



**SINE NOMINE**  
ASSOCIATES

# OpenAFS macOS client status report

Marcio Barbosa  
2022 OpenAFS Workshop



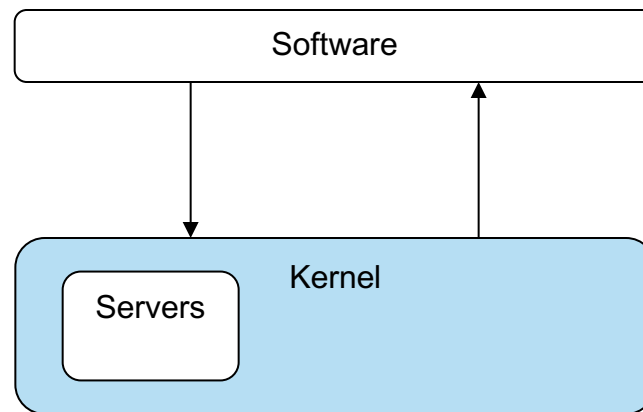
## AGENDA

- XNU
- KERNEL EXTENSIONS
- CODE SIGNING AND NOTARIZATION
- SYSTEM EXTENSIONS
- DEPRECATED KERNEL EXTENSIONS
- SOCKPROXY
- APPLE SILICON
- UNIVERSAL BINARIES
- PREFERENCE PANE
- PRIVILEGED HELPER TOOL
- MACOS MONTEREY
- MACOS VENTURA



# XNU

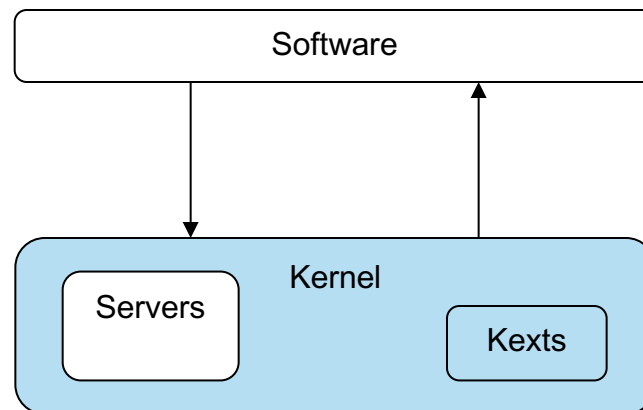
- Monolithic x Microkernels.
- Microkernel architecture? No, XNU is a hybrid kernel.





## KERNEL EXTENSIONS

- Runs in kernel mode, and therefore has full access to kernel space.
- Stability and the security can be compromised.





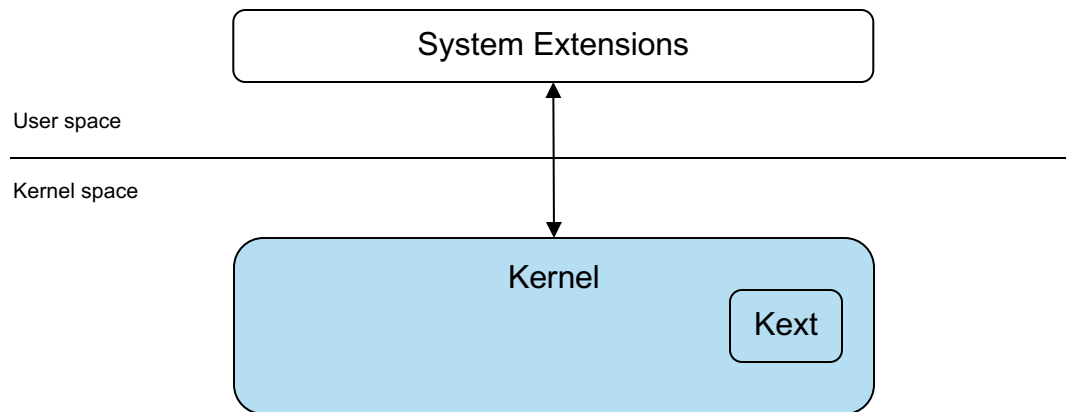
## CODE SIGNING AND NOTARIZATION

- On macOS 10.10, Kext will not be loaded if it is not signed.
- Since April 7, 2019, Kexts must be notarized.
- Patches:
  - <https://gerrit.openafs.org/#/c/13670>
  - <https://gerrit.openafs.org/#/c/13671>



## SYSTEM EXTENSIONS

- Apple introduced System extensions on macOS 10.15 (Catalina).
- Allows the extension of the OS without requiring kernel-level access.





## DEPRECATED KERNEL EXTENSIONS

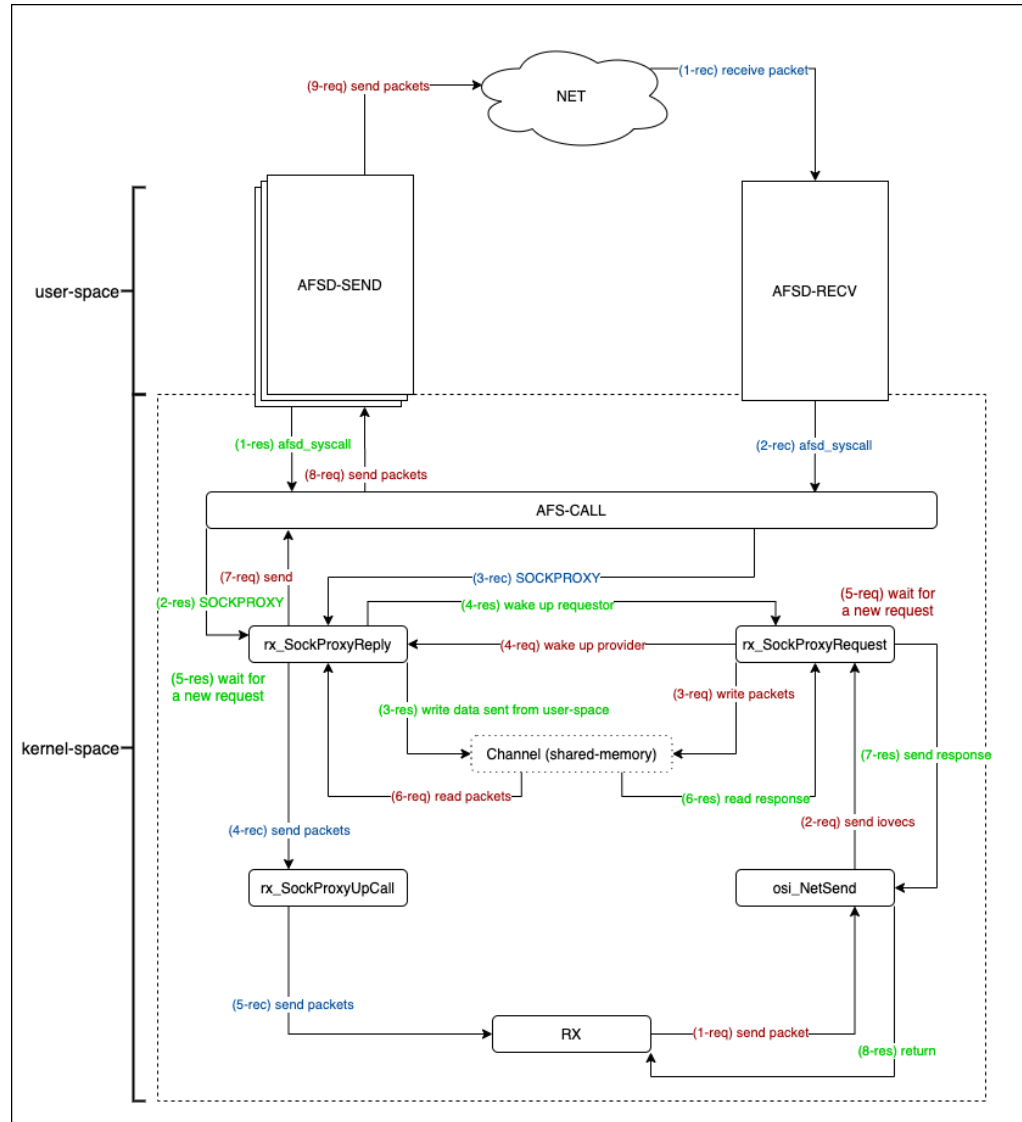
- Apple deprecating macOS kernel extensions.
- In macOS 11, Kexts using certain deprecated KPIs no longer load.
- Network kernel extensions is deprecated.
- No longer provides an in-kernel socket KPI, extensively used by RX.
- An effort by Apple to phase out the use of Kexts.



## SOCKPROXY

- Move pieces of RX to user-space.
- Daemons responsible for sending and receiving packets on behalf of RX.
- Patch:
  - <https://gerrit.openafs.org/#/c/14431>







## APPLE SILICON

- All Kexts must support the arm64e architecture.
- Pointer Authentication
  - Detect and guard against unexpected changes to pointers in memory.
- Variadic Functions
  - x86\_64 and arm64 architectures have different calling conventions.
- Patches:
  - <https://gerrit.openafs.org/#/c/14912>
  - <https://gerrit.openafs.org/#/c/14913>
  - <https://gerrit.openafs.org/#/c/14743>
  - <https://gerrit.openafs.org/#/c/14938>
  - <https://gerrit.openafs.org/#/c/14886>
  - <https://gerrit.openafs.org/#/c/14744>
  - <https://gerrit.openafs.org/#/c/14745>
  - <https://gerrit.openafs.org/#/c/14914>
  - <https://gerrit.openafs.org/#/c/14934>
  - <https://gerrit.openafs.org/#/c/14746>



## UNIVERSAL BINARIES

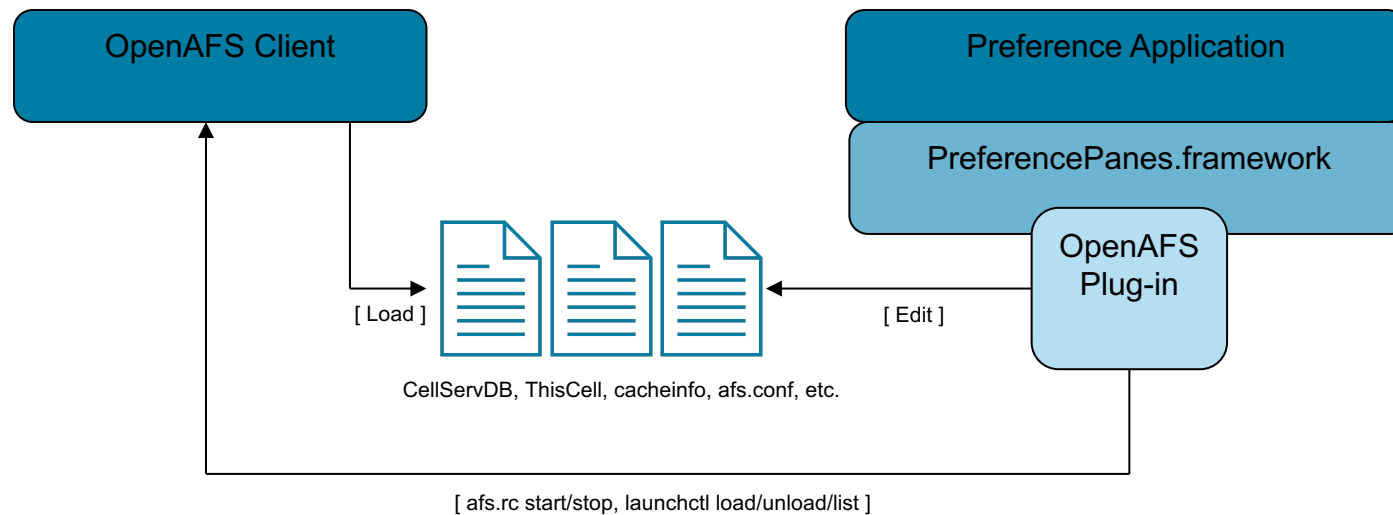
- Binaries for both architectures merged into a single binary.
- Creating an OpenAFS Universal package:
  - `$ ./regen.sh`
  - `$ ARCHFLAGS="-arch x86_64 -arch arm64" ./configure --enable-transarc-paths \`  
`--with-krb5-conf=/usr/bin/krb5-config --enable-debug --enable-debug-kernel \`  
`--libdir=/Library/OpenAFS/Tools/lib`
  - `$ ARCHFLAGS="-arch x86_64 -arch arm64" make dest`
  - `$ sudo sh src/packaging/MacOS/pkgbuild.sh -x --csdb <path/to/CellServDB>`  
`--app-key 'Developer ID Application: <your_dev_id_application>'`  
`--inst-key 'Developer ID Installer: <your_dev_id_installer>' <arch>/dest`
  - `$ sudo ./src/packaging/MacOS/notarize.pl <apple_id> <password> </path/to/dmg>`





## PREFERENCE PANE

- Requires elevated privileges.
- Method used is deprecated and no longer supported.





## PRIVILEGED HELPER TOOL

- Apple introduced a preferred method for managing privilege escalation.
- Privileged helper tool is installed by the application.
- Application delegates privileged operations to this helper tool.
  - Application and the privileged helper tool must be code signed.
  - Must be one of the applications that is approved to make requests to this helper.
- Principle of least privilege.
- Patches:
  - <https://gerrit.openafs.org/#/c/15032>
  - <https://gerrit.openafs.org/#/c/15033>
  - <https://gerrit.openafs.org/#/c/15034>
  - <https://gerrit.openafs.org/#/c/15035>
  - <https://gerrit.openafs.org/#/c/15036>



## PRIVILEGED HELPER TOOL

- src/platform/DARWIN/AFSPreference/Info.plist

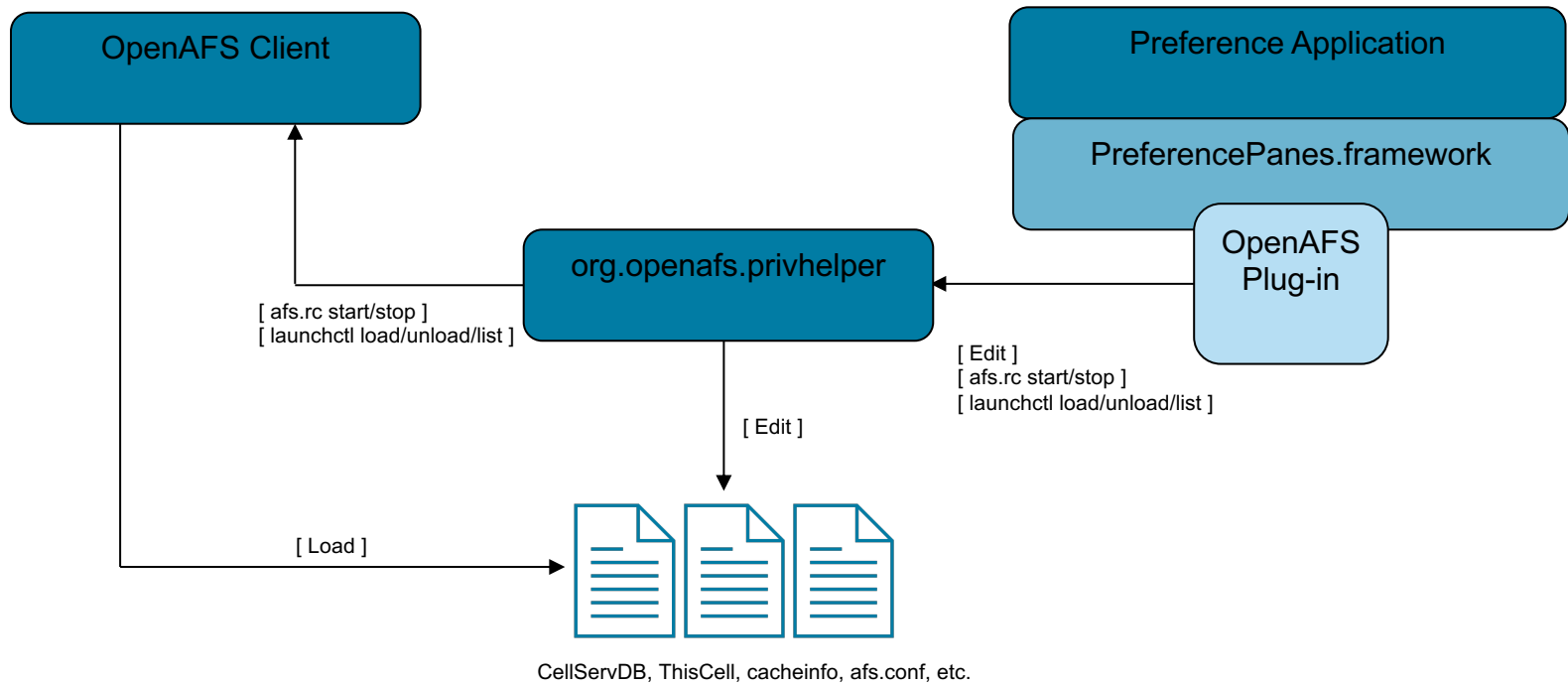
```
31     <key>NSPrincipalClass</key>
32     <string>AFSCCommanderPref</string>
33     <!-- Uncomment the following block and add your signature -->
34     <!--
35     <key>SMPrivilegedExecutables</key>
36     <dict>
37         <key>org.openafs.privhelper</key>
38         <string>identifier "org.openafs.privhelper" <signature> </string>
39     </dict>
40     -->
41 </dict>
42 </plist>
```

- src/platform/DARWIN/PrivilegedHelper/privhelper-info.plist

```
12     <string>1.0</string>
13     <!-- Uncomment the following block and add your signature -->
14     <!--
15     <key>SMAuthorizedClients</key>
16     <array>
17         <string>identifier "it.infn.lnf.network.openafs" <signature> </string>
18     </array>
19     -->
20 </dict>
```



# PRIVILEGED HELPER TOOL

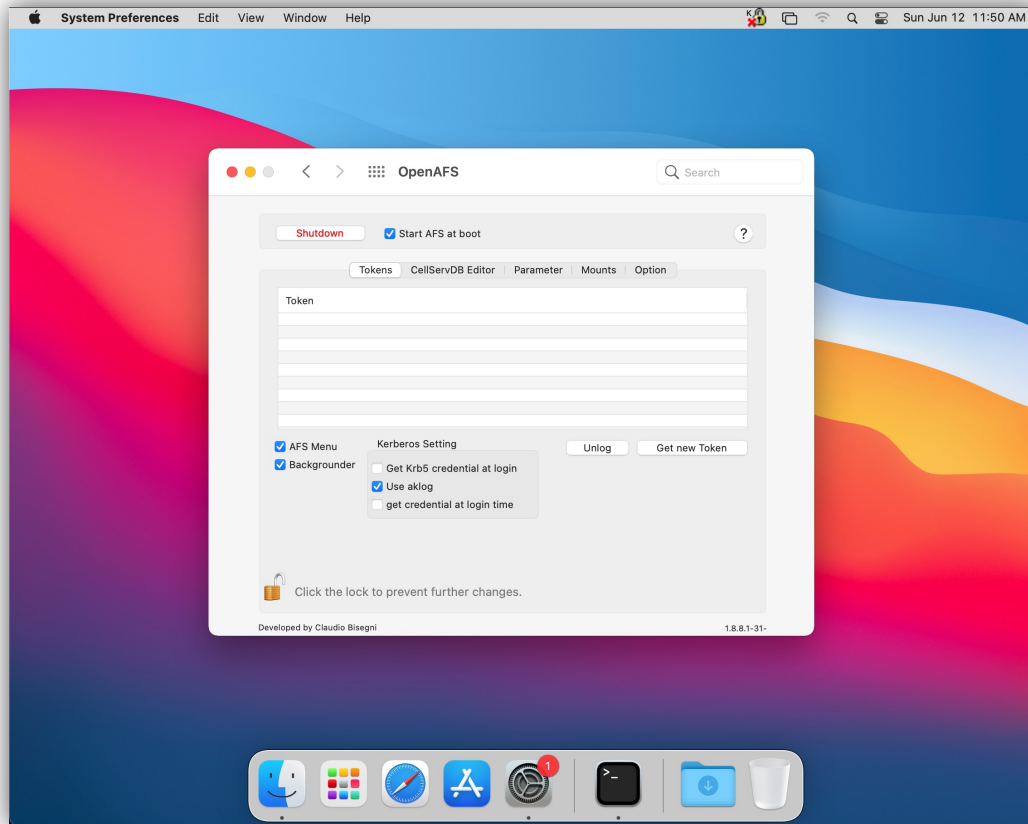






**SINE NOMINE**  
ASSOCIATES

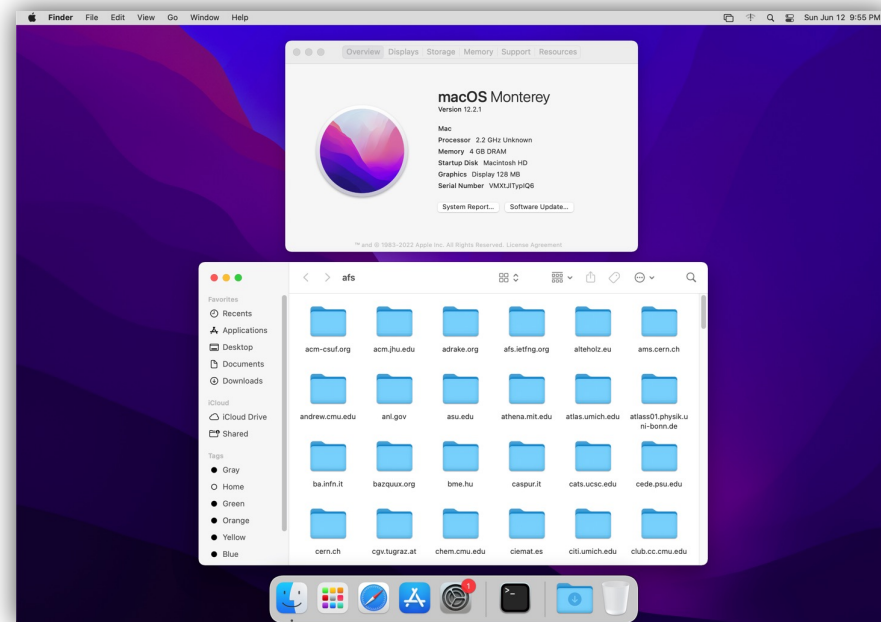
# PRIVILEGED HELPER TOOL





# MACOS MONTEREY

- Supports macOS 12.
- x86\_64 and arm64.
- Patches:
  - <https://gerrit.openafs.org/#/c/14922>
  - <https://gerrit.openafs.org/#/c/14923>
  - <https://gerrit.openafs.org/#/c/14924>
  - <https://gerrit.openafs.org/#/c/14925>

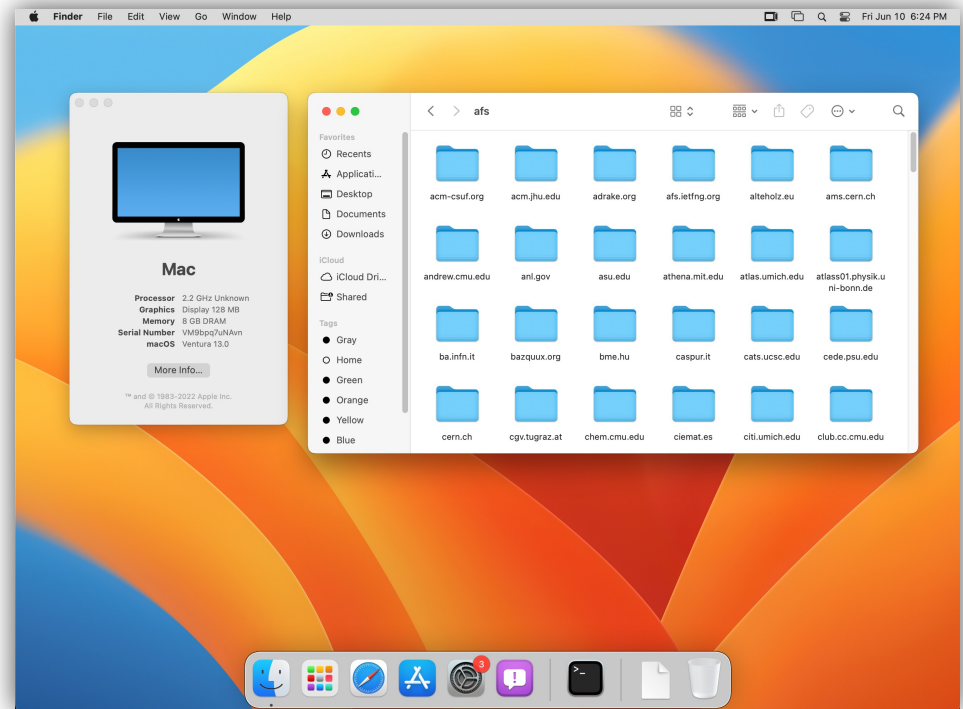




**SINE NOMINE**  
ASSOCIATES

# MACOS VENTURA

- Testing on macOS 13 beta.
- Patches not available yet.





**SINE NOMINE**  
ASSOCIATES

## CONCLUSION

- Apple continues to make significant changes.
- OpenAFS is fully functional on macOS!



**SINE NOMINE**  
ASSOCIATES

Thank you!