gamechanger Game Changing Technology for Future Proofing Your Applications

What to Do Before Microsoft Ends Support for Windows Server 2008

By Brien M. Posey

Microsoft is ending its extended support for Windows Server 2008 and Windows Server 2008 R2 on January 14, 2020. While this date may seem far away, it isn't actually that far off. January 2020 is barely a year away. That does not leave a lot of time for IT pros to come up with a strategy for migrating applications away from Windows Server 2008, and on to a more current platform.

The most obvious solution to the looming end of support date is to migrate applications to a newer version of Windows Server. Doing so however, may not be the best option.

For one thing, not every application that was designed to run on Windows Server 2008 will run on more modern versions of Windows Server including Windows Server 2019. The Windows Server operating system has undergone massive changes in the past decade, and there is a very real chance that these changes will cause problems for legacy applications. Even if an application can be shimmed and run on a newer operating system, there could end up being stability, compatibility, or even data integrity issues that arise as a result of doing that.

Organizations who wish to move legacy applications to a newer version of Windows Server will need to perform

extensive testing to ensure that the applications work as intended in the new environment, without any surprises. In doing so, IT pros are likely to find that some of the applications need to be refactored (recoded) in order to work with the new operating system.

Application migrations, by their very nature, are expensive. There is a significant cost involved in thoroughly testing each application. Furthermore, the testing process ties up IT staffing resources. Being that most IT departments are already stretched thin, committing staff to testing legacy applications might not be the best use of resources.

The lack of active security patching is by far the most compelling reason to move applications off of Windows Server 2008.

Of course the organization must also come up with a plan for dealing with applications that are found to be incompatible with the new operating system. If the application was developed in house, there is a chance that the application can be updated. It is however, doubtful that an organization would be able to recode a commercial application. The organization would more likely have to track down a newer version of the application, or a competing product if the vendor no longer supports the application. In either case, the new application or new application version would also have to be tested to ensure that it does not cause any problems for the organization.

While it is easy to bemoan the cost, complexity, and workload that are incurred in the adoption of a new server operating system, there is a bigger issue that must be considered. Namely, making the transition to a new server operating system is not a one time process. Even though Windows Server 2019 is brand new, there will come a day when it too is outdated and it will be necessary to begin looking at how to migrate the organization's applications to yet another Windows Server release.

Given the fact that application migration planning is such a tedious, time consuming, and expensive process, organizations should seriously consider if such a migration is truly in their best interest. It would be one thing if the migration were a one-time process, but organizations can realistically expect to deal with the application migration issue every few years.

So if an organization were to determine that an endless cycle of application migrations are not in its best interest, what are the other options?

One option is to continue running the applications on Windows Server 2008 for perpetuity. On the surface, this seems like a really attractive option. After all, the server licenses are already bought and paid for, and the applications have been running on the operating system for long enough to be considered stable and reliable.

SECURITY ISSUES

Even though the Windows Server 2008 operating system will continue to function after its effective end of life date, Microsoft will no longer support it. The larger issue however, is that once Microsoft stops supporting the operating system, it will also cease to create security patches for it.

The lack of active security patching is by far the most compelling reason to move applications off of Windows Server 2008. Because no new security patches are being created, anyone who continues to run the aging operating system will be left vulnerable to any future security holes that are discovered. This is obviously unacceptable for businesses in regulated industries such as healthcare or banking, but is no less consequential for businesses in non-regulated industries.

The past several years have seen numerous retailers, government agencies, and financial services companies fall victim to data breaches. In each case, the affected organization suffered significant financial harm stemming from factors such as loss of customers, civil litigation, and regulatory penalties, just to name a few. Organizations simply cannot afford to risk running applications on a potentially insecure operating system.

So if continuing to run applications on Windows Server 2008 is an unacceptable option, and migrating to a new Windows Server operating system proves to be too costly (or perhaps even logistically impossible), what options are left?

Unfortunately, the need for migrating applications off of Windows Server 2008 systems is unavoidable. Even so, that does not mean that an organization must incur the cost and hassle of performing similar migrations every few years. Rather than migrating applications from one Windows Server version to the next as has been done in the past, it may be more effective to look for ways to future proof their applications. One especially compelling option is to containerize the applications.

DOCKER OFFERS A SOLUTION

Docker base containerization is an excellent alternative to performing a traditional application migration, because it addresses nearly all of the usual migration pain points. More importantly though, containers are for all practical purposes, future proof.

One of the things that make containers such a compelling option is their portability. Once an application has been containerized, that container can be moved from system to system on an as needed basis. This portability is especially useful in migration scenarios. An administrator might for example, create a containerized version of an application, and then test that application in a test / dev environment. Once the application has been validated, it can be immediately moved into the production environment.

Container portability is also useful for another reason. One of the biggest IT trends over the last several years has been the gradual transition to the cloud. Because all of the major IaaS public cloud providers support containerization, it is really easy to move a container out of your own datacenter and into the cloud. Even if you do not have any plans to ever run your applications in the cloud, it is nice to know that you have an easy option for cloud migration should your business needs ever change.

Of course containerization also insulates your applications from the never ending cycle of server operating system upgrades. While it is true that the container hosts will probably need to be upgraded from time to time, the containerized applications are completely self-contained. This means that performing an upgrade on a container host does not necessitate application compatibility testing. The containerized applications will continue to function just as they always have. When the next version of Windows Server (or whatever container host operating system you decide to use) is released, you can be confident that it will support your applications.

Containerization can also help to greatly reduce application cost. Containers usually have a much smaller footprint than virtual machines. This means that a container host can typically run a greater number of workloads than a similarly configured virtualization host would be able to handle. This higher density translates directly into cost savings because the hardware is being used more efficiently. As such, you may be able to reduce your hardware footprint, which also brings down the power and cooling costs and hardware maintenance costs. Depending on how you choose to implement the containers, you may even be able to drive down your licensing costs.

Containerization can also be a great way to reduce security and compliance risks. Containerized applications are more resistant to tampering than virtual machines are. This is partially because of the container architecture, and partly because of the various security features provided by Docker.

So even though the eventual end of support for Windows Server 2008 is unavoidable, you can take steps to ensure that you don't have to work through yet another tedious and costly application migration in a few years. Moving your applications off of Windows Server 2008 and into Docker is an easy way to make sure that you will be able to continue using your applications regardless of what the future may hold.

NEW DOCKER PROGRAM

Docker's Windows Server application migration program is the best and only way to containerize and secure legacy Windows Server applications while enabling software-driven business transformation. With industry-leading Docker Enterprise 2.1 and our best-in-class, industry-proven tools and services specifically designed for Windows Server application migration, customers can quickly and easily migrate and modernize legacy Windows applications while driving continuous innovation across any application, anywhere.

Find out more: https://www.docker.com/solutions/windows-server-migration

docker

tovovan / Shutterstock.com