Confidence in a connected world.

Symantec Research Labs

Automatic Generation of String Signatures for Malware Detection

Scott Schneider, Kent Griffin – SRL
Xin Hu – University of Michigan, Ann Arbor
Tzi-cker Chiueh – Stonybrook University

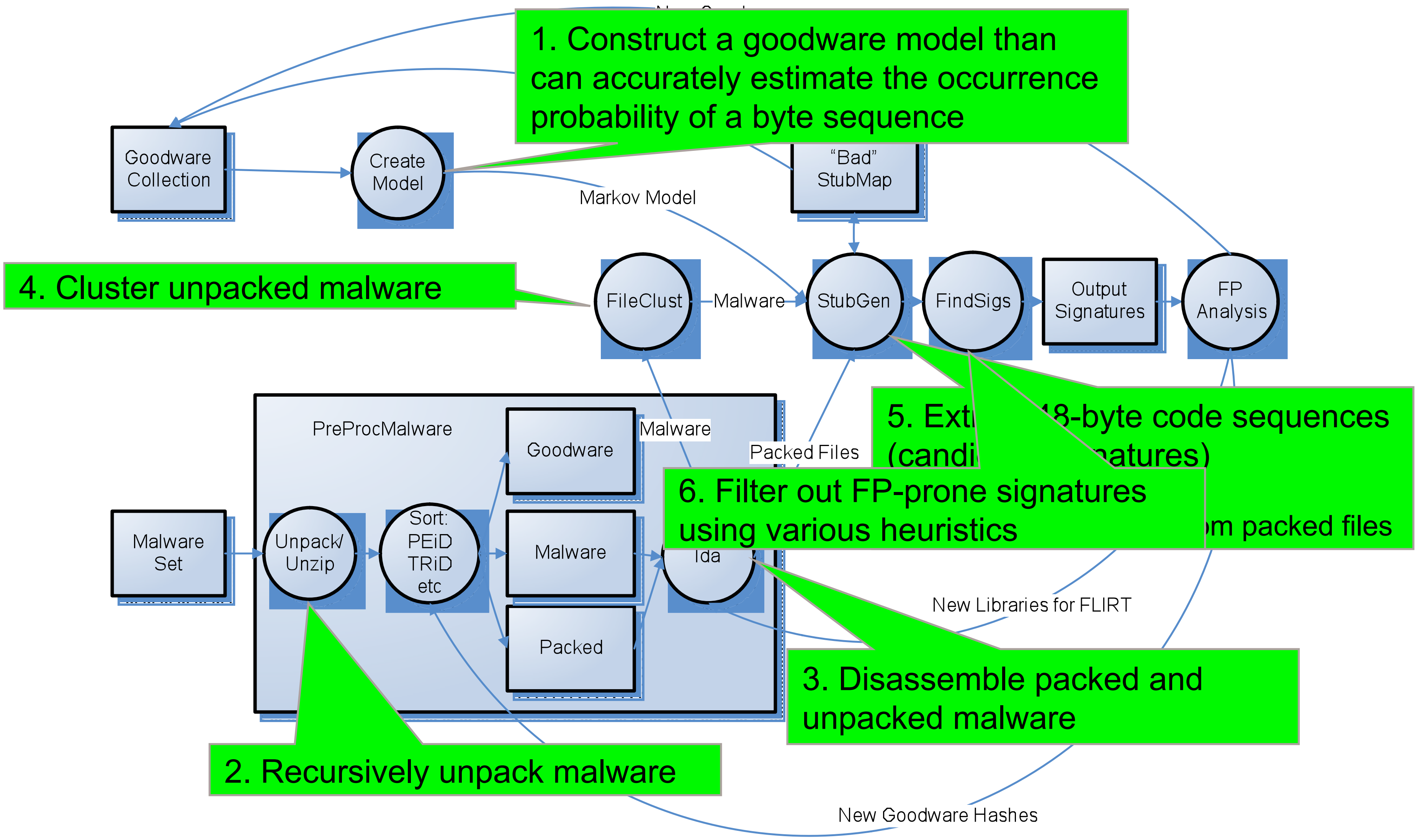
September 24, 2009

String Signature Generation

symantec.

- Goal: Given a set of malware samples, derive a minimal set of string signatures that can cover as many malware samples as possible while keeping the FP rate close to zero
  - 48-byte sequences from code
- Why string signatures?
  - Still one of the main techniques for Symantec and other AV companies
  - Higher coverage than file hashes → smaller signature set
  - Currently created manually!

System Overview



1. Construct a goodware model than can accurately estimate the occurrence probability of a byte sequence

2. Recursively unpack malware

3. Disassemble packed and unpacked malware

4. Cluster unpacked malware

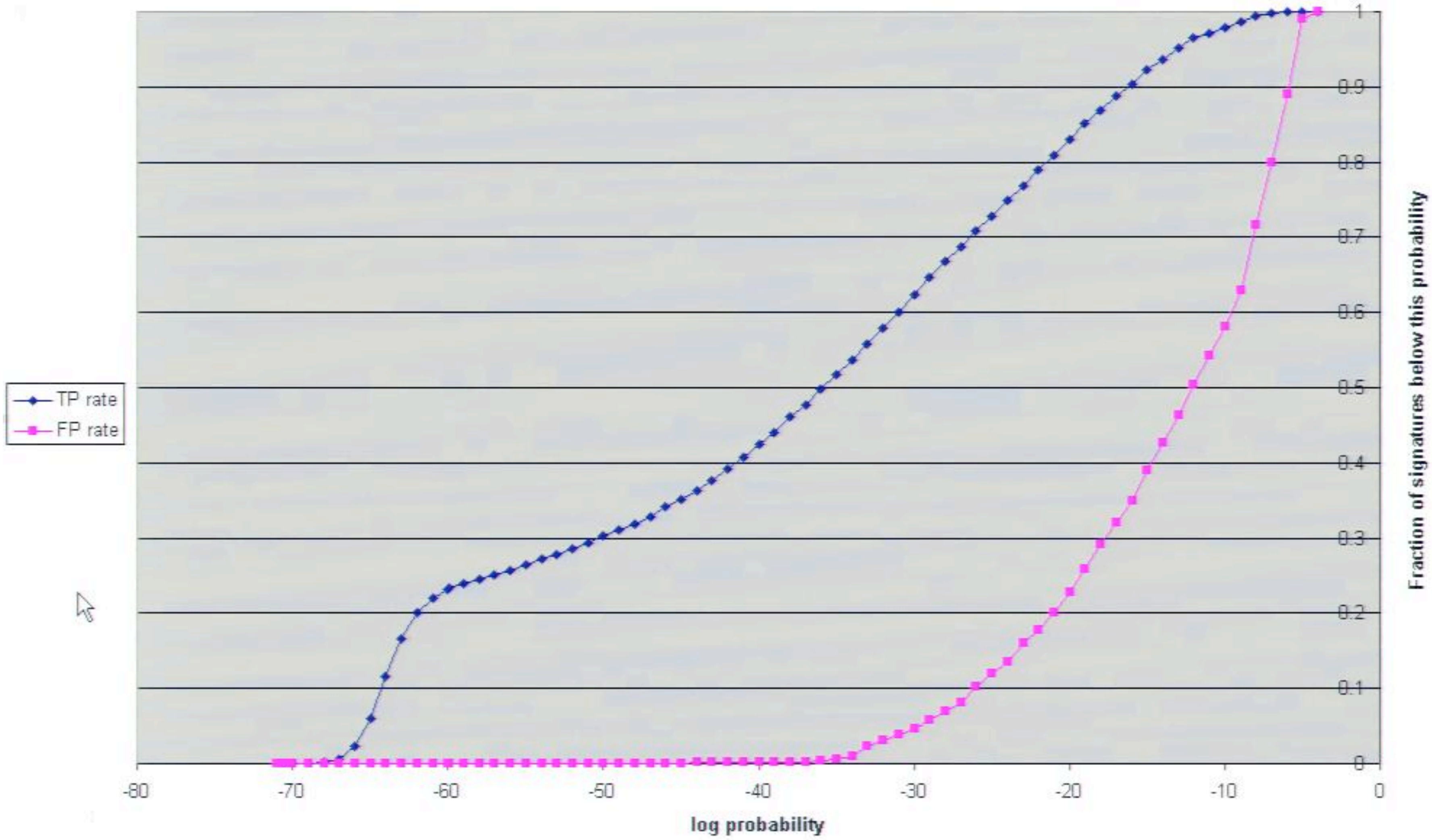
5. Ext 8-byte code sequences (candi natures) m packed files

6. Filter out FP-prone signatures using various heuristics

Heuristics

- 3 main categories:
  - Probability-based – using a Markov chain model
  - Diversity-based – identifies rare libraries and other reused code
  - Disassembly-based – examines assembly instructions
- Discrimination power
  - The best heuristics have high FP reduction and low coverage reduction
  - log (FP_i / FP_f) / log $(Coverage_i / Coverage_f)$
  - Raw vs marginal discrimination power

Goodware Model Effectiveness



Modeling

- Fixed 5-gram Markov chain model
  - Fixed because the rarest byte sequences are the most important
    - LZ-based training backfired
    - Variable-order models use much more memory
- Needed ~100 MB of relevant data to work
- Probability calculated as in Prediction by Partial Matching
  - $p(c|ab) = [c(abc) / c(ab)] * (1-\varepsilon(c(ab))) + p(c|b) * \varepsilon(c(ab))$
  - $\varepsilon(c) = \mathrm{sqrt}(32) / (\mathrm{sqrt}(32) + \mathrm{sqrt}(c))$

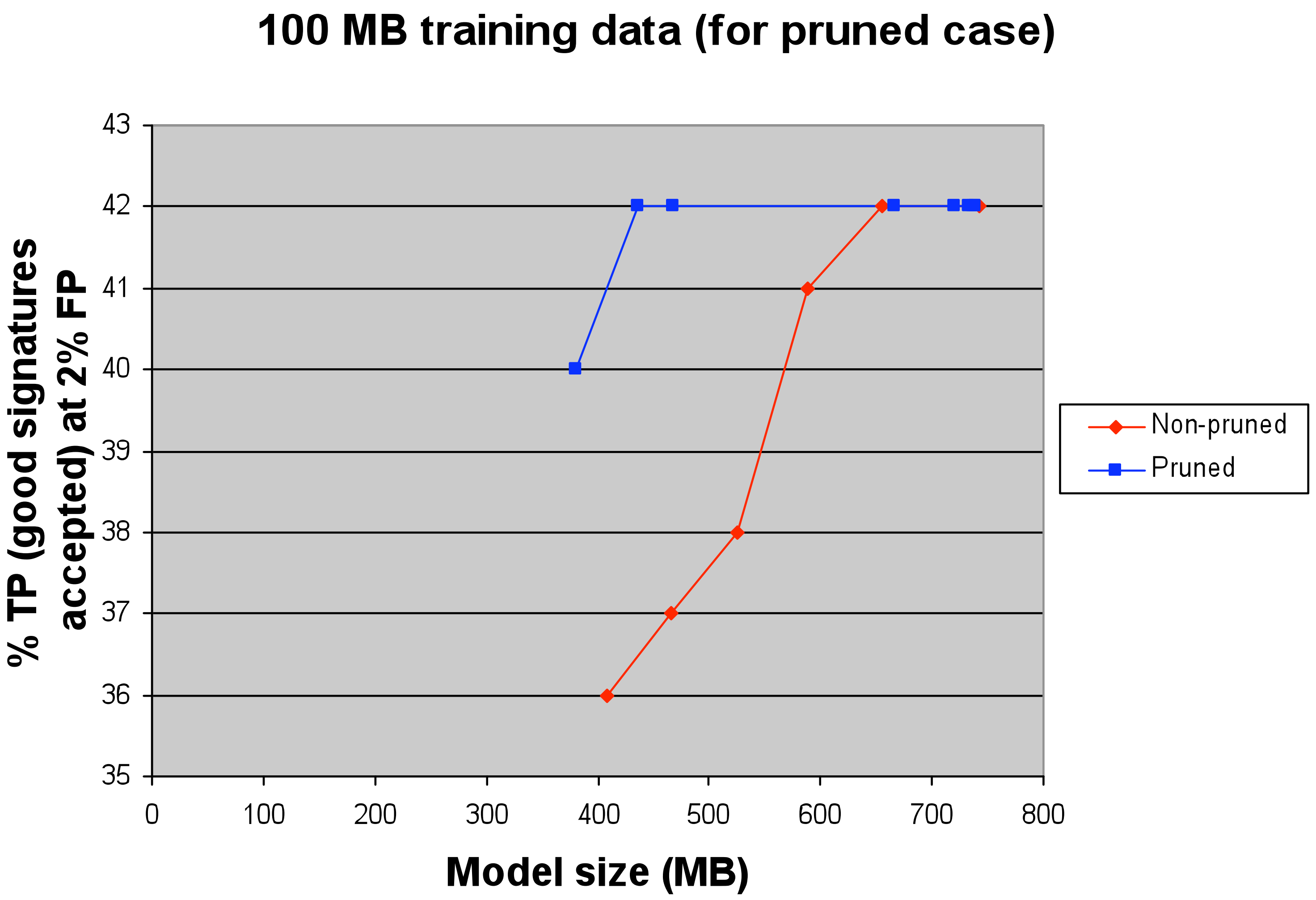
Scaling the Model

- We have TBytes of training data
  - A model trained on this would use too much memory
  - Solution: create several models, then prune and merge them

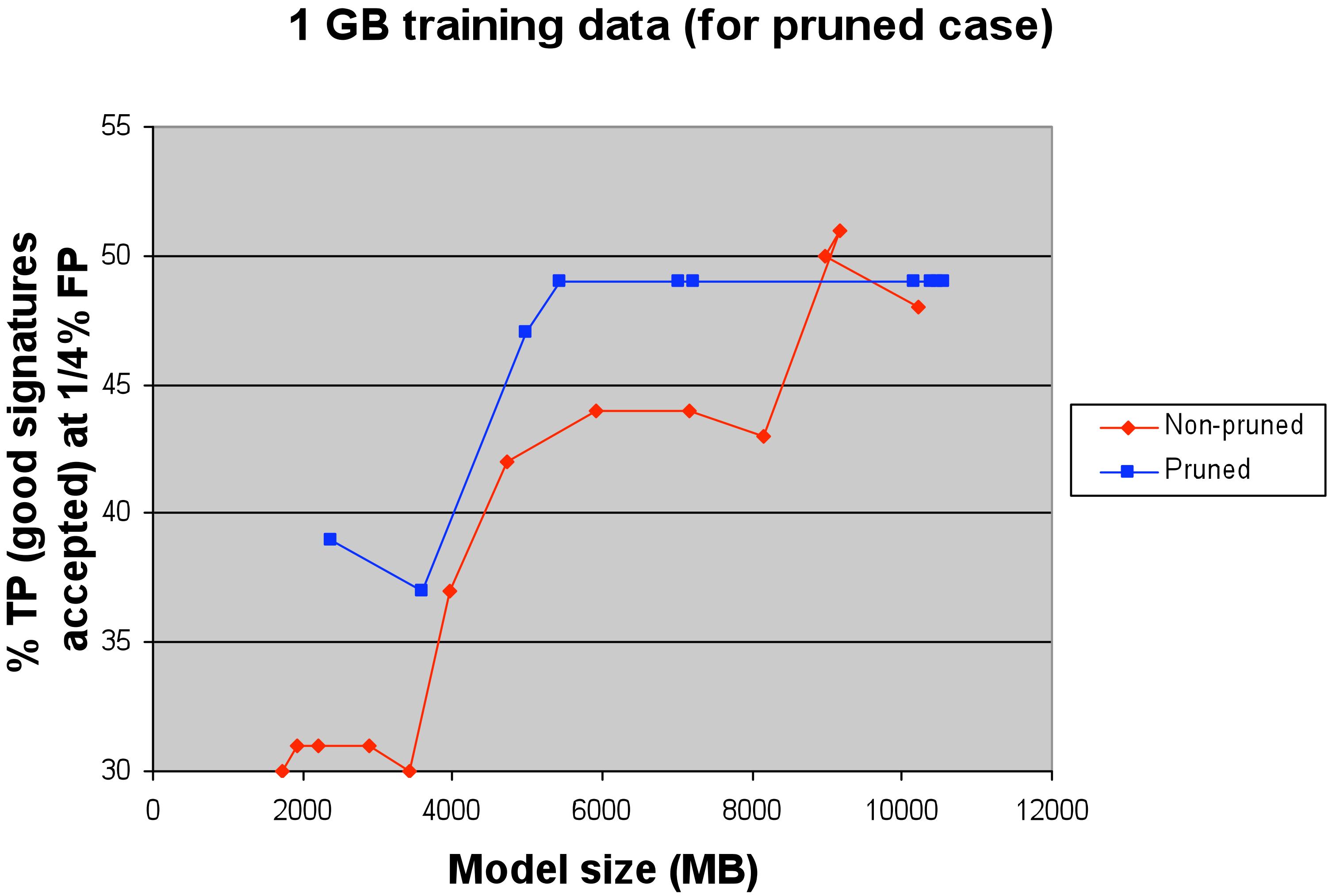
- Pruning
  - If p(c|ab) is close to p(c|b), we don't need node abc
  - If |log(p(c|ab)) – log(p(c|b))| < log(threshold), remove abc
    - Thresholds up to 200 preserve most of the model's effectiveness

Pruned Model Results

symantec.



Pruned Model Results Continued



Diversity-based Heuristics

- High coverage signatures are more likely to be from rare library code
  - Model-only tests had 25-30% FPs
- So we examine the *diversity* of covered malware files
  - If files are from many malware families, it's probably a library

Byte-level Diversity-based Heuristics

- Group count/ratio
  - Cluster malware into families
  - Reject signatures that cover too many groups or have too high a ratio of groups to covered files
- Signature position deviation
  - How much does the signature’s position in the files vary?
- Multiple common signatures
  - Find a 2nd signature a fixed distance (≥1kb) away in all covered files

Instruction-level Diversity-based Heuristics

- Enclosing function count
  - Different enclosing functions indicates code reuse
- Several ways of comparing enclosing functions:
  - Exact byte sequences
  - Instruction op codes with some canonicalization
    - e.g. All ADD instructions are treated the same
  - Instruction sequence de-obfuscation
    - e.g. “test esi, esi” and “or esi, esi” is the same

Method	% FP sig.s Remaining	% all sig.s Remaining	Discrimination Power
Exact byte sequences	17%	54%	2.9
Op code canonicalization	78%	90.5%	2.5
Instruction de-obfuscation	89%	94.7%	2.1

Disassembly-based Heuristics

- IDA Pro’s FLIRT –
  Fast Library Identification and Recognition Technology
  - Universal FLIRT
  - Library function reference heuristic
  - Address space heuristic
- Code interestingness…

Code Interestingness Heuristic

- Encodes Symantec analysts’ intuitions using fuzzy logic
- Targets code that is suspicious and/or unlikely to FP
- Points for
  - Unusual constant values
  - Unusual address offsets
    - May indicate custom structs/classes
  - Local, non-library function calls
  - Math instructions
    - Often done by malware for obfuscation

Results

Thresholds	Coverage	# sigs	# FPs	# Good sigs	# So-so sigs	# Bad sigs
Loose	15.7%	23	0	6	7	1
Normal	14.0%	18	0	6	2	0
Strict	11.7%	11	0	6	0	0
All non-FP	22.6%	220	0	10	11	9

- Used samples for August 2008
  - 2,363 unpacked files

Threshold settings	Prob.	Group ratio	Pos. dev.	# common sig.s	Interesting score	Min. coverage
Loose	-90	0.35	4000	Single	13	3
Normal	-90	0.35	3000	Single	14	4
Strict	-90	0.35	3000	Dual	17	4

Results

- 2007-8 files
  - 46,988 unpacked files

Thresholds	Coverage	# sigs	# FPs
Loose	14.1%	1650	7
Normal	11.7%	767	2
Normal + pos. dev. 1,000	11.3%	715	0
Strict	4.4%	206	0
All non-FP	31.8%	7305	0

Raw Discrimination Power

Heuristic	% FPs Remaining	% Coverage	Discrimination Power
Position deviation (from ∞ to 8,000)	41.7%	96.6%	25
Min File Coverage (from 3 to 4)	6.0%	83.3%	15
Group Ratio (from 1.0 to .6)	2.4%	74.0%	12
*Probability (from -80 to -100)	51.2%	73.7%	2.2
*Interestingness (from 13 to 15)	58.3%	78.2%	2.2
Multiple common sig.s (from 1 to 2)	91.7%	70.2%	0.2
*Universal FLIRT	33.1%	71.7%	3.3
*Library function reference	46.4%	75.7%	2.8
*Address space	30.4%	70.8%	3.5

*Not entirely raw

Marginal Discrimination Power

Heuristic	# FPs	% Coverage
Position deviation (from 3,000 to ∞)	10	121%
Min File Coverage (from 4 to 3)	2	126%
Group Ratio (from 0.35 to 1)	16	162%
Probability (from -90 to -80)	1	123%
Interestingness (from 17 to 13)	2	226%
Multiple common sig.s (from 2 to 1)	0	189%
Universal FLIRT	3	106%
Library function reference	4	108%
Address space	3	109%

Multi-component Signatures

# Components	# Allowed FPs	Coverage	# Signatures	# FPs
2	1	28.9%	76	7
2	0	23.3%	52	2
3	1	26.9%	62	1
3	0	24.2%	44	0
4	1	26.2%	54	0
4	0	18.1%	43	0
5	1	26.2%	54	0
5	0	17.9%	43	0
6	1	25.9%	51	0
6	0	17.6%	41	0

- 16 bytes per component, from code and data
- Tested against a smaller goodware set

Thank You!

Tzi-cker Chiueh

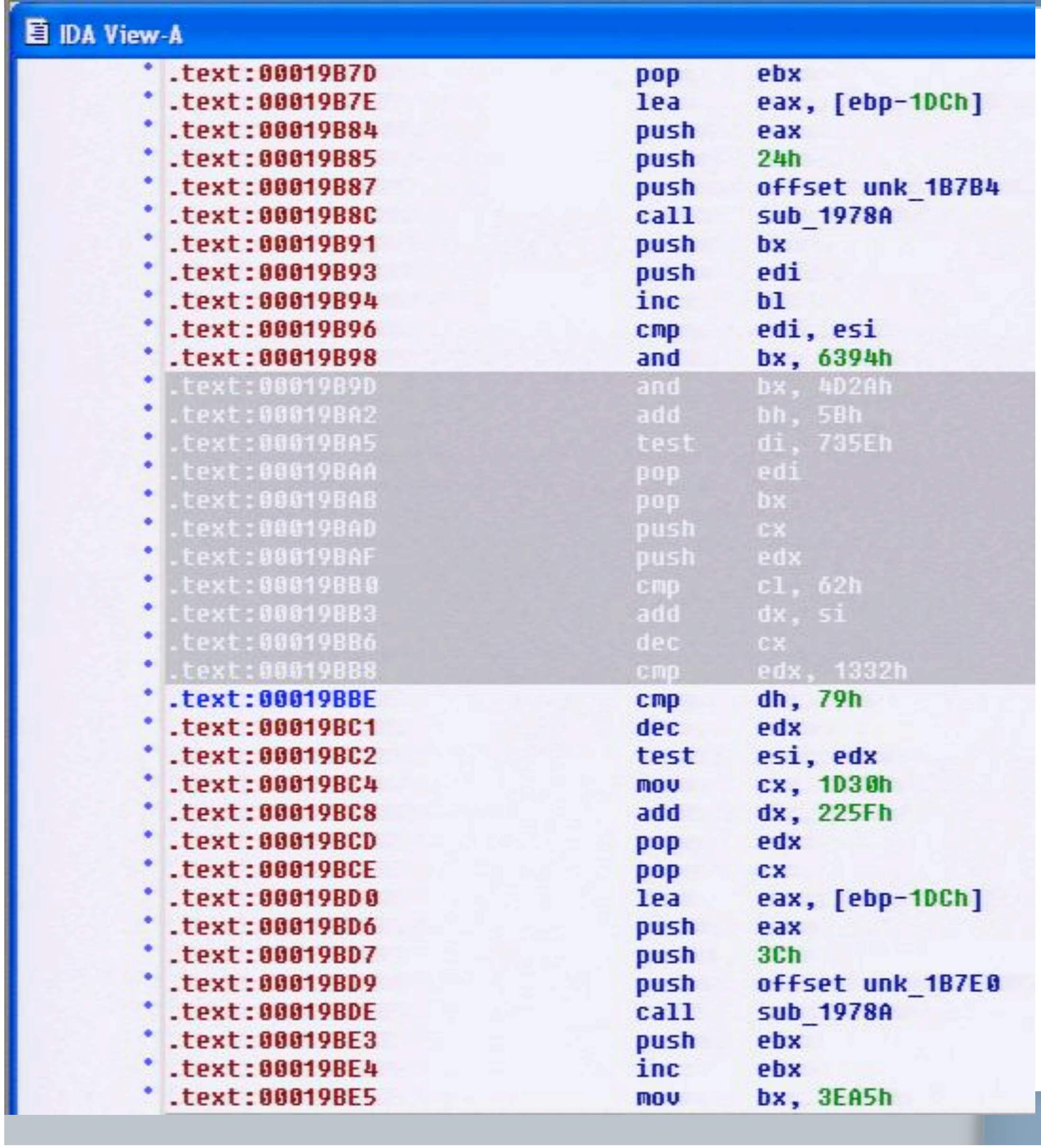
chiueh@cs.sunysb.edu

Kent Griffin
kent_griffin@symantec.com

Xin Hu
huxin@eecs.umich.edu

Scott Schneider
scott_schneider@symantec.com

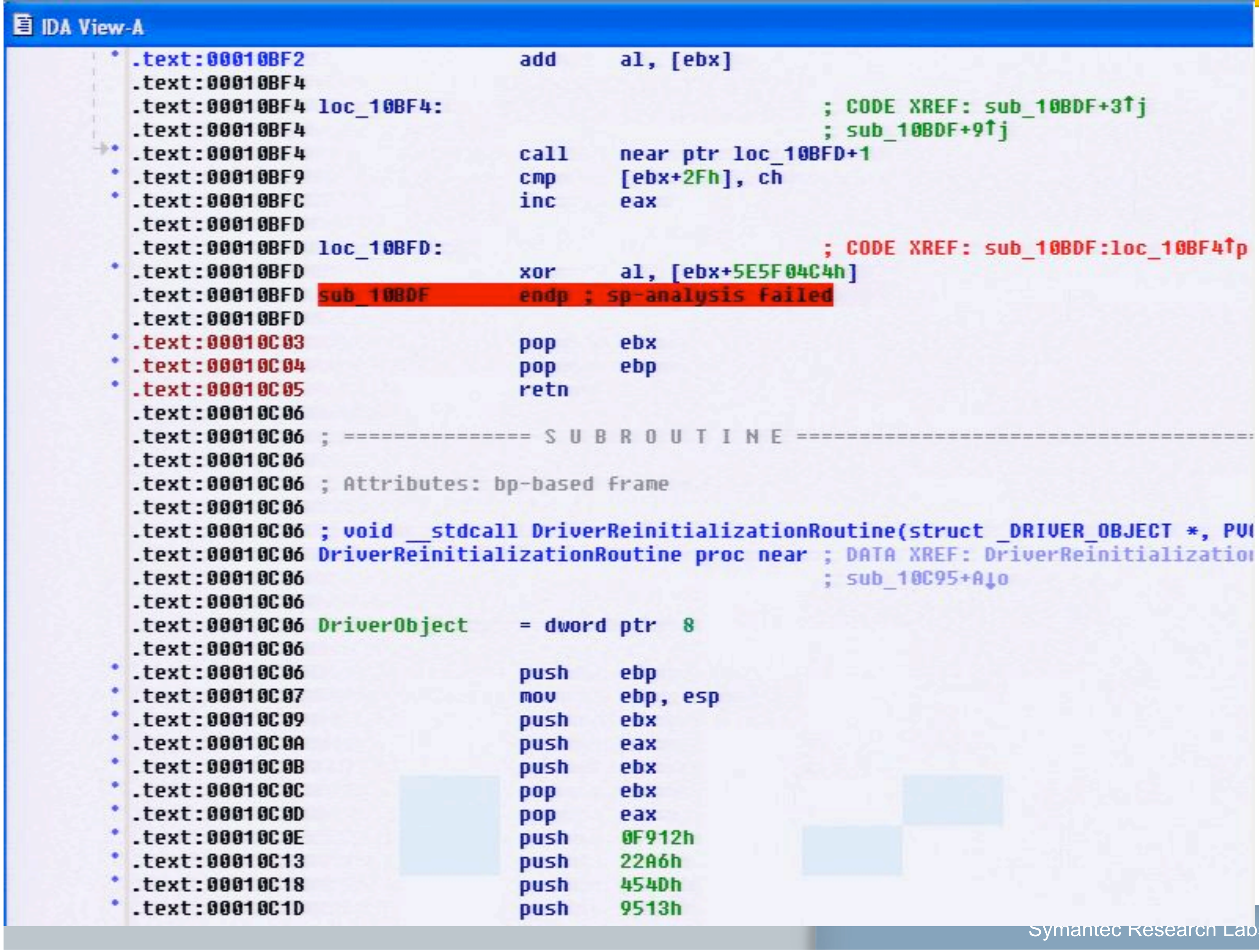
Good Signature #0



```
.text:00019B7D                  pop     ebx
.text:00019B7E                  lea     eax, [ebp-1DCh]
.text:00019B84                  push    eax
.text:00019B85                  push    24h
.text:00019B87                  push    offset unk_1B7B4
.text:00019B8C                  call    sub_1978A
.text:00019B91                  push    bx
.text:00019B93                  push    edi
.text:00019B94                  inc     bl
.text:00019B96                  cmp     edi, esi
.text:00019B98                  and     bx, 6394h
.text:00019B9D                  and     bx, 4D2Ah
.text:00019BA2                  add     bh, 5Bh
.text:00019BA5                  test    di, 735Eh
.text:00019BAA                  pop     edi
.text:00019BAB                  pop     bx
.text:00019BAD                  push    cx
.text:00019BAF                  push    edx
.text:00019BB0                  cmp     cl, 62h
.text:00019BB3                  add     dx, si
.text:00019BB6                  dec     cx
.text:00019BB8                  cmp     edx, 1332h
.text:00019BBE                  cmp     dh, 79h
.text:00019BC1                  dec     edx
.text:00019BC2                  test    esi, edx
.text:00019BC4                  mov     cx, 1D30h
.text:00019BC8                  add     dx, 225Fh
.text:00019BCD                  pop     edx
.text:00019BCE                  pop     cx
.text:00019BD0                  lea     eax, [ebp-1DCh]
.text:00019BD6                  push    eax
.text:00019BD7                  push    3Ch
.text:00019BD9                  push    offset unk_1B7E0
.text:00019BDE                  call    sub_1978A
.text:00019BE3                  push    ebx
.text:00019BE4                  inc     ebx
.text:00019BE5                  mov     bx, 3EA5h
```

- Uses 16-bit registers
- Several interesting constants
- Covers 73 files in our malware set
- Very low probability (-140)
- High interestingness score (33)
- Perfect diversity scores

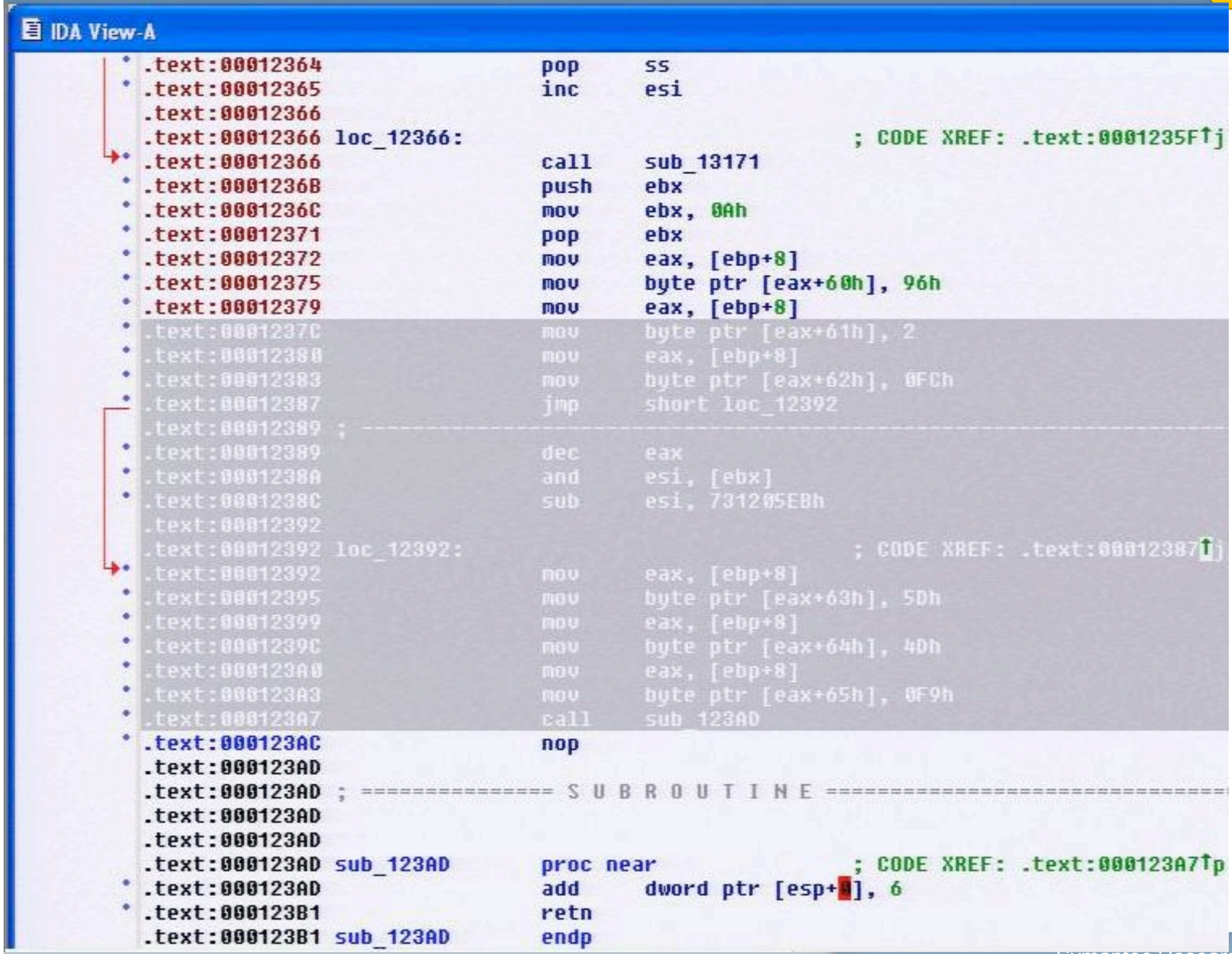
Good Signature #1



```
.text:00010BF2                         add     al, [ebx]
.text:00010BF4
.text:00010BF4 loc_10BF4:                                  ; CODE XREF: sub_10BDF+3↑j
.text:00010BF4                                             ; sub_10BDF+9↑j
.text:00010BF4                         call    near ptr loc_10BFD+1
.text:00010BF9                         cmp     [ebx+2Fh], ch
.text:00010BFC                         inc     eax
.text:00010BFD
.text:00010BFD loc_10BFD:                                  ; CODE XREF: sub_10BDF:loc_10BF4↑p
.text:00010BFD                         xor     al, [ebx+5E5F04C4h]
.text:00010BFD sub_10BDF               endp ; sp-analysis failed
.text:00010BFD
.text:00010C03                         pop     ebx
.text:00010C04                         pop     ebp
.text:00010C05                         retn
.text:00010C06
.text:00010C06 ; =============== S U B R O U T I N E =======================================
.text:00010C06
.text:00010C06 ; Attributes: bp-based frame
.text:00010C06
.text:00010C06 ; void __stdcall DriverReinitializationRoutine(struct _DRIVER_OBJECT *, PV
.text:00010C06 DriverReinitializationRoutine proc near ; DATA XREF: DriverReinitializatio
.text:00010C06                                             ; sub_10C95+A↓o
.text:00010C06
.text:00010C06 DriverObject            = dword ptr  8
.text:00010C06
.text:00010C06                         push    ebp
.text:00010C07                         mov     ebp, esp
.text:00010C09                         push    ebx
.text:00010C0A                         push    eax
.text:00010C0B                         push    ebx
.text:00010C0C                         pop     ebx
.text:00010C0D                         pop     eax
.text:00010C0E                         push    0F912h
.text:00010C13                         push    22A6h
.text:00010C18                         push    454Dh
.text:00010C1D                         push    9513h
```

- Several constants
- Covers 65 in our malware set
- Interestingness score 19
- Perfect diversity scores

Good Signature #2

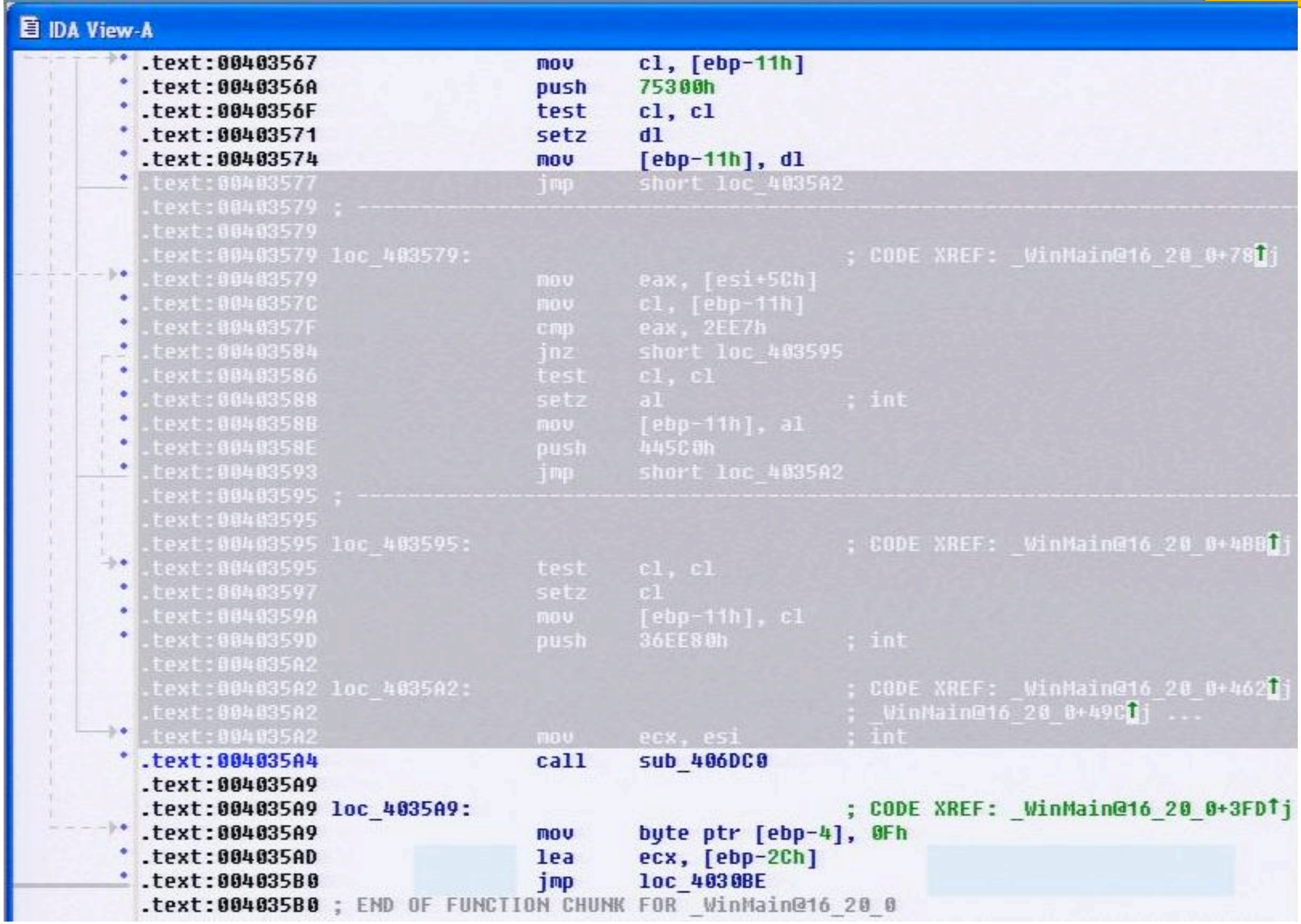


IDA View-A

```
.text:00012364                 pop     ss
.text:00012365                 inc     esi
.text:00012366
.text:00012366 loc_12366:                              ; CODE XREF: .text:0001235F↑j
.text:00012366                 call    sub_13171
.text:0001236B                 push    ebx
.text:0001236C                 mov     ebx, 0Ah
.text:00012371                 pop     ebx
.text:00012372                 mov     eax, [ebp+8]
.text:00012375                 mov     byte ptr [eax+60h], 96h
.text:00012379                 mov     eax, [ebp+8]
.text:0001237C                 mov     byte ptr [eax+61h], 2
.text:00012380                 mov     eax, [ebp+8]
.text:00012383                 mov     byte ptr [eax+62h], 0FCh
.text:00012387                 jmp     short loc_12392
.text:00012389 ; ---------------------------------------------------------------
.text:00012389                 dec     eax
.text:0001238A                 and     esi, [ebx]
.text:0001238C                 sub     esi, 731205EBh
.text:00012392
.text:00012392 loc_12392:                              ; CODE XREF: .text:00012387↑j
.text:00012392                 mov     eax, [ebp+8]
.text:00012395                 mov     byte ptr [eax+63h], 5Dh
.text:00012399                 mov     eax, [ebp+8]
.text:0001239C                 mov     byte ptr [eax+64h], 4Dh
.text:000123A0                 mov     eax, [ebp+8]
.text:000123A3                 mov     byte ptr [eax+65h], 0F9h
.text:000123A7                 call    sub_123AD
.text:000123AC                 nop
.text:000123AD
.text:000123AD ; =============== S U B R O U T I N E =========================
.text:000123AD
.text:000123AD
.text:000123AD sub_123AD       proc near               ; CODE XREF: .text:000123A7↑p
.text:000123AD                 add     dword ptr [esp+4], 6
.text:000123B1                 retn
.text:000123B1 sub_123AD       endp
```

- Several constants
- Covers 63 in our malware set
- Interesting-ness score 21
- Perfect diversity scores

So-so Signature #4

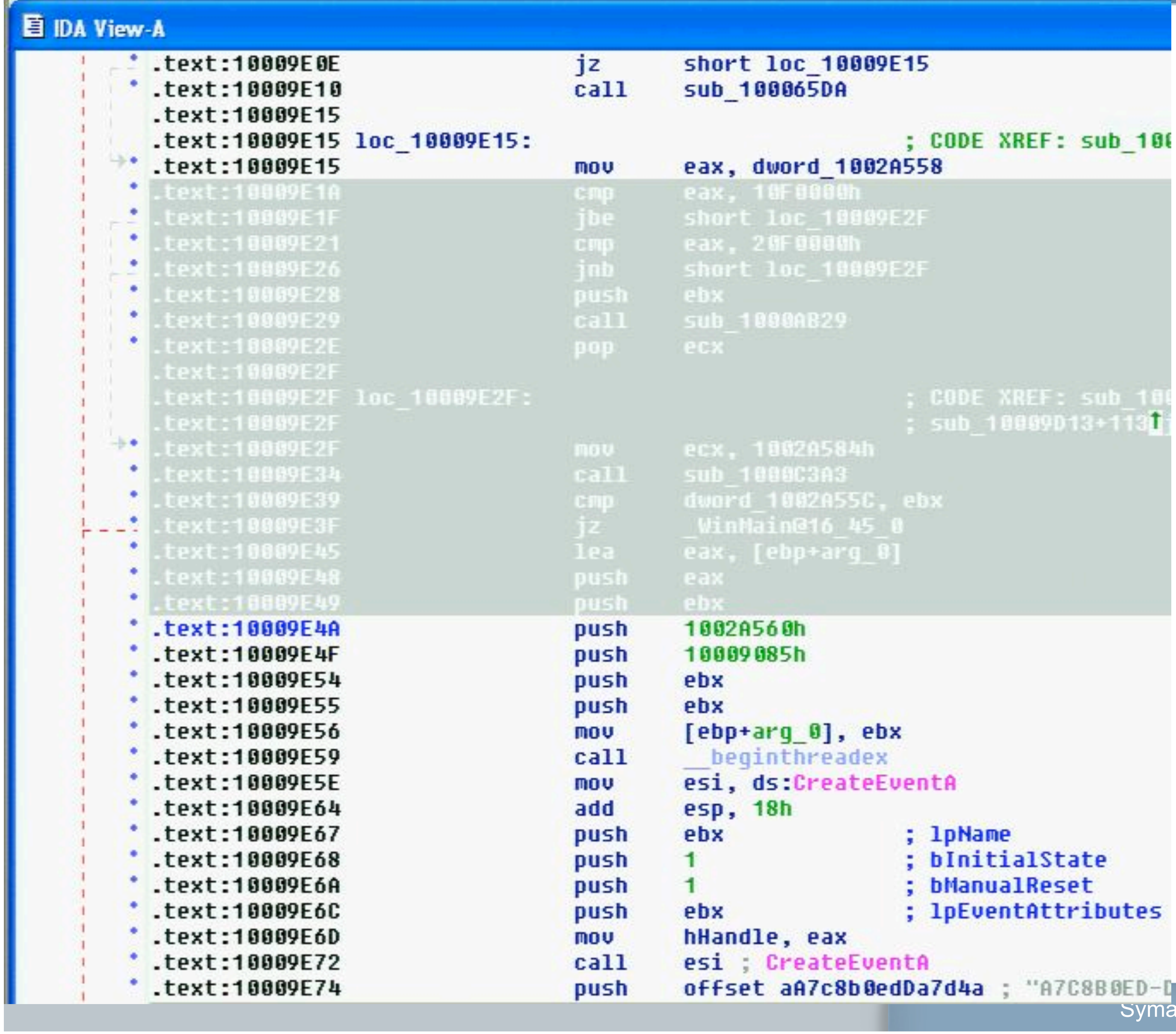


IDA View-A

```
.text:00403567                 mov     cl, [ebp-11h]
.text:0040356A                 push    75300h
.text:0040356F                 test    cl, cl
.text:00403571                 setz    dl
.text:00403574                 mov     [ebp-11h], dl
.text:00403577                 jmp     short loc_4035A2
.text:00403579 ; ---------------------------------------------------------------------------
.text:00403579
.text:00403579 loc_403579:                             ; CODE XREF: _WinMain@16_20_0+78↑j
.text:00403579                 mov     eax, [esi+5Ch]
.text:0040357C                 mov     cl, [ebp-11h]
.text:0040357F                 cmp     eax, 2EE7h
.text:00403584                 jnz     short loc_403595
.text:00403586                 test    cl, cl
.text:00403588                 setz    al              ; int
.text:0040358B                 mov     [ebp-11h], al
.text:0040358E                 push    445C0h
.text:00403593                 jmp     short loc_4035A2
.text:00403595 ; ---------------------------------------------------------------------------
.text:00403595
.text:00403595 loc_403595:                             ; CODE XREF: _WinMain@16_20_0+4B8↑j
.text:00403595                 test    cl, cl
.text:00403597                 setz    cl
.text:0040359A                 mov     [ebp-11h], cl
.text:0040359D                 push    36EE80h         ; int
.text:004035A2
.text:004035A2 loc_4035A2:                             ; CODE XREF: _WinMain@16_20_0+462↑j
.text:004035A2                                         ; _WinMain@16_20_0+49C↑j ...
.text:004035A2                 mov     ecx, esi        ; int
.text:004035A4                 call    sub_406DC0
.text:004035A9
.text:004035A9 loc_4035A9:                             ; CODE XREF: _WinMain@16_20_0+3FD↑j
.text:004035A9                 mov     byte ptr [ebp-4], 0Fh
.text:004035AD                 lea     ecx, [ebp-2Ch]
.text:004035B0                 jmp     loc_4030BE
.text:004035B0 ; END OF FUNCTION CHUNK FOR _WinMain@16_20_0
```

- Suspicious constants – multiples of 10,000
- This sig and variants cover 50+ files
- Interesting-ness score 13
- Good group count, std dev, single sig
- Eliminated by better threshold

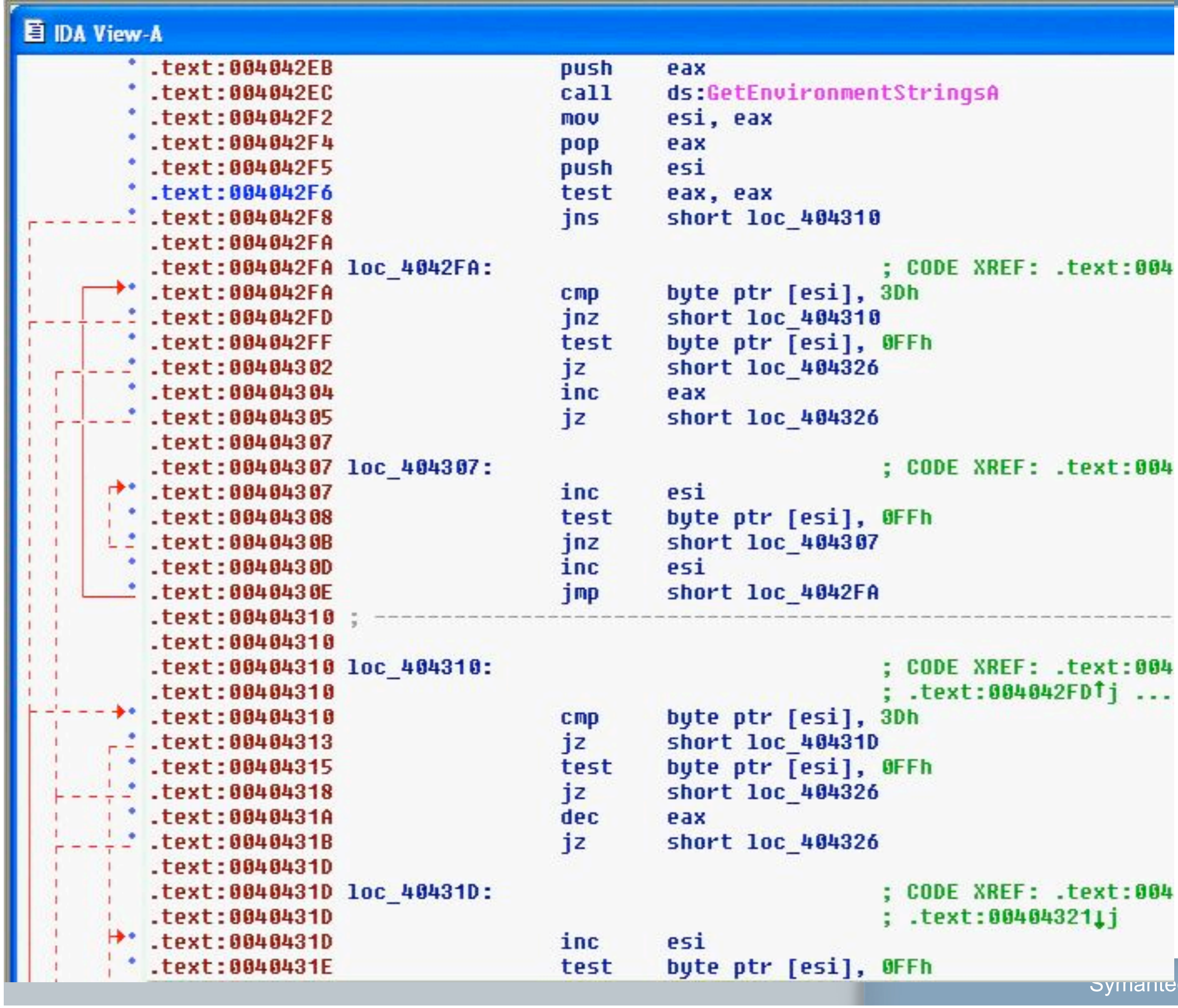
So-so Signature #50



```
.text:10009E0E                 jz      short loc_10009E15
.text:10009E10                 call    sub_100065DA
.text:10009E15
.text:10009E15 loc_10009E15:                           ; CODE XREF: sub_10
.text:10009E15                 mov     eax, dword_1002A558
.text:10009E1A                 cmp     eax, 10F0000h
.text:10009E1F                 jbe     short loc_10009E2F
.text:10009E21                 cmp     eax, 20F0000h
.text:10009E26                 jnb     short loc_10009E2F
.text:10009E28                 push    ebx
.text:10009E29                 call    sub_1000AB29
.text:10009E2E                 pop     ecx
.text:10009E2F
.text:10009E2F loc_10009E2F:                           ; CODE XREF: sub_10
.text:10009E2F                                         ; sub_10009D13+113↑
.text:10009E2F                 mov     ecx, 1002A584h
.text:10009E34                 call    sub_1000C3A3
.text:10009E39                 cmp     dword_1002A55C, ebx
.text:10009E3F                 jz      _WinMain@16_45_0
.text:10009E45                 lea     eax, [ebp+arg_0]
.text:10009E48                 push    eax
.text:10009E49                 push    ebx
.text:10009E4A                 push    1002A560h
.text:10009E4F                 push    10009085h
.text:10009E54                 push    ebx
.text:10009E55                 push    ebx
.text:10009E56                 mov     [ebp+arg_0], ebx
.text:10009E59                 call    __beginthreadex
.text:10009E5E                 mov     esi, ds:CreateEventA
.text:10009E64                 add     esp, 18h
.text:10009E67                 push    ebx             ; lpName
.text:10009E68                 push    1               ; bInitialState
.text:10009E6A                 push    1               ; bManualReset
.text:10009E6C                 push    ebx             ; lpEventAttributes
.text:10009E6D                 mov     hHandle, eax
.text:10009E72                 call    esi ; CreateEventA
.text:10009E74                 push    offset aA7c8b0edDa7d4a ; "A7C8B0ED-D
```

- 1 interesting constant
- Covers 4 files in our malware set
- Interestingness score 16
- Good diversity scores
- Eliminated by best thresholds

Bad Signature #16



```
.text:004042EB                 push    eax
.text:004042EC                 call    ds:GetEnvironmentStringsA
.text:004042F2                 mov     esi, eax
.text:004042F4                 pop     eax
.text:004042F5                 push    esi
.text:004042F6                 test    eax, eax
.text:004042F8                 jns     short loc_404310
.text:004042FA
.text:004042FA loc_4042FA:                             ; CODE XREF: .text:004
.text:004042FA                 cmp     byte ptr [esi], 3Dh
.text:004042FD                 jnz     short loc_404310
.text:004042FF                 test    byte ptr [esi], 0FFh
.text:00404302                 jz      short loc_404326
.text:00404304                 inc     eax
.text:00404305                 jz      short loc_404326
.text:00404307
.text:00404307 loc_404307:                             ; CODE XREF: .text:004
.text:00404307                 inc     esi
.text:00404308                 test    byte ptr [esi], 0FFh
.text:0040430B                 jnz     short loc_404307
.text:0040430D                 inc     esi
.text:0040430E                 jmp     short loc_4042FA
.text:00404310 ; ---------------------------------------------------------------
.text:00404310
.text:00404310 loc_404310:                             ; CODE XREF: .text:004
.text:00404310                                         ; .text:004042FD↑j ...
.text:00404310                 cmp     byte ptr [esi], 3Dh
.text:00404313                 jz      short loc_40431D
.text:00404315                 test    byte ptr [esi], 0FFh
.text:00404318                 jz      short loc_404326
.text:0040431A                 dec     eax
.text:0040431B                 jz      short loc_404326
.text:0040431D
.text:0040431D loc_40431D:                             ; CODE XREF: .text:004
.text:0040431D                                         ; .text:00404321↓j
.text:0040431D                 inc     esi
.text:0040431E                 test    byte ptr [esi], 0FFh
```

- Generic logic
- Only 1 interesting 1-byte constant
- Covers 7 files
- Interestingness score 13
- Bad diversity scores