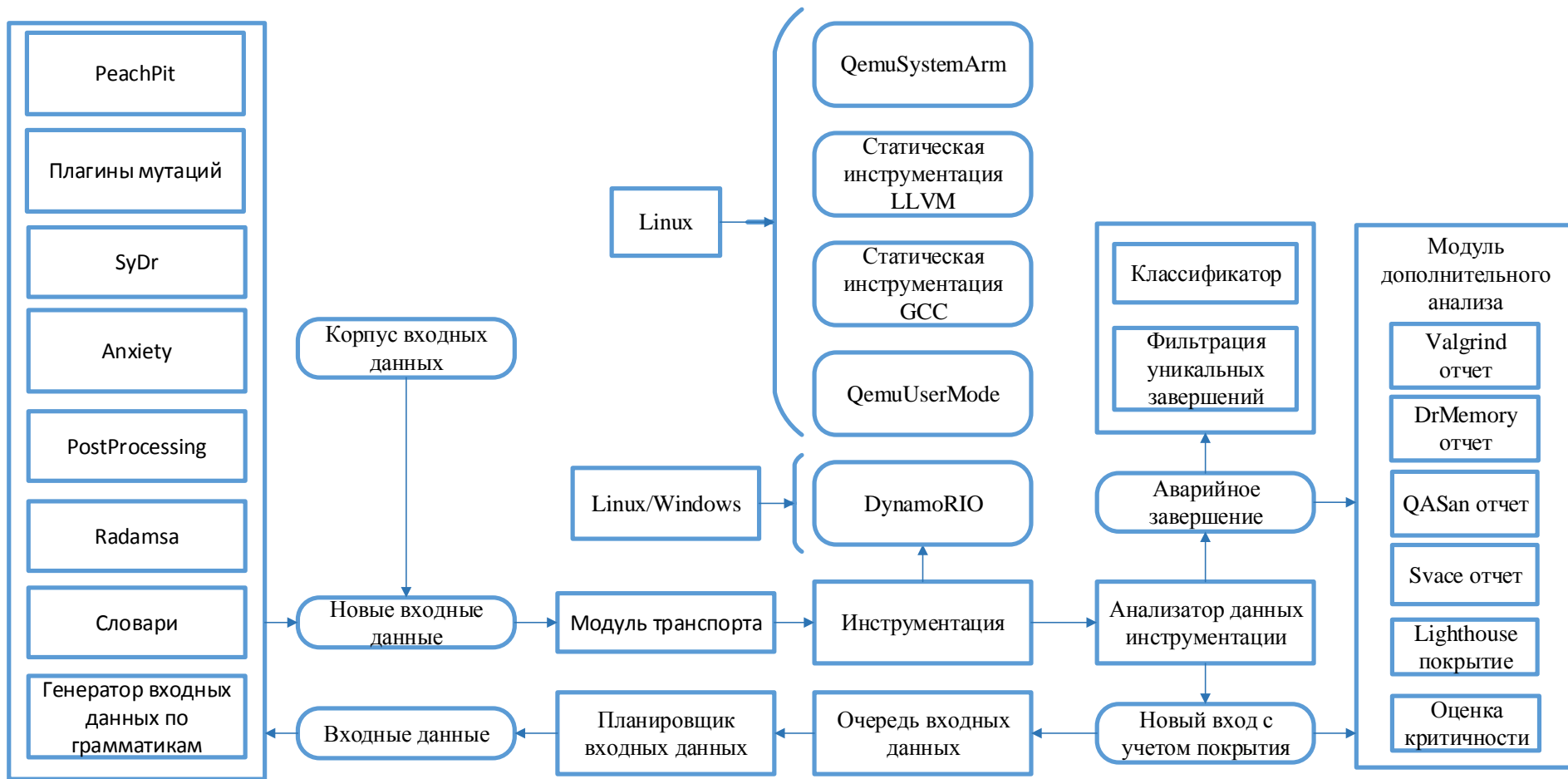


ISP Fuzzer



Ключевые возможности

- Работа в операционных системах Linux/Windows.
- Обратная связь по покрытию.
- Пользовательские мутационные преобразования.
- Модули пред и пост обработки входных данных.
- Фаззинг серверного и клиентского ПО.
- Фаззинг драйверов операционных систем Linux/Windows и встраиваемого ПО («прошивок»).
- Фаззинг на основе описания формата пакета
- Фаззинг сервисов (служб) и COM объектов
- Покрытие кода в формате Lighthouse.

Ключевые возможности

- Процент покрытых базовых блоков в ПО
- Масштабируемость до 1000 ядер.
- Пользовательские плагины отправки данных по сети.
- Интеграция с SVACE.
- Фильтрация аварийных завершений.
- Получение входных данных, на которых проявляется ошибка, размеченная в инструменте статического анализа BINSIDE.
- Использование инструмента динамического символьного выполнения Anxiety (Crusher).
- Использование инструмента динамического символьного выполнения Sydr (Crusher).

Ключевые возможности

- Наличие генератора входных данных на основе грамматик ANTLR.
- Качественная фильтрация аварийных завершений.
- Статическая и динамическая инструментации.
- Интеграция с различными средствами динамического анализа:
 - Valgrind.
 - Qsan.
 - DrMemory.

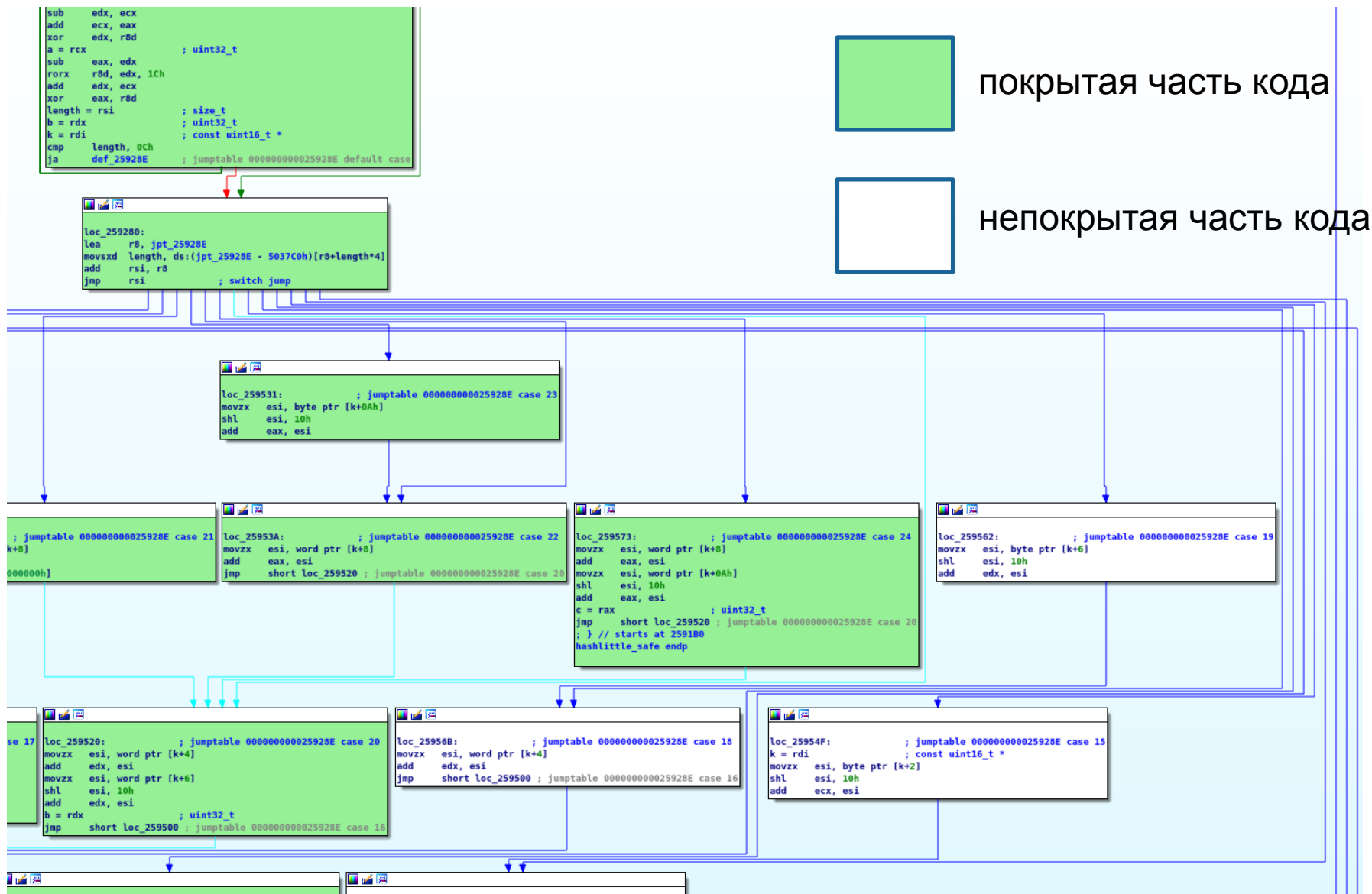
Поддерживаемые входы

- Файл
- Параметры командной строки
- Стандартный поток ввода
- Аргументы переменных окружений
- Сеть (клиент/сервер)
- Установка контекста выполнения
- Пользовательские плагины отправки данных

Поддерживаемые инструментации

- Статическая инструментация :
 - GCC/G++
 - LLVM mode
- Динамическая инструментация:
 - QemuUserMode
 - DynamoRio
 - На основе LuaQemu
 - QemuSystemARM
 - QemuSystemX86
 - QemuSystemX64
 - ...

LIGHTHOUSE покрытие



Анализ аварийного завершения

```
Starting program: /home/jenkins/projects/jasper-1.900.1/src/appl/jasper -f out/crashes/id_crash_dse_0 -t mif -F Out_tmp -T mif

Program received signal SIGABRT, Aborted.
0x00007ffff7739428 in __GI_raise (sig=sig@entry=6)
    at ../sysdeps/unix/sysv/linux/raise.c:54
54      ../sysdeps/unix/sysv/linux/raise.c: Нет такого файла или каталога.
(gdb) bt
#0  0x00007ffff7739428 in __GI_raise (sig=sig@entry=6)
    at ../sysdeps/unix/sysv/linux/raise.c:54
#1  0x00007ffff773b02a in __GI_abort () at abort.c:89
#2  0x00000000041a5e4 in mif_hdr_get (in=in@entry=0x64f4b0) at mif_cod.c:491
#3  0x00000000041a648 in mif_decode (in=0x64f4b0, optstr=0x0) at mif_cod.c:166
#4  0x000000000403341 in jas_image_decode (in=in@entry=0x64f4b0, fmt=<optimized out>,
    optstr=<optimized out>) at jas_image.c:372
#5  0x0000000004014a1 in main (argc=9, argv=<optimized out>) at jasper.c:229
```



Svrace warnings

localhost:8060/history/svace/light/index.html#show/test1/master/43138056b03d188970cbe1e458e3fd93e5882dd8/dd1a3fd3d5d7

test1 master Fri Dec 28 16:51:09 MSK 2018 Show .svres:export import

WARNING_BY_FUZZER (6)

- WARNING_BY_FUZZER (crashed-4) Program can crash here with signal 11 new
prog.c:10
- WARNING_BY_FUZZER (crashed-2) Program can crash here with signal 11 new
prog.c:10
- WARNING_BY_FUZZER (crashed-0) Program can crash here with signal 11 new
prog.c:10
- WARNING_BY_FUZZER (crashed-1) Program can crash here with signal 11 new
prog.c:5
 - [backtrace] crash1 at prog.c:5
 - fun at prog.c:25
 - main at prog.c:34

Add comment

History Undecided Unspecified
- WARNING_BY_FUZZER (crashed-3) Program can crash here with signal 11 new

/home/fedor/fuzzer/svace-test/test1/prog.c

```
1 #include <stdio.h>
2
3 void crash1(void)
4 {
5     (crashed-1) Program can crash here with signal 11 [backtrace] crash1
6     *(int*)0 = 1;
7 }
8 void crash2(void)
9 {
10    *(int*)0 = 2;
11 }
12
13 void fun(int c)
14 {
15    int s = 0;
16    int i;
17    for (i = 0; i < c % 5 + 4; i++) {
18        s += i;
19    }
20    if (c % 2 == 0)
21        s += 1;
22    if (c % 3 == 0)
23        s += 2;
```


Интеграция с SVACE

The screenshot displays the SVACE web interface. At the top, there's a browser tab titled "Svace warnings" and a URL: `localhost:8060/history/svancelight/index.html#show/test1/master/43138056b03d188970cbe1e458e3fd93e5882dd8/dd1a3fd3d5d7`. Below the browser, there are filters for "test1", "master", and a date "Fri Dec 28 16:51:09 MSK 2018". A "Show" button and a ".svres:export import" link are also visible.

The left sidebar shows a list of warnings under the heading "WARNING_BY_FUZZER (6)". The selected warning is:

- WARNING_BY_FUZZER (crashed-1) Program can crash here with signal 11 new
prog.c:5

Below this warning, a backtrace is shown:

- [backtrace] crash1 at prog.c:5
- fun at prog.c:25
- main at prog.c:34

There are also "Add comment", "History", "Undecided", and "Unspecified" buttons.

The right side of the interface shows a code editor for the file `/home/fedor/fuzzer/svace-test/test1/prog.c`. The code is as follows:

```
1 #include <stdio.h>
2
3 void crash1(void)
4 {
5     (crashed-1) Program can crash here with signal 11 [backtrace] crash1
6     *(int*)0 = 1;
7 }
8 void crash2(void)
9 {
10  *(int*)0 = 2;
11 }
12
13 void fun(int c)
14 {
15     int s = 0;
16     int i;
17     for (i = 0; i < c % 5 + 4; i++) {
18         s += i;
19     }
20     if (c % 2 == 0)
21         s += 1;
22     if (c % 3 == 0)
23         s += 2;
```

Рeach описание.

```
<Peach>
  <DataModel name="FuzzDataModel">
    <Blob name="ContentType" valueType="hex" value="16" mutable="false"/>
    <Blob name="Version" valueType="hex" value="03 01" mutable="false"/>
    <Blob name="LengthHandshakeMessage" valueType="hex" value="40 00" mutable="false"/>
    <Blob name="LengthOfChallenge" valueType="hex" value="01 00" mutable="false"/>

    <Number name="DataLength" size="16" signed="false" endian="big" mutable="false">
      <Relation type="size" of="Data"/>
    </Number>

    <Blob name="Data" valueType="hex" value="00" mutators="blob_mutator;bit_flipper_mutator"/>
  </DataModel>
</Peach>
```