Университет ИТМО

Кафедра ВТ

Системное программное обеспечение

Лабораторная работа №2

Выполнил студент 2 курса
Группы P3211 Романов Олег
Преподаватель: Дергачев А.М.

Санкт-Петербург
2016 год

ИНСТРУКЦИЯ ПО РАБОТЕ С УТИЛИТОЙ

prex – первоначальное название было precs (Portable realtime embedded computing system). Микроядро Prex обеспечивает основные функции абстрактного процессора и минимального аппаратного обеспечения. Так же предоставляет библиотеку эмуляции POSIX для взаимодействия с пользователем.

КРАТКОЕ ОПИСАНИЕ НАЗНАЧЕНИЯ КОМАНД TROFF И NROFF

• nroff (new roff)– форматирует документы для вывода на принтер или на экран;
• troff (typesetter roff) – команда, позволяющая форматировать документы для вывода на наборную машину.

Для того, чтобы создать документацию в Unix используется утилита nroff. Данные, передаваемые для обработки nroff содержат текст документа и инструкции, описывающие, в каком виде должен быть распечатан этот текст. troff и nroff похожи между собой, но главное отличие состоит в том, что nroff не поддерживает смену шрифта и размеров.

ТЕКСТ РУКОВОДСТВА С РАЗМЕТКОЙ

```
.TH "prex" "1" "12.11.2016"
.SH НАЗВАНИЕ
.RS 2
prex
\- система отслеживания и управления пробными точками в процессе или ядре
.RE
.SH СИНТАКСИС
.RS 2
.PP
prex [-o файл_трассировки] [-l библиотеки] [-s размер_в_кб] cmd [cmd-аргументы]...
.PP
prex [-o файл_трассировки] [-l библиотеки] [-s размер_в_кб] -p pid
.PP
prex -k [-s размер_в_кб]
.RE
.SH ОПИСАНИЕ
.RS 2
prex загружает библиотеку libtnfprobe, а затем находит все пробы в целевом исполняемом файле или ядре и предоставляет интерфейс пользователю для управления ими. Это позволяет probe измениться для трассировки, отладки или того и друго вместе. Трассировка создает трассировочный файл TNF (Trace Normal Form), который может быть конвертирован в ASCII с помощью tnfdump(1) и использован для анализа производительности. Отладка генерирует строку стандартной ошибки при падении во время выполнения.

prex работает только на динамических исполняемых файлах.
.RE

Вызов PREX
.RS 2
Существует три способа запуска prex:
.TP
1.
Использование prex для старта целевых приложений cmd, которые построенны  без зависимости от libtnfprobe. prex устанавливает переменную окружения LD_PRELOAD для загрузки libtnfprobe в целевом процессе. Затем prex использует переменную окружения PATH для поиска целевого приложения.
.TP
2.
```

Прикрепление prex к запущенному приложению. Запущенное приложение должно уже иметь связь с libtnfprobe. В качестве альтернативы, пользователь может вручную установить LD_PRELOAD для включения libtnfprobe.so.1 до вызова цели.
.TP
3.
Использование prex в режиме ядра с ключом -k. В режиме ядра будут доступны дополнительные команды, но некоторые команды из других режимов станут недоступны.
.RE
.SS
Управление файлом поиска пути
.RS 2
Есть два различных способа общения с prex:
.IP \(bu 5
Характеристика управляющего файла. Во время запуска prex ищет файлы с именем .prexrc в каталогах указаных ниже. prex не останавливается на первом найденом. Порядок поиска:
.nf
  $HOME/
  ./
.fi
.IP \(bu 5
Ввод команд в строке prex.
.RE
.RS 2
Язык команд одинаков для обоих методов и задается в USAGE. Команды, которые что-то возвращают, не будут иметь смысла в управляющем файле. Вывод будет идти на стандартный поток вывода.

При использовании prex в целевом процессе, цель будет в запущенном или приостановленом состоянии. Это выясняется наличием или отсутствием подсказки prex>. Если подсказки нет, то целевой процесс запущен. Нажатие Control-C остановит целевой процесс и вернет пользователю подсказку. Однако, нет гарантии, что возврат подсказки произойдет сразу.
.RE
.SH ОПЦИИ
.SS
Команда поддерживает следующие ключи:
.RS 2
.IP "-k" 20
Режим ядра: В режиме ядра будут доступны дополнительные команды, но некоторые команды из других режимов станут недоступны.
.IP "-l libraries" 20
Упомянутые библиотеки связаны в целевом процессе, использующим LD_PRELOAD. Этот ключ не может быть использован для присоединения к запущенному процессу. Аргумент ключа -l должен быть строкой, разделенной пробелами и заключенной в двойные ковычки.
.IP "-o файл_трассир" 20
В файл будет записан вывод трассировки. Файл_трассировки считается относительно текущей рабочей директории.

Если prex присоединяется к уже запущенноу процессу, то новый файл_трассировки не будет использоваться. Если имя файла не указано, то по уолчанию используется /$TMPDIR/trace-pid, где pid - ИД целевой программы. Если TMPDIR не задан, то используется /tmp.
.IP "-s размер_в_кб" 20
Максимальный размер выходного файла трассировки в кбайтах. По умолчанию, размер 4096 кбайт для нормального использования и 384 кбайт в режиме ядра. Минимальный размер составляет 128 кбайт.
.RE
.SH СМОТРИТЕ ТАКЖЕ
.RS 2
ed(1), kill(1), ksh(1), ld(1)
.RE

КРАТКОЕ ОПИСАНИЕ КАЖДОЙ ИСПОЛЬЗОВАННОЙ В РАЗМЕТКЕ ДИРЕКТИВЫ

```
.TH          Заголовок man-страницы
.SH          Заголовок раздела
.SS          Заголовок подраздела
.RS i        Начало относительного сдвига левой границы на i символов
.RE          Конец  относительного сдвига левой границы на i символов
.PP          Новый параграф
.TP          Параграфа с висячим тегом
.IP x i      Параграф с отступом и возможным висячим тегом (x – тег, i – отступ)
.nf          Начало фрагмента кода
.fi          Конец фрагмента кода
```

ВНЕСЁННЫЕ В СТАРТОВЫЙ КОМАНДНЫЙ ФАЙЛ ИЗМЕНЕНИЯ

```
MANPATH=/home/s207218/._SPO/lab2/man:$MANPATH
```

РЕЗУЛЬТАТ ВЫВОДА КОМАНДЫ MAN ДЛЯ ПОДГОТОВЛЕННОГО РУКОВОДСТВА

```
User Commands                                                   prex(1)

НАЗВАНИЕ
       prex - система отслеживания и управления пробными точками в
       процессе или ядре

СИНТАКСИС
       prex  [-o  файл_трассировки]  [-l  библиотеки]  [-s
       размер_в_кб] cmd [cmd-аргументы]...

       prex  [-o  файл_трассировки]  [-l  библиотеки]  [-s
       размер_в_кб] -p pid

       prex -k [-s размер_в_кб]

ОПИСАНИЕ
       prex загружает библиотеку libtnfprobe, а затем находит все
       пробы в целевом исполняемом файле или ядре и предоставляет
       интерфейс пользователю для управления ими.  Это  позволяет
       probe измениться для трассировки, отладки или того и друго
       вместе. Трассировка создает трассировочный файл TNF (Trace
       Normal  Form),  который может быть конвертирован в ASCII с
       помощью  tnfdump(1)  и  использован  для  анализа
       производительности.  Отладка генерирует строку стандартной
       ошибки при падении во время выполнения.

       prex работает только на динамических исполняемых файлах.

     Вызов PREX
       Существует три способа запуска prex:
       1.  Использование prex для старта целевых приложений cmd,
           которые  построенны  без зависимости от libtnfprobe.
           prex устанавливает переменную окружения LD_PRELOAD
           для  загрузки  libtnfprobe  в целевом процессе. Затем
           prex использует переменную окружения PATH для  поиска
           целевого приложения.

       2.    Прикрепление  prex  к  запущенному  приложению.
           Запущенное  приложение должно уже иметь связь с
           libtnfprobe. В  качестве  альтернативы,  пользователь
           может  вручную установить LD_PRELOAD для включения
           libtnfprobe.so.1 до вызова цели.
```

3. Использование prex в режиме ядра с ключом -k. В режиме ядра будут доступны дополнительные команды, но некоторые команды из других режимов станут недоступны.

Управление файлом поиска пути

Есть два различных способа общения с prex:

- Характеристика управляющего файла. Во время запуска prex ищет файлы с именем .prexrc в каталогах указаных ниже. prex не останавливается на первом найденом. Порядок поиска:
  $HOME/
  ./

- Ввод команд в строке prex.

Язык команд одинаков для обоих методов и задается в USAGE. Команды, которые что-то возвращают, не будут иметь смысла в управляющем файле. Вывод будет идти на стандартный поток вывода.

При использовании prex в целевом процессе, цель будет в запущенном или приостановленом состоянии. Это выясняется наличием или отсутствием подсказки prex>. Если подсказки нет, то целевой процесс запущен. Нажатие Control-C остановит целевой процесс и вернет пользователю подсказку. Однако, нет гарантии, что возврат подсказки произойдет сразу.

ОПЦИИ

Команда поддерживает следующие ключи:

-k	Режим ядра: В режиме ядра будут доступны дополнительные команды, но некоторые команды из других режимов станут недоступны.
-l libraries	Упомянутые библиотеки связаны в целевом процессе, использующим LD_PRELOAD. Этот ключ не может быть использован для присоединения к запущенному процессу. Аргумент ключа -l должен быть строкой, разделенной пробелами и заключенной в двойные ковычки.
-o файл_трассир	В файл будет записан вывод трассировки. Файл_трассировки считается относительно текущей рабочей директории. Если prex присоединяется к уже запущенноу процессу, то новый файл_трассировки не будет использоваться. Если имя файла не указано, то по уолчанию используется /$TMPDIR/trace-pid, где pid - ИД целевой программы. Если TMPDIR не задан, то используется /tmp.

-s размер_в_кб Максимальный размер выходного файла трассировки в кбайтах. По умолчанию, размер 4096 кбайт для нормального использования и 384 кбайт в режиме ядра. Минимальный размер составляет 128 кбайт.

СМОТРИТЕ ТАКЖЕ

ed(1), kill(1), ksh(1), ld(1)