

Vorwort

»Nimm mein Geld!«
»Ich kaufe es sofort!«
Ich war verwirrt.

Ein kleiner Saal mit ein paar hundert Leuten, denen ich Monad (was später PowerShell wurde) präsentierte. Ein Typ im hinteren Teil rief: »Nimm mein Geld!« So habe ich Thomas Lee kennengelernt. Nach dem Vortrag kam Thomas auf mich zu und gab mir eine 20-Dollar-Note mit seinem Namen darauf und sagte, dass Monad genau das war, was Windows-Administratoren brauchten, dass es die Welt revolutionieren würde und dass er der Erste sein wollte, der es kauft. Als ich erklärte, dass es ein kostenloses Feature von Windows werden würde, sagte Thomas, dass ihm das egal sei, er wolle trotzdem der Erste sein, der es kauft.

Ich habe zur Sicherheit noch mal nachgesehen und ja, der 20-Dollar-Schein befindet sich immer noch in meinem Portemonnaie. Ich habe ihn in den vergangenen 17 Jahren bei mir behalten, um nicht zu vergessen, was Thomas an diesem Tag so deutlich gesehen hat – PowerShell ist ein äußerst wichtiges Tool für jeden professionellen Windows-Administrator. Ich benutze das Adjektiv »professionell«, weil es wirklich zwei verschiedene Arten von Windows-Administratoren gibt: professionelle und nicht-professionelle. Einer der Gründe, warum Windows so erfolgreich war, ist, dass nicht-professionelle Administratoren es erfolgreich einrichten und für einfache Szenarien einsetzen konnten. Wenn Sie auf die Schaltfläche *Weiter* klicken können, ist die Wahrscheinlichkeit groß, dass Sie Windows erfolgreich einrichten können. Was Thomas richtig sah, war, dass die Welt der »Klicken Sie auf Weiter«-Administration begrenzt war und dass es einen wachsenden Bedarf an professionellen Administratoren gab, die die spezifischen Bedürfnisse ihres Szenarios verstehen und Windows so nutzen konnten, um diese Anforderungen zu erfüllen. Diese Aufgabe brauchte ein neues Werkzeug. Diese Aufgabe brauchte PowerShell.

Wenn Sie ein professioneller Administrator (oder nur ein leidenschaftlicher Power-User) sind, müssen Sie sich mit PowerShell gut auskennen und wissen, wie die spezifische Version des Betriebssystems, mit dem Sie arbeiten, mit PowerShell verwaltet werden kann. Jede Version von Windows hat neue Funktionen hinzugefügt, die von PowerShell verwaltet werden können. In diesem Buch zeigt Ihnen Thomas, wie Sie Ihre Investition in Windows Server 2019 nutzen können, indem es erläutert, wie Sie es mit PowerShell Windows administrieren und automatisieren können. Automatisierung bedeutet, herauszufinden, wie Sie einen Server verwalten und dann dieses Skript verwenden, um

Vorwort

zehn, hunderte oder hunderttausend andere Server zu administrieren. Es bedeutet Konsistenz, Wiederholbarkeit und Produktivität. Es geht um professionelle Administration.

Im Laufe der Jahre haben mir unzählige Administratoren für PowerShell gedankt und mir ihre Erfolgsgeschichten erzählt. Sie wurden in die Lage versetzt, komplizierte Situationen zu retten – Situationen in Unternehmen, die durch »Klicken-Sie-auf-Weiter«-Mitarbeiter nicht zu lösen waren. Sie wurden mit Boni, Gehaltserhöhungen, Beförderungen und neuen Jobs belohnt. Ich habe viele Leute getroffen, die berichteten, dass sich ihre Gehälter verdreifacht hätten, nachdem sie PowerShell gelernt hätten.

PowerShell stattet SIE mit Power aus.

Thomas war der erste, der dies mit absoluter Klarheit verstand. Er war von Anfang an mit im Team und gab uns aktiv Feedback zu dem, was professionelle Administratoren von PowerShell benötigten. Die Ansichten von Thomas über die Windows-Administration finden sich in der PowerShell – so wie sie sich heute darstellt – wieder. Daher gibt es meiner Meinung nach nur wenig andere Leute, die besser geeignet wären, um Ihnen zu helfen und zu verstehen, wie Sie dieses wunderbare Tool verwenden können. Mit PowerShell können Sie Ihre Windows Server 2019 verwalten, Ihrem Unternehmen einen großen Nutzen bieten, und Sie werden hierfür sicherlich belohnt.

Jeffrey Snover

Microsoft Technical Fellow

Einleitung

PowerShell wurde erstmals im Jahre 2003 auf der Professional Developers Conference in Los Angeles von Jeffrey Snover der Welt vorgestellt. PowerShell – mit dem Codenamen Monad – stellte damals eine komplette Revolution im Management dar. Das in dieser Zeit geschriebene Whitepaper *The Monad Manifesto* (siehe <http://www.jsnover.com/blog/2011/10/01/monad-manifesto>) enthielt eine wunderbare Analyse des damaligen Problems – nämlich die Verwaltung einer großen Anzahl von Windows-Systemen. Ein wichtiger Punkt ist, dass die GUI nicht skaliert wird, während PowerShell dies tut.

PowerShell hat die Verwaltung komplexer, netzwerkbasierter Windows-Infrastrukturen und zunehmend auch von Nicht-Windows-Infrastrukturen verändert. Kenntnisse über PowerShell zu haben und zu wissen, wie man das meiste aus PowerShell herausholen kann, sind heutzutage für jeden professionellen IT Pro obligatorisch.

Dieses Buch führt Sie in einer Vielzahl von Szenarien durch den Einsatz von PowerShell, wobei viele der zahlreichen Features in Windows Server 2019 behandelt werden. Diese Einleitung bietet eine Einführung in die Inhalte des Buches; außerdem erhalten Sie ein paar Tipps dazu, wie Sie das Beste aus dem Buch herausholen können.

In diesem Buch

Kapitel 1, »PowerShell-Umgebung für die Systemverwaltung einrichten«, befasst sich mit der Konfiguration Ihrer Workstation und damit, wie Sie Ihre Infrastruktur für die Nutzung von PowerShell vorbereiten und wie Sie die Remoteserver-Verwaltungstools installieren.

Kapitel 2, »Windows-Netzwerke verwalten«, zeigt Ihnen, wie Sie Windows-Netzwerke mit PowerShell verwalten. Netzwerke sind heute für fast jede Organisation von zentraler Bedeutung, und in diesem Kapitel werden eine Vielzahl netzwerkbezogener Aufgaben behandelt. Sie lernen neue Verfahren kennen, um Altbekanntes mit PowerShell zu erledigen. Außerdem erfahren Sie, wie Sie DNS, DHCP und Active Directory einrichten.

Kapitel 3, »Windows Active Directory verwalten«, erörtert, wie Active Directory installiert, verwaltet und genutzt werden kann. Hierbei lernen Sie auch die Verwendung von AD-Objekten und von Gruppenrichtlinien kennen.

Kapitel 4, »Windows-Speichertechnologien verwalten«, befasst sich mit dem Thema Datenspeicher in Windows Server 2019.

Kapitel 5, »Freigegebene Daten verwalten«, untersucht verschiedene Möglichkeiten, wie Sie Daten mit Windows Server und PowerShell freigeben können, einschließlich SMB-Freigaben, iSCSI und DFS (Distributed File System).

Kapitel 6, »Windows Update verwalten«, hilft Ihnen, die Verwaltung von Updates über Windows Update zu implementieren, weil es wichtig ist, dass all Ihre Windows-Clients und Server immer mit den aktuellsten Patches versehen sind.

Kapitel 7, »Windows-Druckumgebung verwalten«, zeigt Ihnen, wie Sie Drucker, Druckwarteschlangen und Druckertreiber verwalten, und wie Sie Drucker über Gruppenrichtlinien bereitstellen.

Kapitel 8, »Einführung in Container«, stellt die Docker-Container in Windows Server 2019 vor. Sie werden fertige Container-Images herunterladen und verwenden sowie eigene erstellen.

Kapitel 9, »Internetinformationsdienste verwalten«, zeigt, wie Sie eine Vielzahl von IIS-bezogenen Aufgaben durchführen, einschließlich der Installation und Konfiguration von IIS, der Einrichtung von SSL, der Verwaltung der Verschlüsselungssuites sowie der Konfiguration des Netzwerklastenausgleichs.

Kapitel 10, »Desired State Configuration verwalten«, zeigt, wie Sie diese wichtige Funktion verwenden, um sicherzustellen, dass ein Server ordnungsgemäß eingerichtet ist und dass er auch in Zukunft entsprechend konfiguriert bleibt. Hierbei werden auch die Einrichtung eines Pullservers und das Konfigurieren von Teilkonfigurationen behandelt.

Kapitel 11, »Hyper-V verwalten«, veranschaulicht die Verwendung von Hyper-V. In diesem Kapitel erfahren Sie, wie Sie VMs mit Hyper-V erstellen und bereitstellen. Dazu gehört auch die verschachtelte Virtualisierung, bei der eine Hyper-V-VM innerhalb einer anderen Hyper-V-VM ausgeführt wird, was für eine Reihe von Einsatzszenarien nützlich ist.

Kapitel 12, »Azure verwalten«, befasst sich mit der Verwaltung von IaaS- und PaaS-Ressourcen in Azure mithilfe von PowerShell. Zum Testen der Rezepte in diesem Kapitel benötigen Sie Zugriff auf Azure. In diesem Kapitel wird beschrieben, wie Sie einen virtuellen Computer, eine Azure-Website und eine SMB3-Dateifreigabe einrichten.

Kapitel 13, »Leistung und Nutzung überwachen«, erfahren Sie, wie Sie die Leistung Ihrer Windows 2019-Server messen, überwachen und die Ergebnisse in einem Bericht darstellen. Es gibt mehrere Rezepte, die zeigen, wie Sie bestimmte Leistungsmesswerte erhalten und wie Sie aus diesen Messwerten Diagramme erstellen, die Sie für weitergehende Analysen nutzen können.

Kapitel 14, »Probleme mit Windows Server lösen«, befasst sich mit einer Reihe von Aspekten der reaktiven und proaktiven Fehlerbehebung. Dazu gehören das Abrufen von Ereignissen aus dem Ereignisprotokoll und die Verwendung des Best Practice Analyzer in Windows Server 2019.

Was brauchen Sie für dieses Buch?

Um wirklich alles aus diesem Buch herauszuholen, müssen Sie mit dem Code aus den Rezepten experimentieren. Um Fehler zu vermeiden, die sich auf Live-Server in der Produktivumgebung auswirken, sollten Sie stattdessen entweder Testhardware oder virtuelle Computer verwenden, um ein Testlabor zu erstellen, in dem Fehler keinen ernsthaften Schaden verursachen.

Dieses Buch verwendet eine Vielzahl von Servern sowie einen Windows 10-Client; alle Computer sind Mitglieder der Domäne Reskit.Org. Die Computer in dieser Domäne verwenden den IP-Adressblock, 10.10.10/24, der im Abschnitt »Das Buch optimal nutzen« beschrieben wird.

Sie sollten einen Windows 10- oder Windows Server 2019-Host mit aktivierter Hardware-Virtualisierung sowie eine Virtualisierungslösung verwenden. Wenn Sie Zugriff auf eine Cloud-Computing-Plattform haben, können Sie die meisten Rezepte auf in der Cloud gehosteten virtuellen Maschinen ausführen, obwohl dies nicht von uns getestet wurde. Sie können jede Virtualisierungslösung verwenden.

Das Buch wurde mit Hyper-V sowie verschachtelter Hyper-V-Virtualisierung unter Windows 10 und Windows Server 2019 entwickelt.

Für wen ist dieses Buch?

Dieses Buch richtet sich an IT-Experten, einschließlich Systemadministratoren, Systemingenieuren, Systemarchitekten und Berater, die Windows PowerShell nutzen müssen, um ihre täglichen Aufgaben zu vereinfachen und zu automatisieren.

Das Buch optimal nutzen

Dieses Buch wurde auf der Grundlage einiger Annahmen und im Hinblick auf verschiedene mögliche Einschränkungen geschrieben. Sie sollten diesen Abschnitt lesen, um zu verstehen, wie ich dieses Buch konzipiert habe, wie ich mir seine Verwendung vorstelle und welche Annahmen ich über Sie, den Leser, mache. Dies kann Ihnen dabei helfen, das meiste aus diesem Buch herauszuholen.

Die erste Annahme, die ich beim Schreiben dieses Buches gemacht habe, ist, dass Sie die Grundlagen von PowerShell kennen. Dieses Buch ist kein PowerShell-Tutorial. Die Rezepte verwenden eine breite Palette der PowerShell-Features, einschließlich WMI, Remoting, AD usw., jedoch müssen Sie die Grundlagen von PowerShell kennen. Das Buch verwendet PowerShell, seine Syntax und die Cmdlets auf der Basis von Windows Server 2019 und Windows 10.

Die zweite, damit zusammenhängende Annahme ist, dass Sie ein angemessenes Hintergrundwissen über die Windows-Infrastruktur und die verschiedenen Technologien besitzen, einschließlich Active Directory, Netzwerk und Datenspeicher. Die Rezepte geben einen Überblick über die verschiedenen Technologien und enthalten Links zu weiterführenden Informationen. Außerdem finden Sie zahlreiche Links zu Websites, die ausführliche Informationen zu den Themen in diesem Buch enthalten.

Die Rezepte stellen die Grundlagen dar – Sie können sie übernehmen, anpassen und erweitern. Ziel der Rezepte ist, die Grundlagen der Verwaltung bestimmter Aspekte von Windows Server zu veranschaulichen. In vielen Fällen schlägt ein Rezept vor, wie Sie es für Ihre Umgebung verbessern können. Ein Rezept soll Ihnen zeigen, wie ein bestimmtes Feature funktioniert, damit Sie es für Ihre Umgebung nutzen und erweitern können.

Führen Sie die Rezepte Schritt für Schritt in der PowerShell ISE oder mit Visual Studio Code (VS Code) aus. Eine Ausnahme bilden die Rezepte in Kapitel 8 (Einführung in Container). Das Haupttool, das Sie in diesem Kapitel verwenden, `docker.exe`, wird in einer GUI (ISE oder VS Code) anders ausgeführt; daher sollten Sie diese Rezepte in der PowerShell-Konsole ausführen.

Ich habe die Rezepte in diesem Buch Schritt für Schritt erstellt und getestet (d.h., nicht das gesamte Rezept als eine einzige Skriptdatei ausgeführt). Wenn Sie das gesamte Rezept in einem einzigen Schritt ausführen, entspricht ein Teil der Bildschirmausgabe möglicherweise nicht dem, was Sie im Buch sehen, was damit zusammenhängt, wie PowerShell Objekte formatiert.

Sobald eines der Rezepte gut funktioniert, versuchen Sie, den Code des Rezepts zu refaktorisieren und so Ihre eigenen, wiederverwendbaren Funktionen zu erstellen. In einigen Fällen erstellen wir einfa-

Einleitung

che Funktionen, die als Leitfaden für umfangreichere Skripte dienen können, die Sie erstellen. Sobald Sie über funktionierende und nützliche Funktionen verfügen, integrieren Sie diese in Ihre persönlichen Module oder die Ihrer Organisation und verwenden Sie den Code wieder.

Wie jeder Autor weiß, ist das Schreiben von PowerShell-Skripten, die in einem Buch veröffentlicht werden, sowohl ein Layout- und Produktionsalgebraum. Um die Probleme speziell mit der maximalen Zeichenanzahl pro Zeile und Zeilenumbrüchen zu reduzieren, habe ich vielfältige Methoden verwendet, die sicherstellen, dass die Codezeilen ohne zusätzliche Umbrüche in die Seitenbreite des Buches passen. Viele Rezepte verwenden daher Hashtabellen für die Cmdlet-Parameter, um sicherzustellen, dass jede Zeile jedes Rezepts die maximale Zeichenanzahl nicht überschreitet und dass es keine unbeabsichtigten Zeilenumbrüche gibt. Ich hoffe, es gibt nicht zu viele Probleme mit dem Layout!

Viele der in diesem Buch verwendeten Cmdlets, Befehle und Objektmethoden, erzeugen eine Ausgabe, die möglicherweise nicht so hilfreich oder nützlich ist, insbesondere in einer Produktivumgebung. Einige Cmdlets erzeugen eine Ausgabe, die viele Seiten dieses Buches füllen würde, aber nur wenig Mehrwert bieten. Aus diesem Grunde verwenden viele Rezepte das Cmdlet `Out-Null`, wodurch die Ausgabe an `NULL` gesendet und somit gelöscht wird. Fühlen Sie sich frei, dies zu entfernen, wenn Sie irgendwo mehr Details sehen möchten. Außerdem habe ich die Ausgabe in vielen Fällen angepasst, um überflüssigen Leerraum zu vermeiden. Wenn Sie also ein Rezept testen, sehen Sie möglicherweise eine Ausgabe, die etwas anders aufgebaut ist, aber sie sollte die gleichen Informationen enthalten. Denken Sie schließlich daran, dass die spezifische Ausgabe, die Sie sehen, je nach Umgebung und den spezifischen Werten, die Sie in jedem Schritt verwenden, unterschiedlich sein kann.

Beim Schreiben dieses Buches habe ich eine große VM-Farm verwendet, die aus über 20 Windows 2019-Servern und Windows 10-Clients besteht. Mein Hauptentwicklungs-Host war ein gut konfiguriertes Windows 10-System (96 GB RAM, 2 × 6-Kern-Xeon-Prozessoren und schnelle SSDs). Alle in diesem Buch verwendeten Hosts sind eine Kombination aus physischer Hardware (die fast ausschließlich unter Windows 10 und einem großen Satz von VMs ausgeführt wird), wie im Rezept beschrieben.

Als Unterstützung beim Schreiben dieses Buches habe ich eine Reihe von Skripten erstellt, die die Hyper-V-VMs erstellen, die ich für die Entwicklung der Rezepte verwendet habe. Diese Skripte finden Sie unter: <https://github.com/doctordns/ReskitBuildScripts>. Dort finden Sie auch einige Details zu dem Netzwerk, das die mit diesen Skripten erstellten VMs verwenden, einschließlich Hostnamen und IP-Adressen. Der vollständige Satz von VMs nahm, nachdem ich mit dem Schreiben fertig war, rund 600 GB Speicherplatz ein. Glücklicherweise sind Datenträger heute preiswert!

PowerShell bietet eine hervorragende Funktionsabdeckung – Sie können die meisten Funktionen und Features von Windows Server 2019 mit PowerShell verwalten, aber eben nicht alle. In einigen Fällen können Sie mithilfe von CIM-Cmdlets in die Windows Management Instrumentation (WMI) eintauchen, um zu Objekteigenschaften und Methoden zu gelangen, die nicht durch Cmdlets verfügbar sind. Durch das Aufkommen von CDXML-basierten Cmdlets (Commandlet Definition XML) hat die Anzahl der Netzwerk- und anderer Cmdlets, die auf WMI basieren, zugenommen. Aber selbst dann gibt es noch eine Reihe von Stellen, an denen Sie eine Windows-Konsolenanwendung verwenden oder eine nicht verwaltete DLL aufrufen müssen. Unterm Strich bedeutet dies, dass Sie manchmal ältere Tools verwenden müssen, um bestimmte Aspekte von Windows zu verwalten, z. B. die Ereignisweiterleitung oder die Leistungsprotokollierung. Ich versuche, so etwas zu vermeiden, aber in vielen Fällen zeigt das betreffende Rezept, wie die Konsolenanwendungen in PowerShell verwendet werden.

In vielen Fällen gibt es in Windows Server keine offiziellen von Microsoft erstellten Cmdlets, die Ihnen bei der Verwaltung bestimmter Aspekte Ihrer Infrastruktur helfen könnten. Der PowerShell-Katalog (PS Gallery) sowie die großartigen Add-Ins von Drittanbietern haben dazu geführt, dass Sie eine Fülle von Funktionen finden, herunterladen und installieren können.

Der gesamte Code in diesem Buch wurde getestet; es funktionierte und tat, was es sagt (zumindest während der Schreibphase). Der Produktionsprozess ist komplex, und es ist möglich, dass sich während der Produktionsphase Codefehler einschleichen. Bei einigen der komplexeren Schritte können während der Produktion Fehler auftreten. Wenn ein Schritt für Sie fehlschlägt, wenden Sie sich bitte an den Verlag. Sie können natürlich auch im Spiceworks PowerShell-Forum posten, wenn Sie schnell nach einer Lösung suchen.

Beim Schreiben der Rezepte verwende ich die vollständigen Namen der Cmdlets und keine Aliase, außerdem sind auch die Namen der Parameter vollständig ausgeschrieben. Es werden also keine abgekürzten Parameternamen verwendet und auch auf den Einsatz von Positionsparametern verzichtet. Das macht den Code etwas länger, aber hoffentlich ist er so leichter zu lesen und zu verstehen.

Beim Schreiben dieses Buches habe ich mir vorgenommen, Inhalte zu einer Reihe von Features von Windows Server 2019 zu erstellen. Im Verlauf des Buches haben wir schnell mehrere inhaltliche Grenzen erreicht (und überschritten). Um das Buch zu veröffentlichen, war es notwendig, einige Inhalte zu entfernen, was wir nur sehr widerwillig taten. Fast jedes der Kapitel hätte leicht ein eigenes kleines Buch werden können. Um Jeffrey Snover zu paraphrasieren: *To ship is to choose*. Ich hoffe, ich habe die richtige Wahl getroffen.

Einige Rezepte in diesem Buch setzen voraus, dass andere Rezepte bereits abgeschlossen wurden. Diese miteinander zusammenhängenden Rezepte funktionierten gut, als wir sie schrieben, und funktionieren hoffentlich auch bei Ihnen.

Es gibt einen feinen, aber wichtigen Unterschied zwischen PowerShell und einem Windows-Feature. Um PowerShell zum Verwalten eines Windows-Features zu verwenden, müssen Sie das Feature selbst verstehen. Die Kapitel beschreiben kurz jedes Feature. Da der Platz begrenzt ist, kann ich nicht zu jedem Feature alle Details beschreiben. Jedoch finden Sie im Buch zahlreiche Links, die helfen, weiterführende Informationen zu finden. Und wie immer sind Bing und Google Ihre besten Freunde.

Konventionen in diesem Buch

Dieses Buch verwendet eine Reihe von typografischen Konventionen.

Schrift mit fester Laufweite wird für Code im Fließtext, für Namen von Datenbanktabellen, Ordernamen, Dateinamen, Dateierweiterungen, URLs, Benutzereingaben, Namen der Server usw. verwendet. Hier ein Beispiel: »Abschließend rufen Sie die Funktion `Get-HelloWorld` auf.«

Codeblöcke werden wie folgt dargestellt:

```
$Sb = {
    Install-WindowsFeature -Name Hyper-V -IncludeManagementTools
```

```
Invoke-Command -ComputerName HV1, HV2 -ScriptBlock $Sb
```

Fette Schrift wird für neue Begriffe verwendet, für ein wichtiges Wort und für Wörter, die Sie beispielsweise in Menüs oder Dialogfeldern auf dem Bildschirm sehen. Hier ein Beispiel: »Wählen Sie im Aufgabenbereich **Verwaltung** den Eintrag **Systeminformation**.«

Hinweis Warnungen und wichtige Hinweise werden so dargestellt.

Um die Lesbarkeit der Bildschirmabbildungen im Buch zu verbessern, wurde die PowerShell ISE so konfiguriert, dass sie schwarzen Text auf einem hellgrauen Hintergrund verwendet.

Abschnitte und Aufbau der Rezepte

Sie finden in diesem Buch verschiedene Überschriften, die immer wieder auftauchen (*Vorbereitungen*, *So wird's gemacht*, *So funktioniert's*, *Weitere Informationen* und *Siehe auch...*).

Um klare Anweisungen zum Vervollständigen eines Rezepts zu geben, verwenden wir diese Abschnitte wie folgt.

Überschriften der Kapitel und der Rezepte

Jedes Kapitel und jedes Rezept enthalten eine kurze Einführung in den Teil von Windows, den Sie mithilfe des Rezepts administrieren können. Ich habe versucht, die wichtigsten Punkte zu jedem Feature zusammenzufassen – aber wie immer gibt es viel mehr Details, die Sie mit Ihrer Liebessuchmaschine entdecken können.

Vorbereitungen

In diesem Abschnitt erfahren Sie, was Sie im Rezept erwarten können. Außerdem beschreibt er die Softwarevoraussetzungen und weitere Einstellungen, die für das Rezept erforderlich sind. Außerdem erfahren Sie hier, welche Hosts (VMs) Sie für das Rezept benötigen, und welche Dateien, Ordner und andere Ressourcen Sie brauchen, um das Rezept erfolgreich abschließen zu können.

So wird's gemacht

Dieser Abschnitt enthält die Schritte, die erforderlich sind, um dem Rezept zu folgen. Wir zeigen den PowerShell-Code, den Sie zum Ausführen der einzelnen Schritte verwenden. Mit Ausnahme des Kapitels über Container können alle Rezepte in der PowerShell ISE verwendet und ausgeführt werden. Im Kapitel über die Container sollten Sie die PowerShell-Konsole verwenden, um die einzelnen Schritte des Rezepts auszuführen.

So funktioniert's

Dieser Abschnitt enthält eine detaillierte Erklärung dessen, was im vorherigen Abschnitt passiert ist. Hier finden Sie auch die Bildschirmabbildungen, die die Ergebnisse der einzelnen Schritte des Rezepts zeigen.

Weitere Informationen

Dieser Abschnitt enthält zusätzliche Hintergrundinfos, damit Sie besser verstehen, was genau im Rezept abläuft.

Siehe auch...

Bei einigen Rezepten enthält dieser Abschnitt Links zu weiterführenden Informationen, die Sie nützlich finden können.

Die Begleitdateien

Die Begleitdateien zur deutschen Ausgabe dieses Buches können Sie kostenlos unter <https://www.oreilly.de/buecher/13320.html> herunterladen. Diese PowerShell-Skripte sind komplett lokalisiert. Dies bedeutet, dass nicht nur die Kommentare übersetzt wurden, sondern auch der Code so angepasst wurde, dass er mit der deutschsprachigen Version von Windows Server 2019 funktioniert.

Hierfür waren an einigen Stellen Anpassungen erforderlich (beispielsweise bei der Verwendung der vordefinierten AD-Benutzerkonten und der Namen der Leistungsindikatoren). Dies betrifft insbesondere die Rezepte in Kapitel 3 und Kapitel 13. Falls Sie eine englischsprachige Version von Windows Server 2019 verwenden, können Sie die Beispieldateien der englischen Originalausgabe verwenden. Sie finden diese auf GitHub unter <https://github.com/doctordns/PowerShellCookBook2019>

Support und Feedback

Mit Anmerkungen, Fragen oder Verbesserungsvorschlägen zu diesem Buch können Sie sich gerne an den Verlag wenden:

komentar@oreilly.de

Bitte beachten Sie, dass über unsere E-Mail-Adresse kein Software-Support angeboten wird. Für Supportinformationen bezüglich der hier verwendeten Microsoft-Produkte besuchen Sie bitte die Microsoft-Website:

<https://support.microsoft.com>