Malware-multzokatze Teknikak
(Malware Clustering Techniques)

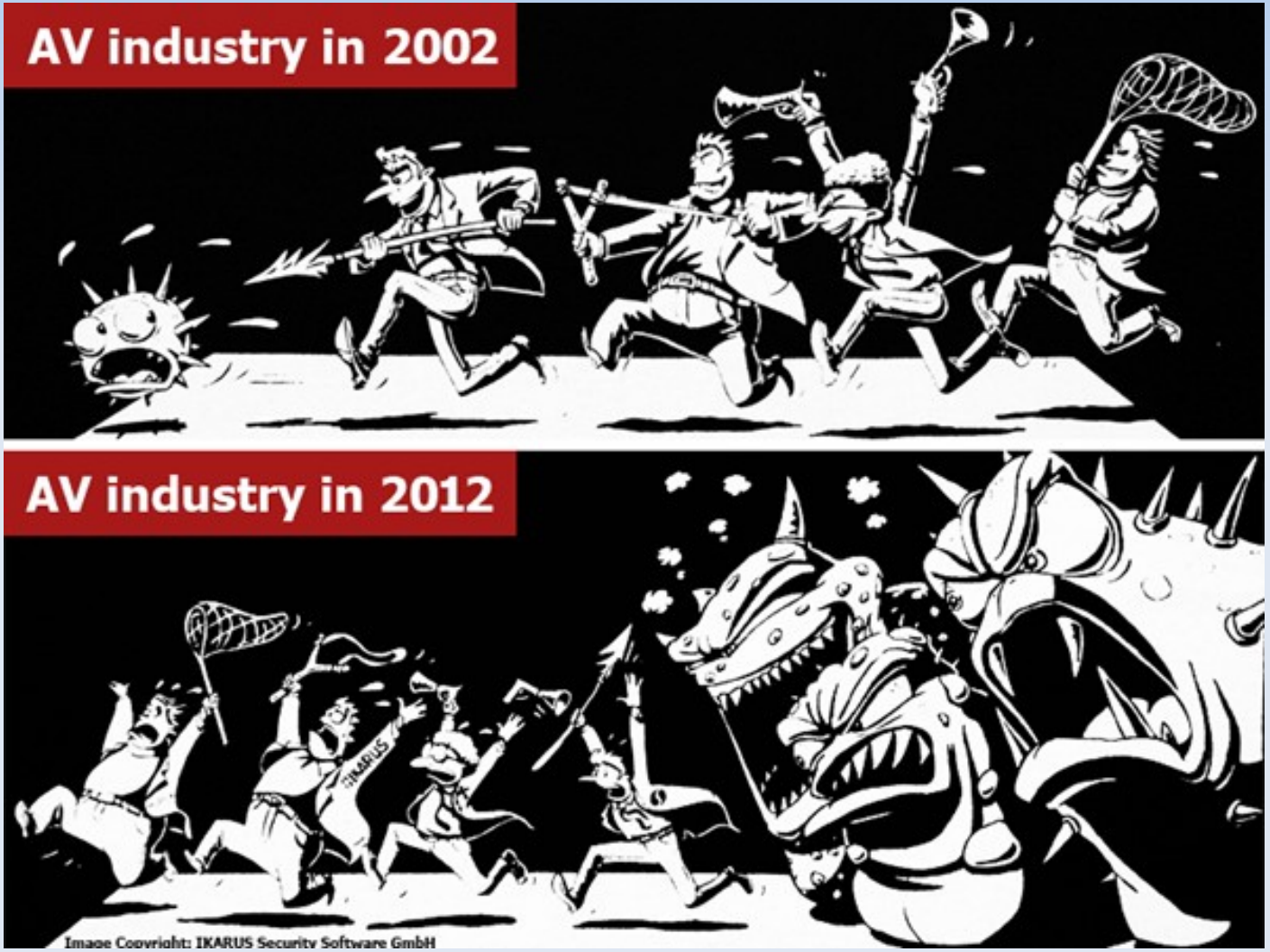
Joxean Koret

Hack & Beers, Donostia
2015

Malwarea

- Zer da malwarea?
  - EuskalTermen araberan: “*Ordenagailuei, sistemei eta sareei kalte egiteko diseinatutako programak, hala nola, birusak, harrak, Troiako zaldiak, ...*”
- Malware mota asko daude:
  - Birusak: fitxategiak infektatzen dituen programak edo Script-fitxategiak.
  - Troiako zaldiak: Itxuraz inolako kalterik egiten ez duen programa baten barnean ezkutatutako programa kaltegarria.
  - ...

Malwaren Eboluzioa



Malware Eboluzioa: 2000an

- 2000. urtean, astean behin ikusten zen birus edo har (“Worm”) berri bat.
  - Antibirus batean teknikari guztiek lan egiten zuten fitxategi horretan ala, kasu txarrenetan, azken agertutako bi horietan.
  - Normalean, lanak egun bat edo, askoz jota, aste bat irauten zuen.
    - 29A-ko birusak ez ziren hain errazak ;)
  - Batez ere, malware horiek sortuak izan ziren “ego” arrazoiengatik ala frogak egiteko.
    - “Begiratu egin daikedana!”, ”Begiratu asmatu dodan tramankulu barri hau!”, “Nirea handiagoa da!”, ...
- ...

Malware eboluzioa: 2010ean

- Gaur egun oso ezberdina da. Erabat. Orain, malwarea ez da sortzen ego arrazoiengatik ala ikasteagatik.
- Malwarea negozio borobil bat bihurtu da:
  - Gure banku datuak lapurtzeko.
  - Gure e-postako kontuak lapurtzeko.
  - Gure konputagailuak zonbie moduan erabiltzeko.
  - Gure makinak erabiltzeko erasotzeko benetako helburua.
  - Kripto-txanponak “minatzeko” (BitCoin, adibidez).
  - ...

Malware eboluzioa: 2010ean

- Atzo (2000. urtean) astero edo agertzen ba zen birus edo har berri bat, orain, milaka agertzen dira.
  - Panda Antibirusen araberan, egunero 2013an 82.000 malware berri agertu ziren.
    - Nik uste dot “fitxategi berriak” esan nahi zutela...
  - Kaspersky-ren araberan, egunero 315.000 fitxategi berriak detektatzen dira eta, euren hitzetan, egunero 125.000 malware berri agertzen dira.
- Nahiz eta enpresa guztiek propaganda egin, ikus dezakegunez, zenbakiak erabat aldatu dira.
- ...

Malware eboluzioa

- Demagun 50.000 malware fitxategi berriak baino ez direla agertzen egunero.
  - Atzo, birus/har bat agertzen zenean ia-ia teknikari guztiak jartzen ziren lan egiten malware berberan.
  - Gaur... 50.000 teknikari behar ditu antibirus enpresa batek?
  - Noski, funtzionamendua aldatu behar izan zuten.
  - Erantzuna “automatizazioa” izan zen (eta izango da bihar).
- ...

Automatizazioa

- Automatizazioak sortu daitezke:
  - Multzoak sortzeko.
  - Familiak sortzeko.
  - Sinadurak automatikoki sortzeko.
    - Baita positibo-faltsoak kentzeko ere...
  - …
- Baina... zelan egin daiteke hau?
- Aukera batzuk ditugu. Niri gehien gustatzen zaizkidanak logika lausoa oinarritzen dira:
  - Edukien oinarritutako algoritmo generikoak.
  - Eta, batez ere, grafoetan oinarrituak.

Hash Lausoak (“Fuzzy Hashing”)

- Zer da hash lauso bat?
  - Logika lausoan oinarritutako algoritmo baten irteera.
  - Adibiderik errazena:
    - Logika aritmetikoa: Igorrek **20** urte ditu.
    - Logika lausoa: Igor **gaztea** da.
    - Logika lausoak ez du zehazten zenbat urte dituen, gaztea edo zaharra ote den besterik ez.
- Zelan aplikatu daiteke logika lausoa informatikan?
  - ...

Logika lausoa (Fuzzy Hashing)

- Algoritmo zein tresna asko daude. Algoritmoek honako propietate hauek izan behar dituzte:
  - Irteera fitxategia baino askoz txikiagoa izan behar da. Hash txikia izan behar da (MD5aren antzekoa, adibidez).
  - Difusiorik ez, ala difusio txikia. Hauxe da, fitxategi baten karaktere gutzi batzuk aldatzeak ez du aldatu behar sinadura osoa, dagokion partea besterik ez (eta hori aldatzekotan).
  - Konfusiorik ez. Lehenengo byteak aldatzen badira, sinaduraren lehenengo byte(ak) baino ez da/dira aldatuko, eta hori aldatzen bada.
  - Talka-abiadura egokia (“Collision rate”). Kolisioak erabiliko ditugu, baina ezin dugu eduki talka gehiegi. Hau da: talka-abiadura egokigarria izan behar da.
- ...

Logika lausoa (Fuzzy Hashing)

- Zorionez, ez dugu sortu behar horrelako algoritmorik nahi ez badugu: kode-irekiko tresnak eskura ditugu.
- Ezagutzen ditudan adibide publiko bakarrak:
  - SSDeep. Kode irekia, GPLa. Nahiko ona, baina blokearen tamaina datuaren tamainaren oinarrituta dago eta, zoritxarrez, ezin da egokitu.
    - http://ssdeep.sourceforge.net
  - DeepToad. Nik egindakoa :) Kode irekia, LGPLa. Blokearen tamaina, talka-abiadura eta beste gauza asko egokitu daitezke.
    - https://code.google.com/p/deeptoad/

SSDeep/DeepToad

- Nola funtsionatzen duten tresna hauek?
  - Fitxategiaren edukiak irakurri eta zatitzen dituzte blokeetan.
  - Bloke bakoitzari funtsio matematiko erraz bat aplikatzen dute. Adibidez: $\sum(datuak) \, mod \, 256$
  - Bloke guztien irteera erabili sortzeko sinadura (hash-a).
  - Fitxategiak konparatzeko, sinadura (hasha) erabiltzen da.
    - Sinadura berbera bada, horrek esan nahi du fitxategiak “oso antzekoak” direla.
    - Sinadura ia-ia bera bada, antzekoak direla.
  - ...

DeepToad

```
sh-4.2$ cp /bin/ls .
sh-4.2$ cp ls ls2
sh-4.2$ echo "TR0L0L0" >> ls2
sh-4.2$ md5sum ls ls2
fa97c59cc414e42d4e0e853ddf5b4745  ls
72d73ee2486bddd3c1ae56358b8684db  ls2
sh-4.2$ deeptoad ls ls2
NTWPj4+PiIiIiLm5ubklJSUl2tra2gMD;j4+IiLm5JSXa2gMDDAxpaTw81dUJCSQk;c3P29pqaZWU/P7q6GBhSUtDQ40BCQqSk;ls
NTWPj4+PiIiIiLm5ubklJSUl2tra2gMD;j4+IiLm5JSXa2gMDDAxpaTw81dUJCSQk;uLjW1o0DFxf19QYGxsYdHbm55eWtrY60;ls2
```

Logika lausoa eta malwareak

- Algoritmo/tresna hauek oso lagungarriak dira malware multzatzeko:
  - Fitxategi asko malware berbera dira baina byte batzuk aldatu dira hash kriptografikoa aldatzeko.
  - Aldaketa txikiak badira, logika lausoa erabiliz multzatu ditzakegu antzeko fitxategiak.
  - Familiak sortzen lagundu dezake horrelako tresna batek.
- Ikus ditzagun adibide batzuk...

Fitxategi asko eta malware ezberdin batzuk...

```
sh-4.2$ md5sum *
78926bd67c736c44e9e8fffe9e54b21c  0de37a1515db37a7cfa3784137fa433e3b84b2e4
502cc63be0062200f174784c572d4234  0de38aab6fa2a4a71d6f492cf0d0b1fef3530078
19364df01191c6b4f653f3062e24cce6  0de42569ed57372b6b105f2553eed370c956624b
249f0556dbfe5bb62834e7b0c782c5c3  0de461c32c4db63bb9cbfe0c54473d53dc255ae1
7deecf2a2d3607b608a4c12d947bbdac  0de782c678dc1243f48afa6782f366d2def9ad46
966fc361047a445ff23208496f4fa837  0dee593bc99e3f7c7c9fade96085fdea03eafc6b
61118656a23648f41456ef7b134f3731  0deeea991703ddd3bcb00c47690d41519c9efeb6
03bc7db1d9471c0a74625cdc284fa6aa  0df04911f1504a46fedfeeed73c99e535919257a
7438064ae4e399576061b499082f7883  0df11ba2d8a85abb214ca68b465617021f899482
778b602e63c2c323d88ed0ab2570fcbe  0df26faef6c6079d64c4b4068b7fd46aa9ece5ff
1675ad43130f1ef1845a0c86479b0aba  0df2ab5d0ecd515e7a15091109f5cb93bd9066cd
b41a4cc2caa7fb8d4e008bf6eba67688  0df37b6bde4c1dcc02ed2eeb1add7ca46cf688cb
ae6e80c36a1e5edfc13a23c93f7f1902  0df3d629060f0c21ca9c55ee7682eb8903ac437d
89445a1571c5d8f559d113f5f2c9a42f  0df3f9037fb792b97da0c8231e7b304d455789a7
7a1db943a12d12a0d5266d9ad8e774e2  0df4f72f5af9d9b41b27752aa4b1bf86ce416220
046f6142d9bac3a2ff3b838e5a0d2a10  0df68649dc71fbf51801d732f44c193a04689f93
0291e0d591276efc10ead572d900ac3a  0df79baf776139cace82915b1723c922cb6d576c
b9122c0ef4fe2854f26c32facbf61977  0dfab1632dc3edbe2e44371a545dec6d061e750b
701ef12e99d6bdeaf0141fd82e8163bd  0dfcb6d56d0de987880eb08e36eb5e628447bde4
8cf7db63c38afc90d3109acc72a551f6  0dfe4e246e132be9e85ff9d1c0f62a85fcffcef5
sh-4.2$ clamscan -r --infected .
./0dfcb6d56d0de987880eb08e36eb5e628447bde4: HTML.Trojan.Blackhole-2 FOUND
./0de782c678dc1243f48afa6782f366d2def9ad46: Trojan.Blackhole-481 FOUND
./0df04911f1504a46fedfeeed73c99e535919257a: Win.Adware.Installerex-11 FOUND
./0dfe4e246e132be9e85ff9d1c0f62a85fcffcef5: Win.Adware.Installerex-2 FOUND
./0de38aab6fa2a4a71d6f492cf0d0b1fef3530078: Win.Adware.Installerex-12 FOUND
./0df3f9037fb792b97da0c8231e7b304d455789a7: JS.Trojan.Iframe-5 FOUND
./0df2ab5d0ecd515e7a15091109f5cb93bd9066cd: Win.Adware.Installerex-2 FOUND
./0deeea991703ddd3bcb00c47690d41519c9efeb6: Win.Adware.Installerex-2 FOUND
./0df4f72f5af9d9b41b27752aa4b1bf86ce416220: Win.Trojan.Installerex-1 FOUND
./0df79baf776139cace82915b1723c922cb6d576c: JS.Trojan.Agent-22 FOUND
./0de461c32c4db63bb9cbfe0c54473d53dc255ae1: Win.Adware.Installerex-11 FOUND
```

Zeintzuk dira antzekoak?

```
sh-4.2$ deeptoad .
Processing file ./0df11ba2d8a85abb214ca68b465617021f899482 ...
eHiPj4aGlJRCQru70tJsbPPz40AVFU9P;./0df37b6bde4c1dcc02ed2eeb1add7ca46cf688cb
eHiPj4aGlJRCQru70tJsbPPz40AVFU9P;./0de42569ed57372b6b105f2553eed370c956624b
h4dGRkpKdnYjIwgIMTH0ztvbLCxdXdbW;./0df26faef6c6079d64c4b4068b7fd46aa9ece5ff
h4dGRkpKdnYjIwgIMTH0ztvbLCxdXdbW;./0dfab1632dc3edbe2e44371a545dec6d061e750b
9fUdHR0dMDAwMI2NjY13d3d3V1dXV7u7;./0df79baf776139cace82915b1723c922cb6d576c
ysoMDLW1BgbR0V5eHb6Lj6+v4uL19Y60;./0df68649dc71fbf51801d732f44c193a04689f93
gID39/f3Y2NjY0VFRUW5ubm5cnJycsTE;./0df04911f1504a46fedfeeed73c99e535919257a
gID39/f3Y2NjY0VFRUW5ubm5cnJycsTE;./0dfe4e246e132be9e85ff9d1c0f62a85fcffcef5
gID39/f3Y2NjY0VFRUW5ubm5cnJycsTE;./0de38aab6fa2a4a71d6f492cf0d0b1fef3530078
gID39/f3Y2NjY0VFRUW5ubm5cnJycsTE;./0df2ab5d0ecd515e7a15091109f5cb93bd9066cd
gID39/f3Y2NjY0VFRUW5ubm5cnJycsTE;./0deeea991703ddd3bcb00c47690d41519c9efeb6
gID39/f3Y2NjY0VFRUW5ubm5cnJycsTE;./0df4f72f5af9d9b41b27752aa4b1bf86ce416220
gID39/f3Y2NjY0VFRUW5ubm5cnJycsTE;./0de461c32c4db63bb9cbfe0c54473d53dc255ae1
s70XI5eXcnJychcXFxehoaGhp6enpxIS;./0dee593bc99e3f7c7c9fade96085fdea03eafc6b
x8cqKioqg40Dg2ZmZmaVlZWV6urq6g8P;./0df3d629060f0c21ca9c55ee7682eb8903ac437d
hoYdHR0dhoaGhgcHBwefn5+f6+vr67S0;./0de37a1515db37a7cfa3784137fa433e3b84b2e4
eXkfH7e3NDTJyWlp4uJZWTMz9vYCAi0t;./0dfcb6d56d0de987880eb08e36eb5e628447bde4
urooKDo6qqqcnDk5XV3s7Kysenre3qqq;./0de782c678dc1243f48afa6782f366d2def9ad46
u7s0DkJCw8PNzSUlraOXF6Kir6/j4/Pz;./0df11ba2d8a85abb214ca68b465617021f899482
p6djY2Nj4+Pj446Ojo6MjIyMXV1dXWtr;./0df3f9037fb792b97da0c8231e7b304d455789a7
```

Zer argipidea ematen digu honek?

- Alde batetik, ez ditugu 21 fitxategi aztertzeko, orain baditugu 3 talde eta 10 fitxategi solte.
  - Egin beharreko lana murriztu dugu.
- Antzekoak diren fitxategiak, malware berbera dira.
  - Aldaketak oso txikiak dira.
- Metodo honek, baina, arazo asko ditu...
  - Ezin da jakin familia berbera diren.
    - Badakigu antzekoak direnentz, baina gehiagorik ez.
  - Enpaketatu badaude, enpaketatuko data multzokatuko dugu.
    - Ezin da jakin, berez, zer dagoen hor barruan.

Arazo gehiagorik

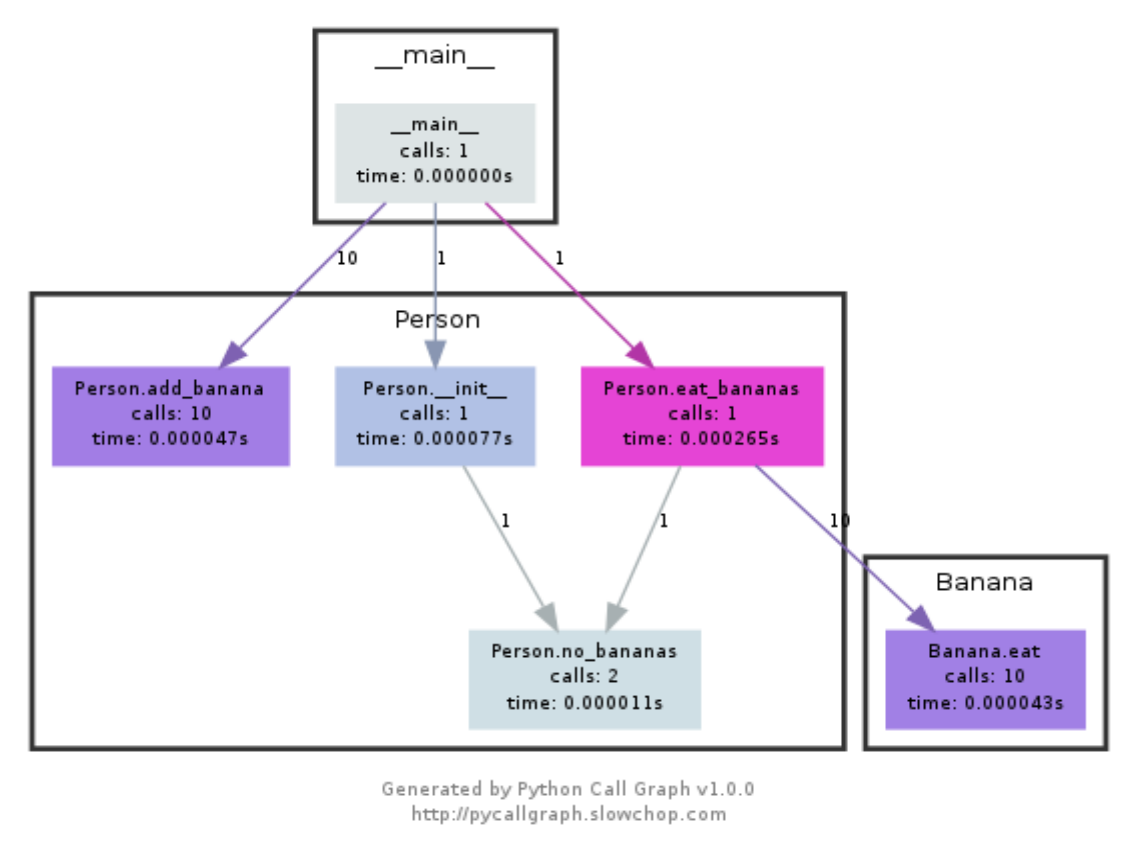
- Beste horrelako algoritmoen arazo batzuk:
  - Ez dago oinarrituta fitxategiaren formatoan.
  - Agnostikoa da, berdi zaio zer ematen diogun, fitxategiaren byteak multzokatuko ditu.
- Abiadura-talka oso handia da oro har.
  - Adibidez, horrelako sinadura/metodo ezin da erabili bakarrik antibirus batean.
  - Oso erraza da kolisioak topatzea goodwarekin.
- Beste aukerarik al dago?

Grafoen oinarritutako sinadurak

- Exekutablei dagokienez, beste aukera batzuk ditugu “eskura”: Programaren grafoak konparatu.
- Nola “zatitu” daiteke programa bat grafoetan?
  - Programa baten funtsio guztien grafoa eta euren loturak grafo batean erakutsi ditzakegu.
  - Funtsio bakoitzaren oinarri-blokeak eta euren loturak grafo batean erakutsi ditzakegu.
  - Ikus dezagun nola...

Dei-grafoak (Call graph)

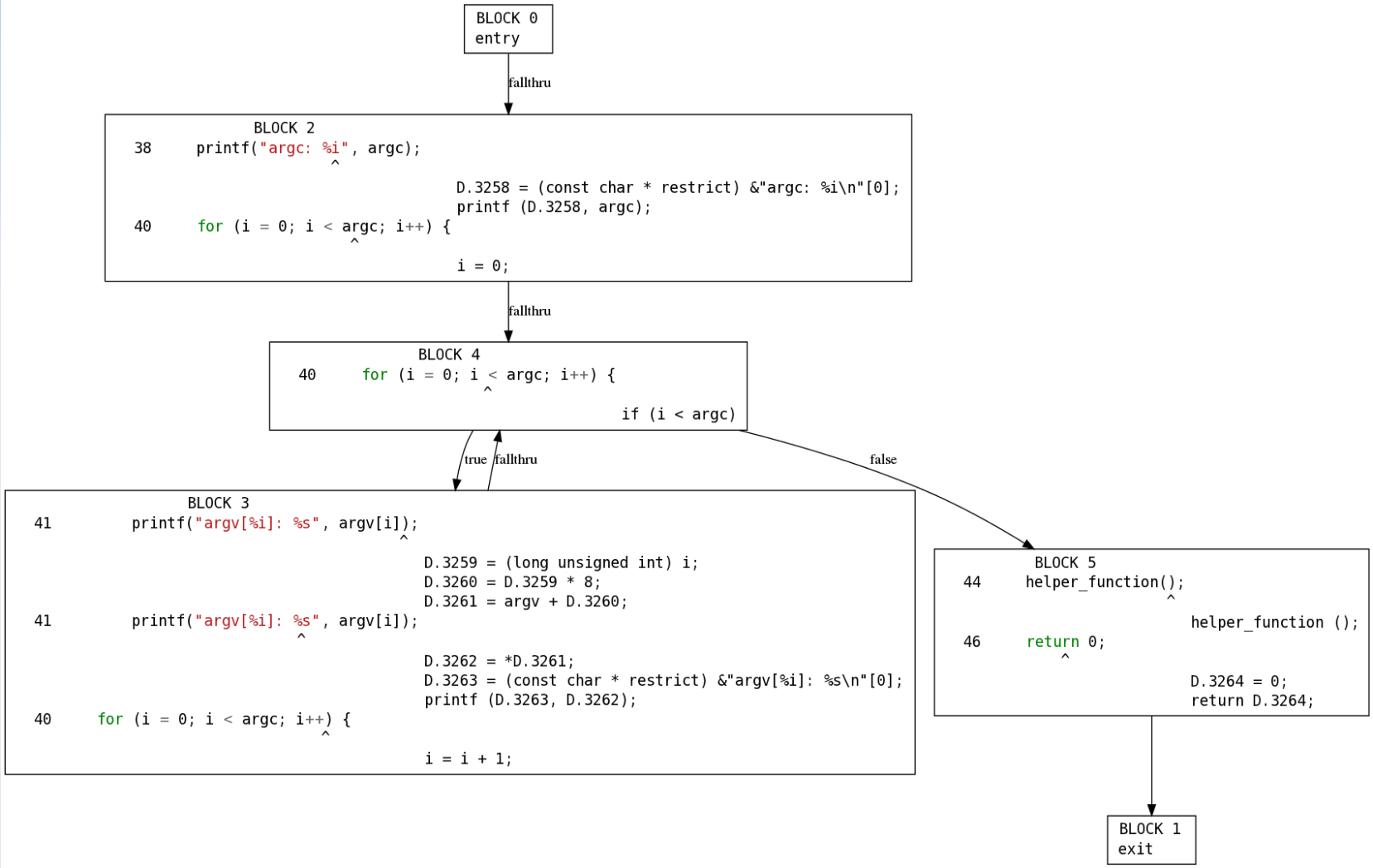
- Dei-grafo batek erakusten ditu zer futsioak dauden programa batean eta nola erlazionatzen diren.



Kontrol-fluxu grafoak (Control Flow Graph)

- Kontrol-fluxu grafo batek erakusten ditu funtsio batean zer oinarri-blokeak dauden eta nola errelazionatzen dira euren artean.
- Zer da oinarri-bloke bat?
  - Instrukzio batzuk segidan baldintza-perpaus bat topatu arte.
    - Ez da %100 zehatza, baina “good enough”.
  - Hobeto ikusiko da hurrengo irudian...

Oinarri-blokeak (Basic Blocks)



Ados, eta orain zer?

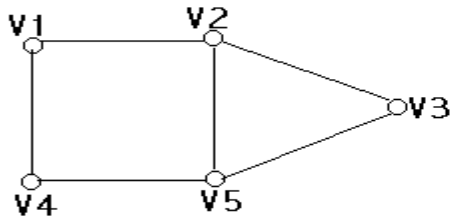
- Badakigu programa bat zatitu dezakegula 2 grafo motan:
  - Kontrol-fluxu grafoetan (CFG)
  - Dei-grafoetan (CG)
- Baina, zelan konparatu ditzakegu grafoak?
  - Aukera asko daude, baina batzuk ez dira oso egokiak.
  - Ikus ditzagun...

Auzokidetasun-matrizeak (Adjacency Matrix)

- Erabil ditzakegu auzokidetasun-matrizeak konparatzeko 2 grafoak.

Adjacency Matrix for a Simple Graph:

Example: Given a graph G as follows



	V1	V2	V3	V4	V5
V1	0	1	0	1	0
V2	1	0	1	0	1
V3	0	1	0	0	1
V4	1	0	0	0	1
V5	0	1	1	1	0

From the chart above, the adjacency matrix for the graph G is:

0	1	0	1	0
1	0	1	0	1
0	1	0	0	1
1	0	0	0	1
0	1	1	1	0

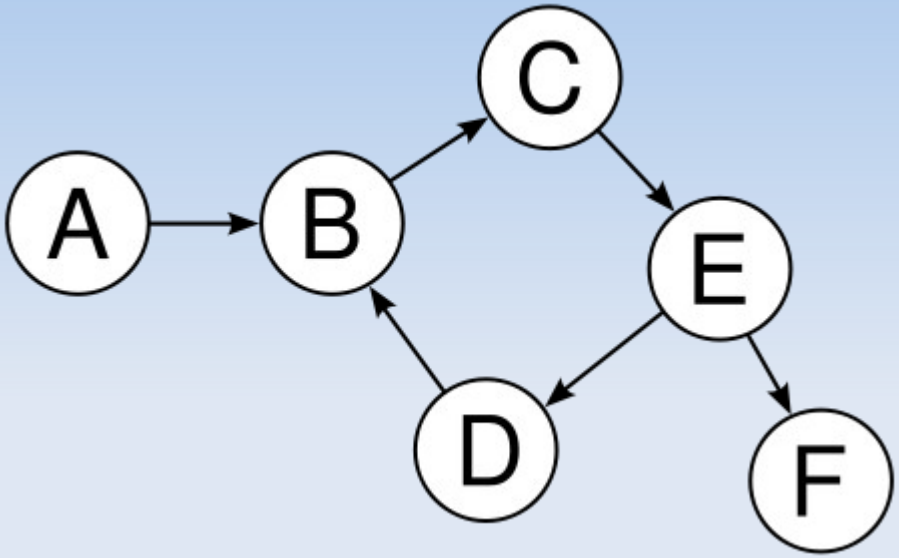
You may input this matrix into to the text area.

Aukera honen arazoak

- Horrelako matrizeak softwarean oso handiak dira:
  - Programa arrunt batek izan ditzake milaka funtsioak.
  - Zenbat memoria behar dugu horrelako matrize bat sortzeko 1.000 funtsio badaude? 976.56 KBak.
  - Eta 100.000 funtsio badaude? 93 Gbak.
    - Ezin...
- Beste gauza bat: funtsio zenbakarritasuna ezberdina bada 2 programetan, ezin dira konparatu. Ez dauka zentzurik.

Auzokidetasun-listak (Adjacency Lists)

- Beste aukera bat: auzokidetasun-listak. Zer dira?



- Eskuineko grafoa auzokidetasun-lista batean erakusten badugu, zera ikusiko dugu...

{A,B}, {B,C}, {B,D},
{C,E}, {E,D}, {E,F}

Auzokidetasun-listak (Adjacency Lists)

- Ikus dezakegunez, orain ditugun datuak konparatzeko txikiagoak dira, ezta?
- Ala eta gustiz ere, nola konparatu ditzakegu 2 lista?
  - Ebakidura-multzo (“Set intersection”) erabil dezakegu.
  - Hau da: begiratu ze elementoak zerrenda horretan ez daude bestean.
  - Ez daudenak, ezberdintasunak dira.
  - Badaudenek, aldiz, antzekotasun ratioa emango dute.

Arazoak...

- Baina aukera honek, noski, arazoak ditu...
  - Nodo bakoitzak lehen jarritako adibidean “izen” bat dauka.
  - 2 programak konparatzen ezin dugu jakin nodoak berdinak diren:
    - A, B, C, etab... izenak jartzen baditugu, izen berak izango dituzte programa bakoitzean baina, ziur asko, funtsio edo oinarri-bloke ezberdinak izango dira.
  - Nola jakin dezakegu funtsio edo oinarri-bloke bat berdina ote den?
  - ...

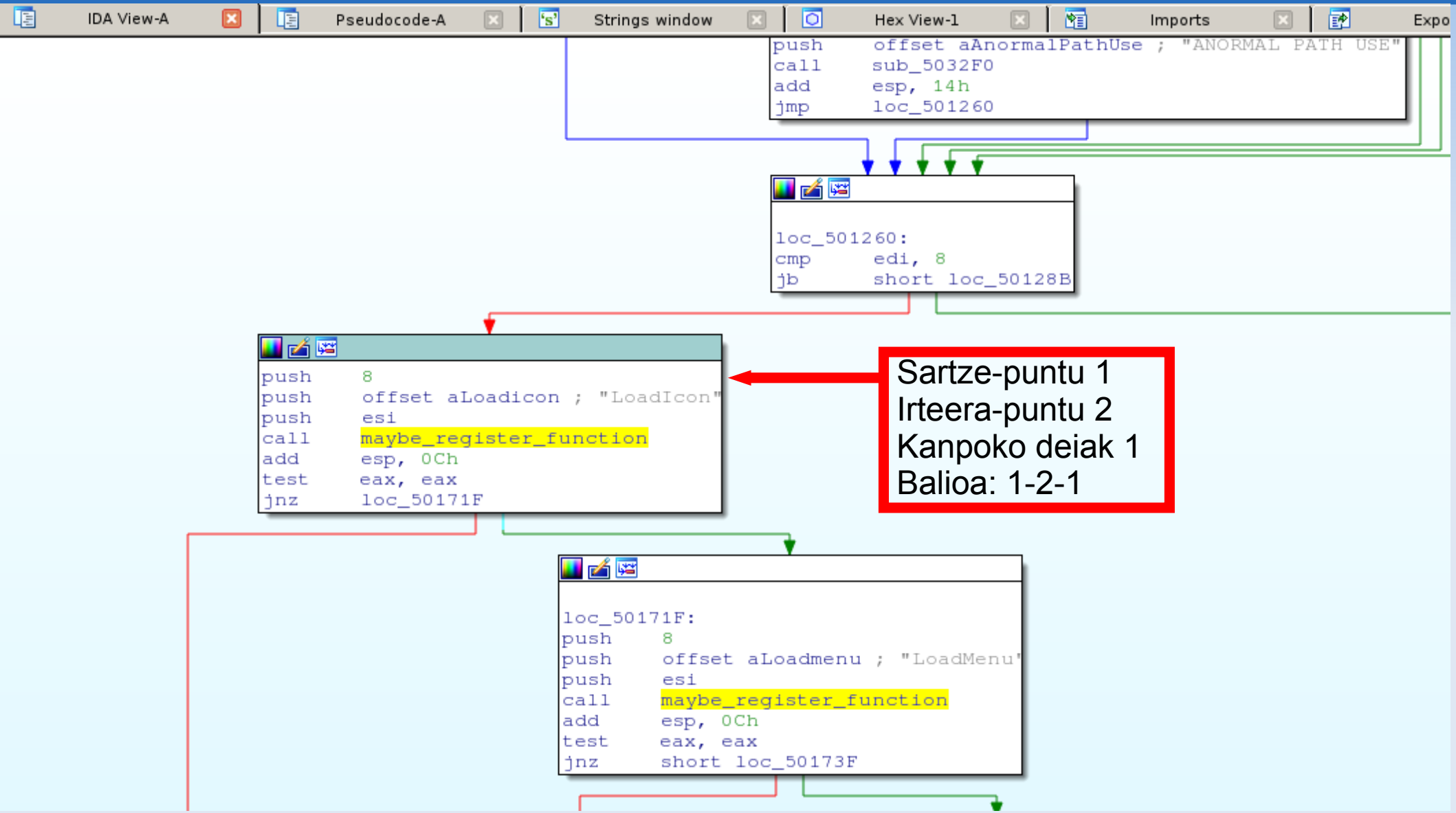
Fingerprinting (batek itzultzen du hau...)

- Zera egin behar dugu: asmatu modu bat “izen” bat (hobe esanda, balio bat) jartzeko nodo bakoitzari.
  - Nodoak izango dira oinarri-blokeak edo funtsioak.
- Nola egin dezakegu hau?
  - ...

Oinarri-blokeak balioak asignatzen

- Oinarri-bloke bati balio bat asignatu dezakegu:
  - Zenbat sartze-puntu dituen zenbatzen.
  - Zenbat irteera-puntu dituen zenbatzen.
  - Zenbat kanpo-deiak dituen zenbatzen.
- Ikus dezagun adibide bat hurrengo irudiarekin...

Oinarri-bloekak balioak asignatzen



Funtsioei asignatzen balioak

- Ados! Badakigu zelan asignatu balioak oinarri-blokeei. Baina, zelan egin daikegu gauza bera funtsioekin?
  - Hasiera batean, gauza bera egiten nuen: sarrera-puntuak, irteera-puntuak eta kanpoko deiak zenbatzen.
    - Aukera ona bat da.
  - Baina esperientziak esaten dit ez dela hain ona.
    - Funtsio **ASKO** metrika bera daukate.
  - Ikus dezakegun beste aukera bat...

Konplexotasun Ziklomatikoak (Cyclomatic Complexity)

- Metrika honek esaten du ze “konplexua” den funtsio bat edo programa bat.
- Normalean, honako funtsio hau erabiltzen da:
  - M = E – N + 2P
- E → Ertzak. N → nodoak. P → inter-konektatutako konponenteak (normalean 2 edo zenbat irteera eta sarrera puntuak dituen funtsioak).
- Balio hau zehatzagoa da.
  - Nahiz eta logika lausoa aplikatuz jarraitzen.

Beste aukera bat...

- Azken aukera erakutsi nahi dudana:
  - Komplexotasun zyklomatikoan oinarrituta, Gödel numbering erabiliz.
  - Gödel numbering erabiliz, balio bakoitzari zenbaki primo bat dagokio.
    - Adibidez, balio 2-ri 2. zenbaki primoa dagokio (3), balio 11-ri 11. zenbaki primoa dagokio (31), etab.
- Eta honek zertarako ox**a balio du?
  - ...

Gödel eta CC

- Badakigu nola sortu programa baten dei-grafoa.
- Badakigu nola kalkulatu funtsio bakoitzaren konplexotasun-ziklomatikoa.
- Badakigu, ere, nola asignatu zenbaki primo bat Gödel metodoa erabiliz.
- Hortaz, ia funtsio bakoitzak, “izen” (balio) ezberdin bat izango du.
- Eta, funtsio guztien balioak biderkatzen baditugu?
  - **Hash lauso bat lortzen dugu non funtsio guztien balioak ditugu eta programa zehatz bati dagokio.**
  - ...

Grafoen oinarritutako hash lausoak

- 2 programen hash lauso hauek berdinak badira, horrek esan nahi du programak, **estrukturalki** berdinak direla, ez antzekoak.
- Ezberdinak badira era, faktorizatzen baditugu programa bakoitzaren funtsio guztien balioak eta talde-intersekzioa aplikatzen badugu, irteera izango da ezberdinak diren funtsioak.
  - Eta atera dezakegu, horrela, ezberdintasun metrika bat.
- ...

Grafoen oinarritutako hash lausoak

- Horrelako hashik nik ez dut asmatu. Hala hori uste dut. Ezin dut jakin.
  - Bere momentuan, antzeko aukerak erabiltzen zituela Zynamics-en jakin dut hango programatzaileekin hitz egiten.
  - Egun ez dakit zer erabiltzen duten.
  - Ez dakit metodo hau erabiltzen duten.
  - Bardin da, egia esan.
- ...

Grafoen oinarritutako hash lausoak

- Algoritmo hau inplementatu egin dut 2 tresnetan:
  - Gcluster. Pyew projektuaren zati bat. Nirea, GPLa.
  - https://code.google.com/p/pyew/source/browse/gcluster.py
  - CAMAL. Komertziala, COSEINC enpresari dagokio (baina neronek idatzi nuen).
  - https://camal.coseinc.com
- Ikus dezagun Gcluster.py-rekin demo bat...

DEMO

```
joxean@backup-server:~/Documentos/research/pyew$ md5sum testing/BypassXtrap.ex_ testing/HGWC.ex_
73be87d0dbcc5ee9863143022ea62f51  testing/BypassXtrap.ex_
e1acaf0572d7430106bd813df6640c2e  testing/HGWC.ex_
joxean@backup-server:~/Documentos/research/pyew$ ssdeep testing/BypassXtrap.ex_ testing/HGWC.ex_
ssdeep,1.1--blocksize:hash:hash,filename
49152:C1vqjdC8rRDMIEQAePhBi70tIZDMIEQAevrv5GZS/ZoE71LGc2eC6JI/Cfnc:C1vqj9fAxYmlfACr5GZAVETeDI/Cvc,"/home
12288:faWzgMg7v3qnCiMErQohh0F4CCJ8lnyC8rm2NY:CaHMv6CorjqnyC8rm2NY,"/home/joxean/Documentos/research/pyew
joxean@backup-server:~/Documentos/research/pyew$ ls -l testing/BypassXtrap.ex_ testing/HGWC.ex_
-rw-r--r-- 1 joxean joxean 2431424 dic  2  2013 testing/BypassXtrap.ex_
-rw-r--r-- 1 joxean joxean  714583 dic  2  2013 testing/HGWC.ex_
joxean@backup-server:~/Documentos/research/pyew$ ./gcluster.py testing/BypassXtrap.ex_ testing/HGWC.ex_
[+] Analyzing file testing/BypassXtrap.ex_
[+] Analyzing file testing/HGWC.ex_
Expert system: Programs are 100% equals
Primes system: Programs are 100% equals
ALists system: Programs are 100% equals
```

Arazoak...

- Aukera guztiek arazoak dituzte, noski.
  - Exekutableekin baino ezin da erabili.
  - Exekutableak enpakatuta badaude, jakina, pakerra multzokatuko dugu.
  - Multzokatzeko benetako kodea desenpaketatu egin behar dugu.
    - Ez da gauza xumea.
  - Ignoratu egin dut nola zatitu programa bat grafoetan...
    - Kode-analisia egin behar da. Berriro: ez da guza xume.
    - ...

Kode-analisia

- Kode-analisia egiteko aukera batzuk ditugu:
  - IDA Pro tresna komertziala erabiliz, Python edo C/C++ plugin edo skript bat sortu ateratzeko informazio hau. Modurik onena, baina garestiena ere.
  - Pyew edo Radare 2 erabiliz. Ez dira IDA Pro bezain indartsu, baina...

Eskerrik asko!
Galderarik bai?