



WaterRoof

version 3.7

Quick Reference



hanyanet.com

Index

1. [About WaterRoof](#)
2. [Ipfw and alf](#)
3. [WaterRoof interface](#)
4. [Static rules editing](#)
5. [Import and export rules](#)
6. [Load rules at system boot](#)
7. [Ready Rule Sets and Wizard](#)
8. [Bandwidth Management and dynamic rules](#)
9. [Manage network connections](#)
10. [Interfaces and DNS](#)
11. [Firewall logs](#)
12. [Logs statistics](#)
13. [Network Address Translation \(NAT\) and forwarding](#)
14. [Deployment](#)
15. [Tips](#)

WaterRoof

WaterRoof is not a firewall.

WaterRoof is just a **firewall frontend**.

Mac OS X features a built-in network firewall called *ipfw*. To configure *ipfw* in Mac OS X you normally need to type shell commands manually in the Terminal. WaterRoof graphic user interface helps configuring the *ipfw* firewall.

The main purpose of WaterRoof is to speed up firewall testing and maintenance. You will be able to make libraries of firewall configuration and to deploy them to other macs. You will read and parse firewall logs and make graphic statistics.

In order to use WaterRoof you need to know the basics of networking. This Quick Reference is intended only for system administrators and professional users.

If you want to learn how *ipfw* works, then WaterRoof is not the place to start. Anyway, it will be a good companion for your test sessions.

Mac OS X firewalls: *ipfw*, *alf*, *pf*

Mac OS X **10.4** Tiger has a built-in network firewall: *ipfw*. From the Mac OS X System Preferences you can set some basic *ipfw* option.

Mac OS X **10.5 and 10.6** have TWO built-in firewalls: the *ipfw* network firewall and the *alf* application level firewall. From the Mac OS X System Preferences you can set some basic *alf* option.

OS X **10.7** Lion has THREE firewalls: *ipfw*, *alf* and *pf*. *Pf* is a network firewall (OpenBSD). ***Ipfw* is deprecated but fully functional**. Apple.com lists IPFW as a OSX Lion Server feature, and firewall GUI is about *ipfw* not *pf*. Otherwise, for example, enabling Internet Sharing in OSX 10.7 enables *pf* rules, not *ipfw*.

Ipfw and *pf* work at network level, allowing or denying system-wide network connections.

Alf works at application level, allowing or denying network connection from/to specific applications.

lpfw and *alf* work together, they do different things, they do not interfere each other. Almost the same for *pf*.

WaterRoof is a frontend for IPFW only.

If you need a **frontend for PF** then try **IceFloor**, a new free and open source project from hanynet.com. IceFloor is a graphic frontend for the *pf* firewall.

More at:

<http://www.hanynet.com/icefloor>

WaterRoof does not and will not support *pf* in future releases. *lpfw* is deprecated in OS X 10.7 and 10.8.

Please note: *pf* comes with traffic shaping (ALTQ) but is disabled by default in OS X 10.7.

WaterRoof interface

When starting WaterRoof you will be prompted for your password. You need to be an **administrator** in order to use WaterRoof. WaterRoof may ask you to **re-authenticate every five minutes** or so. This is a security feature of Mac OS X cocoa-applescript framework.

The main window shows current ipfw rules. From this window you can work with rules or access other WaterRoof tools. For example you can read and analyze logs or check and manage active connections.

Some WaterRoof feature can be accessed only from the **menu bar**.

A lot of WaterRoof windows feature contextual helps.

Click the “?” buttons to open contextual helps. These helps are very useful in some cases, for example to properly configure NAT.

Static rules: editing

Open the “Static Rules” window from the “Firewall” menu.

You can add, move, edit and delete ipv4 and ipv6 rules.

To “add” a rule means actually to activate it and to show it in the rules table. Every rule has 4 values: number, argument, traffic (packets and bytes).

You have 3 ways to add a new rule: the advanced mode (default) or the simplified mode. The latter is a very basic interface so it lacks a lot of *ipfw* options. The advanced mode is instead quite complex and need a bit of *ipfw* knowledge. If you don't find the options you need, so add a new rule manually clicking+

Click the “ipv4/ipv6” button to switch from ipv4 rules to ipv6 rules. When switching from an editing mode to the other, you will notice that the window title changes.

Flushing rules clicking the “Flush” button in “Static Rules” window only flushes the actual editing mode rules. So if you are in ipv4 mode and you click “Flush”, you will not flush ipv6 rules. If you want to flush all ipv4 and ipv6 rules together, you can do it from the “Firewall” menu.

Static rules: editing on Mac OS X Server

adding a rule in **Mac OS X Server** activates IPFW rules and saves it to `/etc/ipfilter/ipfw.conf`. This file is the default place for custom user IPFW rules. This is mandatory on Mac OS X Server since activating an IPFW rule is not enough because the system flushes IPFW rules every 15 minutes and reloads IPFW configuration from local files.

Static rules: import and export

Import and export current ipv4 and ipv6 *ipfw* rules from the “File” menu. You use the “import/export” feature when you want to make or recover backups of your *ipfw* configuration.

You may use the “import” feature the same way as you use system-wide “locations” when changing from a network environment to another. Configuration files include ipv4, ipv6 and dummynet (bandwidth) rules. They do not include NAT settings (natd), port redirection or forwarding options.

When exporting you will be prompted for the ipv4 configuration file path. If you have also ipv6 active rules, WaterRoof will export another file, to the same location, with the same name and the “_v6” suffix appended. This file contains the ipv6 rules.

If you do not have any ipv6 rule (you have only the system ipv6 rule number 65535) WaterRoof will export only the ipv4 configuration file.

When importing rules you will be prompted for the ipv4 configuration file path. WaterRoof will silently search for a second file, with the same name and the “_v6” suffix appended, in order to load ipv6 rules also. If it does not find it, it loads only the ipv4 file. You can also import a configuration from an Injector using the “File” menu. More about Injectors later.

Static rules: load rules at system boot

To load you current rules at boot you need to:

- 1) install a default startup configuration
- 2) install scripts to load the default startup configuration at system boot.

You can do both from the “System” menu.

The default startup configuration is stored in `/etc/firewallrules` for ipv4 rules and `/etc/firewallrules_v6` for ipv6 rules.

The startup scripts are stored in `/etc/waterroof.sh` and `/Library/LaunchDaemons/net.waterroof.rules.plist` .

You can delete the startup script from the “System” menu.

SHORTCUT:

Use the ∞ button to save current rules and install a startup script

Use the 🍏 button to reset the system to factory default

Ready Rule Sets and the Wizard...

Starting from scratch? WaterRoof features Ready Rule Sets, which can help you understanding how ipfw works.

Access it from “Ready Rule Sets” -> Configuration Tools -> Ready Rule Sets.

Every ruleset has its own options and purpose.

The Wizard is another way to approach ipfw. It's a very basic way to start your mac's firewall configuration. Beware: the aim of the Wizard is not to secure your mac... you must use it only for testing purposes.

If you don't make a deep use of ipfw, you will forget its syntax very soon. Ready Rule Sets and the Wizard are a good way to easily remember how to issue common rules.

Scan and block local services

Port scanner scans localhost for open ports and lists active (listening) network services. Select a service and click “Block service” to deny access from remote hosts to the local service. Please remember to always check ipfw rules order. If you want to change rules order use “move” buttons or doubleclick a rule and change its number in the edit panel.

Bandwidth management and dynamic rules (dummynet)

WaterRoof allows you to configure dummynet pipes and queues to manage inbound and outbound network bandwidth.

You can open the “Bandwidth settings” window from the “Firewall” menu.

You can **configure up to 4 pipes**, numbered from 1 to 4.

Every time you click “OK” pipes will be flushed and reconfigured. Every pipe comes with a predefined rule number. Pipes issued from connection inspector will be numbered accordingly to their respective rule number.

To completely disable traffic shaping and resetting dummynet open dummynet window in WaterRoof, uncheck the “enabled” checkbox in all pipes and click OK. All pipes will be flushed and rules 111,222,333,444 will be deleted.

Check the WaterRoof rules window and remove all “pipe” rules, if any.

You can also “reset system”; this will flush all rules and pipes, delete all configuration files and disable startup script (rules do not load at boot any more). IPFW will be deactivated and set as factory default.

You can also put connections-specific dummynet pipes using WaterRoof Connections Inspector.

Manage network connections

View and manage active connections clicking “Connections Inspector” in main window or from the “Firewall” menu. View and manage listening connections clicking “Listening daemons”. Double click a connection to open the Connection Inspector Panel.

Connection Inspector Panel displays information about selected connection and allows you to block or limit connection bandwidth.

Click “**Block connection**” to issue 2 ipfw rules. The block is specific to both local and foreign hosts and ports.

Click “**Block service**” to issue 1 ipfw rule. The block is specific to the service handshake port, assuming the lowest port as the service port.

Click “**Block IP address**” to block all tcp/udp/icmp/igmp traffic from/to this foreign host.

Click “**Limit bandwidth**” to limit bandwidth for this connection. Specify bandwidth in Kbits per second.

The “Connected hosts” window lists all IP addresses and hostnames connected to your mac. Click “**Block IP address**” to block all tcp/udp/icmp/igmp traffic from/to this IP. Double click an IP or hostname to bring up WHOIS information.

Interfaces and DNS/Whois

Access “Network Interfaces list” from the “Network” menu. This window shows you interfaces name, ipv4, ipv6 and MAC address. Click column headers to change item order. The corresponding shell command is “ifconfig”.

You will often need to reverse IP addresses, specially when looking at active connections, processes or logs and statics. In all those tables you can double-click to select and press ⌘C to copy the IP address. Then you can easily do a DNS reverse lookup or a Whois query using the “DNS and Whois queries” tool.

Open “WHOIS & DNS” from the “Network” menu.

Firewall Logs

Open the ipfw logs window clicking “Show ipfw logs” from the “Logs” menu. You will be presented with the ipfw and alf firewall logs mixed together. This is the default for Mac OS X 10.5 and 10.6. You can choose to see only ipfw or only alf logs. You can search logs and save output to file. Click “Real Time Logs” to open a Terminal window with “live” logs. You will see this window populating with logs as long as ipfw rules are matched.

To obtain logs from the Mac OS X firewall you need to:

- 1) activate firewall logging (optionally activate it at system boot)
- 2) activate ipfw rules with the “log” option
- 3) make network connections that match those “log” rules

To activate firewall logging select “Firewall Logging” from the “Network” menu or from the “Logs” menu. Select “Enable firewall logging”. If you have installed a startup script, you have the option to update it in order to start firewall logging at system boot.

Logs are stored in:

/var/log/appfirewall.log for Mac OS X 10.5 and 10.6/10.7

/var/log/ipfw.log for Mac OS X 10.4 and Mac OS X Server.

Logs statistics

Access logs statistics features from the “Logs” menu.

Raw statistics are simple statistics generated by a bash parser and presented in a cocoa-style table.

Graphic statistics are generated by *analog*, a free logs parsing tool and *fwanalog*, a script for *analog*.

You can change preferences and decide what to include in the graphic statistics report. This is very useful because it may speed up statistics processing.

Graphic statistics are presented in a safari window. You also have the option to export the entire directory to your desktop. You will see included in this directory all images and html file. You can publish this statistics report with your corporate or personal web server in seconds, just copying the directory to your web server folder.

Please note: if you do not install WaterRoof in /Applications, graphic statistics may not work. Avoid installing it in a path with a space in its own or in parents name.

NAT (Network Address Translation) and forwarding

With WaterRoof you can manage the **NAT service**. This is done setting up natd options and port redirections, activating port forwarding and installing/updating the startup script in order to load firewall rules and NAT settings at system boot.

Open the "Configure NAT service" from the "Network" -> "Network Address Translation (NAT)" menu. Here you can configure **natd options and add port redirections**.

Click "*Default Configuration*" to activate default options for the natd daemon. Specify your NAT interface in the "*NAT Interface (WAN)*" text field. You can list your active network interfaces clicking "*Network Interfaces List*" button. The natting interface is the WAN interface, in other words the interface connected to the Internet. Click "*Save configuration*" to save your NAT setup. Configuration file is **/etc/nat.conf** . (/etc/nat/nat.conf on Server) Once done click "Start NAT" to start the NAT service and click "Enable NAT Autostart" to update your startup script in order to start the natd daemon at system boot. Enable forwarding from the "Network" menu if you need to route traffic through different network interfaces. You can configure a **full-featured dual-homed router/nat/firewall with port redirections**. Or you can share your iPhone tethering connection to all your macs at home.

NAT on Mac OS X Server 10.7 Lion

From version 10.7, NAT on Mac OS X Server is provided by PF/natmp instead of natd. NAT service in Server Admin is actually a frontend for the PF/natmp subsystem.

Activating NAT with default settings has some limit on Mac OS X Server 10.7 (10.7.3). You are forced to use 192.168.2.1 as IP address on local interface, your LAN clients are forced to use 192.168.2.0/24 network. Port forwarding is not reliable.

The solution is inside Mac OS X Server 10.7 itself. **You can use the “old” natd daemon instead of the new PF/natmp**, which is installed by default. You just need to create a configuration file and a startup script for natd. WaterRoof can do it for you in a few clicks. Just open the NAT window, click the “?” button and follow the how-to.

WaterRoof will save your **NAT settings with port forwarding rules**, and will load NATd at boot. You will be able to **choose whatever IP class for your local network**.

IPFW rules does not need to be loaded by a WaterRoof script at boot because Mac OS X Server loads IPFW rules at boot when Firewall service is enabled on Server Admin. But to load NAT you must install the WaterRoof startup script BEFORE configuring NAT.

Deployment

Firewall and nat configurations can be easily deployed through **WaterRoof Injectors**. The workflow is quite easy:

- 1) configure the firewall with WaterRoof
- 2) open the Deployment window and choose a name for the configuration
- 3) choose what to include in this deployment session (ipv4, ipv6, bandwidth...)
- 4) create and store the Injector.

The Injector can be used to deploy the configuration in one click. Injectors are Mac OS X applications which installs files and activates ipfw rules. Injector configuration includes:

ipv4 rules

ipv6 rules

bandwidth settings

natd options and port redirections


firewall logging option

startup script to load rules at boot.

To deploy a configuration just copy the Injector to the target machine, launch it and click “Inject”. Everything will be installed and activated. Click “Clean” to remove everything.

Example IPFW configuration using states

```
# allow loopback interface, mandatory
01000 allow ip from any to any via lo0
# block garbage
01010 deny log logamount 1000 ip from any to 127.0.0.0/8
01020 deny log logamount 1000 ip from 224.0.0.0/4 to any in
01030 deny log logamount 1000 tcp from any to 224.0.0.0/4 in
# allow BitTorrent using port 56789
11000 allow ip from any to me dst-port 56789
# allow connections started by local clients
12300 allow tcp from any to any established
12301 allow tcp from any to any out
12302 allow udp from any to any out keep-state
# allow DNS
12303 allow tcp from any to any dst-port 53 out keep-state
12303 allow udp from any to any dst-port 53 out keep-state
# allow UDP frags and safe ICMP
12304 allow udp from any to any in frag
12307 allow icmp from any to any icmptypes 0,3,4,8,11,12
# allow IGMP
12308 allow igmp from any to any
# allow connection to local L2TP VPN Server
12309 allow udp from any to any dst-port 1701
# allow traffic with LAN (assuming LAN network is 192.168.2.0/24)
12310 allow ip from 192.168.2.0/24 to any via en0 keep-state
# allow DHCP lease
12311 allow udp from any 68 to any dst-port 67
# allow local FTP clients
12400 allow tcp from any to not me dst-port 20-21 setup
12410 allow tcp from any 20-21 to any dst-port 49152-65535 in
# Block everything else
65500 deny log logamount 1000 ip from any to any
```



This is a minimal configuration and can be used with a client. We assume en0 as the local interface and 192.168.2.0/24 as local network. You can change those values on rules 12310.

This conf allows local clients to connect to remote servers (browsing, mail, ftp, p2p...) but closes ALL local services except local port 56789. This is achieved in rule 11000. You can remove port 56789 and add other ports to this rule if you need to run local services or to open local ports (for example for BitTorrent). If you don't run local services you can delete rule 11000.

This configuration has been created with Mac OS X Server using Server Admin, and with WaterRoof adding rules 10000, 12400, 12410. It can be used also on Mac OS X. If you add a divert rule as first rule, this ruleset can be used for a dual homed firewall/router, assuming en0 is the LAN interface not WAN. Remember that for a router/firewall to work on OSX you need: 1) 2 physical interfaces WAN and LAN 2) enable sysctl for forwarding 3) configure natd daemon (for NAT and port forwarding) 4) add IPFW divert rule. Obviously you can do these things with WaterRoof in a few clicks. See "NAT" section of this user guide or click "?" on NAT WaterRoof window.

WaterRoof tips

Golden rule: when you find a good configuration, backup it using the “Export” tool or deploy it using “Injectors”.

Second golden rule: Rebooting is not a way to fix ipfw issues.

If you mess up everything, the first thing to do is to flush ipv4 and ipv6 rules and pipes. Do it from the “Firewall” menu.

But if you want to really clean your system and restore factory settings so you need to “Clean System” from the “System” menu. This removes all scripts, plists, configuration files and deletes all active rules, disables forwarding, logging and natd.

After that you can safely trash WaterRoof (or NoobProof).

It's not safe to mix firewall tools. If you use WaterRoof please avoid using other firewall frontend, as it may be confusing.

If you do edit rules outside WaterRoof (e.g. from the Terminal), the WaterRoof static rules window may not represent current rules correctly. Click the “Update” button to update the rules table if you are not sure.

In WaterRoof many windows has the “?” button. Those buttons open contextual menus that will help you a lot. Use them.

You can also open manpages from the “Help” menu.

WaterRoof keyboard shortcuts

Use shortcuts to speed up your tasks:

- ⌘S export current ipv4/ipv6 firewall rules to file
- ⌘O import ipv6 and/or ipv4 firewall rules from file
- ⌘W close active window (except main window)
- ⌘M minimize active window

- ⌘0 **flush** ipv4 and ipv6 rules, pipes and queues
- ⌘1 open **firewall rules** window
- ⌘2 open **dynamic rules** window
- ⌘3 open **bandwidth** management window
- ⌘4 open network **processes** window
- ⌘5 open **connections** window (active and listening)
- ⌘6 open **connections** window (active only)
- ⌘7 open network **interfaces** window
- ⌘8 open **WHOIS and DNS** window
- ⌘9 open analyze **foreign hosts** window
- ⌘L open **firewall logs** window
- ⌘X **clean system** from WaterRoof files
- ⌘Q quit WaterRoof

WaterRoof notes

WaterRoof is a Cocoa-AppleScript application coded with XCode using AppleScript Studio and bash

WaterRoof is freeware and open source.

WaterRoof is not and will never be available on the Apple AppStore. WaterRoof and hanynet.com are not officially Apple certified and will never be. We will develop WaterRoof as long as Apple will support both IPFW and ApplescriptStudio applications on their operating systems.

WaterRoof is distributed under the terms of the **NoLicense Public License (NLPL)** which actually has no terms. (:D)

For more info please visit:

<http://www.nolicense.org>

For more info, feature requests and bugs report please mail to:

hany@hanynet.com or

hanymac@gmail.com

Check for updates and get source code at:

<http://www.hanynet.com/waterroof>



PAYPAL DONATION ACCEPTED !

hany@hanynet.com



WE ACCEPT BITCOIN DONATIONS !
16UvmZcqEEYT5gYrTaGrh82d12726fQi5x

Thank you for supporting free software development.

by Hany El Imam • www.hanynet.com

[WaterRoof Quick Reference](#)

