Safe

COLLABORATORS			
	TITLE : Safe		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		April 15, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Safe

1.1

Polska dokumentacja do programu

Safe wersja 11.7

napisanego przez

Zbigniewa ‘Zeeballa‘ Trzcionkowskiego

Proszë, przeczytaj wszystko!

Safe jest programem FREEWARE

1.2 Kontakt z autorem

Zbigniew Trzcionkowski
Astrów 7
43 250 Pawîowice

Ôlijcie do mnie raporty o bîëdach, swoje pomysîy, a takûe
pliki zainfekowane wirusami

100% odpowiedzi dla wszystkich, którzy zaîâczâ dysk.

Dzięki uprzejmoôci Tomasza 'Siumota' Bieliïskiego Safe ma wsparcie
sieciowe w postaci moûliwoôci downloadu Safa z jego strony
i kontaktu ze mnâ przez jego osobisty E-Mail:

E-Mail:
siumot@amiga.org.pl

Strona:
http://amiga.org.pl/~siumot

Szukaj teû nowych wersji na Aminecie - util/virus.

Aby zobaczyê wyniki testowania Safa kliknij tutaj.

Special thanks to:

Jan Andersen z VH-DK za wirusy

Tomasz 'Siumot' Bielinski za Fungusa, testowanie TCP patcha,
oraz liczne pomysîy/bug-reporty

Tomasz 'Error' Wiszkowski za caîoksztaît

1.3 Czym jest Safe

Safe jest maîâ komendâ CLI do wykrywania
wirusów linkowych
w Twoim systemie.
Ten program nigdzie nie rezyduje. Sprawdza pamiëê i siebie
tylko w chwili uruchamiania.
Jedynâ rezydentnâ rzeczâ jest TCP patch - zobacz
parametry
Jeôli potrzebujesz rezydentnego ochroniarza uûyj tego z
pakietu Fungicide by Digital Corruption.
Filozofia Safa róûni sië troszkë, poniewaû jest on
zaprojektowany takûe do odkrywania nowych wirusów.

Wszystko, co masz do zrobienia, to uûyê firmowego skryptu
instalacyjnego albo przerzuciê ikonkë Safe'a na swojâ partycjë
i uruchamiaê w razie potrzeby.
Nie zapomnij, ûe czëôciej uruchamiany Safe = bezpieczniejszy system,
wiëc moûesz takûe dodaê Safe'a do guzików z Opusa, DiskMastera etc.

```
Przykîad uûycia Safe z Diskmasterem:

AddCmd Parent, 10, Parent ; StdIO "CON:0/12/640/100/Alarm!/AUTO"; Extern Safe; ↩
    StdIO CLOSE

Nie zmieniaj nazwy pliku Safe!
Nie próbuj kompresowaê tego pliku!
Wîóû do LIBS: najnowszâ xvs.library, jakâ masz
(ûeby poznaê numery aktualnych wersji xvs i Safe wpisz `safe VER` w Shellu)
Safe potrafi odkryê nowe wirusy tylko gdy jego plik jest umieszczony
w urzâdzeniu niezabezpieczonym przed zapisem i z odrobinâ wolnego
miejsca.
Standardowy RAM: nie moûe byê uûyty poniewaû zawsze jest 100% full.

Jeôli Safe dziaîa, to nic nie zobaczysz
Jeôli znajdzie wiruska, to wgraj duûego antywirusa
jak VirusZ i usuï problem.
Jeôli odkryto nowego/nieznanego wirusa, to wyôlij
ten plik do autora Twojego antywirusa.
Moûesz mi teû wysîaê stosowny plik.
```

1.4 Wymagania

```
System operacyjny 2.0 lub nowszy.

Do rozróûniania i usuwania z pamiëci znanych wirusów potrzebujesz
jak najnowszej xvs.library autorstwa Georga Hormanna.

Do zapisania raportu z Safa parametrem REP potrzebujesz asl v38+

Do instalacji îatki TCPowej potrzebujesz komendy `resident` w C:
```

1.5 Jak to dziaîa

```
                1.Sprawdza pamiêê na obecnoôê HNY99/IOZ
2.Sprawdza swój plik
3.Jeôli coô znajdzie, to dostaniesz w CLI stosowny komunikat
  a program spróbuje rozpoznaê i usunâê problem z pamiëci
  przy uûyciu xvs.library.

Plik jest napisany w specjalnym formacie dla
             wirusów linkowych
             ,
aby je sprowokowaê do infekcji.
Myôlë, ûe 90%
             wirusów linkowych
              zaatakuje ten plik,
wiëc zostanie wykrytych.
Równieû aktywnoôê
             wirusów/trojanów TCPowych
              moûe byê wykryta
jeôli zainstalujesz îatkë TCPowâ.
```

1.6 Korzyôci

```
                Wykrywa obecnoôê
             wirusa linkowego
              w Twoim systemie.
Odkrywa nowe wirusy.
Z zainstalowanâ îatkâ TCPowâ moûesz zobaczyê aktywnoôê

             wirusów/trojanów TCPowych
                Istnieje inny program, podobny do Safa.
Jest to TheUltimateProtector Andreasa Falkenhahna. Ten program daje
uûytkownikowi moûliwoôê sprawdzania kilku plików co okreôlony
przedziaî czasu. Wiëc jeôli masz szybki twardziel (Elboxowy FastATA,
albo SCSI) moûesz uûyê tego programu zamiast Safa, ale nie
zapomnij, ûe musisz wybraê duûo plików, i najlepiej odpakowanych,
aby sprowokowaê infekcjë (lub Safa, ale on wykrywa swojâ infekcjë
samodzielnie). Ludzie z wolniejszymi twardzielami powinni
uûywaê Safa dodanego do guzików Opusa,DiskMastera itd.
```

1.7 Reszta

```
                 Bîëdy: jak zwykle :-)

Do zrobienia: [mnóstwo rzeczy]

Historia:
          rozm.
 v11.0 - 5000, zakryptowany caîy kod i teksty w celu zabezpieczenia
               przed lamerami/zabawnymi ludúmi,
               dodana pokazywaczka dîugoôci pliku przed i po,
               dodane komunikaty o podstawionych instrukcjach przez
               (
             hunk increasing viruses
             ),
               poprawiony bîâd w patchu na TCP (kod nie byî na 100% PURE),
               dodany parametr -  REP/S. Jeôli masz asl v38+ to
               pokaûe sië filerequester do zapisania komunikatów
               z Safa do pliku, teraz teksty z Safa pokaûâ sië
               nawet z ikonki (byî bîâd),
               dodane do dokumentacji inforamcje o powstajâcym
               Safe's site w sieci,
               wszystko w stu procentach sprawdzone Enforcerem,
               Mungwallem i Scratchem
 v11.1 - 5000, dodana współpraca z xfdmaster.library w celu okreôlania
               nazwy cruncherа oraz do testowania struktury hunków,
 v11.2 - 5000, dodany parametr WBLOCK/S, usuniëta XFDowa czëôê kodu,
               bo byîa trochë bezuûyteczna, przetestowany z kilkoma
               istniejâcymi wirusami
 v11.3 - 5000, zoptymizowany, przetestowany z wiëkszâ iloôciâ wirusów,
               rozpoczëte prace nad analizerem wektorów
               (VECS/S w
             parametrach
             )
 v11.4 - 5000, dodany auto-killer dla patcha CrM, dodany heurystyczny
```

```
                test najpopularniejszych wektorów z dos.library
                dodany antistealth (dla HitchHikerów)
    v11.5 - 5000, dodane pokazywanie pr_WaitPkt strategicznych procesów,
                inne dodatki w VECS/S
    v11.6 - 5000, dodany nowy antistealth (dla Beol96),
                dodane czekanie na walidacjë napëdu z którego
                uruchamiasz Safa
    v11.7 - 5000, dodany reset VBRa poprzez przytrzymanie LMB
```

1.8 Do tîumaczy

```
Jeôli chcesz zrobiê tîumaczenie po prostu zrób je i wyôlij mi.

Gîówny plik wykonywalny jest tylko po angielsku i pozostanie.
Tîumaczenia guida muszâ byê jako oddzielny plik.
Tîumaczenia instalera muszâ byê dodane do skryptu.
```

1.9 Parametry

```
                Safe od wersji 10.6 oferuje z CLI/Shella wzorzec:

REBOOT/S,RENRAM/S,TCPPATCH/S,VER/S,REP/S,VECS/S

REBOOT   - daje standardowy reboot bez czyszczenia wektorów resetu
RENRAM   - zmienia nazwë Ram disk: na RAM:
           Pomaga to przy niektórych programach
TCPPATCH - instaluje îatkë do wykrywania
                wirusów/trojanów TCPowych
                VER      - pokazuje wersje Safa i xvs.library
REP      - otwiera filerequester do zapisania raportu z Safa
           do pliku (potrzebujesz do tego asl v38+)
WBLOCK   - wykonuje LockPubScreen(NULL) aby zapobiec zamykaniu WB
VECS     - chwilowo pokazuje kilka systemowych wektorów.
           Pokazuje takûe wyniki specjalnego testu heurystycznego.
           Wiëkszoôê przetestowanych wirusów daîo Suspicity=50+,
           ale nie zapomnij, ûe to tylko podejrzenia, wiëc
           nawet legalne patche mogâ daê duûe numery!
           Innâ waûnâ rzeczâ jest pole pr_WaitPkt w strategicznych
           procesach - zawsze powinno byê 0!
```

1.10

```
hunk - w wykonywalnym pliku AmigaDosu oznacza jego czëôê.
       Kiedy uruchamiasz program systemowa funkcja LoadSeg
       îaduje róûne hunki pliku do róûnych miejsc w pamiëci.
       Najpopularniejsze hunki (zwane w asemblerze sekcjami) to:
       code - binarny program dla procesora MC680x0, drobne dane itp.
       data - dane programu (obrazki, moduîki itp.),
              programy dla Coppera itp.
       bss  - uûywany do umieszczania w programie duûych pustych
```

```
            obszarów bez zwiëkszania rozmiaru programu na dysku.
            Zawiera tylko dane o dîugoôê pustych obszarów.
      reloc- Zawiera dane o relacjach pomiëdzy innymi hunkami,
            które muszâ zostaê przeliczone, gdy hunki sâ
            zaîadowane do pamiëci
      end  - 4 bajty - tylko identyfikator. Uûywany na koïcu innych
            hunków. System nie potrzebuje go przy czëôci hunków,
            dzięki czemu hunk dodawany przez FileShielda jest
            mniejszy o 4 bajty
```

1.11

```
               linkvirus - oznacza prawdziwego wirusa. Klasyczny Amigowy ↩
                  linkvirus
          dodaje swój kod do wykonywalnych plików, by byê
          rozpowszechnianym wraz z nimi. Kiedy uûytkownik
          uruchamia prawidîowo zainfekowany plik kod wirusa
          jest wykonywany i wirus dodaje swój kod do jednej
          z systemowych funkcji (LoadSeg, Write, Open itp.)
          Kiedy funkcja jest uûywana wirus próbuje zainfekowaê
          kolejny plik.
          Na Amidze sâ dwie gîówne drogi infekcji pliku:

          1.
               first hunk increasing
                           2.
               hunk adding
```

1.12

```
              first
              hunk
               increasing
 ( zwiëkszanie pierwszego hunka )
                     - dodawanie kodu wirusa na koïcu
                       pierwszego hunka (jeôli code hunk) i podstawienie
                       jednej z instrukcji procesora MC680x0 skokiem
                       do kodu wirusa.
                       Najpopularniejsze instrukcje do zastâpienia
                       to: RTS, BSR, JSR, MOVE.L 4.W,A6
                       FileShield walczy z tym rodzajem wirusów.
                       Safe od wersji 11.0 potrafi wykryê i pokazaê kilka
                       podmienionych instrukcji.
```

1.13

```
              hunk
               adding
( dodawanie hunka )
          - dodawanie do pliku hunka(ów) z kodem wirusa. Nie jest znowu
```

aû tak îatwo napisaê hunk addera, wiëc jest wiëcej zwiëkszaczy
pierwszego hunka.
FileShield walczy z nimi via uûycie hunka Reloc32Short
nieznanego przez duûâ czëôê z nich
Przy okazji - FileShield pracuje jak ten rodzaj wirusów
poniewaû dodaje wîaônie swój code hunk :-)

1.14

TCP viruses/trojans - normalne wirusy lub trojany(faîszywe programy,
biblioteki), które otwierajâ zdalne drzwi
sieciowe poprzez stworzenie shella w urzâdzeniu TCP:

Przykîadowe nazwy shellów
Fungus linkvirus : TCP:1666
rexxkuang11.library 0.36 : TCP:2551
rexxkuang11.library 0.27 : TCP:2333

Aby wykryê ten rodzaj nielegalnej dziaîalnoôci dodaîem
do Safe'a (od wersji 10.4) parametr 'T', który pokaûe
komunikat, gdy coô spróbuje stworzyê shella w TCP: