Оглавление

[Оглавление 1](#_Toc1)

[Введение 2](#_Toc2)

[Раздел 1. Принципы работы вирусного ПО и цели проекта. 4](#_Toc3)

[Компоненты 5](#_Toc4)

[v Раздел 1. Обработка ошибок. 5](#_Toc5)

[v Раздел 2. Компонент Шифровки. 5](#_Toc6)

[v Раздел 3. Файловый Менеджер. 5](#_Toc7)

[v Раздел 4. Мастер Управления Автозагрузкой. 6](#_Toc8)

[v Раздел 5. Компонент Перезагрузки. 6](#_Toc9)

[v Раздел 6. Интерфейс. 7](#_Toc10)

[v Раздел 7. Логика кнопок интерфейса. 8](#_Toc11)

[v Раздел 8. Очистка папки Temp. 8](#_Toc12)

[v Раздел 9. Запуск От Имени Администратора(Run). 8](#_Toc13)

[Заключение. 9](#_Toc14)

[v Как установить программу. 9](#_Toc15)

[v Как пользоваться программой. 9](#_Toc16)

[v Итог. 10](#_Toc17)

Введение

Компьютеры были изобретены с целью хранения, обработки и передачи данных, компьютеры развивались и приобретали больше свойств, но эти задачи оставались главными, первые компьютерные вирусы были просты, форматировали диск, нанося большой вред пользователю или корпорациям, вирусы совершенствовались, вредя в основном системе, ведь не выгодно просто удалить данные – выгодней воровать их, крадя личные данные или используя ресурсы ПК пользователя как вздумается.

В данный момент проблема с вирусным ПО, достаточно сильно распространена на ОС Windows из-за её большой популярности, а также халатности пользователей. Несмотря на принятые во многих странах законы о борьбе с компьютерными преступлениями и разработку специальных программных средств защиты от вирусов, количество новых программных вирусов постоянно растет. Это требует от пользователя ПК знаний о принципе работы вирусов, способах заражения вирусами и защиты от них. Но даже профессиональные пользователи сталкиваются с проблемами, ведь восстанавливать систему вручную достаточно тяжко.

Современные антивирусы могут обеспечивать защиту ОС Windows в основном от старых или однотипных вирусов, существуют антивирусы более сложные которые стараются обеспечить защиту ОС Windows постоянным сканированием реестра, системных каталогов и запущенных процессов, что, следовательно, приводит к большой нагрузке на ЦП, ОЗУ, и на Диск. Также 99% Антивирусов имеют закрытый исходный код, а также скудный выбор настроек.

«Антивирус Монтировка» даёт право выбора пользователю, даже если в программе нет выбора у программы открытый исходный код, чтобы пользователи могли изменять программу так как им это нужно, а также создавать модификации или участвовать в разработке программы.

Цель проекта создать «Мультитул» с набором утилит для лечения ОС Windows, а также защитить ПК, от возможных воздействий вирусного ПО, на ОС Windows и ПК. Многие вирусы могут скрывать себя в стандартном проводнике, добавляют в автозагрузку, используют ресурсы ПК, или вводят ограничения в систему чтобы усложнить их удаление.

Антивирус «Монтировка» содержит:

* простенький файловый менеджер, с возможностью открывать каталоги; файлы можно копировать, вставлять, переименовывать, удалять, запускать их через нестандартное ПО или от имени администратора.
* Мастер Управления Автозагрузки, в котором содержится вся автозагрузка в каталоге пользователя, запланированная через планировщик задач, а также системную загрузку в разделе Winlogon в реестре. Каждую из них можно удалить, или скопировать путь, а также его значение при наличии.
* Также программа может попытаться выполнить перезагрузку ОС, если её параметры были удалены из меню пуск или при сочетании Ctrl + Alt + Del.
* Компонент LoadProtection, который сканирует процессы на присутствие слов-триггеров в названии процесса, а также их нагрузку на CPU и RAM. В случае, если процесс превышает заданный лимит нагрузки и не находится в базе исключений, то программа замораживает процесс и уведомляет пользователя об этом; пользователь может разморозить процесс или отказаться, если разморозил, программа спросит добавлять ли процесс в список исключений.

Раздел 1. Принципы работы вирусного ПО и цели проекта.

Компьютерные вирусы произошли по аналогии биологических вирусов, главной особенностью является возможность самокопирования и распространения. Цель вируса принести вред файлам, ОС, иногда и самому ПК пользователя. Многие вирусы умеют обороняться от вирусного ПО, и стандартных защит ОС Windows, так как обновления безопасности windows выходят слишком редко, то практически все современные вирусы умеют обходить стандартную защиту Windows. Одни из продвинутых вирусов клонируют себя, и работают параллельно, восстанавливая процесс, следовательно, надо разработать компонент, который получает список всех запущенных процессов и сканирует их на слова-триггеры в названии процесса, а также нагрузку на компоненты ПК.

Другие вирусы прячут себя, чаще всего это выполняется через включения атрибута «скрытый» или «системно важный». Некоторые могут вмешаться в системный проводник и тем самым как-то интегрировать себя в скрытом виде даже при включённых параметрах отображения скрытых файлов, следовательно, в таких случаях может помочь сторонний файловый менеджер, кроме того, можно использовать популярные программы, такие, как Total Commander или использовать архиваторы WinRAR/7-Zip, но две последние программы не совсем на это рассчитаны. Более того, они могут быть заблокированы, ведь эти программы рассчитаны на устойчивую среду и не имеют защиты от вредоносного ПО.

Если у вируса нет цели навредить системе сразу, то вирус добавит себя в автозагрузку. В ОС Windows нет полного и полноценного меню для управления автозагрузкой ОС. Тем более, автозагрузок в Windows очень много: пользовательская, системная в реестре, реестр, планировщик задач и сервисы. Следовательно, надо сделать программу, которая просканирует всю автозагрузку и покажет её, и позволит изменить её и удалить. Некоторые вирусы могут заблокировать перезагрузку, пока вирус не внедрит себя в систему окончательно или принудительно не давая выключить систему через меню пуск. Такое возможно при изменении групповых политик. Можно использовать API Windows, системный вызов shutdown, или запуск х.bat файла. Девяносто процентов вирусов блокируют работу диспетчера задач, 10% блокируют работу ProcessHaker, и скорее всего ни один вирус не может блокировать доступ к спискам процессов, а это означает, что можно создать свой аналог диспетчера задач.

Компоненты

* Раздел 1. Обработка ошибок.

Так как в программе такого типа, ошибки не то что возможны, но и неизбежны. Следовательно, требуется сделать обработку ошибок, чтобы при ошибке в программе, программа логировала ошибку в лог файл, а после продолжала работу принудительно. Это позволит программе не вылетать при ошибке.

* Раздел 2. Компонент Шифровки.

Так как данная программа должна быть неуязвима, все параметры, которые как-то могут повлиять на успех обнаружений вирусного ПО, должны шифроваться. Шифровщик работает по схожему принципу шифра Цезаря, но только буквы сдвигаются по алфавиту не на 2, а на указанное число в переменной. С числами все немного по-другому – числа умножаются при шифровании, делятся при дешифровке. Из минусов можно отметить, что нельзя использовать число 0 и числа кратные 26 и 33.

* Раздел 3. Файловый Менеджер.

Файловый менеджер должен быть универсален, и интуитивно понятен, и самое главное независим от чего либо, а также защищён от вирусов.

Первое, как можно защитить программу, это сделать так, чтобы заголовок окна не имел какой-либо конкретный текст, а набор символов, чтобы вирус не мог, по ключевым словам, определить окно и закрыть его.

Второе, программа должна уметь обрабатывать ошибки, стараться в любом случае продолжать работу, параметры в файлах настройки должны быть зашифрованы, что бы защитить их от изменения.

В окне программы есть 6 кнопок:

* «<-» - назад
* «->» - вперёд
* «|» - вверх
* «|->» - обновить
* «+» - добавить вкладку
* «-» - закрыть вкладку

Ниже есть текстовое поле, в котором указывается текущий каталог, данное поле можно редактировать, чтобы быстро переходить по каталогам.

Ещё ниже есть панель с вкладками, между которыми можно переключатся.

Ещё ниже есть меню выбора диска, при выборе одного из дисков в текущей вкладке открывается каталог [БУКВА\_ДИСКА]:\\.

Остальное пространство программы занимает таблица с колонками:

* №
* Имя файла
* Размер файла
* Дата создания

При двойном нажатии на элемент, если это каталог программа перейдёт в него, если файл, то откроет его в программе по умолчанию.

По ПКМ есть пункты:

* Копировать
* Вставить
* Переименовать
* Удалить
* Свойства
* Запустить от имени Администратора

В самом верху программы есть стандартная панель, с пунктами:

* Открыть – запускает утилиту для запуска файлов (аналог Win + R)
* Настройки – позволяет выключить оповещение о каждом действии
* Открыть с помощью – позволяет открыть указанный файл, через указанную программу
* О программе
* Раздел 4. Мастер Управления Автозагрузкой.

У программы также в заголовке окна генерируется случайный набор букв, и также имеют таблицу с элементами автозагрузки. Данная программа пока что умеет сканировать автозагрузку в папке пользователя, планировщике задач (не полностью), а также системные параметры в usernit и shell. Программа умеет удалять автозагрузки, а также можно изменять параметры реестра. Кроме того, можно скопировать путь к файлу или значение в зависимости от пункта.

* Раздел 5. Компонент Перезагрузки.

С компонентом «Перезагрузки» всё намного проще в плане интерфейса, его попросту не будет! Но нужно выбрать способ, как программа будет перезагружать ОС, самый простой способ – это создать х.bat файл с содержимым: «shutdown /r/ t {second\_to\_restart}», а после запустить его. Существует и другой способ использовать API Windows, в котором есть более обширный выбор параметров, а именно выполнение перезагрузки принудительно или добровольно.

Последний способ выполнить перезагрузку: через библиотеки os или subprocess сделать системный вызов команды shutdown по аналогичному принципу работы с запуском х.bat файла, но данные методы могут быть недоступны из-за блокировок системных вызовов вирусами. Также из-за подобных блокировок, следует сделать обработку ошибок, таким образом, чтобы программа при неудачной попытке перезагрузки указного способа пробовала все остальные.

* Раздел 6. Интерфейс.

Графический интерфейс должен быть простым и интуитивно понятным, а также не зависеть от файлов, то есть без иконок. Весь Интерфейс делится на:

* Иконка с меню в трейе
* Окно Анлокера
* Настройки
* Выход из программы

Иконка появляется после запуска программы в трее, выглядит она, как красный смайлик на синем квадрате, при нажатии ПКМ открывается меню, со следующими пунктами:

* Открыть Монтировка Анлокер – открывает окно Анлокера
* Запустить LoadProtection – запускает компонент LoadProtection
* Запустить Очистку Temp – очищает Папку Temp в каталоге пользователя
* Запустить От Имени Админа – открывает компонент Run
* О Программе – Версия программы и краткое описание
* Настройки
* Выход

Интерфейс Анлокера представляет собой окно с текстом и набором кнопок которые открывают соответствующие файлы или компоненты. Интерфейс Настроек состоит из окна с галочками и полями ввода и кнопкой «Применить». При нажатии кнопки "Применить" программа считывает параметры на галочках и в полях, после шифрует их через компонент шифровки, а затем сохраняет их в соответствующих txt-файлах. Интерфейс после нажатия пункта "Выход", выглядит в виде капчи, после решения которой, программа завершит свою работу. В случае её неправильного решения программа не выполнит "Выход".

* Раздел 7. Логика кнопок интерфейса.

При нажатии кнопки программы, которая написана на Python3, вызывается соответствующая функция. При нажатии кнопки для запуска файла, в функцию передаются переменные с именем и каталогом исполняемого файла и его указанный размер в байтах, после чего программа проверяет существование указанного файла. Когда она получает его размер в байтах, то проверяет его на совпадение с указным весом, если файл не весит, сколько указано в переменной, то программа не запустит его, если же размер совпадёт, то начнётся запуск программы.

* Раздел 8. Очистка папки Temp.

Данный компонент будет производить банальную очистку Temp, в надежде, что там было вирусное ПО. Логика проста, программа получает имя пользователя, получает список всех каталогов и файлов в папке C:\Users\{USER\_NAME}\AppData\Local\Temp\ и удаляет там все файлы, после чего создаёт лог файл, в котором записываются все успешно удалённые файлы и каталоги, а затем те, которые не получилось удалить.

* Раздел 9. Запуск От Имени Администратора(Run).

Данный компонент имеет небольшое окно с текстовым полем и 6 кнопками:

* CMD
* REGEDIT
* GPEDIT
* POWERSHELL
* EXPLORER
* OK

В текстовое поле необходимо прописывать путь к файлу, который пользователь хочет запустить от имени администратора. При нажатии всех кнопок, кроме «ОК», в текстовое поле вставляется путь к указанной программе, после нажатия кнопки «ОК» программа запускает файл от имени администратора. Если были выбраны системные файлы gpedit.msc или regedit.exe, то программа их запустит в обычном режиме, иначе возникнет ошибка, так как эти программы не могут быть запущены не от имени администратора, и потребует подтверждения через окно UAC.

Заключение.

* Как установить программу.

Для установки программы требуется запустить установщик, после чего, если у вас на ПУ отсутствует Python3, тогда следует нажать кнопку «Установить Python 3.12» с нужной вам разрядностью, в установке обязательно требуется поставить галочку на пункте «add python to path». После установки Python потребуется установить библиотеки – делается это через нажатие кнопки «Установить библиотеки», откроется консоль с установкой библиотек. Как установка библиотек завершится, требуется нажать кнопку «Начать Установку» – откроется окно с инструкцией, как скомпилировать программу:

1. отредактируйте файл confing.py, (выберите желательные параметры);
2. откройте командную строку;
3. скопируйте путь к каталогу в котором запущен установщик вставьте в консоль: 'cd [путь\_к\_каталогу]', нажмите Enter;
4. если этот каталог расположен на любом диске кроме диска C: введите в консоль следующее: '[буква\_диска]:', Enter
5. Введите в консоль: 'pyinstaller -F -w -i 'icon\LP\_icon.ico' LP.py', после завершения отредактируйте код в файле confing.py, а именно переменную LP\_size – замените её значение на вес файла LP.exe в байтах (скопируйте значение на против текста 'Размер');
6. после сохранения, введите в консоль: 'pyinstaller -F -w -i 'icon\T\_icon.ico' T.py', после окончания нажмите 'Завершить Установку'.

После нажатия кнопки программа спросит, в какой каталог установить программу, после выбора каталога программа:

1. скопирует скомпилированные файлы в указанный каталог;
2. добавит программу в автозагрузку реестра в параметре usrenit;
3. создаст ярлык на рабочем столе.

После успешной установки Python и библиотеки можно удалить, итоговый вес будет составлять не больше 70 MB.

* Как пользоваться программой.

После запуска в трее на панели задач появляется иконка красного смайлик на синем квадрате, по ПКМ откроется меню с пунктами ([Раздел 6. Интерфейс.](#_v_Раздел_6)). Нажатием на пункт «Открыть Монтировка Анлокер», откроется окно, где можно запустить другие утилиты, как сторонние, так и само писанные. Интерфейс программы интуитивно понятен и не требует подробного описания.

* Итог.

В результате данного проекта я убедился, что всё рано или поздно эволюционирует. Вирусное ПО остаётся и скорее всего будет выгодным и прибыльным: во-первых, из-за халатности пользователей; во-вторых, из-за халатности корпораций. Людей жаждущих лёгких денег на подобном, не пугает даже закон. Они становятся всё хитрей и хитрей. Не редкость услышать анекдот о какой-нибудь антивирусной корпорации, что на первом этаже здания пишут антивирусы, а на 2-ом этаже пишут вирусы. Целью данного проекта является попытка спасти оборудование и программы на компьютере или упростить удаление вирусов с ПК! Пусть некоторые его методы могут выглядеть и работать аналогично некоторым вирусам, но программа создана по принципу: «**Не можешь победить — возглавь**» (Николо Макиавелли). В итоге у меня получилась программа, которая может проводить базовые операции с файлами, реестром и автозагрузкой. К тому же программа может выполнить перезагрузку системы, запустить файл от имени администратора и очистить папку Temp. А компонент LoadProtection способен сканировать процессы на перегрузку составляющих ПК и имени процесса.