

# Download

\$Revision: 1.2 \$

by Nicolas Bélan

## 1. Quick informations

We are working on 0.2.0 release of *open-firewall-core* module and *open-firewall-plugin* module.  
We may release it soon.

You can get release 0.1.1 [here](#).

The gpg [signature](#) of 0.1.1 release is :

```
-----BEGIN PGP SIGNATURE-----  
Version: GnuPG v1.2.4 (FreeBSD)  
  
iD8DBQBAVvDo7Z+vPzyCxIcRAiHEAJ0U4Ku5MJnfvbv0BCAnWY6YqxQWkQCcDIh1  
8RrHjzlwlsS687oajT8aQliY=  
=zSHM  
-----END PGP SIGNATURE-----
```

## 2. CVS access

You are just able to browse our CVS repository using viewcvs :

[open-firewall-core module](#)

To checkout entire module, just hit:

```
you@yourhost $ CVSROOT=:pserver:anonymous@cvs.sf.net:/cvsroot/open-firewall  
you@yourhost $ export CVSROOT  
you@yourhost $ cvs login <hit enter when password is asked>  
you@yourhost $ cvs -z3 co open-firewall-core
```

Detailed informations about CVS can be found [here](#).

Available modules are:

### **open-firewall-core**

Main module

### **docs**

Various documentation, including this web site

Available releases are:

### **HEAD**

Main development branch

### **OF\_RELEASE\_0\_1\_0**

tag of 0.1.1 release

### **OF\_RELEASE\_0\_1\_0**

tag of 0.1.0 release

To retrieve a particular release, do:

```
you@yourhost $ cvs -z3 co -rTAG open-firewall-core
```

where TAG is one of 'HEAD', 'OF\_RELEASE\_0\_1\_0' and so on ...

### 3. The Open Firewall Archive OpenPGP signature

Files placed on the Open Firewall website are OpenPGP signed.

This signature can be used to prove that a file, which may have been obtained from a mirror site or other location, really originated from the Open Firewall website.

Before you can do this, you must gpg --import the key below. This my key. This key is also available from most common PGP key servers, such as <http://wwwkeys.pgp.net:11371/pks/lookup?op=get&search=0x3C82C487>

To import it from the keyserver using GnuPG, do:

```
$ gpg --keyserver wwwkeys.pgp.net --recv-keys 0x3C82C487
```

Using GnuPG, verifying a signature look like this:

```
$ gpg --verify archive-version.tar.gz.asc archive-version.tar.gz
...
```

Unless you have taken explicit steps to build a trust path to the Open Firewall Archives Verification Key, you should expect to see a warning message after gpg has verified the signature. You should not be alarmed by this warning:

```
Could not find a valid trust path to the key.
Let's see whether we can assign some missing owner trust values.

No path leading to one of our keys found.

gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
```

## 4. Building Open-Firewall

### 4.1. Requirements

To build Open-Firewall core product, you need :

- Apache runtime library (APR), version 1.0.
- Apache runtime utility library (APU), version 1.0.

Referer to APR Build on Unix.

#### FIXME (NB):

But, Apache team has not released 1.0 yet.  
So, you may use the latest snapshot of the libraries, which can be found here:  
APR snapshots and  
APR-UTIL snapshots

### 4.2. Building

You just have to do:

```
root@yourhost ~/open-firewall-core $ ./configure
root@yourhost ~/open-firewall-core $ gmake
```

```
root@yourhost ~/open-firewall-core $ gmake test
root@yourhost ~/open-firewall-core $ gmake install
```

**Note:**

On non-GNU systems, GNU make is commonly installed as gmake. On GNU Systems, it is make

There are some important options to *configure*:

**--enable-debug**

build using debugging symbols, and change path of shared library

**--with-apr=DIR**

location of apr-config (may be /usr/bin or /usr/local/apache2)

**--with-apu=DIR**

location of apu-config (may be /usr/bin or /usr/local/apache2)

**--with-efence=[dir]**

link against ElectricFence

**--with-libtool**

use or not libtool

**--with-db-path=DIR**

the path used to link apu (eg: "/usr/local/lib/BerkeleyDB.4.2/lib")

**--enable-deep-dependencies**

enable or not the check of libs (usefull for dev, not for installers)

Currently, there are some known bugs with that procedure.

- You have to set --enable-debug unless linker will complain with unexistent OF libraries
- If you do not install latests apr snapshots, libtool let \*.so into apr[-util]/.libs.  
You have to copy or link these libraries into apr source base directory (/path/to/snapshots/apr-latest/) :

```
me@host ~ $ cd /path/to/snapshots/apr-latest
me@host /path/to/snapshots/apr-latest $ ln -sf .libs/libapr-1.so.0
me@host /path/to/snapshots/apr-latest $ ln -sf .libs/libapr-1.0.so
```

- You have to do the same in apr-util library

### 4.2.1. FreeBSD compilation sample

Options to make it compile on a FreeBSD 5.2.x

**Note:**

We are using uninstalled snapshots

```
me@host ~/OF $ mkdir OF_compilation
me@host ~/OF $ cd OF_compilation
me@host ~/OF/OF_compilation $ sh ../open-firewall-core/configure
--with-apr=/path/to/snapshots/apr-latest \
--with-apu=/path/to/snapshots/apr-util-latest \
--with-db-path=/usr/local/lib/BerkeleyDB.4.2/lib \
--enable-debug
me@host ~/OF/OF_compilation $ gmake all samples tests
```

### 4.2.2. Linux debian compilation sample

Options to make it compile on a Debian

**Note:**

We are using uninstalled snapshots

```
me@host ~/OF $ mkdir OF_compilation
me@host ~/OF $ cd OF_compilation
me@host ~/OF/OF_compilation $ sh ../open-firewall-core/configure
--with-apr=/path/to/snapshots/apr-latest \
--with-apu=/path/to/snapshots/apr-util-latest \
--enable-debug
me@host ~/OF/OF_compilation $ make all samples tests
```

© 2003-2004 Open Firewall