



Objective-See
a non-profit 501(c)(3) foundation.



tools



blog

Support Us!

Netiquette



↓ [download](#)

In today's connected world, it is rare to find an application or piece of malware that *doesn't* talk to a remote server.

Netiquette is a network monitor that allows one to explore all network sockets and connections, either via an interactive UI, or from the commandline.



Supported OS: macOS 10.10+



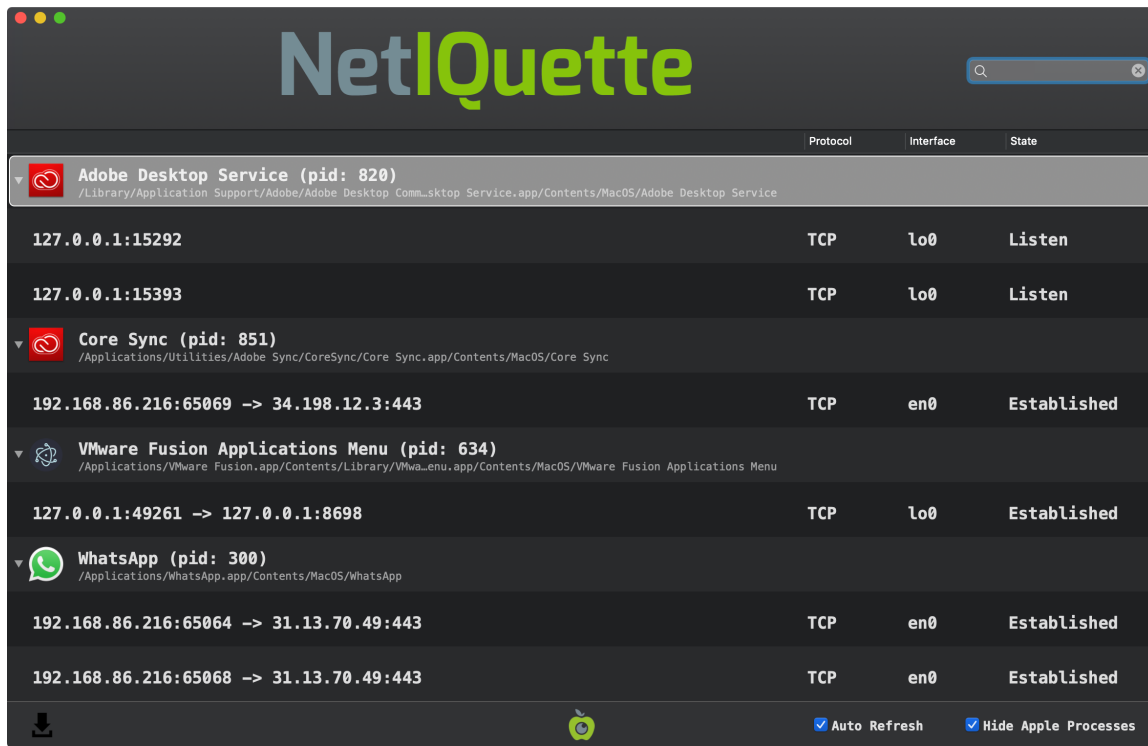
Current version: 2.1.1 ([change log](#))



Zip's SHA-1: F74C466606726D8CC912E44ED18132A7166C27CA



Source Code: [Netiquette](#)



The screenshot shows the Netiquette application window with the title bar and a search bar. The main content area displays a list of active network connections for several applications. The connections are organized by application, with each application name and PID expanded to show its file path. The connections are listed in a table with columns for IP addresses, ports, protocols, interfaces, and states.

| Application (pid) | Path | Local Address | Local Port | Protocol | Interface | State |
|--|--|----------------|------------|----------|-----------|-------------|
| Adobe Desktop Service (pid: 820) | /Library/Application Support/Adobe/Adobe Desktop Comm_sktp Service.app/Contents/MacOS/Adobe Desktop Service | 127.0.0.1 | 15292 | TCP | lo0 | Listen |
| | | 127.0.0.1 | 15393 | TCP | lo0 | Listen |
| Core Sync (pid: 851) | /Applications/Utilities/Adobe Sync/CoreSync/Core Sync.app/Contents/MacOS/Core Sync | 192.168.86.216 | 65069 | TCP | en0 | Established |
| | | 34.198.12.3 | 443 | | | |
| VMware Fusion Applications Menu (pid: 634) | /Applications/VMware Fusion.app/Contents/Library/VMwa_enu.app/Contents/MacOS/VMware Fusion Applications Menu | 127.0.0.1 | 49261 | TCP | lo0 | Established |
| | | 127.0.0.1 | 8698 | | | |
| WhatsApp (pid: 300) | /Applications/WhatsApp.app/Contents/MacOS/WhatsApp | 192.168.86.216 | 65064 | TCP | en0 | Established |
| | | 31.13.70.49 | 443 | | | |
| | | 192.168.86.216 | 65068 | TCP | en0 | Established |
| | | 31.13.70.49 | 443 | | | |

At the bottom of the window, there are two checked checkboxes: "Auto Refresh" and "Hide Apple Processes".

Special mahalo to [Jonathan Levin](#) for inspiring the creation of this tool! 🙏

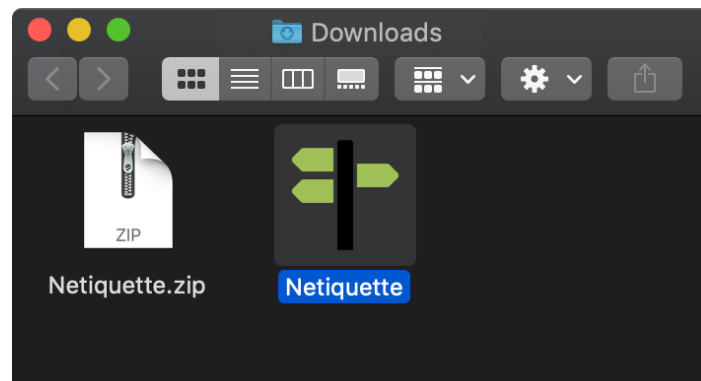
Netiquette, is a lightweight **open-source** network monitor designed exclusively for macOS.

Via its simple intuitive UI, you can quickly:

- view all network connections
- uncover any listening sockets
- filter and export network events

To use Netiquette, simply **download** the zip archive containing the application. Depending on your browser, you may need to manually unzip the application by double-clicking on the zipped archive.

As there is nothing to install, simply double-click on 'Netiquette.app' to launch the network monitor.



When launched, Netiquette will display all active network connections and sockets. By default, it will auto-refresh every few seconds (to ensure up-to-date network information is continually displayed). Also, networking information belonging to Apple processes, by default is not shown. Both these options can be toggled off and on via check boxes in the UI:

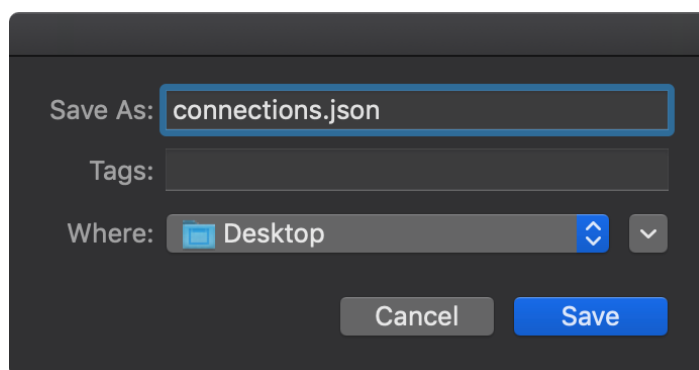


This website uses cookies to improve your experience.

There is also an input box, (top-right of the UI), that will filter displayed network connections. Filter on things such as process (name, path, pid) or network information (ip address, port, interface, state, etc). For example, type "Listen" to reveal all network sockets that are listening for (possibly) remote network connections:



Click the "save" icon (bottom left), to export the network connections to a JSON file:



The saved JSON contains detailed information about processes (pid, path, signing information, etc), and their network connections or sockets. For example, here we see output related to netcat (`/usr/bin/nc`) listener on port 666:

```
[{
  "process": {
    "pid": "5885",
    "path": "/usr/bin/nc",
    "signature(s)": {
      "signatureIdentifier": "com.apple.nc",
      "signatureStatus": 0,
      "signatureSigner": 1,
      "signatureEntitlements": {
        "com.apple.security.network.client": true,
        "com.apple.security.network.server": true
      }
    },
    "signatureAuthorities": [
      "Software Signing",
      "Apple Code Signing Certification Authority",
      "Apple Root CA" ]
  },
  "connections": [{
    "interface": "",
    "protocol": "TCP",
    "localAddress": "0.0.0.0",
    "localPort": "666",
    "remoteAddress": "0.0.0.0",
    "remotePort": "0",
    "remoteHostName": "n/a",
    "state": "Listen"
  }]
}]
```

COMMANDLINE INTERFACE

Netiquette can also be run via the commandline, to list all active network connections.

Note:

This is similar to Apple's built-in nettop tool. However, Netiquette provides more comprehensive information such as:

- full process path
- connection filtering
- code-signing information
- detailed output in standard JSON

Execute the Netiquette binary (note: specify the full path to the Netiquette binary *within* its application bundle) with `-h` or `-help` to display information about the self-explanatory commandline options:

```
$ Netiquette.app/Contents/MacOS/Netiquette -h

NETIQUETTE USAGE:
-h or -help  display this usage info
-list       enumerate all network connections
-names      resolve remote host names (via DNS)
-pretty     JSON output is 'pretty-printed' for readability
-skipApple  ignore connections that belong to Apple processes
```

The `-list` commandline flag will generate a (JSON) list of all active network connections and sockets:

```
$ Netiquette.app/Contents/MacOS/Netiquette -list -pretty

[
  {
    "process" : {
      "pid" : "5885",
      "path" : "\usr\bin\nc",
      "signature(s)" : {
        "signatureIdentifier" : "com.apple.nc",
        "signatureStatus" : 0,
        "signatureSigner" : 1,
        "signatureEntitlements" : {
          "com.apple.security.network.server" : true,
          "com.apple.security.network.client" : true
        },
        "signatureAuthorities" : [
          "Software Signing",
          "Apple Code Signing Certification Authority",
          "Apple Root CA"
        ]
      },
    },
    "connections" : [
      {
        "remoteHostName" : "n/a",
        "protocol" : "TCP",
        "interface" : "",
        "localAddress" : "0.0.0.0",
        "state" : "Listen",
        "remotePort" : "0",
        "localPort" : "666",
        "remoteAddress" : "0.0.0.0"
      }
    ]
  },
  ...
]
```

Note:

To capture the output from Netiquette, (as it writes to STDOUT), simply pipe it to a file out of your choice:

```
$ ./Netiquette.app/Contents/MacOS/Netiquette -list > /path/to/some/file.json
```