
Subject: [PATCH 4/6] perf: teach "perf inject" to work with files

Posted by [Andrey Vagin](#) on Wed, 07 Dec 2011 13:55:59 GMT

[View Forum Message](#) <> [Reply to Message](#)

Before this patch "perf inject" can only handle data from pipe.

I want to use "perf inject" for reworking events. Look at my following patch.

Signed-off-by: Andrew Vagin <avagin@openvz.org>

```
tools/perf/builtin-inject.c | 33 ++++++-----  
1 files changed, 31 insertions(+), 2 deletions(-)
```

```
diff --git a/tools/perf/builtin-inject.c b/tools/perf/builtin-inject.c
```

```
index 09c1061..f87779f 100644
```

```
--- a/tools/perf/builtin-inject.c
```

```
+++ b/tools/perf/builtin-inject.c
```

```
@@ -14,7 +14,12 @@
```

```
#include "util/parse-options.h"
```

```
-static char const *input_name = "-";
```

```
+static char const *input_name = "-";
```

```
+static const char *output_name = "-";
```

```
+static int pipe_output = 0;
```

```
+static int output;
```

```
+static u64 bytes_written = 0;
```

```
+
```

```
static bool inject_build_ids;
```

```
static int perf_event_repipe_synth(struct perf_tool *tool __used,
```

```
@@ -27,12 +32,14 @@ static int perf_event_repipe_synth(struct perf_tool *tool __used,  
size = event->header.size;
```

```
while (size) {
```

```
- int ret = write(STDOUT_FILENO, buf, size);
```

```
+ int ret = write(output, buf, size);
```

```
 if (ret < 0)
```

```
 return -errno;
```

```
size -= ret;
```

```
buf += ret;
```

```
+
```

```
+ bytes_written += ret;
```

```
}
```

```
return 0;
```

```
@@ -239,8 +246,14 @@ static int __cmd_inject(void)
```

```

if (session == NULL)
    return -ENOMEM;

+ if (!pipe_output)
+ lseek(output, session->header.data_offset, SEEK_SET);
    ret = perf_session__process_events(session, &perf_inject);

+ if (!pipe_output) {
+ session->header.data_size += bytes_written;
+ perf_session__write_header(session, session->evlist, output, true);
+ }
    perf_session__delete(session);

    return ret;
@@ -254,6 +267,10 @@ static const char * const report_usage[] = {
static const struct option options[] = {
    OPT_BOOLEAN('b', "build-ids", &inject_build_ids,
                "Inject build-ids into the output stream"),
+ OPT_STRING('i', "input", &input_name, "file",
+             "input file name"),
+ OPT_STRING('o', "output", &output_name, "file",
+             "output file name"),
    OPT_INCR('v', "verbose", &verbose,
              "be more verbose (show build ids, etc"),
    OPT_END()
@@ -269,6 +286,18 @@ int cmd_inject(int argc, const char **argv, const char *prefix __used)
if (argc)
    usage_with_options(report_usage, options);

+ if (!strcmp(output_name, "-")) {
+ pipe_output = 1;
+ output = STDOUT_FILENO;
+ } else {
+ output = open(output_name, O_CREAT| O_WRONLY | O_TRUNC,
+               S_IRUSR | S_IWUSR);
+ if (output < 0) {
+ perror("failed to create output file");
+ exit(-1);
+ }
+ }
+ if (symbol__init() < 0)
    return -1;

--
```

1.7.1
