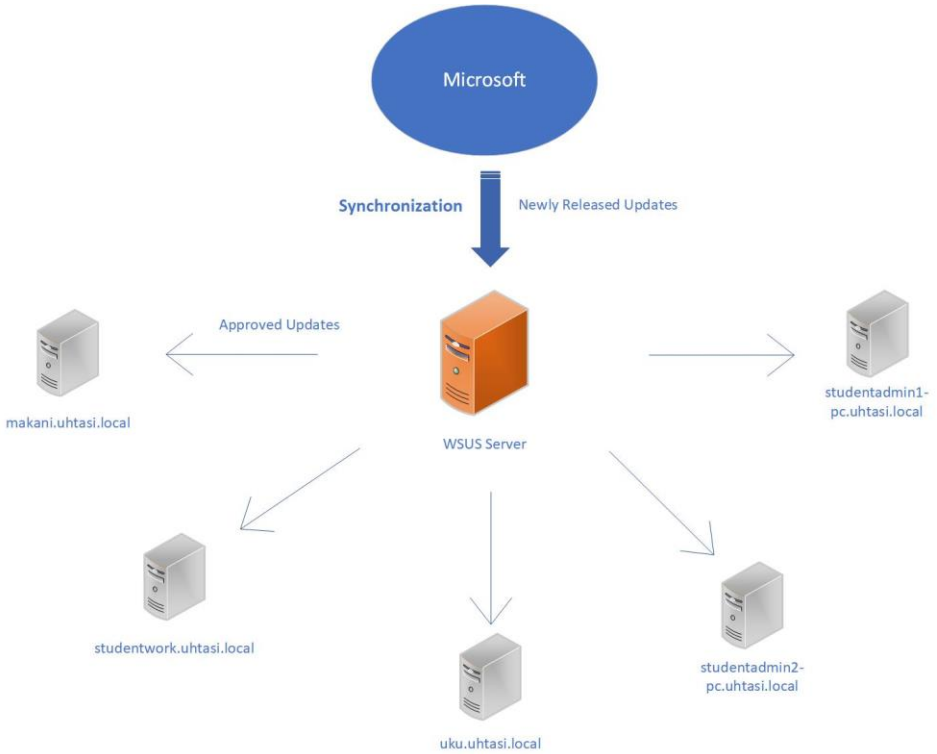
Windows Server Update Services (WSUS) Overview + User Guide

What is WSUS?

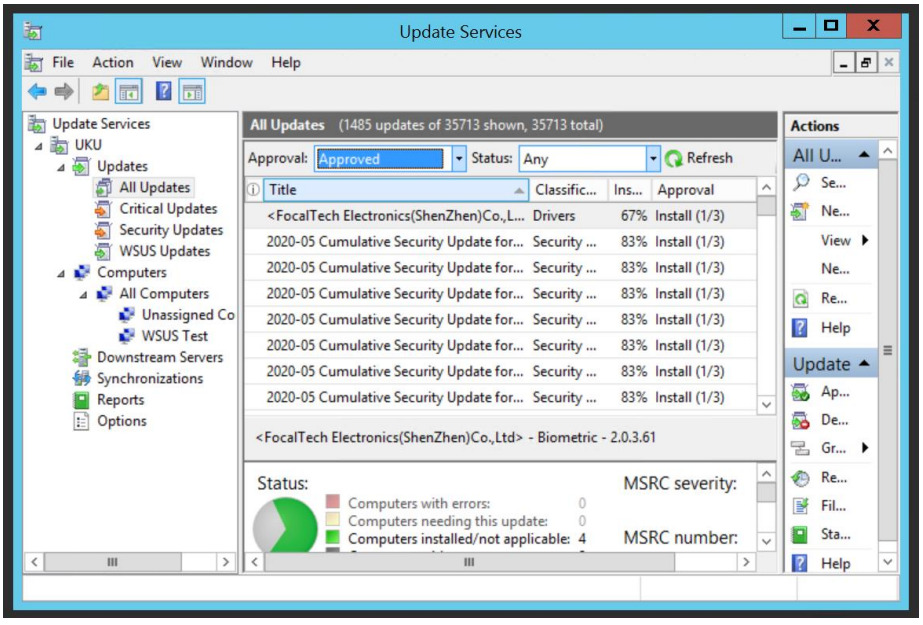
WSUS is a computer program and network service developed by Microsoft Corporation that enables administrators to manage the distribution of updates, patches, and hotfixes released for Microsoft products to computers in a corporate environment. It can be configured to download and approve updates for specific workstations either manually or automatically through scripting or the software itself.



- *WSUS cycle for the TASI-PHIDC WSUS Test group* -

How does WSUS work?

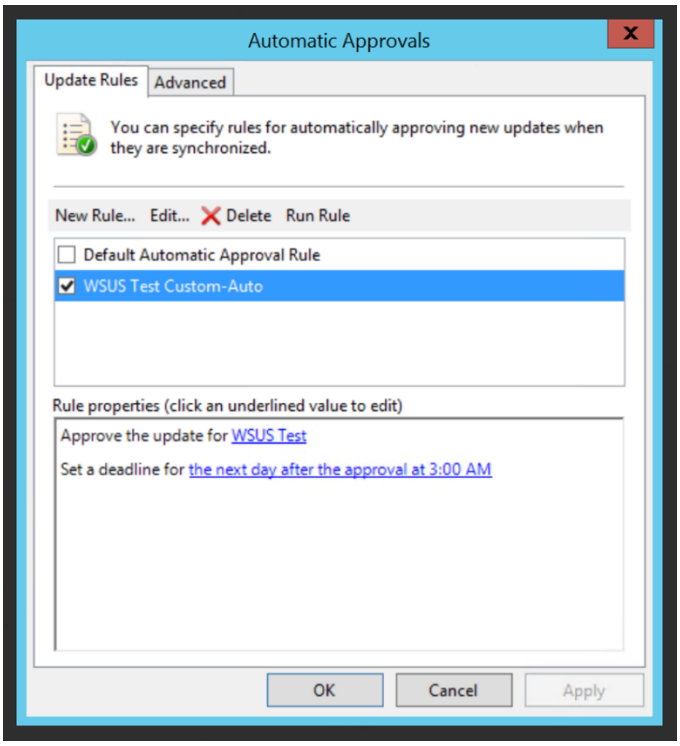
1. Synchronization
   - During synchronization, a WSUS server downloads updates from an update source (for example, Microsoft).
   - Synchronization can be done manually through the software at any time, or it can be scheduled to automatically sync at a particular time of day through Options -> Synchronization Schedule.
2. Approval Process
   - Once the update is synchronized into the WSUS system, it can be approved manually through Update Services -> UKU -> Updates -> All Updates. To see unapproved updates, filter the Approval attribute to Unapproved. Users can select if they want to approve for a workstation group or a specific workstation.



   - Approvals can also be done automatically through Options -> Automatic Approvals. Users can design an Automatic Approval Rule. This rule approves updates for a group of workstations based on attributes. These attributes sort updates by Classification and Product. Some examples of Classifications are Critical Updates,

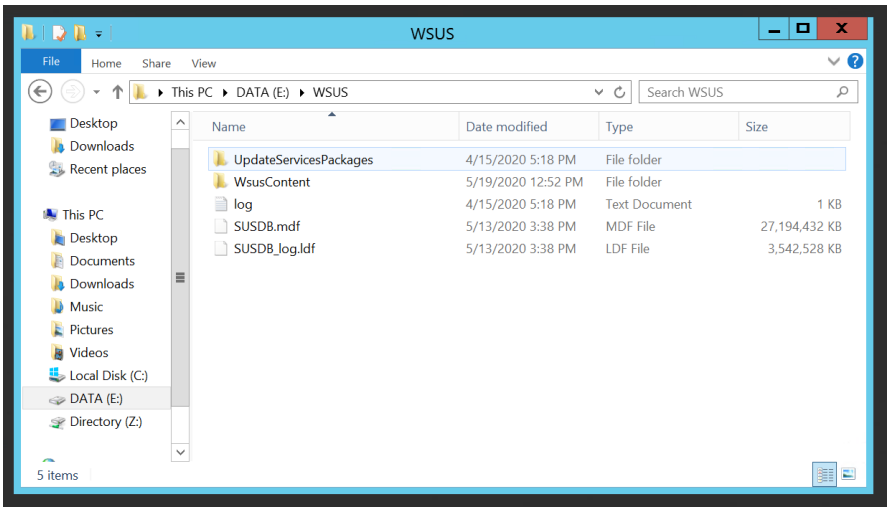
Drivers, Feature Packs, etc. Some examples of Products are specific types of Windows servers/operating systems, Microsoft products, SQL products, etc.

- The Automatic Approval Rule sets a deadline to force restart and install updates the next day after the approval at 3:00 AM. The rule will kick in at this time if the update has not managed to be installed by that time (which is supposed to be rare). It is set at 3:00 AM because that is outside work/waking hours. However, this deadline rule should not apply to servers, because we have a group policy setting to not auto-restart servers.



Where does WSUS store data (updates)?

The updates are stored within the WSUS database file called SUSDB.mdf. The SUSDB.mdf file is located in a drive on a WSUS server.

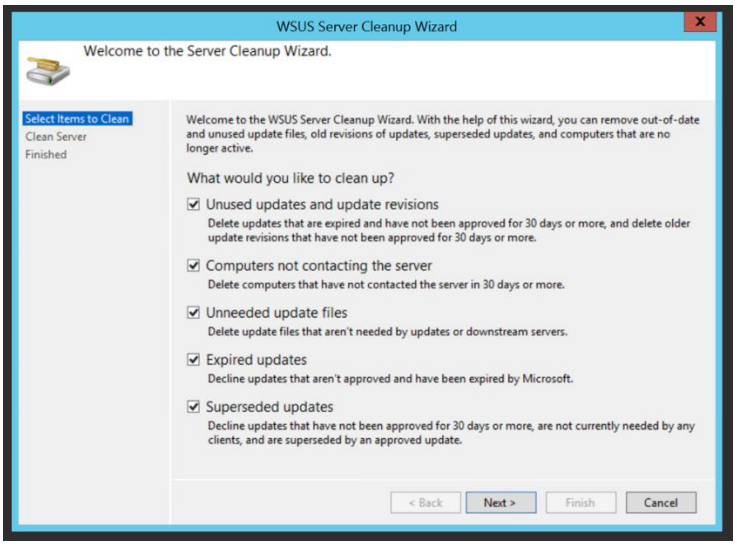


- *The location of the SUSDB.mdf file on the WSUS server* -

How can WSUS be managed or "cleaned"?

WSUS has issues with maxing out its space. Updates that are synced to the system quickly add up. Several thousand updates can be downloaded over the span of a few weeks. Therefore, if left unchecked WSUS can quickly become overwhelmed and the SUSDB.mdf can grow to massive sizes. Once the updates grow to be too many, the user will begin to notice problems with the software. A major problem is loading times. When the user is trying to view updates, or workstations, WSUS will often take an extremely long time to load the information. In many cases, the software will simply timeout and display "Error: Connection Error – An error occurred trying to connect to the WSUS Server". Instead of trying to push through these timeouts, it is better to remove the outdated and obsolete updates from the SUSDB.mdf file.

1. Server Cleanup Wizard
    - In Options -> Server Cleanup Wizard, users can remove unused, unneeded, expired, and superseded updates.



2. <u>Deny Updates</u>
   - Updates that have been approved and then Installed or deemed Not-Applicable by the workstation take up space. They should be denied in order to be removed from the workstation's storage and report space in WSUS. This can be done manually or through script. A PowerShell script is useful when there is a bulk of updates to deny. Note that denying an update that has the status of Installed does not uninstall the update on the workstation.

Deny-NotApplicables.ps1*

```
1  Get-WSUSUpdate -Classification All -Status InstalledOrNotApplicable  -Approval Any | Deny-WsusUpdate
2
3  Invoke-WsusServerCleanup -CleanupObsoleteUpdates -CleanupUnneededContentFiles
```

3. <u>Clean-WSUS script</u>
   - This PowerShell script is recommended to clean out WSUS databases. The user can run this script if the database they are working with is very bloated. Note to use this script a user may need to download SQL Command Line Utilities and an ODBC Driver from Microsoft. Sample output of the script:

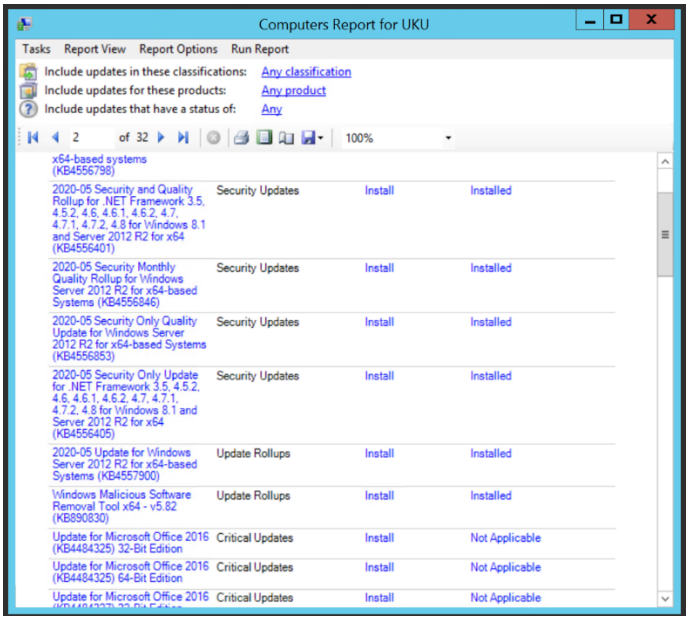
```
9357/10910: Deleting 164969 Apr 25 2020  3:59PM
9358/10910: Deleting 164967 Apr 25 2020  3:59PM
9359/10910: Deleting 164965 Apr 25 2020  3:59PM
9360/10910: Deleting 164963 Apr 25 2020  3:59PM
9361/10910: Deleting 164961 Apr 25 2020  3:59PM
9362/10910: Deleting 164855 Apr 25 2020  3:59PM
9363/10910: Deleting 164803 Apr 25 2020  3:59PM
9364/10910: Deleting 164751 Apr 25 2020  3:59PM
9365/10910: Deleting 164699 Apr 25 2020  3:59PM
9366/10910: Deleting 163724 Apr 25 2020  4:00PM
9367/10910: Deleting 163722 Apr 25 2020  4:00PM
9368/10910: Deleting 163720 Apr 25 2020  4:00PM
9369/10910: Deleting 163718 Apr 25 2020  4:00PM
9370/10910: Deleting 163716 Apr 25 2020  4:00PM
9371/10910: Deleting 163714 Apr 25 2020  4:00PM
9372/10910: Deleting 163712 Apr 25 2020  4:00PM

      Clean-WSUS Script Duration: 01:02:18:50


      ################################
      #    End of the WSUS Cleanup   #
      ################################
```

What are the options for WSUS reporting?

1. Through WSUS
    - Computer Status Summary: Shows diagram of updates and their statuses for the workstation.
    - Computer Detailed Status: Lists all updates that were pushed to the workstation. Shows Approval and Installation statuses. Also shows the diagram of updates for the workstation.
    - Update Status Summary: Shows a summary of the update status, with one page per update.
    - Update Detailed Status: Shows a summary of the update status of all computers per update, with one page per update.
    - The above reports also have Tabular versions.
    - WSUS also has an email tool in Options -> E-Mail Notifications. Can be configured to send reports via email on a daily or weekly basis.



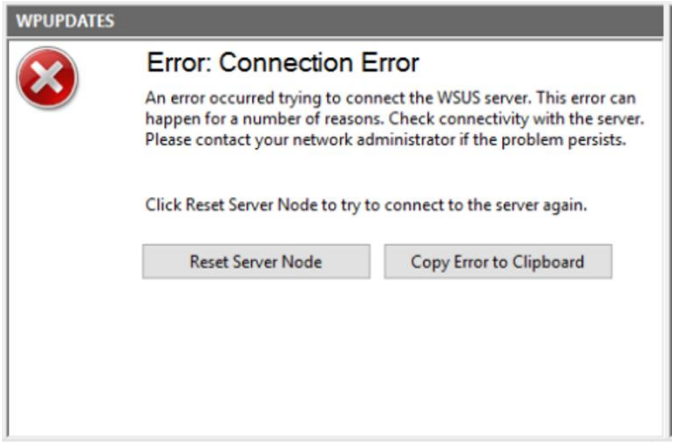
2. Through Scripting
   - It is possible to get creative through PowerShell scripting to obtain information from WSUS and send it through email as a report.
   - Microsoft has created a host of cmdlets for WSUS tasks.
   - There are also many scripts created online that can be used if trusted.

Index	Status	Computer Name	IP Address	Last Contact	Total updates	Awaiting reboot	Ready to install	Download pending	Failed	Unknown State
1	Reboot needed	studentadmin1-pc.uhtasi.local	10.100.10.125	Since 2 days.	252	4	0	10	0	238
2	Reboot needed	studentadmin2-pc.uhtasi.local	10.100.10.128	Yesterday	251	3	0	10	0	238
3	Ready to install	studentwork2-pc.uhtasi.local	10.100.10.127	Since 3 days.	442	0	12	20	0	410
4	Reboot needed	studentwork-pc.uhtasi.local	10.100.10.136	Yesterday	255	4	0	13	0	238
5	Pending	uku.uhtasi.local	10.100.10.252	Today	72	0	0	72	0	0

- *One of the reports generated from a PowerShell script* -

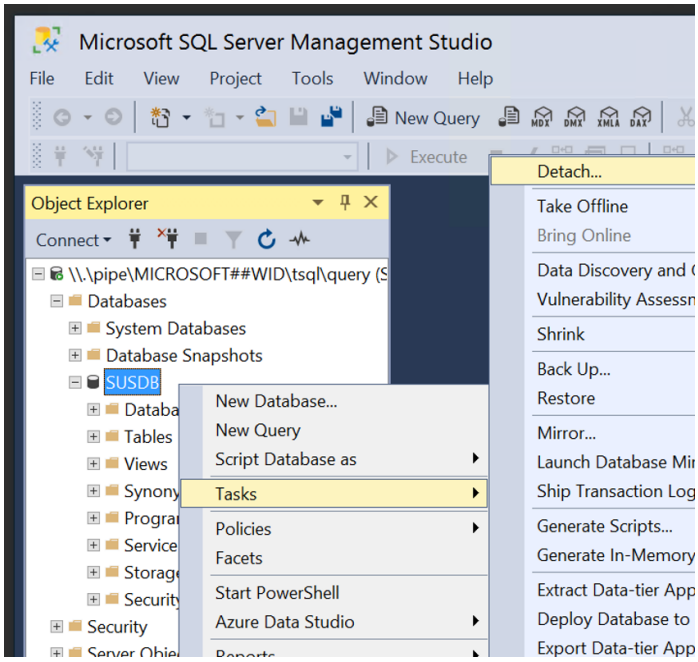
My Experience and Configuration of WSUS

Overview: When I first was exposed to the WSUS software, I immediately noticed it was slow and laggy. It was difficult to use because of how clunky it was. It seemed slow to respond to my mouse-clicks. When I tried to load reports, sometimes it would work after a while, other times it would simply time-out and I would be forced to restart.



Doing some research online, I learned that a way to solve connection issues were to go into IIS (Internet Information Services) and give WSUS access to more memory than it was configured. After doing this, I had some minor success. However it soon returned back to its state of being stuck in the muck. It was annoying to keep going back and forth between resetting the IIS internet pool and the WSUS software.

I then installed WinDirStat and ran the program. I learned that there was a massive file occupying space on the C drive: the SUSDB.mdf file. It was taking up over 26 GB on the drive. I realized that the WSUS's slow speed was likely attributed to how bloated that file was. I noticed that some of the updates in WSUS were stretching as far back as 2001. Also, I was told that it would be ideal for the SUSDB.mdf file to be moved from the C drive to the E drive.



- *Detaching the SUSDB database file in SSMS* -

After managing to move the file to the E drive using SQL Server Management Studio, I then looked into how to clean out the file of all the obsolete updates. I tried running the Server Cleanup Wizard in WSUS, but it would just get stuck and timeout. I ended up finding a script online that was highly recommended: Clean-WSUS PowerShell script. I downloaded the script and installed its needed components. I ran the script and it ended up taking a few days to work through.

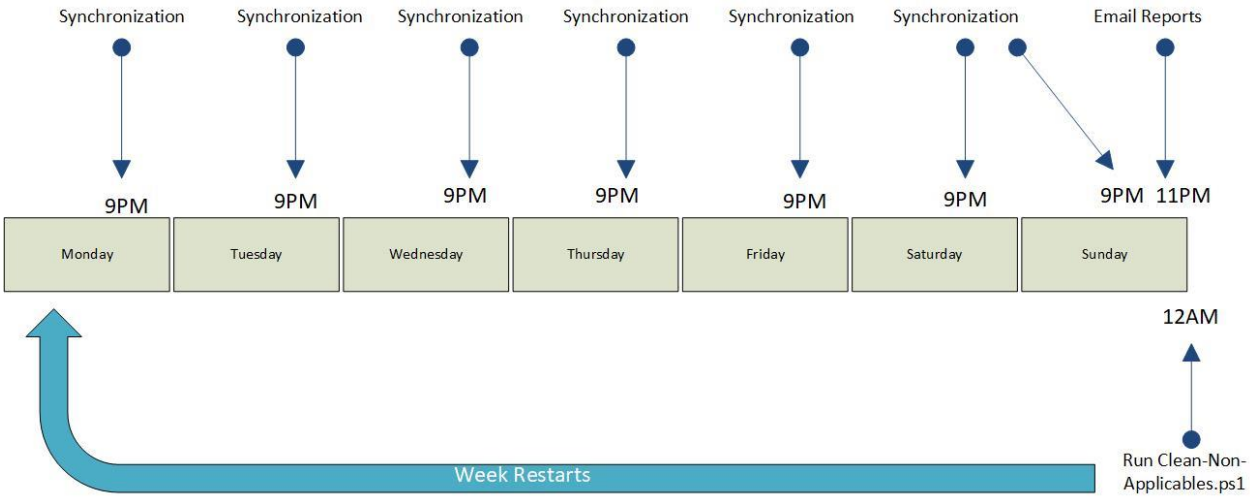
However the results of the script were immediate and apparent. The software was now much smoother to use, I could load reports within seconds, and timeouts were now nonexistent. The script brought down the update count from around 127,000 to around 33,000. The updates within the workstations were brought down to around 1600 (overall <2% of the original volume).

After this point, I began automating the approvals and reporting through a combination of the WSUS software and PowerShell scripting. One thing I noticed were that many Not-Applicable updates were getting through the approval filter. These Not-Applicable updates serve no use and just take up space in WSUS. My theory for why so many Not-Applicable updates are

getting approved is because the Automatic Approval Rule is approving updates for All Products (there are dozens of products that the rule looks at). I do not want to disable any products because I am not 100% certain what products we need and what we do not (there are many obscure products that I do not have experience with). As of now, the approving of the Not-Applicable updates is virtually harmless, and can be removed quickly via script.

WSUS Scheduling

Here I will list the schedule I have designed for WSUS. WSUS works in a cycle, so having a concrete schedule in place is very helpful.



1. Synchronize at 9:00:00 PM daily
2. Auto Approval works in tandem with the synchronization, but is also setting a deadline for [the next day after the approval at 3:00 AM]
3. Set the reports to be emailed at Sunday 11:00PM.
    - I set the reports to be emailed to info@uhtasi.org
    - I will also send the tabulated report created with the PowerShell script.

4. The Clean-Not-Applicables script will be run on Monday at 12:00AM to remove all updates that have already been Installed or deemed Not-Applicable, to save space in the reporting.
5. The Clean-WSUS script can be run if needed (if the database grows too bloated and decreases performance). Once we start noticing timeouts, this script should be run.