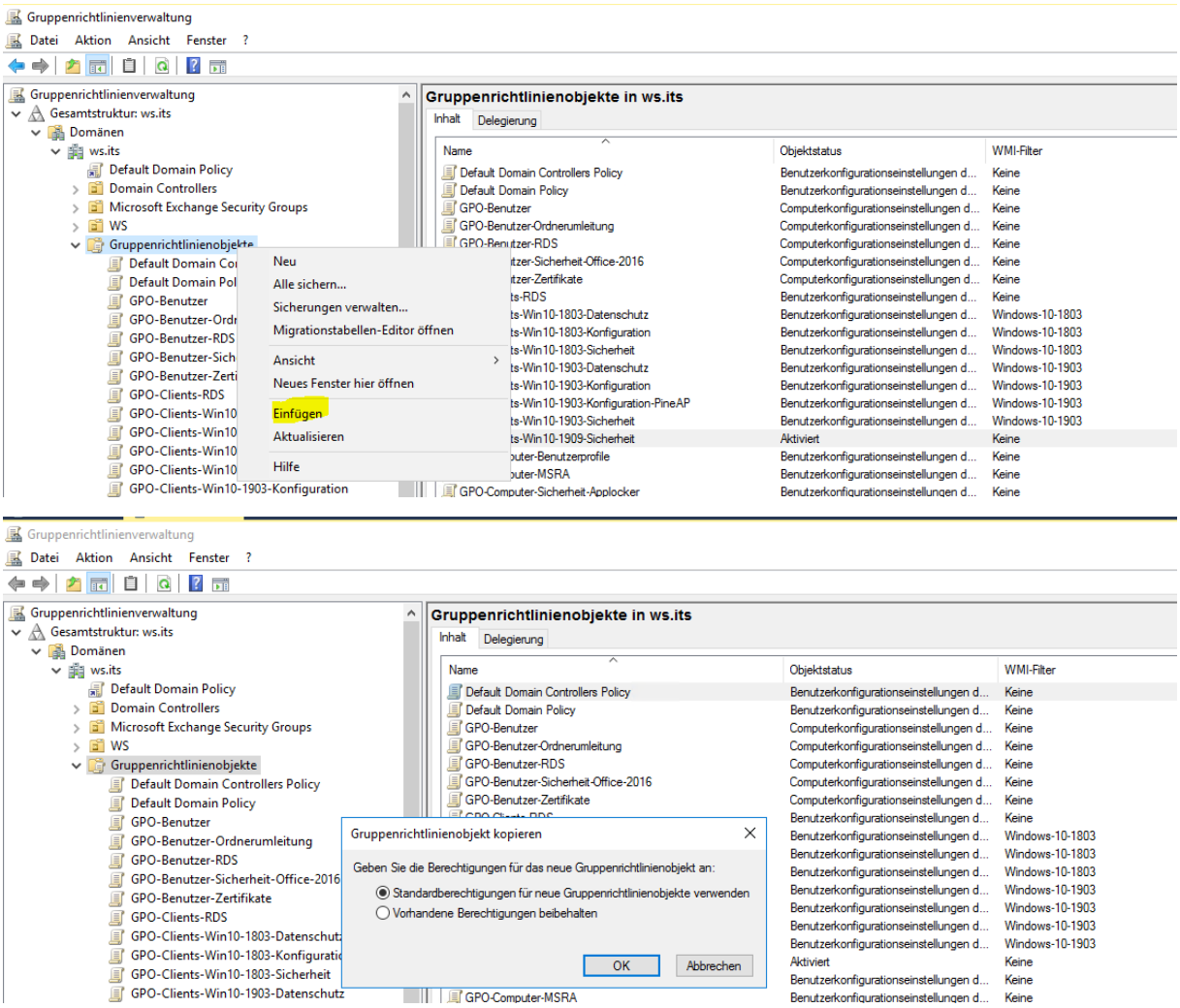
Wenn ihr heute auf dieses Schema umsteigen wollt, dann beginnt ihr mit leeren GPO. Wenn ihr aber nur für ein neues Betriebssystem erweitert (so wie ich hier in diesem WSHowTo), dann könnt ihr die bestehenden GPO kopieren und anpassen.

Ich benötige drei neue GPO. Die erste ist die einfachste, denn ihre Einstellungen importiere ich von der Microsoft-Baseline. Dafür benötige ich eine neue, leere GPO:

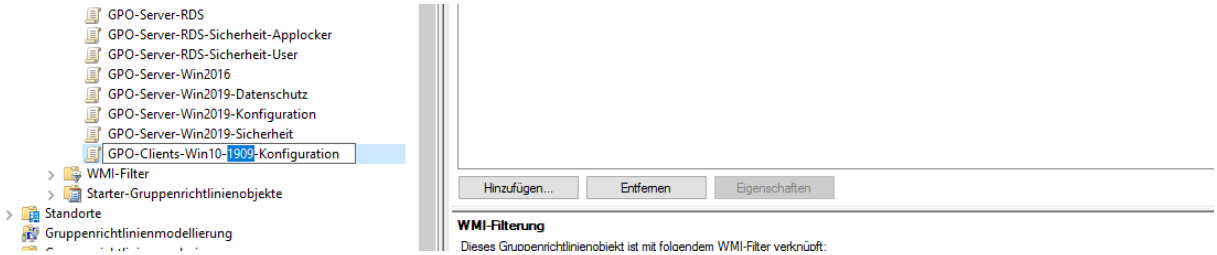


Die beiden anderen GPO (Konfiguration und Datenschutz) habe ich bereits in der Vorgängerversion. Für beide GPO kann ich also eine Kopie erstellen (wie gesagt: wer neu einsteigt, der muss 2 leere GPO erstellen und manuell befüllen):

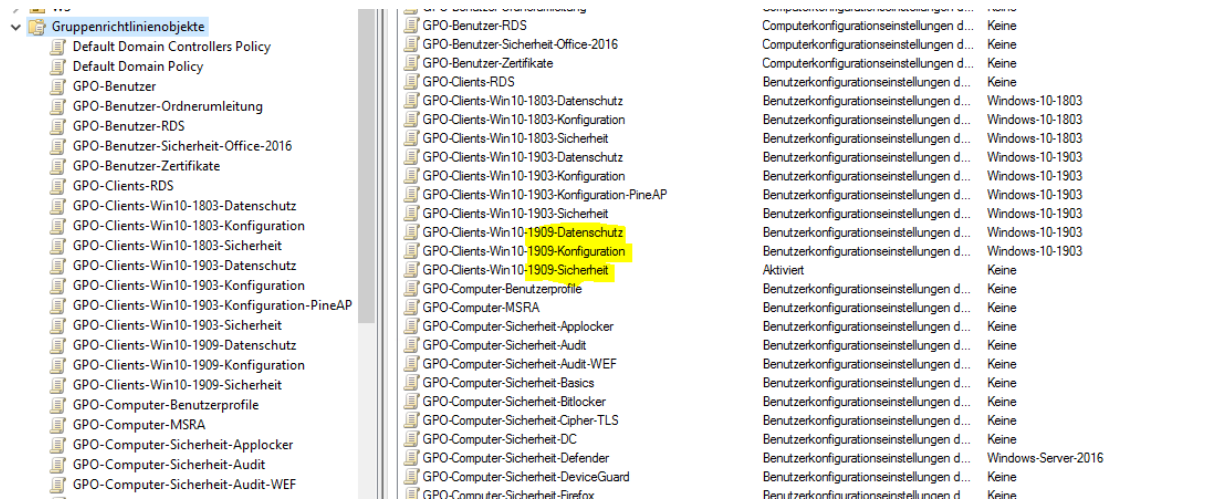




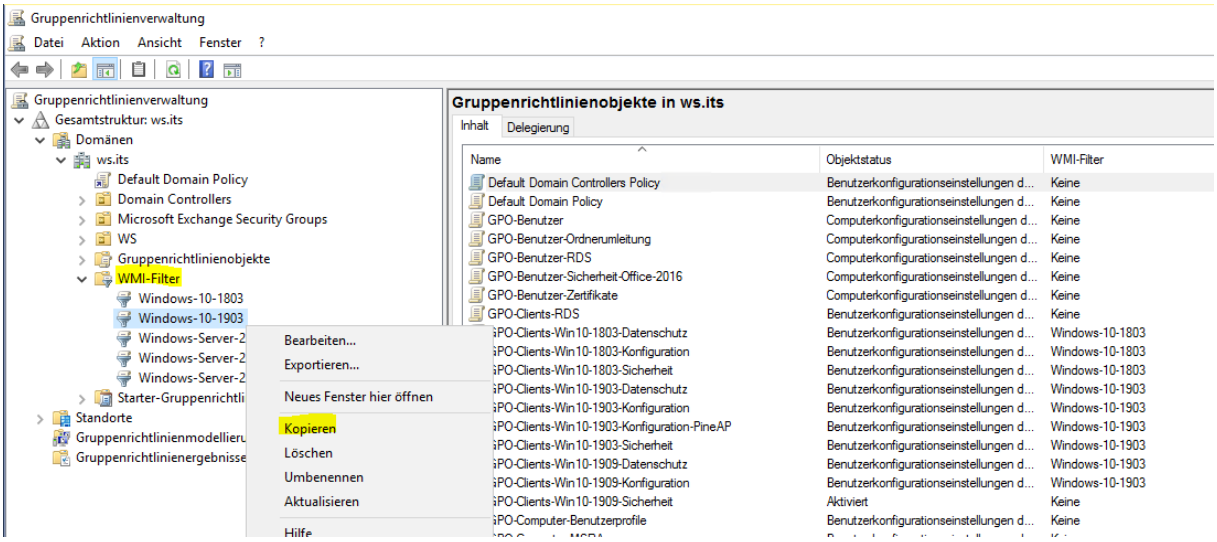
Die erste Kopie benenne ich entsprechend um:

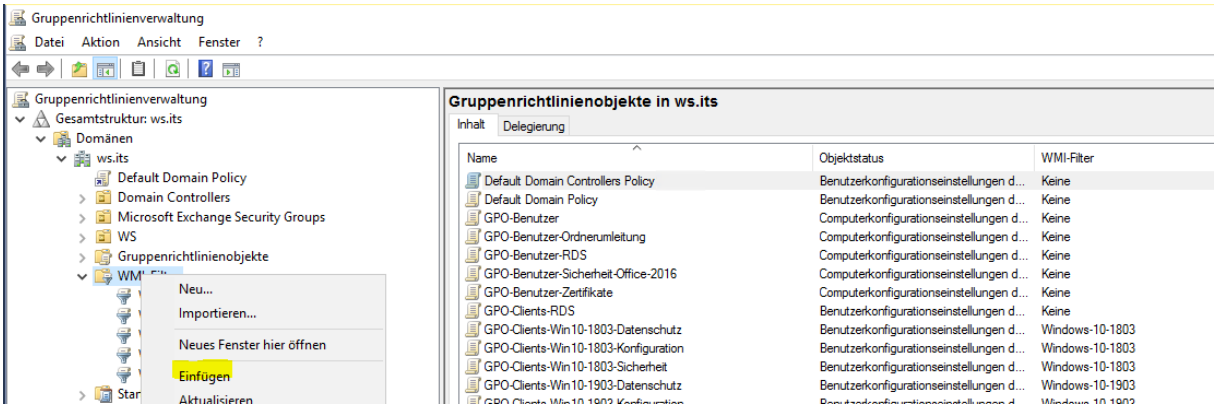


Für die Datenschutz-GPO erstelle ich nach dem gleichen Verfahren eine Kopie. Im Ergebnis sehe ich die drei neuen GPO:

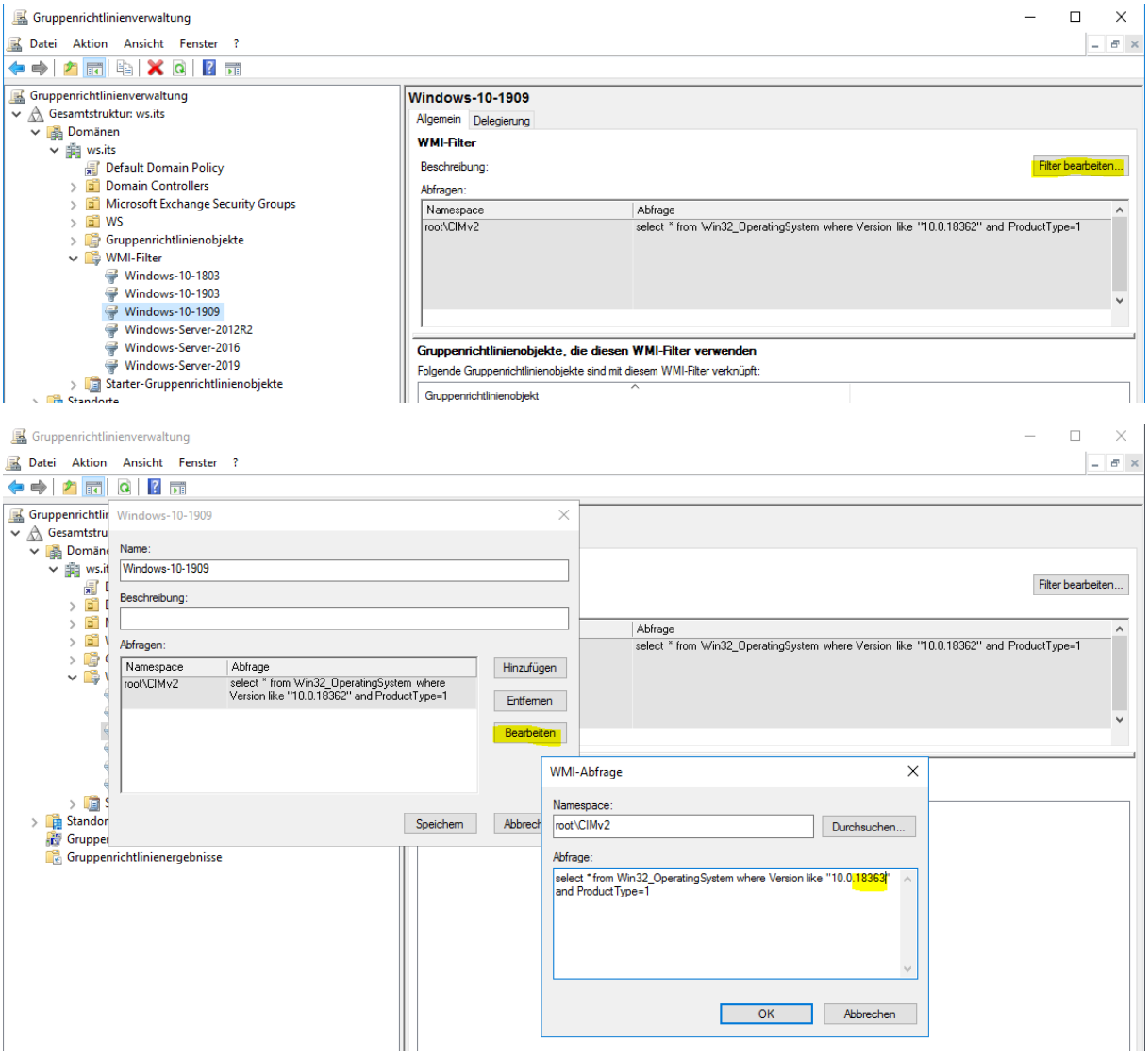


Damit diese Richtlinien nur von Clients mit Windows 10 Version 1909 verarbeitet werden, benötige ich einen WMI-Filter. Auch hier kann ich einen Vorgänger kopieren. Wichtig ist die WMI-Abfrage – man kann also auch einfach einen neuen Filter ohne Kopie erstellen:

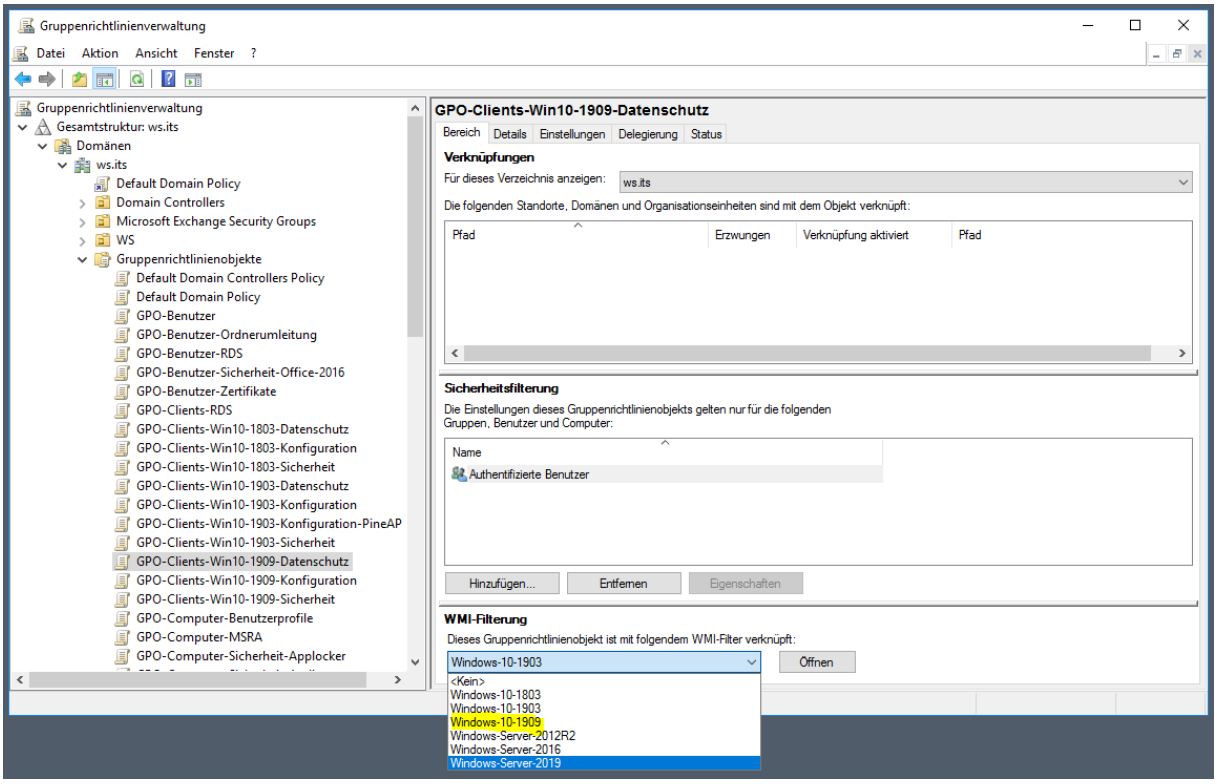




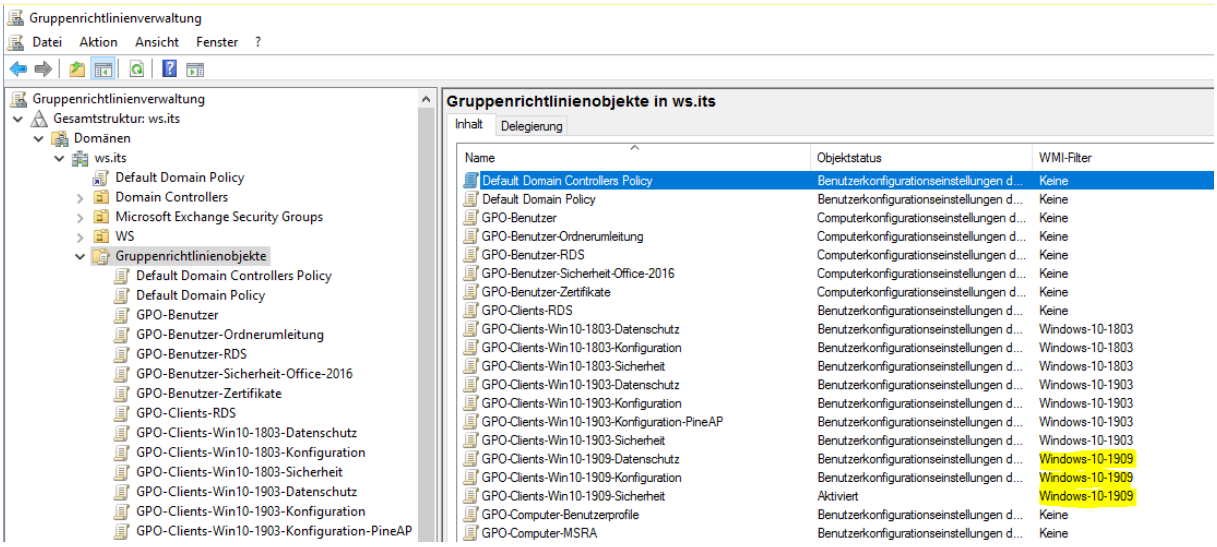
Nach dem Umbenennen passe ich den Filter an. Die Versionsnummer lese ich mit winver.exe aus:



Jetzt kann ich den neuen WMI-Filter den neuen GPO zuweisen:

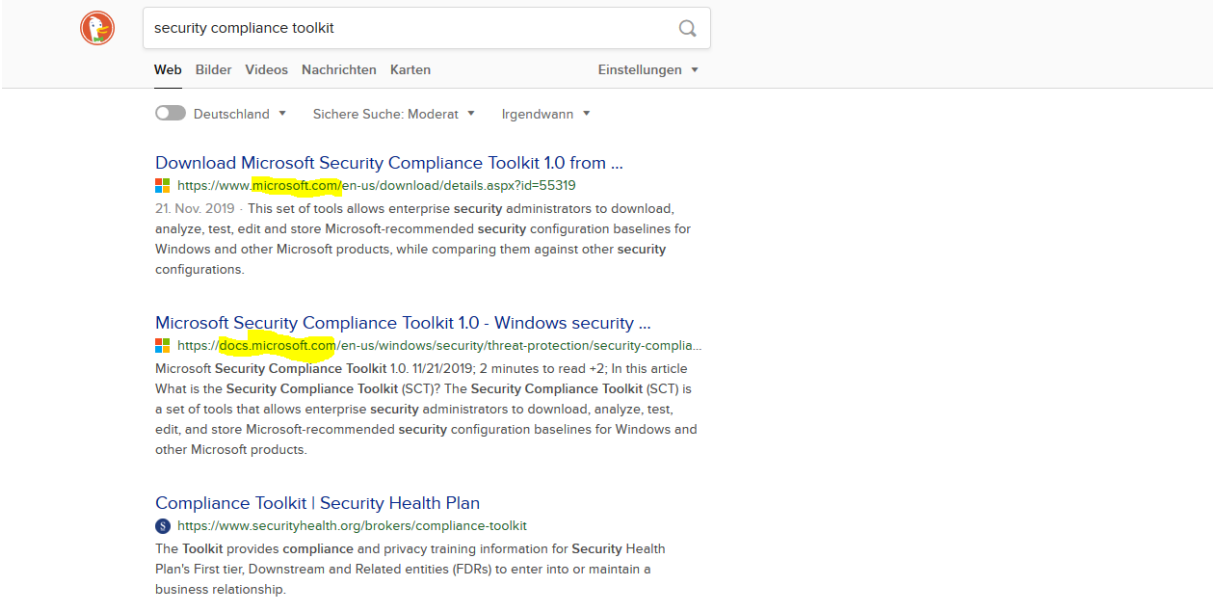


Jetzt sind die GPO bereit für den inhaltlichen Abgleich:

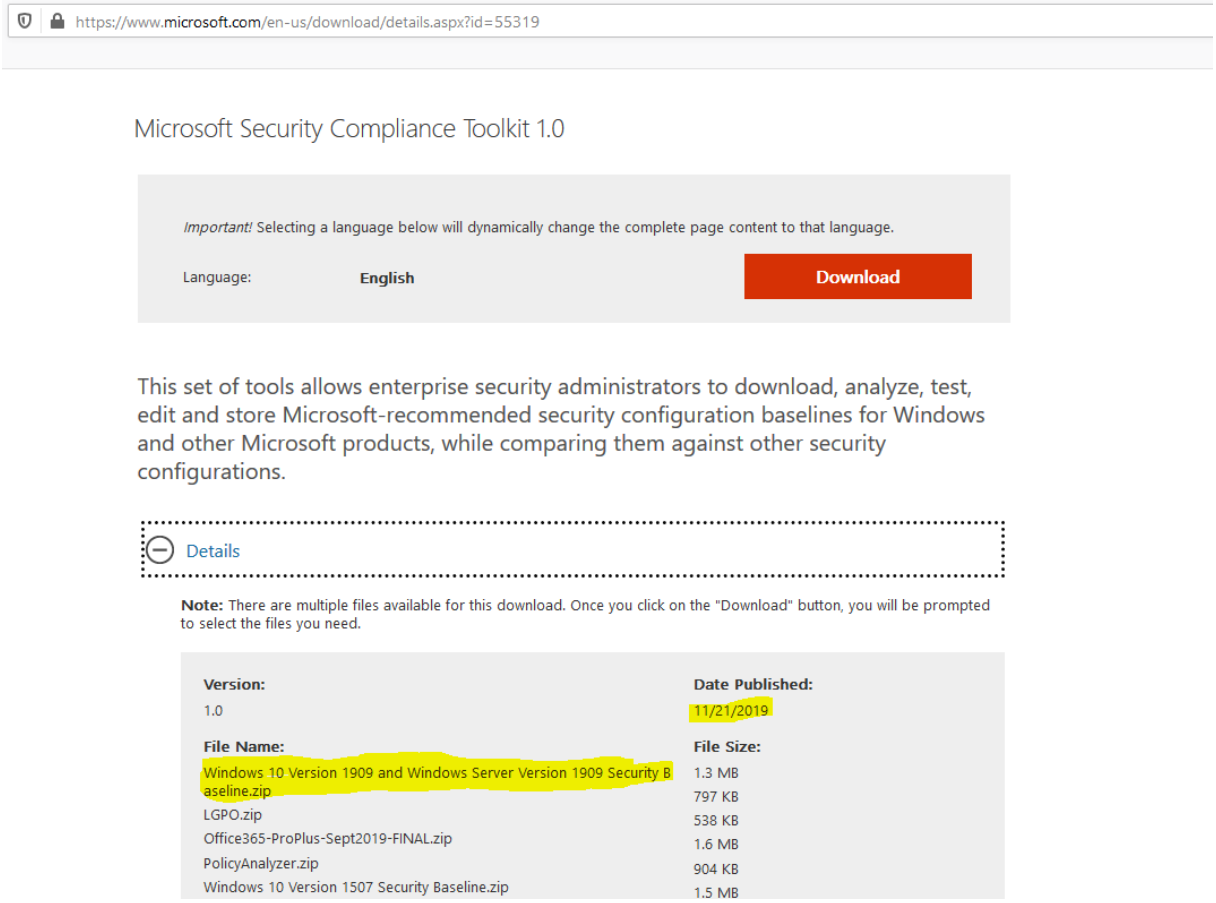


GPO Sicherheit (SCT-Baseline)

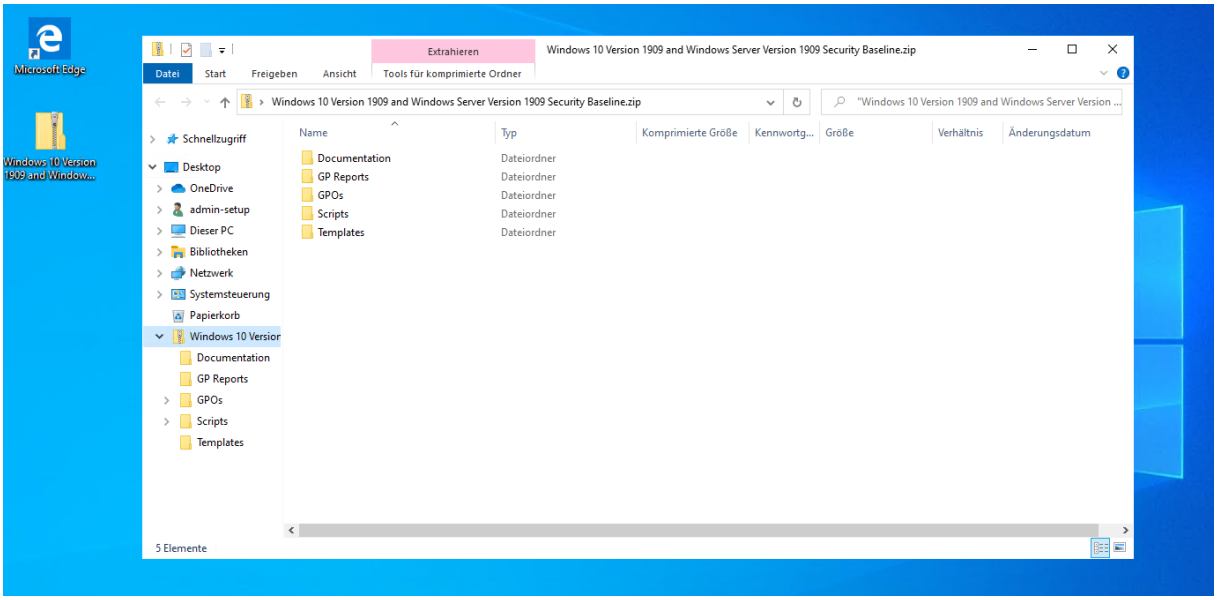
Die Sicherheitseinstellungen empfiehlt Microsoft für jede Betriebssystemversion. Die Qualität der Empfehlungen schwankt durchaus. Daher sollte man die Einstellungen immer testen. Man sucht die neusten Releases online. Achtet hier bitte auf die Quelle der Informationen: die Seite sollte schon von Microsoft kommen:



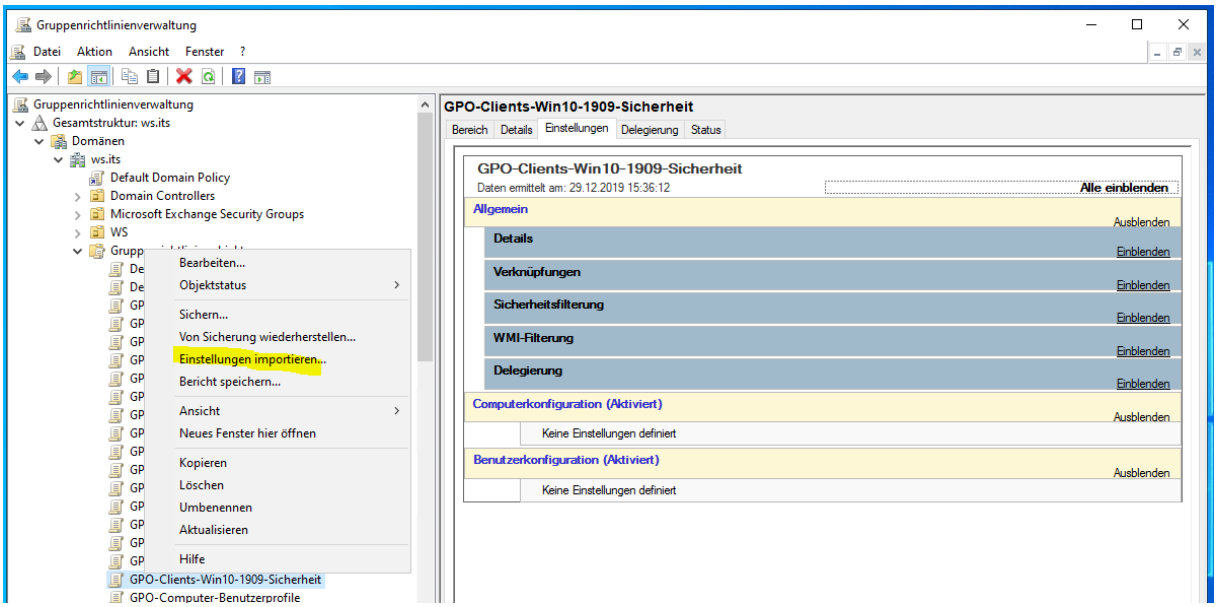
Der anschließende Download ist einfach. Achtet bitte auf das Release-Datum und eventuelle „Drafts“ (Entwürfe):

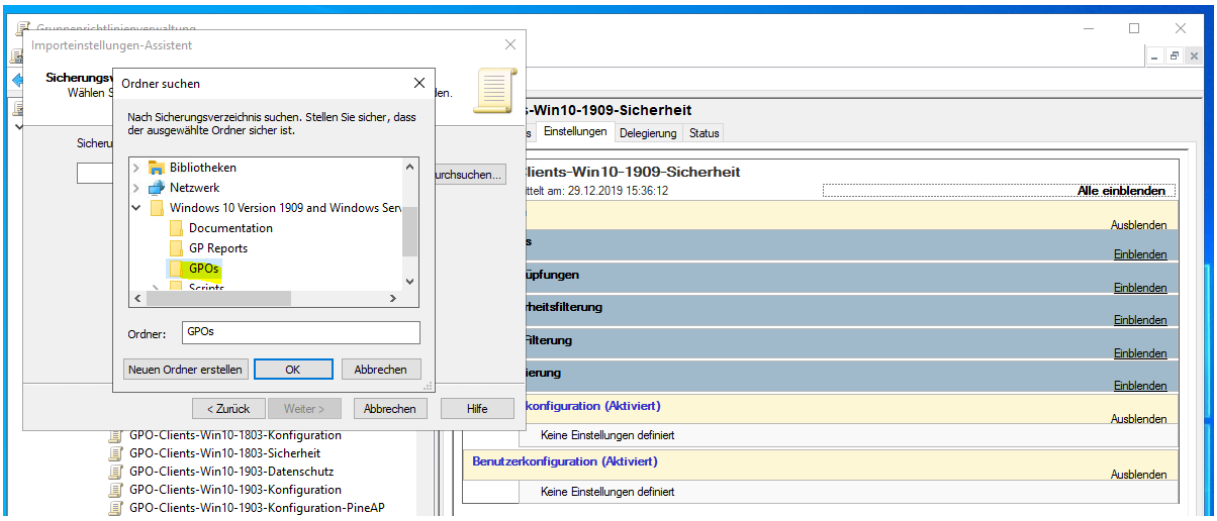


In dem ZIP-Archiv befinden sich etliche Unterordner. Der Ordner „GPO“ enthält die exportieren Richtlinien:

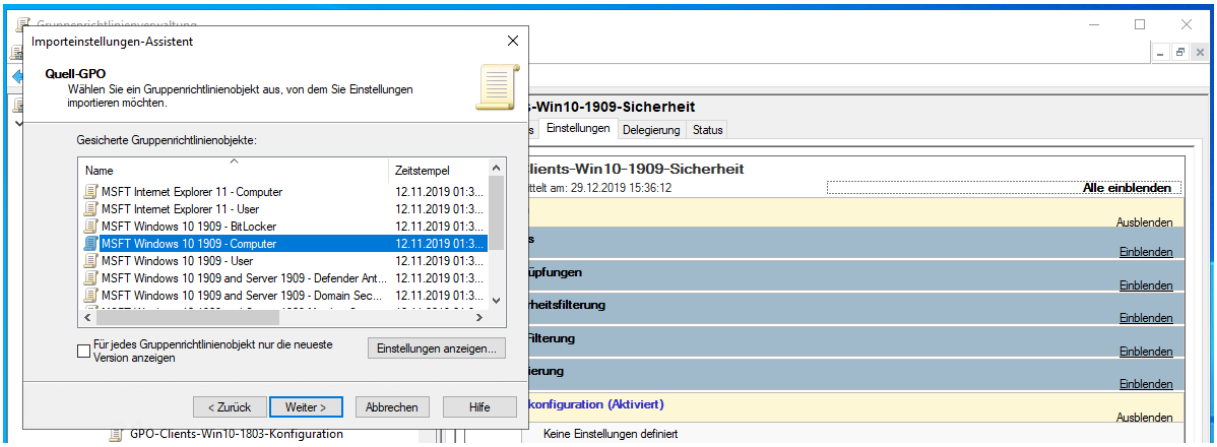


Diese Exports kann man nun einfach importieren:

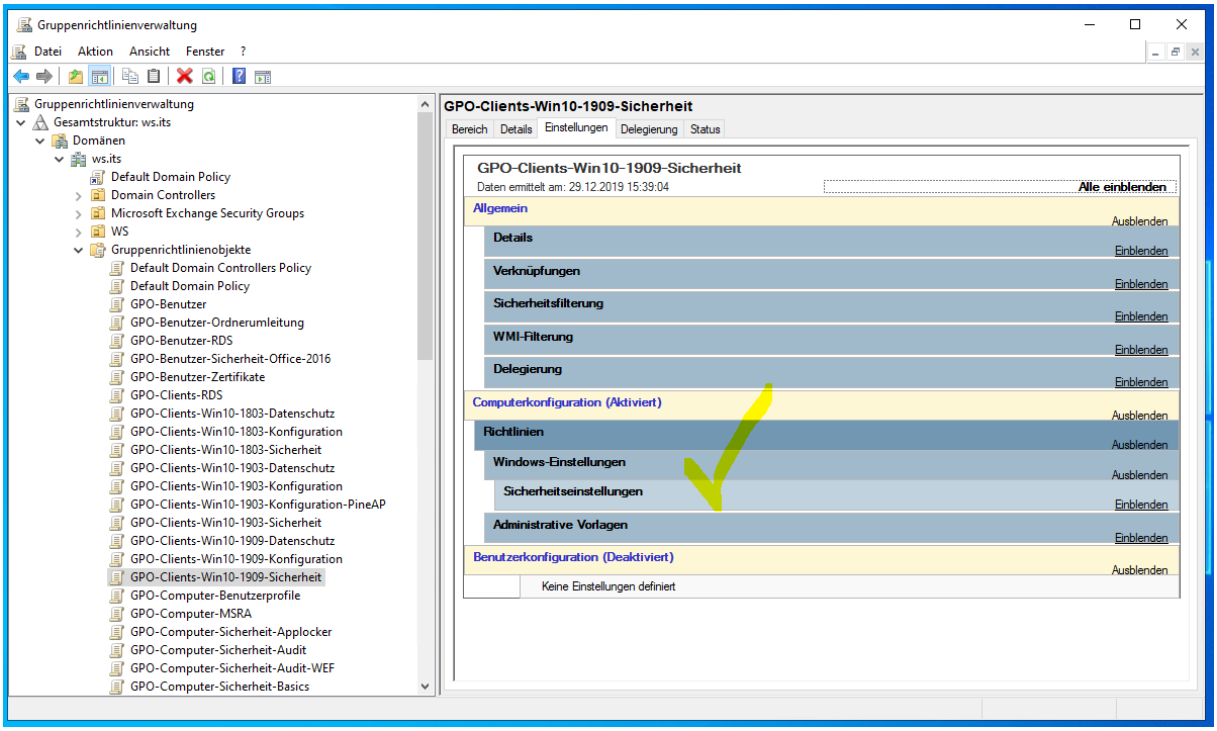




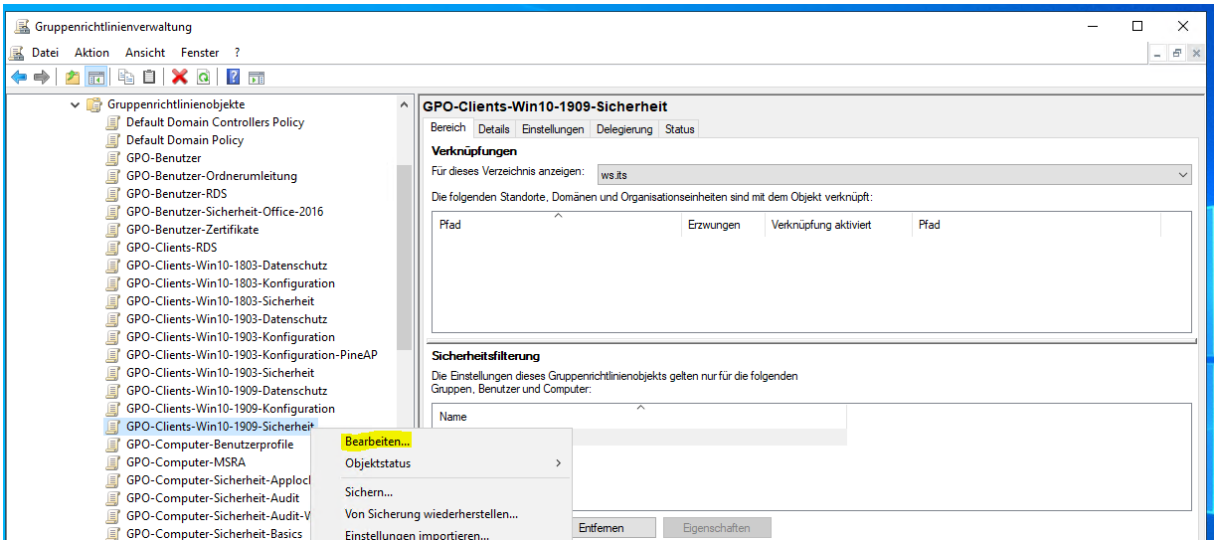
Nach der Auswahl des Verzeichnisses sieht man die einzelnen GPO mit ihrem Namen. Mein Zielsystem ist ein Client. Daher wähle ich diese Richtlinie aus:



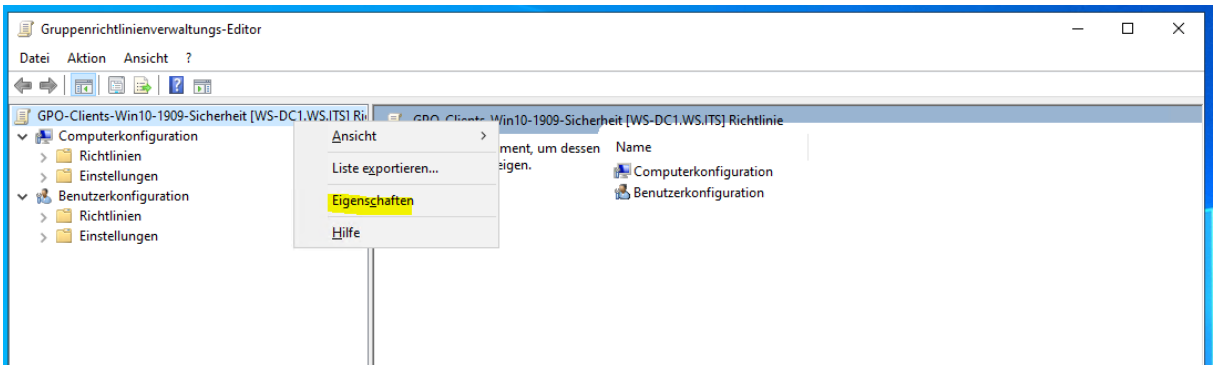
Wenige Klicks später ist der Export in die leere GPO importiert:



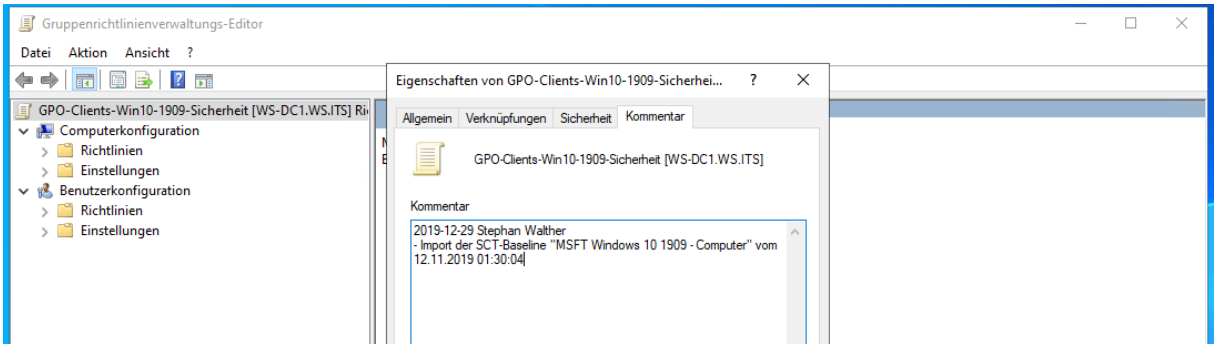
Ich kommentiere gerne die Richtlinien. Daher öffne ich die neue GPO mit dem Editor:



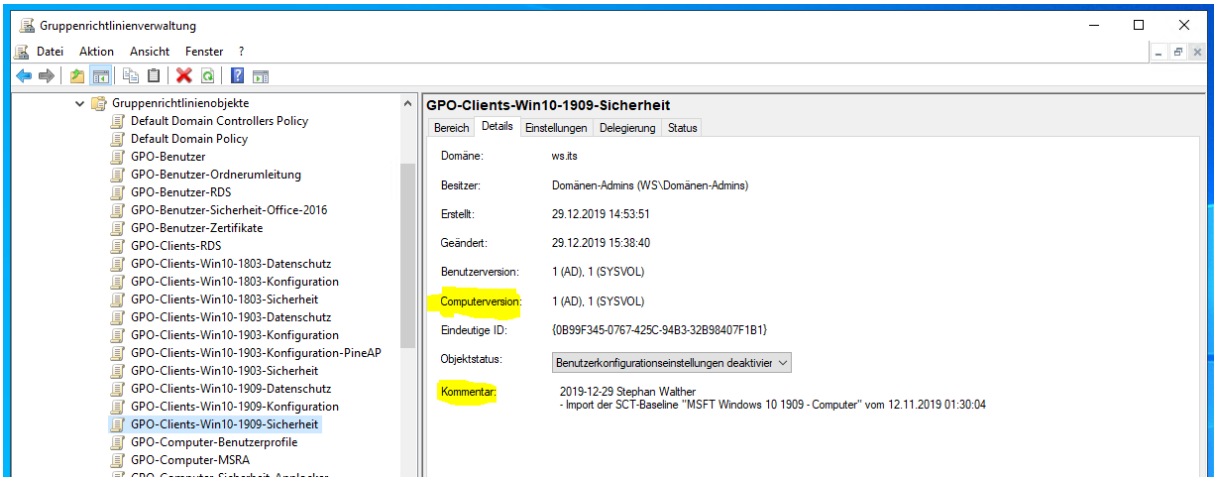
In den Eigenschaften geht es weiter:



Hier vermerke ich mir genau die Informationen der originalen GPO mit dem dazugehörigen Release-Date. Falls es später eine neue Version gibt, kann ich das eindeutig nachvollziehen. Das Datum und den Namen übernehme ich aus dem „Quell-Import“-Dialog des Importes:



An der Computerversion kann ich genau erkennen, dass diese GPO nach dem Import nicht verändert wurde. Jede Anpassung würde den Zähler nach oben korrigieren:



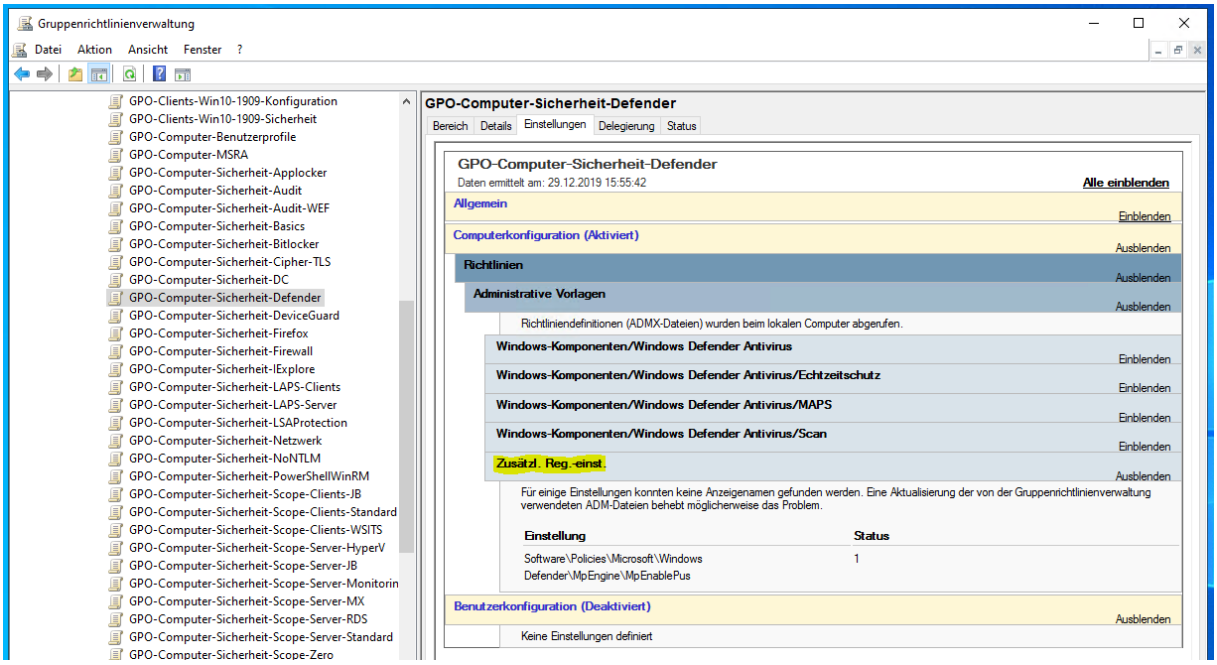
Damit wäre die erste GPO einsatzbereit – von einer Testphase mal abgesehen.

GPO Datenschutz

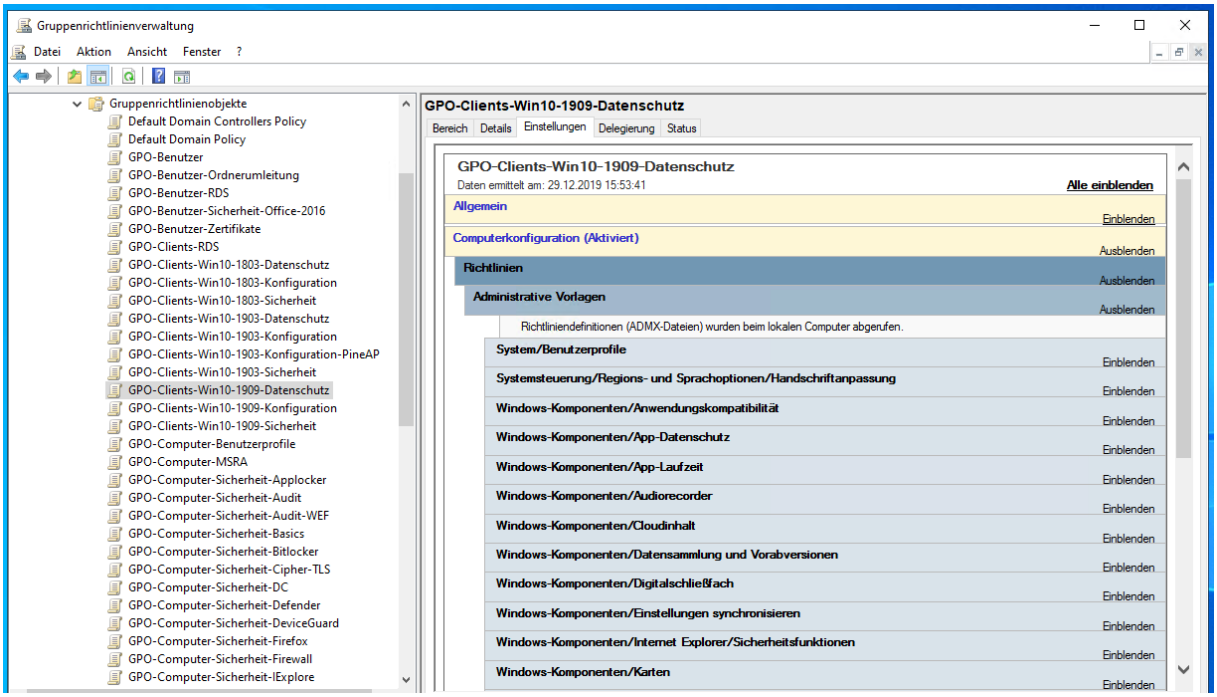
Die zweite GPO ist eine Kopie der Vorgänger-Richtlinie für Windows 10 Version 1903. Für mich sind hier 2 grundsätzliche Aktionen wichtig:

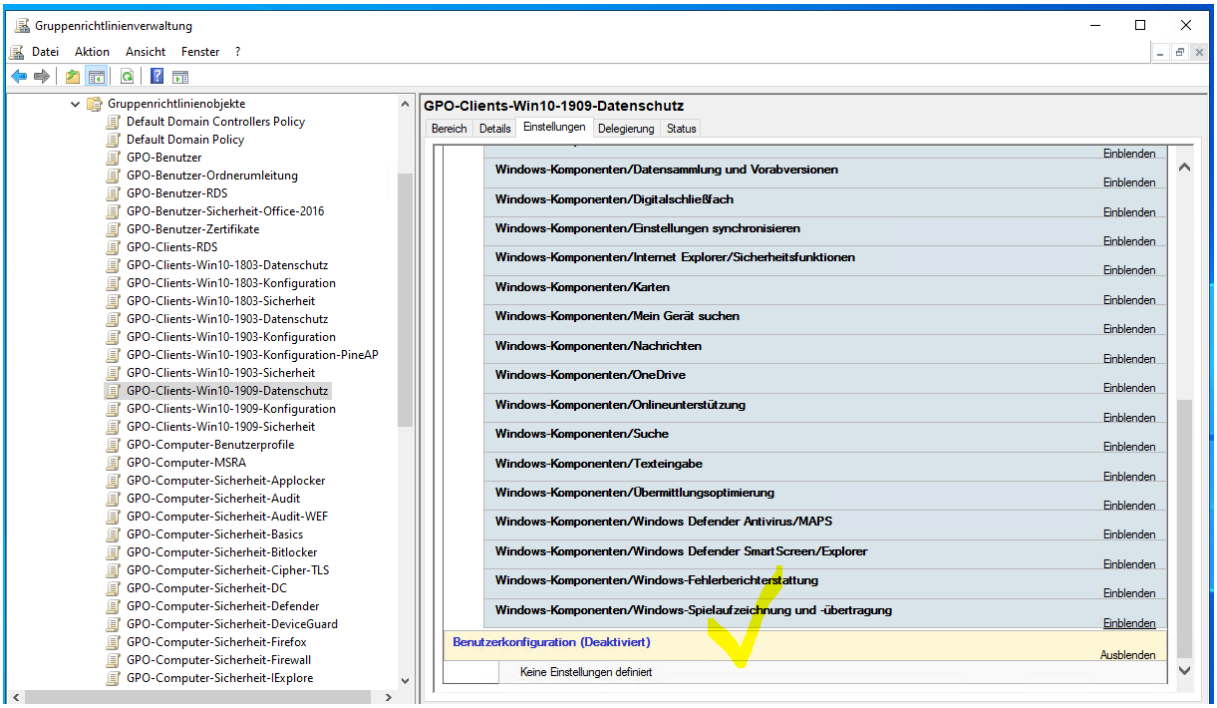
- Ich muss nach nicht mehr kompatiblen Einstellungen suchen und diese für 1909 entfernen
- Ich muss aber auch nach neuen Einstellungen suchen, die es für 1903 noch nicht gab.

Beides erreiche ich, indem ich die GPO mit einem Editor-Rechner bearbeite, der die neuen Vorlagen verwendet. Inkompatible Einstellungen finde ich unter „Extra Registry Settings“. Hier seht ihr ein Beispiel für eine Problem-GPO:



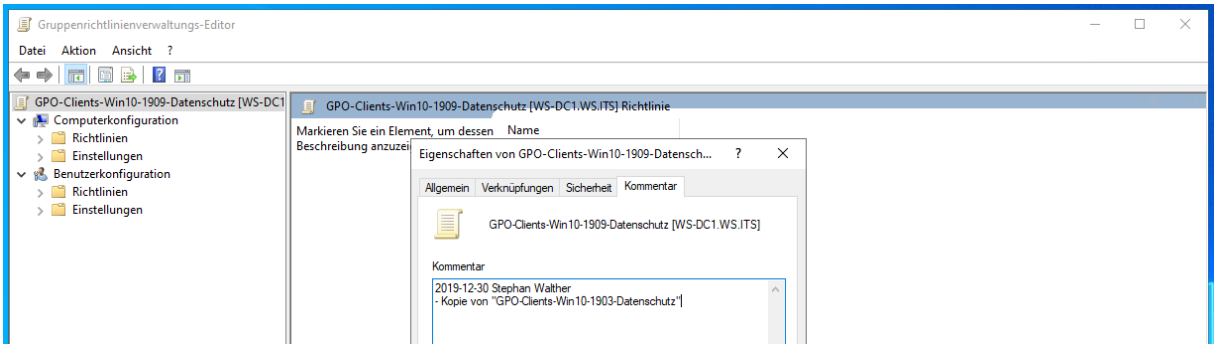
Fehlt dieser Punkt, dann besteht eine gute Chance auf Kompatibilität:



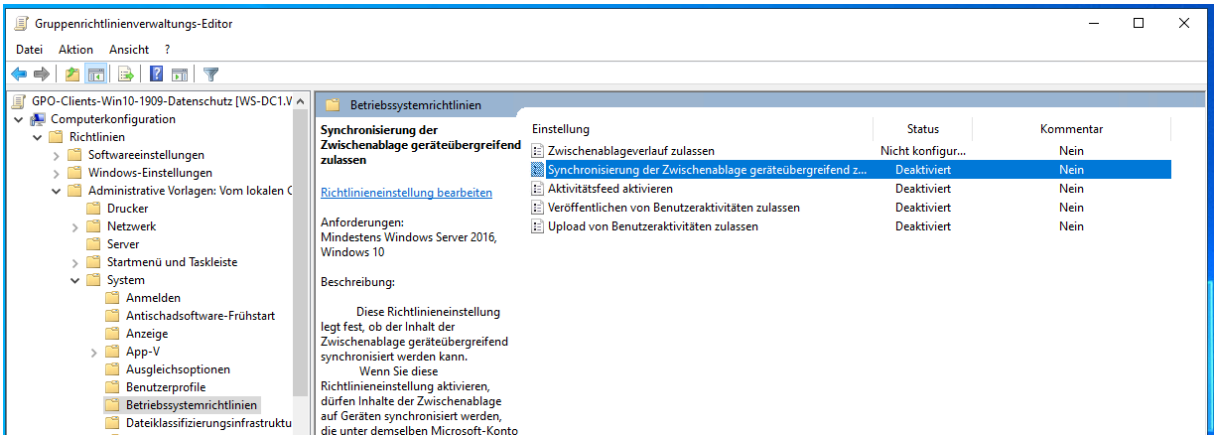


Sehr gut: Der zusätzliche Punkt fehlt. Natürlich kann es jetzt immer noch zu den Problemen wie in den Szenarien 2 und 3 kommen. Aber diese kann ich anders prüfen. Dazu später mehr.

Weiter geht es in der Kommentar-Sektion. Hier sollte eine Quellenangabe platziert werden:



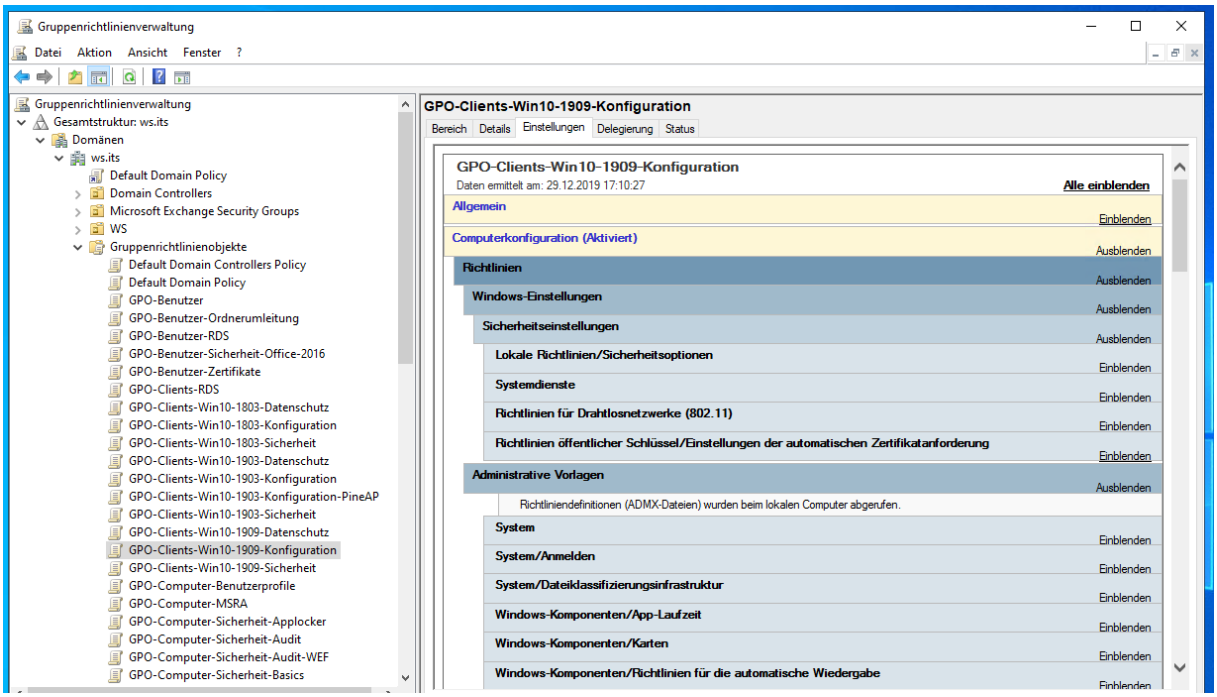
Und im nächsten Schritt gehe ich alle Einstellungen mit dem Editor durch und suche nach zusätzlichen Einstellungen zum Datenschutz. Hier kann natürlich auch eine Internet-Recherche helfen. Ich finde ein paar Einstellungen, die ich vorher noch nicht konfiguriert hatte. Das hole ich nun nach:



Die Datenschutzeinstellungen können durchaus etwas Zeit in Anspruch nehmen. Gerade in Umgebungen mit Cloud-Anbindung ist hier Fingerspitzengefühl gefragt. Microsoft hat beim Datenschutz einfach eine wenig deutsche Mentalität…

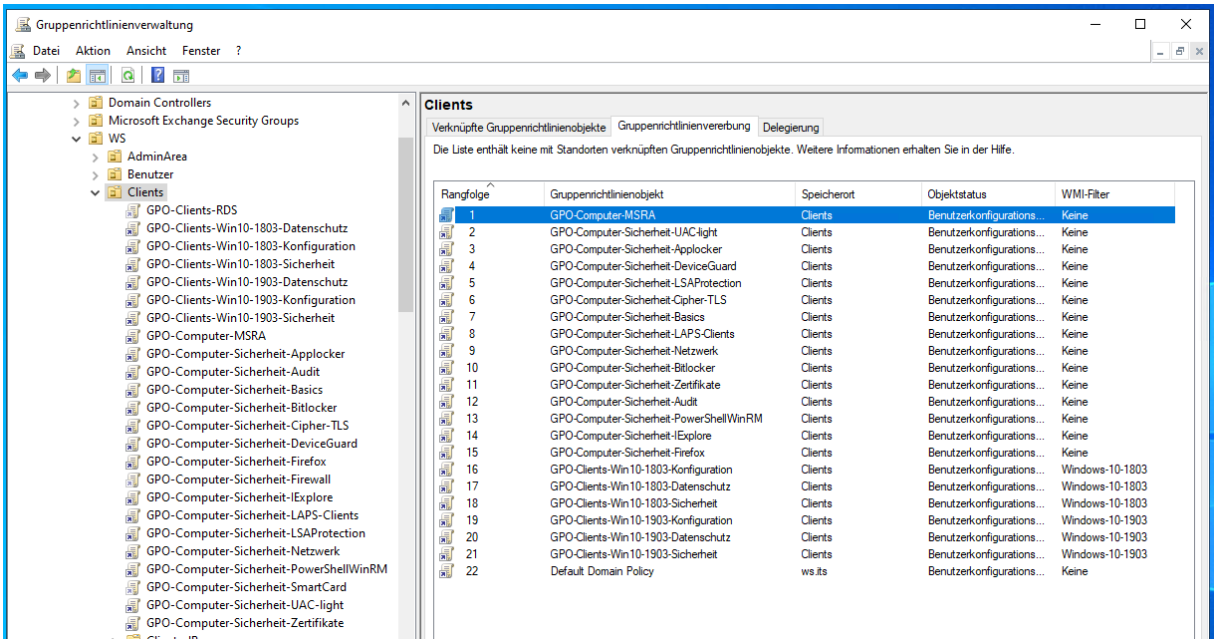
GPO Konfiguration

Weiter geht es mit der dritten GPO. Das Vorgehen entspricht dem der Datenschutz-GPO. Ich habe auch hier wenig zu tun: Es gibt keine verwaisten Einstellungen und neue Elemente sind auch nicht dazu gekommen. Der Wechsel von Windows 10 1903 nach 1909 ist sehr einfach. Das muss aber nicht so sein.

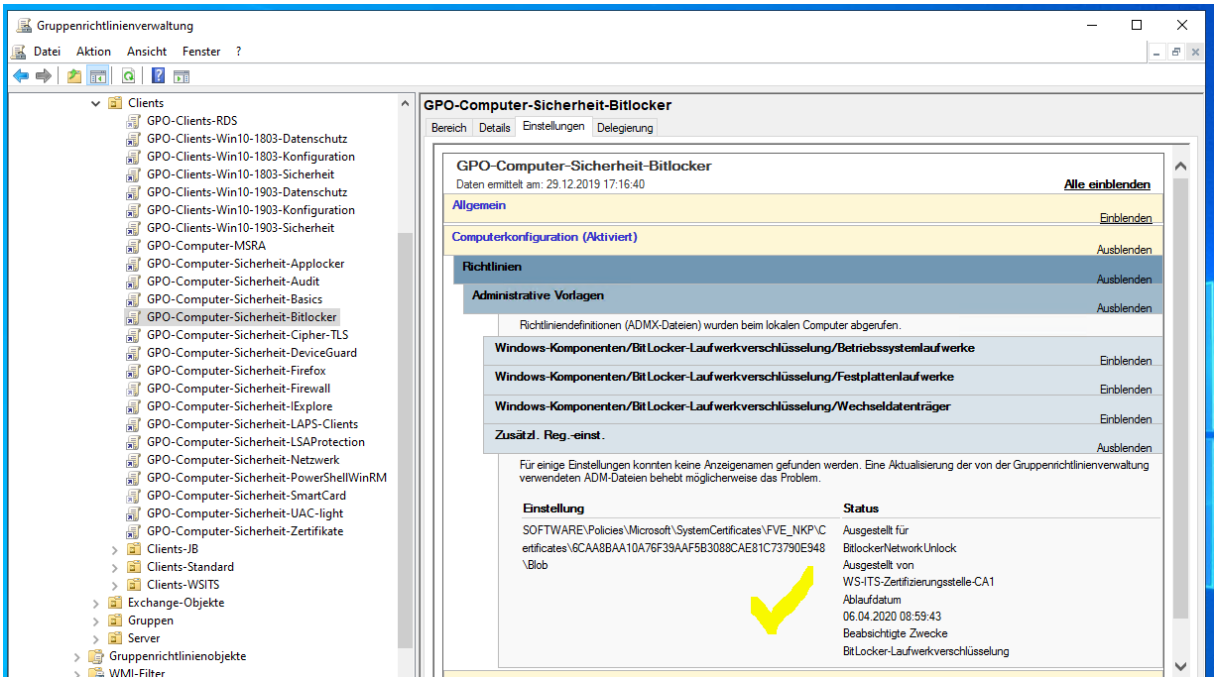


Kompatibilität sonstiger GPO

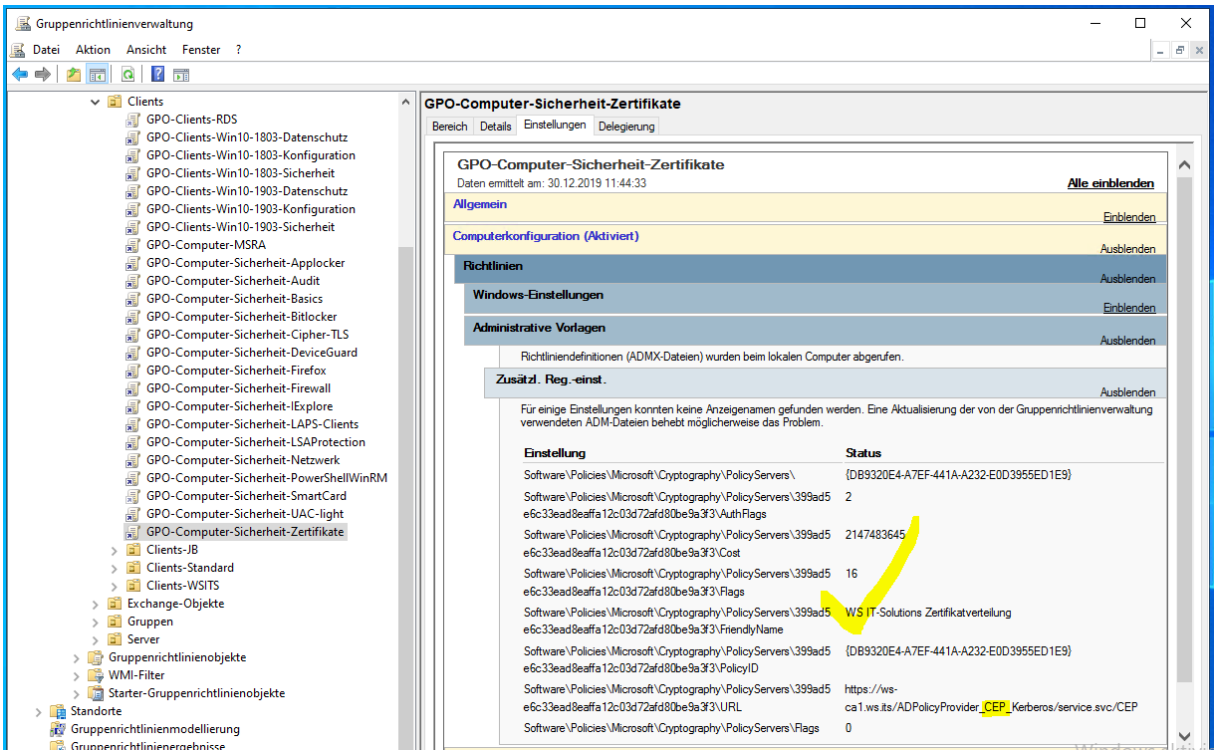
Wie man vielleicht im oberen Bild erkennen kann, verwende ich momentan nicht durchgängig das Modell mit den drei Gruppenrichtlinien je Betriebssystem. Ich habe dazu noch etliche versionsübergreifende GPO. Damit kann ich Einstellungen relativ einfach einzeln zurücknehmen (für das TroubleShooting). Aber bei einem Versionswechsel muss ich auch jede einzeln auf Kompatibilität prüfen. Das sind meine GPO:



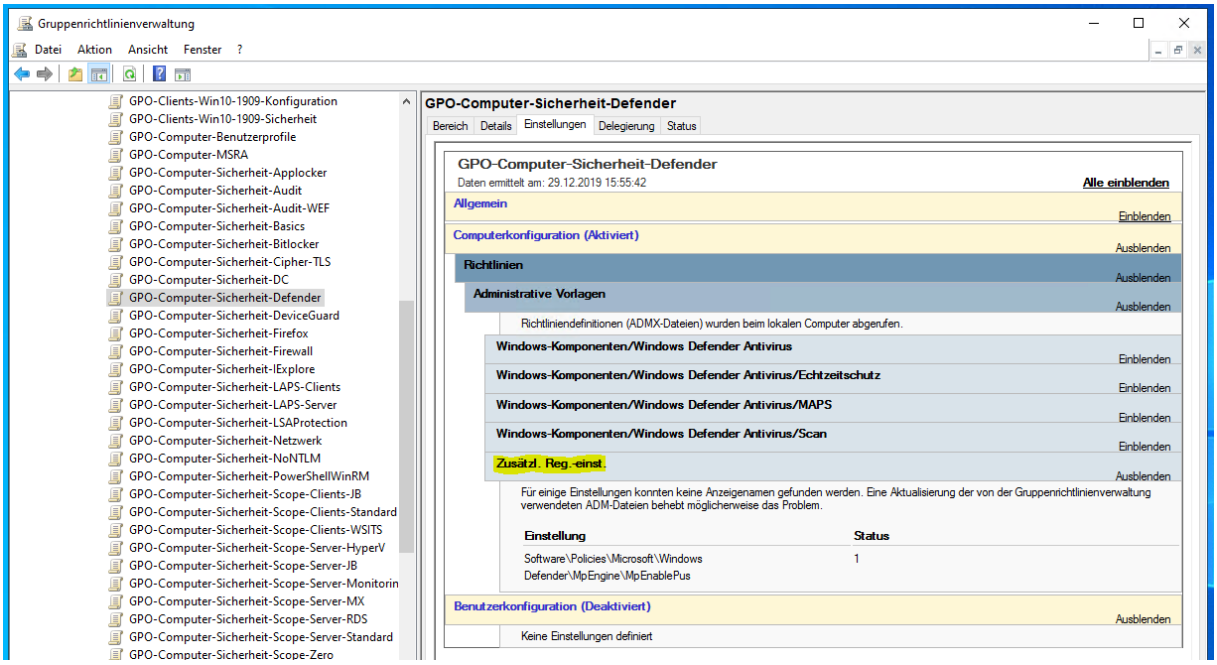
Die Validierung nehme ich direkt in der Gruppenrichtlinien-Verwaltungskonsole vor. Dazu selektiere ich eine GPO und prüfe die Anzeige in den Einstellungen. Finde ich einen „zusätzlichen Registry-Eintrag“, dann prüfe ich dessen Ursprung genau. Dieser gehört zu einer bekannten Extension: Dem Bitlocker-Network-Unlock. Dieser wird hier generell nicht richtig angezeigt. Gleichzeitig bestehen aber keine Kompatibilitätsprobleme



Gleiches gilt für die automatische Zertifikat-Verteilung. Die ist unproblematisch:



Alle anderen GPO werden korrekt angezeigt und sind damit kompatibel. Nur die Defender-GPO spielt nicht mit. Diese muss ich separat editieren:

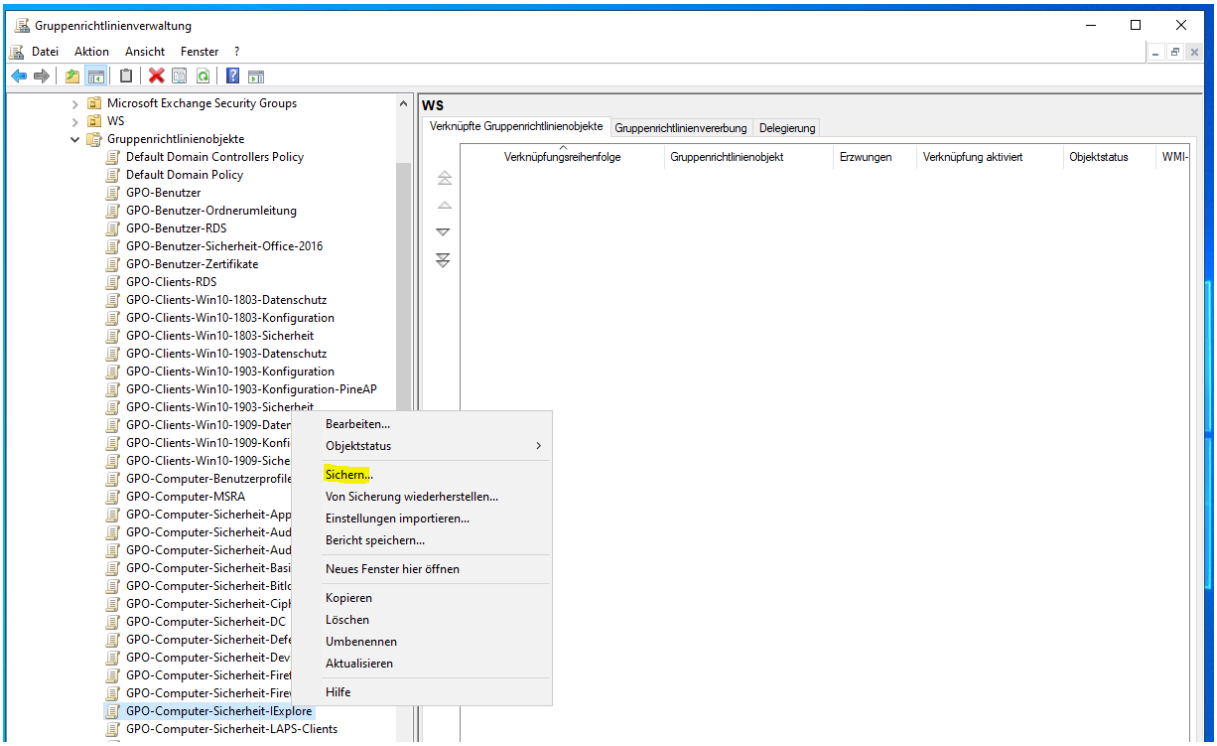


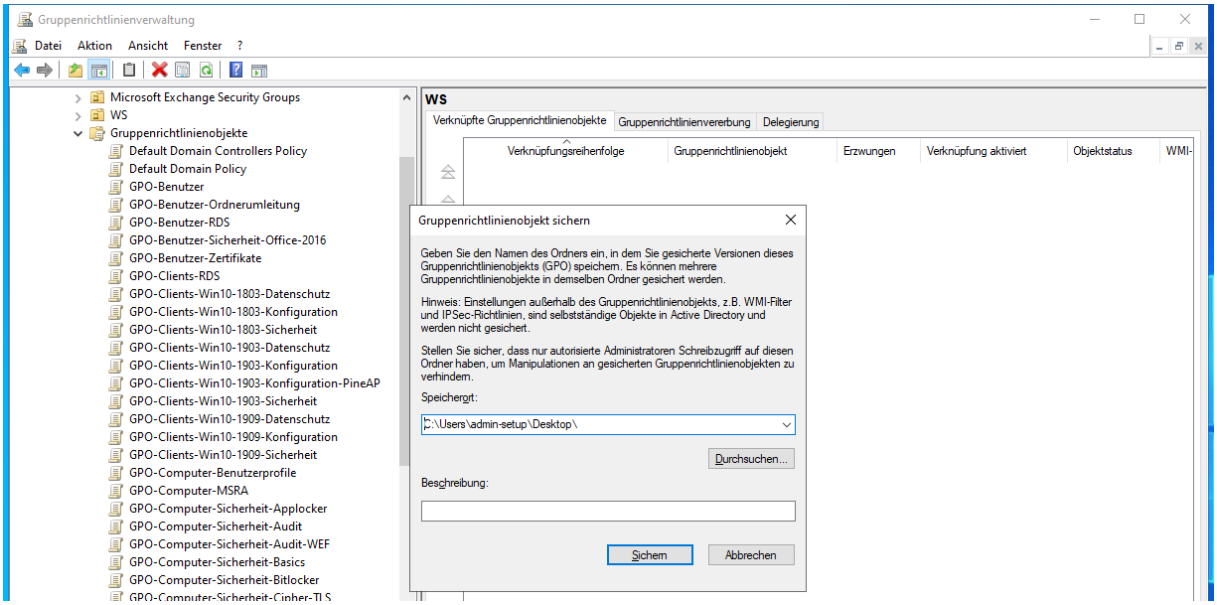
Vergleich zwischen zwei GPO mit dem PolicyAnalyzer (SCT)

Einen Sonderfall möchte ich hier aufzeigen. Ich verwende eine alte GPO für die Härtung des Internet Explorers. Diese GPO hat ihren Ursprung ebenfalls in einer Microsoft-Security-Baseline-GPO aus dem Security Compliance Toolkit. Im aktuellen Paket des SCT ist auch eine Version mit dabei. Da der Internet Explorer auch auf anderen Betriebssystemen in der Version 11 vorhanden ist, stehe ich nun vor folgender Frage: „Hat sich etwas zwischen den beiden GPO-Versionen verändert?“ Wenn sich nichts verändert hat, dann kann ich die neue Version auch ungeprüft auf die bestehenden Systeme anwenden. Wurden aber Veränderungen vorgenommen, dann muss ich einen Test der GPO durchführen.

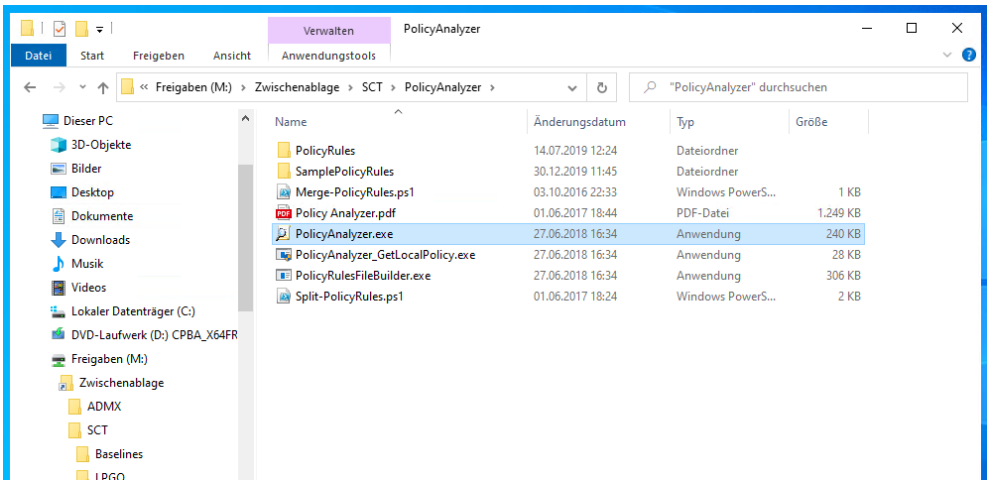
Solche Testphasen können sehr zeitintensiv sein. Schneller geht daher ein Vergleich der beiden GPO. Leider sind hier unzählige Einstellungen in beiden Versionen vorhanden. Eine manuelle Sichtung ist nicht möglich! Den Vergleich kann ich aber mit dem kostenlosen PolicyAnalyzer von Microsoft vornehmen. Der gehört zum SCT dazu.

Zuerst exportiere ich meine aktive GPO durch eine Sicherung:

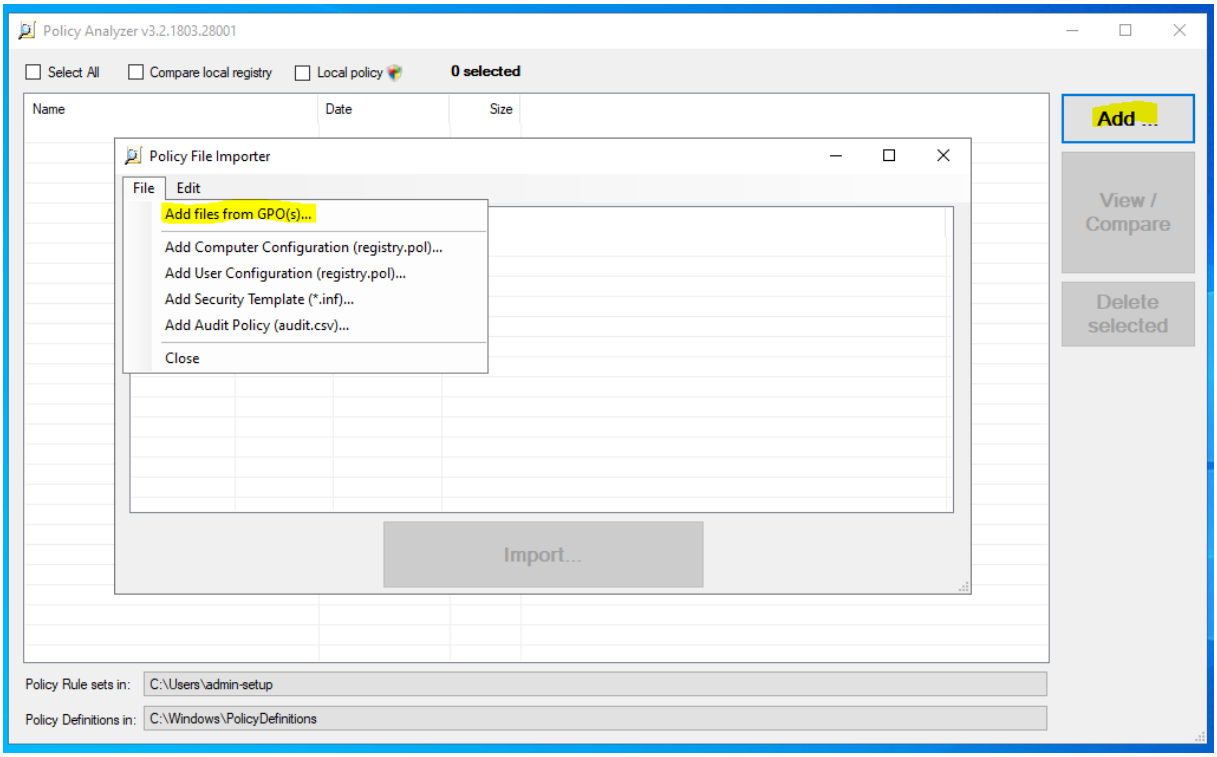




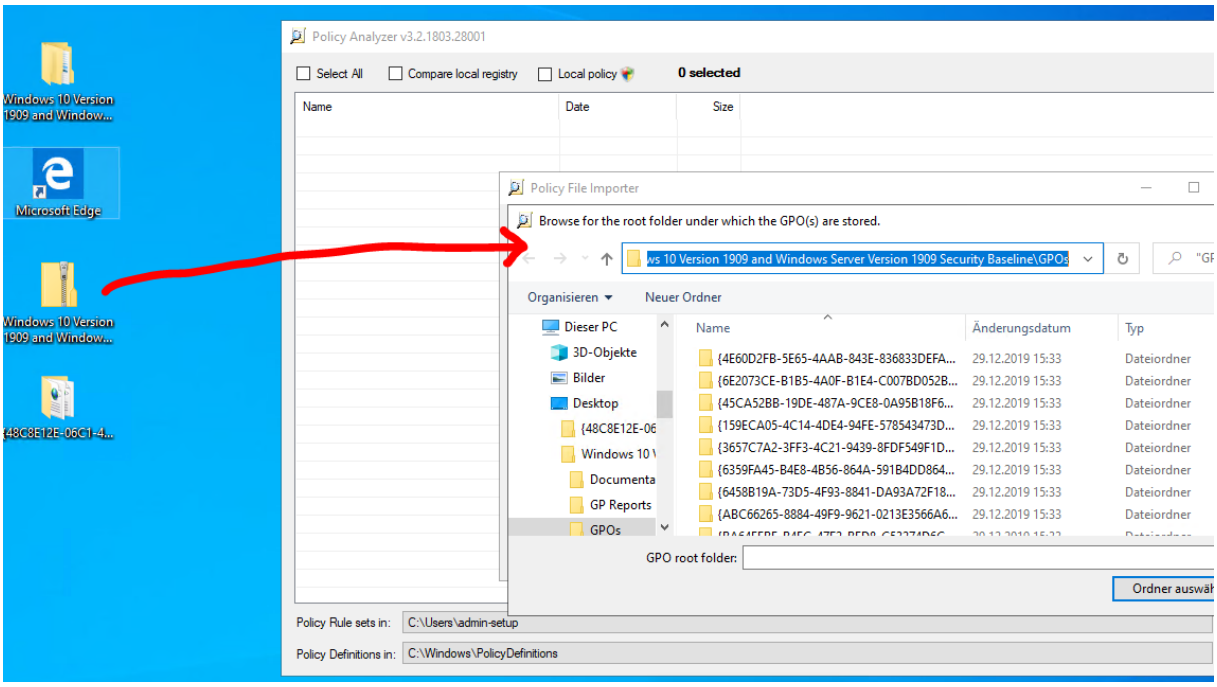
Jetzt starte ich den PolicyAnalyzer:



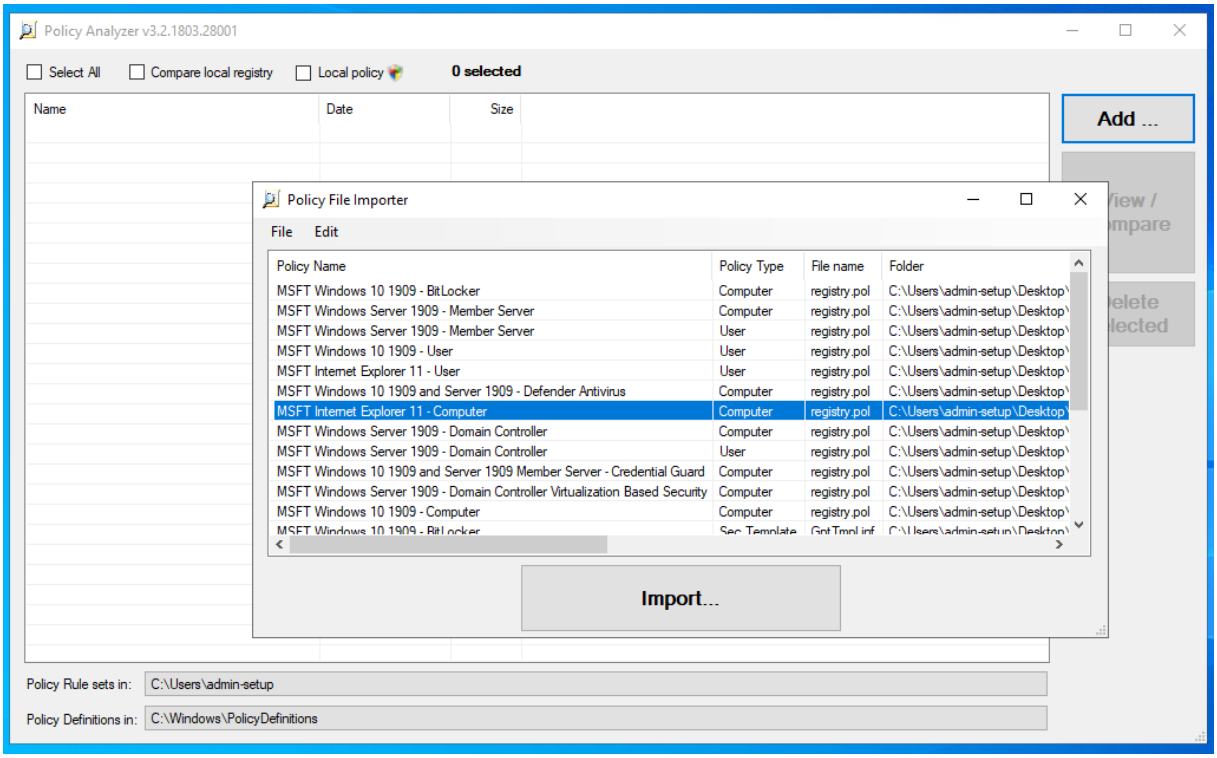
Im Menü kann ich nun die GPOs importieren:



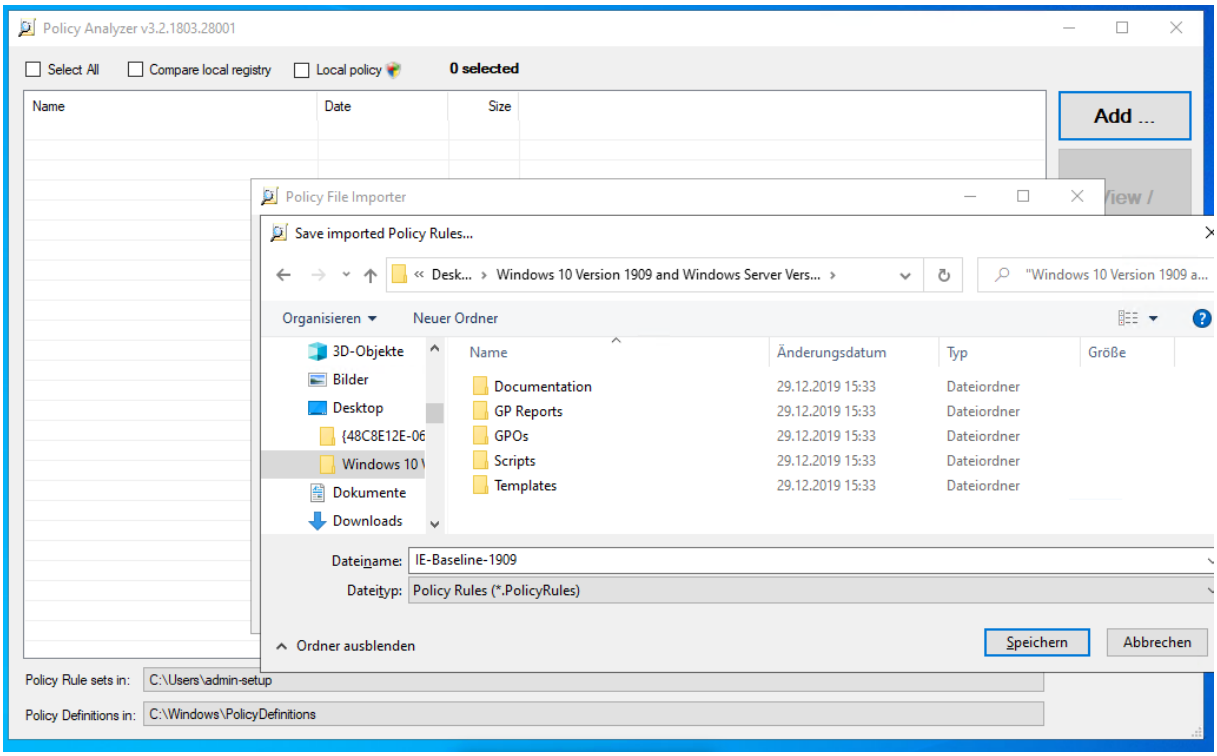
Ich beginne mit der aktuellen Microsoft-Baseline. Im Hauptverzeichnis aus dem ZIP-Archiv finde ich die Ordner mit den Unique-ID-Bezeichnern. Es genügt, wenn der Ordner geöffnet wird:



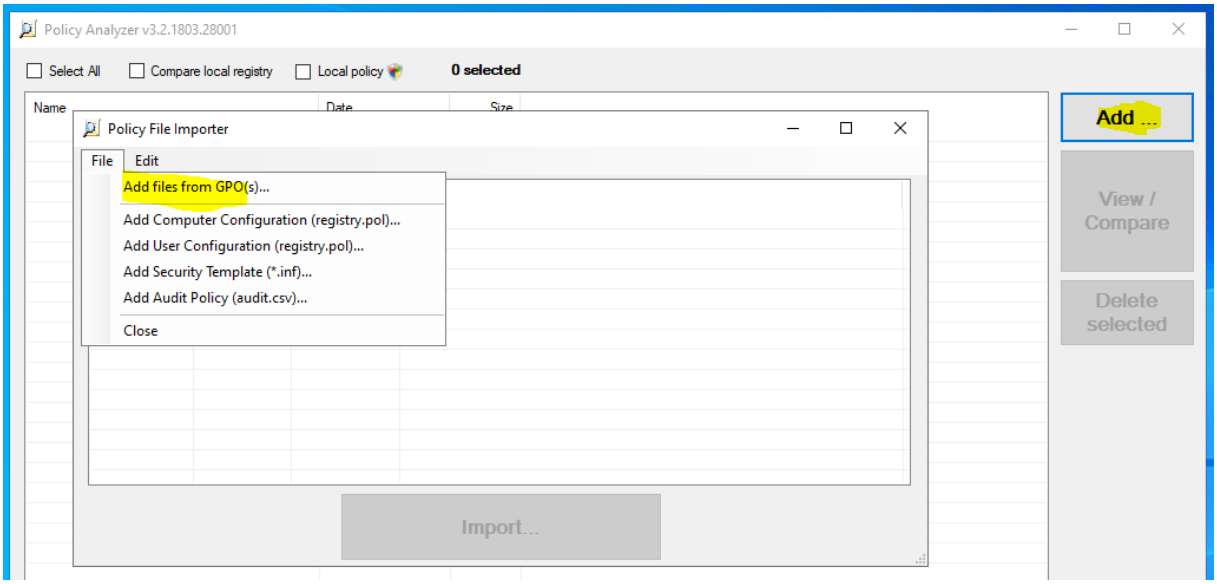
Der PolicyAnalyzer erkennt in den Unterverzeichnissen automatisch die GPOs. Hier finde ich die neue IE-Policy:

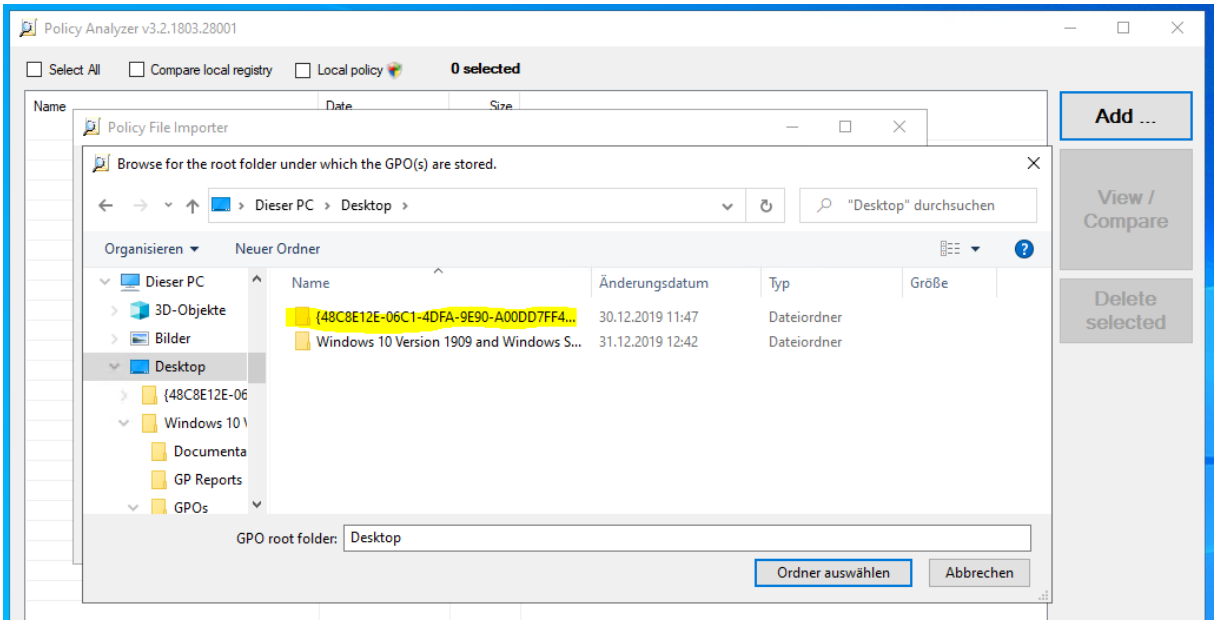


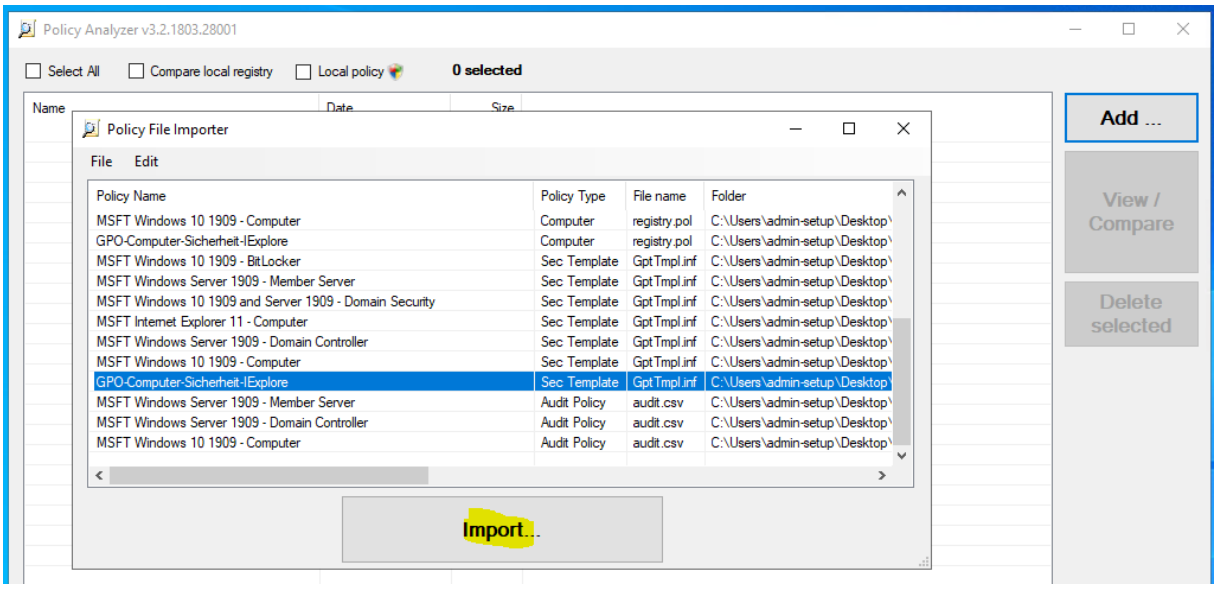
Der PolicyAnalyzer konvertiert die GPO in eine PolicyRule-Datei. Diese muss wieder gespeichert werden:



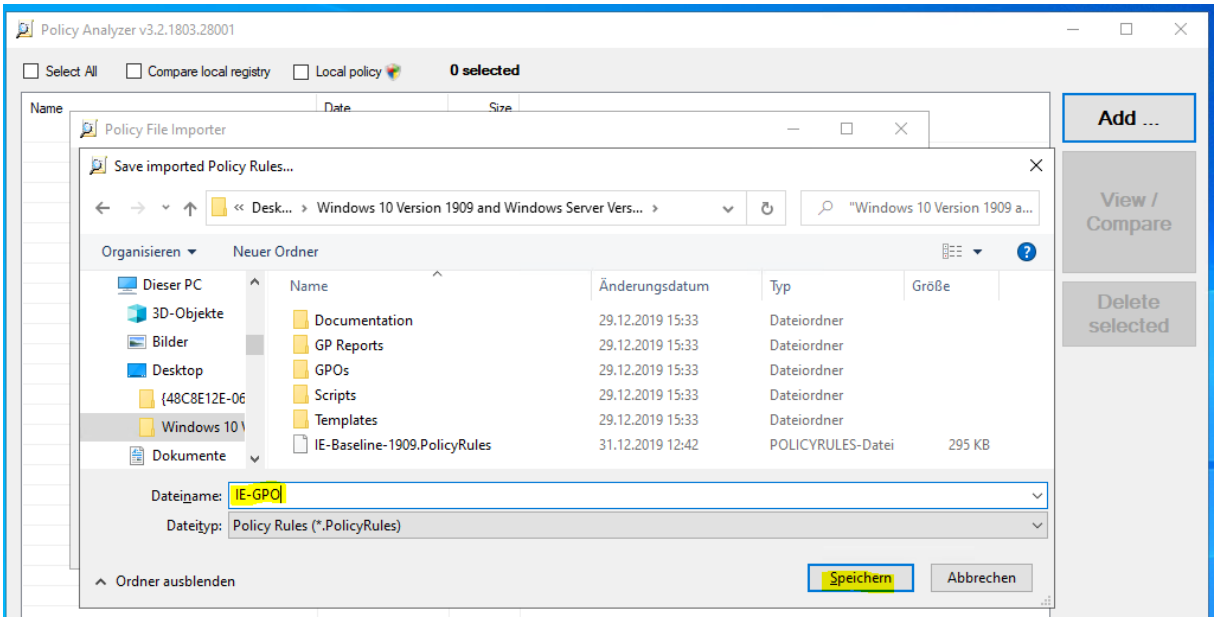
Jetzt kommt meine aktive GPO dran. Der Import-Assistent funktioniert genauso:



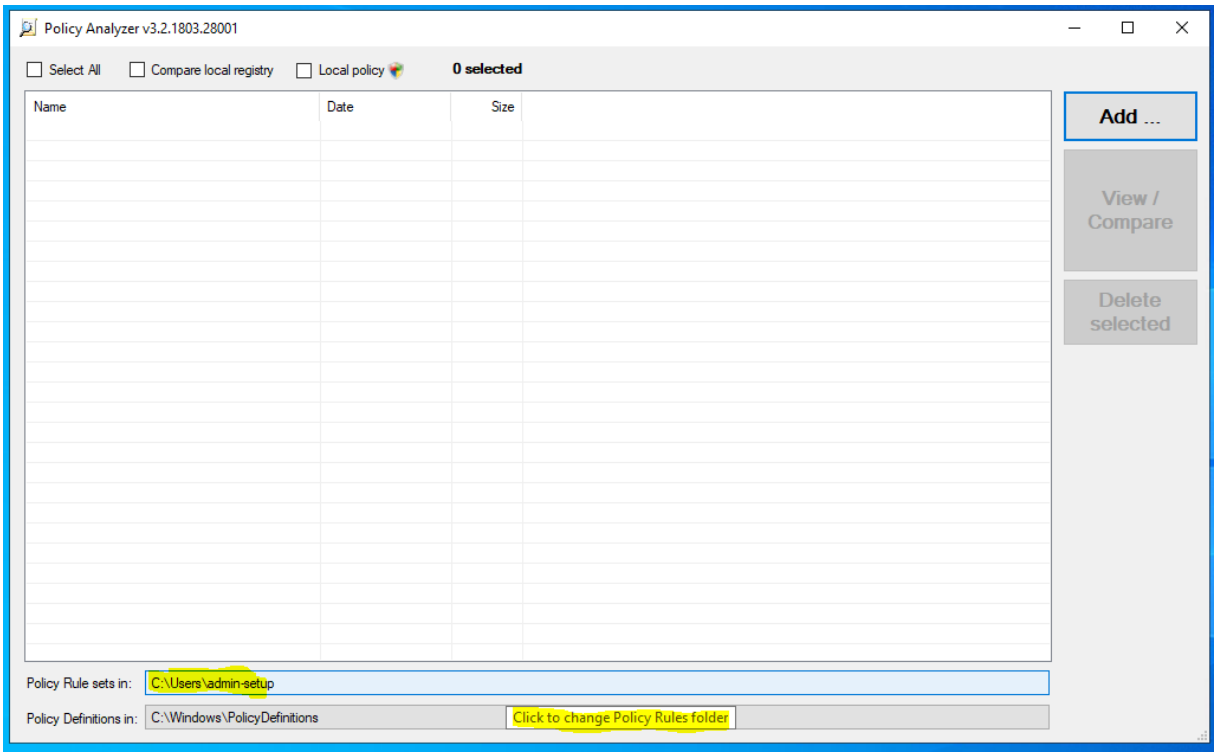


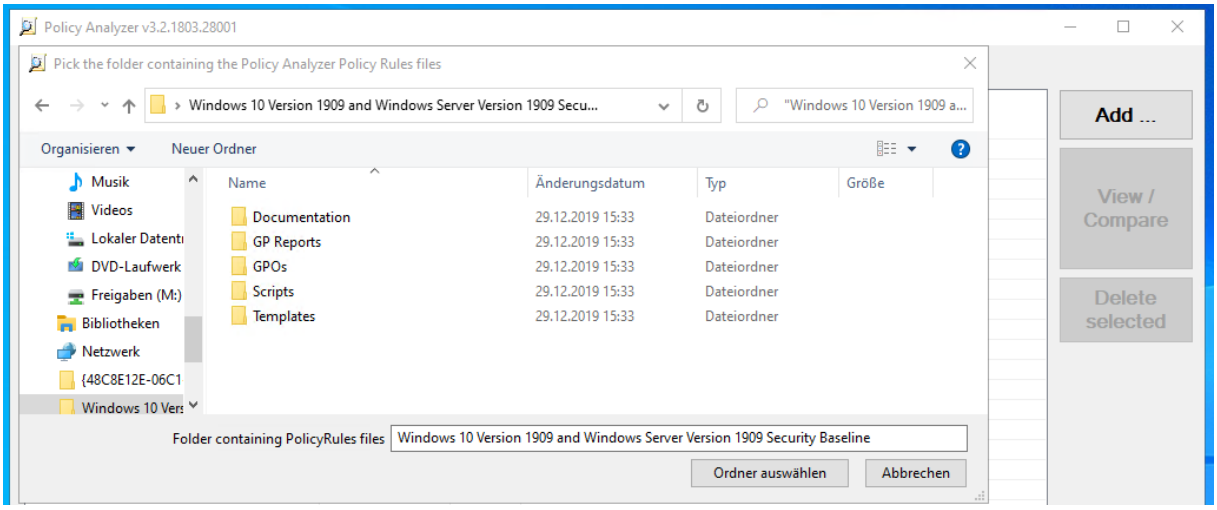


Und der Prozess wird mit der Erstellung der zweiten PolicyRule-Datei beendet:

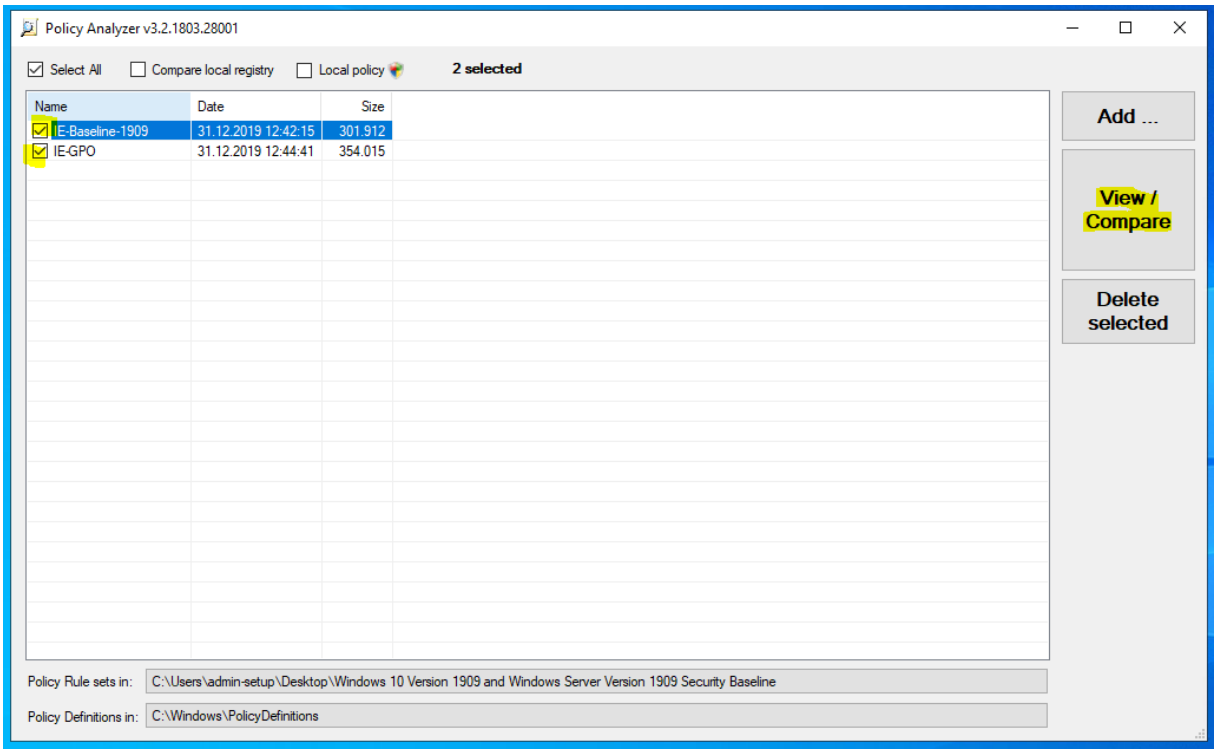


Jetzt stelle ich den Suchpfad um. Dazu muss auf den Schalter im unteren Bereich geklickt werden:

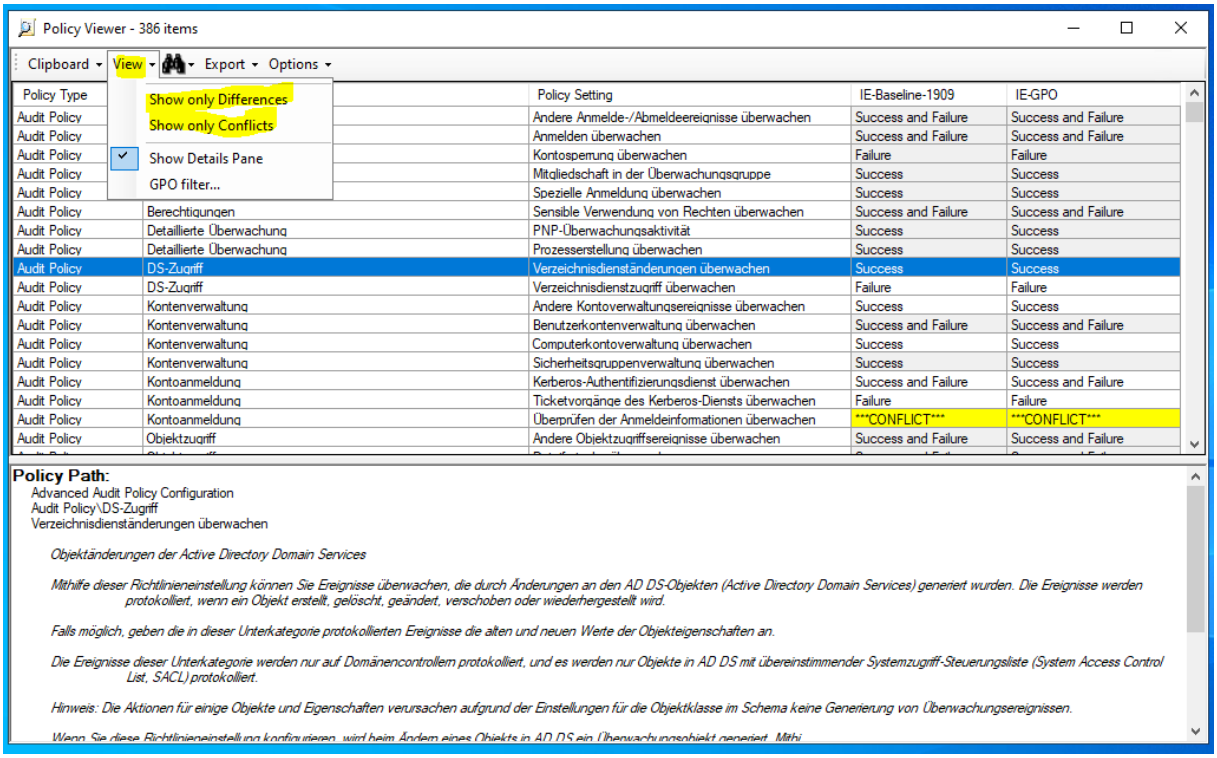




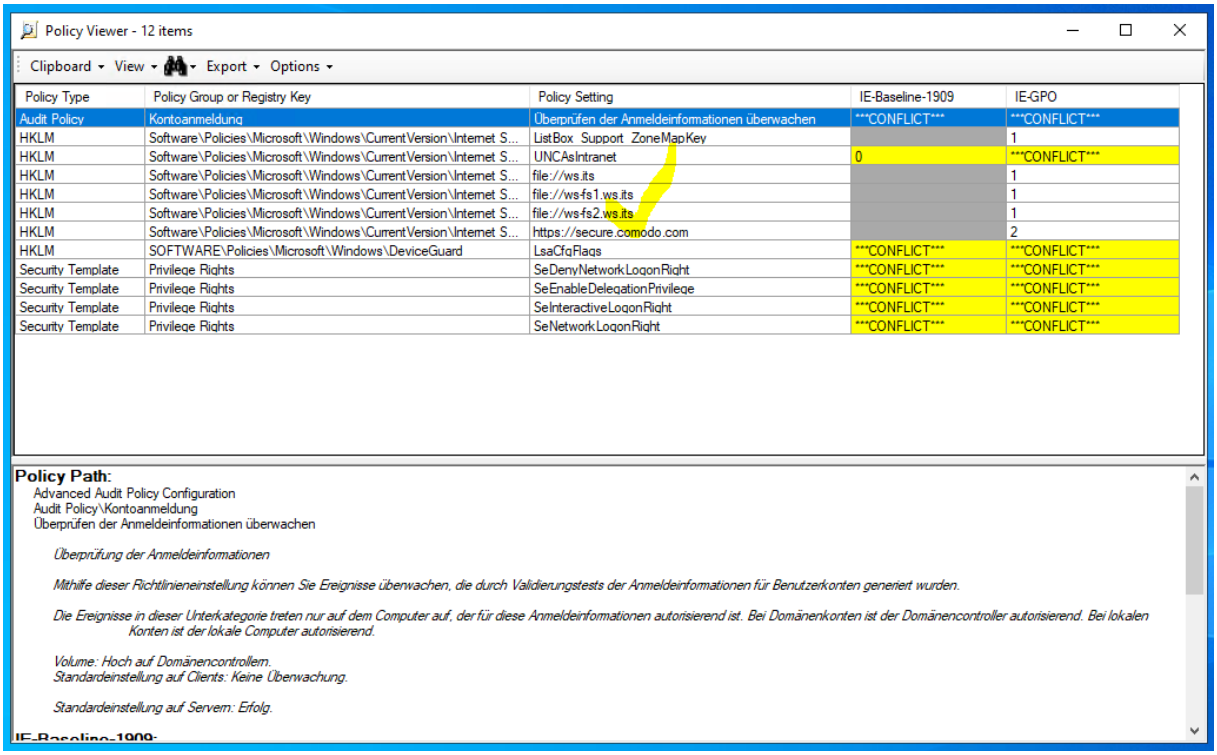
Jetzt werden die beiden PolicyRule-Dateien angezeigt. Der Rest ist einfach: Für den direkten Vergleich wähle ich beide aus:

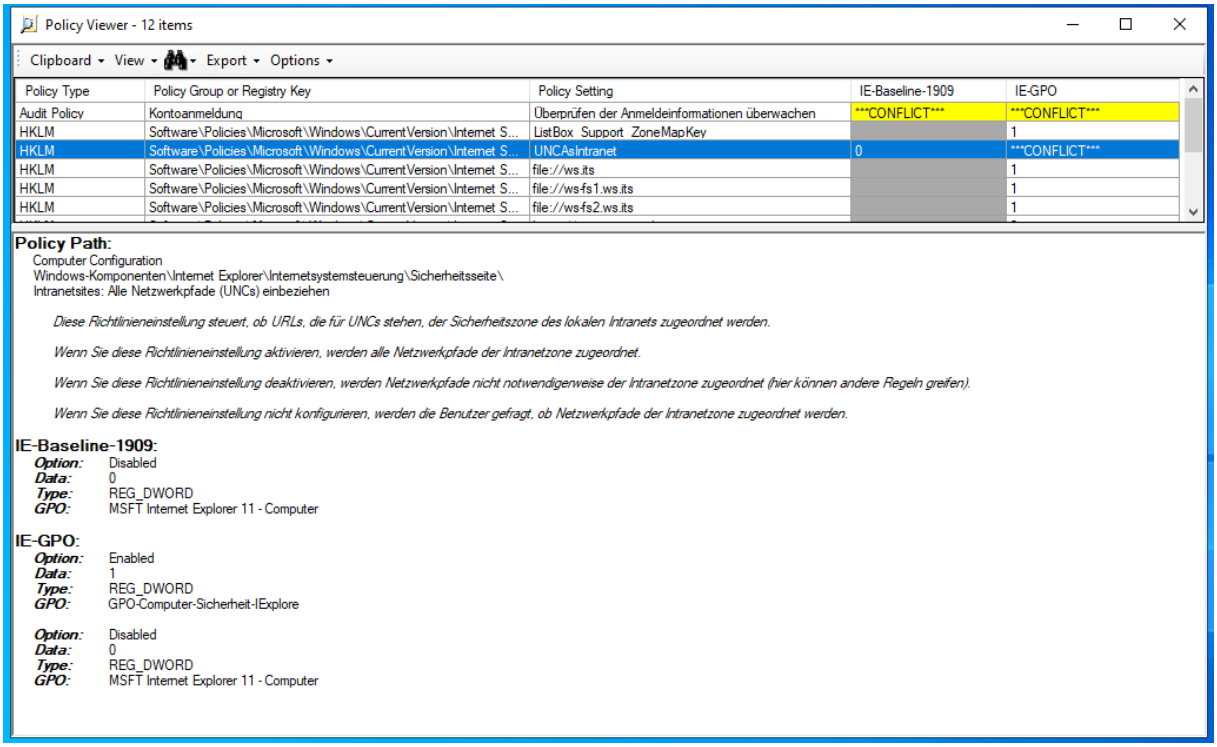


Im Vergleichsfenster kann ich nun die Anzeige filtern und identische Einstellungen ausblenden:



Und aus unzähligen Einstellungen in beiden GPO sehe ich die wenigen Unterschiede. Diese kann ich nun einzeln prüfen und danach entscheiden, ob die GPO mit den neuen Einstellungen freigegeben werden kann:

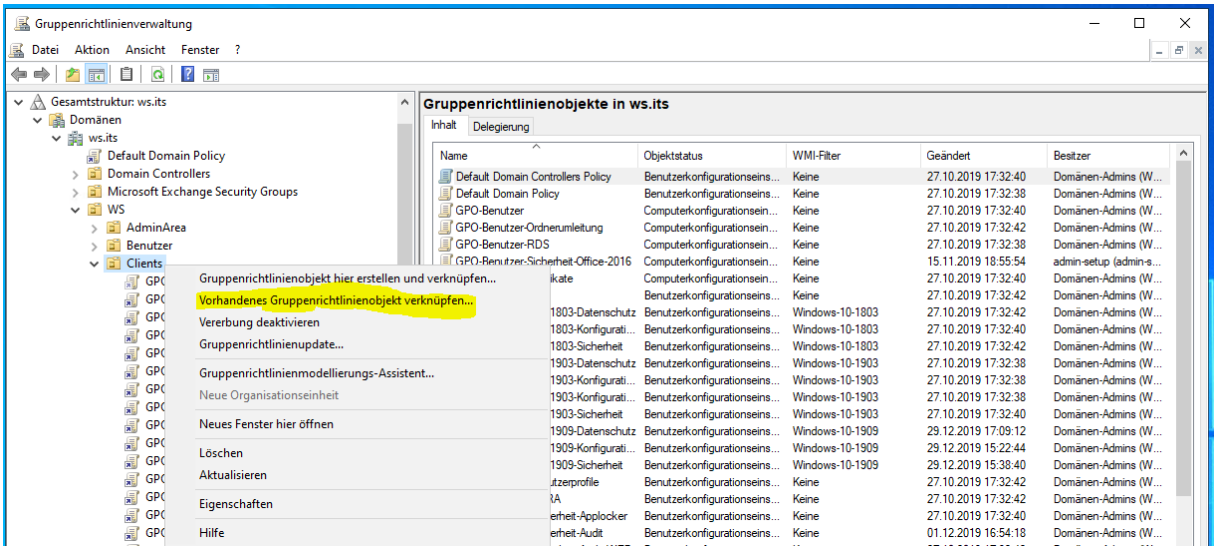


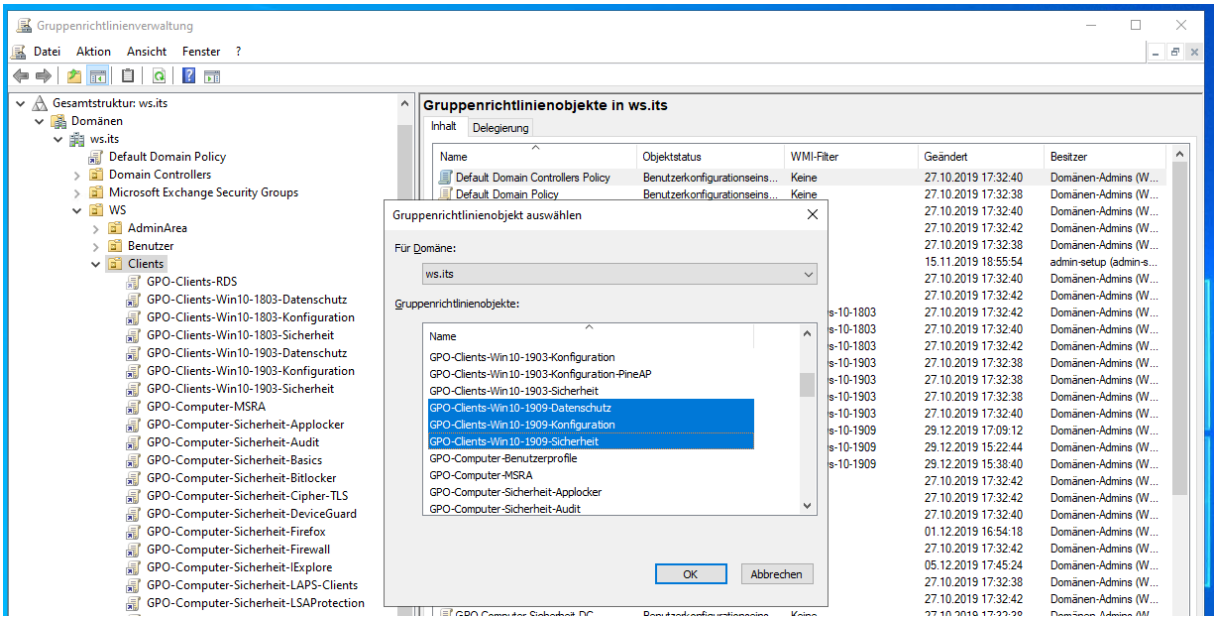


Das geht doch wesentlich schneller als ein vollständiger Testlauf, oder?

GPO anwenden

Meine neuen GPO sind fertig. Jetzt verknüpfe ich sie auf die gewünschten Organisationseinheiten. Natürlich sollten nur Testsysteme damit konfiguriert werden. Ggf. gibt es ja noch Anpassungsbedarf:

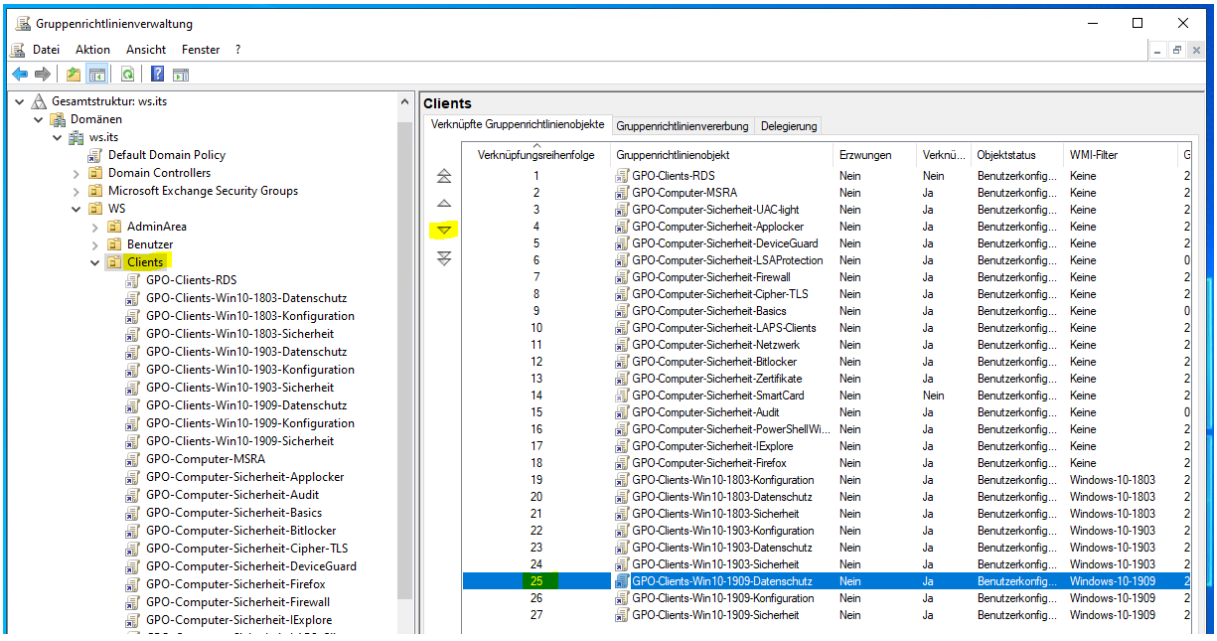




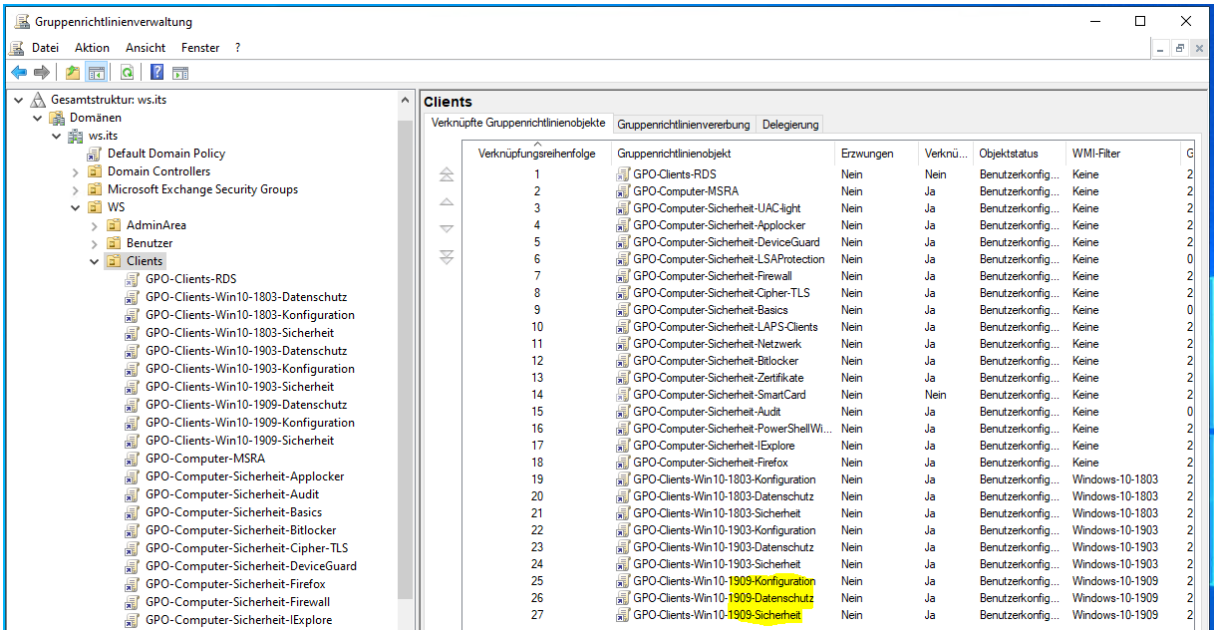
Die Reihenfolge in der GPO-Verarbeitung spielt in meinem Schema eine wichtige Rolle. Die Richtlinien werden in der Reihenfolge von „unten“ nach „oben“ verarbeitet. Dabei überschreibt eine GPO mit einer kleineren Rangfolge die Einstellungen der GPO mit größerer Rangfolge. Die Reihenfolge meiner GPO muss zwingend so aussehen:

- Zuerst wird die unmodifizierte Sicherheits-GPO angwendet.
- Deren Einstellungen werden von der Datenschutz-GPO ergänzt und überlagert (Sicherheit vs. Datenschutz…).
- Da auch die Datenschutz-GPO von externen Quellen stammen kann, würde ich auch deren Einstellungen mit der Konfigurations-GPO korrigieren. Daher kommt diese GPO an der dritten Stelle. Dadurch wird auch die Sicherheits-GPO überlagert.

Durch das freie Verbinden der GPOs auf die Organisationseinheit passt die Reihenfolge nicht:



Mit wenigen Klicks kann die Reihenfolge aber leicht angepasst werden. So ist es fein:

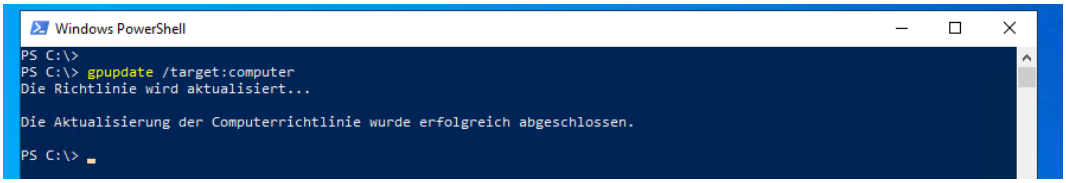


Testlauf

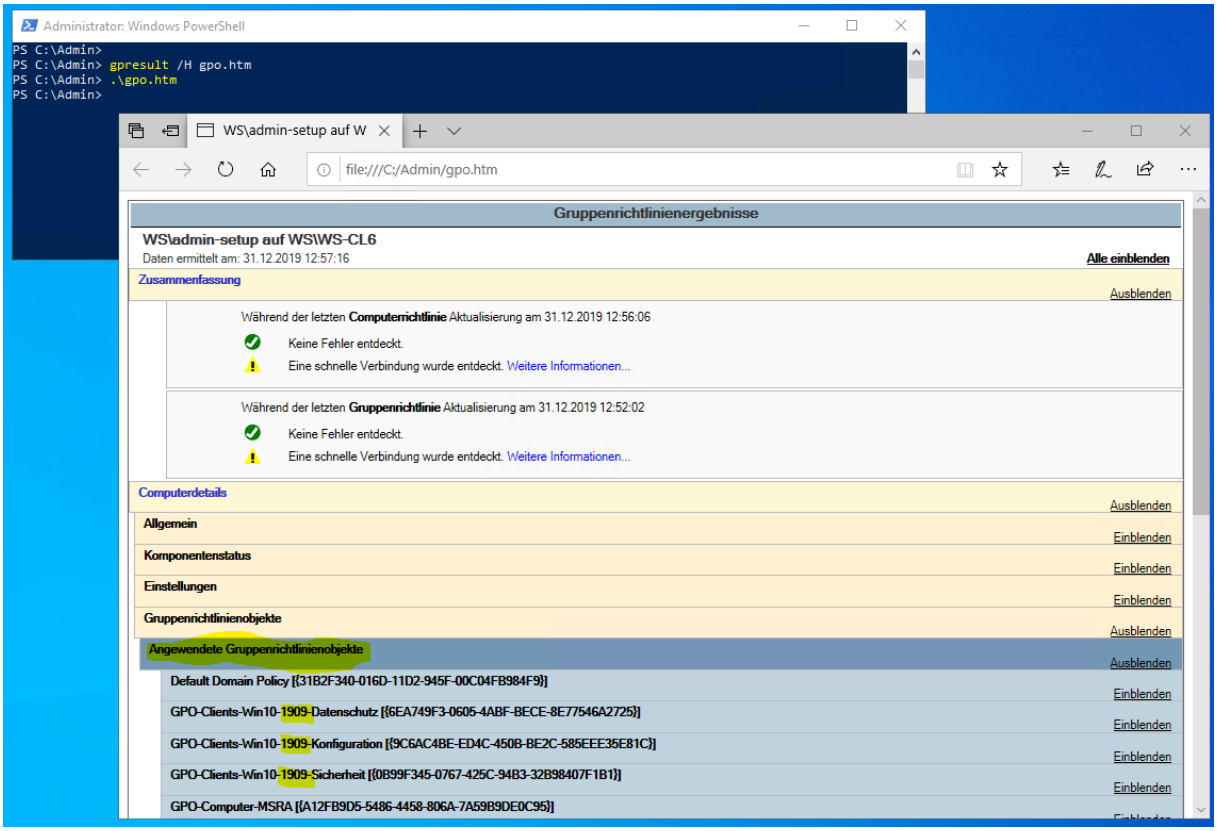
Normalerweise würde ich noch einen Sicherheitsfilter auf eine Testgruppe von Computern einrichten, bevor ich die GPOs verknüpfe. In meinem Fall gibt es aber nur den Editor-PC und einen bereits produktiven Client mit Windows 10 Version 1909. Alle anderen Clients arbeiten noch mit 1903. Das ist Einschränkung genug.

Dennoch teste ich die Richtlinien mit meinem Editor-Rechner. In größeren Umgebungen würde ich dafür separate Computer verwenden. Falls ich mich durch die GPOs aussperre, komme ich sonst vielleicht nicht mehr an die Konfigurationsoberfläche heran. Und nicht jede GPO-Einstellung wird wirkungslos, wenn man den Link der GPO entfernt.

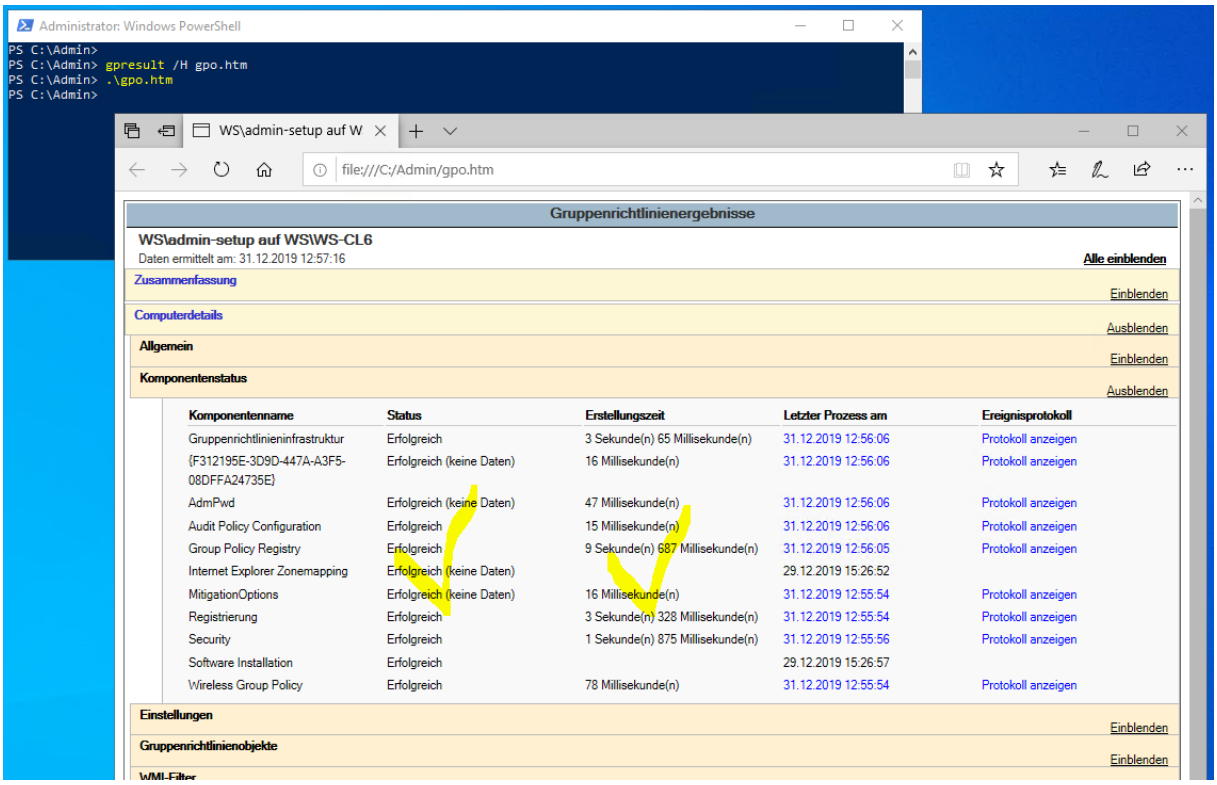
Ich aktualisiere die Richtlinien des Computers:



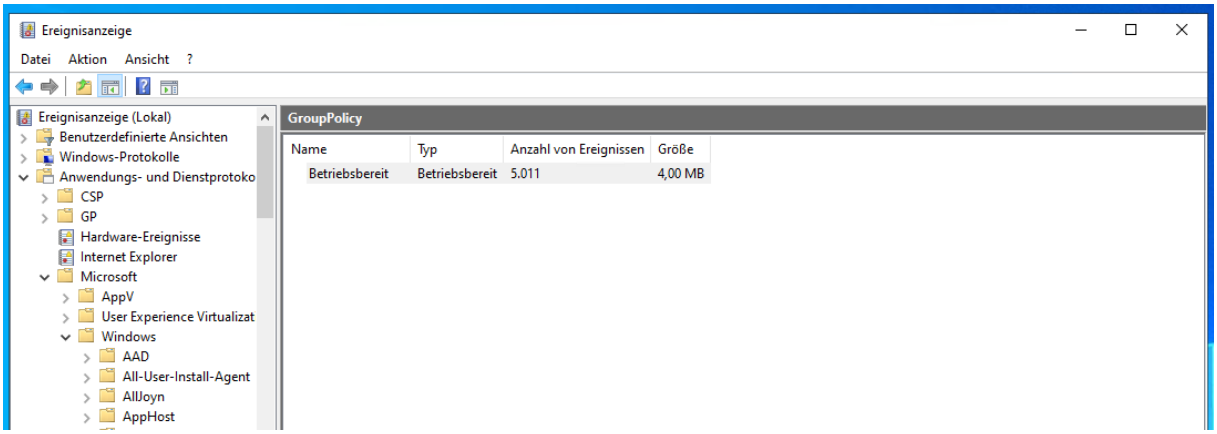
Anschließend erstelle ich einen Gruppenrichtlinien-Bericht. Die neuen GPO werden als angewendet gelistet:



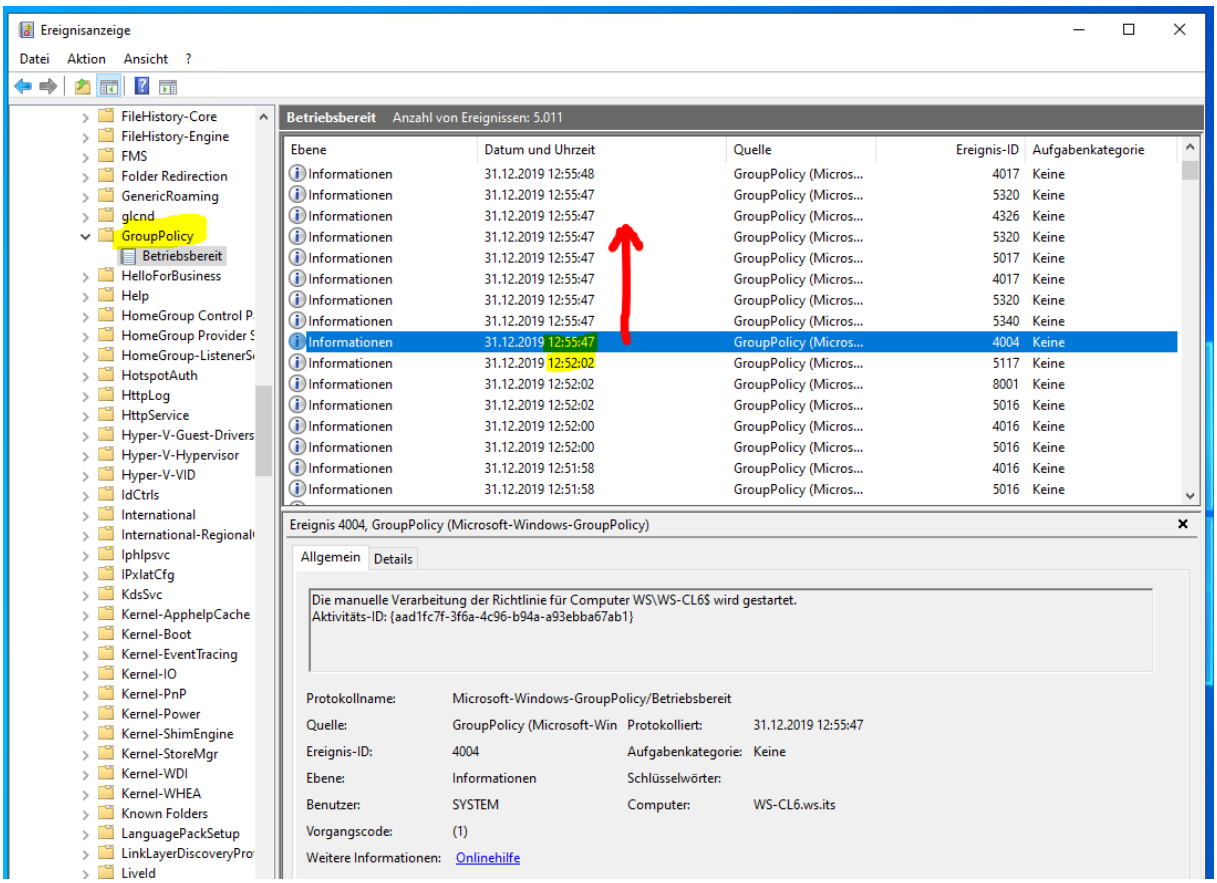
Jede GPO transportiert Einstellungen, die von lokalen Hilfsprogrammen – den Client Side Extensions (im deutschen Client „Komponente“ genannt) verstanden und verarbeitet werden. Auch diese sind alle erfolgreich durchgelaufen: Die erste Verarbeitung hat einige Sekunden in Anspruch genommen. Insgesamt wurden 12 Sekunden benötigt. Das ist noch in Ordnung:



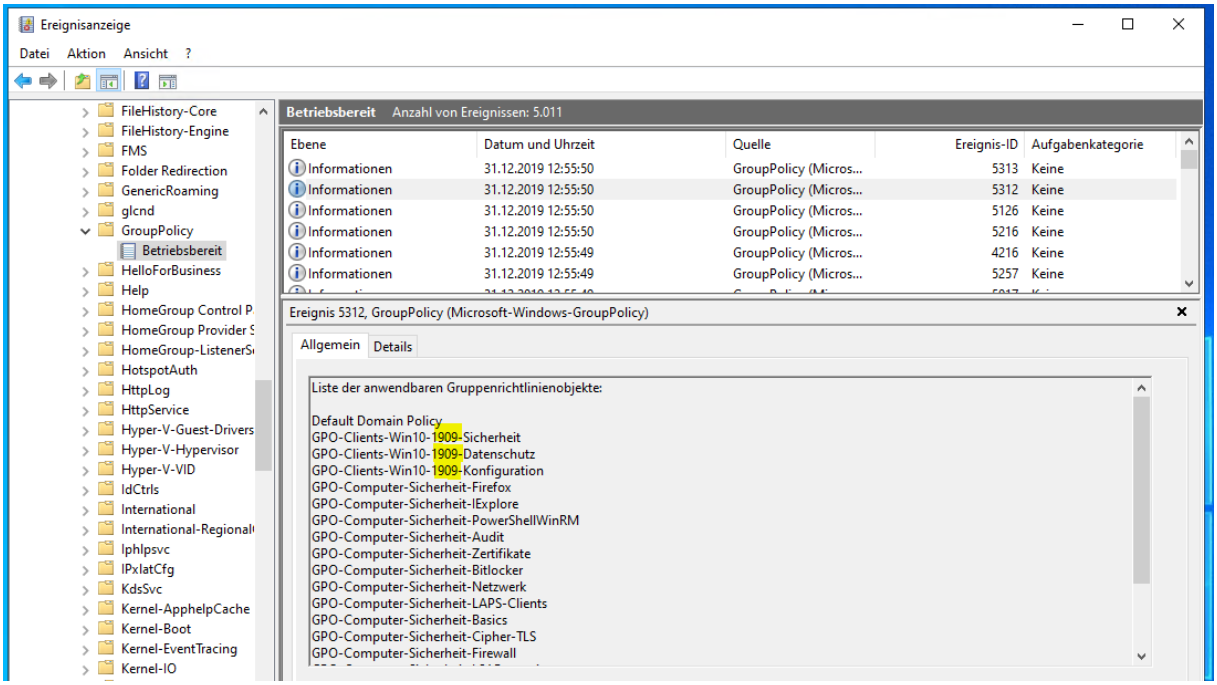
Weitere Details finde ich in der Ereignisanzeige. Die Gruppenrichtlinien haben ein eigenes Eventlog:



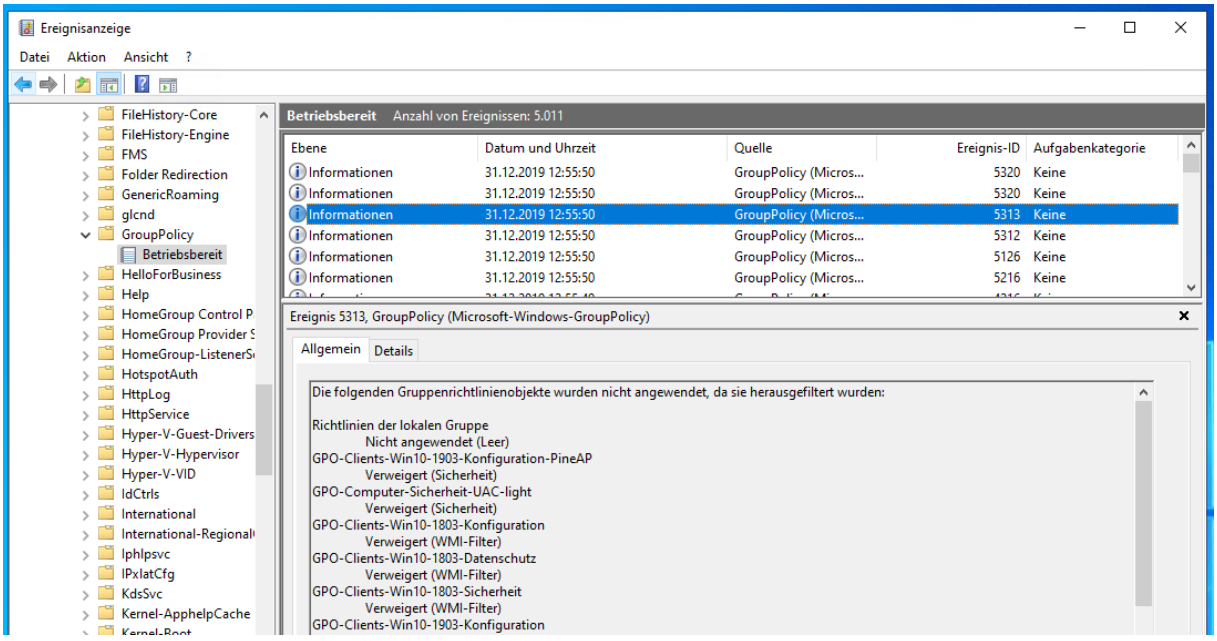
Die Einträge liegen zeitlich dicht beieinander. Dennoch kann ich den Beginn der Gruppenrichtlinienverarbeitung leicht finden. Ab hier prüfe ich die Events von unten nach oben:



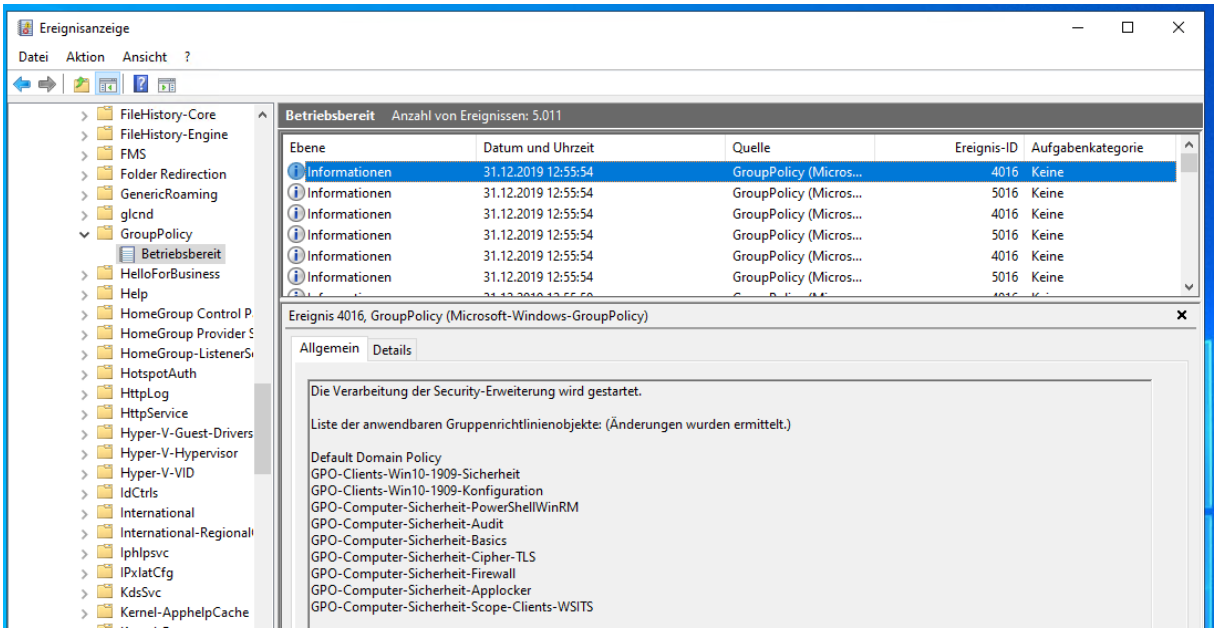
Auch hier wird bestätigt, welche Gruppenrichtlinien gefunden wurden:



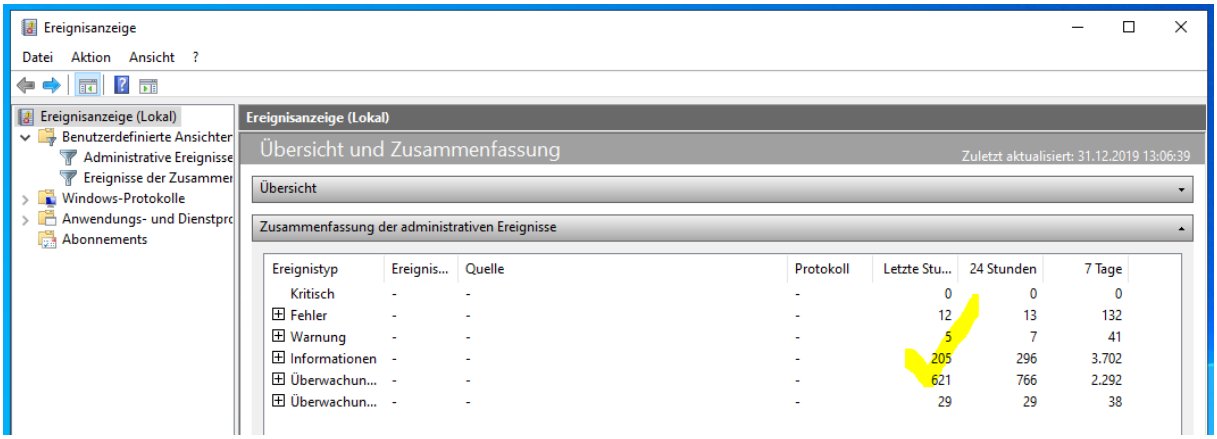
Ebenso zeigt das Eventlog, welche Richtlinien ausgelassen wurden. Natürlich mit Begründung:



Und hier wird aufgezeigt, welche Richtlinien verändert wurden:



Das scheint alles zu passen. Im nächsten Schritt starte ich den Client neu und prüfe anschließend die Eventlogs in der Zusammenfassung auf Warnungen und Fehler. Auch in den Details sind keine Probleme zu finden, deren Ursprung in den neuen Richtlinien liegt:



Weitere funktionale Tests sind natürlich auch notwendig. Ebenso wie die Überprüfung von Anwendungen. Dies kann üblicherweise an eine Pilotbenutzergruppe delegiert werden. Diese Personen erhalten das neue Betriebssystem und melden Probleme und Anpassungswünsche auf kurzen Wegen direkt zum GPO-Designer. In meinem Fall sind das meine lieben Kolleginnen in meinem Außenstandort.

Nachtrag: Auch nach mehreren Wochen gab es keine Probleme. Aus meiner Perspektive ist das neue Betriebssystem mit den Richtlinien wie gewohnt administrierbar. Einer Umstellung der anderen Clients auf Windows 10 Version 1909 steht nichts mehr im Wege.