
Wirusy
dla systemu
Linux

Spis treści

Czym są wirusy?

Wirus – program lub blok wykonywalnego kodu, napisany by infekować pliki przez przyłączanie się do nich, nadpisywanie lub zastępowanie innych programów. Typowy wirus ma dwie funkcje:

1. Powielanie i rozpowszechnianie
2. Dostarczanie implementacji procedury destrukcyjnej (lub innej)

Czym są wirusy?

Wirusy, w przeciwieństwie do robaków, potrzebują pliku nosiciela

Sieć komputerowa nie jest dla wirusów podstawową drogą rozmnażania i rozpowszechniania

Opinia fachowca

“Zdecydowanie wzrasta liczba wirusów linuxowych, obecnie przybywa ich 20-30 miesięcznie.”

Marek Sell

Computerworld nr 16-2002

Pierwszy wirus pod Linuksem

W 1996 roku autor wirusa Bliss wysłał wczesne wersje swojego kodu źródłowego na listy dyskusyjne comp.security.unix, alt.comp.virus i comp.os.linux.misc

W 1997r. początkowo uznany za trojana lub wirusopodobnego robaka

Ostatecznie, po analizie przeprowadzonej przez Alana Coxa, okrzyknięto go pierwszym wirusem dla Linuxa

Pierwszy wirus pod Linuksem

Kod źródłowy Blissa jest znany – możliwe są porty na inne systemy (SunOS, FreeBSD)

Zawiera opcję –bliss-uninfect-files-please

Infekuje wszystkie pliki binarne do których ma prawo zapisu lub dostęp przez rsh

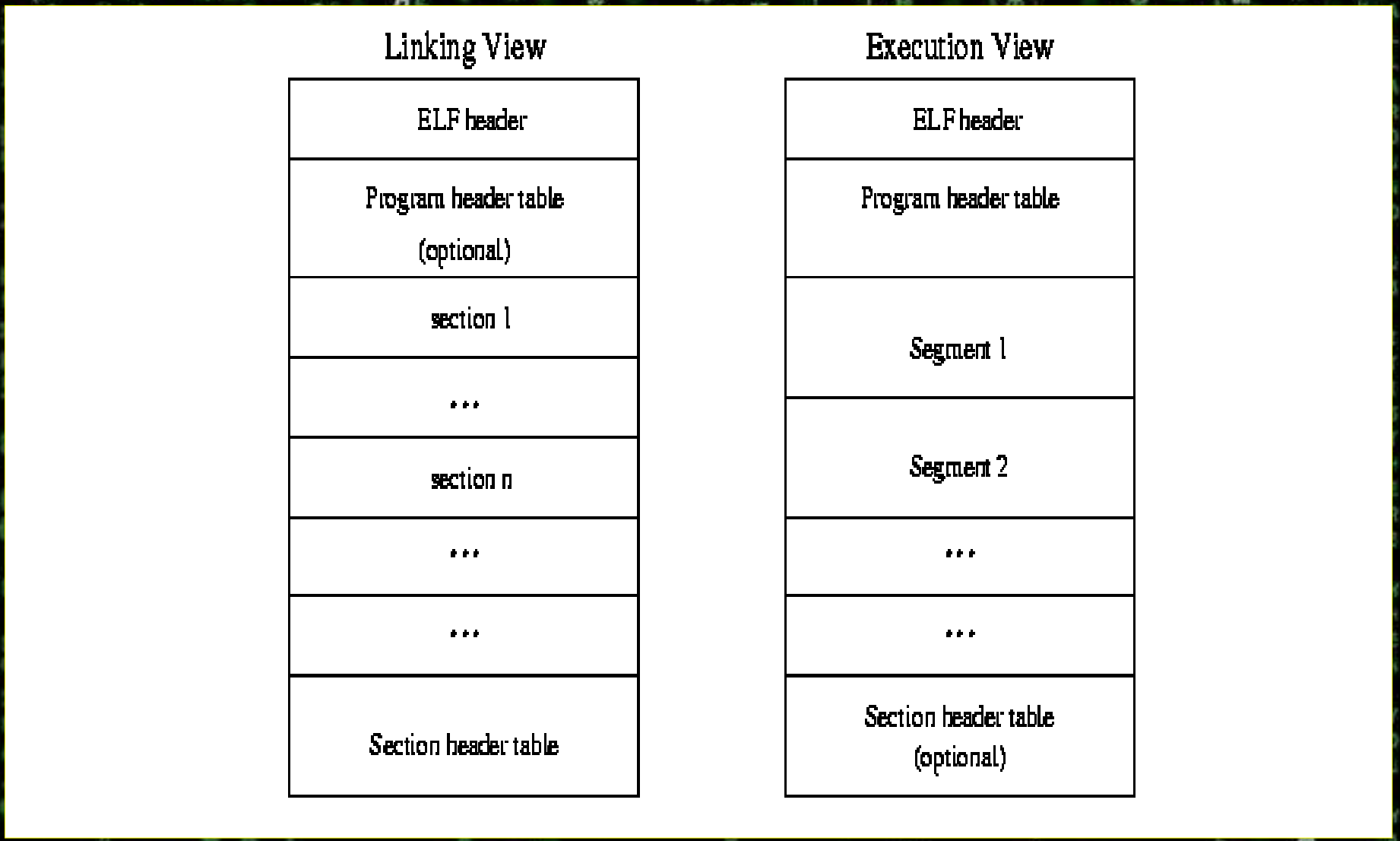
Format plików ELF

ELF – Executable and Linkable Format, zastąpił w systemach UNIX-owych foramt a.out

Niektóre z mozliwości formatu ELF:

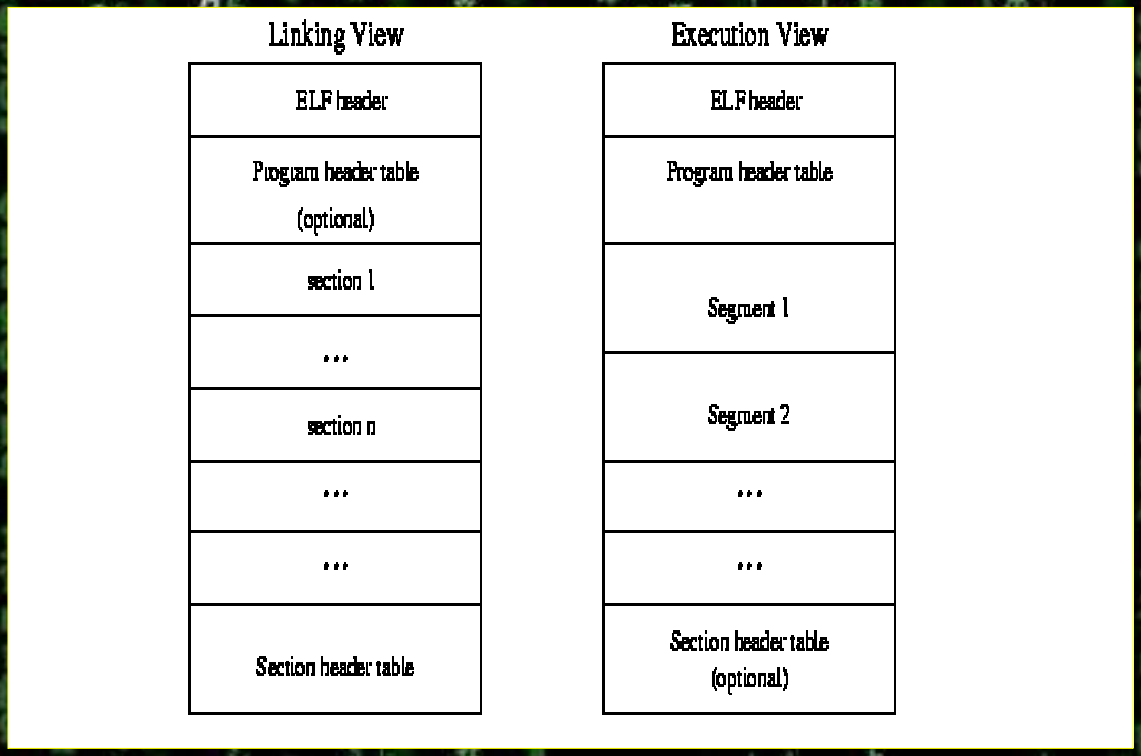
- Dynamiczne linkowanie
- Dynamiczne ładowanie
- Zapewnienie kontroli nad czasem wykonania (*runtime*) programu
- Udoskonalona metoda tworzenia bibliotek dzielonych (kompatybilne na wielu platformach)

Format plików ELF



Dualny widok struktury pliku ELF

Format plików ELF



Plik wykonywalny nie musi zawierać informacji o podziale na sekcje.

Plik obiektowy (wynik pracy kompilatora) nie musi zawierać nagłowka programu.

Format plików ELF

```
#define EI'NIDENT    16

typedef struct -
    unsigned char     e'ident[EI'NIDENT];    //  file ID, interpretation
    Elf32'Half        e'type;                //  object file type
    Elf32'Half        e'machine;             //  target architecture
    Elf32'Word         e'version;            //  ELF version
    Elf32'Addr         e'entry;              //  starting virtual address
    Elf32'Off         e'phoff;               //  file offset to program hdr
    Elf32'Off         e'shoff;               //  file offset to section hdr
    Elf32'Word         e'flags;              //  processor-specific flags
    Elf32'Half        e'ehsize;              //  the ELF header's size
    Elf32'Half        e'phentsize;           //  program hdr entry size
    Elf32'Half        e'phnum;               //  program hdr entry number
    Elf32'Half        e'shentsize;           //  section hdr entry size
    Elf32'Half        e'shnum;               //  section hdr entry number
    Elf32'Half        e'shstrndx;            //  section hdr index for strings
" Elf32'Ehdr;
```

Nagłówek pliku ELF: dane identyfikacyjne, typ pliku, architektura, wersja specyfikacji, punkt wejścia, offset nagłówka programu, offset nagłowków sekcji, flagi, rozmiar nagłówka, rozmiar elementu nagłowka programu ilość elementów nagłowka programu, rozmiar nagłówka sekcji, ilość nagłówków sekcji, numer nagłówka wskazującego nazwy sekcji

Format plików ELF

```
typedef struct -
    Elf32'Word          p'type;       //  type of the segment
    Elf32'Off          p'offset;     //  file offset to segment
    Elf32'Addr          p'vaddr;      //  virtual address of first byte
    Elf32'Addr          p'paddr;      //  segments' physical address, i
    Elf32'Word          p'filesz;     //  size of file image of segment
    Elf32'Word          p'memsz;      //  size of memory image of se
    Elf32'Word          p'flags;      //  segment-specific flags
    Elf32'Word          p'align;      //  alignment requirements
" Elf32'Phdr;
```

Nagłówek programu: odpowiada za podział programu na bloki, zwane segmentami; określa gdzie załadować segment i jakiego interpretera użyć; zawiera następujące pola: typ segmentu, offset w pliku, adres wirtualny segmentu, adres fizyczny segmentu, rozmiar segmentu w pliku, rozmiar segmentu w pamięci, flagi, wyrównanie

Format plików ELF

```
typedef struct -
    Elf32'Word          sh'name;          //  name of section
    Elf32'Word          sh'type;          //  type of the section
    Elf32'Word          sh'flags;         //  section-specific attributes
    Elf32'Addr          sh'addr;          //  memory location of sectio
    Elf32'Off           sh'offset;        //  file offset to section
    Elf32'Word          sh'size;          //  size of section
    Elf32'Word          sh'link;          //  section type dependent
    Elf32'Word          sh'info;          //  extra information
    Elf32'Word          sh'addralign;     //  address alignment constrai
    Elf32'Word          sh'entsize;       //  size of an entry in section
"  Elf32'Shdr;
```

Nagłówek sekcji: opisuje plik z punktu widzenia linkera; zawiera następujące pola: nazwa sekcji (indeks w tablicy nazw), typ sekcji (kod, dane, informacje dla debuggera), flagi, adres wirtualny sekcji, offset w pliku, rozmiar sekcji, numer sekcji zawierającej dane pomocnicze (dla niektórych sekcji), dodatkowe informacje, wyrównanie, rozmiar elementu tablicy (jeśli istnieje)

Jak napisać wirusa?

Przykładowe sposoby to:

1. Dołączenie kodu wirusa na końcu pliku lub w poszerzonym segmencie danych. Należy zmodyfikować w nagłówku punkt wejścia oraz sprawić by dodany kod był ładowany do pamięci. Wirus musi zatroszczyć się o powrót do kodu ofiary.

Jak napisać wirusa?

Przykładowe sposoby to:

2. Stworzenie nowej sekcji i uzupełnienie nagłówka o wszystkie wpisy – wirus będzie ładowany do pamięci. Można także nadpisać kod początku programu, kopiując przedtem oryginalny kod na koniec pliku i uzupełnić go o odpowiednie skoki.

Wnioski

Znane wirusy dla Linuksa według www.viruslist.com:

Vit.4096 – drugi po Blissie wirus atakujący pliki binarne

Satyr – nie czyni szkód, nie jest rezydentny

Zipworm – infekuje archiwa ZIP

RST – infekuje pliki binarne, udostępnia root-shell

Winter, Kagob, Nuxbee, Diesel – infekują pliki binarne, nie czynią szkód

Wnioski

Znane wirusy dla Linuksa według www.viruslist.com:

- Jac – infekuje pliki ELF w tym samym katalogu
- Gildo – rezydentny, napisany w asemblerze
- Winux – wirus jednocześnie infekujący pliki wykonywalne typu PE (Windows) oraz ELF (Linux). Stanowi demonstrację możliwości stworzenia mikroba wieloplatformowego

Wnioski

Wszystkie prawdziwe wirusy na Linuksa bazują na niedbalstwie administratorów.

Niektórzy twierdzą, że Linux może stać się ofiarą wirusa tylko z “pomocą” roota.

Literatura

The ELF Virus Writing HOWTO - http://www.lwfug.org/~abartoli/virus-writing-HOWTO/_html/

artykuł o wirusach na LinuxNews - http://hedera.linuxnews.pl/_news/2002/05/17/_long/1268.html

"Infekcja plików ELF, czyli wirusy w Linuksie" artykuł Tomasza Potęgi w Software 2.0

list Alana Coxa z wyjaśnieniem różnicy między wirusami i robakami - http://math-www.uni-paderborn.de/~axel/bliss/trojan.txt

wyczerpujacy opis formatu ELF - http://www.loser-console.org/ps2/info/elf%20format.pdf

opis formatu ELF (diagramy w referacie) - http://www.cs.ucdavis.edu/~haungs/paper/node11.html

Opis statycznie linkowanego pliku ELF - http://140.198.144.65/tlk-html/node62.html

Encyklopedia wirusów - http://www.wiruslist.com