

DOI [10.28925/2663-4023.2022.15.7184](https://doi.org/10.28925/2663-4023.2022.15.7184)

УДК 004.05

Булатецький Віталій Вікторович

к. фіз.-мат. наук, доцент, доцент кафедри комп’ютерних наук та кібербезпеки
Волинський національний університет імені Лесі Українки, Луцьк, Україна

ORCID ID: 0000-0002-9883-4550

Bulatetsky.Vitaly@vnu.edu.ua**Булатецька Леся Віталіївна**

к. фіз.-мат. наук, доцент кафедри комп’ютерних наук та кібербезпеки
Волинський національний університет імені Лесі Українки, Луцьк, Україна

ORCID ID: 0000-0002-7202-826X

Bulatetska.Lesya@vnu.edu.ua**Гришанович Тетяна Олександровна**

к. фіз.-мат. наук, завідувач кафедри комп’ютерних наук та кібербезпеки
Волинський національний університет імені Лесі Українки, Луцьк, Україна

ORCID ID: 0000-0002-3595-6964

Hryshanovych.Tatiana@vnu.edu.ua**АНАЛІЗ ФАЙЛОВИХ ОБ’ЄКТІВ ОПЕРАЦІЙНОЇ СИСТЕМИ WINDOWS 10 ДЛЯ
ОЧИЩЕННЯ Й ОПТИМІЗАЦІЇ ПРОСТОРУ СИСТЕМНОГО РОЗДІЛУ**

Анотація. У роботі проведено аналіз причин та ряд способів вирішення проблеми браку вільного місця на системному розділі в процесі експлуатації операційної системи Windows 10. Досліджено особливості організації розділів носія при встановленні на нього поширених операційних систем Microsoft. Розглянуто основні системні файлові об’єкти, шляхи до них, їх призначення та роль у використанні простору системного розділу та у функціонуванні операційної системи. Наведено перелік шляхів до файлів та папок, що займають найбільші об’єми на системному розділі, коректно очистивши або налаштувавши які можна зекономити значний простір на системному розділі. Розглянуто основні інструменти операційної системи для реалізації методів очищення: інтерпретатор командного рядка, середовище PowerShell, засоби редактування системного реестру та засоби виклику функцій системних бібліотек. Особливу увагу приділено пакетним файлам, як найпростішим та ефективним засобам реалізації автоматизованого використання можливостей системних утиліт, з врахуванням особливостей роботи інтерпретатора командного рядка. Як один із способів маніпулювання об’єктами за допомогою командлетів, розглядається середовище PowerShell, яке здатне не обробляти команди, а генерувати потоки об’єктів, що стосуються цієї команди та формувати їх оброблене представлення за допомогою як командного, так і графічного інтерфейсу. Описана логічна та фізична будова системного реестру, як одного із засобів керування налаштуваннями Windows. Проведено порівняльний аналіз деяких програмних продуктів, таких як системна консольна утиліта обслуговування образів розгортання і управління ними та ряд безкоштовних програмних засобів. Сформовано перелік функціоналу, який використовується для оптимізації та очистки системного розділу від надмірної інформації. На основі сформованого переліку проаналізовано ефективність використання розглянутих програмних засобів для очистки системного розділу.

Ключові слова: операційна система, системний розділ, оптимізація, надмірні дані, системний реєстр, пакетний файл, PowerShell, інтерпретатор командного рядка

ВСТУП

Постановка проблеми. Сучасна найпоширеніша операційна система Windows в останніх версіях присутня у кількох редакціях, від Starter до Enterprise (в залежності від

версій набір може дещо відрізнятись). Проте загалом такі збірки операційної системи носять універсальний характер і призначенні для широкого кола користувачів та апаратних засобів, а, відповідно, і для виконання широкого спектру завдань з різними вимогами. З цієї причини ці збірки досить часто містять у собі надмірну кількість файлової інформації. Самі операційні системи є динамічно змінюваними і нарощують кількість системних файлів у процесі експлуатації, що пов'язано з постійним їх оновленням, виправленнями, встановленням додаткових програмних засобів, пристрійв та їх драйверів, створенням резервних копій, збоїв, тощо. Проте розмір системного розділу є скінченим, а, отже, перед користувачем рано чи пізно постане проблема браку вільного місця, що в свою чергу призводить до гальмування роботи системи, неможливості розгортання останніх актуальних оновлень та виправлень, збоїв у роботі тощо. Тому виникає потреба у корегуванні налаштувань та періодичні очистці (англ. cleanup - прибиранні) системного розділу від непотрібної або надмірної файлової інформації, завантажених і встановлених пакетів оновлень та виправлень, драйверів пристрійв, які уже не використовуються, застарілих резервних копій, тимчасових файлів, файлів, пов'язаних з функціями операційної системи, які вже не використовуються тощо. Тобто, виникає потреба в оптимізації системного розділу. [1]

Аналіз останніх досліджень і публікацій. В процесі розвитку операційної системи Windows, починаючи з найдавніших (Windows 3.1, Windows 95...) і аж до Windows XP, розробники дотримувались позиції, згідно якої, на відміну від *nix-систем, всі файли та папки, як системні, так і користувальницькі, повинні розміщуватись на одному первинному розділі. Цей розділ повинен бути активним та завантажувальним. Такий підхід в першу чергу був зумовлений особливістю операційної системи MsDos, яка є основою операційних систем Windows аж до Windows Millenium (1999р) включно. Адже MsDos могла повноцінно працювати лише з одним первинним розділом на одному жорсткому диску, розміченому за стандартом MBR (Master Boot Record). Починаючи з початку 90-х паралельно велась розробка операційних систем типу Windows NT, які вже не були надбудовою над MsDos, і, відповідно, не мали пов'язаних з нею обмежень, що стосувались розділів на жорсткому диску. Навіть сучасна Windows 11 базується на коді, що став основою Windows NT. Проте, з міркувань сумісності з попередніми версіями операційних систем на рівні файлової структури, розробники і надалі притримувались принципу «Все на одному розділі». У цьому були свої переваги та недоліки. Одним розділом легше маніпулювати, робити його резервні копії, відновлювати, використовувати єдину файлову систему для усіх даних, використовувати декілька операційних систем (кожна на своєму розділі), тощо. Але в цьому був і основний недолік – у випадку незворотного пошкодження такого розділу, користувачі зазнавали втрат власних даних. Частина OEM-виробників почала використовувати декілька розділів, застосовуючи не вмонтовані в операційну систему програмні засоби для реалізації механізмів відновлення та підвищення стійкості операційної системи до пошкоджень. Такі засоби виробники розміщували на інших розділах. Проте лише з появою Windows Vista, а точніше, із появою підтримки в ній нового стандарту GPT (GUID (Globally Unique Identifier) Partition Table) розмітки накопичувачів, з'явилася можливість розподілити по розділах (створити їх тепер можна було аж до 128) системні та користувальницькі дані [2]. Але, нажаль, розробники і далі за замовчуванням залишають користувальницькі та системні дані на одному розділі, залишаючи користувачам можливість та право вибору самостійно переносити власні папки на інші розділи, хоча в процесі встановлення (починаючи з Windows Vista) створюється або використовується більше, ніж один

первинний розділ, але призначені вони в основному для підвищення стійкості самої операційної системи.

Так, наприклад, Windows 10 при своєму встановленні створює за замовчуванням 4 розділи на нерозміченому накопичувачі [3]:

- Recovery – зарезервований операційною системою розділ, де зберігаються файли, необхідні для скидання операційної системи до початкового стану, і її завантажувач.
- EFI System (Hidden from disk management) – використовується для запису завантажувачів операційних систем, які (завантажувачі) пов'язані через UEFI (заміна базової системи введення-виведення (BIOS)).
- MSR – захищений розділ Microsoft до 128 Мб, створений для резервування частини простору з можливим подальшим використанням операційною системою Windows, встановленої в іншому окремому розділі.
- C: Drive – основний системний розділ.

Оскільки в більшості випадків звичайні користувачі не приділяють значення місцю розміщення власних даних (або просто не мають навиків їх переміщення), то на системному розділі накопичується ряд непотрібних користувачьких файлів. При цьому виникає потреба періодичної оптимізації системного розділу шляхом його очищення (прибирання) як від зайвих користувачьких файлів, так і від файлових даних самої операційної системи, які утворюються в процесі її експлуатації та оновлення [4].

Операційна система містить у собі засоби для прибирання (cleanmgr та ін.) та їх зазвичай виявляється недостатньо. Не завжди, або не вчасно, вони спрацьовують в автоматичному режимі або користувачі не мають до них доступу. Тому користувачі часто використовують стороннє програмне забезпечення або здійснюють прибирання та налаштування вручну [1].

Метою статті є розглянути структуру файлів і папок операційної системи та їх роль в її функціонуванні; проаналізувати умови та підібрати методи для зменшення розміру системного розділу, шляхом видалення деяких системних файлів та папок, або визначити налаштування для оптимізації їх розміру. Проаналізувати існуючі програмні продукти, приховані від звичайного користувача утиліти, сторонні утиліти для очистки та оптимізації системного розділу.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Вміст системних каталогів операційної системи Windows 10 та їх призначення. Наведемо перелік основних системних каталогів досліджуваної операційної системи та їх призначення. З самого початку зазначимо, що %systemroot% – системна змінна, що позначає каталог Windows у корені системного розділу (те саме що і %windir%, не плутати з %systemdrive%, яка позначає корінь системного розділу). Поза всіма сумнівами це найважливіша папка, розміщена на активному розділі жорсткого диску. У ній зберігаються всі виконувані файли операційної системи, драйвери, модулі та ін. Варто зазначити, що в деяких випадках дана папка може мати іншу назву. Це трапляється в тому випадку, якщо користувач її замінив в процесі встановлення операційної системи. Тому, якщо її назва наперед невідома, в шляхах можна використовувати саме %systemroot%, або %windir%.

Оскільки нас цікавлять папки порівняно великого розміру та папки, які впливають на вивільнення простору системного розділу, то лише на них законцентруємо свою увагу.



%Systemroot%\INF\ – включає в себе всі встановлені на комп’ютері INF-файли, що стосуються наперед визначених стандартних пристройів – як реальних, так і віртуальних. Дані папки приховані.

%Systemroot%\Installer\ – зберігає більшість з пакетів установників Windows, які коли-небудь інсталювали на комп’ютер. Якщо необхідно встановити або перевстановити програмне забезпечення, що інсталюється за допомогою MSI-пакетів, і при цьому інсталяційний диск відсутній (але раніше таке програмне забезпечення уже встановлювалось), то можна спробувати знайти відповідний інсталяційний файл в цій папці. За замовчуванням вона прихована, а пакети, які в ній містяться, мають числові номери (але не реальні назви), тому найпростішим способом визначити приналежність пакета є використання опції вкладки «Додатково» діалогу «Властивості пакета». [5]

%Systemroot%\LastGood\ (або \LastGood.TMP\) – якщо в системі присутня ця папка, то вона може зберігати резервні копії різних файлів, які пов’язані з останнім вдалим запуском системи.

%Systemroot%\System32\DllCache\ – містить копії всіх системних файлів Windows (не тільки бібліотек, а й програм) і є необхідним при пошкодженні або несанкціонованій системою зміні оригінального системного файла для його перевстановлення. Даний каталог стиснутий і прихований.

У папці %Systemroot%\System32\drivers\ знаходяться файли драйверів працюючих пристройів. Проте існує також споріднена папка %Systemroot%\System32\DriverStore\, яка є сховищем драйверів, в тому числі застарілих версій та непрацюючих на даний момент пристройів.

У папці %Systemroot%\system32\config\ містяться файли, які працюють з реєстром:

- файл AppEvent.Evt є журналом подій додатків, що використовується реєстром Windows;
- файл SAM містить розділ реєстру HKLM\SAM;
- файл SecEvent.Evt є журналом подій безпеки, використовуваного реєстром Windows;
- файл SECURITY містить розділ реєстру HKLM\SECURITY;
- файл software містить розділ реєстру HKLM\Software;
- файл SysEvent.Evt є журналом подій системи, що використовується реєстром Windows;
- файл system містить розділ реєстру HKLM\System.

Папка %Systemroot%\system32\DriverStore\ являє собою сховище драйверів пристройів. Тут містяться драйвери пристройів, які працюють на даний момент, і драйвери пристройів, які на даний момент не працюють або тимчасово відсутні у системі і підключаються за необхідності. Особливою уваги заслуговує підкаталог FileRepository, в якому містяться як актуальні драйвери пристройів, так і попередні (застарілі) їх версії. Файли у цьому каталозі призначенні для «відкату», тобто повернення їх у систему у випадку, якщо актуальній драйвер, на відміну від попередньої своєї версії, пристрою працює некоректно [6], [7].

%Systemroot%\SysWOW64\ – аналог папки system32 для 64-розрядних версій операційних систем. Саме до цієї папки звертається в першу чергу операційна система в пошуках штатних утиліт та інтерпретаторів різноманітних скриптов, сценаріїв тощо.

%Systemroot%\WinSxS\ – призначена для збереження резервних копій системних файлів під час здійснення оновлень операційної системи. При чому така папка містить також жорсткі посилання на такі системні файли, які можуть бути розміщені за іншими шляхами у системі і на даний момент навіть можуть використовуватись операційною



системою. Підпапка \Backup використовується для тимчасового збереження файлів розпакованих оновлень під час їх установки і в подальшому може бути використана для відміни таких оновлень [8], [9], [10].

У корені системного розділу розміщені папки \Program Files, \Program Files (x86), призначенні для збереження файлів 64- та 32- розрядного встановленого програмного забезпечення, відповідно, в тому ж числі і UWP (плиткових) застосунків разом із їх резервними копіями попередніх та видалених версій.

За шляхом %SystemDrive%\System Volume Information\ розміщено приховану системну папку. Вона містить інформацію для бази даних служби індексування вмісту, що прискорює пошук файлів, службу тіньового копіювання розділів для резервного копіювання, та бази даних служби відстеження розподілених посилань, які використовуються для відновлення ярликів і посилань.

У корені системного розділу можуть міститись також системні файли великого розміру, такі як hiberfil.sys – пов’язаний з режимом гібернації та pagefile.sys і swapfile.sys – пов’язані із віртуальною дисковою пам’яттю операційної системи (swap-файли або файли підкачки). Такі файли створюються при увімкнених відповідних режимах роботи: режим гібернації – передбачає збереження даних з оперативної пам’яті у файл для можливості повного вимкнення живлення з подальшим швидким відновленням стану системи при увімкненні живлення; віртуальна дискова пам’ять – дозволяє за рахунок файлів розширити об’єм оперативної пам’яті до необхідного у випадку її нестачі для запуску тих або інших процесів.

І нарешті %SystemDrive%\Users\, яка містить дані користувачів, зареєстрованих (в тому числі і локально) у системі. Це і файли налагодження програмного забезпечення для кожного конкретного користувача: кеші, тимчасові файли працюючих пакетів; і документи користувача: текстові файли у вигляді документів різного формату, зображення, відео, звукові файли, файли завантажені з мережі; і синхронізовані папки хмарних сховищ користувачів, якщо такі використовуються.

Це перелік далеко не усіх системних файлів та папок, які містяться на системному розділі, але вищеописані – це ті з них, які займають порівняно великі об’єми і найбільше вливають на розмір вільного простору на системному розділі.

Визначившись із причинами зменшення вільного простору в системі, локалізувавши на розділі системні файли та папки великого розміру, необхідно підібрати способи і, відповідно, інструменти зменшення їх розміру.

У переважній більшості сама операційна система містить такий інструментарій і надає механізми вирішення проблеми. Проте на даному етапі такі інструменти є ще досить далекі від ідеальних, існують переважно у вигляді консольних утиліт операційної системи, недоступні безпосередньо через графічний користувальський інтерфейс, вимагають використання ключів та параметрів у своїй роботі, а, отже, і додаткових знань. Крім того такі консольні утиліти можуть виконуватись в окремих середовищах, для функціонування яких необхідне використання інтерпретатора командного рядка (cmd), або оболонки з розширеними можливостями PowerShell.

Інтерпретатор командного рядка, як основний інструмент для роботи з консольними утилітами операційної системи. Незважаючи на простоту використання графічного інтерфейсу користувача Windows (GUI), інтерфейс командного рядка залишається корисним способом виконання багатьох завдань обслуговування, конфігурації та діагностики операційної системи. Багато важливих інструментів доступні тільки з командного рядка. І хоча термін «bat-файл» може видатись пов’язаним із операційною системою MS-DOS, bat-файли та програмні скрипти залишаються

потужними інструментами, що надають ефективний спосіб інкапсулювати в собі загальні функції управління. Утиліти командного рядка, bat-файли та скрипти, засновані на WSH (Windows Script Host) разом надають повний набір будівельних блоків, з яких можна створити високорівневі утиліти для виконання складних завдань [11].

Основою для відображення роботи інтерпретатора командного рядка є вікно, яке реалізує командний інтерфейс операційної системи у вигляді так званого запрошення командного рядка (Command Prompt), в якому власне і виконуються консольні застосунки. Тому часто таке вікно називають консоллю інтерпретатора командного рядка операційної системи. Таке вікно призначено як і для введення команда вручну, в інтерактивному режимі, так і для відображення перебігу виконання консольних програм та команд. А при виконанні неконсольних команд викликаються вікна графічного інтерфейсу поточної виконуваної команди або програми.

Розширення імені для запуску файлів. У багатьох файлах Windows є так зване розширення імені файлу – суфікс, що слідує після крапки у назві файла і зазвичай складається, як правило, з трьох латинських символів. Це розширення призначено для визначення типу файла. Деякі з розширень позначають виконувані файли програм різного виду, це такі розширення, як .exe, .bat і .vbs.

Для виконання команди не обов'язково вводити ім'я програми повністю, достатньо ввести ім'я без розширення. У випадку введення команди без розширення, операційна система Windows вважає, що це програма, і додає розширення файла самостійно, коли здійснюється пошук програми. Якщо така програма знайдена, то операційна система запускає її. Після цієї програми, як правило, перед запуском вводиться список аргументів, вказаний в командному рядку після імені програми, і програма повинна відповідним чином обробляти ці аргументи [10].

Запуск команд із підвищеними привілеями. Деякі утиліти командного рядка (а іноді програми з графічним інтерфейсом) вимагають для своєї коректної роботи підвищених адміністративних привілеїв (права для користувачів та груп задаються через інструментарій User Account Control). Щоб запустити програму підвищеними привілеями з командного рядка, необхідно запустити її з консолі командного рядка, яка і сама має підвищені права.

У такому режимі необережні дії адміністратора можуть привести до пошкодження Windows. Це також стосується і запуску з підвищеними привілеями графічних (GUI) програм Windows. Тому варто використовувати це лише для спеціальних завдань, які потребують адміністративних привілеїв, і після завершення виконання цих завдань консоль повинна бути закрита.

Bat-файли (пакетні файли). Хоча в операційній системі існує велика кількість потужних інструментів для створення власних корисних програм, іноді достатньо використати мову bat-файлів. Bat-файли (пакетні файли) дозволяють отримати доступ до тисяч готових програм командою рядка, які можна знайти для Windows.

Bat-файл на найпростішому рівні — це просто список команд для вікна Command Prompt (консолі), які розміщені в одному файлі, рядок за рядком, і цьому файлу присвоєно розширення *.bat або *.cmd. Коли вводять ім'я bat-файлу в запрошені командного рядка, Windows шукає файл з таким ім'ям у поточній директорії і потім у шляхах папок, що позначені змінною оточення PATH. Windows обробляє кожен рядок bat-файлу як команду, і запускає їх одну за одною, ніби їх вводять одну за одною вручну. Навіть на такому простому рівні bat-файл суттєво підвищить рівень автоматизації роботи користувача.

Крім цього простого сценарію використання, є кілька команд, які можна використовувати для написання найпростіших «програм» у bat-файлі, щоб виконувати різні дії залежно від того, що було введено користувачем у командному рядку, або залежно від результату виконання попередніх команд. Ці команди значно покращилися з часів MS-DOS, тому написання корисних bat-файлів у Windows набагато простішим. Зокрема, оператори IF та FOR були значно розширені за функціоналом. Можна надавати запрошення введення даних користувача, маніпулювати рядками та іменами файлів а також створювати підпрограми в одному файлі [11].

Використовуючи інтерпретатор командного рядка та пакетні файли, запускаючи їх з підвищеними привілеями, користувач здатен отримати доступ до інструментарію операційної системи, а, отже, маніпулювати вмістом системних папок та системними файлами, регулюючи об'єм вільного простору системного розділу.

Проте не всі команди або інструментарій доступний через інтерпретатор командного рядка, або використання штатного інструментарію може бути суттєво ускладнене. Наприклад, відсутність можливості коректного виконання пакетного файлу із застосунків, створених у різноманітних середовищах програмування. Оскільки такі застосунки можуть бути вже 64-розрядними, вони вимагають відповідного інтерпретатора команд. Тоді як навіть в 64-розрядній операційній системі cmd є 32-розрядним транслятором, розміщеним у системній папці \System32\ і його 64-розрядна версія може бути не вказана за замовчуванням. У таких випадках використовують інші транслятори або середовища, такі як, наприклад, PowerShell.

PowerShell, як один із засобів маніпулювання об'єктами операційної системи. Microsoft розробила нове робоче середовище командного рядка, яке дістало назву Windows PowerShell (WPS). Це середовище встановлене за замовчуванням починаючи з Windows XP. WPS найчастіше виглядає та діє, як вікно командного рядка, але насправді це дуже особлива сутність, яка дає доступ до деяких дуже потужних інструментів програмування.

З одного боку, більшість команд Windows PowerShell (які правильно називати cmdlet) генерують потоки об'єктів, а не текст. Об'єкти — це спроба представлення у комп'ютері речей із реального світу. Вони мають властивості, які описують атрибути змодельованих речей, та методи, які дозволяють керувати речами. Наприклад, об'єкт представляє певний файл на жорсткому диску і має властивості Name (ім'я), Size (розмір) і LastWriteTime (час останнього запису), і методи, наприклад, Delete (видалити), Edit (редагувати) і Open (відкрити). Windows PowerShell працює з об'єктами новим, незвичайним і, зрештою, дуже потужним способом.

При введені команди «dir» у звичайному вікні Command Prompt середовище інтерпретації команд обробить dir і у відповідь поверне блок тексту-лістингу, де будуть імена файлів і папок поточного каталогу. Команда «dir» була спеціально запрограмована для виведення інформації про файли в текстовому вигляді. І це все, що вона може робити.

У системі WPS можна ввести dir, і це також призведе до виведення списку файлів, але неявно відбувається зовсім інше. У WPS — це просто коротке посилання на Get-Childitem cmdlet, який при найпростішому використанні генерує потік об'єктів File. Кожен об'єкт представляє один із файлів у папці і у кожного такого об'єкта є властивості та методи (наприклад, ім'я та розмір). Коли об'єкт (будь-якого виду) потрапляє у вікно-запрошення WPS, система друкує рядок тексту про об'єкт з найбільш важливими його властивостями. Для об'єкта File це включає ім'я файла, розмір та дату створення. Таким чином, коли при введенні команди «dir» WPS генерує потік об'єктів File і він представляється як зручний, розділений табуляціями список файлів.

Кінцевий результат виглядає так, як і в робочому оточенні командного рядка, але все відбувається абстрактнішим способом. Cmdlet не звертає уваги на форматування тексту: він просто генерує набір об'єктів File. І уже вікно WPS перетворить будь-який список об'єктів на читабельний лістинг. Файли, облікові записи користувачів, служби Windows, і т. д. – будь-який об'єкт, який викликає відповідний cmdlet, система WPS перетворює у своїй консолі на зручний лістинг.

Додатково WPS включає повнофункціональну об'єктно-орієнтовану мову програмування, яка має доступ до платформи програмування .NET компанії Microsoft. Це означає, що скрипти WPS можуть виконувати складніше обчислення та обмінюватися інформацією з іншими комп'ютерами та мережевими («хмарними») сервісами.

WPS навіть дозволяє робити складні операції з об'єктами без програмування. Можна використовувати символ каналу «|» для направлення потоків об'єктів від одного cmdlet до іншого, і це дозволяє виконувати дуже складні, специфічні дії з інструментами, які окремо дуже прості та за природою виглядають звичайними.

WPS це повноцінна мова програмування зі змінними, циклами, підпрограмами, визначеннями користувачем об'єктами тощо. Можна використовувати це в запрошені введення команди або у файлах скриптів. Також можна створити ярлики (які називаються псевдонімами, aliases) для команд і скриптів, які найчастіше використовуються, щоб їх можна було простіше використовувати надалі [12,13].

Крім того через PowerShell можна створювати графічні користувальські інтерфейси для написаних скриптів. Для цього можна скористатись як окремим командами так і сторонніми розробками, наприклад PowerShell Studio 2021 і, звичайно, засобами Microsoft Visual Studio [12].

У нашому випадку нас цікавлять можливості PowerShell, які дозволяють обійти обмеження по розрядності при виконанні пакетних файлів із створюваних застосунків, та можливість маніпулювання UWP-застосунками і зверненнями до магазину застосунків Microsoft. Такі застосунки через PowerShell можна вимикати, видаляти, встановлювати, відновлювати, тим самим заощаджуючи вільний простір на системному розділі.

Проте деякі маніпуляції із системними папками вимагають написання пакетних файлів та скриптів, які звертаються до служб та інших засобів операційної системи для коректної очищення цмх системних папок. Такі звертання можна здійснювати через роботу із локальними груповими політиками. Але не всі версії операційної системи мають змогу це зробити, особливо це стосується редакцій початкового рівня, такі як Starter або Home. Іншим способом роботи з такими об'єктами є робота через системний реєстр операційної системи.

Реєстр операційної системи Windows 10 та способи його редагування. Реєстр Windows 10 – основне сховище операційної системи, побудоване у вигляді ієрархічної бази даних. В ній зберігається кожен параметр, кожне налаштування операційної системи, інформація про встановлене на комп'ютері програмне забезпечення та функціонуючі апаратні компоненти, дані про всі облікові записи користувачів. У цій базі даних записані асоціації файлів та системні політики. Кожна програма, що працює в операційній системі, обов'язково звертається до реєстру, наприклад, з метою отримання інформації про систему [14].

Логічно реєстр являє собою набір ієрархічних дерев (кушів), кожне з яких має власне призначення:

- HKCR (HKEY_CLASSES_ROOT) – містить інформацію про відомі типи документів та їх асоціації з програмами для відкриття;

- HKCU (HKEY_CURRENT_USER) – зберігається налаштування параметрів користувача, що увійшов до системи;
- HKLM (HKEY_LOCAL_MACHINE) – записані відомості про апаратну конфігурацію комп’ютера (перелік обладнання, режими його функціонування, версії драйверів);
- HKCC (HKEY_CURRENT_CONFIG) – містить дві гілки, у першій з яких зберігається конфігурація програмного забезпечення, у другій – параметри системи (більшість ключів розділу, доступні для редагування вигляду графічного інтерфейсу Windows);
- HKU (HKEY_USERS) – містяться налаштування кожного облікового запису завантаженої операційної системи [14].

Фізично файли системного реєстру Windows розташовуються у різних підкаталогах системного каталогу. За кожен кущ відповідає окремий файл (за тим же шляхом розміщена і його резервна копія) [14]:

HKEY_LOCAL_MACHINE\SYSTEM:
%systemroot%\system32\config\system
%systemroot%\sysWoW64\config\system
HKEY_LOCAL_MACHINE\SAM:
%systemroot%\system32\config\sam
%systemroot%\ sysWoW64\config\sam
HKEY_LOCAL_MACHINE\SECURITY:
%systemroot%\system32\config\security
%systemroot%\ sysWoW64\config\security
HKEY_LOCAL_MACHINE\SOFTWARE:
%systemroot%\system32\config\software
%systemroot%\ sysWoW64\config\software
HKEY_USERS\UserProfile:
%systemdrive%\Users%\username%
HKEY_USERS.DEFAULT:
%systemroot%\system32\config\default
%systemroot%\ sysWoW64\config\default.

Кожен кущ містить розділи, що в свою чергу містять ключі, яким можна присвоювати ті чи інші значення, в залежності від типу ключів. Передбачено наступні основні типи ключів:

- REG_SZ – рядковий;
- REG_EXPAND_SZ – розширений рядковий;
- REG_BINARY – двійковий, містить необроблені двійкові дані;
- REG_DWORD – числовий, містить ціле число, зазвичай служить як перемикач, де 0 — вимкнено, а 1 — увімкнено;
- REG_LINK – рядковий, вказує шлях до файлу, це символічне посилання у форматі Unicode;
- REG_MULTI_SZ – багаторядковий, визначає масив рядків;
- та інші, менш поширені типи [15].

Працювати з реєстром, а фактично його редагувати (вносити зміни), можна кількома способами:

- через редактор реєстру (інструмент схожий до провідника, замість папок та файлів – кущі, розділи та ключі), що викликається командами regedit або regedit32 з командного рядка;

- за допомогою .reg-файлів (текстових файлів з асоційованим з редактором реєстру розширенням, з наперед визначенім синтаксисом введення куща, розділу, ключа та його типу і значенням, вони містять відповідний заголовок та один і більше рядків із записами про зміни параметрів реєстру);
- за допомогою команди командного рядка REG, яка допускає роботу в пакетних файлах, а, отже, придатна для автоматизації процесів [16].

Rundll32, як інструмент виклику функцій системних бібліотек. У процесі застосування визначених методів та засобів час від часу може виникати необхідність використання вмонтованих в операційну систему штатних інструментів. І хоча звичайний користувач може скористатись графічним інтерфейсом операційної системи для виклику таких інструментів, такий підхід не дозволить автоматизувати цей процес, а, отже, і не дозволить використати його при проектуванні програмних систем. Тому часто для опису виклику таких інструментів у пакетних файлах та різноманітних скриптах використовують виклики функцій системних бібліотек за допомогою утиліти rundll32.

Така утиліта завантажує та виконує 32-розрядні бібліотеки динамічного компонування (DLL). Настроювання параметрів для rundll32 відсутні. Довідкові відомості надаються для конкретної бібліотеки DLL, яка виконується за допомогою команди rundll32 з підвищеними привілеями (від імені адміністратора). Rundll32 може викликати лише функції з бібліотеки DLL, які були явно написані для виклику за допомогою rundll32. [17]

У 64-бітних версіях операційних систем сімейства Windows присутні 2 варіанти програми rundll32.exe:

64-бітна версія, розташована в %SystemRoot%\System32\;
32-бітна версія, розташована в %SystemRoot%\SysWOW64\.

Для запуску програми rundll32 використовується синтаксис командного рядка такого вигляду:

rundll32.exe <бібліотека_dll>,< функція> <необов'язкові параметри_функції>.

Наприклад, для виведення вікна регіональних налаштувань операційної системи через командний рядок: rundll32.exe shell32.dll,Control_RunDLL intl.cpl,,0

Огляд існуючих розробок для очищення та оптимізації системного розділу Windows 10. Існує велика кількість різноманітних програмних продуктів, як комерційних, так і вільних, для реалізації методів оптимізації системного розділу, шляхом його очищення (прибирання). За досить тривалий період різними виробниками було розроблено цілий ряд таких готових рішень, проте жодне із них повністю не охоплює усі можливі варіанти очищення системного розділу. Проте існує цілий ряд програмних продуктів, які досить ефективно реалізують частину таких методів.

Для порівняння було розглянуто штатну консольну утиліту DISM [18] та три безкоштовних, проте досить ефективних програмних засоби:

- PatchCleaner [4];
- Wise Disk Cleaner X [19];
- DISM++ [20].

Деякі з цих інструментів вузько спеціалізовані і поки єдині, які зустрічаються серед безкоштовних розробок (наприклад, PatchCleaner), а інші мають розширеній функціонал, який може стосуватись не тільки для очищення системного розділу: Wise Disk Cleaner X та DISM++. Деякий функціонал не може забезпечити жодна із розглянутих програм, а лише штатні вузькоспеціалізовані системні утиліти, або взагалі

лише послідовність дій користувача від імені адміністратора. Для порівняльного аналізу таких розробок було сформовано порівняльну таблицю (табл. 1).

Таблиця 1

Порівняння програмного забезпечення по досліджуваному функціоналу («+» – наявна функція, «-» – відсутня, «+/-» – частково присутня)

Каталоги для очистки, або інші методи вивільнення простору	PatchCleaner	Wise Disk Cleaner	DISM	DISM++
%systemroot%\installer\	+	-	-	-
%systemroot%\installer\\$PatchCache\$	-	-	-	-
%systemroot%\installer*.tmp-	-	-	-	-
%systemroot%\winSxS\	-	+/-	+	+
%systemroot%\winSxS\Backup	-	+	-	+
Стиснення системних файлів	-	-	-	-
%systemroot%\system32\DriverStore	-	-	-	+
%systemroot%\system32\DriverStore\ \FileRepository	-	-	-	+
Керування гібернацією	-	-	-	-
Керування віртуальною пам'яттю	-	-	-	-
Видалення тимчасових файлів	-	+	-	+
Керування UWP	-	-	-	+
Видалення файлів пакетів оновлень	-	+/-	+	+
System Volume Information	-	-	-	-

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Отже, у роботі проаналізовано основні файлові об'єкти Windows 10, які найсуттєвіше впливають на розмір вільного простору системного розділу. Розглянуто засоби, за допомогою яких можна реалізувати комплексне обслуговування системного розділу на предмет його очищення (прибирання) з метою вивільнення вільного простору на ньому. Це можна здійснювати як в ручному режимі, що може зайняти досить тривалий час, або скористатись програмними продуктами сторонніх розробників, такими як Patch Cleaner, Driver Store Explorer, DISM++, Wise Disk Cleaner та іншими, які пропонують окремі виробники. Оскільки деякий функціонал не може забезпечити жодна з існуючих розглянутих програм, а лише штатні вузькоспеціалізовані системні утиліти, або взагалі лише послідовність дій користувача від імені адміністратора, то на практиці, варто поєднувати у процесі оптимізації системного розділу декілька програмних засобів, які за своїми функціональними можливостями перекривають один одного та дозволяють максимально збільшити об'єм вивільненого простору на системному розділі та реалізувати налаштування операційної системи, що теж вивільняє простір, або не допускає його зменшення в майбутньому в процесі експлуатації системи. Такі маніпуляції досить тривалі, деякі з них вимагають дотримання послідовності їх виконання, отримання доступу до суміжних засобів та об'єктів, пошуку сторонніх засобів тощо. Іншим шляхом оптимізації системного розділу могло б бути створення програмної розробки, яка дозволила б автоматизувати процеси очищення та оптимізації, таким чином прискорити їх виконання та передбачити можливі нестандартні ситуації, які можуть виникнути в процесі оптимізації.



СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1 Булатецький, В. В., Булатецька, Л. В., Пруц, Г. С. Методи та засоби вивільнення простору системного розділу ОС Microsoft Windows 10. *Ком'ютерно-інтегровані технології: освіта, наука, виробництво*, (32), 85–89.
- 2 *Windows and GPT FAQ*. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/windows-and-gpt-faq>
- 3 *What partitions does windows automatically create during instalation* - Windows 10 Help Forums. Windows 10 Help Forums. <https://www.tenforums.com/installation-upgrade/34336-what-partitions-does-windows-automatically-create-during-instalation.html>
- 4 Dunteemann, J., & Degunking, B. J. (2005). Windows. (2-е вид.). Paraglyph Press.
- 5 *4 Ways to Safely Delete Unused MSI and MSP Files from Windows Installer Folder*. Raymond.CC Blog. <https://www.raymond.cc/blog/safely-delete-unused-msi-and-mst-files-from-windows-installer-folder/>
- 6 Bashlak, D. (2017, 26 квітня). *Как очистить папку FileRepository в DriverStore*. remontka.pro. <https://remontka.pro/driverstore-filerepository-folder-windows/>
- 7 *Что за папка Driverstore и можно ли её удалить в Windows*. Настрой всё своё домашнее оборудование. <https://nastroyvse.ru/opersys/win/papka-driverstore-v-windows.html>
- 8 Bashlak, D. (2016, 11 лютого). *Очистка папки WinSxS в Windows 10, 8 и Windows 7*. remontka.pro. <https://remontka.pro/winsxs-windows/>
- 9 Huculak, M. (2019, 28 жовтня). *Is your WinSxS folder getting too big? Here's how to clean it on Windows 10*. Windows Central. <https://www.windowscentral.com/how-reclaim-space-reducing-size-winsxs-folder-windows-10>
- 10 McLean, I., Thomas, O. (2010). *MCTS Self-Paced Training Kit (Exam 70-680): Configuring Windows 7*. Microsoft Press.
- 11 *Инструменты командной строки и автоматизации* | windows | adminstuff. microsin.net - заметки радиолюбителя. <http://microsin.net/adminstuff/windows/command-line-and-automation-tools.html>
- 12 *What is PowerShell?* – PowerShell. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/powershell/scripting/overview?view=powershell-7.1>
- 13 *How to Create a GUI for PowerShell Scripts?* – TheITBros. TheITBros. <https://theitbros.com/powershell-gui-for-scripts>
- 14 *Реестр Windows 10 в подробностях*. (б. д.). О проекте Windowsprofi.ru и настройке операционных систем. <https://windowsprofi.ru/win10/reestr-windows-10.html>
- 15 *Windows registry for advanced users* - Windows Server. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/troubleshoot/windows-server/performance/windows-registry-advanced-users>
- 16 *Reg commands*. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/reg>
- 17 Rundll32. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/ru-ru/windows-server/administration/windows-commands/rundll32>
- 18 *Deployment Image Servicing and Management (DISM) Best Practices*. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/deployment-image-servicing-and-management--dism--best-practices?view=windows-11>
- 19 *WiseCleaner Software - Optimize, Clean and Speed Up Your Windows PC*. WiseCleaner Software - Optimize, Clean and Speed Up Your Windows PC. <https://www.wisecleaner.com/>
- 20 *Beware of Dism++ with Windows customizations*. Born's Tech and Windows World. <https://borncity.com/win/2019/08/16/beware-of-dism-with-windows-customizations/>

**Vitalii V. Bulatetskyi**

PhD, Associate Professor, Associate Professor at the Department of Computer Science and Cybersecurity
Lesya Ukrainka Volyn National University, Lutsk, Ukraine
ORCID ID: 0000-0002-9883-4550
Bulatetsky.Vitaly@vnu.edu.ua

Lesia V. Bulatetska

PhD, Associate Professor, Associate Professor at the Department of Computer Science and Cybersecurity
Lesya Ukrainka Volyn National University, Lutsk, Ukraine
ORCID ID: 0000-0002-7202-826X
Bulatetska.Lesya@vnu.edu.ua

Tetiana O. Hryshanovych

PhD, Associate Professor, Head of the Department of Computer Science and Cybersecurity
Lesya Ukrainka Volyn National University, Lutsk, Ukraine
ORCID ID: 0000-0002-3595-6964
Hryshanovych.Tatiana@vnu.edu.ua

ANALYSIS OF OS WINDOWS 10 FILE OBJECTS FOR SYSTEM PARTITION SPACE CLEANING AND OPTIMIZATION

Abstract. The paper analyzes the causes and ways of solving the problem of free space lack on the system partition during the OS Windows exploitation. The peculiarities of the system partition organization during the installation of common Microsoft operating systems are analyzed. The main system file objects, paths to them, their purpose and role in the use of system partition space and in the functioning of the operating system are considered. Represented the list of paths to the files of the folders that occupy the largest volumes on the system partition. These folders correct cleaning or configuring may save a lot of space on the system partition. The main tools of the operating system for the cleaning methods implementation are considered: command line interpreter, PowerShell environment, system registry editing tools, and system library function calling tools. Particular attention is paid to batch files as the simplest and most effective means of implementing automated use of system utilities, taking into account the peculiarities of the command line interpreter. One of the ways to manipulate objects with commands is to use the PowerShell environment. This environment is able not to process commands, but to generate streams of objects related to this command and form their processed representation using both command and graphical interface. Described the logical and physical structure of the registry as one of the tools for managing Windows settings. Was made the comparative analysis of some software products, such as the system console utility for servicing deployment and management images, as well as a number of free software. It has generated the list of functions used for the system partition optimization and cleaning from excessive data. Based on the generated list, the efficiency of using the considered software tools for cleaning the system partition is analyzed.

Keywords: operating system, system partition, optimization, excessive data, system registry, batch file, PowerShell, command line interpreter.

REFERENCES (TRANSLATED AND TRANSLITERATED)

- 1 V. V. Bulatetsky, L. V. Bulatetska, G. S. Prutz, Methods and tools for clearing the space of the system partition of microsoft windows 10., *Computer-integrated technologies: education, science, production* (32) 85–59.
- 2 *Windows and GPT FAQ*. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/windows-and-gpt-faq>
- 3 *What partitions does windows automatically create during instalation* - Windows 10 Help Forums. Windows 10 Help Forums. <https://www.tenforums.com/installation-upgrade/34336-what-partitions-does-windows-automatically-create-during-instalation.html>
- 4 Duntemann, J., Degunking, B. J. (2005). Windows. (2nd ed.). Paraglyph Press.

- 5 4 Ways to Safely Delete Unused MSI and MSP Files from Windows Installer Folder. Raymond.CC Blog. <https://www.raymond.cc/blog/safely-delete-unused-msi-and-mst-files-from-windows-installer-folder/>
- 6 Bashlak, D. (2017, 26 April). Kak ochistit' papku FileRepository & DriverStore. remontka.pro. <https://remontka.pro/driverstore-filerepository-folder-windows/>
- 7 Chto za papka Driverstore i mozno li yego udalit' v Windows. Настрой всё своё домашнее оборудование. <https://nastroyvse.ru/opersys/win/papka-driverstore-v-windows.html>
- 8 Bashlak, D. (2016, 11 February). Ochistka papki WinSxS v Windows 10, 8 i Windows 7. remontka.pro. <https://remontka.pro/winsxs-windows/>
- 9 Huculak, M. (2019, 28 October). Is your WinSxS folder getting too big? Here's how to clean it on Windows 10. Windows Central. <https://www.windowscentral.com/how-reclaim-space-reducing-size-winsxs-folder-windows-10>
- 10 McLean, I., Thomas, O. (2010). *MCTS Self-Paced Training Kit (Exam 70-680): Configuring Windows 7*. Microsoft Press.
- 11 Instrumenty komandnoy stroki i avtomatizatsii / windows / adminstuff. (б. д.). microsin.net - заметки radiolyubatelya. <http://microsin.net/adminstuff/windows/command-line-and-automation-tools.html>
- 12 What is PowerShell? - PowerShell. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/powershell/scripting/overview?view=powershell-7.1>
- 13 How to Create a GUI for PowerShell Scripts? – TheITBros. TheITBros. <https://theitbros.com/powershell-gui-for-scripts>
- 14 Reystr Windows 10 v podrobnostyakh. О проекте Windowsprofi.ru и настройке операционных систем. <https://windowsprofi.ru/win10/reestr-windows-10.html>
- 15 Windows registry for advanced users - Windows Server. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/troubleshoot/windows-server/performance/windows-registry-advanced-users>
- 16 Reg commands. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/reg>
- 17 Rundll32. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/ru-ru/windows-server/administration/windows-commands/rundll32>
- 18 Deployment Image Servicing and Management (DISM) Best Practices. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/deployment-image-servicing-and-management--dism--best-practices?view=windows-11>
- 19 WiseCleaner Software - Optimize, Clean and Speed Up Your Windows PC. WiseCleaner Software - Optimize, Clean and Speed Up Your Windows PC. <https://www.wisecleaner.com/>
- 20 Beware of Dism++ with Windows customizations. Born's Tech and Windows World. <https://borncity.com/win/2019/08/16/beware-of-dism-with-windows-customizations/>



This work is licensed under Creative Commons Attribution-noncommercial-sharealike 4.0 International License.