JAMES-RANKIN.COM

technical how-tos | software reviews | industry news | forthright opinions | videos | podcasts | rants and raves

April 10, 2020 / James Rankin

Spreading users over multiple file shares with FSLogix Profile Containers

Search ...

Recent Posts

- Is it time to rethink application delivery? March 9, 2022
- Creating a Chocolatey package for the Citrix Virtual Delivery Agent January 24, 2022
- QuickPost – how to start a process when a Citrix published application launches

Introduction

The COVID-19 outbreak lies at the heart of this blog post. Recently, we were asked to scale up a Citrix deployment, shall we say, rather heavily, and indeed, rather rapidly. When you're talking about adding three-figure percentages to your workloads, you need to plan for an unprecedented capacity increase.

First port of call was Citrix infrastructure, which all looked good. Next we looked at network capacity – again, plenty of headroom for the expected increase. And then we came to storage – and oh boy, the numbers looked heavy.

Storage issues

The environment in question was already using FSLogix Profile Containers. The main thought was – how big do we possibly expect the profile to get for each user? This is a very important consideration in every Profile Containers deployment, and it's really impossible to tell without sending users into the environment to test it. There are of course maintenance routines to consider – shrinking, compacting, pruning, and (dare I say it?) exclusions – but this is all a matter for another post (coming soon!) In this environment, nothing had yet been put into place for maintenance purposes, so we had to have a look at existing users and try and take an educated guess at how much storage we potentially required for each user.

Most profiles were in the 5-10GB mark (Teams, it would appear, throws out 4.6GB of data every time it is run for the first time, although this might have been down to an erroneous

logon time optimizations, part #11 November 29, 2021
- The ultimate guide to Windows logon time optimizations, part #10 November 19, 2021

at OneDrive usage gave us an average of around 8-10GB – so being cautious, we anticipated looking at 30-40GB per user.

When you're talking about tens of thousands of users, this number gets very big, very quickly. Storage wasn't an issue (this was a cloud-based deployment), but the volumes we could attach to the file servers were limited to 16TB each, and therefore we needed quite a sizeable amount of these volumes. There were cloud services we could have leaned towards to accommodate this, but they had not yet been cleared for usage within the environment, and were unlikely to anytime soon.

You could stand up Scale Out File Services clusters and essentially combine all these volumes into a SAN-like pool (Leee Jefferies has done some great stuff on this), but again, this wasn't an option because it would involve architectural changes

The natural response to this problem is usually to front some DFS onto multiple shares, but several reasons prevented this – a) I hate DFS, b) there were authentication issues between the various domains in use and DFS would have exacerbated this, and c) directing users to DFS file shares seemed no more intelligent than simply directing them to a list of Windows file shares. The main problem we had was – what would happen when the first file share filled up? How would we direct users to the next one instead? The only way that it seemed possible to do this would be to use some sort of variable to direct subsets of users to particular file shares – but if something happened and one file share suddenly started using substantially more capacity than the others, we'd have to intervene and direct new users somewhere else.

Site archives

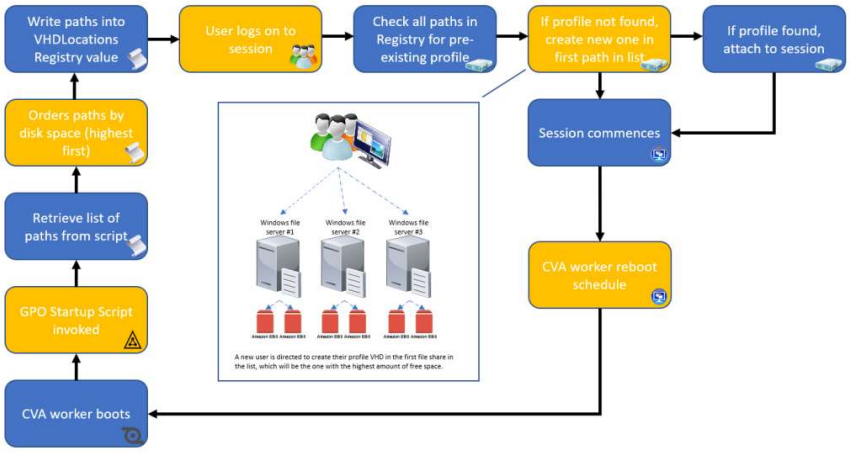
Select Month

Blogroll - CTPs

- Aaron Parker
- Benny Tritsch
- Carl Stalhood
- Carl Webster
- Christiaan Brinkhoff
- Dane Young
- Dave Brett
- Gareth Carson
- George Spiers
- Helge Klein
- Marius Sandbu
- Neil Spellings
- Nicolas Ignoto
- Patrick Coble
- Paul Stansel
- Remko Weijnen
- Rory Monaghan
- Ryan Revord
- Sacha Thomet
- Samuel Legrand
- Theresa Miller
- Thomas Poppelgaard
- Tim Mangan

Powershell to the rescue!

Ryan had experienced a similar problem, and his response was simply to use a PowerShell startup script to iterate through a list of file servers and order them by their free disk space. This value would then be written to the FSLogix Registry value for VHDLocations. So technically, new users would *always* hit the file share with the most available space. Essentially, as long as the script was run often enough for the volume of new user onboarding, the profiles would be load balanced across the file shares. Here's a diagram spelling out the process:-



When a user logs on, FSLogix iterates through the entire list of VHDLocations searching for an existing profile. If it finds one, fine. If it doesn't find one, though, it is created in the first entry in the list – which would be the file share with the most available space.

For our environment, we were potentially onboarding thousands of users a day, so we had to run this as a Scheduled Task rather than a Startup Script. However, as long as the Scheduled Task was run with admin access (so it could write an HKLM value), this worked fine. Also, it is worth noting that the script (below) uses a

Tags

Permissions-wise, you need to make sure that you give Domain Computers (or at the very least the Citrix worker computer accounts) RX access to the root of the share so that you can determine free space. If running as a Scheduled Task, obviously this also needs to include the user the task is configured to run as.

Here's the script – all that is needed is for you to populate the list of file share paths with your own, and it is ready to go! It populates both the Profile Containers and ODFC Registry values for VHDLocations, but if you don't use both, it won't make any difference – the unused one is just ignored. Big kudos to Ryan for his hard work on this!

```
#######################################
##### Begin profile path ordering ####
#######################################

$test=@()
$orderedShares=@()
$gb=(1024 * 1024 * 1024)
$tb=(1024 * 1024 * 1024 *1024)
# Share Array for stage values.....populate this array
with all of the storage paths that can be addressed in
the environment
$ProfileShares=@(
       "\\server1\share1"
       "\\server1\share2"
       "\\server2\share1"
       "\\server2\share2"
       "\\server3\share1"
       "\\server3\share2"
       "\\server4\share1"
       "\\server4\share2"
       "\\server5\share1"
       "\\server5\share2"
       "\\server6\share1"
```

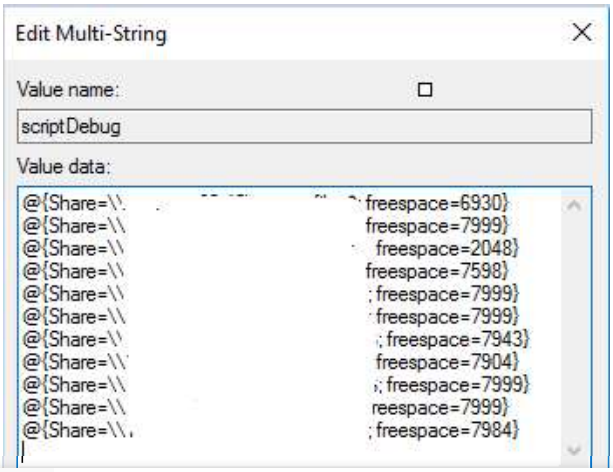
LICENSING (2)
LOGONS (14)
LOGON TIMES (11)
MFA (2)
MICROSOFT TEAMS (5)
MONITORING (14)
PERFORMANCE (2)
POLICYPAK (2)
POWERSHELL (11)
PROFILE CONTAINERS (3)
PROFILE MANAGEMENT (3)
PROFILES (12)
RDSH (7) REGISTRY (2)
SECURITY (5)
SERVER 2016 (2)
SERVER 2019 (3)
START TILES (3)
STORAGE (2)
TEAMS (4)
UBERAGENT (14)
UPM (3)
USER PROFILES (2)
UWP (2) WEM (2)
WINDOWS (16)
WINDOWS 10 (21)
WINDOWS LOGONS (9)
XENAPP (3)
XENDESKTOP (2)

```
    $nwobj=new-object -comobject WScript.Network
    $status=$nwobj.mapnetworkdrive("Z:",$share)
    $drive=get-psdrive Z
   $blah = [math]::Round($Drive.free / $gb)
    $shareSpace = New-Object -TypeName psobject
    $sharespace | Add-Member -membertype NoteProperty
-Name Share -value $share
    $sharespace | Add-Member -membertype NoteProperty
-Name freespace -value $blah
   $test+=$shareSpace
    # remove network driveexcel
    $status=$nwobj.removenetworkdrive("Z:")
}
$test2 = $test | Sort-Object -Descending freespace |
select share
foreach ($item in $test2) {
    $orderedShares +=  $item.Share.ToString()
}
# set FSlogix share path:
# cleanup

$FSLogixProfilePath="HKLM:\software\FSLogix\Profiles"

$FSLogixODFCPath="HKLM:\SOFTWARE\Policies\FSLogix\ODFC
"
       $FSLogixKeyName="VHDLocations"
              if ((get-item -path
$FSLogixProfilePath).GetValue($FSLogixKeyName) -ne
$null) {
                     Remove-itemProperty -path
$FSLogixProfilePath -Name $FSLogixKeyName -force
              } else {
                     # do nothing, no key to delete
              }
              if ((get-item -path
$FSLogixODFCPath).GetValue($FSLogixKeyName) -ne $null)
{
                     Remove-itemProperty -path
$FSLogixODFCPath -Name $FSLogixKeyName -force
              } else {
                     # do nothing, no key to delete
              }
```

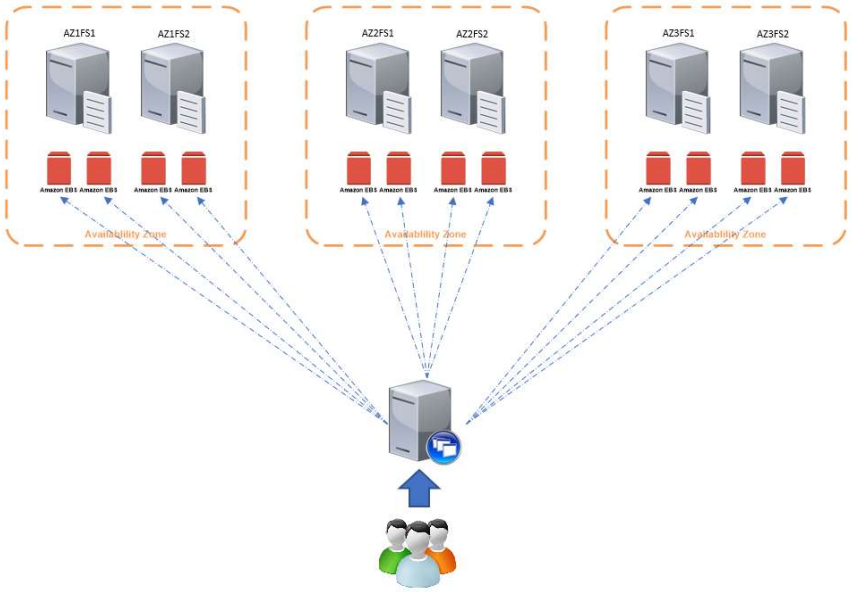
```
# ODFC Path
New-ItemProperty $FSLogixODFCPath -Name
$FSLogixKeyName -Value $orderedShares -PropertyType
MultiString -Force
#debug values, show me sizes
New-ItemProperty $FSLogixProfilePath -Name
"scriptDebug" -Value $test -PropertyType MultiString -
Force
New-ItemProperty $FSLogixODFCPath -Name "scriptdebug"
-Value $test -PropertyType MultiString -Force
#######################################
##### END profile path ordering ######
#######################################
```

The script should be run as required – Startup Script would be fine if your reboot schedule means that the amount of users you are onboarding in between reboots doesn't potentially exceed the capacity of a single file share, otherwise run on a Scheduled Task. We have been running it every hour, as we are seeing up to two thousand users per day being onboarded (and our Citrix workers are never rebooted anyway).

The script also writes a handy extra value to the Registry key called *scriptDebug* which shows the disk space of all the target file shares last time it was run – really handy



So with this being run, our users are directed to any one of (in this particular case) twelve file shares of 16TB each. This can simply be added to by provisioning additional file servers and volumes and adding them to the list in the script, so their new capacity will be instantly utilized.



Resiliency

Of course, this doesn't provide resiliency. There was no requirement here to provide resiliency, merely the capacity to absorb an unprecedented and exceptional uplift of user numbers. There is resiliency in terms of absorbing the loss of an Availability Zone, but only in terms of users being able to log on – the users in the AZ (or on that server, or file share) will still lose their profiles and have a new one created elsewhere.

If you do need resiliency for the profiles themselves then there are lots of options here (will hopefully cover some of them off in my upcoming blog post about FSLogix best practices). In this situation it simply wasn't required, because the business didn't

their data, Teams cache and Outlook cache and then put back any other minor customizations at their leisure. Not ideal for people like myself who like to provide a seamless, smooth roaming experience – but this was a risk that they chose to absorb.

Summary

So, Ryan has provided us with a nice simple way to spread users across multiple file shares in a quick and easy fashion, and to address the issue of each file share potentially filling up.

A lot of people will say "why not just use Cloud Cache", but there are a number of reasons around this. Firstly, Cloud Cache replicates profiles rather than distributing them and we were primarily looking to split the load across the file shares rather than provide redundancy. Second is that in the past Cloud Cache has been very buggy and it is only in more recent releases that it has improved, so I was loth to hang a production environment on it based on past experience. It is also unclear how Cloud Cache deals with a file share being at capacity – as far as I know it looks for availability only, although I am open to being educated if I am mistaken. Also, local cache potentially could have given us a 300GB storage increase for each server that was deployed, and that again would be a cost implication that would be unpalatable.

For our purposes this has worked very well, however there are a couple of points to be aware of.

Firstly, if someone expands their profile massively they could still potentially fill the file share. We have set profiles to a limit of

Parker has some good articles on this, will also touch on it in an upcoming post).

Secondly, finding a user's profile when they are one of a large number of file shares is a bit annoying and takes quite a while! To this end we are going to write an environment variable into the user profile with the name of the configured file share and display it using BGInfo to save us from this problem.

However, I have to say, that for use cases like ours, this has been a really good method to use. Hopefully some more of you out there may benefit from it, and huge round of applause due to Ryan Revord for developing this and sharing it – as I've said many times before, community rocks!

ALL POSTS ARTICLES CITRIX FSLOGIX HOW-TOS

CITRIX FSLOGIX POWERSHELL PROFILE CONTAINERS

« QuickPost – Improve Windows 10 logon times with FastFirstSignIn

uberAgent explored, part #1 – getting up and running »

24 comments

Reinhard Travnicek says:

April 11, 2020 at 7:56 am

Why did you not use the group-sid based feature built into the product? No need for your startup script then!

Best

Reinhard

Reply

James Rankin says:

April 11, 2020 at 5:34 pm

Sorry, what feature are you referring to?

Reply

James Rankin says:

April 11, 2020 at 5:48 pm

Oh you mean this – https://docs.microsoft.com/en-us/fslogix/configure-per-user-per-group-ht – this wouldn't have helped in this situation, we would still have the overhead of managing the groups and where they map to and what happens when those shares reach capacity and having to move AD groups if we want to move the users. Easier simply to send each user to the least-loaded file share I think, in this case we're just getting back towards splicing them like we have done before in UPM which just becomes a headache.

Reply

James Rankin says:

April 11, 2020 at 5:49 pm

We have upwards of 500 on some file shares and not seeing any issues (nothing reported, anyway)

James Rankin says:

April 11, 2020 at 5:50 pm

In fact now that you've said that we have 3231 on one file share. Whoosh!

Reply

Nishith Gupta says:

May 3, 2020 at 10:54 am

Hi James,

With VHDLocations, there is to automatic replication of VHDs from primary location to all other secondary locations, correct? And even if we manage to replicate the VHDs, there is no seamless failover if the primary location is unavailable.

CloudCache provides both.

I have an upcoming project, almost finalized where I have pitched CloudCache to fulfilling all the requirements. So far I am positive but I am trying to find use cases where CloudCache was dropped out as a probable solution.

Thank you.

Reply

Using VHDLocations provides no replication, correct. However given that we would require another 70TB+ for replication, we have elected not to provide any replication. As for seamless failover – have you done it with CloudCache? Newest version may be better but I would advise testing.

However the main thing for me is – do you need absolutely seamless profile failover in a DR situation? If all the user data can be synced back, how much inconvenience is it to lose a profile (and map this against the storage requirements)? When you ask a business if they want to pay for hundreds of terabytes of extra storage so Brian in Accounts doesn't have to reset his Outlook views and colour settings after an outage, are they going to say yes? For me, OneDrive with KFM is rapidly becoming a way to achieve this without the storage uplift. Seamless profile failover is probably not worth the effort (YMMV)

Reply

Junaid Yaseen says:

May 18, 2020 at 12:36 am

Thanks James,

Aaahh, had been looking for something similar for my upcomming solution for FSLogix. Had something similar in ming. But this script will make an easy way for me. ☺️

However, I also am looking at option to get rid of OneDrive data getting dumped into user profile. This unnecessary will fill in the profile storage when the data is out there in one drive as well. I want to make use of File On demand to give user better experience but don't want to retain data locally in the end just because it will be available on cloud as well. Why let my storage grow...

Regards,

Reply

Fredrik Endresen says:

May 21, 2020 at 7:51 pm

Hi,

With Storage Sense enabled and correctly configured, you can avoid the profile filled up with local copy of OneDrive data.

We uses this in our Win 2019 RDS / FSLogix environment. You should also use one of the tools to shrink the .VHDX files.

Reply

Junaid Yaseen says:

July 3, 2020 at 3:34 pm

Thanks Fredrik,

We had carved out a totally different solution to address our concern. using logoff scripts etc etc, which technically worked well – but had no blessings from Microsoft to support that. Understood from ther point of view. So yes Storage Sense was

a. Storage Sense will not be helpful in case a user uses "Always keep on this device" option for data downloaded from One Drive.

b. To control above, we looked out for options to grey out or remove this option from the right click itself. (Identified some registry settings that can do it, but that actually removes all options from One Drive including above – but again that will be on our own risk to use).

c. Other way that was suggested was to use attrib commands to un-pin the cached files that Storage Sense can't help with, but again, this is not configurable anywhere in One Drive Client. Rather had to use custom script at logoff or so.

d. Even enabling Storage Sense for a computer, a user will still need to manually set the caching policy to least available 1 Day from default value of "None". Again, we have to get help using a script to predefined this for a user as the entries/key is named with user's SID. By the way we have Win10 1809 and Server 2019 1809 in prod and have to use GPP to define Storage Sense settings.

e. The least time duration that we can set for a Cloud storage dehydration threshold in Storage Sense is 1 day or 24 hrs. Assume a user dumping Gigs of data in One Drive local cache folder before leaving for his/her vacation. System will need to wait till user login back satisfying 24 hr duration for One Drive cache to get cleared up.

We are primarily looking to make use of One Drive with roaming profiles, may it be a Non-Persistent Desktop or Terminal Server shared desktops. Over all our aim is to automate and have full control on the data that a user potentially can dump into the profile cache. Having such capability withing One Drive client

is that disk should not be in use while shrinking. But, at least good thing that Microsoft is considering to add a feature to shrink the VDI's on attach or detach.

https://windowsvirtualdesktop.uservoice.com/forums/921289-fslogix/suggestions/38440933-vhdx-auto-shrink-on-attach-or-detach

Regards,
Junaid Yaseen

Reply

Pingback: Group Policy Computer Settings for VDAs – Carl Stalhood
Pingback: Detailed Change Log – Carl Stalhood

Charles A. Windom Sr. says:
January 22, 2021 at 3:09 am

Does the powershell script go on the virtual desktop template (Using Horizon View) as a scheduled task

Reply

James Rankin says:
January 22, 2021 at 8:30 am

You can put the script wherever you want. We run it from a

Andrew Simpson says:
April 26, 2021 at 2:52 pm

Hi James,

I'm just performing some testing using this script before rolling it out. I ran the script on one workstation and it worked and created the profile on the share with the most resource. Unfortunately it looks like its not detecting if a user currently has a profile.

VHD location is currently set through GPO and creates the users profile inside of folder called firstinital.surname.

When using the script i've noticed that it creates it with firstinitial.surname_SID. Could this possibly be why its not detecting that it currently has a profile and do you know how we can resolve this issue?

Thanks,
Andrew

Reply

James Rankin says:
April 26, 2021 at 2:56 pm

What format is the "current" profile in?

Reply

Andrew Simpson says:

April 26, 2021 at 3:45 pm

current profile is firstinitial.surname.

With script it is firstinitial.surname_SID

Reply

James Rankin says:

April 26, 2021 at 3:53 pm

Have you got a custom SID pattern set via the FSLogix Registry values or GPOs?

Reply

MR ANDREW D SIMPSON says:

April 26, 2021 at 3:59 pm

We have swap directory name components set so this causes new containing directories to be named with the username first followed by the SID.
Is there a way to manipulate the script to do it in this way?

Thanks,
Andrew

Reply

James Rankin says:

April 26, 2021 at 5:10 pm

agent is still creating the containers, it's merely iterating through the list to find one to create it on.

I'd start by removing all customized FSLogix Registry entries and try again, especially any ObjectSpecific ones, if you have those configured. It sounds like something is either set incorrectly or being ignored.

Reply

Stevens Demorcy says:

May 21, 2021 at 10:21 pm

I have a question that I think this script might address for me. A former engineer built the FXlogix container on a drive that is no filling up and we can no longer grow. I have about 300 users, so I am looking for a way to move the 3TB of profile to another location. GPO is handling the profile pointing to the current file server.

Do you know of a way to seamlessley migrate the profile data or atleast maybe introduce another location that data is copied too?

Reply

James Rankin says:

May 23, 2021 at 11:12 am

Santos Martinez says:

November 19, 2021 at 4:32 pm

Hi James,

I have a question regarding the script. The user logins, how does the machine know to reference the script?

Lets say we implement the script, will it create new accounts for those already existing accounts? Sorry I am very new to these things so apologies in advance for any confusion.

Many thanks

Reply

James Rankin says:

November 19, 2021 at 4:36 pm

The script is a Startup Script, it does not run at logon. You can also run it as a Scheduled Task if you want.

When the script runs, it adjusts the Registry value so that a new user will get their profile created on the first share in the list. If a user logs in with an existing profile, FSLogix simply iterates through the listed shares in the Registry value until it finds the profile. So in answer to your question, no, a user with an existing profile container will always use that. A new one will only be created if it does not exist on ANY of the configured file shares.

Reply

Leave a Reply

Comment *

Name *

Email *

Website

Post Comment

WordPress Theme: Wellington by ThemeZee.