УДК 004.056.57

ОБ ОДНОМ ПОДХОДЕ К ЗАЩИТЕ ОТ ПРОГРАММ-ВЫМОГАТЕЛЕЙ В ОПЕРАЦИОННЫХ СИСТЕМАХ СЕМЕЙСТВА WINDOWS

И. Б. БЕРЕЖНОЙ
(Белорусский государственный университет, г. Минск)

Аннотация. *В статье рассматривается вопрос защиты от программ-вымогателей (ransomware) в ОС семейства Windows. Представлена общая схема функционирования таких программ и рассмотрена роль источника случайности в этой схеме. Предложен подход к организации защиты от программ-вымогателей на основе контроля системных генераторов случайных чисел.*

Ключевые слова: *ransomware, программы-вымогатели, защита информации, псевдослучайные числа, состояние генератора, ОС Windows.*

Программа-вымогатель (англ. ransomware, от слов ransom – выкуп и software – программное обеспечение) – тип вредоносного программного обеспечения (ВПО), который используется в целях вымогательства путем зашифровывания данных, ценных для пользователя компьютерной системы, и последующего требования выкупа за их восстановление. Концепция программы-вымогателя, шифрующей пользовательские файлы, была представлена еще в 1996 году на конференции IEEE Security & Privacy conference [1], однако наибольшее распространение данный тип ВПО получил после 2013 года в связи с развитием криптовалют как трудноотслеживаемых способов платежей [2]. По данным компании Chainanalysis суммарный ежегодный объем платежей на криптовалютные кошельки вымогателей с 2020 года превышает 500 миллионов долларов США, а за 2023 год превысил 1 миллиард долларов США [3]. Другими словами, ландшафт угроз данного ВПО не только обширен, но и продолжает развиваться, демонстрируя все более агрессивный подход.

У программ-вымогателей разных семейств имеются отличия в способах проникновения, особенностях горизонтального распространения и закрепления в системе, некотором дополнительном функционале [4]. Во многих случаях выявить работу программ-вымогателей можно с помощью систем анализа сетевого трафика по признакам генерируемого трафика, характерным для каждого семейства. Однако данный подход является эффективным только в отношении предварительно изученных образцов ВПО.

Для выявления работы программ-вымогателей на конкретном компьютере системы класса EDR используют особенности процесса повышения привилегий и/или контроль перечня запускаемых/останавливаемых процессов и сервисов [4].

Однако такие подходы требуют предварительной настройки на конкретный образец ВПО и не могут быть эффективными в случае использования в ВПО уязвимостей нулевого дня.

Основные рекомендации для борьбы с программами-вымогателями, предлагаемые на текущий момент, представляют собой только общие рекомендации по борьбе с ВПО произвольного типа с повышенным акцентом на необходимость регулярного резервного копирования информации [5]. Другими словами, вместо способов защиты предлагается рассматривать способы минимизации последствий.

Несмотря на все многообразие семейств программ-вымогателей, их криптографические ядра, непосредственно отвечающие за шифрование файлов, эволюционируют по одинаковым путям и по состоянию на 2023 год наиболее эффективные из них соответствуют следующей схеме [6]:

1) злоумышленник генерирует пару открытого и личного ключей для асимметричного шифрования и загружает в ВПО (криптер) открытый ключ, после чего криптер некоторым образом доставляется на компьютер-жертву;

2) при запуске криптер осуществляет поиск файлов, предположительно представляющих ценность для владельца компьютера, по определенным признакам, чаще всего – по расширению имен файлов;

3) для каждого найденного файла с помощью системного криптографически сильного генератора случайных чисел генерируется ключ симметричного шифрования, затем с помощью этого ключа файл зашифровывается (либо полностью, либо частично – в случае большого объема файла), после чего к нему дописывается блок с использованным ключом, зашифрованный на открытом ключе злоумышленника;

4) жертва после оплаты выкупа получает программное обеспечение для расшифрования файлов, содержащее личный ключ злоумышленника (декриптер);

5) декриптер для каждого файла расшифровывает на личном ключе злоумышленника дописанный блок, извлекает из него ключ симметричного шифрования и с его помощью расшифровывает исходный файл.

Указанная схема обладает следующими преимуществами для злоумышленника:

1) ни при каких обстоятельствах закрытый ключ злоумышленника не передается жертве до оплаты выкупа;

2) злоумышленнику не требуется от жертвы никаких дополнительных данных, кроме идентификатора использованной ключевой пары;

3) жертве от злоумышленника требуется один файл фиксированного размера вне зависимости от размера зашифрованных данных;

4) при соблюдении известных требований на размер ключа восстановление личного ключа по открытому для жертвы невозможно;

5) при симметричном шифровании разных файлов использованные ключи не могут совпадать; как следствие, существенно затруднен криптоанализ алгоритма шифрования файлов либо процесс восстановления личного ключа.

Основное тонкое место описанной схемы действий криптографического ядра программы-вымогателя – получение данных от системного генератора псевдослучайных чисел (ГПСЧ). Во время работы вымогателя случайный ключ размером не менее 16 байтов требуется для каждого подходящего файла. А так как количество таких файлов, подходящих по шаблоны поиска, обычно исчисляется тысячами, то, соответственно, и случайная последовательность, которая требуется за короткий промежуток времени с системного ГПСЧ, будет большого размера. При этом следует отметить тот факт, что конкретный поток криптера не сможет продолжить свою работу до получения случайного ключа шифрования.

В актуальных операционных системах семейства Windows начиная с Windows 10 все системные ГПСЧ построены в соответствии с рекомендациями NIST SP 800-90 на базе алгоритма AES-256 в режиме счетчика [7]. При этом реализована буферизация в 128-байтном состоянии, позволяющая сократить нагрузку на процессор при запросах на генерацию псевдослучайных данных малого размера. В целях безопасного многопоточного доступа к буферу реализовано использование системных мьютексов, блокирующих одновременный доступ. После запроса данных из буфера соответствующие байты буфера зануляются в целях конфиденциальности.

В ОС семейства Windows в целях безопасности предусмотрены два режима доступа к процессору для программного кода: режим ядра и пользовательский режим. Пользовательские приложения выполняются в пользовательском режиме, а системные службы и драйверы устройств работают в режиме ядра. Для режима ядра с помощью драйвера CNG.SYS реализован набор из нескольких состояний ГПСЧ по количеству логических процессоров. При вызове генератора псевдослучайных чисел в режиме ядра определяется активный логический процессор и используется соответствующее ему состояние ГПСЧ для выработки данных. В пользовательском режиме для каждого процесса инициализируется собственный аналогичный набор состояний ГПСЧ с использованием библиотеки BcryptPrimitives.dll. На работающем пользовательском компьютере с N логическими процессорами и M запущенными процессами может быть до (N+1)(M+1) используемых состояний ГПСЧ.

Исходя из вышеуказанного, предлагается следующий подход к организации защиты от программ-вымогателей:

1) с помощью специального ПО реализовывается мониторинг создаваемых состояний ГПСЧ и, в частности, скорости их зануления;

2) в случае запуска в системе программы-вымогателя, вне зависимости от конкретных схем маскировки вызова функций CNG.SYS или BcryptPrimitives.dll,

потребление 16-байтовых блоков из состояний ГПСЧ резко возрастет, причем еще до непосредственного шифрования файлов;

3) ПО мониторинга обнаруживает скачок скорости и информирует ПО активной защиты о подозрительном приложении;

4) ПО активной защиты блокирует доступ к соответствующим состояниям ГПСЧ (например, захватывает мьютексы доступа) либо устанавливает большую временну́ю задержку для устранения возможных ложных срабатываний;

5) программа-вымогатель останавливает свое выполнение в ожидании получения доступа к состоянию ГПСЧ, либо замедляется до очень низкой скорости обработки набора файлов, при которой возможно срабатывание других методов защиты.

Достоинства указанного подхода в следующем:

1) использование конструктивных недостатков принципиальной схемы работы программ-вымогателей, как следствие пригодность для борьбы с программами-вымогателями различных семейств;

2) реализация проактивной защиты, то есть предотвращение шифрования пользовательских файлов (или потеря только малого их числа, зашифрованных до момента срабатывания блокировки).

К недостаткам подхода можно отнести необходимость низкоуровневого системного программирования и отсутствие совместимости с более ранними версиями ОС Windows.

ЛИТЕРАТУРА

1. Young A. Cryptovirology: extortion-based security threats and countermeasures / A. Young, M. Yung // Proceedings 1996 IEEE Symposium on Security and Privacy. – Oakland, CA, USA, 1996. – P. 129–140.
2. Fruhlinger J. Ransomware explained: How it works and how to remove it [Electronic resource] / J. Fruhlinger // CSO Online. – Mode of access: https://www.csoonline.com/article/563507/what-is-ransomware-how-it-works-and-how-to-remove-it.html. – Date of access: 10.03.2024.
3. Ransomware Payments Exceed $1 Billion in 2023, Hitting Record High After 2022 Decline [Electronic resource] // Chainalysis Team. – Mode of access: https://www.chainalysis.com/blog/ransomware-2024. – Date of access: 10.03.2024.
4. Омерзительная восьмерка: Тактики, техники и процедуры современных группировок вымогателей [Электронный ресурс] // Kaspersky Threat Intelligence Team. – Режим доступа: https://go.kaspersky.com/rs/802-IJN-240/images/Report_Common%20TTPs%20of%20modern%20ransomware.pdf. – Дата доступа: 10.03.2024.
5. Ransomware protection: How to keep your data safe in 2024 [Electronic resource] // Kaspersky Resource Center. – Mode of access: https://www.kaspersky.com/resource-center/threats/how-to-prevent-ransomware. – Date of access: 10.03.2024.
6. Full source of the Conti Ransomware [Electronic resource] // Github. – Mode of access: https://github.com/gharty03/Conti-Ransomware. – Date of access: 10.03.2024.
7. Ferguson N. The Windows 10 random number generation infrastructure / Niels Ferguson // Microsoft. – Mode of access: https://aka.ms/win10rng. – Date of access: 10.03.2024.