

CRIU: сохранение временных пользовательских файлов в дампе

Студент: Святослав Власов
Руководитель: Евгений Баталов

Что такое CRIU

- Набор системных утилит для сохранения и восстановления состояния дерева процессов в Linux
- CRIU позволяет сохранять и восстанавливать:
 - Отношения процессов
 - Память процессов
 - Потoki процессов
 - Идентификаторы процессов
 - Сокеты
 - Объекты IPC
 - Пространства имен Linux, cgroup'ы
 - И многое другое
- Основная часть написана на Си



Цели и задачи практики

Цель: избежать потери восстанавливаемости дерева процессов в результате удаления одного из используемых им файлов

Задачи:

- Реализовать новую опцию `--tmp-file`, позволяющую пользователю добавлять произвольные файлы в дампы дерева процессов
- Сохранить файлы, их пути, метаданные в дампе дерева процессов
- Восстановить файлы из дампа дерева процессов во время восстановления состояния дерева процессов

<https://github.com/xemul/criu/issues/65>

Опция для добавления файла в дамп

```
INIT_LIST_HEAD(&opts.inherit_fds);
INIT_LIST_HEAD(&opts.new_cgroup_roots);
INIT_LIST_HEAD(&opts.irmap_scan_paths);
+ INIT_LIST_HEAD(&opts.tmp_files);
opts.cpu_cap = CPU_CAP_DEFAULT;
opts.manage_cgroups = CG_MODE_DEFAULT;

+ { "ghost-limit",          required_argument, 0, 1069 },
  { "irmap-scan-path",     required_argument, 0, 1070 },
  { "lsm-profile",         required_argument, 0, 1071 },
+  { "tmp-file",           required_argument, 0, 1072 },
  { },

+ case 1072:
+     if (tmp_file_add(optarg)) {
+         pr_err("Can't add tmp file %s\n", optarg);
+         return 1;
+     }
+     break;
```

Сохранение файлов в дампе

```
CR_FD_SECCOMP,  
_CR_FD_GLOB_TO,  
+ CR_FD_TMP_FILES,  
CR_FD_TMPFS_IMG,  
CR_FD_TMPFS_DEV,  
CR_FD_BINFMT_MISC,  
  
+ char **args;  
+ static char *default_args[] = {  
+     "tar",  
+     "--create",  
+     "--absolute-names",  
+     "--gzip",  
+ ...  
+     n = 0;  
+     for (; default_args[n]; n++)  
+         args[n] = default_args[n];  
+     list_for_each_entry(pos, &opts.tmp_files, list) {  
+         args[n++] = pos->filepath;  
+     }  
+     args[n] = NULL;  
  
+     struct cr_img* img = open_image(CR_FD_TMP_FILES, O_DUMP);  
+     ret = cr_system(-1, img_raw_fd(img), -1, "tar", args, 0);  
+     close_image(img);
```

Восстановление файлов из дампа

```
+ struct cr_img *img = open_image(CR_FD_TMP_FILES, O_RSTR);
+ ret = cr_system(img_raw_fd(img), -1, -1, "tar",
+             (char *[]) {"tar", "--extract", "--gzip",
+             "--no-unquote", "--no-wildcards",
+             "--absolute-names", "--directory", "/", NULL}, 0);
+ close_image(img);
```

Результаты

- Патч одобрен руководителем
- Патч отправлен команде criu на ревью
- Команда CRIU разделилась во мнениях - нужна ли новая функциональность или реализовать ее с помощью более обобщенных механизмов (pre/post-dump scripts)

Выводы:

- Участие в open-source проекте
- Знакомство с чистым Си
- Знакомство с архитектурой серьезного pure-C приложения