Code similarity analysis with r2 and diaphora

Fernando Domínguez // r2con2021

AT&T Business

About me

```
$ whoami

Fernando Domínguez
Malware analyst / Security researcher @ AT&T Cybersecurity - AlienLabs

@FernandoDoming
```

Objectives

- Port diaphora to radare & MySQL
  - What is diaphora?
  - “Diaphora [...] version 2.0 is the most advanced program diffing tool, working as an IDA plugin, available as of today (2019). It was released first during SyScan 2015 and is actively maintained.”
  - Initial port
    Presentation (r2con 2017)
- Why?
  - Part of a bigger project

The bigger project

- Identify malware based on code similarity
- Mostly focused to IoT malware
  - Most families are open source or derived from them
    Open-source makes signature-based detection tedious
  - Packing is not usual in these platforms

New sample from an unidentified family arrives

Analyst identifies IoCs (functions and strings)

IoCs are persisted to database

Unknown sample arrives

Diaphora DB is generated for new sample

Comparison

IoCs database

Family identification

AT&T Business

Objectives

- Why radare?
  - It’s OSS
  - It’s free (and IDA is quite expensive)
  - Excellent support for all aimed architectures

- Why MySQL?
  - Enables to share the database across team members
  - Better concurrent access
  - More robust DB engine to deploy a service over

Problems

- Existing fork was very old
  - Python 2
  - Diaphora 1
  - SQLite
  - Some features were incomplete
    - No decompiler
    - No Koret-Karamitas hash (CFG hash)
    - No info about switches, mnemonics_spp ...

First steps

- Port code to Python 3
- Port sqlite dialects to MySQL
- Trial and error until the crashes stop
- First “working” version took ~30m to analyze a Gafgyt sample

Optimization

- First approach:
  - Log executed commands and its execution time
  - Observations:
    - Several commands were repeated several times each
  - Solution:
    - Implement a cache in r2pipe (by pancake)

```
r2 = r2pipe.open("/path/to/file")
r2.use_cache = True
```

    - Cache is invalidated by hand (call `r2.invalidate_cache()`)
    - Speedup increase of ~ 6x (depending on use case)
  - Use r2pipe in native mode

```
r2pipe.open("ccall://path/to/bin")
```

    - Speedup increase of ~ 4x
  - Reduced execution time to 1010s with test sample

Optimization

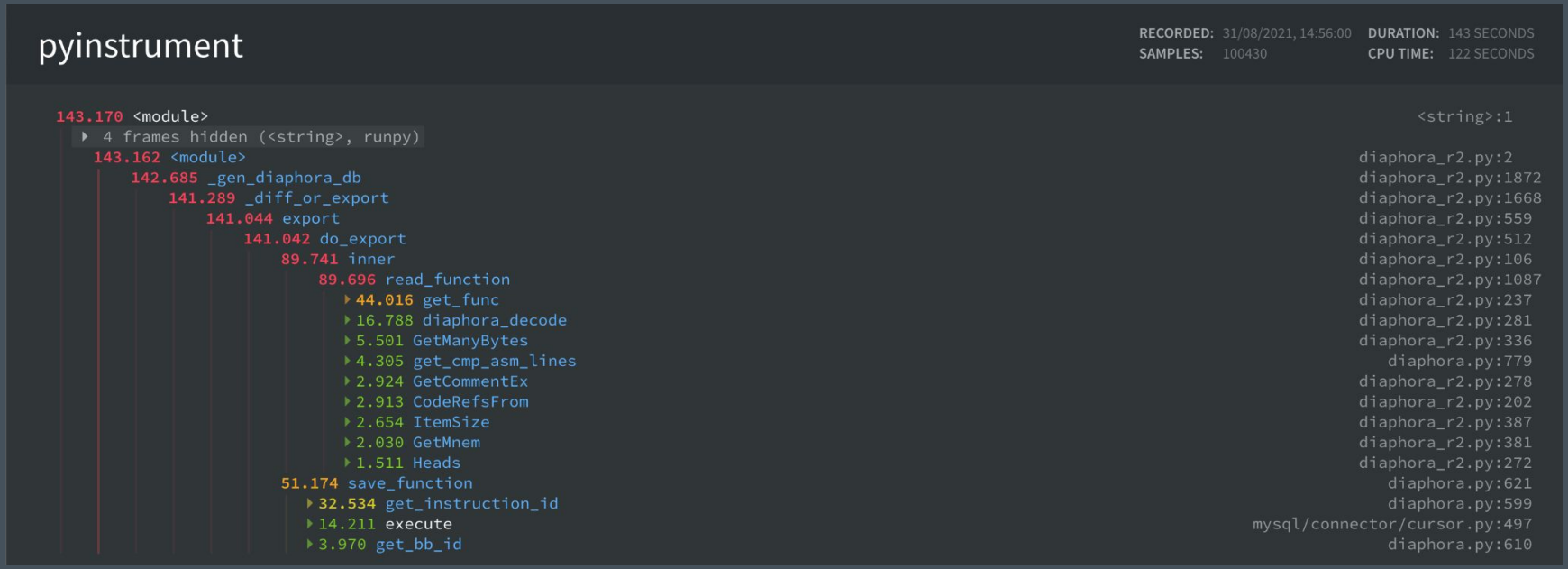
pyinstrument

RECORDED: 31/08/2021, 14:01:40 DURATION: 1,010 SECONDS
SAMPLES: 159440 CPU TIME: 151 SECONDS

```
1,005.022 <module>                                                    <string>:1
  ▸ 4 frames hidden (<string>, runpy)
    1,005.013 <module>                                                diaphora_r2.py:2
      1,004.663 _gen_diaphora_db                                      diaphora_r2.py:1871
        1,003.260 _diff_or_export                                     diaphora_r2.py:1667
          1,003.022 export                                            diaphora_r2.py:558
            1,003.016 do_export                                       diaphora_r2.py:511
              900.705 save_function                                      diaphora.py:620
                ▸ 876.970 get_instruction_id                             diaphora.py:598
                  18.422 execute                                  mysql/connector/cursor.py:497
                    ▸ 5 frames hidden (mysql, <built-in>)
              102.133 inner                                           diaphora_r2.py:105
                102.092 read_function                                 diaphora_r2.py:1086
                  52.035 get_func                                     diaphora_r2.py:236
                    ▸ 52.006 log_exec_r2_cmdj                         diaphora_r2.py:117
                  18.347 diaphora_decode                              diaphora_r2.py:280
                    ▸ 18.225 log_exec_r2_cmdj                         diaphora_r2.py:117
```

Forgot to create indexes for tables

Optimization



`get_func` is the main bottleneck (`afi`). Attempt to substitute `afi` usages where possible

Optimization - r2ghidra

pyinstrument

RECORDED: 04/09/2021, 12:58:44 DURATION: 77.1 SECONDS
SAMPLES: 55714 CPU TIME: 64.7 SECONDS

```
77.051 <module>                                                    <string>:1
  ▸ 4 frames hidden (<string>, runpy)
    77.044 <module>                                                diaphora_r2.py:2
      76.779 _gen_diaphora_db                                      diaphora_r2.py:1865
        75.438 _diff_or_export                                     diaphora_r2.py:1661
          75.248 export                                            diaphora_r2.py:560
            75.246 do_export                                       diaphora_r2.py:513
              46.279 inner                                         diaphora_r2.py:107
                46.240 read_function                               diaphora_r2.py:1084
                  ▸ 17.237 guess_type                              diaphora_r2.py:1038
                  ▸ 7.382 diaphora_decode                          diaphora_r2.py:282
                  ▸ 4.442 GetManyBytes                             diaphora_r2.py:337
                  ▸ 2.581 CodeRefsFrom                             diaphora_r2.py:203
                  ▸ 2.515 get_cmp_asm_lines                           diaphora.py:779
                  ▸ 2.460 GetMnem                                  diaphora_r2.py:382
                  ▸ 2.379 GetCommentEx                             diaphora_r2.py:279
                  ▸ 1.950 ItemSize                                 diaphora_r2.py:388
                  ▸ 1.115 is_constant                              diaphora_r2.py:1075
                    0.933 GetDisasm                                diaphora_r2.py:385
                      0.894 log_exec_r2_cmd                        diaphora_r2.py:125
                    0.877 block_preds                              diaphora_r2.py:150
                      0.850 log_exec_r2_cmdj                       diaphora_r2.py:119
                        0.805 cmdj                                 r2pipe/open_base.py:256
                          ▸ 1 frames hidden (r2pipe)
              28.875 save_function                                 diaphora.py:621
                ▸ 16.849 get_instruction_id                        diaphora.py:599
                ▸ 8.001 execute                                    mysql/connector/cursor.py:497
                ▸ 3.711 get_bb_id                                  diaphora.py:610
        1.340 _r2_open                                             diaphora_r2.py:1875
          1.340 cmd                                                r2pipe/open_base.py:238
            ▸ 4 frames hidden (r2pipe)
```

Optimization - pdc

pyinstrument

RECORDED: 04/09/2021, 14:14:23 DURATION: 60.5 SECONDS
SAMPLES: 54205 CPU TIME: 49 SECONDS

```
60.450 <module>                                                    <string>:1
  ▸ 4 frames hidden (<string>, runpy)
    60.443 <module>                                                diaphora_r2.py:2
      60.236 _gen_diaphora_db                                      diaphora_r2.py:1864
        58.965 _diff_or_export                                     diaphora_r2.py:1660
          58.776 export                                            diaphora_r2.py:559
            58.774 do_export                                       diaphora_r2.py:512
              30.691 inner                                         diaphora_r2.py:106
                30.652 read_function                               diaphora_r2.py:1083
                  ▸ 7.158 diaphora_decode                          diaphora_r2.py:281
                  ▸ 4.403 GetManyBytes                             diaphora_r2.py:336
                  ▸ 2.550 CodeRefsFrom                             diaphora_r2.py:202
                  ▸ 2.464 guess_type                               diaphora_r2.py:1037
                  ▸ 2.400 GetMnem                                  diaphora_r2.py:381
                  ▸ 2.356 get_cmp_asm_lines                           diaphora.py:779
                  ▸ 2.219 GetCommentEx                             diaphora_r2.py:278
                  ▸ 1.927 ItemSize                                 diaphora_r2.py:387
                  ▸ 0.980 is_constant                              diaphora_r2.py:1074
                  ▸ 0.908 GetDisasm                                diaphora_r2.py:384
                  ▸ 0.907 block_preds                              diaphora_r2.py:149
              27.998 save_function                                    diaphora.py:621
                ▸ 16.460 get_instruction_id                           diaphora.py:599
                ▸ 7.497 execute                          mysql/connector/cursor.py:497
                ▸ 3.657 get_bb_id                                     diaphora.py:610
        1.270 _r2_open                                             diaphora_r2.py:1874
          1.270 cmd                                          r2pipe/open_base.py:238
            ▸ 4 frames hidden (r2pipe)
```

Further optimizations

- Use FLIRT signatures to detect library functions and skip them
  - By default all .sig files on `<diaphora_root>/signatures/flirt/` will be used
  - Matches are excluded from analysis
  - Used https://github.com/IridiumXOR/uclibc-sig for uclibc (frequent in IoT malware)

Usage

Step 1 - Install & configure radare / MySQL server
Step 2 - Fill MySQL credentials in <project_root>/db.json
Step 3 - Install requirements (pip install -r requirements.txt)

```
$ ./diaphora_r2.py -h
usage: diaphora_r2.py [-h] [-f] [-o O] file1 [file2]

positional arguments:
  file1       File to analyze
  file2       (Optional) File to diff against

optional arguments:
  -h, --help  show this help message and exit
  -f          Force DB override
  -o O        Diff output file (HTML) - Default value: <db1name[0:10]>_vs_<db2name[0:10]>.html
```

Note: DB name is the SHA256 hash of the sample

Demo 1

Sakura samples (a Gafgyt variant)

AT&T Business

```
$ file *
132948bef56cc5b4d0e435f33e26632264d27ce7d61eba85cf3830fdf7cb8056: ELF 32-bit LSB executable, ARM, EABI4 version 1 (SYSV), statically linked, with debug_info, not
stripped
136dbd3cfa947f286b972af1e389b2a44138c0013aa8060d20c247b6bcfdd88c: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), statically linked, not stripped
17c62e0cf77dc4341809afceb1c8395d67ca75b2a2c020bddf39cca629222161: ELF 32-bit LSB executable, ARM, EABI4 version 1 (SYSV), statically linked, with debug_info, not
stripped
19e0f329b5d8689b14d901b9b65c8d4fb28016360f45b3dfcec17e8340e6411e: ELF 32-bit MSB executable, Motorola m68k, 68020, version 1 (SYSV), statically linked, not stripped
4cc11ffb3681ebced1f9d88e71b70a87e6d4498abca823245c118afead67b6a5: ELF 32-bit LSB executable, MIPS, MIPS-I version 1 (SYSV), statically linked, not stripped
562b4c9a40f9c88ab84ac4ffd0deacd219595ab83ed23a458c5f492594a3a7ef: ELF 32-bit LSB executable, ARM, EABI4 version 1 (SYSV), statically linked, with debug_info, not
stripped
594a6b2c1e9beac3ad5f84458b71c1b7ec05ee0239808c9a63bc901040e413a3: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), statically linked, not stripped
5fec87479a8d2fa7f0ed7c8f6ba76eeea9e86c45123173d2230149a55dcd760d: ELF 32-bit MSB executable, MIPS, MIPS-I version 1 (SYSV), statically linked, not stripped
603d14671f97d12db879cc1c7cd6abfa278bf46431ac73aeb6b3a4c4c2b16b9f: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), statically linked, not stripped
6ce1739788b286cc539a9f24ef8c6488e11f42606189a7aa267742db90f7b18d: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), statically linked, not stripped
770363f9fd334c3f3c4ba0e05a2a0d4701f56a629b09365dfe874b2a277f4416: ELF 32-bit LSB executable, ARM, version 1 (ARM), statically linked, with debug_info, not stripped
7c8ba5f88b1c4689a64652f0b8f5e3922e83f9f73c7e165f3213de27c5fb4d05: ELF 32-bit MSB executable, PowerPC or cisco 4500, version 1 (SYSV), statically linked, not stripped
8090c3a1a930849df42f7f796d42e0211344e709a5ac15c2b4aca8ca41de2cd3: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), statically linked, not stripped
94a279397b8c19ec7def169884a096d4f85ce0e21ff9df0be3ce264ef4565ea7: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), statically linked, not stripped
96bb3e5209e083544ea6a78bc6fc4ebc456e135a786d747718d936af3b063298: ELF 32-bit LSB executable, ARM, EABI4 version 1 (SYSV), statically linked, with debug_info, not
stripped
a079dfd60b55a7d74dd32d49a984bea43665b8b225beceae5b272944889217f6: ELF 32-bit MSB executable, MIPS, MIPS-I version 1 (SYSV), statically linked, not stripped
cef15aa60dc2c09fe117e37e07399f0ef89dca9f930ce13ac1e29f8cf63d9a31: ELF 32-bit MSB executable, Motorola m68k, 68020, version 1 (SYSV), statically linked, not stripped
e984334bbdd1179aadbde949f7c1b0fb02b6c18cb4a56d146150853b18adfa79: ELF 32-bit LSB executable, MIPS, MIPS-I version 1 (SYSV), statically linked, not stripped
```

Results (same arch. & variant)

```
File 1: [REDACTED]/samples/sakura/562b4c9a40f9c88ab84ac4ffd0deacd219595ab83ed23a458c5f492594a3a7ef
Type: ELF 32-bit LSB executable, ARM, EABI4 version 1 (SYSV), statically linked, with debug_info, not stripped
SHA256: 562b4c9a40f9c88ab84ac4ffd0deacd219595ab83ed23a458c5f492594a3a7ef
```

```
File 2: [REDACTED]/samples/sakura/770363f9fd334c3f3c4ba0e05a2a0d4701f56a629b09365dfe874b2a277f4416
Type: ELF 32-bit LSB executable, ARM, version 1 (ARM), statically linked, with debug_info, not stripped
SHA256: 770363f9fd334c3f3c4ba0e05a2a0d4701f56a629b09365dfe874b2a277f4416
```

Found 40 matches across compared files

Type	Name	Address	BB1	Name 2	Address 2	BB2	Ratio	Description
best	sym._call_via_r6	0x000080e8	1	sym.call___do_global_dtors_aux	0x00008128	1	1.000	Same cleaned pseudo-code
best	sym._call_via_r5	0x000080e4	1	sym.call_frame_dummy	0x00008188	1	1.000	Same cleaned pseudo-code
best	sym._call_via_r3	0x000080dc	1	sym.call___do_global_ctors_aux	0x0001686c	1	1.000	Same cleaned pseudo-code
best	sym.prints	0x000089a0	20	sym.prints	0x00008988	20	1.000	Same cleaned pseudo-code
partial	sym.getArch	0x0000c8d8	1	sym.getArch	0x0000ca4c	1	0.770	Same constants
partial	sym.init_rand	0x000081ec	4	sym.init_rand	0x000081cc	4	0.854	Same constants
partial	sym.getOurIP	0x00008540	21	sym.getOurIP	0x00008508	21	0.900	Same constants
partial	sym.getPortz	0x0000c8fc	10	sym.getPortz	0x0000ca68	10	0.912	Same constants
partial	sym.SendSTDHEX	0x0000a82c	6	sym.SendSTDHEX	0x0000a83c	6	0.904	Same MD Index and constants
partial	sym.SendSTD	0x0000b248	6	sym.SendSTD	0x0000b328	6	0.904	Same MD Index and constants
partial	sym.stdhexflood	0x0000b388	6	sym.stdhexflood	0x0000b47c	6	0.890	Same MD Index and constants
partial	sym.vseattack	0x0000abd4	41	sym.vseattack	0x0000ac2c	41	0.818	Same MD Index and constants
partial	sym.SendSTD_HEX	0x0000b4d8	6	sym.SendSTD_HEX	0x0000b5e0	6	0.902	Same KOKA hash and constants
partial	sym.printi	0x00008b28	18	sym.printi	0x00008b0c	18	0.876	Same rare MD Index
partial	sym.processCmd	0x0000c9d0	205	sym.processCmd	0x0000cb34	205	0.888	Same MD Index and constants
partial	sym.getRandomIP	0x00008408	1	sym.getRandomIP	0x000083d8	1	0.800	Similar pseudo-code and names
partial	sym.print	0x00008d14	33	sym.print	0x00008cf4	33	0.848	Same rare KOKA hash
partial	sym.initConnection	0x0000e114	11	sym.initConnection	0x0000e274	11	0.910	Same rare KOKA hash
partial	entry.fini0	0x0000810c	3	sym.__do_global_ctors_aux	0x00016838	3	0.560	Mnemonics small-primes-product
partial	sym.recvLine	0x000092f4	19	sym.recvLine	0x00009298	19	0.948	Same rare KOKA hash

Results (x86 vs ARM / same variant)

```
File 1: [REDACTED]/samples/sakura/136dbd3cfa947f286b972af1e389b2a44138c0013aa8060d20c247b6bcfdd88c
Type: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), statically linked, not stripped
SHA256: 136dbd3cfa947f286b972af1e389b2a44138c0013aa8060d20c247b6bcfdd88c
```

```
File 2: [REDACTED]/samples/sakura/17c62e0cf77dc4341809afceb1c8395d67ca75b2a2c020bddf39cca629222161
Type: ELF 32-bit LSB executable, ARM, EABI4 version 1 (SYSV), statically linked, with debug_info, not stripped
SHA256: 17c62e0cf77dc4341809afceb1c8395d67ca75b2a2c020bddf39cca629222161
```

Found 18 matches across compared files

Type	Name	Address	BB1	Name 2	Address 2	BB2	Ratio	Description
partial	sym.trim	0x08048579	11	sym.trim	0x000087fc	11	0.762	Same rare MD Index
partial	sym.initConnection	0x0804ce3b	11	sym.initConnection	0x0000e114	11	0.762	Same rare MD Index
partial	sym.printi	0x08048740	18	sym.printi	0x00008b28	18	0.766	Same rare MD Index
partial	sym.prints	0x08048666	20	sym.prints	0x000089a0	20	0.758	Same rare MD Index
partial	sym.getOurIP	0x08048356	21	sym.getOurIP	0x00008540	21	0.738	Same rare MD Index
partial	sym.rtcp	0x0804a756	21	sym.rtcp	0x0000b63c	21	0.726	Same rare MD Index
partial	sym.SendSTD	0x0804a3d1	6	sym.SendSTD	0x0000b248	6	0.748	Same MD Index and constants
partial	sym.SendUDP	0x08049362	24	sym.SendUDP	0x00009d10	24	0.736	Same rare MD Index
partial	sym.stdhexflood	0x0804a4f3	6	sym.stdhexflood	0x0000b388	6	0.744	Same MD Index and constants
partial	sym.print	0x0804887f	33	sym.print	0x00008d14	33	0.760	Same rare MD Index
partial	sym.atcp	0x0804af39	36	sym.atcp	0x0000c0c8	36	0.730	Same rare MD Index
partial	sym.vseattack	0x08049efa	41	sym.vseattack	0x0000abd4	41	0.740	Same rare constant
partial	sym.SendSTDHEX	0x08049c33	6	sym.SendSTDHEX	0x0000a82c	6	0.742	Same rare constant
partial	sym.getPortz	0x0804b5ec	10	sym.getPortz	0x0000c8fc	10	0.822	Same rare constant
partial	sym.ftcp	0x080496b4	36	sym.ftcp	0x0000a16c	36	0.732	Same rare MD Index
partial	sym.audp	0x0804aade	37	sym.audp	0x0000bb34	37	0.744	Same rare MD Index
partial	main	0x0804cf89	54	main	0x0000e2a4	54	0.728	Same rare MD Index
partial	sym.processCmd	0x0804b686	205	sym.processCmd	0x0000c9d0	205	0.772	Same rare constant

Results (diff. archs & variants)

```
File 1: /Users/fdd/dev/identikit/samples/sakura/594a6b2c1e9beac3ad5f84458b71c1b7ec05ee0239808c9a63bc901040e413a3
Type: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), statically linked, not stripped
SHA256: 594a6b2c1e9beac3ad5f84458b71c1b7ec05ee0239808c9a63bc901040e413a3
```

```
File 2: /Users/fdd/dev/identikit/samples/yakuza/91392f5dbbfd4ad142956983208a484b91ac5e84c4f9a9fcb530a9b085644c93
Type: ELF 32-bit LSB executable, ARM, version 1 (ARM), statically linked, with debug_info, not stripped
SHA256: 91392f5dbbfd4ad142956983208a484b91ac5e84c4f9a9fcb530a9b085644c93
```

Found 11 matches across compared files

Type	Name	Address	BB1	Name 2	Address 2	BB2	Ratio	Description
partial	sym.initConnection	0x00405444	11	sym.initConnection	0x0000df08	11	0.752	Same rare MD Index
partial	sym.SendSTD	0x004027ae	6	sym.SendSTD	0x0000c254	6	0.736	Same MD Index and constants
partial	sym.trim	0x00400583	11	sym.trim	0x00008838	11	0.750	Same rare MD Index
partial	sym.connectTimeout	0x0040118f	14	sym.connectTimeout	0x000097f8	14	0.734	Same rare MD Index
partial	sym.printi	0x0040077a	18	sym.printi	0x00008b58	18	0.760	Same rare MD Index
partial	sym.recvLine	0x00400f55	19	sym.recvLine	0x00009524	19	0.732	Same rare MD Index
partial	sym.prints	0x0040069b	20	sym.prints	0x000089d4	20	0.750	Same rare MD Index
partial	sym.SendUDP	0x0040165f	24	sym.SendUDP	0x0000c3a8	24	0.714	Same rare MD Index
partial	sym.atcp	0x00403315	36	sym.SendTCP	0x0000c874	36	0.728	Same rare MD Index
partial	sym.init_rand	0x004001c0	4	sym.makeRandomStr	0x0000949c	4	0.770	Call address sequence
unreliable	sym.processCmd	0x00403ac4	203	sym.processCmd	0x0000d260	113	0.380	Same rare constant

Demo 2

Uptycs claims Mirai code is found in Gafgyt samples

https://www.uptycs.com/blog/mirai-code-re-use-in-gafgyt

1b3bb39a3d1eea8923ceb86528c8c38ecf9398da1bdf8b154e6b4d0d8798be49

Demo 2

```
v15 = gethostbyname(a1, 2LL);
v11 = 0LL;
v12 = 0LL;
bcopy(**(_QWORD **)(v15 + 24), (char *)&v11 + 4, *(signed int *)(v15 + 20));
v11 = (unsigned __int16)*(_DWORD *)(v15 + 16);
WORD1(v11) = a2;
for ( i = 0; ; ++i )
{
  v4 = "/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x9"
       "3/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x"
       "38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/"
       "xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38"
       "/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xI"
       "D/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/x"
       "FJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/"
       "x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ"
       "/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9A/x38/xFJ/x93/xID/x9"
       "A/x38/xFJ/x93/xID/x9A";
  v5 = "/x53/x65/x6c/x66/x20/x52/x65/x70/x20/x46/x75/x63/x6b/x69/x6e/x67/x20/x4e/x65/x54/x69/x53/x20/x61/x6e/x64/x20/x5"
       "4/x68/x69/x73/x69/x74/x79/x20/x30/x6e/x20/x55/x72/x20x46/x75/x43/x6b/x49/x6e/x47/x20/x46/x6f/x52/x65/x48/x65/x4"
       "1/x64/x20/x57/x65/x20/x42/x69/x47/x20/x4c/x33/x33/x54/x20/x48/x61/x78/x45/x72/x53/x0a";
  v6 =
  v7 =
  v8 =
  v9 =
  v10 =
  if (
  {
    se
    co
```

Gafgyt

```
void sendSTD(unsigned char *ip, int port, int secs) {
int iSTD_Sock;
iSTD_Sock = socket(AF_INET, SOCK_DGRAM, 0);
time_t start = time(NULL);
struct sockaddr_in sin;
struct hostent *hp;
hp = gethostbyname(ip);
bzero((char*) &sin,sizeof(sin));
bcopy(hp->h_addr, (char *) &sin.sin_addr, hp->h_length);
sin.sin_family = hp->h_addrtype;
sin.sin_port = port;
unsigned int a = 0;
while(1){
char *randstrings[] = {"VSzNC0CJti3ouku", "yhJyMAqx7DZa0kq
"qHjTXcMbzH", "chickennuggets", "ilovecocaine", "666666",
"password", "encrypted", "suckmydick", "guardiacivil", "2
"lizardsquad", "aNriBnTRi", "1OD8vpG86", "IVkLWYiLe", "nex
```

Mirai

File Edit Jump Search View Debugger Options Windows Help

Pseudocode-A

```
150       v17 = v85;
151       v18 = "all";
152       v19 = 4LL;
153       do
154       {
155         if ( !v19 )
156           break;
157         v15 = *v17 < (const unsigned __int8)*v18;
158         v16 = *v17++ == *v18++;
159         --v19;
160       }
161       while ( v16 );
162       if ( (!v15 && !v16) == v15 )
163       {
164         *(_BYTE *)(v99 + 13) |= 2u;
165         *(_BYTE *)(v99 + 13) |= 4u;
166         *(_BYTE *)(v99 + 13) |= 1u;
167         *(_BYTE *)(v99 + 13) |= 0x10u;
168         *(_BYTE *)(v99 + 13) |= 8u;
169       }
170       else
171       {
172         v18 = v85;
173         for ( i = (_BYTE *)strtok(v85, ","); ; i = (_BYTE *)strto
174         {
175           v23 = 0;
176           v24 = i == 0LL;
177           if ( !i )
178             break;
179           v74 = i;
180           v73 = "syn";
181           v72 = 4LL;
182           v20 = i;
183           v21 = "syn";
184           v22 = 4LL;
185           do
186           {
```

00001D0C atcp:150 (401D0C)

Gafgyt

```
void SendTCP(unsigned char *target, int port, int timeEnd, unsigned char *flags, int packetsize, int
pollinterval, int spoofit) {
        register unsigned int pollRegister;
        pollRegister = pollinterval;
        struct sockaddr_in dest_addr;
        dest_addr.sin_family = AF_INET;
        if(port == 0) dest_addr.sin_port = rand_cmwc();
        else dest_addr.sin_port = htons(port);
        if(getHost(target, &dest_addr.sin_addr)) return;
        memset(dest_addr.sin_zero, '\0', sizeof dest_addr.sin_zero);
        int sockfd = socket(AF_INET, SOCK_RAW, IPPROTO_TCP);
        if(!sockfd) { return; }
        int tmp = 1;
        if(setsockopt(sockfd, IPPROTO_IP, IP_HDRINCL, &tmp, sizeof (tmp)) < 0) { return; }
        in_addr_t netmask;
        if ( spoofit == 0 ) netmask = ( ~((in_addr_t) -1) );
        else netmask = ( ~((1 << (32 - spoofit)) - 1) );
        unsigned char packet[sizeof(struct iphdr) + sizeof(struct tcphdr) + packetsize];
        struct iphdr *iph = (struct iphdr *)packet;
        struct tcphdr *tcph = (void *)iph + sizeof(struct iphdr);
        makeIPPacket(iph, dest_addr.sin_addr.s_addr, htonl( getRandomIP(netmask) ), IPPROTO_TCP, sizeof(st
tcphdr) + packetsize);
        tcph->source = rand_cmwc();
        tcph->seq = rand_cmwc();
        tcph->ack_seq = 0;
        tcph->doff = 5;
        if(!strcmp(flags, "all")) {
                tcph->syn = 1;
                tcph->rst = 1;
                tcph->fin = 1;
                tcph->ack = 1;
                tcph->psh = 1;
        } else {
                unsigned char *pch = strtok(flags, ",");
                while(pch) {
                        if(!strcmp(pch,          "syn")) { tcph->syn = 1;
```

Mirai

Procedure

- 1 - Download & compile source code for Mirai and Gafgyt
- 2 - Add “most prominent” functions in both families as IoCs
- 3 - Run r2diaphora comparing blog sample with IoC DB

Demo 2

```
[*] 9 matches for family  'Gafgyt'
      [...]
      address: 00404227
      name:  sym.sendTLS
      ratio: 0.734
      bb1: 28
      bb2: 28
      description: Same rare MD Index
      type: partial
      matched_ioc_name:  sym.sendJUNK
      ----------
      address: 004037c5
      name:  sym.sendHOLD
      ratio: 0.732
      bb1: 30
      bb2: 30
      description: Same rare MD Index
      type: partial
      matched_ioc_name:  sym.sendHOLD
      ----------
      address: 00401fd1
      name:  sym.sendSTD
      ratio: 0.620
      bb1: 10
      bb2: 9
      description: Same rare constant
      type: partial
      matched_ioc_name:  sym.SendSTDHEX
      ----------
      [...]
```

```
[*] 1 matches for family  'Mirai'
      address: 0040b240
      name:  sym.killer_kill_by_port
      ratio: 0.500
      bb1: 55
      bb2: 55
      description: Same rare constant
      type: partial
      matched_ioc_name:  sym.killer_kill_by_port
```

If you search for the SendHTTP function prototype on Google -> https://ideone.com/fork/ADE7RJ

```
////////////////////////////////////
// Orignal qBot Modified by xFyfa.//
//                v4              //
// Thanks to ZoneHax, Cheats, Narc//
// This bot is not to be shared   //
////////////////////////////////////
```

Future plans

- Keep optimizing performance
- Implement pseudocode primes hash
- Improve quality of matches
- Create a pip package

Links

Code: https://github.com/FernandoDoming/diaphora

We are hiring!

```
if questions or feedback:
    answer()
else:
    farewell()
```