

# Содержание

|                                                                   |    |
|-------------------------------------------------------------------|----|
| Прием IPFIX данных ipfixreceiver .....                            | 3  |
| <b>Введение</b> .....                                             | 3  |
| <b>Инсталляция и обновление</b> .....                             | 3  |
| CentOS6 .....                                                     | 3  |
| CentOS7 .....                                                     | 3  |
| <b>Важные изменения в версии 1.0.3 по отношению к 1.0.2</b> ..... | 4  |
| <b>Файлы поставки</b> .....                                       | 4  |
| <b>Дополнительные настройки ОС</b> .....                          | 5  |
| <b>Параметры запуска программы</b> .....                          | 6  |
| <b>Конфигурация</b> .....                                         | 6  |
| Служебные разделы .....                                           | 6  |
| logger_root .....                                                 | 6  |
| handler_ipfixreceiverlogger .....                                 | 7  |
| formatter_ipfixreceiverlogger .....                               | 7  |
| connect .....                                                     | 8  |
| dump .....                                                        | 8  |
| InfoModel .....                                                   | 8  |
| ExportModel .....                                                 | 9  |
| ExportModelFile .....                                             | 10 |
| Создаем сервис в CentOS7 .....                                    | 10 |
| <b>Проблемы и решения</b> .....                                   | 11 |



# Прием IPFIX данных ipfixreceiver

## Введение

Утилита предназначена для приема потока данных от устройств по протоколу IPFIX и сохранением данных в виде файла для последующей обработки их другими средствами.

## Инсталляция и обновление

### CentOS6

1. подключите репозиторий VAS Experts

```
rpm --import http://vasexperts.ru/centos/RPM-GPG-KEY-vasexperts.ru
rpm -Uvh
http://vasexperts.ru/centos/6/x86_64/vasexperts-repo-1-0.noarch.rpm
```

2. установите ipfixreceiver:

```
yum install -y ipfixreceiver
```

3. проверьте изменения в конфигурационных файлах на соответствие версии см. раздел "Важные изменения"

### CentOS7

1. подключите репозиторий VAS Experts

```
rpm --import http://vasexperts.ru/centos/RPM-GPG-KEY-vasexperts.ru
rpm -Uvh
http://vasexperts.ru/centos/6/x86_64/vasexperts-repo-1-0.noarch.rpm
```

2. установите репозиторий epel

```
yum -y install epel-release
```

3. установка репозитория forensics:

```
rpm --import https://forensics.cert.org/forensics.asc
rpm -Uvh
https://forensics.cert.org/cert-forensics-tools-release-el7.rpm
```

4. установите ipfixreceiver:

```
yum -y install libfixbuf --disablerepo=forensics
yum -y install netsa-python netsa_silk
```

```
yum -y install ipfixreceiver --disablerepo=forensics
```

5. проверьте изменения в конфигурационных файлах на соответствие версии см. раздел "Важные изменения"

## Важные изменения в версии 1.0.3 по отношению к 1.0.2

1. изменен конфигурационный файл в части преобразования IP адресов, с версии 1.0.3 необходимо указывать decodeipv4, decodeipv6 в экспортной модели, пример:

```
source_ip4, decodeipv4
```

```
destination_ip4, decodeipv4
```

2. сохранение информации в файл вынесено в отдельный процесс, учитывайте что при большом количестве сессий (>25k сес/сек) процесс будет полностью загружать 2 ядра процессора. Для проверки, что процесс успевает обработать весь поток данных в режиме DEBUG добавлены сообщения  
(a)cnt=NNNNN - отправлен буфер с данным номером  
(b)cnt=YYYYY - сохранен буфер с данным номером.
3. введен параметр buffer\_size - размер буфера обмена между процессом приема и записи в файл, используется в разделе [dump], по умолчанию значение параметра 100000 записей (ориентировано на 20Гбит трафика или 25 000 сессий в сек). Если количество сессий в секунду значительно меньше, то обязательно пропорционально измените данный параметр.

## Файлы поставки

1. примеры конфигурации:

```
/etc/dpiui/ipfixreceiver.conf - пример конфигурации для clickstream (http запросы)  
/etc/dpiui/ipfixreceiverflow.conf - пример конфигурации для получения информации о сессиях (аналог netflow)  
/etc/dpiui/ipfixreceiversip.conf - пример конфигурации для получения информации о sip соединениях
```

2. файлы программы располагаются в директории:

```
/usr/local/lib/ipfixreceiver.d/
```

3. вспомогательные файлы:

```
/etc/dpiui/port_proto.txt - информация о трансляции идентификатора протокола в наименование,  
используется в утилите для получения текстового имени протокола
```

4. ссылки на исполняемый модуль:

```
/usr/local/bin/ipfixreceiver -> линк на  
/usr/local/lib/ipfixreceiver.d/ipfixreceiver
```

## Дополнительные настройки ОС

### 1. настройте iptables для приема внешних данных

Для работы ipfixreceiver'a требуется открыть порты которые так же будут использоваться в конфигурации в разделе [connect]

Например вами используются протокол TCP, 1500 порт и IP=212.12.11.10

```
[connect]  
protocol=tcp  
host=212.12.11.10  
port=1500
```

Для приема IPFIX потока у вас в /etc/sysconfig/iptables должно быть следующее правило:

```
-A INPUT -p tcp -m state --state NEW -m tcp --dport 1500 -j ACCEPT
```

Не забудьте что после внесения правила в iptables требуется перезапуск:

```
service iptables restart
```

### 2. настройте ротацию логов

Пример ротации для лог файла /var/log/dpiuiflow.log, создайте в директории /etc/logrotate.d/ файл flowlog следующего содержания

```
/var/log/dpiui*.log {  
    rotate 5  
    missingok  
    notifempty  
    compress  
    size 10M  
    daily  
    copytruncate  
    nocreate  
    postrotate  
    endscript  
}
```

Обратите внимание на использование метода copytruncate, иначе файл будет пересоздан и запись лога из процесса прекратится.

Соответственно в конфигурации ipfixreceiver у вас в разделе [handler\_ipfixreceiverlogger] указано следующее:

```
args=('/var/log/dpiuiflow.log', 'a+')
```

3. Настройте удаление старых файлов. Например, удаление старых архивов (более 31 дня) с записями о сессиях запакованных gzip:

```
15 4 * * * /bin/find /var/dump/dpiui/ -name url_*.dump.gz -cmin +44640  
-delete > /dev/null 2>&1
```

Измените строчку под ваши требования и добавьте в файл `/var/spool/cron/root`.

## Параметры запуска программы

Утилита `ipfixreceiver` имеет следующие параметры запуска:

```
usage: ipfixreceiver start|stop|restart|status|-v [-f <config file>]
```

где

- `start` - запуск в режиме сервиса
- `stop` - останов сервиса
- `state` - состояние работы сервиса
- `restart` - перезапуск сервиса
- `-v` - вывести информацию о версии
- `-f <config file>` - указать файл конфигурации для запуска сервиса

Пример:

```
ipfixreceiver start -f /etc/dpiui/ipfixreceiverflow.conf
```

## Конфигурация

По умолчанию используется файл конфигурации `/etc/dpiui/ipfixreceiver.conf`.

📌 Больше информации о конфигурировании логирования можно найти по ссылке [Logging](#)

### Служебные разделы

1. `loggers` - определяет используемые лог идентификаторы
2. `handlers` - определяет используемые обработчики для сохранения лога
3. `formatters` - определяет используемые форматы для лога

### logger\_root

1. `level` - определяет уровень логирования (верхний уровень)

Возможные значения:

|                       |                                                            |
|-----------------------|------------------------------------------------------------|
| <code>CRITICAL</code> | - только критические ошибки, минимальный уровень сообщений |
| <code>ERROR</code>    | - включая ошибки                                           |
| <code>WARNING</code>  | - включая предупреждения                                   |
| <code>INFO</code>     | - включая информацию                                       |

```
DEBUG      - включая отладочные
NOTSET     - Все, максимальный уровень сообщений (включая все выше
перечисленные)
```

Пример:

```
level=DEBUG
```

2. handlers - используемые обработчики сообщений

Пример:

```
handlers=ipfixreceiverlogger
```

## handler\_ipfixreceiverlogger

1. class - класс обработчика

Пример:

```
class=FileHandler
```

2. level - уровень сообщений

```
level=DEBUG
```

3. formatter - наименование формата сообщений

```
formatter=ipfixreceiverlogger
```

4. args - параметры обработчика

```
args=('/var/log/dpiuiflow.log', 'a+')
```

## formatter\_ipfixreceiverlogger

1. format - описание формата сообщения

Пример:

```
format=%(asctime)s - %(name)s - %(levelname)s - %(message)s
где
%(name)s      - имя лога
%(levelname)s - уровень сообщения ('DEBUG', 'INFO', 'WARNING', 'ERROR',
'CRITICAL').
%(asctime)s   - дата, по умолчанию формат "2003-07-08 16:49:45,896" (поле
запятая указаны миллисекунды).
%(message)s   - сообщение
```

2. datefmt - описание формата даты

Пример:

```
datefmt='%m-%d %H:%M'
```

## connect

1. protocol - протокол (tcp или udp).

```
protocol=udp
```

2. host - IP или имя сервера.

```
host=localhost
```

3. port - номер порта.

```
port=9996
```

## dump

1. rotate\_minutes - период в минутах, по прошествии которого временный файл в dumpfiledir/<port>.url.dump будет перемещен в архив (mv) и создан новый временный файл.

```
rotate_minutes=10
```

2. processcmd - команда, которая будет запущена по окончании ротации файла, параметр имя файла с путем к нему.

```
processcmd=gzip %s
```

3. dumpfiledir - директория куда будут сохраняться файлы с принятыми данными.

```
dumpfiledir=/var/dump/dpiui/ipfixflow/
```

4. buffer\_size - размер буфера обмена между процессом приема и записи в файл, по умолчанию значение параметра 100000 записей (ориентировано на 20Гбит трафика или 25 000 сессий в сек). Если количество сессий в секунду значительно меньше, то обязательно пропорционально измените данный параметр.

## InfoModel

Блок описывает получаемые данные по IPFIX протоколу.

1. InfoElements - параметр с описанием элементов информационной модели для IPFIX

```
InfoElements =  octetDeltaCount,      0,      1, UINT64, True
                packetDeltaCount,     0,      2, UINT64, True
                protocolIdentifier,    0,      3, UINT8
                session_id,            43823, 2000, UINT64, True
```



где,

session\_id - наименование поля из описания IPFIX см. разделы  
43823 - уникальный номер организации (enterprise number)

1 - уникальный номер поля

UINT64 - тип поля

True - использовать обратный порядок байт (endian). Значения - True или пусто.

Типы полей:

| Type         | Length | Type IPFIX           |
|--------------|--------|----------------------|
| OCTET_ARRAY  | VARLEN | octetArray           |
| UINT8        | 1      | unsigned8            |
| UINT16       | 2      | unsigned16           |
| UINT32       | 4      | unsigned32           |
| UINT64       | 8      | unsigned64           |
| INT8         | 1      | signed8              |
| INT16        | 2      | signed16             |
| INT32        | 4      | signed32             |
| INT64        | 8      | signed64             |
| FLOAT32      | 4      | float32              |
| FLOAT64      | 8      | float64              |
| BOOL         | 1      | boolean              |
| MAC_ADDR     | 6      | macAddress           |
| STRING       | VARLEN | string               |
| SECONDS      | 4      | dateTimeSeconds      |
| MILLISECONDS | 8      | dateTimeMilliseconds |
| MICROSECONDS | 8      | dateTimeMicroseconds |
| NANOSECONDS  | 8      | dateTimeNanoseconds  |
| IP4ADDR      | 4      | ipv4Address          |
| IP6ADDR      | 16     | ipv6Address          |

Наименование полей и описание можно взять по ссылкам:

1. [Шаблон экспорта Netflow в формате IPFIX](#)
2. [Шаблоны экспорта clickstream и SIP](#)
3. [Шаблон экспорта AAA в формате IPFIX](#)

Дополнительная информация:

[Information Model for IP Flow Information Export](#)

## ExportModel

определяет параметры модели для экспорта, зарезервировано для будущего использования.

1. Mode - тип используемого экспорта

```
Mode = File
```

## ExportModelFile

Описание модели экспорта File.

1. Delimiter разделитель полей в строке ( \t - табуляция, еще примеры - |,;) )

```
Delimiter = \t
```

2. ExportElements - описание полей которые будут сохранены в файл.

```
ExportElements = timestamp, seconds, %%Y-%%m-%%d %%H:%%M:%%S.000+03
                  login
                  source_ip4
                  destination_ip4
                  host, decodehost
                  path, decodepath
                  referral, decodereferer
                  session_id
```

где поля в каждой строке:

имя - наименование поля из информационной модели [InfoModel] (login, session\_id и т.п.)

обработчик - процедура обработки поля перед выводом

|               |                                                    |
|---------------|----------------------------------------------------|
| seconds       | - поле в секундах, ожидается формат                |
| milliseconds  | - поле в миллисекундах, микросекундах,             |
| nanoseconds   | наносекундах ожидается формат                      |
| decodehost    | - перекодировать из punycode в UTF-8               |
| decodepath    | - перекодировать из urlencoding в UTF-8            |
| decodereferer | - перекодировать из (punycode,urlencoding) в UTF-8 |
| decodeproto   | - перекодировать идентификатор протокола в строку  |

формат - описание формата для seconds, milliseconds.

Пример: %%Y-%%m-%%d %%H:%%M:%%S.%%f+0300

Результат: 2016-05-25 13:13:35.621000+0300

## Создаем сервис в CentOS7

Создание сервиса в CentOS7 по шагам, название сервиса **ipfix1**, используемая конфигурация **/etc/dpiui/ipfixreceiver.conf**, используемый порт **1500**.

Создаем файл /etc/systemd/system/ipfix1.service следующего содержания:

```
[Unit]
Description=ipfix test restart
After=network.target
After=syslog.target
```

```
[Service]
Type=forking
PIDFile=/tmp/ipfixreceiver.1500.pid
ExecStart=/usr/local/bin/ipfixreceiver start -f
/etc/dpiui/ipfixreceiver.conf
ExecStop=/usr/local/bin/ipfixreceiver stop -f /etc/dpiui/ipfixreceiver.conf
ExecReload=/usr/local/bin/ipfixreceiver restart -f
/etc/dpiui/ipfixreceiver.conf
Restart=always
RestartSec=10s

[Install]
WantedBy=multi-user.target
```

Выполняем:

```
systemctl enable ipfixl.service
systemctl start ipfixl.service
systemctl daemon-reload
```

Проверяем:

```
systemctl status ipfixl.service -l
```

 **не забудьте проверить поднятие сервиса после перезагрузки**

## Проблемы и решения

1. как получить версию утилиты?  
Используйте следующие команды:

```
ipfixreceiver -v
```

```
yum info ipfixreceiver
```

2. можно ли на один порт отправлять IPFIX потоки с разных DPI?  
Да. Единственное в записываемом потоке их будет не различить.
3. как понять, что утилита работает?
  - а) проверьте, что порт из конфигурации прослушивается утилитой, например 1500:

```
netstat -nlp | grep 1500
```

б) проверьте лог, нет ли ошибок

с) Проверьте, что запись в промежуточный файл происходит, например для 9996 порта (директория для файлов - /var/dump/dpiui/ipfixurl):

```
tail -f /var/dump/dpiui/ipfixurl/9996.url.dump
```

4. все проверено, но приема сообщений нет?

- a) забыли открыть порт в iptables.
  - b) инициализировали ipfixreceiver с неверным IP сервера.
5. с DPI идет большое количество сессий (более 2 млн сессий/мин), при включенном DEBUG режиме видно, что счетчик обмена буферами не успевает записать до получения следующего блока записей, что можно сделать?
- a) удалите преобразование даты в строку, это уменьшит процессорное время на обработку и дополнительно получите уменьшение объема результирующего файла
  - b) удалите преобразование decodeipv4, незначительно, но так же получите ускорение записи файла
  - c) настройте buffer\_size при к-ве сес /сек более 30к совместно с п.d
  - d) увеличьте частоту процессора и объем памяти