

Ворота в Сеть

Где бы мы ни находились, всегда хочется быть в курсе последних событий и знать прогноз погоды на следующий день. Владельцы смартфонов решают этот вопрос просто: в их телефонах есть браузер, который может просматривать любые веб-страницы с помощью GPRS. Обладателям телефонов более простых моделей приходится искать другие пути.

Первое, что приходит в голову, — воспользоваться WAP, благо его поддержка есть почти в каждом телефоне. Однако после недели использования от этого решения хочется отказаться: тарификация поминутная, скорость соединения не превышает 9600 бит/с, а цены и вовсе кусаются. Зачастую минута связи стоит столько же, сколько минута общения с абонентами городской сети. Тут уж не до вдумчивого чтения!

Вместо этого некоторые операторы советуют пользоваться услугой GPRS+WAP. Предложение разумное, но при анализе расценок выясняется, что провайдеры постарались сделать их такими, что выгода для абонента выглядит очень сомнительной. В результате он вынужден платить гораздо больше, чем если бы пользовался смартфоном и обычным браузером, оплачивая трафик GPRS. Чтобы понять, почему это происходит, и попытаться устранить несправедливость, надо разобраться в основах работы GPRS+WAP.

| Как работает WAP? |

WAP (Wireless Application Protocol) — протокол передачи данных по беспроводным сетям с очень ограниченной пропускной способностью. Однако в отличие от HTTP или FTP он не доставляет содержимое сайтов напрямую с сервера на телефон, а требует промежуточное звено — WAP-шлюз (в английской терминологии — WAP-Gate). Мини-браузер на телефоне использует шлюз в качестве посредника при обращении к сайтам. При этом шлюз обменивается данными с телефоном по протоколу WAP, а с сайтом — по HTTP. Это обусловлено техническими причинами: HTTP был изначально создан для компьютеров с мощными про-

цессорами и широкими каналами связи, но всего этого у телефонов нет. По той же причине для телефонов не подходит HTML. Его гибкость и сложность World Wide Web Consortium (www.w3c.org) принес в жертву строгости и простоте, разработав новый язык разметки — WML (Wireless Markup Language).

Именно таким шлюзом нам дает пользоваться оператор, когда требует плату за услугу GPRS+WAP. Обычно в руководствах по настройке телефона его обозначают предельно просто и ясно — «IP-адрес». Давайте настроим шлюз, обеспечив практически бесплатный доступ к WAP-ресурсам! Ведь даже при активном ежедневном просмотре новостей, чтении анекдотов и прогнозов погоды тратится не более 1 Мбайт трафика в месяц.

| Установка и настройка шлюза |

Нам понадобится специально предназначенная для этого программа kannel (www.kannel.org). Последняя версия на момент написания статьи — 1.4.0. Далее приведена последовательность действий для ASPLinux 10, но их можно с легкостью распространить и на любой другой дистрибутив. Создадим нового пользователя kannel. В дальнейшем будем работать от его имени. Скачаем и распакуем исходные тексты:

```
$ mkdir /usr/arc/kannel && cd /usr/arc/kannel
$ wget -c http://kannel.org/download/1.4.0/gateway-1.4.0.tar.bz2
$ tar -jxvf gateway-1.4.0.tar.bz2
$ cd gateway-1.4.0
```

Запустим скрипт ./configure со следующими параметрами:
► --disable-docs — отключить файлы документации. Опыт показывает, что при сборке документации установится большое

количество дополнительных пакетов, значительно превышающих по объему уже готовые файлы, которые можно получить на сайте производителя.

► `--enable-localtime` — отмечать в логах локальное время, а не по Гринвичу.

► `--disable-sms` — отключить возможность отправки SMS. Для работы этой опции потребуются доступ к SMS-центру оператора, которого у нас в данный момент нет.

Теперь соберем и установим исполняемые файлы в каталог `/usr/arc/kannel/bin/`:

```
$ make
```

```
$ sudo make bindir = /usr/arc/kannel/bin install
```

Синтаксис конфигурационного файла очень прост и чем-то напоминает файл конфигурации Samba. Весь файл разбит на три части, которые относятся к разным компонентам kannel: `bearerbox` (основной компонент шлюза), `wapbox` (компонент, отвечающий за поддержку WAP) и `smsbox` (модуль поддержки транспорта SMS). Его мы использовать не будем, поэтому соответствующую секцию оставим пустой. Каждая часть начинается со слова `group`. Команда `a = b` присваивает переменной `a` значение `b`, а если `b` строкового типа, значение следует заключить в кавычки. Строки-комментарии начинаются со знака «#». Создадим пустой файл `/usr/arc/kannel/kannel.conf` следующего содержания:

```
# Основная группа
```

```
group = core
```

```
# порт и пароль для управления kannel
```

```
admin-port = 13000
```

```
admin-password = password1
```

```
# порт, к которому будут подключаться wapbox
```

```
wapbox-port = 13002
```

```
# пароль для просмотра статуса шлюза
```

```
status-password = password2
```

```
# список разрешенных хостов для доступа к шлюзу
```

```
# адреса отделяются друг от друга точкой с запятой, «*» означает любой адрес
```

```
box-allow-ip = "*"

```

```
udp-allow-ip = "*"

```

```
# имя файла для записи логов

```

```
log-file = "/tmp/kannel.log"

```

```
# числами от 0 до 5 будут регистрироваться все события,

```

```
# уровень которых не ниже установленного

```

```
# рекомендуется поставить 0 на этапе настройки,

```

```
4 — при использовании

```

```
log-level = 0

```

```
# имя файла для регистрации сообщений о доступе к серверу

```

```
access-log = "/tmp/kannel.access.log"

```

```
# интерфейс, который будет «слушать» kannel

```

```
# укажите здесь внешний IP-адрес сервера, на который будут
приходить запросы

```

```
# формат такой же, как у параметра "box-allow-ip"

```

```
wdp-interface-name = "111.111.111.111"

```

```
# группа wapbox

```

```
group = wapbox

```

```
bearerbox-host = localhost

```

```
log-file = "/tmp/kannel.wap.log"

```

Здесь надо сделать несколько довольно важных замечаний. Kannel, как было сказано, состоит в нашем случае из двух модулей — `bearerbox` и `wapbox`. К одному имеющемуся `wapbox` может быть подключено несколько `bearerbox`. Например, для того чтобы распределить нагрузку на серверную машину, можно разместить `bearerbox` на одном сервере, а `wapbox` — на трех других. Директива `bearerbox-host` в группе `wapbox` указывает, на каком конкретном сервере работает `wapbox` и к какому из них следует подключаться.

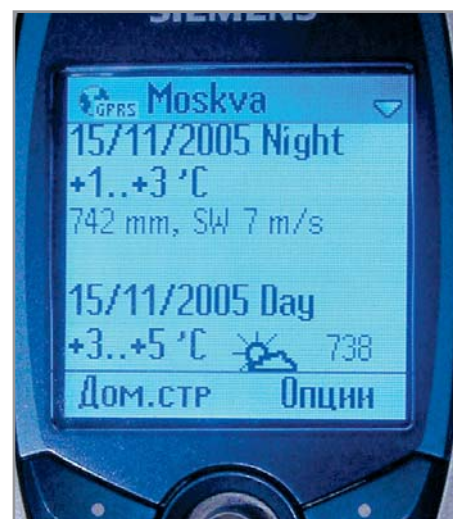
Настало время проверить, все ли мы правильно сделали. Запустим `bearerbox` и `wapbox` с максимальным уровнем отладки в разных терминальных окнах:

```
$ /usr/local/sbin/bearerbox -v 0 /usr/arc/kannel/gateway-1.4.0/kannel.conf

```

```
$ /usr/local/sbin/wapbox -v 0 /usr/arc/kannel/gateway-1.4.0/kannel.conf

```



Не правда ли удобно всегда иметь под рукой последние новости и сводки погоды?

Внимательно просмотрите вывод этих команд. Должны появиться сообщения только типов INFO и DEBUG. Если вы видите сообщения ERROR или тем более PANIC, то постарайтесь найти причину. Это могло произойти, например, если вы неправильно указали адрес интерфейса в переменной wdp-interface-name.

Для проверки работоспособности всей связки в целом используется эмулятор WAP-клиента fakewap, входящий в состав kannel. В третьем терминальном окне перейдите в каталог с исходными текстами kannel и запустите следующую команду (вместо значения 111.111.111.111 впишите адрес сервера, на котором работает warbox):

```
$ test/fakewap -g 111.111.111.111 wap.rbc.ru
```

Если вы видите надпись «fakewap complete» — шлюз работает!

Создадим простой скрипт для запуска шлюза следующего содержания. Если вы хотите, чтобы шлюз запускался при старте системы, добавьте путь к скрипту в конец файла /etc/rc.local:

```
#!/bin/bash
sudo -u kannel /usr/local/sbin/bearerbox -v 5 -d -P
/usr/arc/kannel/gateway-1.4.0/kannel.conf
sleep 3
sudo -u kannel /usr/local/sbin/wapbox -v 5 -d -P /usr/arc/kannel/gateway-1.4.0/kannel.conf
```

Как вы, наверное, заметили, добавилось два параметра. Параметр -d запускает сервисы в виде демонов. Второй, дополнительный, параметр -P обеспечивает автоматический перезапуск

сервиса в случае его краха. К сожалению, это нужный аргумент. Бывает, warbox «падает» даже от сканирования портов с помощью nmap. Трехсекундная пауза между запуском bearerbox и warbox необходима, чтобы bearerbox успел проинициализироваться, перед тем как warbox начнет к нему подключаться.

Настройка телефона

Для работы со свеженастроенным шлюзом надо подходящим образом подготовить телефон. Подробное руководство вы можете найти на сайте своего оператора и выставить все настройки согласно его рекомендациям. Изменить нужно лишь один параметр — в одних аппаратах он называется «IP-адрес», в других — «шлюз». Сделайте его равным IP-адресу сервера, на котором установлена kannel. Теперь выберите GPRS как предпочтительный способ соединения. Телефон готов к работе!

Последний нюанс

Итак, теперь у вас есть полнофункциональная замена WAP-шлюзу оператора. Она не требует денег, ее работа зависит только от вас, что, согласитесь, немаловажно. Особо следует отметить, что использование этого шлюза абсолютно законно и не нарушает условий контракта. Однако в качестве меры предосторожности не стоит публиковать его IP-адрес, потому что заблокировать доступ к нему со стороны оператора очень просто (но на функциональности сервера это, конечно, не скажется).

Дополнительные услуги

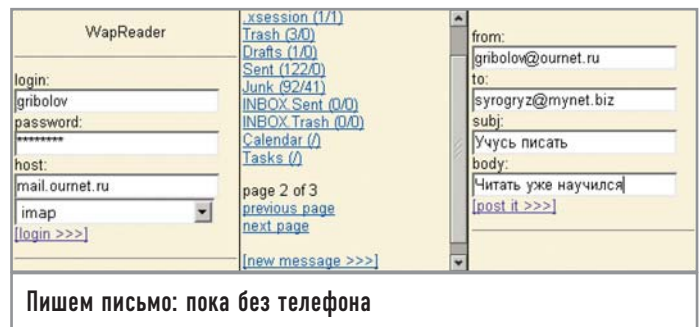
Читаем почту

Пользователи сервиса Mail.ru вскоре обнаружат возможность чтения и отправки электронной почты через WAP-интерфейс. Но что делать, если администратор сервера, на котором располагается ваш почтовый ящик, не озаботился настройкой подобной службы? Сделайте это самим! Для этого понадобится почтовый клиент с WAP-интерфейсом warreader. Его можно скачать с сайта <http://warreader.dp.ua/ru/about>. Он может работать по протоколам IMAP и POP3, поддерживает защищенные соединения SSL и позволяет не только читать, но и отправлять новые письма. Установите и настройте веб-сервер, например Apache. Вполне подойдут начальные настройки. Пусть корневой папкой для него станет /var/www/html. Создайте

в ней папку mail и распакуйте туда файлы из папки warreader следующего архива: <http://citkit.dl.sourceforge.net/sourceforge/warreader/warreader-1.1.tar.bz2>. Установите модуль php-imap для поддержки протокола IMAP в PHP:

```
# yum install apache php-imap
```

Опять же будем считать адресом сервера, например, 111.111.111.111. Введите следующий адрес в строке вашего браузера (на компьютере): <http://111.111.111.111/mail/warreader.php?mode=html>. На экране должно появиться приглашение ввести имя почтового сервера, учетной записи и пароля. Именно это вы увидите на своем телефоне, когда warreader будет полностью готов к работе. Если приглашение не появилось, попробуйте открыть



Пишем письмо: пока без телефона

страничку <http://111.111.111.111/mail/test.php>. Там вы обнаружите более подробное описание найденных ошибок. После того как вы убедились, что все подобающим образом работает, можно приступить к чтению почты с помощью телефона. Для этого зайдите на страницу <http://111.111.111.111/mail/warreader.php>. Если не хотите вводить имя пользователя, пароль и другие данные при каждом

заходе на сервер, добавьте в закладки телефона ссылку следующего вида: <http://111.111.111.111/mail/warreader.php?<login>=<password>:<host>:<protocol>>, где <login> — имя учетной записи; <password> — пароль; <host> — доменное имя почтового сервера, <protocol> — IMAP или POP3. Она может выглядеть, например, так: <http://111.111.111.111/mail/warreader.php?gribolov:mukhomor@mail.ournet.ru:imap>.