Discovering Microsoft's Unified Write Filter(UWF) on 10ZiG Windows 11 IoT LTSC 2024 Thin Clients Version 2.0

Document and Version Control			
Version	Created by	Date	Authorized & checked by
1.0	Jason Hudson	20/11/2025	K. Greenway
2.0	Jason Hudson	02/07/2026	K. Greenway

1.
2.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

1. About This Document

In this document we'll be discovering what you need to know about Microsoft's Unified Write Filter, or UWF, and talking about its benefits and also its limitations of use.

We'll be looking at overlays, what they are, where they reside and outline the pros and cons of using them, and also look at what happens when they become exhausted and how you might be able to mitigate any potential operational issues.

The guide covers a lot of ground, with regards to setup, guidance and monitoring of your devices when using the UWF. However, if you're familiar with the concepts and want to see what the best practice is for running Microsoft's UWF with Windows 11 IoT Thin Clients, then head to the "Advice, Guidance and Recommendations – Best Practice" section at the end of this document, where we include extra content for configuring specific build configurations and bespoke setups.

Note: While this guide is tailored to **Windows 11 IoT Enterprise LTSC 2024**, all of the content related to UWF, including the "**10ZiG UWF Wizard**"(Section 12 on page 41), also applies to **Windows 10 IoT Enterprise LTSC 2021**. In other words, this guide covers UWF for both of these Windows operating system platforms.

1.1 Important Information

Throughout this document, references will be made to the fact that currently, any exclusions made to the UWF overlay will write permanent content to those locations. However, they will also write to the overlay during machine uptime, thus consuming overlay space and reducing the uptime of the device. Even though we reported this as a potential flaw to Microsoft, they explained that this is how the UWF has been designed and writes "All" content to the overlay and then writes this permanently to disk if it's part of an exclusion.

In this document we also refer to Powershell commands, that enable you to list the contents of the overlay space being consumed. However, once you've identified overlay content that you wish to exclude from the UWF process and then physically created exclusions for those locations, any subsequent exports of the overlay content will not list those files or folders. So, from then on, or until you remove the exclusions, you won't have visibility of those excluded files and or folders.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

1.2 Recent Advancements in and Benefits of SSD Technology

Modern industrial SSD technology as featured in 10ZiG Thin Clients with 128GB storage or higher has improved significantly compared with earlier Disk-On-Module (DOM) storage, offering **higher endurance**, **improved wear levelling**, **better garbage collection**, **DRAM caching**, and substantially **better sustained read/write performance**.

These advancements make the use of a UWF Disk Overlay the recommended option, as modern SSDs can handle overlay redirection and temporary write workloads with minimal performance impact and negligible endurance risk. **This also reduces the need to install additional system RAM purely to support a RAM overlay, which is particularly beneficial as memory costs continue to rise**.

NOTE: If you need advice regarding the UWF, Disk Overlays and which SSD you're using, please consult your 10ZiG Tech Support Team, to make sure you're making the right choice for your thin clients.

1.3 Recommended UWF Configuration(Summary)

The section below provides a high-level summary of the key findings and recommendations from this guide. Readers looking for the main conclusions and recommended Unified Write Filter (UWF) configuration can use this section as a concise overview, while the remainder of the document provides the detailed configuration and technical "how to" methods in setting up your UWF environment.

Recommended Configuration

- UWF Enabled
- Disk Overlay on C:
- Overlay Size: 8GB minimum (larger if workload requires)
- Page File: System Managed on an unprotected volume (recommended D:)
- Storage: 128GB industrial-grade SSD
- Memory: 8GB RAM minimum

This configuration provides the best balance of system stability, overlay capacity, memory efficiency, storage endurance, and long-term maintainability. Testing showed that disk overlays are generally better suited to modern thin client workloads than RAM overlays, particularly on devices with 8GB of installed memory.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

2 What is the purpose of the UWF?

You may be aware that the UWF is part of Windows 10 and 11 versions of the OS and is designed to provide a clean experience for thin clients and workspaces that have frequent guests, like schools, libraries, or hotel lobby computers.

Guests can work inside a session, change settings, and even install software. However, after the device reboots, unless specifically configured to do so, the next guest receives a clean experience.

It increases security and reliability for kiosks, IoT-embedded devices, or other devices where new apps are not expected to be frequently added.

3 Overlays

The overlay is where the UWF stores its data and can either be in RAM or on the physical disk on the Windows 11 IoT Enterprise LTSC 2024 thin client.

When setting up any overlay, you need to first consider what your end goal is and then plan your decisions and strategies around that goal.

3.2 Overlay Exhaustion

If the size of the overlay is close to or equal to the maximum overlay size, any write attempts may fail, returning an error indicating that there is not enough space to complete the operation. If the overlay on your device reaches this state, your device may become unresponsive and sluggish, and you may need to restart your device.

When Windows shuts down, it attempts to write several files to the disk. If the overlay is full, these write attempts fail, causing Windows to attempt to rewrite the files repeatedly until the UWF can determine that the device is trying to shut down and resolve the issue.

Attempting to shut down by using normal methods when the overlay is full or near to full can result in the device taking a long time, in some cases up to an hour or longer, to shut down.

3.3 RAM Overlay

If we choose to store the virtual overlay in RAM, then this is cleared during a reboot and has the benefit of reducing writes to the physical storage media, which has historically helped minimize wear on write-sensitive devices such as solid-state drives. However, with modern industrial SSDs offering significantly improved endurance, wear levelling, garbage collection,

and sustained write performance, the historical disadvantages of using a Disk Overlay are now greatly reduced, making Disk Overlay a much more viable option than in the past. There are still exceptions to this rule, as some data may need to be written directly to the physical drive, such as excluded files or folders discussed previously. See the section Recent Advancements in and Benefits of SSD Technology.

Considerations for using a RAM Overlay

It's worth noting that if you decide on a RAM overlay for your temporary writes, you will also need to budget for some extra memory for your thin clients.

If the OS requires 4 gigabytes of RAM for example, and your device has 8 gigabytes installed, then set the maximum size of the overlay to 4 gigabytes or less. The more memory you can afford to equip your devices with, the better.

Content might still write to disk

If your intention is to prevent disk writes altogether, thus further extending the life of the drives, then this isn't entirely possible, as certain applications, such as Windows Updates and Windows Defender, for example will still need to save content to the disk at some point in time, for functionality and security update purposes.

Pros and Cons of using a RAM overlay

Pros	Cons
Can extend the life of physical disks. This depends on use case of thin client.	Content will still need to write to disk from time to time, so not completely write-free.
	Any exclusions that write to disk, will also write to the overlay, increasing the overlay consumption, causing exhaustion to occur sooner and reduced time between reboots.
	Steals from RAM, so hungry applications may struggle to perform well.
	Cost of extra RAM from a financial position.
	Overlay exhaustion can occur more frequently during use and affect operability, i.e., more frequent reboots are required.
	Overlay exhaustion will continuously write to the disk before finally shutting down.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

	This causes unnecessary writes to disk media and defeats the object of trying to reduce wear and tear.

Initial observations when testing a RAM Overlay

During testing, it was discovered that you can give your RAM overlay more space than is physically available to your thin clients. This could prove to be particularly hazardous, especially for the OS and “Live” applications that want to use that memory also.

3.4 Disk Overlay

If we choose to store the virtual overlay on Disk, then just like the RAM overlay, this is cleared during a reboot and its contents discarded. Obviously, using disk for your overlay, will give you a lot more space to work with and give your thin clients all available RAM in which to run the OS and its applications, depending on the amount of free disk space to begin with. This way, your sessions may last a lot longer between required reboots and reduce the instances of memory exhaustion and crashing your devices as a result.

Considerations for using Disk Overlay

Historically, if you chose to use a disk overlay, you had to ensure the thin client had sufficient storage, as any space allocated to the overlay is reserved immediately.

On older thin clients fitted with a 32GB DOM or less, storage capacity was often a significant constraint. If Windows and installed applications consumed 20GB of disk space, only 12GB would remain available. Allocating an 8GB disk overlay in this scenario would leave just 4GB of free storage, providing limited capacity for additional applications, temporary files, patches, or Windows updates.

This storage limitation was one of the main reasons disk overlays required careful planning on legacy devices. Modern 10ZiG thin clients, however, now ship with larger storage capacity, typically a 128GB SSD. This provides greater disk space, allowing larger overlay sizes while still leaving adequate capacity for operating system growth, application updates, pagefile and maintenance.

Based on our performance and endurance testing, such SSDs are well suited to disk overlay deployments, offering improved capacity and higher write endurance. Provided the overlay is sized and monitored as part of normal operational management, disk overlays are now a practical and recommended option for many modern 10ZiG thin client deployments.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

Pros and Cons of using a Disk Overlay

Pros	Cons
Memory is available for applications and the Operating System solely.	Disk wear and tear is increased on solid state drives and more of a concern when overlay exhaustion is encountered.
Depending on the size of disk and space allocated to the overlay, time between reboots of thin clients can be extended.	Disk space is reserved immediately, so if an emergency occurs and you need more space, this will require maintenance to free up storage.
	Disk is written to(overlay) even if you plan to discard or write its content.
	New application installs will consume free disk space unless system maintenance is carried out.
	Some exclusions may still contribute to overlay activity depending on write behavior and configuration, potentially increasing the overlay consumption.
	Excluded writes may increase write activity in certain scenarios, but modern SSD endurance and UWF behaviour mean the impact is typically less significant than previously assumed.

Note: Many of the disk-related concerns listed above were historically more significant on older thin clients equipped with small-capacity DOM(for example 4GB–32GB devices). Modern 10ZiG thin clients ship with 128GB SSDs, which provide substantially greater storage capacity, higher endurance, improved wear levelling, and better sustained write performance. As a result, concerns around overlay-related write wear, disk exhaustion, and storage longevity are considerably reduced in modern deployments, provided overlay sizing, exclusion configuration, and routine monitoring are implemented correctly. Additionally, testing has shown that excluded writes do **not** necessarily result in a simple “double write” scenario in all cases, meaning the practical impact on SSD lifespan is often lower than previously assumed.

4 What are UWF Exclusions?

Exclusions are either file/folder locations or registry entries on the disk and allow you to exclude them from the UWF's filtering process. However, data being written to any UWF defined exclusions, will still be written to the overlay during uptime, thus increasing its size. This is especially important to note when you're operating a UWF on a device with limited resources with a RAM overlay, as this will be limited to the physical RAM in your device and may mean overlay exhaustion happens sooner than with a DISK overlay. However, if you are running a device with a limited amount of storage and use this as your overlay, then similar principles will apply. We'll discuss this topic in more detail further on in the guide.

The only benefit of using exclusions, is that your content is permanently committed to disk during the write process.

NOTICE: During our testing, we discovered that if we added and then subsequently deleted any content that had been written to an excluded area of disk, that the overlay didn't decrease by the size of that file. This means that if, during your current booted and logged on session, you write to and then even delete excluded content, the overlay will grow and increase during the write and remain that growth size, even if you then delete the file.

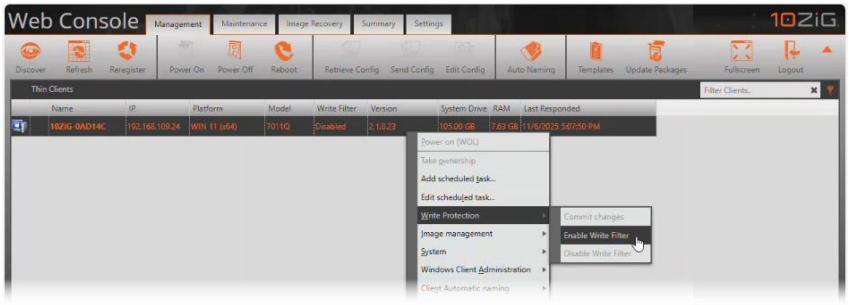
10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

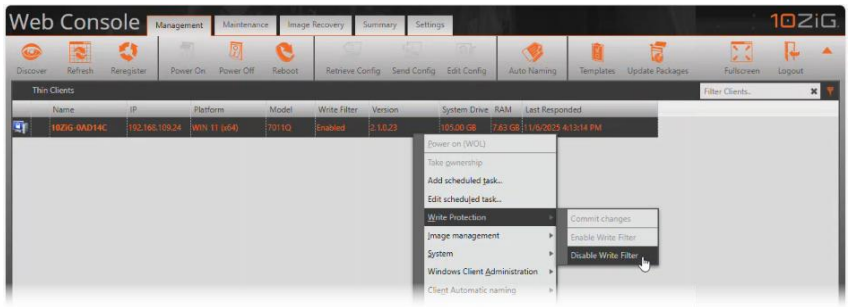
5 Configuring the UWF for first time use and useful commands

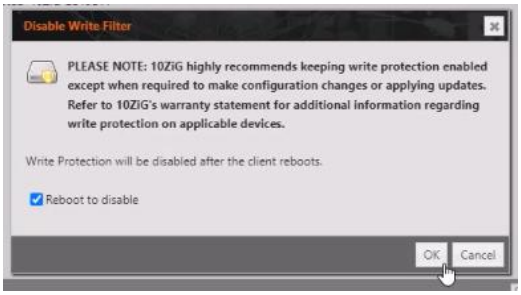
This section will explain how to enable the UWF for the first time and mention some of the most common commands that you'll use to fine tune it for your own thin clients and your specific infrastructure environment.

There are several ways in which to manage the UWF, you can use Powershell, Windows CMD commands 10ZiG's Quick Start Guide and 10ZiG's UWF Wizard. Most of the examples in this guide will use the command line and Powershell to query and configure the UWF but it's worth mentioning that these command line and Powershell scripts can be executed from within script and batch files for automation purposes. Further on in this guide, we'll give you a brief tour of 10ZiG's UWF Wizard and show you how to setup the UWF inside there, giving you an easier way to manage an all-in-one GUI environment.

The 10ZiG Manager also allows you to manage the UWF filter and can be enabled or disabled from here. Note, that if you do this, then you'll be prompted before issuing the command and have the option to cancel. Here are some screens showing enabling and disabling of the UWF on a remote client using the 10ZiG Manager Web Console.

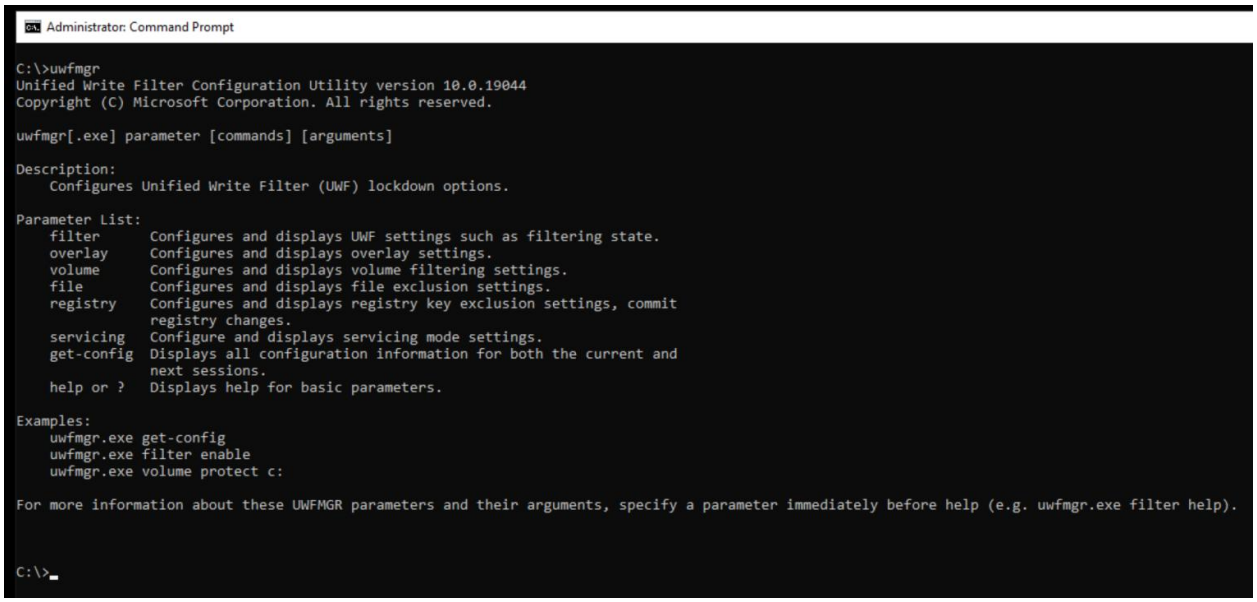






5.1 UWFMGR command options

Inside a DOS command window, if you type in uwfmgr and press ENTER, you'll see some helpful options to get you started and working with the UWF itself.



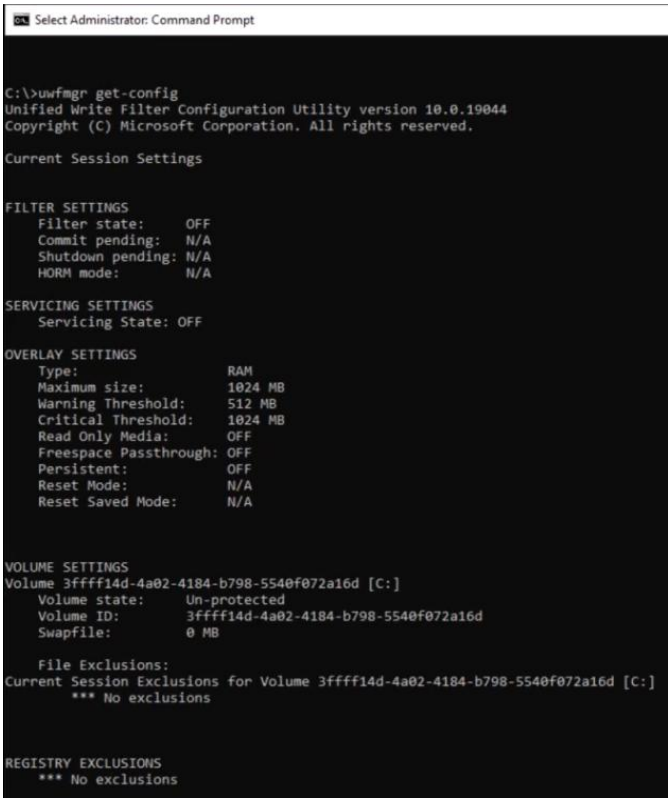
5.2 Displaying the Configuration of the UWF

This command will show you the "current" and "next" session settings and is a way of being able to confirm any changes that you are about to apply following the next reboot.

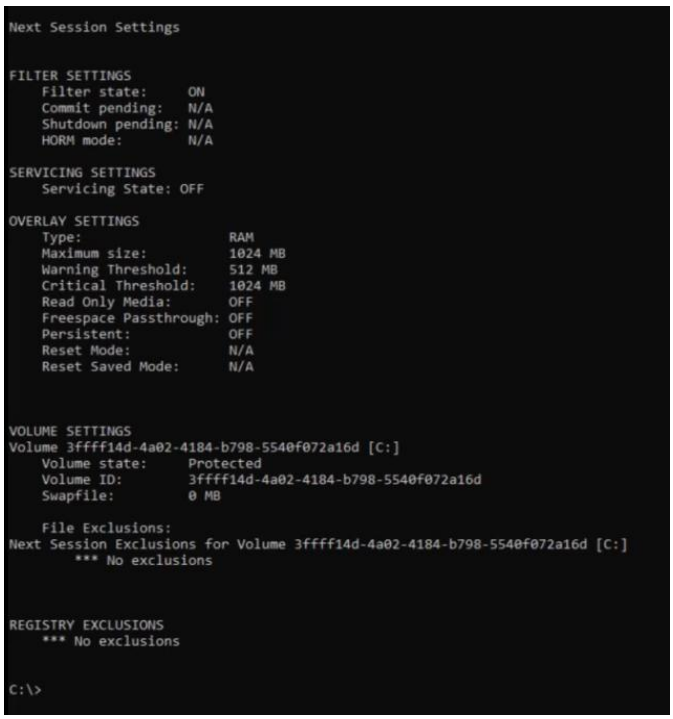
Type in uwfmgr get-config and press ENTER and you'll see 2 sets of data, current session settings and next session settings. Current session settings are ones that are in force now, during this Windows login. If you make any changes to the UWF settings in this current session, then running uwfmgr get-config again, will reflect what they will look like after a reboot.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

Current Session Settings



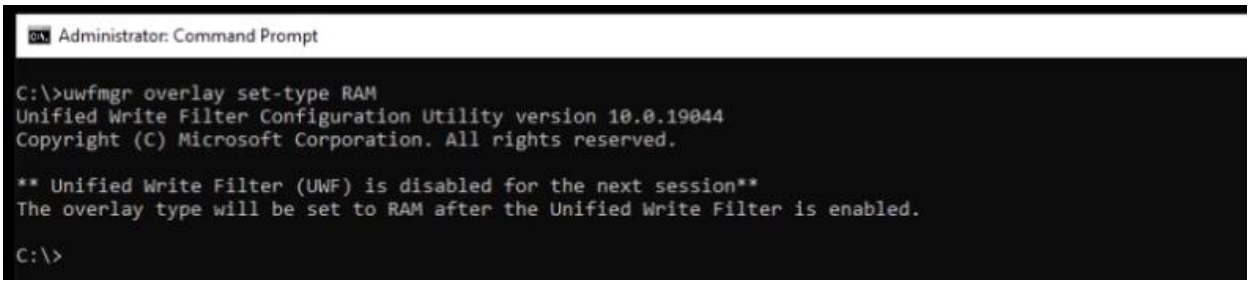
Next Session Settings



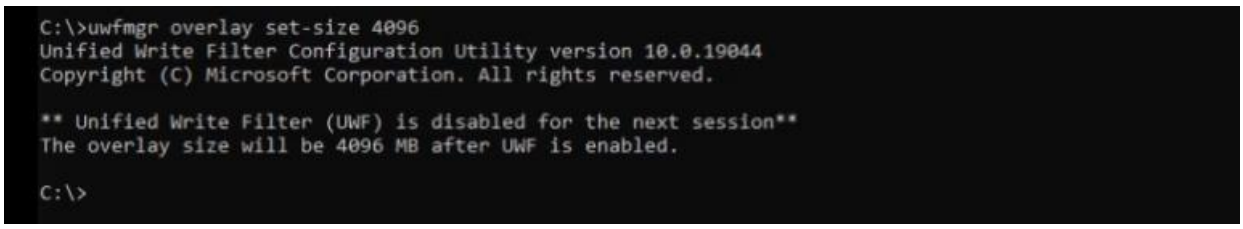
5.3 Using UWFMGR to setup the Overlay

We're going to setup the overlay in RAM to start with, set its size and then finally protect drive "C", enable the write filter and reboot for the changes to take effect.

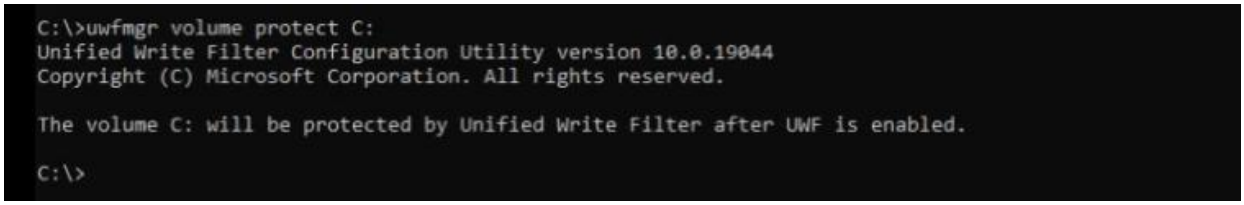
Inside the command window, type in uwfmgr overlay set-type RAM and press ENTER. Notice that the changes we're making here, will be set when the UWF is enabled. We'll set the filter state to "enable" a few steps further on.



Next, type in uwfmgr overlay set-size 4096 and press ENTER. This will set the size of the RAM overlay to be 4096MB or 4GB. This allocation of the overlay in RAM will grow during uptime, according to requirements of the UWF. It's worth mentioning that disk overlay on the other hand, is allocated and reserved the moment you set it. We'll explain a bit more about disk overlays later.



We need to protect the drive now, so that the UWF knows which specific disk we're going to be filtering reads and writes to. If you only have one disk, then this is obviously the one we want to protect. Type in uwfmgr volume protect C: and press ENTER.



The next thing to do before we can utilize the UWF, is to enable it so that when it reboots and restarts, the changes we have made here will take effect.
Type in uwfmgr filter enable and press ENTER.

```
C:\>uwfmgr filter enable
Unified Write Filter Configuration Utility version 10.0.19044
Copyright (C) Microsoft Corporation. All rights reserved.

Unified Write Filter will be enabled after system restart.

C:\>
```

Now, before we shut down and restart the thin client, we're going to create a folder on drive "C" called TEMP and then add it to the file exclusion list.

Type in MD C:\TEMP and press ENTER.

```
C:\>md C:\TEMP

C:\>
```

Now we've created the folder, we type in uwfmgr file add-exclusion C:\TEMP and press ENTER.

```
C:\>uwfmgr file add-exclusion C:\TEMP
Unified Write Filter Configuration Utility version 10.0.19044
Copyright (C) Microsoft Corporation. All rights reserved.

The file/folder "C:\TEMP" will be excluded from protection after system restart.

C:\>
```

Following the next reboot, any writes or deletes carried out inside this folder will behave as per normal operation on the drive as it bypasses the filter, it will however still write content to the overlay as we mentioned earlier.

We'll restart the thin client now and see what the UWF looks like once the protection is in place and the filter is enabled.

Type in shutdown /r /t 3 and press ENTER. This basically shuts down with a /r for restart and /t 3 to timeout in 3 seconds.

```
C:\>shutdown /r /t 3
```

The shutdown message will be displayed, and the device will restart.

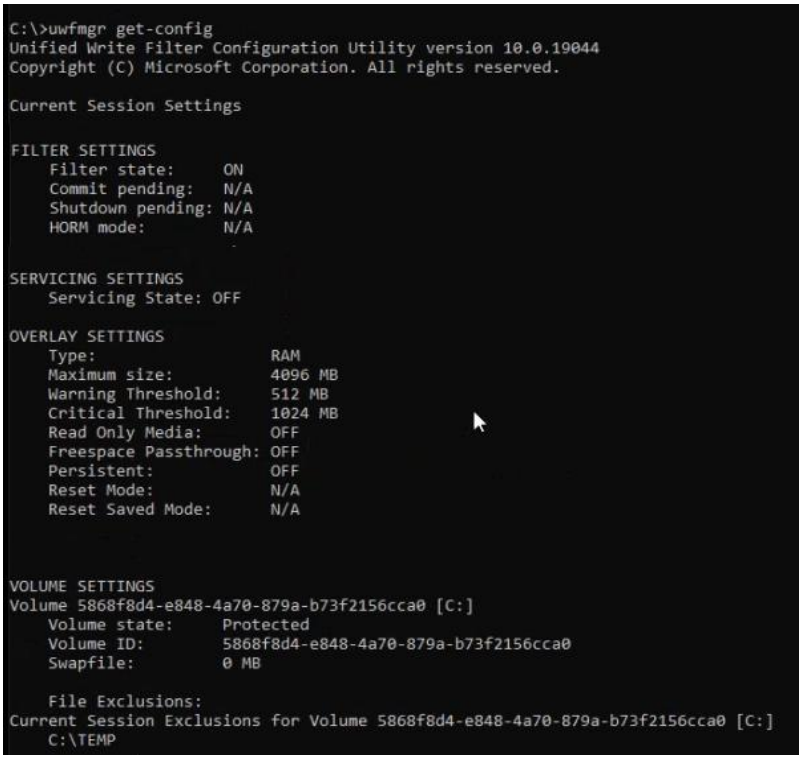
5.4 Checking the UWF Status following filter enable

Now that the thin client has rebooted, if we run the uwfmgr get-config command again inside the CMD window, we can see that the changes we made earlier are now in effect.

Under “Filter Settings”, we can see the filter is on and in the “Overlay Settings” section, we now have an operational Overlay in RAM with a “Maximum size” of 4096MB.

Just beneath, you’ll notice a “Warning” and “Critical” Threshold setting. These limits can be modified to suit, and their purpose is to write an event record to the Windows Event log on reaching these levels on the overlay. We’ll give you a more detailed breakdown of these events in the “Monitoring” section further on in this guide.

In “Volume settings” below, we can see that the “C:” is protected and “C:\TEMP” is excluded from the filtering process as we configured earlier.

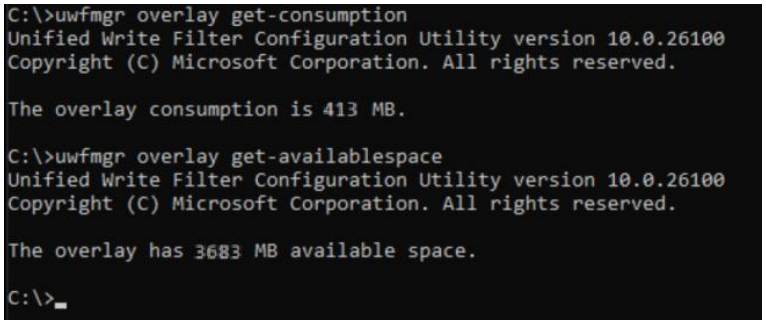


6 Overlay Management, Monitoring and Exhaustion

6.1 UWF Overlay Commands

There are 2 useful uwfmgr overlay commands that you can use during testing of your overlay usage, these are “uwfmgr overlay get-consumption” and “uwfmgr overlay get-availablespace”.

They do exactly what they suggest, get-consumption will tell you how much of the overlay has been used and get-availablespace lets you know how much is left. Here’s an example below :-



Remember that we set the overlay size to be 4096MB, well if you add up the consumption of 413MB and available space of 3683, then you reach approximately that overlay size.

6.2 Exporting the Overlay file content

We’re using some “Powershell” commands inside a .ps1 script file that will output the content to a .csv file, that we can then open as a text doc or spreadsheet for further manipulation. The “Powershell” commands are shown below and need to be run in succession inside a “Powershell” console or called as a script file from a CMD window.

```
$OverlayVolume=$args[0]
$outputfile=$args[1]
$wmiobject = get-wmiobject -Namespace "root\standardcimv2\embedded" -Class UWF_Overlay
$files = $wmiobject.GetOverlayFiles(“$OverlayVolume”)
$files.OverlayFiles | select-object -Property FileName,FileSize  | export-csv -Path $outputfile
```

We’ve placed these into a “Powershell” script called “GetOverlayFilesInUWF.ps1” and added in some additional code to allow you to pass the drive and the output file as parameters.

To increase the size of the overlay we downloaded a Windows Update, that was about 301MB in size, so we could then demonstrate output results with the WMI command.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

Here is an example, calling the script from a CMD window and passing the “C:” as the $OverlayVolume and “C:\TEMP\OverlayFilesContent-After-Downloaded-WindowsUpdate.csv” as the $outputfile filename.

Once this has completed, we can open the .csv file inside a compatible spreadsheet editor and see what has been logged.

On our 10ZiG Windows 11 IoT thin client, we’ve logged on as the user named “Administrator” and so we know that our recently downloaded Windows Update should be in the folder “C:\Users\Administrator\Downloads”, so let’s search for that location inside the .csv and see if it was logged correctly.

	A	B
1	#TYPE Selected.System.Management.ManagementBaseObject	
2	FileName	FileSize
1460	\Users\Administrator\Downloads\windows10.0-kb5007253-x86_a623409.msu::$DATA	315854848
1461	\Users\Administrator\Downloads\windows10.0-kb5007253-x86_a623409.msu::$ATTRIBUTE_LIST	4096
1462	\Users\Administrator\Downloads:$I30:$INDEX_ALLOCATION	4096

We can see an entry for the Windows Update file in the “Downloads” folder as expected and that the file size taken up inside the overlay is approx. 300MB. Because we’re using a RAM overlay, then this 300MB will have been taken from our physical RAM.

6.3 Demonstrating Overlay Exhaustion

You’re probably wondering why you’d ever want to exhaust the overlay on your Windows IoT thin clients, as it seems counter-intuitive why you’d want to crash out the devices. If you’re using a RAM overlay and not familiar with its behavior, then you’ll be surprised how quickly it can fill up and cause a reboot to occur.

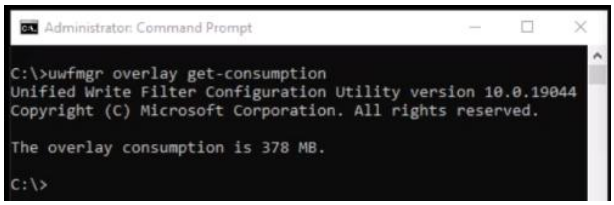
In our tests, we repeatedly downloaded some really large files from the Microsoft Update Catalog site and then recorded the results here for you. This type of thing could happen to your users if they have complete freedom to visit these sites and run manual Windows Updates by downloading the “KB” installers themselves for example.

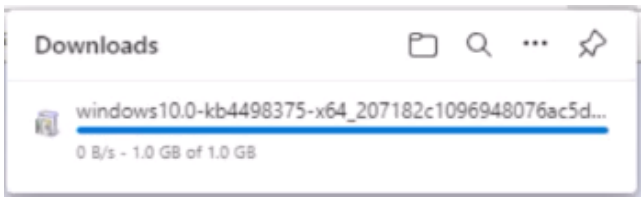
10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

Overlay Exhaustion Testing

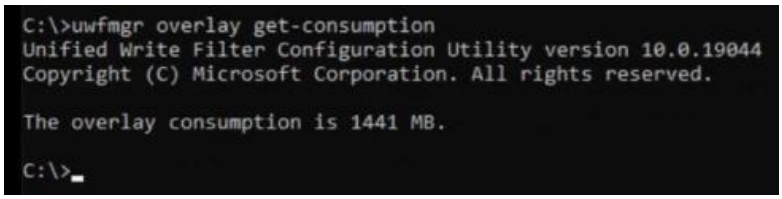
We started by downloading a 1GB Windows Update file named KB4498375 and tracked the overlay consumption.

At the beginning of the download, we checked the overlay consumption, and this was 378MB.

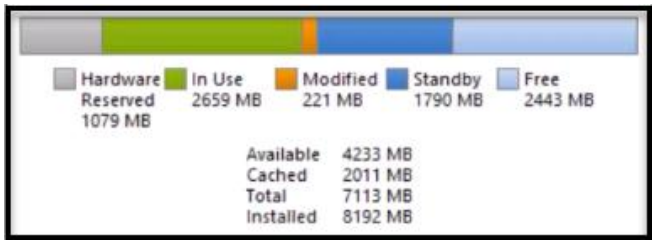




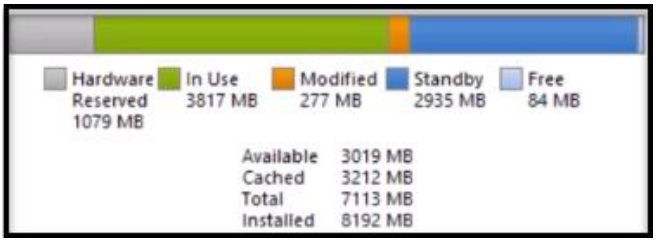
After the download had completed, we took another check on all stats, and they seemed to add up. We can see that the “In use” RAM has gone from 2659MB to 3817MB and if we run the uwfmgr overlay get-consumption command again, this has gone from 378MB to 1441MB.



Memory at Start of 1GB download.



Memory at End of 1GB download.

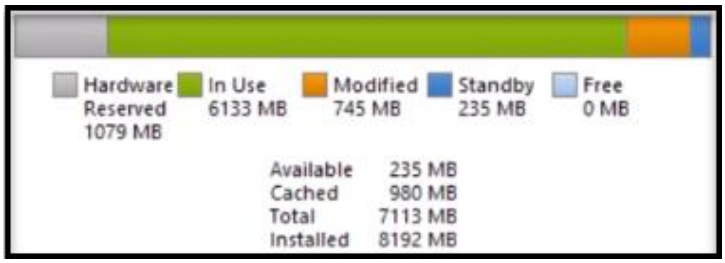


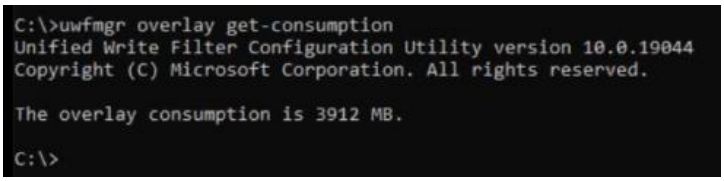
Pushing the Overlay to its limits

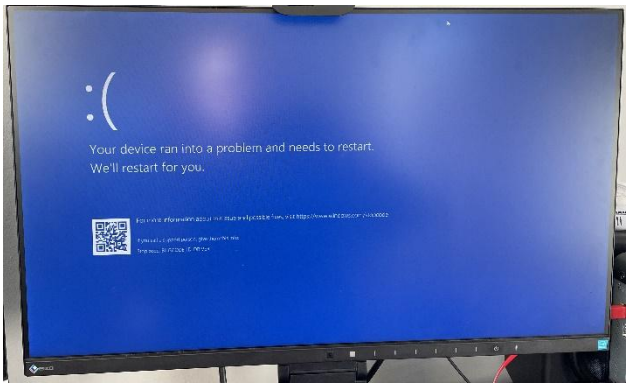
We continued downloading more “Windows Updates”, just to see if we could fill up the overlay to the point of exhaustion.

After approximately 45 minutes of uptime, Windows was using 6133MB of a possible 7168MB of RAM, it had 0MB of “Free” memory and a “Standby” of 235MB. The overlay had consumed 3912MB of a potential 4096MB and the thin client had no option but to shut down. You can see the “Blue Screen” taken by camera phone as the OS crashed and rebooted.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu







6.4 Freespace passthrough

Freespace passthrough is supposed to give your overlay, either in RAM or disk, additional dynamic physical drive space in which to continue writing overlay content if the overlay should fill up. This would give us the benefit of being able to keep the thin client session up for longer, without having to restart it.

REMEMBER : By using Freespace Passthrough, you're still going to be writing to that solid state drive and increase excessive wear and tear over time. If that is a major concern for your organization, then you really need to think long and hard before using Freespace Passthrough.

In testing we created 2 scenarios, in the first test, we created a 4096GB disk overlay and pushed the overlay to its limit by copying large files to the physical disk, thus attempting to fill the overlay within 5 minutes.

The second test was carried out exactly the same way, but with the overlay in RAM.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

To enable the FreeSpace Passthrough, you need to disable the filter, as with any changes, reboot, run the command uwfmgr overlay **set-passthrough on** and press ENTER. Then enable the filter and reboot the thin client.

Disk Overlay Freespace Passthrough Test

With the **disk** overlay, we had generated 4GB of extra space by copying these large files, and the overlay had only consumed 189MB, so the "Freespace Passthrough" was definitely doing its job. The physical disk was down to about 500MB total free space, and any other disk related demands were met with "not enough disk space" errors as you'd expect.

The only thing to be aware of here is that you can run out of physical space on your storage device and defeat the object of using passthrough altogether, resulting in the need to restart the device anyway, just to get it functioning again.

If you decide to implement a disk overlay for your write filter, then our recommendation is to **switch off** freespace passthrough altogether and set a definitive size for your disk overlay. This way, you can allocate the desired amount of space for your reserved disk overlay and know that's all you will ever use. That way, you can plan the layout of your disk architecture in a more structured and tactical manner.

RAM Overlay Freespace Passthrough Test

We set up the 4096MB overlay in RAM and carried out the same tests, copying large files to the physical disk, the overlay consumption only reached 250MB, with the large file content of 4GB, being stored in the freespace passthrough region again.

The benefit of using this method for writing content to the overlay is that your chance of having to restart the machine frequently is greatly reduced now, as physical RAM will not be frequently affected.

Freespace Passthrough test Conclusions

In summary, both of these methods using passthrough still created writes to the disk and seem to be focused mainly on writing to the disk primarily, without any intelligent decision making process involved.

One would have thought that using a 4GB RAM overlay might be balanced with that of disk and RAM, but the test showed that the RAM overlay was barely touched and rarely exceeded 200MB.

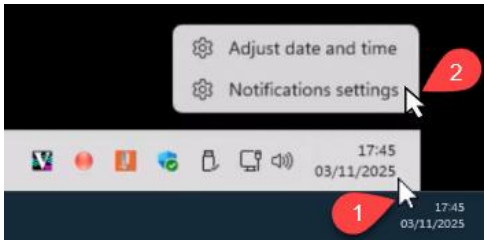
10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

You don’t seem to have the ability either to state how much of the “Freespace Passthrough” your thin clients can have access to. In our tests, we could push the unit’s physical storage to run out of space, until we saw “There is not enough space on the disk” errors. However, the device would still remain “alive” to give us a chance to resolve any space issues and save our work if we needed to, giving us time to restart the unit fresh if we decided to.

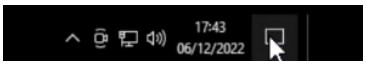
7 Notifications

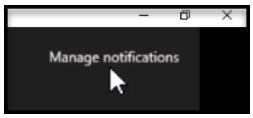
7.1 Notifications – System

Windows will write entries into the “System Event” log(see section 8) to notify you if your Overlay is reaching the “Warning” or “Critical” state and also send “Notifications” to the screen. To make sure that the UWF is setup for notifications, right-click on or near the date and time at the bottom right of the “Taskbar” and then “Notifications settings” from the drop down.

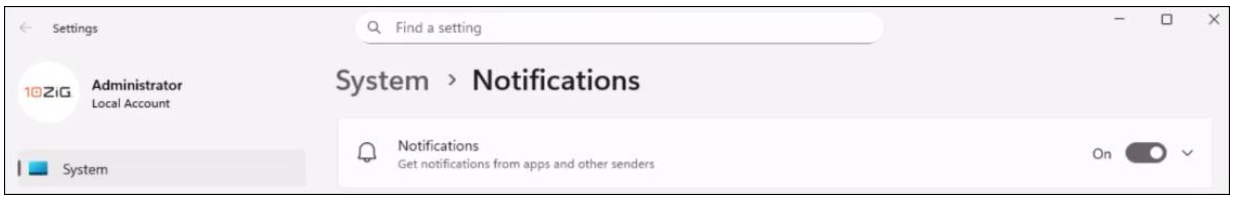


If you’re running **Windows 10 IoT**, then to make sure that the UWF is setup for notifications, click on the “Notification” icon in the bottom right of the “Taskbar” and then “Manage Notifications” at the top right corner.

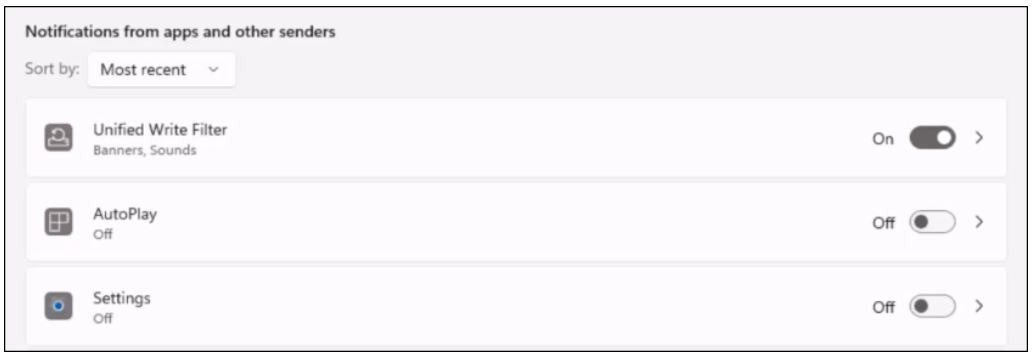




Once the “System > Notifications” screen is displayed, make sure that “Get notifications from apps and other senders” is switched “On”.



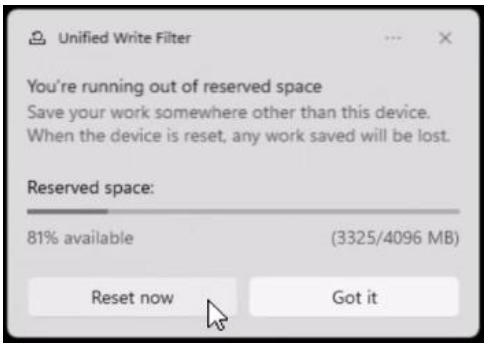
Further down the screen, you will see a list of senders of “Windows Notifications”, sorted by “Most Recent”.

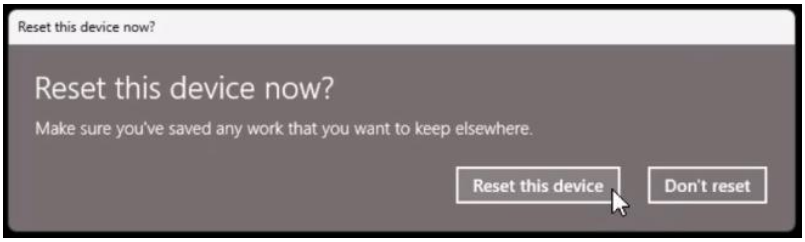


10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

NOTE: You will only see the “Unified Write Filter” sender appear in here once your Overlay has hit its first “System Event” log entry limit. For example, if the Overlay “Warning Threshold” or the “Critical Threshold” is breached, and you go back into the “Notifications and Actions” screen above, you will see the “Unified Write Filter” as a recent sender.

If Windows detects that the Overlay is running out of space, then a notification will be displayed on screen and sent to the notification area. Here is an example below, where you have the option to click “Got it”, where Windows will just continue as normal or “Reset Now”. If you click “Reset Now”, then you’re asked again if you want to “Reset this device”. If you then decide to click “Reset this device”, Windows will carry out a forced shutdown and clear the overlay content. IT WON’T PROMPT YOU TO SAVE ANY OPEN WORK. It just restarts, closing everything.





10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

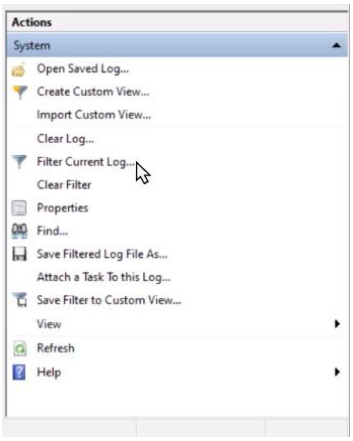
8 Additional Monitoring

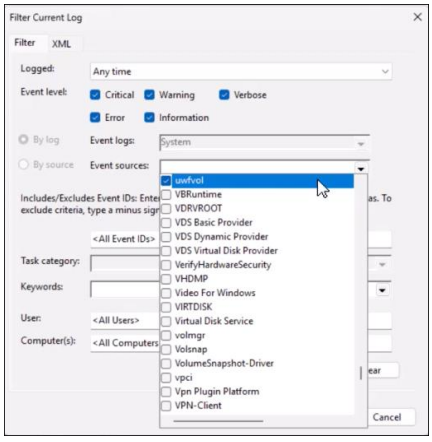
8.1 Windows Event Viewer GUI and Command Line

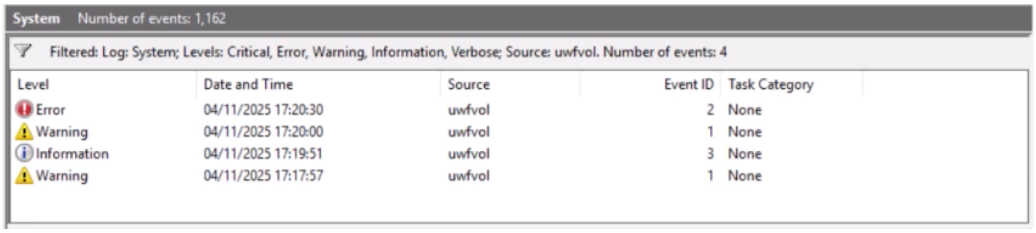
When monitoring UWF events, Windows Event Viewer logs 3 event types that are generated by the source “uwfvol”. You can filter on these types and even export the content as logs in several formats.

Launch the event viewer by typing in **eventvwr** and pressing enter in a CMD window. When it opens, click on “Windows Logs” and then “System” in the “Event Viewer(Local)” left menu.

Click “Filter Current Log…” in the right hand menu, then click in “Event sources:”, scroll down and tick “uwfvol” as the source, ticks all “Event Levels” and we will see any events generated by the UWF. There are 3 events that the UWF will generate, these are listed in the table below with references to the respective “UWF Overlay Threshold”.







Event ID	Event Level	UWF Overlay Threshold	Definition
1	Warning	Warning	The overlay has reached or exceeded the warning threshold set by uwfmgr command.
2	Error	Critical	The overlay has reached or exceeded the critical threshold set by uwfmgr command.
3	Information	Normal	The overlay has dropped back down below the warning threshold, back to normal level.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

9 Advice, guidance and recommendations – Best Practice

The information in this section is intended to give suggestions and recommendations on how to get more up-time from your Windows 11 IoT LTSC OS, when running a UWF. It is by no means a guaranteed solution to minimizing overlay usage, and any ideas mentioned here will need to be tried and tested in your own environment and see if any of them work for you. These suggestions are meant to try and alleviate the possible impact that the UWF overlay might pose and the fact that it gets written to regardless of whether you exclude content or not.

9.1 Overlay Exclusions, including Operating System locations, Operating System logs and Application logs

Exclusions have been mentioned throughout this guide and play a vital role in determining the performance, stability, and overall usability of your Windows 11 IoT LTSC thin clients.
In this section we'll list common OS files and folders and application-specific locations to either exclude or not to exclude as the operating system needs to be in control of them.

We will list all our recommended exclusions in more detail in the final section of the guide, where we'll include a setup of a typical device with UWF, Page File and device configurations.

9.2 Microsoft Recommended Exclusions

Microsoft recommend that you **DO NOT ADD THE FOLLOWING EXCLUSIONS**.

\Windows\System32\config\DEFAULT
\Windows\System32\config\SAM
\Windows\System32\config\SECURITY
\Windows\System32\config\SOFTWARE
\Windows\System32\config\SYSTEM
\Users\<User Name>\NTUSER.DAT
\Windows\BOOTSTAT.DAT
<System Drive>\EFI\Microsoft\Boot\BOOTSTAT.DAT
<System Drive>\Boot\BOOTSTAT.DAT

The volume root. For example, C:
The \Windows folder on the system volume.
The \Windows\System32 folder on the system volume.
The \Windows\System32\Drivers folder on the system volume.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

Adding an exclusion for any of these items is unsupported and may lead to unpredictable results. It's OK to exclude subdirectories and files under these locations, however.

For further information on this, search for “common write filter exclusions Microsoft” in your web browser or visit the following website

https://learn.microsoft.com/en-us/windows-hardware/customize/enterprise/uwfexclusions

9.3 Recommended Exclusions

We will provide a complete list of all recommended exclusions and non-exclusions in more detail in the final section of the guide, where we’ll include a setup of a typical device with all UWF and device configurations.

Just a recap on how to add file/folder exclusions and also how to add a registry key exclusion.

To add a file/folder exclusion for Windows Defender as listed above, use the command uwfmgr file add-exclusion C:\Program Files\Windows Defender and press ENTER.

To add a registry exclusion for Windows Defender, use the command uwfmgr registry add-exclusion C:\Program Files\Windows Defender and press ENTER.

If you want to see what your exclusions will look like when the machine reboots, then type in the commands uwfmgr file get-exclusions and press ENTER, or for the registry type in uwfmgr registry get-exclusions and press ENTER.

9.4 Operating System and Application Log Files

You can use the “Powershell” overlay export method we showed you earlier on, to find out what gets frequently written to your overlay, regarding logs and then consider whether it’s worth checking the application that writes those logs gives you the ability to trim or filter certain types of content, so you write less data and less frequently.

This is especially important, as we know that anything that creates “chatter”, so for example, verbose logging, will fill the overlay quicker than normal.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

Listed below are examples of some of the areas that you might want to think about when you're planning on including or excluding log content.

- Windows Event Logs – These don't get captured if you don't exclude them.
- Application specific logs of your VDI apps, for example Omnissa Horizon, Citrix Workspaces, Remote Desktop
- Any application that uses a local lightweight database, such as sqlite.
- Internet browsers that frequently cache content, cookies, and browser history for example.

What if you want to keep your logs!

If you're certain that you're going to want to run your thin client overlay in RAM, and you have a requirement to keep some logfile content, then you'll need to write your logs to storage.

So, you'll have to do 2 things, exclude the locations from UWF and also consider the level of logging, as we mentioned before. The more log content written, the less uptime your device may have, so just test logging with your devices and find the happy medium that balances log quality and device uptime.

Windows Log Locations

C:\Windows\System32\LogFiles
C:\Windows\System32\winevt\Logs

Common VDI Client log locations

Omnissa Horizon Client :
C:\Users\{logged in user}\AppData\Local\Omnissa\Horizon\logs
C:\ProgramData\Omnissa\Horizon\Logs

Citrix Workspaces App :
C:\Users\{logged in user}\AppData\Local\Citrix\{logs in subfolders}

Windows 365 App :
C:\Users\{logged in User}\AppData\Local\Packages\MicrosoftCorporationII.Windows365_8wekyb3d8bbwe\LocalCache\WebView2\EBWebView\Default\{logs in subfolders}

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

9.5 Windows Updates

Windows Updates are enabled and setup to run by default, as and when they are deemed necessary by the operating system. Microsoft schedules the release of security updates on "Patch Tuesday," the second Tuesday of each month at 10:00 AM PST. If for example, you're operating your devices with the UWF during this time, it could cause issues for the devices in several ways.

An update might download automatically and start the update process during normal business operating hours, consume all available overlay space and make the device unstable, causing it to reboot automatically or need a manual restart. With the UWF write filter enabled as in this case, once the device restarts, the Windows Update will start all over again as the overlay content was discarded on reboot, creating an "Update Loop", continually rebooting, and updating, but never installing anything.

If any of these Windows Updates were to install successfully, say for example because it was a small update that didn't push the overlay, the moment the device restarts as part of a natural process, the UWF write filter will discard that last Windows Update anyway.

9.6 Automatic Windows Updates using UWF Servicing Mode and 10ZiG Solutions

You can setup "**UWF Servicing Mode**" to run out of hours on a scheduled task and Microsoft UWF Servicing will carry out all necessary updates, rebooting and accepting any EULAs in the process, eventually bringing the device back to an updated and fully locked UWF state.

10ZiG have developed a process that will allow you to create this schedule task, so that you can run your update process out of normal business hours, when it suits your business operating hours for your devices. We will mention this in more detail in the final section of the guide, where we'll include a setup of a typical device with all UWF and device configurations.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

9.7 10ZiG WUService.exe

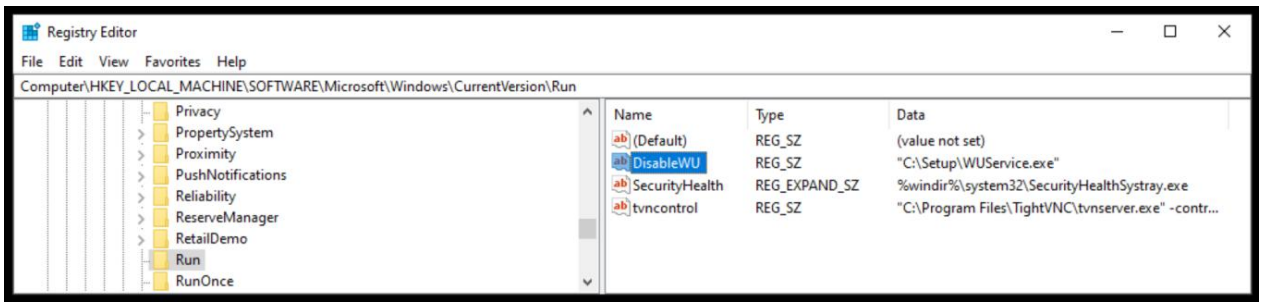
10ZiG have written an executable called WUService.exe, that is responsible for controlling 2 Windows Update services WuauServ and UsoSvc. When the device boots up, a registry entry is run that calls this app and this app either turns these 2 services on or off, based on the current status of the UWF.

If the UWF is enabled, then these 2 services are disabled on boot. This stops Windows Updates from running ad-hoc and potentially causing Overlay exhaustion.

If the UWF is disabled, then these 2 services are enabled on boot. This assumes that the device can be written to, and that Windows Updates might need to take place.

Registry "Run" Command

Shown below, is the registry "run" command that is required to call the WUService.exe on boot, to set the Windows Update services state. The key is in HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run



9.8 10ZiG UWFService.exe

This app that resides in C:\Setup folder is designed to be called when you want to run your Windows Updates controlled by the OS, in UWF Servicing mode. This app specifically does the following :

- Disables the UWF filter.
- Puts the device in UWF Servicing mode.
- Manipulates the 2 Windows Update services mentioned above.
- Carries out all Windows Updates needed.
- Enables the UWF filter again once all updates have finished.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

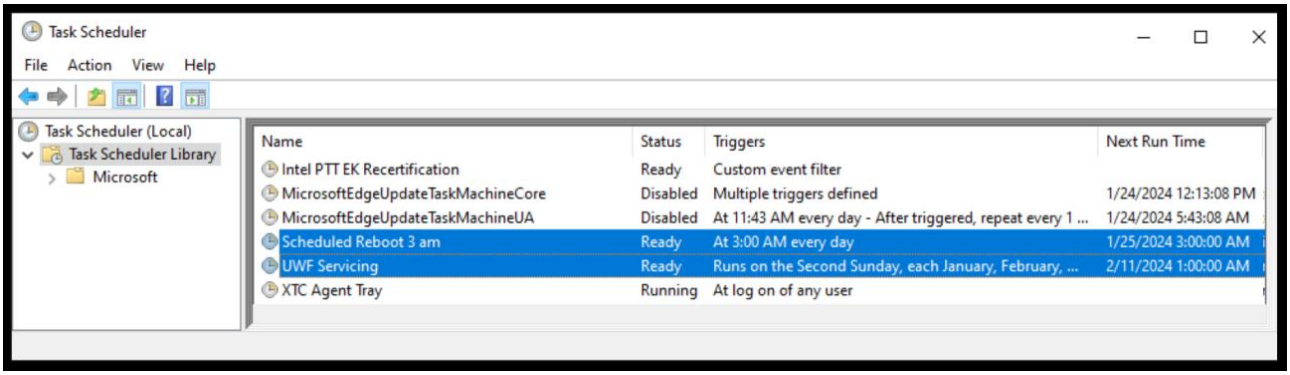
The idea behind this app is to run it on a scheduled basis that totally suit your business operational needs, so out of normal hours, where your users and devices aren't disrupted.

9.9 10ZiG Scheduled Tasks

10ZiG advise using 2 scheduled tasks that they have created. The first one called "Scheduled Reboot 3 am" in our example, will carry out a scheduled reboot each day, just to make sure the overlay is refreshed, ready for the next operational day.

The second one, called "UWF Servicing", is very important as it is intended to replace the "Patch Tuesday" we mentioned earlier. This one will call the 10ZiG UWF specific application we mentioned above called UWFService.exe.

Below, are the 2 scheduled tasks set up inside the "Windows Task Scheduler".



Note the "Next Run Time" for both tasks, the "Scheduled Reboot 3 am" runs every day and the "UWF Servicing" runs every second "Sunday" each month.

The dates and time mentioned in task examples are just guidelines and should be modified to suit your own business requirements.

NOTE: Talk to your local 10ZiG support team for further information on obtaining these applications.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

10 Standard 128GB Disk Layout with Dedicated Page File Partition and Optional UWF Implementation on 8GB RAM devices

Going forward, all 128GB or higher SSD-based Windows 11 thin clients will be deployed using a two-partition disk layout. This structure is now considered the standard configuration for devices expected to run modern workloads and, where required, Unified Write Filter (UWF).

Historically, 128GB SSD or higher devices were often deployed using a single OS partition containing both Windows and application data. While functional, this layout does not provide the same flexibility or resilience as a dedicated multi-partition design, particularly when page file placement and UWF overlay management need to be considered.

Partition Layout

The disk is configured using the following partition structure as an example for a 128GB SSD:

- **C: (100GB OS Partition)**
  This partition hosts the Windows 11 operating system, installed applications, and standard system data. When UWF is enabled, this should be configured as the protected volume.

- **D: (8GB Unprotected Partition)**
  This partition must be reserved exclusively for a **System Managed Page File** and should remain outside UWF protection.

Page File Partition

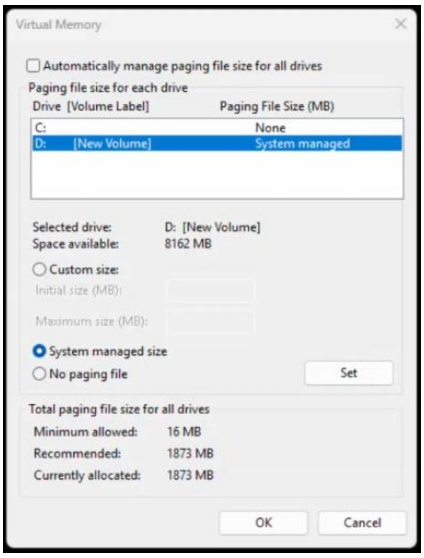
On all new devices a **System Managed Page File will be configured on D:**. This ensures Windows has sufficient virtual memory during periods of elevated memory pressure and improves overall system stability when running modern Windows 11 workloads such as virtual desktop sessions, browsers, collaboration tools, and background services.

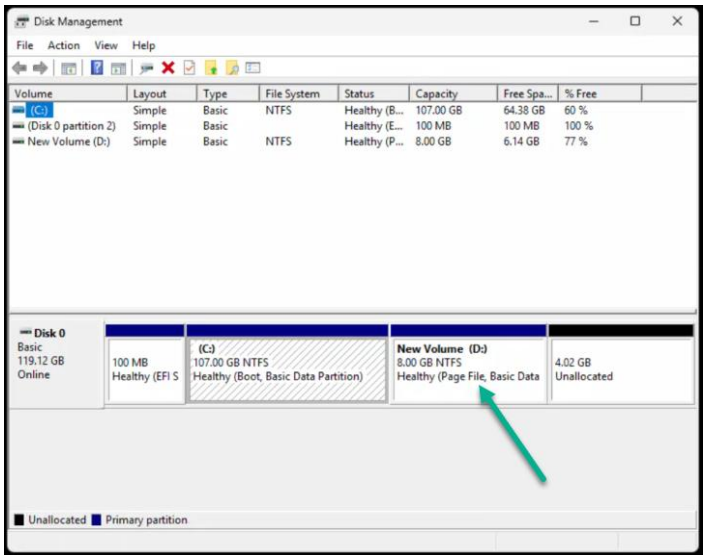
Hosting the page file on the unprotected D: partition ensures that paging activity does not consume UWF overlay capacity and prevents unnecessary write activity within the protected OS partition.

Key benefits include:

- Prevents page file writes from consuming UWF overlay space
- Reduces risk of overlay exhaustion
- Improves stability during memory-intensive workloads
- Reduces unnecessary writes to the protected OS volume

Example Disk and Page File Settings





When a system-managed page file is located on a UWF-protected volume such as C:, Windows may be unable to maintain the page file persistently across reboots because write activity is redirected to the overlay. This can result in Windows recreating the page file during startup and may trigger warnings indicating that a temporary paging file has been created. To avoid contention between the Windows memory manager and the Unified Write Filter, 10ZiG recommends placing the page file on an unprotected volume where possible.

UWF Overlay Requirement

Where the **Unified Write Filter (UWF)** is enabled on 8GB RAM devices, an **8GB disk overlay should be configured on the protected C:** partition to preserve system RAM.

This ensures adequate overlay capacity for standard thin-client workloads while maintaining sufficient operating space for Windows and installed applications.

With this standard configuration:

- The **C:** partition is protected by UWF
- Recommended UWF file and registry exclusions remain active
- Temporary and discardable writes are redirected into the overlay
- The **D:** partition remains permanently unprotected for page file operations

Note: This partition model should now be regarded as the preferred deployment standard for modern 128GB 10ZiG Windows 11 thin clients, providing the best balance of performance, stability, overlay efficiency(if using the UWF), and long-term SSD endurance.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

11 Example of a working build, with “Best Practice” features included

This section will pull together all we have mentioned in the early stages of the guide and advised in the previous “Advice, guidance and recommendations – Best Practice” section and show you what the UWF and Windows 11 IoT LTSC device looks like following the full setup.

We’ll also be mentioning what device we’re using and importantly, the resources it has available to it, regarding RAM and storage.

For these demonstrations, we’ll be listing the UWF commands next to the values in a table, but you can bundle these inside DOS CMD batch files and run them from within a command window for ease of execution. Just copy the content beneath the column heading “Command Line” into a DOS batch file or straight onto a command line.

It’s worth noting that these recommendations are all also included in the 10ZiG UWF Wizard, that we’ll touch on briefly at the end of this guide.

11.1 10ZiG 7011Q – 128GB of Storage, 8GB of RAM and an 8GB disk overlay with Page File

Applying the base UWF Overlay configuration

Overlay

Property	Value	Command Line
Type	DISK	uwfmgr overlay set-type DISK
Maximum Size	8192MB	uwfmgr overlay set-size 8192
Warning Threshold	4096	uwfmgr overlay set-warningthreshold 4096
Critical Threshold	6144MB	uwfmgr overlay set-criticalthreshold 6144

We’re setting the overlay on DISK, the overlay size to be 8GB and the warning and critical notification thresholds to be 4096MB and 6144MB respectively.

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

11.2 Recommended Exclusions for all UWF enabled devices

This list of exclusions is recommended for all UWF enabled devices, it contains mainly recommendations from Microsoft and 10ZiG. These can be added into your batch files as we mentioned earlier.

NOTE: These should be added into your config batch files by default, unless you have a specific reason not to do so. Also, we use “ThinClientUser” in these examples, but this depends on local users of the machine, you may have additional users to be considered.

11.3 Recommended File Exclusions

```
uwfmgr File Add-exclusion "C:\Windows\Prefetch"
uwfmgr File Add-exclusion "C:\ProgramData\Microsoft\dot3svc\Profiles\Interfaces"
uwfmgr File Add-exclusion "C:\ProgramData\Microsoft\wlansvc\Profiles\Interfaces"
uwfmgr File Add-exclusion "C:\Users\ThinClientUser\AppData\LocalLow"
uwfmgr File Add-exclusion "C:\Users\ThinClientUser\AppData\Local"
uwfmgr File Add-exclusion "C:\Users\Administrator\AppData\LocalLow"
uwfmgr File Add-exclusion "C:\Users\Administrator\AppData\Local"
uwfmgr File Add-exclusion "C:\ProgramData\Microsoft\Windows Defender"
uwfmgr File Add-exclusion "C:\Program Files\Windows Defender"
uwfmgr File Add-exclusion "C:\Windows\Temp\MpCmdRun.log"
uwfmgr File Add-exclusion "C:\Windows\WindowsUpdate.log"
uwfmgr File Add-exclusion "C:\Windows\wlansvc\Policies"
uwfmgr File Add-exclusion "C:\Windows\dot2svc\Policies"
uwfmgr File Add-exclusion "%ALLUSERSPROFILE%\Microsoft\Network\Downloader"
uwfmgr File Add-exclusion "C:\Setup"
uwfmgr File Add-exclusion "C:\Program Files (x86)\10ZiG"
```

11.4 Recommended Registry Exclusions

```
uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender"

uwfmgr Registry Add-exclusion
"HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\GraphicsDrivers\Configuration"

uwfmgr Registry Add-exclusion
"HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\BITS\StateIndex"

uwfmgr Registry Add-exclusion
"HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Wireless\GPTWirelessPolicy"
```

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\WiredL2\GP_Policy"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\wlansvc"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\dot3svc"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Wlansvc"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\WwanSvc"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\dot3svc"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation"

uwfmgr Registry Add-exclusion "HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\10ZiG"

11.5 Filter Enable and Volume Protection

At the beginning of the guide, we showed you how to protect the volume and enable the write filter. Here are those 2 commands below.

uwfmgr volume protect C:
uwfmgr filter enable

11.6 Switching OFF FreeSpace Passthrough on DISK Overlay Enabled UWF Installations

When enabling a disk-based UWF filter using the **uwfmgr filter enable** command shown previously, the UWF automatically sets **FreeSpace Passthrough** to **ON**. As this allows Windows to write outside the overlay without explicit control, it is recommended to disable it using the command below.

uwfmgr overlay set-passthrough off

What you need to be aware of, that any time you need to **enable the UWF** and you have a

disk overlay in operation, you will need to run this command, or the UWF will switch it back "**ON**" again. So, the order is uwfmgr filter enable then uwfmgr overlay set-passthrough off.

11.7 Windows Updates, 10ZiG UWF Related Scheduled Task and Apps

Install and set the 2 scheduled tasks "Scheduled Reboot" and also the "UWF Servicing" task, that controls the execution of UWF refresh and "UWF Servicing" mode.

Ensure that the apps WUService.exe and UWFService.exe are copied to the local C:\Setup folder.

To make sure that WUservice.exe is run on boot, create the registry key for HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run with the String Value name that suits your requirements, our was "DisableWU" and give it the Data value of "C:\Setup\WUService.exe".

11.8 Further Possible Exclusions for Consideration – VDI Clients

You might also want to add in some VDI specific Exclusions for your Overlay. We mentioned adding in "C drive" VDI log locations earlier on in the guide, but you might also want to add in VDI app specific locations too, especially if they reside on your "C: drive" UWF protected volume. Here are some examples below. **Note, these might change over time if your VDI vendors decide to change locations for certain content. Also note that {*logged on user*} is used below to denote different users. You will need to replace them with your specific user names.**

NOTE: If you plan on using the Microsoft Windows App, then make sure that the two locations listed below are NOT excluded. So, if you have them excluded already as part of the "Recommended File Exclusions" in section 11.3, then you must remove them using the commands below.

```
uwfmgr File Remove-exclusion "C:\Users\ThinClientUser\AppData\LocalLow"
uwfmgr File Remove-exclusion "C:\Users\Administrator\AppData\LocalLow"
```

Omnissa Horizon Client

Files/Folders

uwfmgr File Add-exclusion “"C:\Program Files\Omnissa\Omnissa Horizon Client"
uwfmgr File Add-exclusion "C:\ProgramData\Omnissa\Horizon"
uwfmgr File Add-exclusion "C:\ProgramData\Omnissa\Horizon\logs"
uwfmgr File Add-exclusion “C:\Windows\Temp\vmware-SYSTEM”
uwfmgr File Add-exclusion “C:\Users\{*logged on user*}\AppData\Roaming\Omnissa”
uwfmgr File Add-exclusion “C:\Users\{*logged on user*}\AppData\Local\Omnissa”

Registry

uwfmgr Registry Add-exclusion “HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Omnissa”
uwfmgr Registry Add-exclusion “HKEY_LOCAL_MACHINE\SOFTWARE\Omnissa”
uwfmgr Registry Add-exclusion “HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Omnissa”
uwfmgr Registry Add-exclusion “HKEY_CURRENT_USER\Software\Omnissa\Horizon\Client”

Microsoft Windows App

Files/Folders

uwfmgr File Add-exclusion "C:\Program Files\WindowsApps\MicrosoftCorporationII.Windows365_8wekyb3d8bbwe_<version>_x64__8wekyb3d8bbwe"

uwfmgr File Add-exclusion "C:\Users\{*logged on user*}\AppData\Local\Temp\MicrosoftCorporationII.Windows365_8wekyb3d8bbwe"

Citrix Workspace App

Files/Folders

uwfmgr File Add-exclusion “C:\Program Files (x86)\Citrix”
uwfmgr File Add-exclusion “C:\Users\{*logged on user*}\AppData\Local\Citrix”
uwfmgr File Add-exclusion “C:\Users\All Users\Citrix”

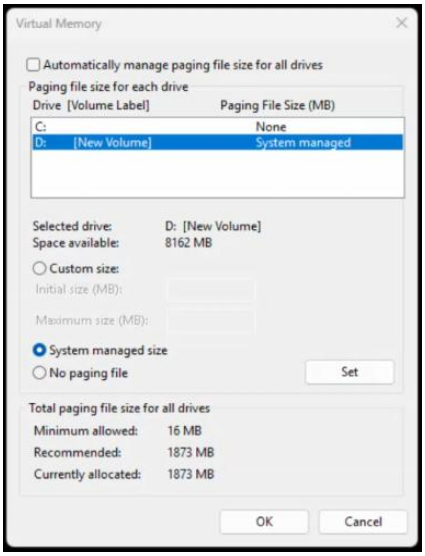
Registry

uwfmgr Registry Add-exclusion “HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Citrix”
uwfmgr Registry Add-exclusion “HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Citrix”

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

11.9 Setting the Page File

If you're using a Page File as we recommended in section 10 previously, the page file should be ideally set to "System Managed" and reside on the "un-protected" D: partition as shown below.

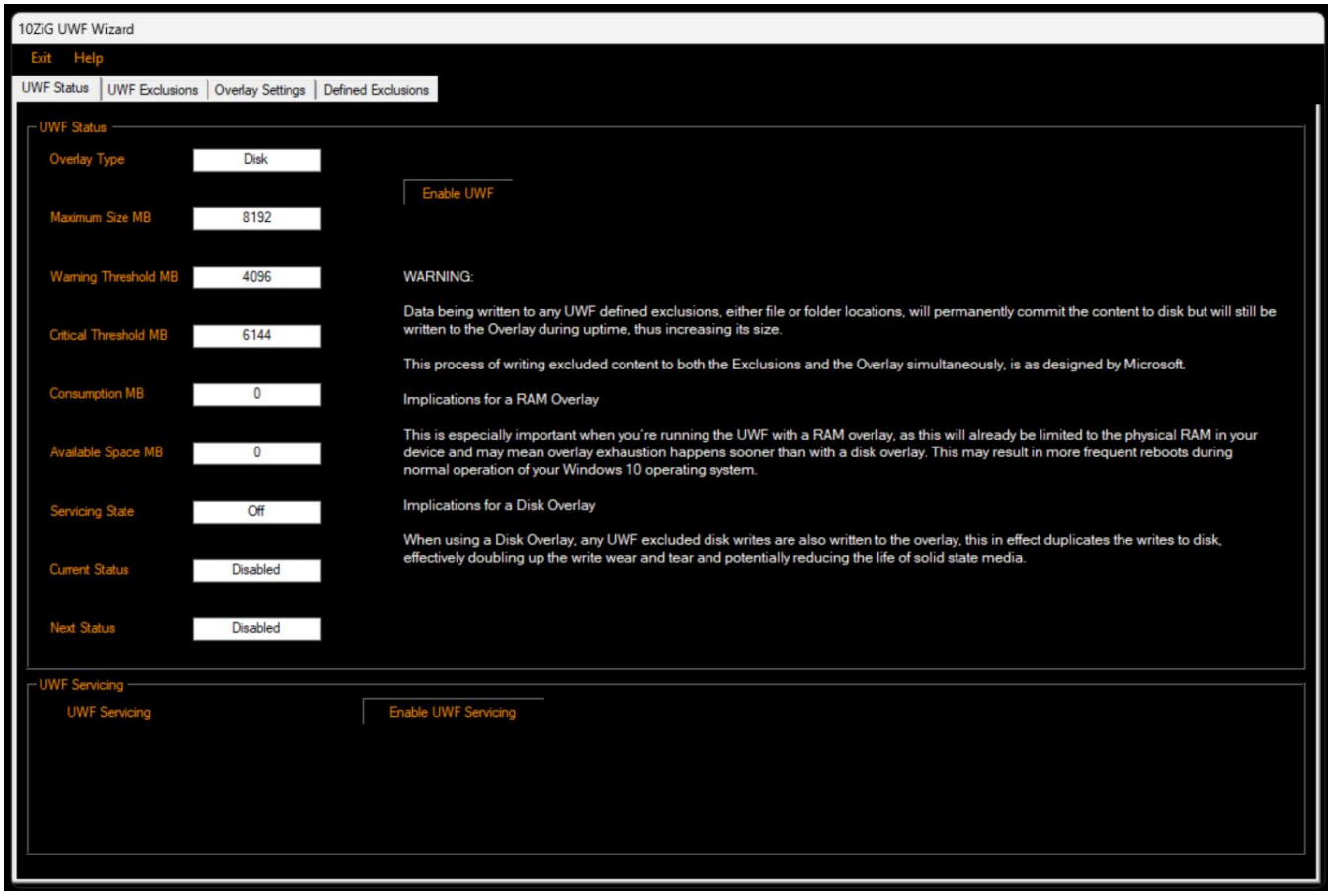


12 Introduction to the 10ZiG UWF Wizard

In this section we'll give you a brief introduction to the 10ZiG UWF Wizard and show you around some of the GUI features we've already mentioned from within the DOS command line environment.

12.1 UWF Status Screen

On the UWF Status tab, you can see the basic configuration and state of the UWF. Note that the "WARNING" message serves as a reminder of what happens to the UWF overlay, especially when using exclusions.

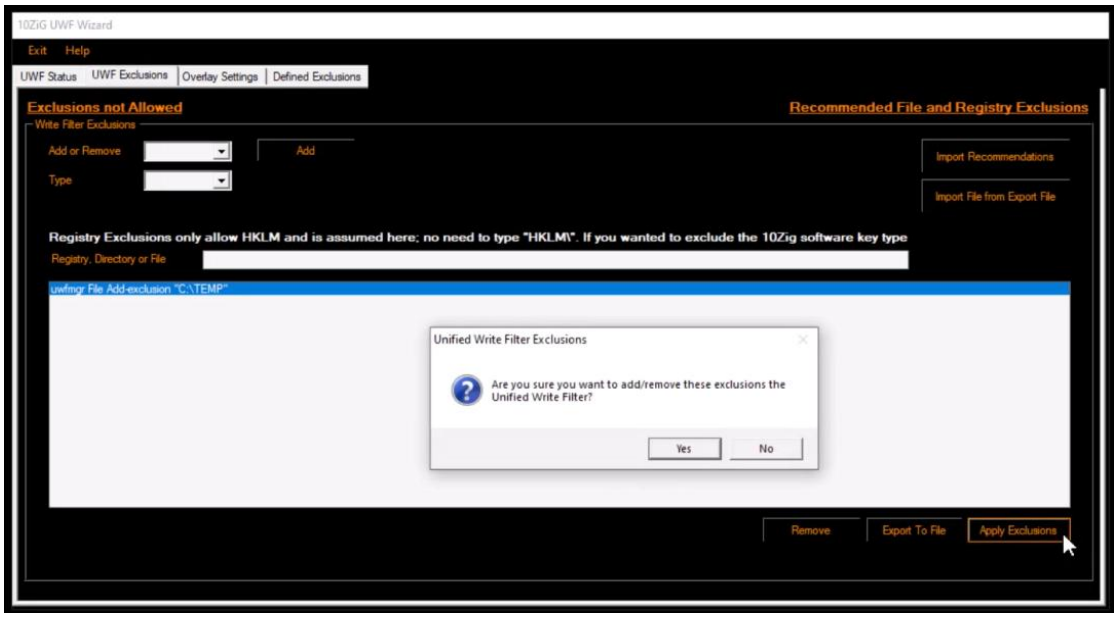


10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

12.2 UWF Exclusions

On the UWF Exclusions tab, you can either create your own file or registry exclusions, export or import any exclusions to or from a file for later use or import recommended exclusions, as we mentioned earlier in the guide.

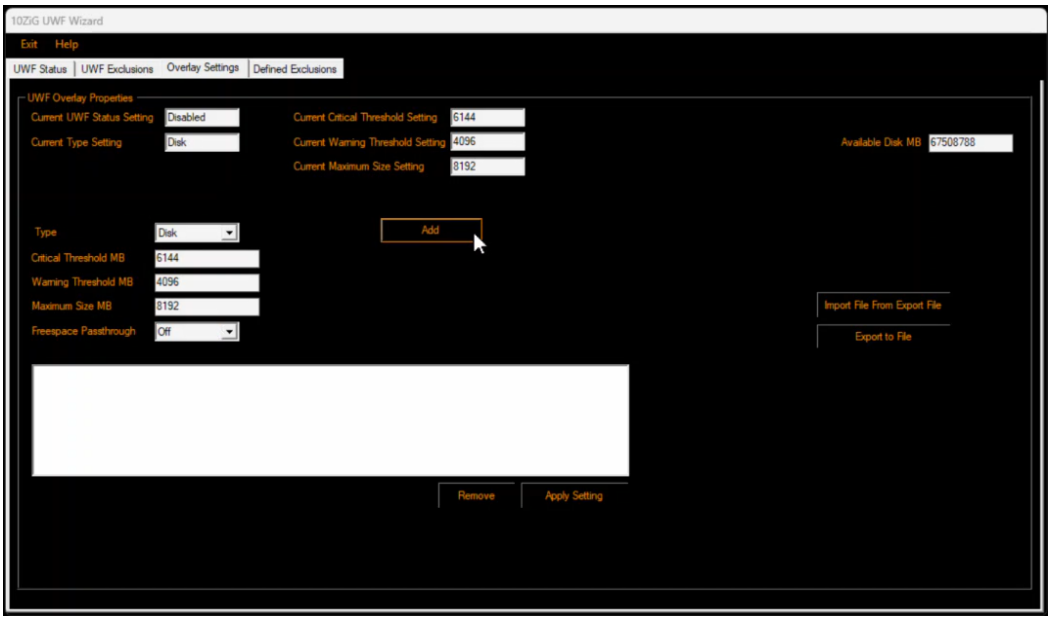
The example below shows setting up an exclusion for the folder C:\TEMP. You can see this in the second screen, where the command uwfmgr File Add-exclusion “C:\TEMP” is displayed and highlighted in blue.

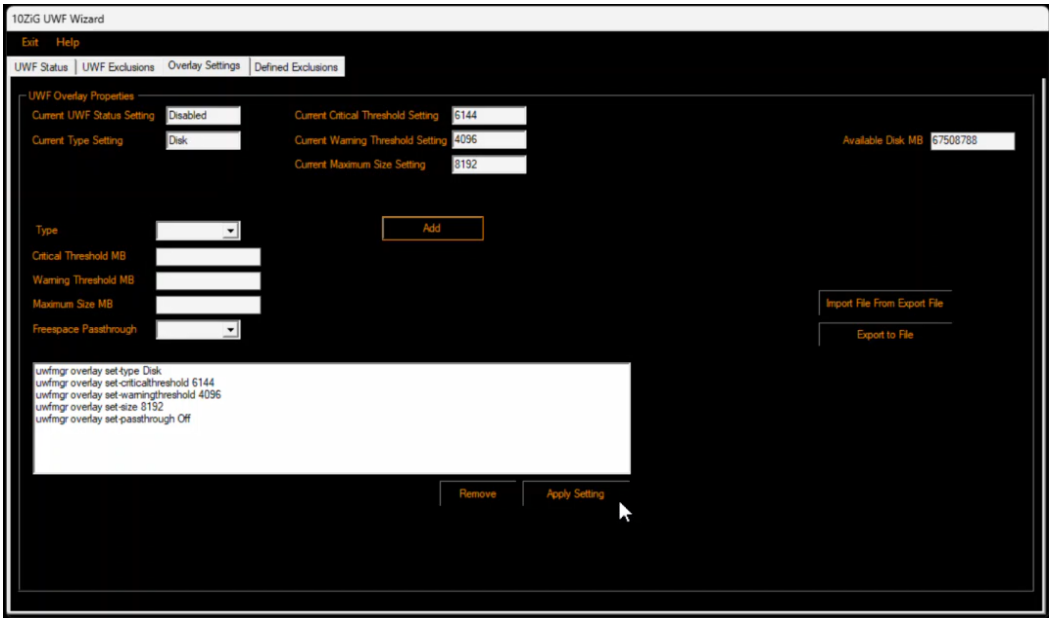


12.3 Overlay Settings

Inside the Overlay Settings tab, you have the option to change the Overlay type, size, and its notification thresholds. As with the exclusions tab, you also have the ability to export the config or import from a previously exported one.

The example below shows changing the “CriticalThreshold” and “WarningThreshold” and size to the values 6144MB, 4096MB respectively and the second screen shows the relative uwfmgr commands, where you just click “Apply Setting” to change the Overlay.



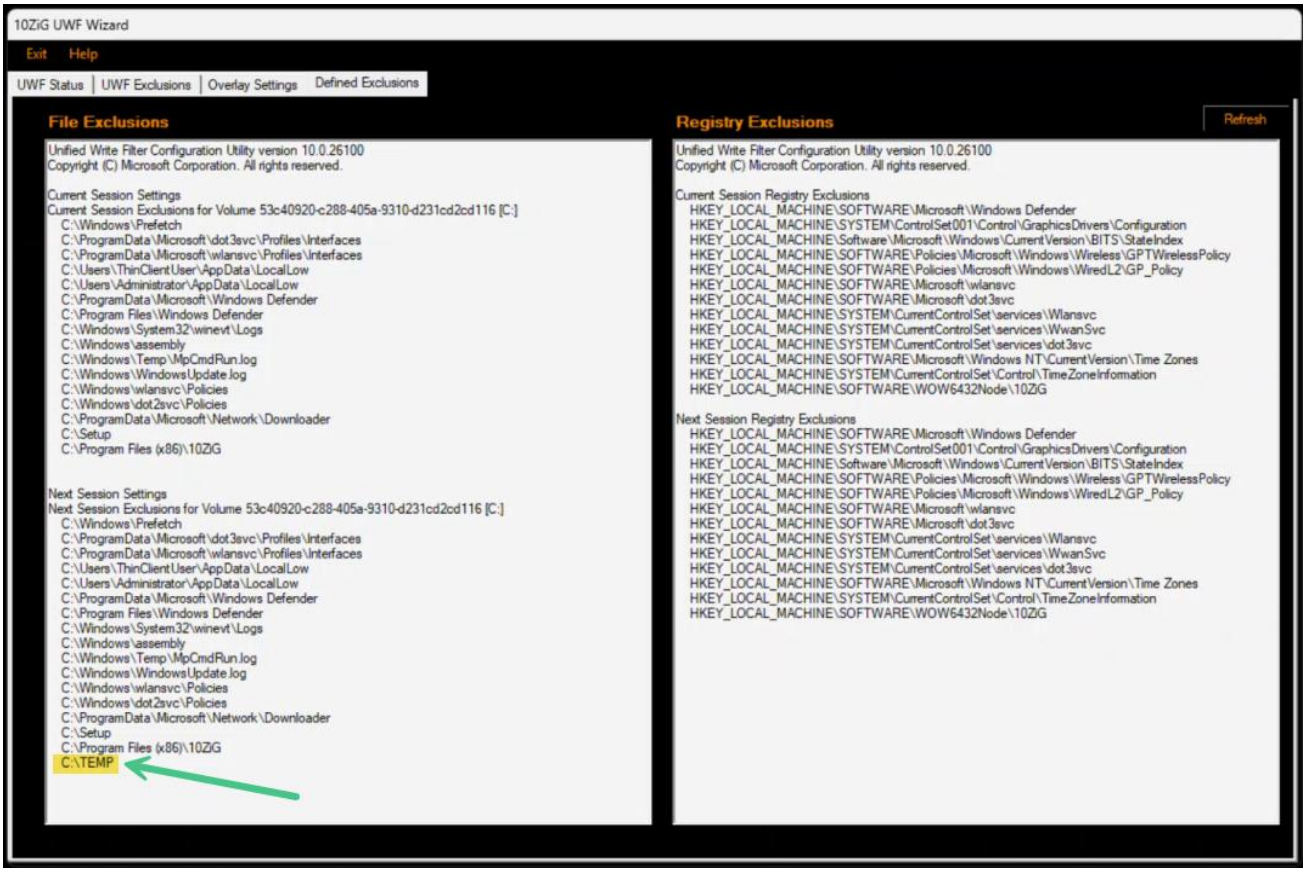


10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

12.4 Defined Exclusions

In the Defined Exclusions tab, you can see the exclusions that are already defined in the “Current Session Settings” and the “Next Session Settings”. You can make changes inside the “UWF Exclusions” tab and then once you have applied them and open this tab, they will be reflected in the “Next Session Settings”. If you have changed them and they don’t seem to have appeared, just click the “Refresh” button in the top right corner of the screen.

The screen below shows our already added C:\TEMP exclusion, we added a couple of pages back. The remaining file/folder and registry exclusions were imported as the recommended exclusions prior to this guide.



13 Supporting Complimentary Documents and Links

Below are some useful supporting online website content and links to 10ZiG videos.

UWF Specific Content

https://learn.microsoft.com/en-us/windows-hardware/customize/enterprise/unified-write-filter

UWF Overlays

https://learn.microsoft.com/en-us/windows-hardware/customize/enterprise/uwfoverlay

UWF Servicing Mode and Windows Updates

https://learn.microsoft.com/en-us/windows/configuration/unified-write-filter/uwf-apply-windows-updates

10ZiG YouTube channel for Video relating to the UWF and Windows Updates.

https://www.youtube.com/watch?v=O0DrvqSgTP4&t=2120s

10ZiG | US P: (866) 865-5250 E: info@10ZiG.com | EMEA P: +44 (0) 116 214 8650 E: info@10ZiG.eu

Support

If you require support for any of the information within this document, please contact your region's nearest Technical Support Center.

10ZiG Technology, Inc.

Headquarters USA (North America)

2043 W. Lone Cactus Drive
Phoenix, AZ 85027

Phone 866-865-5250

support@10zig.com

sales@10zig.com

www.10zig.com

Headquarters UK (EMEA)

10ZiG Technology Limited

7 Highcliffe Road
Leicester
LE5 1TY
UK

Phone +44 (0)116 2148661

support@10zig.eu

sales@10zig.eu

www.10zig.eu

END OF DOCUMENT