کانتینرها چه هستند و چه ارتباطی با ویندوز سرور 2019 دارند؟

بیشتر فناوری‌های جدید ویندوز سرور 2019 به گونه‌ای طراحی شده‌اند که انعکاس دهنده قابلیت‌های رایانش ابری هستند. نسخه‌های جدیدتر ویندوز سرور فناوری‌هایی همچون مجازی‌سازی را بهبود بخشیند و از ایده کانتینرها و نحوه به‌کارگیری ساده‌تر، ایمن‌تر و کارآمدتر آن‌ها پشتیبانی کردند. کانتینرها ایده نسبتاً جدیدی در دنیای مایکروسافت هستند. توسعه‌دهندگان برنامه‌ها نسبت به ورود کانتینترها به دنیای سیستم‌عامل‌های سرور مایکروسافت ابراز خوشحالی کردند، زیرا درک مفاهیم کانیتنرها به مراتب ساده‌تر از مباحث مدیریت سنتی سرور است.

برای مطالعه قسمت قبل آموزش رایگان **ویندوز سرور 2019 اینجا** کلیک کنید.

تخصیص مجوز دسترسی به سرورها توسط ماشین‌ها یا ایستگاه‌های کاری غیر عضو

اگر با سرورهایی کار می‌کنید که همگی بخشی از یک دامنه شرکتی هستند که در بیشتر موارد این‌گونه است، فرآیند احراز هویت بین دستگاهی به راحتی انجام می‌شود، زیرا به‌طور خودکار در این سطح به یکدیگر اعتماد دارند. با این حال، اگر قرار است روی سروری که آماده دریافت اتصالات از راه دور است؛ ارتباطات مربوط به ماشین‌هایی که عضو دامنه نیستند یا عضوی از یک ایستگاه کاری هستند را وارد کنید، مجبور هستید به شکل دستی فرمانی را فراخوانی کنید تا سایر کامپیوترها بتوانند به آن متصل شوند. به عنوان مثال، اگر قرار است تمامی سرورها را توسط یک کامپیوتر کلاینت که Win10Client نام دارد و مورد تایید سرورها نیست مدیریت کنم، باید روی سرورها فرمان زیر را اجرا کنم.

Set-Item wsman:\localhost\client\trustedhosts Win10Client

اگر در نظر دارید به هر دستگاهی از راه دور وصل شوید، می‌توانید نام رایانه شخصی را با یک * جایگزین کنید، اما به‌طور کلی، این روش خوبی نیست، زیرا ممکن است به هر دستگاهی اجازه دهید به این روش به سرورتان متصل شود.

اتصال به سرور از راه دور

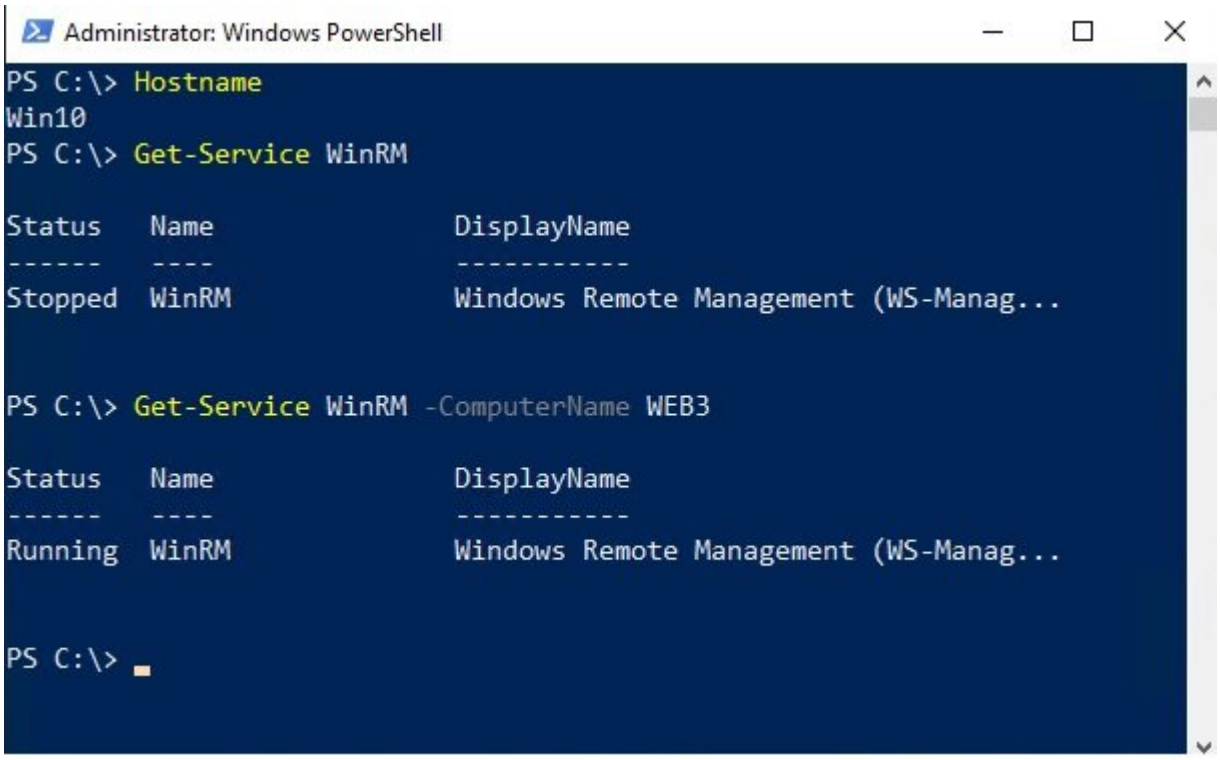
به‌طور معمول مشاهده می‌کنیم که مدیران به دو روش از PowerShelling استفاده می‌کنند. شما می‌توانید به شکل موقت برخی از دستورات را روی سامانه‌های راه دور اجرا کنید یا می‌توانید یک نشست از راه دور کامل برای پاورشل راه‌اندازی کنید تا بتوانید در کوتاه‌ترین زمان پاورشل را برای برقراری یک ارتباط راه دور تنظیم کنید. اجازه دهید هر دو گزینه را بررسی کنیم.

به‌کارگیری -ComputerName

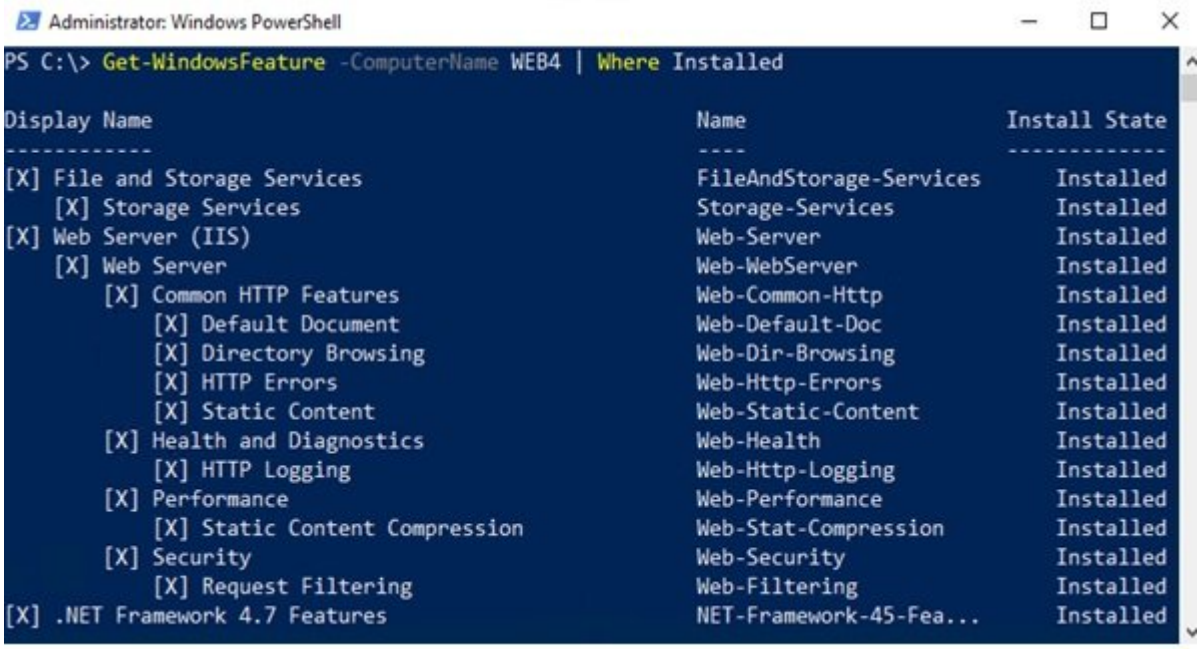
بیشتر فرمان‌های پاورشل به ویژه فرامینی که با Get- آغاز می‌شوند قابل استفاده با پارامتر -ComputerName هستند. این پارامتر نشان می‌دهد که شما قصد اجرای فرامین روی یک ماشین راه دور را دارید. به‌طور مثال، می‌توانید روی کامپیوتر کلاینت ویندوز 10 برای دسترسی به اطلاعاتی که روی سروری در شبکه‌تان قرار دارند از پاورشل استفاده کنید. فرض کنید قصد اجرای محاوری روی سرویس WinRM دارید تا مطمئن شوید سرویس فوق در حال اجرا است. برای انجام این‌کار از فرامین زیر استفاده می‌کنیم. توجه داشته باشید در مثال زیر WEB3 نام سرور ما است و شما باید نام سرور خود را وارد کنید.

```
Hostname
Get-Service WinRM
Get-Service WinRM -ComputerName WEB3
```

در تصویر زیر مشاهده می‌کنید که سرویس WinRM محلی متوقف شده، اما زمانی که همان دستور را به صورت ComputerName WEB3 اجرا می‌کنم (دستوری که قرار است روی سرور اجرا شود)، مشاهده می‌کنیم که سرویس WinRM در سرور فعال است.

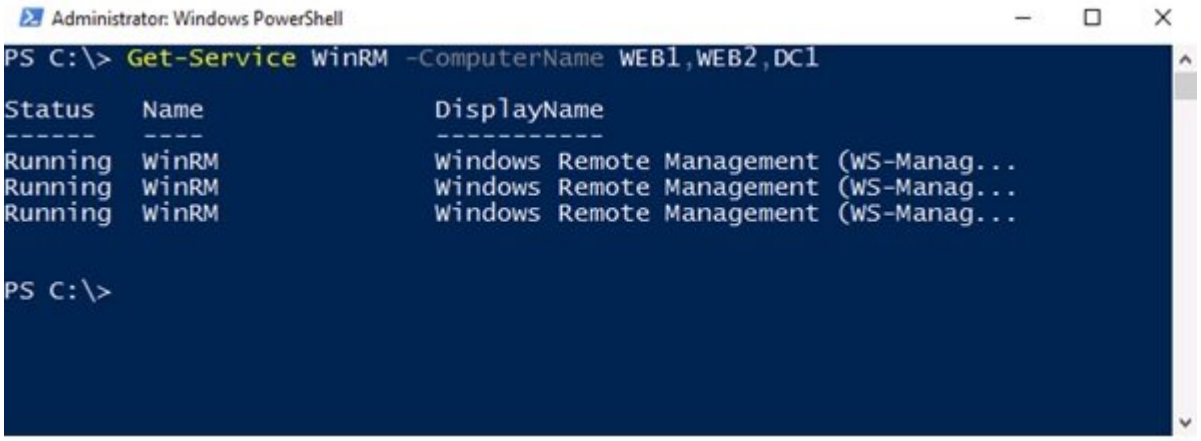


شاید بخواهم محاوره‌ای روی یک نمونه Server Core اجرا کنیم تا ببنیم چه نقش‌هایی در حال حاضر روی WEB4 نصب شده‌اند، برای انجام این‌کار فرمان زیر را اجرا می‌کنیم.




```
Get-WindowsFeature -ComputerName WEB4 | Where Installed
```

پارامتر -ComputerName می‌تواند نام چند سرور را همزمان دریافت کند. اگر در نظر داشتیم تا وضعیت سرویس WinRM را روی سرورهای مختلف و با یک دستور مشاهده کنیم از فرمان زیر استفاده می‌کنیم.



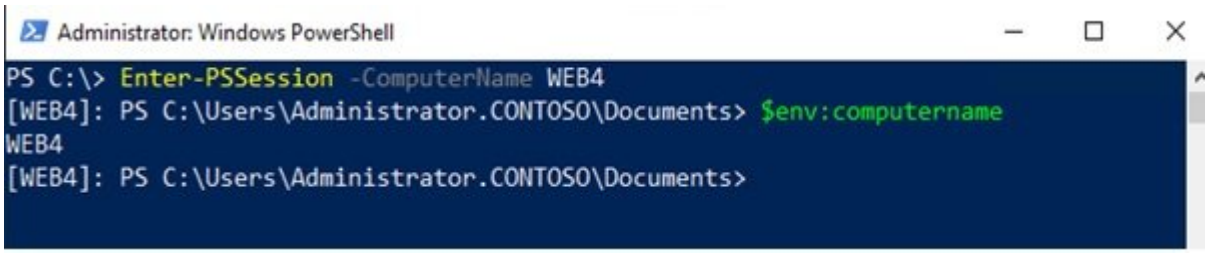

```
Get-Service WinRM -ComputerName WEB1,WEB2,DC1
```

به‌کارگیری Enter-PSSession

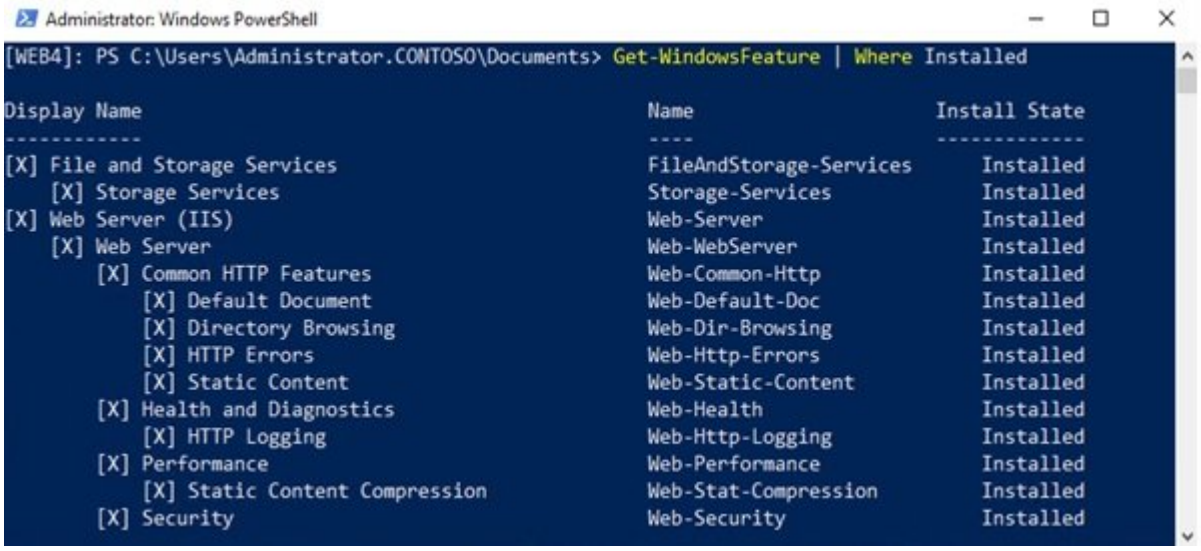
گاهی اوقات فرمان‌های مختلف زیادی باید روی یک سرور خاص اجرا شود. در این حالت ، فراخوانی بیشتر از مثال PowerShell کاملاً توانمند و کاملاً از راه دور در آن سرور از راه دور ، منطقی تر است. اگر پاورشل را روی سیستم محلی خود باز کنید و فرمان Enter-PSSession را همراه با نام سروری اجرا کنید، خط فرمان پاورشل آماده دریافت دستوری می‌شود که قرار است روی سرور راه دور اجرا شوند. به‌طور مثال فرمان زیر را تصور کنید که برای اتصال به سرور WEB4 اجرا می‌شود:

```
Enter-PSSession -ComputerName WEB4
```

با اجرای فرمان فوق به سرعت مشاهده می‌کنید که پاورشل تغییر وضعیت می‌دهد و به سرور WEB4 متصل می‌شود. اکنون می‌توانیم دستوراتی که قرار است روی سرور WEB4 اجرا شوند را وارد کنیم. دقت کنید که نام میزان WEB4 نشان داده می‌شود.

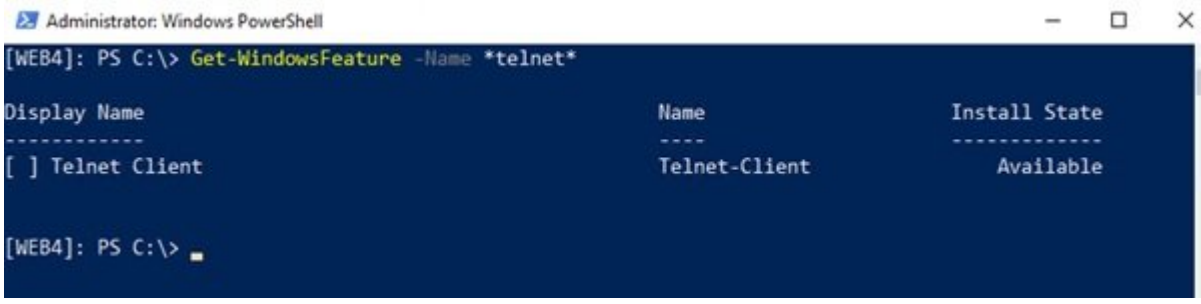


به‌طور مثال، برای مشاهده نقش‌ها و مکان نصب آن‌ها روی سرور (WEB4) که درون آن قرار داریم از فرمان زیر می‌توانیم استفاده کنیم.



Get-WindowsFeature | Where Installed

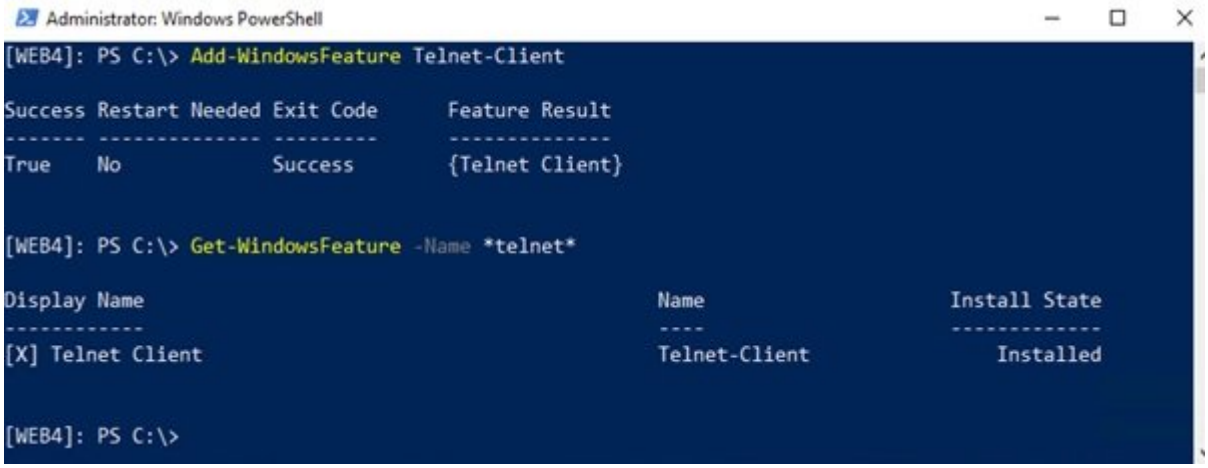
راهکار فوق قدرتمند است. ما در کامپیوتر دسکتاپ محلی خود قرار داشیم، یک نشست راه‌دور پاورشل به سرور خود (WB4) برقرار کردیم و اکنون می‌توانیم به واکشی انواع مختلفی از اطلاعاتی بپردازیم که درون سرور WEB4 قرار دارد. اجازه دهید یک قدرم به جلوبرداریم و تغییری در پیکربندی WEB4 اعمال کنیم. برخی از مدیریان شبکه از Telnet Client برای آزمایش اتصال شبکه استفاده می‌کنند، اما ممکن است در برخی از سرورها همچون مثال ما (WEB4) نصب نشده باشد.



Get-WindowsFeature -Name *telnet*

با استفاده از فرمان Add-WindowsFeature به سرعت می‌توانیم ویژگی فوق را روی سرور نصب کنیم.

Add-WindowsFeature Telnet-Client



صحبت درباره پاورشل زیاد است، زیرا یکی از مولفه‌های مهم سرورها است که تقریبا اجازه انجام هر کاری را به مدیران شبکه می‌دهد. ما در این‌جا مبحث پاورشل را خاتمه می‌دهیم و به سراغ مبحث کانتینرها و نانو سرور می‌رویم.

کانتینرها و نانو سرور

بیشتر فناوری‌های جدید ویندوز سرور 2019 به گونه‌ای طراحی شده‌اند که انعکاس دهنده قابلیت‌های رایانش ابری هستند. نسخه‌های جدیدتر ویندوز سرور فناوری‌هایی همچون مجازی‌سازی را بهبود بخشیند و از ایده کانتینرها و نحوه به‌کارگیری ساده‌تر، ایمن‌تر و کارآمدتر آن‌ها پشتیبانی کردند. کانتینرها ایده نسبتاً جدیدی در دنیای مایکروسافت هستند. توسعه‌دهندگان برنامه‌ها نسبت به ورود کانتینترها به دنیای سیستم‌عامل‌های سرور مایکروسافت ابراز خوشحالی کردند، زیرا درک مفاهیم کانیتنرها به مراتب ساده‌تر از مباحث مدیریت سنتی سرور است.

کانیتنرها چه هستند؟

کانتینرها چه معنایی دارند و منظور از داشتن یک کانتینر چیست؟ این روزها مبحث مجازی‌سازی سرور به یکی از موضوعات داغ دنیای فناوری تبدیل شده است. تهیه یک سخت‌افزار فیزیکی، تبدیل آن به یک میزبان مجازی (Hyper-V) و سپس اجرای ماشین‌های مجازی روی آن یکی از مهارت‌های روز است. در رویکرد فوق ما منابع و سخت‌افزار یک سرور را با ماشین‌های مجازی که روی آن در حال اجرا هستند به‌اشتراک قرار می‌دهیم. در همان زمان که منابع سخت‌افزاری را به اشتراک می‌گذاریم، می‌توانیم به بهترین شکل ماشین‌های مجازی را از یکدیگر متمایز کنیم و مطمئن شویم که ماشین‌ها هیچ‌گونه دسترسی به یکدیگر ندارند و اگر برای یک ماشین اتفاقی رخ دهد، روی عملکرد سایر ماشین‌ها اثرگذار نیست. مفهوم Application container رویکردی مشابه با ماشین‌های مجازی دارد، اما در سطح متفاوتی اجرا می‌شود. در حالی که ماشین‌های مجازی حول محور سخت‌افزاری مجازی کار می‌کنند، کانتینرها بیشتر در ارتباط با مجازی‌سازی سیستم‌عامل هستند. به جای این‌که ماشین مجازی را برای میزبانی برنامه‌های خود ایجاد کنیم، ما کانتینرها را می‌سازیم که به مراتب کوچک‌تر هستند. پس از ایجاد کانتینرها قادر هستیم برنامه‌های درون کانتینرها را اجرا کنیم، در حالی که برنامه‌ها تصور می‌کنند روی یک نمونه اختصاصی از یک سیستم‌عامل در حال اجرا هستند. مزیت بزرگی که کانتینرها دارند به یکپارچگی و هماهنگی که میان تیم‌های توسعه و عملیات ایجاد می‌شود باز می‌گردد. ما این روزها اصطلاح دوآپس را زیاد می‌شنویم که ترکیبی از فرآیندهای توسعه و عملیات است تا فرآیند ساخت برنامه‌ها کارآمدتر شود. کانتینرها روی مفهوم دوآپس تأثیر زیادی گذاشتند و به توسعه‌دهندگان اجازه دادند تا بدون نیاز به تأمین برنامه‌های جانبی و زیرساخت‌های مربوطه به شیوه ساده‌تری فرآیند توسعه برنامه‌ها را دنبال کنند.

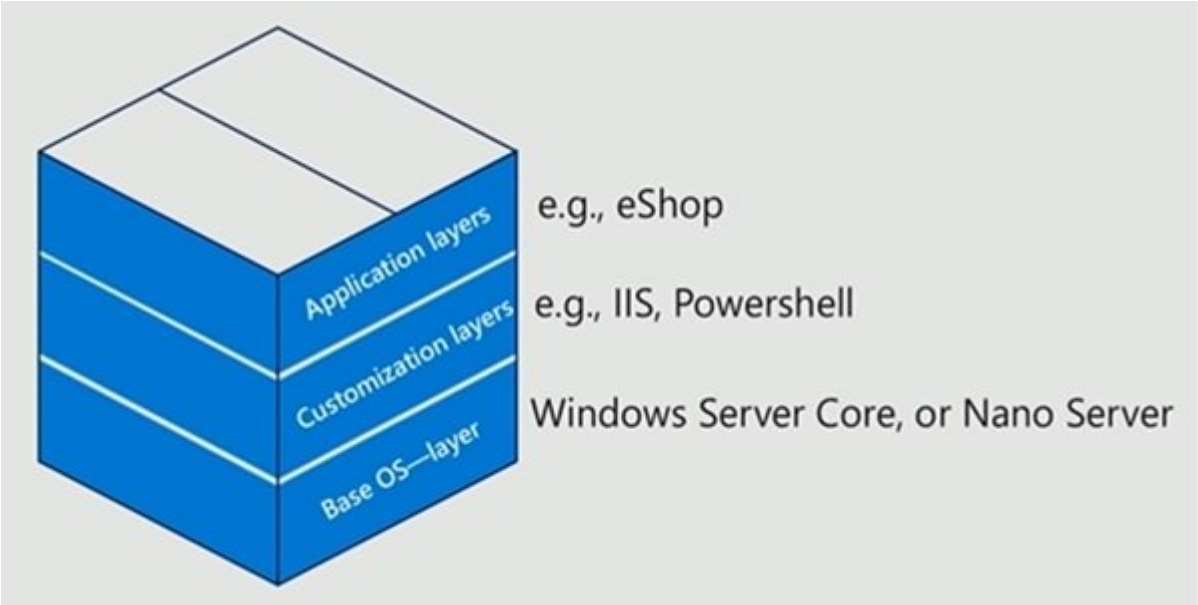
به‌اشتراک‌گذاری منابع

درست مانند زمانی که در مورد تقسیم سخت‌افزار در میان ماشین‌های مجازی صحبت می‌کنیم، چنین مفهومی در مورد کانتینرها نیز صدق می‌کند، به این معنا که برنامه بخش‌های سخت‌افزاری فیزیکی را دریافت می‌کند و میان کانتینرها تقسیم می‌کند. راهکار فوق اجازه می‌دهد تا کانتینرهای بیشتری را روی یک سرور اجرا کنیم، صرفنظر از این‌که در حال استفاده از یک سرور مجازی یا فیزیکی هستیم.

در ظاهر این‌گونه به نظر می‌رسد که کانتینرها هیچ‌گونه مزیتی نسبت به ماشین‌های مجازی ندارند، زیرا به راحتی سخت‌افزار را به‌اشتراک می‌گذارند، اما تفاوت مهمی وجود دارد، تمامی کانتینرها می‌توانند یک سیستم‌عامل پایه یکسان را همراه با منابع مشابه به‌اشتراک بگذارند. از دیگر مفاهیم مهمی که پیرامون کانتینرها قرار دارد به ایزوله‌سازی که به یک توسعه‌دهنده اجازه می‌دهد تا برنامه‌های خود را روی یک کانیتنر و روی ایستگاه کاری خود اجرا کند و گسترش‌پذیری اشاره کرد که هر یک از این مفاهیم خود توضیح مفصل دارند.

کانتینرها و نانو سرور

قبل از پرداختن به موضوع نانو سرور اجازه دهید نگاهی گذرا به ساختار یک کانتینر مبتنی بر ویندوز داشته باشیم.



در تصویر فوق پایین‌ترین لایه کانتینر سیستم‌عامل پایه است. این سیستم عامل‌پایه می‌تواند یا Server Core یا Nano Server باشد.

لایه بعدی کانتینری است که درون یک لایه سفارشی قرار دارد. فناوری‌هایی که قرار است توسط برنامه‌های شما استفاده شوند در این لایه قرار می‌گیرند. به عنوان مثال، کانتینرها می‌توانند شامل IIS برای میزبانی وب‌سایت‌ها، پاورشل یا حتی ماهیت‌های بزرگ‌تری همچون دات‌نت باشند. تمامی این مولفه‌ها در لایه فوق قرار دارند.

آخرین لایه، لایه کاربرد است. در این لایه برنامه‌ای که قصد دارید درون کانتینر میزبانی شود را تعیین می‌کنید تا سایر کاربران بتوانند به آن دسترسی داشته باشند.

در حالی که Server Core یک سیستم‌عامل عالی برای ساخت سرورهای کوچک و کارآمد است، اما در مقایسه با Nano Server کمی حجیم‌تر است. دقت کنید که یک ایمیج پایه نانو سرور می‌تواند کمتر از 500 مگابایت باشد!

Nano Server اکنون فقط به عنوان یک سیستم‌عامل پایه برای کانتینرها در نظر گرفته می‌شود. این تغییر بزرگ از زمان انتشار سرور 2016 رخ داد. با توجه به محدودیت‌هایی که پیرامون نانوسرور به وجود آمده مایکروسافت پیشنهاد کرده تا شرکت‌ها به سراغ Server Core بروند.

برخ از سرپرستان شبکه سوال می‌کنند که چرا باید Server Core به عنوان یک ایمیج کانتینتر استفاده شود؟ ساده‌ترین پاسخ برای این سوال به سازگاری برنامه‌ها باز می‌گردد. نانو سرور فوق‌العاده کوچک است و تنها کدهای ضروری هسته در آن قرار گرفته است. هنگامی که به دنبال استفاده از کانتینرها برای میزبانی برنامه‌های خود هستید، ایده خوبی است که در صورت امکان از نانو سرور به عنوان پایه استفاده کنید، اما غالباً برنامه‌های شما به راحتی روی آن اجرا نمی‌شوند، در چنین شرایطی بهتر است از Server Core به عنوان سیستم‌عامل پایه استفاده کنید.

کانتینرهای ویندوز سرور در مقابل کانتینرهای Hyper-V

دقت کنید که دو گروه کانتینترها وجود دارند که می‌توانید روی ویندوز سرور 2019 اجرا کنید. تمامی خصایص کانتینترهایی که در مورد آن‌ها صحبت کردیم در مورد هر دو گروه Windows Server containers و Hyper-V containers صدق می‌کند. کانتینرهای Hyper-V همانند کانتینرهای ویندوز سرور می‌توانند ایمیج‌ها یا کدهای یکسانی را درون خود اجرا کنند و همچنین یک مکانیزم یکپارچه جداسازی را ارائه دهند. تصمیم‌گیری در مورد استفاده از Windows Server Containers یا Hyper-V Containers به خود شما بستگی دارد که به چه میزان امنیت در ارتباط با کانتینرها نیاز دارید.

Windows Server Containers

مشابه با روشی که کانتینرهای لینوکس فایله‌ای هسته سیستم‌عامل میزبان را به‌اشتراک می‌گذارند، کانتینرهای ویندوز سرور نیز بر مبنای روش مشابهی چنین کاری را انجام می‌دهد. به عبارت دیگر، در حالی که فضای نام، فایل سیستم و جداسازی شبکه برای جداسازی کانتینرها از یکدیگر اعمال می‌شوند، این احتمال وجود دارد که آسیب‌پذیری‌هایی میان کانتینرهای مختلف ویندوز سروری که روی یک میزبان در حال اجرا هستند به وجود آید. به عنوان مثال، اگر می‌خواهید به سیستم‌عامل میزبان روی سرور کانتینر خود وارد شوید، می‌توانید فرایندهای در حال اجرا روی هر کانتینر را مشاهده کنید. کانتینر قادر به مشاهده میزبان یا سایر کانتینرها نیست و هنوز هم به روش‌های مختلفی از میزبان جدا شده، اما دانستن اینکه میزبان قادر به مشاهده فرایندهای داخل کانتینر است به ما نشان می‌دهد که ممکن است برخی تعامل با میزبان به‌اشتراک قرار گیرد. کانتینرهای ویندوز سرور در شرایطی که سرور میزبان کانتینر و خود کانتینر در محدوده مطمئنی هستند و به یکدیگر اعتماد دارند مفید هستند. به بعارت دقیق‌تر کانتینترهای ویندوز سرور بیشتر برای سرورهایی که تحت مالیکیت شرکت قرار دارند و خود شرکت قادر به مدیریت آن‌ها است مفید هستند. اگر به سرور میزبان و کانتینر خود اعتماد دارید استفاده از کانتینرهای ویندوز سرور کارآمدترین روش به‌کارگیری از منابع سخت‌افزاری را ارائه می‌کنند.

کانتینرهای Hyper-V

اگر به دنبال افزایش سطح بیشتری از ایزوله‌سازی و مرزبندی هستید، Hyper-V Containers پاسخ‌گوی نیاز شما هستند. کانتینرهای Hyper-V شباهت زیادی به یک نسخه بهینه شده از یک ماشین مجازی دارند. در حالی که منابع هسته هنوز توسط Hyper-V Containers به‌اشتراک گذاشته می‌شوند، اما عملکرد بهتری نسبت به ماشین‌های مجازی دارند. هر کانتینر Hyper-V پوسته اختصاصی ویندوز خاص خود را دارد که کانتینر در آن پوسته اجرا می‌شود. کانتینرهای Hyper-V در زیرساخت‌های چند مستاجر عملکرد خیلی خوبی دارند، جایی که می‌خواهید مطمئن شوید هیچ کد یا فعالیتی قادر به استخراج اطلاعات میان کانتینر و میزبان نیست. در پاراگراف قبل به این موضوع اشاره کردیم که چگونه سیستم‌عامل میزبان می‌تواند فرآیندهای موجود در یک Windows Server Container را مشاهده کند، اما در مورد Hyper-V Containers چنین وضعیتی وجود ندارد. سیستم‌عامل میزبان هیچ اطلاعی در مورد سرویس‌ها ندارد و قادر نیست سرویس‌هایی که درون Hyper-V container اجرا می‌شوند را مشاهده کند. به عبارت دیگر فرآیندها کاملا نامرئی هستند.

در شماره آینده مبحث کانتینرها در **ویندوز سرور 2019** را ادامه خواهیم داد.

برای مطالعه تمام بخش‌های آموزش **ویندوز سرور 2019** روی لینک زیر کلیک کنید:

آموزش رایگان ویندوز سرور 2019

تاریخ انتشار:
27 آذر 1398

نشانی منبع:
https://www.shabakeh-mag.com/networking-technology/16390/%DA%A9%D8%A7%D9%86%D8%A

A%DB%8C%D9%86%D8%B1%D9%87%D8%A7-%DA%86%D9%87-
%D9%87%D8%B3%D8%AA%D9%86%D8%AF-%D9%88-%DA%86%D9%87-
%D8%A7%D8%B1%D8%AA%D8%A8%D8%A7%D8%B7%DB%8C-%D8%A8%D8%A7-
%D9%88%DB%8C%D9%86%D8%AF%D9%88%D8%B2-%D8%B3%D8%B1%D9%88%D8%B1-2019-
%D8%AF%D8%A7%D8%B1%D9%86%D8%AF%D8%9F