Принципы Бернштейна для написания безопасного кода

Минко В.С. vitaly.minko@infotecs.ru
Отдел ПиР ПАК

27.04.2015

“I often see people saying 'Nobody has produced an invulnerable software system;
therefore, nobody will ever produce an invulnerable software system.'
By the same bogus reasoning, nobody will ever reach Mars; nobody will ever find MD5 collisions;
nobody will ever cure cancer; nobody will ever build a 1GHz CPU; etc."
— Daniel J. Bernstein

Daniel J. Bernstein

Известен как автор:

- Эллиптической кривой Curve25519.
- Почтового сервера qmail.
- DNS-сервера djbdns.

Как избежать уязвимостей?

Каждая уязвимость — это «баг».
Допустим: 1 «баг» на каждые N строк.
Если 1000N строк, то ~1000 «багов».

- Избежание «багов» (основное объяснение исключительной безопасности qmail)
  - Юнит-тестирование
  - Статический анализ кода
  - Code review
- Снижение объёма исходного кода
- Снижение объёма доверенного кода
  - Помещать большую часть кода в «недоверенные зоны»

Избежание ошибок

- Использование ясных потоков данных
  - Архитекутра qmail многопроцессная (нет прямого доступа к переменным друг друга)
  - Каждый процесс имеет относительно малое число возможных состояний
  - Внутренние потоки данных максимально ясные

Избежание ошибок

- Безопасные типы данных
  - Целочисленные операции (y=x+1; y>x?). Проверка превышения границ или специальные библиотеки.
  - Используются строки, содержащие информацию о длине (вместо C-строк).

```
typedef struct stralloc
{
char *s;
unsigned int len;
unsigned int a;
}
```

Избежание ошибок

- Избежание синтаксического анализа
  - Зачастую парсеры содержат ошибки.
  - В qmail парсинг по возможности избегается.
  - Для межпроцессоного взаимодействия и внутренний файловых форматов используется примитивнейший формат.
  - Если полностью избежать нельзя, каждое синтаксическое правило покрывать отдельным тестом.

Избежание ошибок

- Обобщение ввода данных
  - Каким образом можно избегать редко проявляющихся ошибок?
  - Абстрагировать механизм ввода данных (взаимодействовать не только с файловой системой).
  - Создание исчерпывающего набора тестов для такого обобщённого механизма ввода.

Снижение объёма исходного кода

- Выделение общих функций

Отрывок из Sendmail:

```
if (dup2(fdv[1], 1) < 0)
{
    syserr("%s: cannot dup2 for stdout", argv[0]);
    _exit(EX_OSERR);
}
close(fdv[1]);
```

Аналогичный код в Qmail:

```
int fd_move(int to,int from)
{
    if (to == from) return 0;
    if (fd_copy(to,from) == -1) return -1;
    close(from);
    return 0;
}
```

Снижение объёма исходного кода

- Автоматическая обработка ошибок

Отрывок из Qmail:

```
void die_write()
{
    substdio_putsflush(subfderr,"qmail-pw2u: fatal: unable to write output\n");
    _exit(111);
}


/* ... */
if (substdio_puts(subfdout,uugh) == -1) die_write();
if (substdio_puts(subfdout,dashcolon) == -1) die_write();
if (substdio_put(subfdout,x,i) == -1) die_write();
/* ... */
```

Решение — использовать языки с механизмами обработки исключений.

Снижение объёма исходного кода

- Использование сетевых утилит

  Qmail использует `inetd` для прослушивания портов. Когда соединения создано, `inetd` запускает `qmail-smtpd` для обработки входящего SMTP-соединения.

  Sendmail сама занимается сетевым взаимодействием.

Снижение объёма исходного кода

- Использование файловой системы

  Как MTA находит инструкции по доставке для username, получив `username@domain`?

  - Обычно используется БД. Sendmail хранит инструкции в общем «aliases file», извлечение инструкций требует проведения синтаксического анализа.
  - Qmail хранит инструкции в файле `.qmail-username`. Поиск инструкций не требует пасинга — простое открытие файла.

Снижение объёма исходного кода

- Использование контроля доступа ОС

  Как обрабатывать `.forward`?

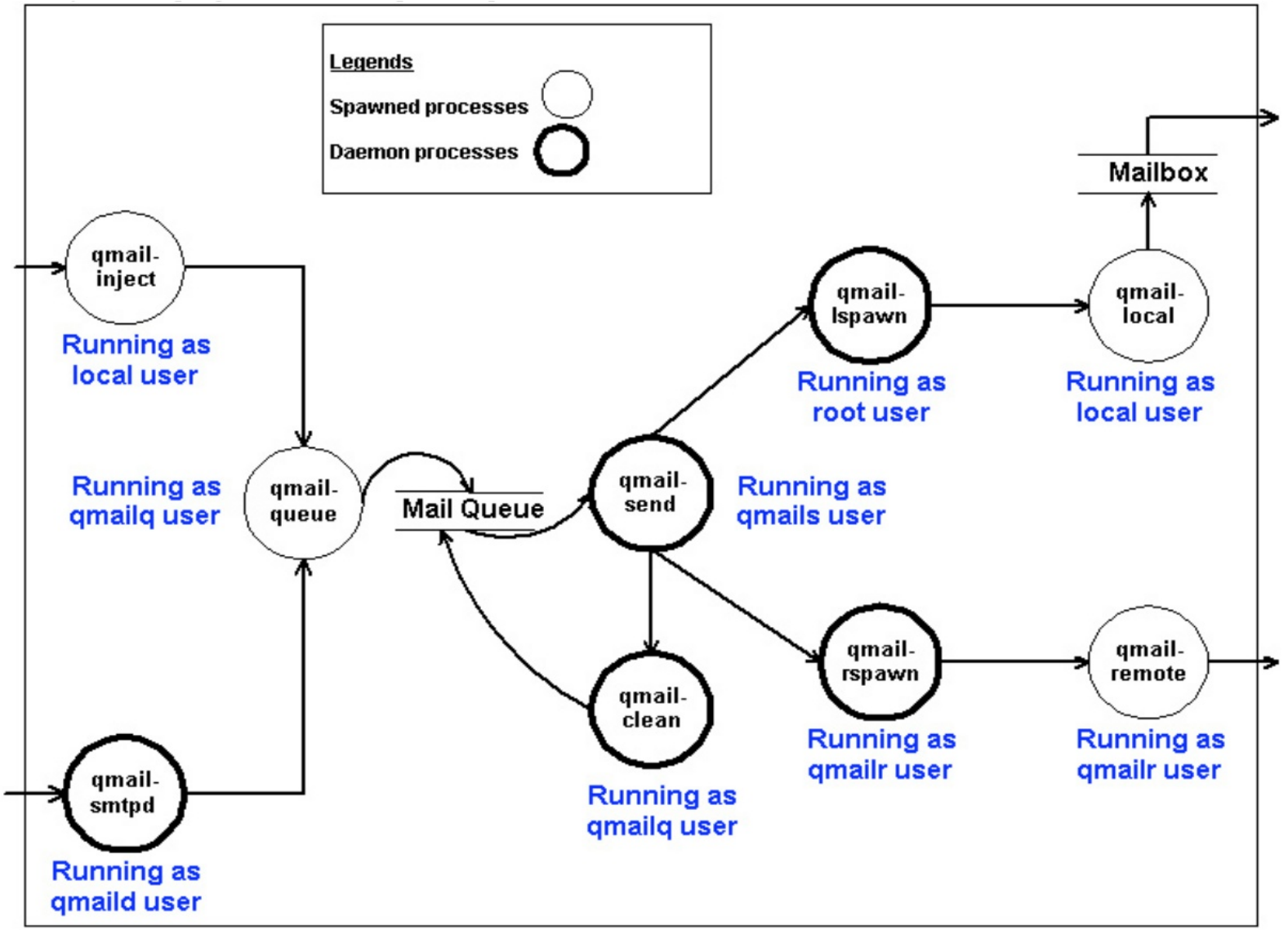
  - Sendmail проверяет имеет ли пользователь права на чтение, является ли файл ссылкой, запоминает пользователя для запуска программ и т.п.
  - Qmail запускает `qmail-local` с правами пользователя. `qmail-local` просто считывает файл `.forward` и запускает указанные пользователем программы.

Снижение объёма исходного кода

Сравнение объёмов исходных кодов разных почтовых серверов.

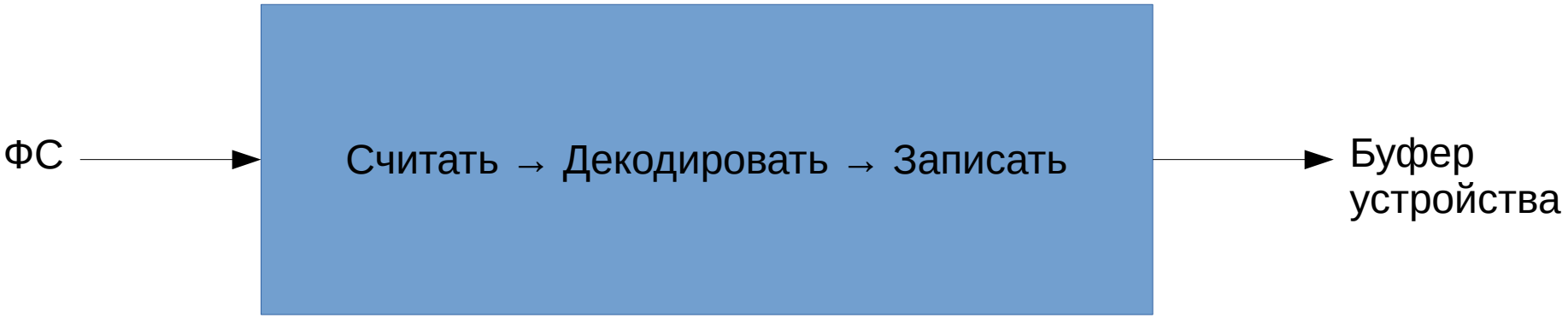
MTA	Строк	Слов	Символов	Файлов
qmail-1.01	16028	44331	370123	288
sendmail-8.8.8	52830	179608	1218116	53
zmailer-2.2e10	57595	205524	1423624	227
smail-3.2	62331	246140	1701112	151
exim-1.90	67778	272084	2092351	127

Компартментализация



Снижение объёма доверенного кода

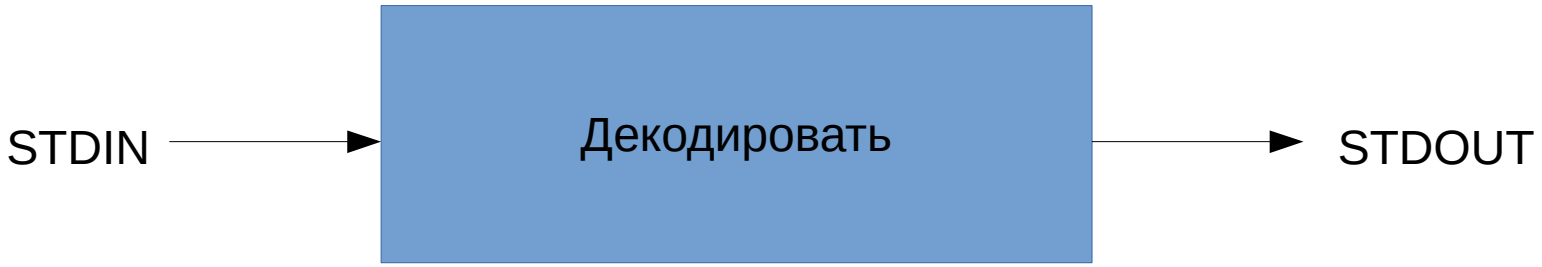
Программа, трансформирующая данные от одного источника (например, просмотрщик изображений или других мультимедийных файлов)



Ошибка в коде декодирования может привести к чтению/модификации файлов пользователя

Снижение объёма доверенного кода

Программа, трансформирующая данные от одного источника (например, просмотрщик изображений или других мультимедийных файлов)



Программа изолируется, помещается в «недоверенную зону».

Снижение объёма исходного кода

- Создание «недоверенной зоны» в UNIX:
  - Запретить новые файлы, сокеты, т. п.

    `RLIMIT_NOFILE = 0`
  - Запретить доступ к ФС
    `chdir, chroot` в пустую директорию
  - Выбрать `UID` для процесса
  - Убедиться, что никто больше не работает под этим UID:

    `fork, setuid(targetuid), kill (-1, SIGKILL)`
  - Запретить `kill` и `ptrace`, установив UID/GID
  - Запретить `fork`
    `RLIMIT_NPROC = 0`
  - Установить ограничение по памяти и другим ресурсам.

Заблуждения

- Для многих безопасность заключается не в том, чтобы избавиться от уязвимостей в коде, а в том, чтобы воспрепятствовать эксплуатации этих уязвимостей:
  - Системы обнаружения вторжений (IDS)
  - Обфускация (запутывание) кода
  - Microsoft EMET (DEP, ASLR, SEHOP, HSA)

Заблуждения

Complete Microsoft EMET Bypass Developed

Posted by **Unknown Lamer** on Monday February 24, 2014 @11:04PM
from the just-a-teeny-tiny-bug dept.

msm1267 writes

> "Researchers at Bromium Labs are expected to announce today they have developed an exploit that bypasses all of the mitigations in Microsoft's Enhanced Mitigation Experience Toolkit (EMET). Principal security researcher Jared DeMott is delivered a presentation at the Security BSides conference explaining how the company's researchers were able to bypass all of the memory protections offered within the free Windows toolkit. The work is significant given that Microsoft has been quick to urge customers to install and run EMET as a temporary mitigation against zero-day exploits targeting memory vulnerabilities in Windows or Internet Explorer. The exploit bypasses all of EMET's mitigations, unlike previous bypasses that were able to beat only certain aspects of the tool. Researchers took a real-world IE exploit and tweaked it until they had a complete bypass of EMET's ROP, heap spray, SEHOP, ASLR, and DEP mitigations."

Заблуждения

- Если мы определяем успех как сдерживание вчерашних атак, вместо того, чтобы прикладывать усилия к предотвращению всех возможных атак, то не стоит удивляться, что наши программы остаются уявзимыми к атакам завтрашнего дня.

Заблуждения

- Принцип минимизации привилегий
  - Может снизить ущерб от уязвимости, но почти никогда не устраняет уязвимость.
  - Netscape DNS Helper предотвращает доступ к диску.
    Баг в libresolv позволил перехватить контроль над DNS Helper, модифицировать DNS-трафик и перехватывать соединения пользователя.
  - ➔ Преобразование DNS Helper в недоверенный код — более комплексная мера, чем наложение ограничений на ресурсы ОС, к которым имеет доступ программа. Если программа обрабатывает данные от нескольких источников, каждый источник должен быть изолирован от модификации данных других источников.

Заблуждения

- Отвержение принципов как априори неприемлемо замедляющих программу:
  - Запуск процесса для использования контроля доступа ОС, изоляции парсинга и т. п.
  - Использование ФС как хранилище вместо СУБД.

- Безопасность важнее скорости:
  1. Обеспечить отсутствие уязвимостей;
  2. Обеспечить быстродействие.

Спасибо за внимание

Вопросы?