

*Зайченко Е. А.**ст. преп.,**Шанцов А. В.**студент 3-го курса,**Капытков А. С.**студент 3-го курса**Белорусско-Российский университет, г. Могилев, Беларусь*

ИСПОЛЬЗОВАНИЕ МЕТАДАННЫХ ДЛЯ ЗАЩИТЫ ФАЙЛОВ

Системы электронного документооборота (ЭДО) охватывают все стороны современного общества. Большинство современных устройств, получающие данные из сети Интернет, сохраняют копию просматриваемого файла в своей временной памяти (cache, cookie). Данное действие является неотъемлемой частью функционирования, используемого устройством программного обеспечения. Говоря о защищенном документообороте, часто подразумевают защиту той информации, которую в себе несут файлы документов. Однако нельзя забывать, что с любым файлом связана дополнительная информация о его свойствах — метаданные, которая может представлять ценность для злоумышленников.

Можно выделить следующие категории метаданных.

Описательные метаданные — общие сведения о файле документа, такие как название, автор, реферат, ключевые слова, которые помогают поисковым системам понимать, о чем речь в документе. С их помощью можно быстро найти необходимую информацию, а также увеличить шансы на попадание в топ поисковой выдачи.

Административные метаданные — данные о способах сохранения, дате создания и изменения, управления правами, программном обеспечении, форматах файлов, проверках целостности файлов.

Структурные метаданные — сведения о связи данных друг с другом для облегчения навигации, например, как упорядочиваются страницы для формирования глав или структура гиперссылок в документе [1].

Операционные системы обеспечивают защиту файлов, присоединяя к файлу информационную структуру, хранящую данные о том, кто является владельцем файла, разрешен ли к файлу доступ других пользователей и групп, какие именно виды доступа разрешены (только чтение или изменение файла) [2]. Набор операций над файлом зависит от его содержимого, например, для документов, может быть запрещено изменение основного текста документа, но разрешено добавление в документ примечаний, в электронных таблицах защита может распространяться на ячейки и листы рабочей книги. Универсальным средством защиты от несанкционированного доступа является установка пароля, который также можно отнести к метаданным, но хранящимся в зашифрованном виде.

Все перечисленные метаданные могут быть прочитаны средствами интерфейса прикладного программирования, предоставляемого операционной системой, а некоторые метаданные могут быть изменены.

Важным средством защиты файлов является создание резервной копии, особенно с помощью архивирования. Программы архиваторы позволяют не только кодировать файлы для уменьшения их объема, но и дают дополнительную возможность шифрования файлов и установки пароля на архив. Фактором оптимальной стратегии защиты является выбор данных, подлежащих архивированию и периодичность [3].

Разумеется, пользователь может выполнять защиту файлов от несанкционированного доступа как с помощью приложений, в которых создаются файлы, так и средствами операционной системы, однако будет удобно выполнять защиту и архивирование файлов с помощью одного приложения.

Разработано программное обеспечение на языке высокого уровня Python, которое посредством работы с метаданными выполняет следующие функции:

- управление доступом к файлам и обеспечение безопасности. Имеется возможность задавать права доступа к выбранным файлам и группам файлов;

- систематизация файлов. После чтения метаданных выбранных пользователем файлов предоставляется возможность выбрать свойство, по которому файлы будут сгруппированы. Набор критериев обширен — от времени создания или изменения файла до атрибутов защиты или автора;

- поиск файлов: Программа может искать файлы по практически любым критериям, взятыми из метаданных, таким как автор и т. д.;

- архивация: Программа архивирует файлы, которые не использовались в течение определенного времени, освобождая тем самым пространство на жестком диске. После чтения метаданных нужных пользователю файлов, файлы, которые длительное время не используются, будет предложено заархивировать. В случае, если пользователь хочет заархивировать файлы по другому атрибуту, ему будет предоставлена эта возможность. В качестве примера, пользователь сам может настроить нужный ему промежуток времени, в течение которого файлы не используются, или же выбрать файлы определенного расширения, или сочетать несколько атрибутов при настройке архивации. Предоставлен выбор типа конечного архива из списка: zip, tar, gztar, bztar, xztar. Любой из созданных архивов можно защитить паролем.

Программа разработана на основе двух категорий библиотек языка Python. Первая из них — stat, предоставляет возможность работы с метаданными файлов, включая их чтение, изменение, удаление [4].

Вторая категория предоставляет для работы с архивированием файлов. Для поддержки широкого спектра форматов архивов используются библиотеки zlib, bz2, lzma.

Графический интерфейс пользователя разработан с помощью модуля tkinter и обеспечивает удобный доступ ко всем функциям, предоставляемым программой. Интерфейс интуитивно понятен для любой категории пользователей.

Метаданные представляют собой ценный ресурс, позволяющий с помощью специального программного обеспечения повысить эффективность и безопасность электронного документооборота. Разработанное программное обеспечение даст возможность пользователям защищать свои файлы, оптимизировать систематизацию файлов, выполнять их поиск по широкому набору критериев и обеспечивать надежное хранение за счет ведения архивов.

Список источников

1. Библиотека для открытой науки. Метаданные [Электронный ресурс] // Библиотека для открытой науки : [сайт]. [2024]. URL: <http://lib-os.ru/issledovatellyam/rukovodstvo-po-upravleniyu-dannymi-issledovaniy/metadannye/> (дата обращения: 13.04.2024).

2. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб. : Питер, 2015. 1120 с.

3. Пашуева И. М. Обеспечение защиты информации в компьютерной сети с использованием матрицы доступа на основе приоритетности файлов // Актуальные направления научных исследований XXI века: теория и практика. 2018. Т. 6. № 6 (42). С. 269–270.

4. The Python Standard Library. File and Directory Access. shutil [Electronic resource] // Python Software Foundation : [website]. [2024]. URL: <https://docs.python.org/3/library/shutil.html> (date of treatment: 13.04.2024).