Unix Content Filtering Proxy

Руководство пользователя

© Компания «Риланс»
Тел. (343) 375-78-62

http://www.relans.ru
mailto:support@relans.ru

1. Общая информация

1.1 О программе

Unix Content Filtering Proxy (UCFP) является собственной коммерческой разработкой компании «Риланс».

UCFP не является продуктом Лаборатории Касперского и только использует «антивирусный движок» Kaspersky Anti-Virus (KAV), поэтому обращение к Лаборатории Касперского с вопросами, касающимися этого продукта не будут иметь результата.

Вся необходимая информация о UCFP содержится в данной документации и на сайте проекта (http://www.ucfp.ru/).

1.2 Область применения продукта

Продукт предназначен для:

1. Фильтрации интернет-контента с использованием баз данных (регулярно пополняемые списки URL, которые можно автоматически загружать с ftp-сервера компании «Риланс») по 31 категории (порно, спорт, музыка и т.д.)
2. Анализа содержания web-страниц с помощью метода весовых коэффициентов по категориям.
3. Антивирусной проверки интернет-трафика (http и ftp) средствами Антивируса Касперского.
4. Лимитирования трафика пользователей за определенный промежуток времени
5. Ведения статистики с указанием стоимости по заданным категориям трафика (внешний, внутригородской и т.д.)

Продукт представляет собой полнофункциональный непрозрачный некэширующий proxy-сервер, и может работать как совместно с другим proxy (в частности squid), так и самостоятельно.

1.3 Схемы работы

Возможны следующие схемы работы UCFP:

1. LAN -> UCFP -> internet
2. LAN -> UCFP -> ISP Proxy -> internet
3. LAN -> proxy -> UCFP -> internet
4. LAN -> UCFP -> proxy -> internet
5. LAN -> proxy -> UCFP -> ISP Proxy -> internet
6. LAN -> UCFP -> proxy -> ISP Proxy -> internet

1.4 Поддерживаемые операционные системы

Продукт функционирует на операционных системах:

- Linux
- FreeBSD
- OpenBSD

Продукт тестировался и работал стабильно на следующих операционных системах (OS):

- Linux RedHat 7.2 – 9.0, RedHat Enterprise 1 – 4, Fedora Core 1 – 4
- Linux Debian 3.0, 3.1
- Linux Mandrake 9.2 – 10.1, Mandriva 2006
- SuSE Linux 9.3 – 10.0
- ASPLinux
- FreeBSD 4.x – 6.x

- OpenBSD 3.4 – 3.9

Разработчики считают, что никаких противопоказаний к использованию UCFP на любом другом Linux или других версиях FreeBSD/OpenBSD нет.

Другие программные требования приведены в разделе «FAQ: Минимальные требования».

Компания «Риланс» будет крайне признательна за любую информацию о функционировании UCFP на любой (в том числе и неподдерживаемой) OS, а также за информацию о всех замеченных ошибках (см. раздел «Обратная связь»).

1.5 Основные возможности продукта

1. Антивирусная проверка трафика в режиме удержания
2. Анализ содержания web-страниц на основе подсчета терминов с весовыми коэффициентами
3. Поддержка предустановленных баз и шаблонов по 31 категории
4. Создание и использование собственных шаблонов для блокировки доступа к сайтам с определенным содержанием
5. Управление доступом клиентов в интернет
6. Оповещение администратора о любых попытках обращения к запрещенным ресурсам
7. Установка лимита трафика
8. Ведение статистики

Преимущества использования UCFP:

1. Блокировка проникновения вирусов через интернет
2. Лимитирование доступа пользователей к ресурсам интернета
3. Экономия трафика
4. Мониторинг активности работы пользователей в интернете и выявление нарушителей

1.6 Схема совместной работы с антивирусным daemon-ом

1. Все файлы, которые проходят через UCFP сохраняются во временном каталоге
2. Антивирусному daemon-у через сокет дается команда на проверку каждого файла
3. После окончания проверки антивирусный daemon возвращает UCFP результат
4. На основе результата проверки принимается решение о возможности доставить проверенный файл по назначению
5. В случае обнаружения в проверенном файле вируса, UCFP предпринимает определенные, заданные администратором, действия

См. раздел «Примечания: Обновление антивирусных баз»

1.7 Базы данных

Базы данных - это регулярно пополняемые списки URL, которые можно автоматически загружать с ftp-сервера компании «Риланс».

Продукт содержит утилиту «обратной связи», что позволяет получать постоянно обновляемые базы данных по всем поддерживаемым категориям. Данная утилита может быть использована для отправки на сервер компании «Риланс» ссылок на подозрительные ресурсы, обнаруженные за время работы UCFP. Полученные таким образом ссылки проверяются и вносятся в базу данных. Обновление баз данных планируется не реже 2 раз в неделю.

1.8 Обратная связь

Компания «Риланс» будет признательна:

- за информацию об обнаруженных в продукте ошибках (см. раздел «FAQ: Вопросы в support»)
- за замечания и пожелания по улучшению функционала продукта
- за поправки к документации
- за отрицательные и положительные отзывы
- за списки ресурсов по поддерживаемым категориям

Адрес для обратной связи support@ucfp.ru

Мы будем стараться ответить на **любое и каждое** Ваше письмо в течение не более 2-х часов в рабочее время (с 7-00 до 16-00 московского времени).

Существует список рассылки новостей о проекте, подписаться на который можно, отправив письмо на news@ucfp.ru

2. Инсталляция

2.1 Состав дистрибутива

Все дистрибутивы продукта доступны на сайте в разделе downloads

Важно: в состав дистрибутива UCFP **не входят** файлы KAV. Если у Вас на сервере уже установлен KAV for File/Mail Servers, то необходим только дистрибутив UFCP. Если KAV не используется, то его «облегченный» (только демон aveserver и updater, без сканера и т.п.) вариант с триальным ключом также можно скачать с сайта (раздел downloads).

Рабочий каталог программы **/usr/local/ucfp**

Состав:

/usr/local/ucfp/bin/ucfp - исполняемый файл прокси-сервера

/usr/local/ucfp/bin/ucfpdb – скрипт для получения и отправки списков и баз данных

/usr/local/ucfp/bin/config – скрипт начальной настройки основных параметров

/usr/local/ucfp/bin/genstat – исполняемый файл для генерации статистики

/usr/local/ucfp/bin/maketpl – исполняемый файл для генерации шаблонов

/usr/local/ucfp/bin/limdump – исполняемый файл для генерации отчетов о лимитах трафика

/usr/local/ucfp/modules – каталог с модулями программы

/usr/local/ucfp/etc – каталог конфигурационных файлов

/usr/local/ucfp/errors – каталог с html-шаблонами сообщений об ошибках

/usr/local/ucfp/parts – каталог для хранения слов

/usr/local/ucfp/internal – каталог внутреннего http-сервера

/usr/local/ucfp/notifies – каталог с примерами скриптов для оповещения администратора

/usr/local/ucfp/quarantine – каталог карантина

/usr/local/ucfp/db – каталог для хранения баз данных

/usr/local/ucfp/tpl – каталог для хранения шаблонов фильтрации

В дистрибутив для Linux также входят необходимые для работы продукта системные библиотеки (каталог /usr/local/ucfp/lib)

См. раздел «Примечания: Собственные наборы иконок для ftp»

2.2 Инсталляция

Программные требования приведены в разделе «FAQ: Минимальные требования».

Продукт поставляется в пакетах и архивах.

Установка пакетов:

Linux rpm-based – **rpm –i ucfp-x.x.x-x.rpm**

Linux debian – **dpkg –i ucfp-x.x.x-x.deb**

FreeBSD/OpenBSD – **pkg_add ucfp-x.x.x-x.tgz**

Установка из архива:

tar xvfz ucfp-x.x.x-x.tgz –C /usr/local

Прежде чем запустить конфигурационный скрипт /usr/local/ucfp/bin/config и, отвечая на его вопросы, задать основные параметры UCFP, мы рекомендуем Вам ознакомиться со следующим разделом, чтобы получить полное представление о существующих возможностях продукта.

3. Конфигурационный файл

Конфигурационный файл (**/usr/local/ucfp/etc/config**) состоит из секций (названия секций задаются в квадратных скобках):

[ucfp.common] – общие настройки

[ucfp.logs] – настройки логов

[ucfp.hold] – настройки схемы удержания

[ucfp.check] – настройки параметров антивирусной проверки

[ucfp.notify] – настройки оповещения администратора

[ucfp.license] – настройки лицензирования

[ucfp.control] – настройки управления

[ucfp.limits] – настройки лимитов трафика

[ucfp.stat] – настройки ведения статистики

3.1 Секция common (общие настройки)

Мы рекомендуем схему работы №3, где под proxy в большинстве случаев подразумевается squid.

Преимущества схемы №3:

- Кэширование трафика.
- При выключении UCFP squid автоматически переключится на работу «напрямую».

Параметры этой секции:

listen = address:port Интерфейс и порт, который «слушает» UCFP. Рекомендуется «слушать»:

локальный интерфейс если UCFP является единственным прокси на сервере (на firewall следует разрешить коннекты из LAN на используемый UCFP порт)

127.0.0.1 если ufcp используется по схемам № 3 и 5 (LAN -> squid -> UCFP -> [ISP Proxy] -> internet) и squid находится на этом же сервере (см. раздел FAQ: Настройка squid).

Если *«пусто»*, то = *default*.

Примечание: Может быть или один интерфейс или *, т.е. все.

*Default: *:12345*

peer_address = address:port Если параметр задан, то UCFP перенаправляет все запросы на указанные адрес и порт.

Примечание: Актуален для схем работы № 2,4,6. Если же UCFP работает по другим схемам (отправляет запросы напрямую в Internet), то параметр пустой.

Default: «пусто»

behavior = stop/pass Параметр, указывающий на то, как поступит UCFP в случае, если по какой-либо причине не получит ответа от kavdaemon-а (например если kavdaemon не запущен).

stop – вместо запрашиваемого файла клиенту будет возвращена ошибка

pass – пропустит файл непроверенным. Если *«пусто»*, то = *default*

Default: pass

db_swap = yes/no Использование свопинга для баз данных

Примечание: Значение этого параметра влияет на объем занимаемой памяти и скорость обработки запросов (см. разделы «FAQ: Минимальные требования» и «FAQ: Скорость работы»).

Default: yes

errors_templates_folder = foldername Папка с html-шаблонами об ошибках («Virus found», «URL is blocked», «Server reject query» и т.д.). Все возможные шаблоны входят в дистрибутив.

Default: /usr/local/ucfp/errors

db_folder = foldername Папка для хранения баз данных.

Default: /usr/local/ucfp/db

tpl_folder = foldername Папка для хранения предустановленных шаблонов

Default: /usr/local/ucfp/tpl

tmp_folder = foldername Папка для хранения временных списков запрета доступа

Default: /usr/local/ucfp/lists.tmp

quarantine = foldername Папка карантина, куда помещаются копии всех обнаруженных вирусов. Если *«пусто»*, то карантин не используется.

Примечание: Карантин ведется в следущем виде, например, для файла ftp://ftp.relans.ru/test/eicar.com будет создана папка /usr/local/ucfp/quarantine/ftp.relans.ru и в нее будет помещен файл eicar.com.

Default: /usr/local/ucfp/quarantine

ctl_socket = filename Управляющий сокет UCFP.

Default: /var/run/ucfp.ctl

kav_version = 4/5 Версия KAB. Поддерживаются версии 4.x и 5.x.

Default: 5

kav_socket = filename Сокет KAB демона. Для версии 5 по умолчанию /var/run/aveserver, для версии 4 всегда /var/run/AvpCtl

Default: /var/run/aveserver

hostname = IP_address – имя сервера, на котором запущен UCFP. Используется для запросов к внутреннему http-серверу UCFP, который генерирует сообщения об ошибках («отказано в доступе», «обнаружен вирус» и т.п.). Рекомендуется использовать IP адрес локального интерфейса.

internal_address = address – URL для запросов к внутреннему http-серверу UCFP, который генерирует сообщения об ошибках («отказано в доступе», «обнаружен вирус» и т.п.). В отличии от *hostname* возможно указание любого адреса и протокола.

Default: значение параметра hostname

tmp_dir = foldername Временная папка. Если *«пусто»*, то = *default*

Default: /tmp

db_swap_dir = foldername Папка для свопинга баз данных.

Default: значение параметра tmp_dir

forwarded_for = yes/no/строка

yes – UCFP отдаст дальше полученное значение хэдера X-Forwarded-For

no или *«пусто»* - хэдер не передается

строка - хэдера X-Forwarded-For передается с указанным значением

Если UCFP работает по схемам №3 и 5, то см. раздел «Примечания: хэдер X-Forwarded-For»

Default: «пусто»

connect_ports = номера портов, к которым разрешен метод CONNECT через UCFP. См. раздел «Примечания: Перечисление значений параметров» . 5190 – используется ICQ, 443 – SSL.

Default: 443, 5190

nameservers = IP_address – IP-адреса DNS-серверов. Если параметр не указан, то IP-адреса DNS-серверов берутся из /etc/resolv.conf.

Default: «пусто»

resolver_timeout = кол-во секунд ожидания ответа от DNS-сервера.

Default: 5

resolver_tries = кол-во попыток опроса каждого DNS-сервера.

Default: 2

enable_dns_lookup = yes/no Проверка IP-адресов в базах данных.

Default: «yes»

3.2 Секция logs (настройки логов)

Формат файла отчета уровня 0:

Дата время IP_клиента логин_клиента код_ответа/тип_запроса IP_сервера URL тип_контента размер_файла статус_проверки [результат_проверки] [кто_проверял] [информация]

Поле «статус_проверки» может принимать значения:

yes - проверка была произведена.

no – проверки не было (поля «кто_проверял», «результат_проверки» и «информация» пустые)

Поле «результат_проверки» может принимать значения:

allow – проверка была произведена, доступ разрешен.

block – данный ресурс содержится в одном из черных списков/баз или, в результате анализа контента, данная страница была отнесена к одной из категории фильтрации. Поле «информация» содержит или название вируса, или наименование листа/базы, или суммарный балл анализатора.

Поле «кто_проверял» может принимать значения:

KAV – антивирусная проверка

CF – анализатор контента

См. раздел «Примеры: Примеры отчета»

Параметры этой секции:

append = yes/no Добаление в файл лога после рестарта программы. Если *«пусто»*, то = *default*.

yes – файл логов не обнуляется

no – при старте программы лог начинает вестись заново

Default: yes

logfile = filename Файл отчета. Если *«пусто»*, то лог не ведется.

Default: /var/log/ucfp.log

loglevel = Числовой показатель уровня глубины логирования (от 0 до 3). Чем выше числовой показатель, тем более детально в логах описывается работа программы (например, на уровне 2 предоставляется детальная информация о работе анализатора контента). Использование ненулевого уровня в обычном режиме работы не рекомендуется, т.к. это заметно уменьшает быстродействие.

Default: 0

show_all = yes/no

yes – в отчет пишется информация о всех прошедших через UCFP файлах

no – только информация о зараженных и заблокированных объектах.

Если *«пусто»*, то = *default*.

Default: yes

banners = yes/no Запись в отчет информации о блокировках баннеров.

Если *«пусто»*, то = *default*.

Default: yes

3.3 Секция hold (настройки схемы удержания)

Работа схемы удержания

Основным преимуществом предлагаемого продукта (UCFP) является возможность работы в режиме удержания.

Проиллюстрируем работу схемы удержания на **примере**:

Клиент посылает запрос на скачивание файла размером 10Мб, скорость скачивания 6Мб/мин (1Мб/10 сек), значение удержания равно 20%. Закачка происходит в два этапа:

Первый этап: составляет 20 сек. В течение этого времени скачиваются 20% (2Мб) запрашиваемого файла, которые сохраняются во временную папку на сервере. В течение 20 секунд клиент не получает информации от UCFP, пользователь видит на экране застывший индикатор прогресса.

После получения первых 20% файл размером 2Мб проверяется kavdaemon-ом и:

в случае обнаружения в нем вредоносного кода, дальнейшее скачивание прекращается, клиенту отдается html-страница с соответствующим сообщением, предпринимаются некоторые действия, указанные администратором;

если в первых 2Мб вируса нет, то скорость передачи данных клиенту вычисляется, исходя из следующих соображений: необходимо «растянуть» передачу клиенту полученных 2Мб настолько, чтобы за это время успеть скачать оставшиеся 8Мб. Скорость закачки примерно известна - оставшиеся 8Мб будет получено за 80 секунд. Итого получается, что UCFP должен отдать клиенту 2Мб за 80 секунд, т.е. скорость отдачи клиенту первой части запрашиваемого им файла будет составлять 25Кб/сек.

Второй этап: продолжается 80 секунд и состоит из двух параллельных процессов:

UCFP отдает клиенту скачанные в первом этапе 2Мб со скоростью 25Кб/сек (на мониторе пользователя медленно двигается индикатор прогресса закачки, с точки зрения пользователя все выглядит как «плохой интернет-канал») и одновременно докачивает оставшиеся 8Мб.

После окончания докачки полученный UCFP конечный файл размером в 10Мб проверяется kavdaemon-ом и если вирус в нем не обнаружен отдается с высокой скоростью клиенту.

Основные преимущества схемы удержания:

- Используя процент удержания, администратор имеет возможность регулировать время, проведенное его пользователями перед неподвижным экраном. Опыт внедрения существующих продуктов, использующих 100%-ное удержание, показывает, что если пользователь не получает ответа на запрос в течение 10-20 секунд, он решает, что ресурс просто недоступен и отменяет запрос, при этом в большинстве случаев прокси докачивает файл до конца и потом его удаляет.
- Известно, что Антивирус Касперского в большинстве случаев способен обнаружить вирус в первых 5-20Кб файла, поэтому проверка в два этапа может сэкономить трафик.
- Режим 100% удержания не исключен, но UCFP предоставляет администратору выбор.
- У администратора есть возможность регулировать минимальный размер файлов, к которым будет применена данная схема (параметр hold_min_size), т.к. проверять в два этапа файлы размером 10-100Кб нет необходимости.
- Администратор имеет возможность задать размер первого этапа в байтах (параметр hold_size).

Примечание: В единичных случаях запрашиваемый сервер не отдает UCFP размер файла. В такой ситуации существующие на рынке продукты оставляют клиента (т.е. пользователя) перед неподвижным экраном до завершения закачки и проверки файла антивирусом. В UCFP администратор имеет возможность применять к подобным файлам схему удержания, используя параметр hold_min_ size, указывающий размер первого этапа в байтах. Таким образом пользователь через короткое время видит, что произошла закачка некоторой части запрашиваемого файла и, следовательно, не отменит запрос.

Параметры этой секции:

known_size_hold_perc = процент удержания при известном размере запрашиваемого файла. Значения «пусто», 100 и 0 – удержание не используется. При этом если hold_size не равен 0, то используется **размер** удержания.

Default: 20

hold_size = размер удержания запрашиваемого файла в байтах. Значения *«пусто»* и *0* – удержание не используется. Если размер файла известен и known_size_hold_perc не равен 0, то используется **процент** удержания. Рекомендуемое значение в диапазоне от 100000 до 500000 (выбор конкретного значения зависит от скорости канала).

Default: 200000

hold_min_size = минимальный размер (файла в байтах), при котором используется схема удержания. Значения *«пусто»* и *0* – ограничения нет, т.е. схема используется всегда на любом файле.

Default: 150000

not_hold_ext = список расширений файлов, для которых будет использоваться схема удержания, но не будет производится антивирусная проверка первой части файла. Рекомендуется перечислить все известные расширения архивов, т.к. проверить антивирусом часть архива невозможно. Перечисление через «,». * означает «не выполнять проверку первой части для всех файлов». См. раздел «Примечания: Перечисление значений параметров»

Default: arj,cab,gz,lzh,rar,tar,tgz,zip,bz2,sbz2,stbz2,bzip,iso

3.4 Секция check (настройки параметров антивирусной проверки)

См. разделы «Примечания: Перечисление значений параметров» и «Примечания: Типы контента»

Параметры этой секции:

max_check_size = числовой параметр Файлы больше указанного размера не будут передаваться антивирусу для проверки. *«пусто»* - параметр не используется.

Default: «пусто»

block_suspicious = yes/no Блокировать доступ к ресурсам, которым в результате антивирусной проверки был присвоен статус «подозрительный».

Default: no

check_ext - расширения файлов, которые будут проверены антивирусом.

Default: «пусто»

check_ct – типы контента файлов, которые будут проверены антивирусом.

Default: «пусто»

check_ranged = yes/no Необходимость антивирусной проверки запросов с хедером Range (см. раздел «Многопоточная закачка»).

*Default: «*yes*»*

3.5 Секция notify (настройки оповещения администратора)

Параметры этой секции:

license_notify_script = filename Файл скрипта для оповещения о лицензировании.

mail_notify - для оповещения администратора по e-mail

popup_notify - для оповещения администратора по winpopup

Если *«пусто»* оповещения не используются.

Default: /usr/local/ucfp/notifies/mail_notify

admin_addr = IP или e-mail адрес администратора для отправки оповещений. *«пусто»* - оповещения не используются.

Default: «пусто»

notify_timeout = числовое значение таймаута в секундах, в течение которого не будет повторных нотификации. Этот параметр полезен, если Вы используете оповещение администратора по email.

Default: 5

См. раздел «Примечания: Переменные скрипта для извещения администратора»

3.6 Секция license (настройки лицензирования)

Параметры этой секции:

keyfile = filename Ключевой файл.

Поддерживаются два типа лицензирования:

- по количеству ip-адресов клиентов
- по объему трафика

При лицензировании по количеству ip-адресов клиентов UCFP считает количество уникальных IP-адресов клиентов, от которых получены запросы, при этом запросы от клиентов, порядковый номер которых превышает количество купленных лицензий, не проверяются, о чем в лог вносится соответствующая запись (также возможно оповещение администратора).

Важно: Без ключевого файла UCFP не проверяет каждый третий запрос (См. раздел «Примечания: Ограничения пробной версии»).

notify_threshold = числовое значение, которое задает ту разницу между достигнутым значением счетчика лицензий или трафика и их количеством, при которой администратор получит соответствующее уведомление.

0 – оповещение при равенстве. *«пусто»* - оповещение не используется.

Default: 1

check_clients = filename Список лицензионных ip-адресов. В этот список UCFP автоматически добавляет все лицензированные ip-адреса. Если *«пусто» = default.*

Default: /usr/local/ucfp/etc/clients/checked

auto = yes/no

yes - автоматически дополнять check_clients при поступлении запросов от новых клиентов

no – не менять check_clients, т.е. работать с вручную сформированным администратором списком лицензированных IP адресов.

Default: yes

dhcp_mode = yes/no

yes - автоматически очищать check_clients раз в сутки (только при auto=yes)

Default: no

3.7 Секция control (настройки управления)

Параметры этой секции:

objects = filename Файл с описанием объектов (См. раздел «Описание объектов»)

Default: /usr/local/ucfp/etc/objects

lists = filename Файл с описанием категорий (См. раздел «Описание категорий»)

Default: /usr/local/ucfp/etc/lists

rules = filename Файл с описанием правил (См. раздел «Правила доступа»)

Default: /usr/local/ucfp/etc/rules

ucfpdb = filename Файл с настройками обновлений (См. раздел «Настройка обновлений»)

Default: /usr/local/ucfp/etc/ucfpdb

limits = filename Файл с правилами лимитирования трафика (См. раздел «Лимиты трафика»)

Default: /usr/local/ucfp/etc/limits

unlimited = filename Файл с серверами, на которые нет лимита доступа (См. раздел «Лимиты трафика»)

Default: /usr/local/ucfp/etc/unlimited

3.8 Секция stat (настройка статистики)

Для генерации статистики предназначен исполняемый файл /usr/local/ucfp/bin/genstat.

Параметры этой секции:

dir = foldername Папка для хранения файлов статистики.

Default: /usr/local/ucfp/stat

cols = filename Файл с описанием колонок статистики.

Default: /usr/local/ucfp/etc/stat

rows = ip/login значение строк статистики, *ip* – IP-адрес клиента, *login* - логин клиента.

Default: ip

count_cache = yes/no Считать в статистике трафик из кэша от следующего за UCFP прокси. Актуален для схем работы № 2,4,6.

Default: no

html = yes/no Статистические данные представляются в формате html.

Default: yes

Описание категорий трафика (колонок статистики) задается в файле, указанном в конфигурационном файле, в секции [ucfp.stat], параметр cols (по умолчанию */usr/local/ucfp/etc/stat)*

Описания категорий трафика задаются в формате «параметр = значение;». Завершающий символ «;» обязателен. Перечисление значений указывается через знак «,».

Общие принципы описания категорий трафика:

1. Описания категории трафика начинается с параметра name.
2. Перечисление параметров возможно как в одну строку, так и в несколько.
3. В описании категории трафика обязательным является только параметр name.
4. При определении категорий трафика можно использовать следующие параметры:

- **name = имя** категории трафика (латинские буквы без пробелов);
- **urls = список серверов** для данной категории трафика (в качестве значений для объектов данного типа можно так же указывать описание подсети в формате x.x.x.x/x или диапазон IP-адресов в формате x.x.x.x-x.x.x.x)
- **price = число**, означающее стоимость 1Мб трафика для данной категории;
- **comment = комментарий** (строка без каких-либо ограничений);

5. При определении значения параметра urls допустимо использование объектов типа urls в формате $object_name, где object_name - значение параметра name объекта.
6. При пустом файле с описанием категорий статистика не ведется.
7. Прохождение описаний категорий трафика осуществляется сверху вниз, до первого удовлетворения параметров запроса определенной категории.
8. Заголовком колонок статистики является значение параметра comment (если параметр comment не задан, то значение параметра name) из описания категории трафика.

9. При формировании колонки client происходит поиск объектов типа clients или logins с value равным значению строки статистики. Если такой объект найден, то вместо IP-адреса или логина в статистике указывается значение поля comment объекта.
10. Стоимость указывается в скобках после объема трафика.

Пример:

Имеются следующие **объекты** (файл objects):

name = ip01; type = clients; value = 10.0.0.1; comment = Директор;

name = ip02; type = clients; value = 10.0.0.2; comment = Василий Пупкин;

name = ip04; type = clients; value = 10.0.0.4; comment = Федор Сумкин;

Конфигурационный файл config, секция **[ucfp.stat]**

dir = /usr/local/ucfp/stat

cols = /usr/local/ucfp/etc/stat

rows = ip

html = no

Описание категорий трафика (файл **/usr/local/ucfp/etc/stat**):

name = isp;

urls = my_provider.ru;

price = 0.01;

comment = Провайдер;

name = in;

urls = 222.222.222.0/19;

price = 0.2;

comment = Городской;

name = all;

price = 2;

comment = Внешний;

Статистика за определенный промежуток будет выглядеть следующим образом:

	Провайдер	Городской	Внешний	Всего
Директор	42 (0.42)	20 (4.00)	10 (20.00)	72 (24.42)
Василий Пупкин	10 (0.10)	10 (2.00)	15 (30.00)	35 (32.10)
10.0.0.3	30 (0.30)	15 (3.00)	18 (36.00)	63 (39.30)
Федор Сумкин	58 (0.58)	25 (5.00)	32 (64.00)	115 (69.58)
Всего	140 (1.40)	70 (14.00)	75 (150.00)	285 (165.40)

Важно: Порядок описания категорий трафика имеет большое значение, например при:

name = all; price = 2; comment = Внешний;

name = isp; urls = my_provider.ru; price = 0.01; comment = Провайдер;

name = in; urls = 222.222.222.0/19; price = 0.2; comment = Городской;

весь трафик будет принадлежать категории all (внешний).

3.9 Описание объектов

Объект – некоторая совокупность параметров одного типа. Объекты используются в правилах управления доступом и нужны исключительно для удобства понимания и администрирования.

Описание объектов задается в файле, указанном в конфигурационном файле, в секции [ucfp.control], параметр objects (по умолчанию */usr/local/ucfp/etc/objects)*

Объекты описываются в формате «параметр = значение;». Завершающий символ «;» обязателен. Перечисление значений указывается через знак «,».

Общие принципы описания объектов:

1. Описания объекта начинается с параметра name.
2. Перечисление параметров возможно как в одну строку, так и в несколько.
3. Объект характеризуется совокупностью следующих параметров:

- **name = имя** объекта (латинские буквы без пробелов);
- **type = тип** объекта (одно из предустановленных значений);
- **value = значение**, характеризующее объект (зависит от type);
- **comment = комментарий** (строка без каких-либо ограничений);

4. Все параметры кроме comment являются обязательными.

Значение параметра **type** может быть одним из:

- **urls = список URL** запрашиваемых ресурсов
- **clients - список IP-адресов** клиентов (в качестве value для объектов данного типа можно так же указывать описание подсети в формате x.x.x.x/x или диапазон IP-адресов в формате x.x.x.x-x.x.x.x)
- **logins - список логинов** клиентов
- **ct – типы контента** запрашиваемого файла
- **ext – расширение** запрашиваемого файла
- **method – HTTP метод** запроса (OPTIONS, GET, HEAD, POST, PUT, DELETE, TRACE или CONNECT)
- **notify – скрипт** оповещения

(См. раздел «Примеры: Примеры описания объектов»)

3.10 Описание категорий

Категория – описание сайтов определенной тематики. Используются для запрета/разрешения доступа в правилах. Поддерживаются 26 предустановленных категорий:

Название	Тематика
viruses	Сайты содержащие вредоносный код (вирусы)
porno	Порно и эротические сайты, секс-шопы
humor	Юмористические сайты (анекдоты, шутки, розыгрыши, комиксы и т.п.)
music	Музыка и танцы
video	Сайты о кино (продажа и обзоры видеофильмов, кинотеатры, киноафиши, сайты киноактеров, анимэ, мультфильмы)
games	Игры (прохождения и обзоры игр, чит-коды, online-игры, online-казино)
sport	Спортивные сайты
animals	Сайты о природе (животные, птицы, рыбы, охота, рыбалка и т.д.)

books	Книги, литература, сайты писателей
referats	Рефераты, дипломные работы, шпаргалки
warez	Варез (cracks, serials, hacking, p2p и т.п.)
travel	Путешествия (туристические агентства, туризм, отели, санатории, пансионаты)
cigar	Сайты по продаже сигарет
medic	Сайты о медицине (здоровье, продажа медикаментов, препараты для увеличения различных частей тела)
cars	Сайты автомобильной тематики
dating	Службы знакомств
pictures	Коллекции картинок, скринсэйверов, обоев для рабочего стола, постеров
beauty	Косметика, парфюмерия, украшения, макияж, мода, женское белье и т.п.
job	Сайты по поиску работы
chat	Чаты и irc
phones	Полифония (мелодии, игры и картинки для сотовых телефонов)
food	Кулинария (рецепты, напитки, рестораны)
horo	Гороскопы
ecards	Электронные поздравительные открытки
rest	Сайты для отдыха (дайвинг, бильярд, боулинг, ночные клубы, хобби, коллекционирование и т.д.)
amuse	Сайты развлекательного характера (развлекательные порталы и домашние странички)
webmail	Почтовые web сервера
proxies	Открытые прокси-сервера и Web-анонимайзеры
banners	Баннерообменные сети
icq	Сервера сервисов обмена сообщениями (ICQ, AOL Instant Messenger, MSN Messenger, Yahoo Messenger, Google Talk, Rambler ICQ, Mail.ru Agent, Jabber и т.п.)
white	«Белые» сайты (kernel.org, squid-cache.org, novell.com, microsoft.com и т.д.)

Существуют следующие методы детектирования принадлежности сайта к определенной категории:

1. База данных (db)
2. Список характерных слов в URL сайта (parts) (См. также раздел «Примечания: Отрицательные parts»)
3. Анализатор контента на основе подсчета слов с весовыми коэффициентами (tpl)

Описание категорий задается в файле, указанном в конфигурационном файле, в секции [ucfp.control], параметр lists (по умолчанию */usr/local/ucfp/etc/lists).* По сути категории являются объектами типа list, поэтому указание в описании type=list обязательно.

Категории описываются в формате «параметр = значение;». Завершающий символ «;» обязателен. Перечисление значений указывается через знак «,».

Общие принципы описания категорий:

1. Описания категории начинается с параметра name.
2. Перечисление параметров возможно как в одну строку, так и в несколько.
3. Предустановленные категории характеризуются совокупностью следующих параметров:

- **name = имя** категории (латинские буквы без пробелов);
- **parts = список слов,** при обнаружении которых в доменном имени или URL запрашиваемого ресурса доступ к нему будет заблокирован (См. также раздел «Примечания: Отрицательные parts»).

  Default: */usr/local/ucfp/parts/listname,* где listname = имя категории.

- **db = filename,** файлы баз данных категории. Маска файлов – filename*.db

Default: *db_folder/listname,* где db_folder = значение параметра db_folder конфигурационного файла в секции [ucfp.common], listname = имя категории.

- **tpl = filename,** файлы шаблонов (перечень характерных для данной категории слов с весовыми коэффициентами). Маска файлов – filename*.tpl.

Default: *tpl_folder/listname,* где tpl_folder = значение параметра tpl_folder конфигурационного файла в секции [ucfp.common], listname = имя категории.

- **tmp_list = filename** Файл текущего (рабочего) черного списка. В этот список программа автоматически заносит все ресурсы, на которых за время работы обнаружила вирусы, или ресурсы, доступ к которым был заблокирован в результате анализа контента. Попытка обращения к ресурсам из данного списка приводит к ошибке blocked с соответствующим диагнозом (если *«пусто»*, то не используется).

Default: *tmp_folder/listname,* где $tmp_folder = значение параметра tmp_folder конфигурационного файла в секции [ucfp.common], listname = имя категории.

- **block = db, tpl, dns_parts/url_parts, tmp**

  Значение параметра может состоять из:

  db - доступ к ресурсу блокируется на основе баз данных ресурсов

  dns_parts или url_parts - доступ к ресурсу блокируется на основе слов из файла parts данной категории в доменном имени запрашиваемого сервера (*dns_parts*) или во всем URL запрашиваемого ресурса (*url_parts*). См. также раздел «Примечания: Отрицательные parts».

  tpl - доступ к ресурсу блокируется по результатам проверки на вирусы средствами kavdaemon (для категории viruses) или на основе подсчета весовых коэффициентов

  tmp - соответствующая запись будет сделана в tmp_list, если он используется

- **threshold = числовое значение** суммы коэффициентов поисковых фраз, при котором запрашиваемая страница относится к данной категории.

  Изменяя значение этого параметра можно регулировать качество анализа контента.

  Значение *0* или *«пусто»* - анализ контента не производится.

  Default:1000

- **reason = строка**, которая указывается в поле Reason html-страницы «отказано в в доступе».

  Default:listname, где listname = имя категории.

- **comment = комментарий** (строка без каких-либо ограничений);
- type = тип, всегда равен list

4. Все параметры кроме type и name и action являются необязательными

   (См. раздел «Примеры: Примеры описания категорий»)

3.11 Правила доступа

Правило доступа – некоторая совокупность параметров для запрета или разрешения доступа к запрашиваемому ресурсу.

Правила доступа задается в файле, указанном в конфигурационном файле, в секции [ucfp.control], параметр rules (по умолчанию */usr/local/ucfp/etc/rules)*

Правила задаются в формате «параметр = значение;». Завершающий символ «;» обязателен. Перечисление значений указывается через знак «,».

Общие принципы описания правил:

1. Описания правила начинается с параметра do.
2. Перечисление параметров возможно как в одну строку, так и в несколько.
3. В правиле обязательным является только параметр do
4. При определении правила можно использовать следующие параметры:

- **do = deny/allow/pass**

  deny - запретить доступ

  allow - разрешить доступ

  pass – пропустить правило (правило неактивно)

- **urls = список URL** запрашиваемых ресурсов
- **clients - список IP-адресов** клиентов. В качестве значения для данного параметра можно так же указывать описание подсети в формате x.x.x.x/x, или диапазон IP-адресов в формате x.x.x.x-x.x.x.x
- **logins - список логинов** клиентов
- **ct – типы контента** запрашиваемого файла
- **ext – расширения** файлов
- **method – HTTP метод** запроса
- **notify – скрипт** оповещения
- **size – максимальный размер файла** в байтах
- **lists – категории** для проверки
- **comment = комментарий** (строка без каких-либо ограничений);

5. При определении значений параметров urls, clients, logins, ct и ext допустимо использование объектов в формате $object_name, где object_name - значение параметра name объекта. При этом тип объекта должен соответствовать параметру правила
6. При определении значений параметра lists допустимо только описание категорий в формате $object_name, где object_name - значение параметра name категории.
7. Пустой файл с описанием правил означает «разрешить всем и все без каких-либо проверок», т.е. do = allow; является последним правилом по умолчанию
8. Прохождение правил осуществляется сверху вниз, до первого удовлетворения параметров запроса значениям параметров правила. Если запрос не удовлетворяет ни одному из правил, то действует правило по умолчанию. т.е. доступ к запрашиваемому ресурсу разрешается
9. Сравнение параметров запроса и параметра правила осуществляются через логическое «и». Исключение составляют пары параметров ct/ext и urls/lists, сравнение которых происходит через логическое «или». Также через логическое «или» происходят проверки внутри категории, т.е. проверка по категории читается как: если запрашиваемый URL находится в базе данных (db) или в URL содержится запрещенное слово (parts) или анализатор контента отнес страницу к данной категории (tpl).
10. Если в правиле указаны urls и lists одновременно, то ресурсы, перечисленные в urls, заносятся в базу данных первой в списке lists категории, т.е. блокировка таких ресурсов происходит с диагнозом «ресурс находится в базе данных категории».

    (См. раздел «Примеры: Примеры правил»)

3.12 Настройка обновлений

На сайте компании «Риланс» доступны обновления:

1. Баз данных, в том числе антивирусных (db)
2. Шаблонов для анализатора контента (tpl)

Настройки обновлений задаются в файле, указанном в конфигурационном файле, в секции [ucfp.control], параметр ucfpdb (по умолчанию */usr/local/ucfp/etc/ucfpdb)*

Правила задаются в формате «параметр = значение;». Завершающий символ «;» обязателен. Перечисление значений указывается через знак «,».

У настроек обновления только один параметр – имя категории. Значения параметра:

ip_db, strip_db или full_db - скачивать утилитой ucfpdb с ftp компании «Риланс» обновленный ip-ориентированный (*ip_db*), оптимизированный (*strip_db*) или полный (*full_db*) набор баз для данной категории (См. раздел «Примечания: Типы баз»)

tpl – скачивать утилитой ucfpdb с ftp компании «Риланс» обновленный вариант шаблонов анализатора контента для данной категории (шаблоны доступны для всех категорий кроме banners и webmail). Для категории viruses шаблонами являются антивирусные базы.

send - отправлять утилитой ucfpdb tmp-лист данной категории на сервер компании «Риланс»

Важно: После обновления из tmp- списков удаляются имеющиеся в базах данных записи, поэтому рекомендуется производить обновление баз данных до отправки tmp-списков в компанию «Риланс». Например занести в /etc/crontab следующие записи:

45 23 * * * root /usr/local/ucfp/bin/ucfpdb –get

50 23 * * * root /usr/local/ucfp/bin/ucfpdb --send

См. также «Примеры: Примеры настроек обновления» «Примечания: Обновление антивирусных баз» и « Примечания: Типы баз»

3.13 Лимиты трафика

Лимитирование трафика производится на основе правил лимита.

Правило лимита – некоторая совокупность параметров для ограничения количества трафика пользователей в определенный промежуток времени.

Правила лимита задается в файле, указанном в конфигурационном файле, в секции [ucfp.control], параметр limits (по умолчанию */usr/local/ucfp/etc/limits)*

Правила задаются в формате «параметр = значение;». Завершающий символ «;» обязателен. Перечисление значений указывается через знак «,».

Общие принципы описания правил:

1. Описания правила начинается с параметра limit.
2. Перечисление параметров возможно как в одну строку, так и в несколько.
3. В правиле обязательным является только параметр limit.
4. При определении правила можно использовать следующие параметры:

- **limit = кол-во мегабайт трафика** («пусто» означает «ограничения нет»)
- **time – временной интервал,** в формате: *число[символ]*, где символ может принимать следующие значения:

  «не задан» - *число* означает количество дней

  «w» - *число* означает количество недель

  «m» - *число* означает количество месяцев

  Значение «0» означает «разрешить на неопределенный срок».

- **clients - список IP-адресов** клиентовВ качестве значения для данного параметра можно так же указывать описание подсети в формате x.x.x.x/x, или диапазон IP-адресов в формате x.x.x.x-x.x.x.x

- **logins - список логинов** клиентов
- **comment = комментарий** (строка без каких-либо ограничений);

5. При определении значений параметров clients и logins допустимо использование объектов в формате $object_name, где object_name - значение параметра name объекта. При этом тип объекта должен соответствовать параметру правила
6. Пустой файл с описанием правил лимитов означает «ограничений нет»
7. Прохождение правил осуществляется сверху вниз, до первого удовлетворения параметров запроса значениям параметров правила. Если запрос не удовлетворяет ни одному из правил, то действует правило по умолчанию. т.е. ограничения не накладываются
8. Сравнение параметров запроса и параметра правила осуществляются через логическое «и»
9. В файле, указанном в конфигурационном файле в секции [ucfp.control], параметр unlimited (по умолчанию */usr/local/ucfp/etc/unlimited),* можно задавать сервера, трафик которых не учитывается в лимитах.

   Для просмотра данных по лимитам предназначена утилита /usr/local/ucfp/bin/limdump.

   (См. раздел «Примеры: Примеры лимитов»)

4. Удаленное управление

5. Ключи командной строки

Исполняемый файл /usr/local/ucfp/bin/ucfp имет следующие ключи командной строки:

–v – версия продукта

-h – помощь по ключам командной строки

-P filename - pid-файл, по умолчанию /var/run/ucfp.pid

-f – запуск в foreground mode

–r – загрузка измененной конфигурации

–R – ротация лога

-T – проверка файла конфигурации

-t – сохранить tmp листы

-l – загрузить tmp листы

–c filename – использование альтернативного конфигурационного файла

-g filename - сохранение в указанный файл реальных параметров, с которыми при данном конфигурационном файле будет работать UCFP. Опция полезна для выявления неправильно заданных параметров в конфигурационном файле.

-s – отключение порта статистики.

По умолчанию для получения статистики слушается порт listen_port+1, и, выполнив

telnet localhost listen_port+1

можно получить статистику по работе UCFP в виде:

```
-=> ucfp status [build: 0253] <=-
  Total threads runned: 7
  Total requests recived: 671
  Descriptors used 7 of 1024 (0%)
  Total bytes recived: 8398709
  Total bytes send   : 8401103
  Uptime: 0 day(s) 14:53:23
  Traffic saved: 0 (bytes)
-=> License info <=-
 License owner       : Relans
 License UID         : 0100-af3467-eeff-632548
 License type        : IP addrs
 License valid until : 03/09/2004 11:40:05
 Connections used    : 2
 Connections remain  : 98
 Name         DB     TMP     TPL     BLACK     WHITE
 viruses      0      0       0       0         0
 ..
 webmail        0      0       0       0         0
-=> Lists RECORDS stats <=-
Name         DB     TMP     TPL     BLACK     WHITE
 viruses     31     0       0       0         0
 ..
 webmail        352    0       0       0         0
```

6. Примеры

6.1 Примеры описания объектов

Для более удобного описания объектов создадим несколько файлов:

/home/admin/good_sites

www.good-site1.ru

www.good-site2.ru

www.good-site3.ru

/home/admin/music_by_ct

audio/midi

audio/mpeg

audio/x-realaudio

audio/x-wav

audio/x-mpegurl

/home/admin/do_log

#!/bin/sh

echo "$UCFP_DATE IP:$UCFP_IP URL:$UCFP_URL RES:$UCFP_RESULT Size:$UCFP_SIZE" >> /home/admin/ucfp_log

Приведем несколько примеров описания объектов, которые в дальнейшем будут использоваться в примерах правил:

name = mysite; type = urls; value = www.mydomain.ru;
comment = Мой сайт;

name = good-sites; type = urls; value = /home/admin/good_sites;
comment = Хорошие сайты;

name = bad-ip; type = urls; value = 111.111.111.111;
comment = Плохой ip-адрес;

name = my_network; type = clients; value = 10.0.0.0/8;
comment = Моя сеть;

name = some-users; type = clients; value = 10.0.0.2-10.0.0.22;
comment = Некоторые пользователи;

name = admin; type = clients; value = 10.0.0.1;
comment = IP аминистратора;

name = bad-user; type = clients; value = 10.0.0.13;
comment = Нехороший пользователь;

name = bosses; type = logins; value = boss,chief,papa;
comment = Директорат (логины);

name = music_ext; type = ext; value = mp3,wav,m3u;
comment = Расширения музыкальных файлов;

name = music_ct; type = ct; value = /home/admin/music_by_ct;
comment = Музыкальные типы контента;

name = pictures; type = ext; value = gif,jpg;
comment = Картинки;

name = log_notify; type = notify; value = /home/admin/do_log;
comment = Лог нарушителей;

6.2 Примеры описания категорий

Приведем несколько описаний предустановленных категорий:

name = viruses; type = list; block = db, tpl, tmp;

Блокировать вирусы по базе данных (db) и по результатам проверки антивируса (tpl), добавлять записи об обнаруженных вирусах в tmp-лист (tmp).

name = banners; type = list; block = db, url_parts;

Блокировать баннеры по базе данных баннерообменных сетей (db), а также блокировать все URL содержащие слова banner и baner (parts)(содержимое файла /usr/local/ucfp/parts/banners)

name = sport; type = list; block = db, tpl, url_parts, tmp;

Блокировать сайты спортивной тематики по базе данных (db) и по результатам проверки контента (tpl), добавлять записи об обнаруженных анализатором контента спортивных сайтах в tmp-лист (tmp).

name = humor; type = list; block = db;

Блокировать юмористические сайты только по базе данных (db).

Приведем несколько примеров создания **собственных категорий**:

1. Администратор хочет использовать в правилах управления доступом ресурсы, содержащие в URL слово forum. Для этого необходимо создать следующую категорию:

   name = forums; type = list; parts = forum; block = url_parts; comment = Форумы;

2. Администратор считает, что пользователи не должны посещать сайты, на которых более 2-х раз встречается слово badword. Для запрета доступа необходимо создать файл-шаблона для анализатора контента (См. раздел Примеры: Пример создания шаблона) с одним словом badword1 = 1, далее с помощью утилиты maketpl скомпилировать файл-шаблон например /home/admin/my.tpl, и создать категорию:

   name = badword_sites;

   type = list;

   tpl = /home/admin/my.tpl;

   threshold = 3;

   block = tpl;

   reason = Уважаемый пользователь! Доступ к этой странице запрещен, т.к. она содержит слово badword более двух раз;

   comment = Сайты содержащие badword;

3. Сотрудникам компании должен быть запрещен доступ к сайтам конкурента (компания «Enemy»), а так же к сайтам рекламирующим и продающим продукцию конкурентов (продукты «Enemy_food», «Enemy_stuff» и «Enemy_drink»), при этом все сайты, содержащие запрещенный контент, неизвестны.

Для блокировки нужно создать шаблон со словами enemy, enemy_food, enemy_stuff и enemy_drink и назначить им некоторые весовые коэффициенты (См. раздел Примеры: Пример создания шаблона), с помощью утилиты maketpl скомпилировать файл /home/admin/enemy.tpl, а также перечислить известные сайты конкурента в файле /home/admin/enemy.db. Далее описать категорию:

   name = enemy_sites;

type = list;

tpl = /home/admin/enemy.tpl;

db = /home/admin/enemy.db;

threshold = 100;

parts = enemy;

block = tpl, db, url_parts;

reason = Сайты компании enemy;

comment = Категория enemy;

4. Администратор считает, что существующий анализатор контента недостаточно «жестко» блокирует порно-ресурсы, и желает для этих целей использовать свой шаблон /home/admin/my_porno.tpl, а также использовать стандартные для UCFP базы данных порно-ресурсов. Для этого администратор может создать свою категорию:

   name = my_porno;

   type = list;

   tpl = /home/admin/my_porno.tpl;

   db = porno;

   threshold = 100;

   block = db, tpl, dns_parts, tmp;

   reason = Порно;

   comment = Моя порнолочилка;

   или исправить параметр tpl у предустановленной категории porno:

   name = my_porno;

   type = list;

   tpl = /home/admin/my_porno.tpl;

   block = db, tpl, dns_parts, tmp;

6.3 Примеры правил

Важно: В правилах используются примеры объектов и категории рассмотренные выше.

Метод прочтения правила:

Запретить (**do = deny**) или разрешить (**do = allow**) доступ с клиентских компьютеров с IP-адресами **clients** и с логинами пользователей **logins** к запрашиваемому ресурсу размером более **size** байт, если происходит попытка применить **method**, а также ресурс принадлежит к одной из категорий **lists** или ресурс имеет адрес **urls**, тип контента **ct** или расширения файла **ext**. Помимо запрета/разрешения доступа выполнить скрипт **notify**. Перечисляемые значения каждого из параметров читаются через «или».

Если параметр не задан, то соответствующую ему часть правила нужно опустить.

Приведем несколько **простейших однострочных правил**:

1. Осуществлять только антивирусную проверку всего трафика, используя настройки категории viruses, в случае обнаружения вируса доступ запретить и выполнить скрипт $mail (оповещение администратора по e-mail), иначе доступ разрешить:

   do = deny; lists = $viruses; notify = $mail;

2. Запретить доступ ко всем ресурсам сайтов спортивной тематики, при попытке доступа выполнить скрипт, описанный объектом log (т.е. добавить соответствующую запись в файл /home/admin/my_log):

   do = deny; lists = $sport; notify = $log;

3. Запретить показ картинок на порно-ресурсах, используя расширения графических файлов, указанные в объекте pictures:

   do = deny; lists = $porno; ext = $pictures;

4. Полностью запретить метод POST:

   do = deny; method = POST;

5. Лишить компьютеры с IP-адресами 10.0.0.10 и 10.0.0.11 доступа в интернет:

   do = deny; clients = 10.0.0.10, 10.0.0.11;

6. Запретить пользователю с логином bad_user скачивать файлы более 1000000 байт:

   do = deny; logins = bad_user; size = 1000000;

7. Запретить доступ ко всем ресурсам сайта www.bad-site.com и ко всем сайтам, имеющим IP-адрес объекта bad-ip:

   do = deny; urls = www.bad-site.com, $bad-ip;

8. Отнести сайт bad-site.com к порно-ресурсам, т.е. к категории porno:

   do = deny; urls = www.bad-site.com; lists = $porno;

9. Запретить пользователям с логинами vasya и pupkin с компьютеров, принадлежащих подсети 10.0.0.0/8, скачивать с музыкальных и порно сайтов файлы, размером более 1000000 байт, и расширениями указанными в объекте music_ext или типами контента указанными в объекте music_ct:

   do = deny; clients = 10.0.0.0/8; logins = vasya, pupkin; lists = $music, $porno; ct = $music_ct; ext = $music_ext; size = 1000000;

Используя метод прочтения, правило из последнего примера можно прочитать как: запретить доступ с клиентских компьютеров подсети 10.0.0.0/8 и с логинами пользователей vasya или pupkin к запрашиваемому ресурсу размером более 1000000 байт, если ресурс принадлежит к категории music или porno, имеет тип контента, описанный в объекте music_ct или расширение, описанное в объекте music_ext запроса.

Также большое значение имеет **порядок следования правил**. Приведем несколько примеров:

1. Проверять на вирусы весь трафик, кроме компьютера с IP-адресом 10.0.0.1:

   do = allow; clients = 10.0.0.1;

   do = deny; lists = $viruses;

2. Проверять весь трафик на вирусы. После антивирусной проверки запретить всем доступ к сайтам всех категорий, но разрешить пользователю с логином boss доступ к порно-ресурсам:

   do = deny; lists = $viruses;

   do = allow; lists = $porno; login = boss;

   do = deny; lists = $porno, $amuse, $humor, $music, $video, $games, $sport, $books, $warez, $travel, $cigar, $medic, $cars, $dating, $pictures, $beauty, $job, $chat, $phones, $food, $horo, $ecards, $webmail, $banners;

3. Запретить доступ в интернет всем компьютерам, кроме подсети, описанной в объекте my_networks, трафик которых проверять на вирусы:

   do = deny; clients = $my_network; lists = $viruses;

   do = allow; clients = $my_network;

   do = deny;

4. Использовать по прямому назначению категорию white (не блокировать сайты данной категории и не проверять их на вирусы) и запрещать доступ к сайтам всех остальных категорий:

   do = allow; lists = $white;

do = deny; lists = $viruses; $porno, $amuse, $humor, $music, $video, $games, $sport, $books, $warez, $travel, $cigar, $medic, $cars, $dating, $pictures, $beauty, $job, $chat, $phones, $food, $horo, $ecards, $webmail, $banners;

5. Разрешить всем компьютерам, кроме администратора (объект admin), доступ только к сайтам, указанным в файле /home/admin/good_sites. Администратор должен иметь полный доступ ко всему:

   do = allow; clients = $admin;

   do = allow; urls = /home/admin/good_sites;

   do = deny;

6. Запретить всем, кроме пользователей с логинами из объекта bosses качать архивы (расширения exe,zip,rar) более 5000000 байт. Все архивы проверять на вирусы.

   do = deny; logins = $bosses; ext = exe,zip,rar; lists = $viruses;

   do = allow; logins = $bosses; ext = exe,zip,rar;

   do = deny; size = 5000000;

7. Запретить всем, кроме пользователей с логинами из объектов bosses и admin, оставлять сообщения на любых сайтах:

   do = allow; logins = $bosses, $admin; method = POST;

   do = deny; method = POST;

8. Разрешить всем пользователям читать форумы и чаты , но оставлять сообщения в них разрешить только с компьютеров, перечисленных в объекте some-users:

   do = allow; lists = $chat, $forums; clients = $some-users; method = POST;

   do = deny; lists = $chat, $forums; method = POST;

Использование объектов в правилах совершенно необязательно. Т.е. можно задавать значения параметров правил и в явном виде:

do = allow; lists = $chat, $forums; clients = 10.0.0.2-10.0.0.22; method = POST;

do = deny; clients = 10.0.0.2-10.0.0.22; method = POST;

но в этом случае, когда понадобится, например, добавить еще один IP-адрес, которому разрешено оставлять сообщения в форумах и чатах придется исправлять 2 строки, а в случае использования объекта только одну.

Между понятиями «не проверять по определенной категории» и «разрешить доступ к сайтам определенной категории» есть некоторая разница. Например, для компьютера с IP-адресом 10.0.0.1 существует следующее правило:

do = deny; clients = 10.0.0.1; lists = $sport, $book;

Для того, чтобы **не проверять** запросы с 10.0.0.1 по категории sport (спортивные сайты) нужно просто убрать $sport, т.е.:

do = deny; clients = 10.0.0.1; lists =$book;

В этом случае доступ к сайту sport-referats.ru (рефераты о спорте) будет запрещен категорией books.

Для того, чтобы **разрешить** доступ с 10.0.0.1 ко всем сайтам категории sport, включая рефераты о спорте, нужно добавить разрешающее правило:

do = allow; clients = 10.0.0.1; lists = $sport;

do = deny; clients = 10.0.0.1; lists =$book;

В этом случае доступ к сайту sport-referats.ru будет разрешен.

6.4 Примеры настроек обновлений

viruses = strip_db, tpl, send;

Отсылать утилитой ucfpdb –-send tmp-лист с ресурсами, на которых во время работы были обнаружены вирусы, в компанию «Риланс» (send) и обновлять базу данных вирусных сайтов (strip_db) и антивирусные базы (tpl) с помощью утилиты ucfpdb –-get.

porno = strip_db, tpl, send;

Отсылать утилитой ucfpdb –-send порно tmp-лист в компанию «Риланс» (send) и обновлять базу данных (strip_db) и списки слов для анализатора контента (tpl) порно-сайтов с помощью утилиты ucfpdb –-get.

banners = strip_db;

Обновлять базы баннерообменных сетей с помощью утилиты ucfpdb –-get (strip_db)

См. также «Примечания: Обновление антивирусных баз» и «Примечания: Типы баз»

6.5 Примеры лимитов

Важно: В правилах используются примеры объектов и категорий рассмотренные выше.

Метод прочтения правила лимита:

Установить ограничение по трафику с клиентских компьютеров с IP-адресами **clients** и с логинами пользователей **logins** в **limit** мегабайт за промежуток времени **time**. Перечисляемые значения каждого из параметров читаются через «или».

Если параметр не задан, то соответствующую ему часть правила нужно опустить.

Приведем несколько примеров **правил лимита**:

1. Установить ограничение трафика в 100Мб/месяц для всех:

limit = 100; time = 1m;

2. Разрешить всем скачивать не более 10Мб в день, кроме компьютера с IP-адресов объекта admin, для которого ограничения нет:

limit = 0; clients = $admin;

limit = 10; time = 1;

3. Установить ограничение 50Мб/неделю для всех и разрешить пользователю с логином vasya скачать 100Мб единовременно:

limit = 100; logins = vasya;

limit = 50; time = 1w;

4. Не устанавливать ограничений для логинов из объекта bosses. Компьютерам с IP-адресами 10.0.0.8 и 10.0.0.12 установить ограничение 10Мб/месяц. Компьютерам с IP-адресами, описанными в объекте some-users, установить лимит 50Мб/неделю. Всем остальным компьютерам из подсети, описанной в объекте my_networks, установить ограничение 100Мб/месяц.

limit = 0; logins = $bosses;

limit = 10; time = 1m; clients = 10.0.0.8, 10.0.0.12;

limit = 50; time = 1w; clients = $some-users;

limit = 100; time = 1m; clients = $my_network;

6.6 Конфигурация по умолчанию

6.6.1 Объекты

По умолчанию в файле /usr/local/ucfp/etc/objects определены следующие объекты:

Расширения и Content-Types:

Для баннерорезалки:
name = banners_ext; type = ext; value = /usr/local/ucfp/etc/ext/banners;
name = banners_ct; type = ct; value = /usr/local/ucfp/etc/ct/banners;

Для запрета музыкальных файлов:
name = music_ext; type = ext; value = /usr/local/ucfp/etc/ext/music;
name = music_ct; type = ct; value = /usr/local/ucfp/etc/ct/music;

Для запрета графических файлов:
name = pictures_ext; type = ext; value = /usr/local/ucfp/etc/ext/pictures;
name = pictures_ct; type = ct; value = /usr/local/ucfp/etc/ct/pictures;

Для запрета видео-файлов:
name = video_ext; type = ext; value = /usr/local/ucfp/etc/ext/video;
name = video_ct; type = ct; value = /usr/local/ucfp/etc/ct/video;

Файлы для внесения IP-адресов клиентов и серверов:
name = unchecked_clients; type = clients; value = /usr/local/ucfp/etc/clients/unchecked;
name = unblocked_clients; type = clients; value = /usr/local/ucfp/etc/clients/unblocked;
name = unchecked_urls; type = urls; value = /usr/local/ucfp/etc/urls/unchecked;
name = unblocked_urls; type = urls; value = /usr/local/ucfp/etc/urls/unblocked;

Скрипты оповещения:

name = mail; type = mail_notify; value = /usr/local/ucfp/notifies/mail_notify;
name = popup; type = popup_notify; value = /usr/local/ucfp/notifies/popup_notify;
name = log; type = log_notify; value = /usr/local/ucfp/notifies/log_notify;

6.6.2 Категории

По умолчанию в файле /usr/local/ucfp/etc/lists определены следующие категории:

name = white; type = list; block = db, tpl;

name = viruses; type = list; block = db, tpl, tmp;

name = banners; type = list; block = db;

name = proxies; type = list; block = db;

name = webmail; type = list; block = db, dns_parts;

name = porno; type = list; block = db, tpl, dns_parts, tmp;

и далее тоже самое для категорий с именами:

rest, humor, music, video, games, sport, animals, books, referats, warez, travel, cigar, medic, cars, dating, pictures, beauty, job, chat, phones, food, horo, amuse, ecards.

6.6.3 Правила

По умолчанию в файле /usr/local/ucfp/etc/rules определены следующие правила:

do = allow; clients = $unchecked_clients;

do = allow; clients = $unchecked_urls;

do = deny; list = $banners;

do = allow; lists = $white;

do = deny; list = $viruses;

do = allow; clients = $unblocked_clients;

do = allow; urls = $unblocked_urls;

do = deny; lists = $proxies, $rest, $dating, $video, $warez, $porno, $ecards, $phones, $animals, $humor, $music, $sport, $games, $referats, $books, $travel, $cigar, $medic, $cars, $pictures, $beauty, $job, $chat, $food, $horo, $amuse, $webmail;

Таким образом, по умолчанию администратор имеет возможность вносить в файл:

/usr/local/ucfp/etc/clients/unchecked – IP-адреса клиентов, трафик которых не должен проверятся

/usr/local/ucfp/etc/urls/unchecked – сайты, трафик с которых не должен проверяться

/usr/local/ucfp/etc/clients/unblocked – IP-адреса клиентов, трафик которых проходит только антивирусную проверку, а также режутся баннеры

/usr/local/ucfp/etc/urls/unblocked – сайты, для которых производится только антивирусная проверка, а также режутся баннеры

6.6.4 Настройки обновлений

По умолчанию в файле /usr/local/ucfp/etc/ucfpdb определены следующие настройки обновлений:

white = strip_db, tpl;

viruses = strip_db, tpl, send;

banners = strip_db;

webmail = strip_db;

proxies = strip_db;

porno = strip_db, tpl, send;

и далее тоже самое для категорий с именами:

rest, humor, music, video, games, sport, animals, books, referats, warez, travel, cigar, medic, cars, dating, pictures, beauty, job, chat, phones, food, horo, amuse, ecards.

6.7 Пример создания шаблона

В файле **user_tpl** содержатся следующие записи:

badword1 = 10

badword2 = 20

badword3 = 5

В конфигурационном файле у данной пользовательской категории параметр **threshold** = 50

Пользователь запрашивает html-страницу, которая помимо всего прочего содержит в себе указанные в user_tpl слова в таком количестве:

badword1 2

badword2 1

badword3 4

Умножая на весовые коэффициенты получаем 2*10+1*20+4*5=60, это значение больше указанного в **threshold**, значит данная страница попадает в **user_tmp,** т.е. будет отнесена к данной категории.

Слова в user_tpl задаются во всех русских кодировках. Для этого в состав дистрибутива входит скрипт maketpl, который используется следующим образом:

maketpl –i words.file –o user.tpl –l koi8-r

где words.file – файл с исходными словами, user.tpl – полученный файл-шаблон, koi8-r – исходная кодировка.

Если *maketpl* выдает ошибку:

Can't setup [ru_RU.koi8-r] locale: No such file or directory

это означает что не заданы настройки соответствующей локали. Установить их можно командами:

localedef -f koi8-r -i ru_RU ru_RU.koi8-r

localedef -f cp1251 -i ru_RU ru_RU.cp1251

localedef -f utf8 -i ru_RU ru_RU.utf8

6.8 Пример отчета

10.01.2004 12:30:00 10.0.0.2 – 200/GET 194.226.146.5 http://www.e1.ru/title.gif image/gif 512 yes allow -

10 января 2004 в 12:30 неавторизованный клиент с IP-адреса 10.0.0.2 обратился к ресурсу http://www.e1.ru и ему был передан файл title.gif размером 512 байт. Файл не был проверен ни антивирусом (например в настройках не указано проверять подобного рода файлы), ни анализатором контента, который анализирует только тектовые ресурсы. Сайт e1.ru не содержится в базах данный ни одной категории, поэтому доступ был разрешен.

10.01.2004 12:31:00 10.0.0.14 egor 200/GET 195.12.77.146 ftp://ftp.relans.ru/updates/101_45095.exe application/octet-stream 549222 yes allow KAV

Пользователь с логином egor с IP-адреса 10.0.0.14 обратился к ресурсу ftp://ftp.relans.ru и скачал файл updates/101_45095.exe размером 549222 байта. Файл был проверен, вирусов обнаружено не было. Ресурс relans.ru не запрещен ни одним из листов других категорий, поэтому доступ был разрешен.

10.01.2004 12:33:00 10.0.0.16 badguy 200/GET 195.12.77.146 ftp://ftp.relans.ru/test/eicar.com application/octet-stream 72 yes block Infected: EICAR-Test-File / (quarantined)

Пользователь с логином badguy с IP-адреса 10.0.0.16 обратился к ресурсу ftp://ftp.relans.ru и пытался скачать test/eicar.com размером 72 байта. В файле был обнаружен тестовый вирус, который был помещен на карантин. Антивирусная проверка была выполнена первой, поэтому проверки по всем другим категориям не производились.

10.01.2004 12:33:20 10.0.0.16 badguy 200/GET 195.12.77.146 ftp://ftp.relans.ru/test/eicar.com application/octet-stream 0 yes block Viruses temporary list

Пользователь с логином badguy с IP-адреса 10.0.0.16 повторно пытался скачать файл ftp://ftp.relans.ru/test/eicar.com, но, т.к. данный ресурс уже присутствует в текущем черном списке, ни единого байта передано не было.

10.01.2004 12:34:00 10.0.0.3 – 200/GET 81.176.69.78 http://www.avp.ru/index.html text/html 30451 yes allow White database

Неавторизованный клиент с IP-адреса 10.0.0.3 обратился к ресурсу http://www.avp.ru, адрес которого есть в белом списке антивирусной проверки (white_list), поэтому доступ был разрешен без каких-либо дополнительных проверок.

10.01.2004 16:22:00] 10.0.0.1 – 200/GET 194.87.11.112 http://porno.ru/index.html text/html 12524 yes block Porno database

Неавторизованный клиент с IP-адреса 10.0.0.1 обратился к ресурсу http://porno.ru/index.html. URL содержится в базе данных порносодержащих ресурсов. Доступ к ресурсу был заблокирован.

10.04.2004 19:04] 10.0.0.2 – 200/GET 194.87.11.112 http://www.porno.ru/ru/ text/html 12545 yes block Porno Template

Неавторизованный клиент с IP-адреса 10.0.0.1 обратился к ресурсу http://www.porno.ru/ru/, который в результате проверки контента по шаблонам был отнесен к порносодержащим ресурсам. Доступ к ресурсу был запрещен.

7. Примечания

7.1 Перечисление значений параметров

Перечисление значений любых параметров можно задавать, разделяя их «,».

Также существует возможность задавать значения параметров в формате пути к файлу; В этом случае значения будут браться из указанного файла, который должен содержать по одному значению в строке.

Например,

not_hold_ext = zip,rar,arj

можно задать как:

not_hold_ext = /usr/local/ucfp/etc/my_not_hold_ext

и в файле /usr/local/ucfp/etc/my_not_hold_ext прописать:

zip

rar

arj

При задании значений не поддерживаются wildcards и regexp, за исключением:

* - означает «все». Если среди перечисляемых параметров встречается данный символ, то все остальные не учитываются

. - означает «пустое значение». Например . в файле check.ext означает «производить антивирусную проверку файлов без расширений».

7.2 Типы контента

Описаны в файле */etc/mime.types*

Также можно использовать входящие в дистрибутив файлы с перечисление Content-Types для музыкальных, видео и графических файлов (каталог /usr/local/ucfp/etc/ct).

При задании значений не поддерживаются ни wildcards (*,?) ни regexp (.*,$,^).

7.3 Собственные наборы иконок для ftp

В состав дистрибутива входят следующие иконки для отображения различных типов файлов на ftp:

- mico-dir.gif
- mico-unknown.gif
- mico-up.gif
- mico-image.gif
- mico-script.gif
- mico-binhex.gif
- mico-movie.gif

- mico-sound.gif
- mico-xbm.gif
- mico-c.gif
- mico-octet-stream.gif
- mico-tar.gif
- mico-xpm.gif
- mico-compressed.gif
- mico-pdf.gif
- mico-tex.gif
- mico-dir.gif
- mico-ps.gif
- mico-text.gif
- mico-dvi.gif
- mico-rpm.gif

Есть возможность использовать свой набор иконок или, например, от squid-а из /usr/local/squid/share/icons, поместив их в /usr/local/ucfp/internal и переименовав соответствующим образом.

7.4 Переменные скрипта для извещения администратора

В скрипте для оповещения администратора допустимо использование следующих переменных:

$UCFP_QTYPE – тип события (block или license)

$UCFP_TIME – Дата и время

$UCFP_IP – IP-адрес клиента

$UCFP_USER – логин клиента

$UCFP_URL – URL

$UCFP_SIZE – размер файла

$UCFP_STATUS – статус проверки

$UCFP_RESULT – результат проверки

$UCFP_CHECKER – кто проверял

$UCFP_INFO – дополнительная информация (название вируса или категории блокировки)

$UCFP_LIST – название категории блокировки

$UCFP_VIRUS – название вируса

$UCFP_TOTAL – кол-во лицензий

$UCFP_USED – текущее кол-во лицензированных адресов

$UCFP_REMAIN – кол-во свободных лицензий

Примеры извещения администратора:

Subject: Warning! UCFP event type: block

Access to resource blocked.

Date: 29/05/2004 18:18:20

Client IP: 10.0.0.4

Client Login: vasya

Requested url (GET): ftp://ftp.relans.ru/test/eicar.com

File size: 72

Check result: Infected

Checker: KAV

Info: Infected: EICAR-Test-File / (quarantined)

Subject: Warning! UCFP event type: block

Access to resource blocked.

Date: 29/05/2004 18:23:50

Client IP: 10.0.0.1

Client Login: badguy

Requested url (GET): http://www.sex.com/s.html?cn=russia

File size: 1448

Check result: Porno

Checker: ACL

Info: Porno database

7.5 Обновление антивирусных баз

Для антивирусной проверки трафика средствами KAV рекомендуется использовать расширенный набор антивирусных баз, который содержит сигнатуры различных программ, которые, по сути, не являются вирусами, но при этом имеют очень неприятные свойства (например, прописывают адреса на порно-ресурс в стартовую страницу браузера или периодически при работе с интернетом открывают дополнительные окна с рекламой).

Для перехода на расширенный набор антивирусных баз необходимо или использовать ucfpdb –get или поле обновления антивирусных баз и перед рестартом антивирусного демона в папке с антивирусными базами выполнять команду *cp avp_x.set avp.set*

7.6 Типы баз

Существуют три типа баз: **полный, оптимизированный** и **ip-ориентированный**.

Полный набор баз включает себя все доменные имена и ip-дреса.

В ***оптимизированный набор*** не входят ресурсы, DNS-имена которых содержат слова из parts для данной категории.

В ***ip-ориентированный набор*** не входят ресурсы, DNS-имена которых содержат слова из parts для данной категории, и ресурсы блокируются преимущественно на основе ip-адресов запрашиваемых серверов.

Например в оптимизированный набор не входят записи для superporno.com, mega-sex.ru, gamer.ru, basketball.sport.ru и т.д., т.к. они блокируются соответствующими словами из parts.

Оптимизированный набор меньше полного по объему примерно на 25%. IP-ориентированный набор меньше полного по объему примерно на 50%. Мы рекомендуем использовать оптимизированный набор баз совместно с parts (block содержит url_parts или dns_parts).

7.7 Отрицательные parts

Существует возможность задавать отрицательные значения слов (начинаются со знака «-»), при обнаружении которых в доменном имени или URL запрашиваемого ресурса, блокировки не будет. Данный механизм полезен например при блокировке доменов третьего уровня chat.ru и mail.ru, а также разрешения таких ситуаций, как отнесение в категорию sport ресурсов, содержащих в доменном имени или URL слов passport или transport.

7.8 Ограничения пробной версии

В состав дистрибутива входит ключевой файл, позволяющий UCFP работать в полнофункциональном режиме 1 месяц с момента первого запуска.

8. FAQ

Данный раздел содержит ответы на несколько наиболее часто задаваемых вопросов.

Весь список доступен в разделе сайта FAQ.

8.1 Как проверить, что UCFP работает?

Для начала можно просто зайти на любой ftp, если Вы видите красивые иконки и бело-серую разлиновку (См. также «Примечания: Собственные наборы иконок для ftp»), значит запросы на UCFP поступают.

Примечания:

Если иконок нет совсем, значит неправильно задан параметр [ucfp.common]

hostname

Проверить его правильность можно, запросив с рабочей станции http://hostname:12345/err-virus.gif

Если Вы видите на ftp иконки, стандартные для Вашего прокси-сервера, значит, если Вы используете squid, Вам имеет смысл ознакомится с FAQ: Настройка squid, или, в случае использования другого прокси-сервера, каким-либо образом сказать ему, что UCFP является теперь для него parent-ом.

Если с иконками на ftp все в порядке, можно попытаться скачать **тестовый** вирус:

ftp://ftp.relans.ru/test/eicar.com

http://www.relans.ru/test/eicar.com

Если все работает, то Вы должны увидеть следующее: **Infected: EICAR-Test-File /**

Также можно помотреть статистику по

telnet localhost listen_port+1 или лог-файл ucfp.log

8.2 X-Forwarded-For

Если UCFP работает по схемам №3 и 5, т.е. после прокси-сервера, то подсчет лицензированных IP производится по хэдеру X-Forwarded-For, который обязательно должен быть разрешен (в squid-е по умолчанию разрешены все хэдеры), иначе проверка осуществляться не будет.

X-Forwarded-For - это один из хэдеров, в котором прокси передает запрашиваемому серверу IP-адрес клиента.

Признаком того, что UCFP не получает этот хэдер, является неизвестный ip-адрес клиента в ucfp.log:

[13/03/2005 21:02:46] unknown - 200/GET 213.180.204.8 http://ya.ru/ text/html 1490 no

За этот хэдер X-Forwarded-For в squid.conf отвечают два параметра. Первый это **forwarded_for**, который может принимать значения *on* (передавать X-Forwarded-For) и *off* (не передавать). По умолчанию *on*. Если у Вас этот параметр в *off*, то, соответственно, нужно сделать *on*, и выполнить *squid –k reconfigure*.

Второй параметр, имеющий отношение к данному хэдеру - **anonymize_headers** (для squid 2.4) или **header_access** (для squid 2.5). Версию squid можно посмотреть по *squid –v*

Для squid 2.4:

В конфигурационном файле squid.conf про anonymize_headers сказано:

There are two methods of using this option. You may either allow specific headers (thus denying all others), or you may deny specific headers (thus allowing all others).

Что означает: Вы можете разрешить конкретные хэдеры, тем самым запретив все остальные, или наоборот, запретить некоторые, при этом разрешив все остальные.

Если Вы решите пойти по пути разрешения определенных хэдеров и запрета остальных, то не забудьте включить:

anonymize_headers allow X-Forwarded-For

Для squid 2.5:

В конфигурационном файле есть возможность разрешать/запрещать конкретные хэдеры. Таким образом досточно в squid.conf добавить:

header_access X-Forwarded-For allow all

После изменения squid.conf не забудьте выполнить

squid –k reconfigure

8.3 Настройка squid

Для того, чтобы направить squid через UCFP необходимо в squid.conf добавить следующую запись:

cache_peer ucfp_ip_listen parent ucfp_port_listen 7 no-query default login=PASS

где,

ucfp_ip_listen - интерфейс, который слушает UCFP (рекомендуется 127.0.0.1)

ucfp_port_listen - порт, который слушает UCFP (по умолчанию 12345)

login=PASS – актуально при работе по схемам №3 и 5 и при использовании squid-ом basic authentification, т.е. для передачи параметров авторизации от squid-а к UCFP. Если авторизация не используется, то параметр необязателен.

Не забудьте выполнить

squid –k recnfigure

Есть два момента:

1. squid по умолчанию не передает на parent (т.е. в данном случае на UCFP) запросы к неанонимным ftp серверам

2. squid по умолчанию не передает на parent SSL запросы (https), что, в случае антивирусной проверки, и не нужно, т.к. SSL контент проверить нельзя.

Если такое положение дел Вас устраивает, то дальше читать не нужно.

Если же Вы хотите проверять файлы с неанонимных ftp серверов, то необходимо добавить в sduid.conf следующий параметр:

nonhierarchical_direct off

Но при этом squid начнет слать на parent и SSL запросы. В этом нет ничего страшного, т.к. UCFP умеет «прокидывать» https, но будет ненужная нагрузка на сервер. Поэтому мы все же рекомендуем настроить squid на работу по https напрямую. Для этого в squid.conf нужно добавить следующие строки:

acl SSL method CONNECT

always_direct allow SSL

8.4 Вопросы в support

Во всех случаях падения или любого другого некорректного поведения UCFP разработчикам понадобится следующая информация:

- диагностические записи о PANIC («паник-скрин»)
- название и версия операционнной системы
- вывод /usr/local/ucfp/bin/ucfp -v
- кусок лог-файла, а лучше весь (в разумных пределах)

8.5 Многопоточная закачка

Важно: При использовании многопоточной закачки с **check_ranged = no существует вероятность необнаружения вирусов**.

Суть многопоточной закачки заключается в следующем: **один** пользователь (клиент) осуществляет закачку **одного** файла в **несколько** потоков (запросов), при этом в каждом потоке скачиваются разные части файла (смещение от начала и количество запрашиваемых байт из файла определяется в хэдере Range запроса, используемом, в частности, при «докачке»).

Параметр **check_ranged** определяет необходимость удержания запросов с хэдером Range, с целью дальнейшей антивирусной проверки полученного файла.

Использование **check_ranged = yes** исключает пропуск вирусов, но приводит к увеличению трафика, т.к. программы, осуществляющие многопоточную закачку (ReGet, FlashGet и т.п.) в первом потоке не используют хэдер Range, а в остальных запрашивают часть файла с определенного смещения и до конца файла (например Range: bytes=2713652-). Таким образом при использовании 5 потоков лишний трафик составит 200% от размера запрашиваемого файла.

При **check_ranged = no** данные от запросов с хэдером Range сразу передаются клиенту, что позволяет избежать возникновения лишнего трафика, однако в этом случае при многопоточном запросе zip-архива не производится антивирусная проверка потоков, т.к. для проверки содержимого архива его нужно распаковать. При многопоточном запросе exe-файла есть довольно большая вероятность (но не 100%) того, что KAV обнаружит вирус в одной из частей, и эту часть пользователь не получит, но при этом все остальные части будут успешно скачены.

8.6 Скорость работы

Информация по таймингу отображается на первом уровне лога (loglevel = 1).

Скорость проверки в базах данных зависит от использования свопинга (параметр [ucfp.common] db_swap)

На тестовом сервере (Celeron633/256RAM) проверка страницы объемом 1,1Мб (документация по sendmail) с включенным свопингом (db_swap=yes) дает следующие результат:

```
[28/06/2006 14:22:01] 10.0.0.1 - 200/GET 195.12.77.146
http://ucfp.ru/test.htm text/html 1120436 yes allow CF/KAV
[28/06/2006 14:22:01] DEBUG Query (http://ucfp.ru/test.htm) timing stats:
all: 1244 (msec), acl: 0.002611 (sec), CF: 0.382350 (sec), KAV: 0.069595
(sec)
```

Таким образом потрачено времени:

- на проверку наличия адреса сервера в базах данных - 0.002611 сек
- на анализ контента (на странице более 23600 значимых слов) - 0.382350 сек
- на проверку файла антивирусом - 0.069595 сек

Тот же самый запрос с выключенным свопингом (db_swap=no):

```
[28/06/2006 15:15:34] 10.0.0.1 - 200/GET 195.12.77.146
http://ucfp.ru/test.htm text/html 1120436 yes allow CF/KAV
[28/06/2006 15:15:34] DEBUG Query (http://ucfp.ru/test.htm) timing stats:
all: 697 (msec), acl: 0.001364 (sec), CF: 0.384969 (sec), KAV: 0.063235 (sec)
```

Таким образом отсутствие свопинга в 2 раза повышает скорость поиска в базах данных, но требует дополнительного объема свободной оперативной памяти (см. раздел «FAQ: Минимальные требования»).

Для примера приведем временные параметры анализа средней по объему html-страницы, например yandex.ru (свопинг включен):

```
[28/06/2006 15:27:13] 10.0.0.1 - 200/GET 213.180.204.11 http://yandex.ru/
text/html 24755 yes allow CF/KAV
[28/06/2006 15:27:13] DEBUG Query (http://yandex.ru/) timing stats: all: 727
(msec), acl: 0.002375 (sec), CF: 0.004563 (sec), KAV: 0.044347 (sec)
```

8.7 Минимальные требования

CPU:

Пиковый прирост использования CPU:

- Linux – 20%
- FreeBSD/OpenBSD – 40%

RAM:

Необходимый объем свободной оперативной памяти зависит от использования свопинга баз данных (параметр [ucfp.common] db_swap):

- db_swap = yes (свопинг включен) - объем зависит только от количества используемых категорий и не зависит от размера баз данных. Для каждой категории требуется примерно 0.7Mb. При использовании всех 31 категории необходимо 22M свободной оперативной памяти.
- db_swap = no (свопинг выключен) - объем зависит от размера баз данных и на данный момент составляет 72-112Mb в зависимости от типа баз данных.

Для антивирусной проверки необходимо дополнительно 10Mb свободной оперативной памяти.

HDD:

Антивирус Касперского – 8Mb

UCFP (без баз данных) – 7Mb

Базы данных UCFP (на данный момент) – 28-60Mb в зависимости от типа баз данных

При использовании свопинга баз данных необходимо дополнительно 20-40Mb свободного дискового пространства.