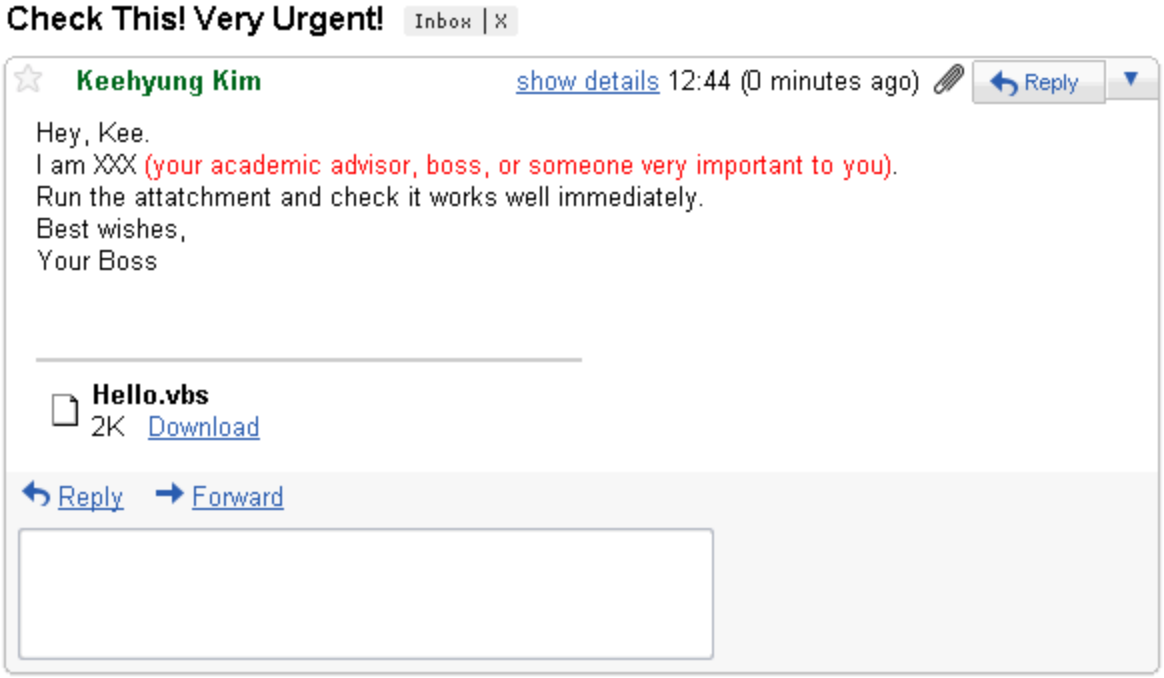
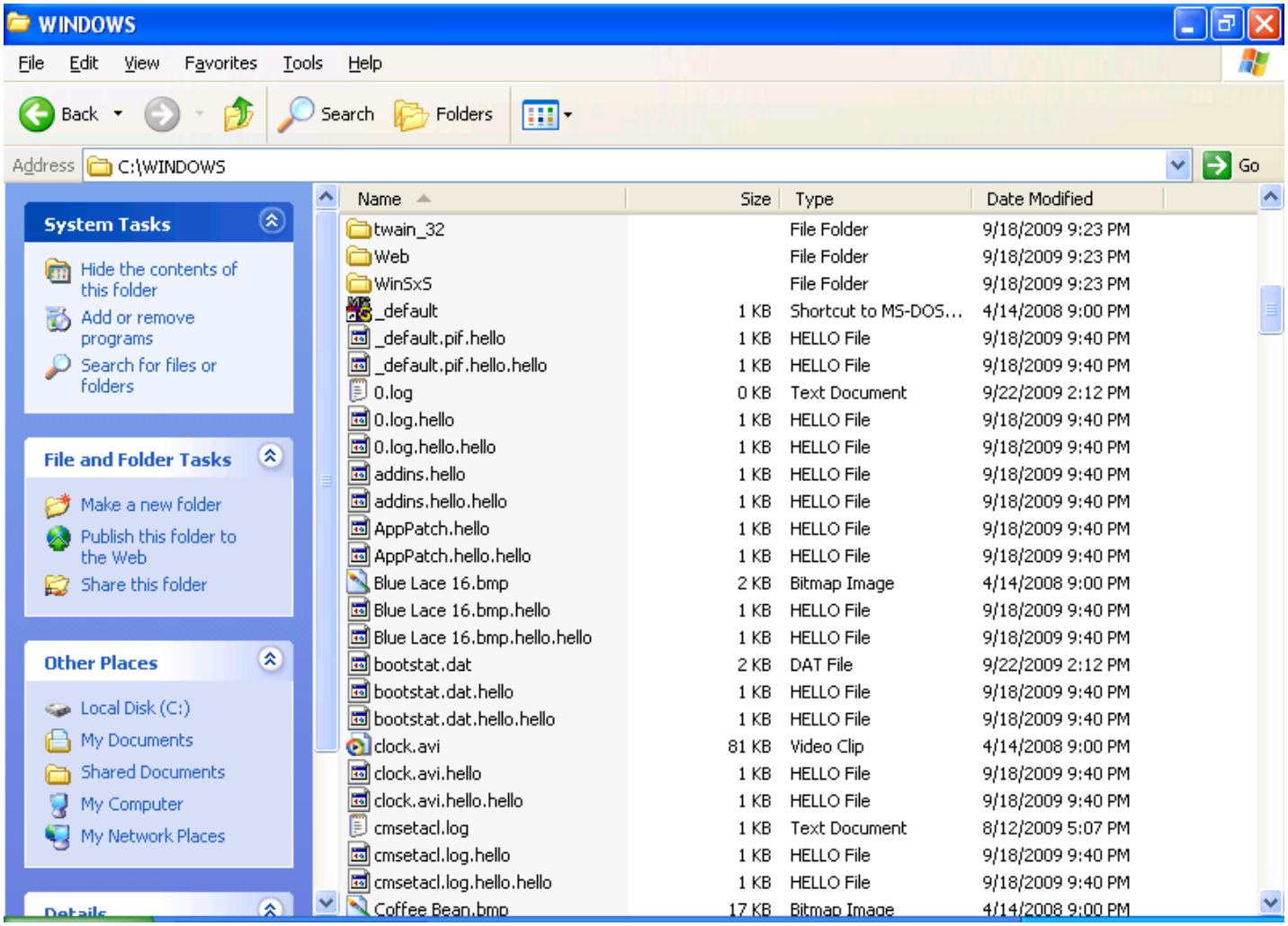
Malware Detection based on Dependency Graph using Hybrid Genetic Algorithm

Keehyung Kim, Byung-Ro Moon
keehyung@snu.ac.kr
School of Computer Science and Engineering
Seoul National University
July 11, 2010 / Portland, Oregon, US / GECCO 2010
(Jin Hyun Kim, for ROSAEC workshop)


Check This! Very Urgent!
Inbox | X
Keehyung Kim
show details 12:44 (0 minutes ago)
Reply
Hey, Kee.
I am XXX (your academic advisor, boss, or someone very important to you).
Run the attatchment and check it works well immediately.
Best wishes,
Your Boss
Hello.vbs
2K Download
Reply
Forward


WINDOWS
File Edit View Favorites Tools Help
Back Search Folders
Address C:\WINDOWS
Go
System Tasks
Hide the contents of this folder
Add or remove programs
Search for files or folders
File and Folder Tasks
Make a new folder
Publish this folder to the Web
Share this folder
Other Places
Local Disk (C:)
My Documents
Shared Documents
My Computer
My Network Places
Details
Name Size Type Date Modified
twain_32 File Folder 9/18/2009 9:23 PM
Web File Folder 9/18/2009 9:23 PM
WinSxS File Folder 9/18/2009 9:23 PM
_default 1 KB Shortcut to MS-DOS... 4/14/2008 9:00 PM
_default.pif.hello 1 KB HELLO File 9/18/2009 9:40 PM
_default.pif.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
0.log 0 KB Text Document 9/22/2009 2:12 PM
0.log.hello 1 KB HELLO File 9/18/2009 9:40 PM
0.log.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
addins.hello 1 KB HELLO File 9/18/2009 9:40 PM
addins.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
AppPatch.hello 1 KB HELLO File 9/18/2009 9:40 PM
AppPatch.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
Blue Lace 16.bmp 2 KB Bitmap Image 4/14/2008 9:00 PM
Blue Lace 16.bmp.hello 1 KB HELLO File 9/18/2009 9:40 PM
Blue Lace 16.bmp.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
bootstat.dat 2 KB DAT File 9/22/2009 2:12 PM
bootstat.dat.hello 1 KB HELLO File 9/18/2009 9:40 PM
bootstat.dat.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
clock.avi 81 KB Video Clip 4/14/2008 9:00 PM
clock.avi.hello 1 KB HELLO File 9/18/2009 9:40 PM
clock.avi.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
cmsetacl.log 1 KB Text Document 8/12/2009 5:07 PM
cmsetacl.log.hello 1 KB HELLO File 9/18/2009 9:40 PM
cmsetacl.log.hello.hello 1 KB HELLO File 9/18/2009 9:40 PM
Coffee Bean.bmp 17 KB Bitmap Image 4/14/2008 9:00 PM

12 anti-viruses detect Hello virus.

File **dlH.str** received on **2009.09.18 12:19:17 (UTC)**
Current status: **finished**
Result: **12/41 (29.27%)**

Compact | Print results

Antivirus	Version	Last Update	Result
a-squared	4.5.0.24	2009.09.18	-
AhnLab-V3	5.0.0.2	2009.09.18	VBS/Pluta
AntiVir	7.9.1.19	2009.09.18	-
Antiy-AVL	2.0.3.7	2009.09.18	-
Authentium	5.1.2.4	2009.09.18	-
Avast	4.8.1351.0	2009.09.17	VBS:Malware-gen
AVG	8.5.0.412	2009.09.18	-
BitDefender	7.2	2009.09.18	Win32.Worm.VBS.A
CAT-QuickHeal	10.00	2009.09.18	-
ClamAV	0.94.1	2009.09.17	-
Comodo	2358	2009.09.18	-
DrWeb	5.0.0.12182	2009.09.18	VBS.Pluto
eSafe	7.0.17.0	2009.09.17	VBS.Peneluta.b
eTrust-Vet	31.6.6745	2009.09.18	VBS/Pluta.D
F-Prot	4.5.1.85	2009.09.18	-
F-Secure	8.0.14470.0	2009.09.18	Worm.VBS.Pluta.b
Fortinet	3.120.0.0	2009.09.18	VBS/Pluta.B!worm
GData	19	2009.09.18	Win32.Worm.VBS.A

Test website: http://www.virustotal.com

Hello variant

Hello	Hello.v1 (format alteration)

'rem - VBS/dlH "DL Hello" Virus - By D.L.
'rem - Written on November 12th, 2003
 	
On Error Resume Next
dim FSobj,orgMes,finalMes
set FSobj=CreateObject("Scripting.FileSystemObject")
 	
orgMes="Hello! Don't be mad...I'm not a bad bug :) - by *|%|"
orgMes=replace(orgMes,chr(42),chr(68))
orgMes=replace(orgMes,chr(124),chr(46))
finalMes=replace(orgMes,chr(37),chr(76))
 	
On Error Resume Next
dim drive,machine
 	
set machine=FSobj.Drives
for each drive in machine
  if (drive.DriveType=2)or(drive.DriveType=3) then
  indexFolders(drive.Path&"\")
  end If
next
 	
sub indexFolders(location)
  .....
sub writeData(location)
  .....
	On Error Resume Next
	dim FSobj,orgMes,finalMes
	set FSobj=CreateObject("Scripting.FileSystemObject")
	
	orgMes="asdf"
	orgMes=replace(orgMes,chr(42),chr(68))
	orgMes=replace(orgMes,chr(124),chr(46))
	finalMes=replace(orgMes,chr(37),chr(76))
	
	On Error Resume Next
	dim drive,machine
	
	set machine=FSobj.Drives
	for each drive in machine
	  if (drive.DriveType=2)or(drive.DriveType=3) then
	  indexFolders(drive.Path&"\")
	  end If
	next
	
	sub indexFolders(location)
	  .....
	sub writeData(location)
	  .....

Delete comment

Change message

Only 5 out of 12 detect Hello.v1 virus.

File **dlH_b.str** received on **2009.09.18 12:34:02 (UTC)**
Current status: **finished**
Result: **5/41 (12.20%)**

Compact | Print results

Antivirus	Version	Last Update	Result
a-squared	4.5.0.24	2009.09.18	-
AhnLab-V3	5.0.0.2	2009.09.18	-
AntiVir	7.9.1.19	2009.09.18	-
Antiy-AVL	2.0.3.7	2009.09.18	
Authentium	5.1.2.4		
Avast	4.8.1		
AVG			
BitDefe		.09.18	-
CAT-Quic		2009.09.18	-
ClamAV	0.94.1	2009.09.17	-
Comodo	2359	2009.09.18	-
DrWeb	5.0.0.12182	2009.09.18	-
eSafe	7.0.17.0	2009.09.17	VBS.Peneluta.b
eTrust-Vet	31.6.6745	2009.09.18	VBS/Pluta.D
F-Prot	4.5.1.85	2009.09.18	-
F-Secure	8.0.14470.0	2009.09.18	Worm.VBS.Pluta.b
Fortinet	3.120.0.0	2009.09.18	-
GData	19	2009.09.18	-

Even the simplest polymorphism can confuse Anti-viruses!

Test website: http://www.virustotal.com

Polymorphic Malwares

- Polymorphic variants of Hello virus
  - Change appearance, but keep functionality

New Malware	Plymorphic techniques	Detection Rate
Hello	-	12/41
Hello.v1	Format alteration	5/41
Hello.v2	Statement replacement	3/41
Hello.v3	Format alteration, variable renaming	1/41
Hello.v4	Format alteration, variable renaming, statement replacement, junk code insertion, spaghetti code	0/41

- Anti-viruses are weak against polymorphism
  - Mostly based on signature-based scanner

```
dim n, p, i
n = 5
p = 1
for i = 1 to n do
   p = p * i
end for
```

(a) Original code

```
dim a, b, c
a = 5
b = 1
for c = 1 to a do
   b = b * c
end for
```

(b) Variable renaming

```
dim i, p
p = 1
dim n
n = 5
for i = 1 to n do
   p = i * p
end for
```

(c) Statement reordering

```
dim n, p, i
n = 5
p = n / 5
for i = 1 to n do
   p = p * i
end for
```

(d) Statement replacement

```
dim n, p, i
n = 5
p = 1
i = 1
while i <= n do
   p = p * i
   i = i + 1
end while
```

(e) Control replacement

```
dim n, p, i
n = 5
p = 1
for i = 1 to n do
   if i>0 then
      p = p * i
   end if
end for
```

(f) Junk code insertion

```
dim n, p, i
goto X:
Y:
for i = 1 to n do
   p = p * i
end for
goto Z:
X:
n = 5
p = 1
goto Y:
Z:
```

(g) Spaghetti code

```
dim n, p, i
n = 5
p = 1
for i = 1 to n do
   p = prod(p, i)
end for

function prod(a, b)
return a * b
```

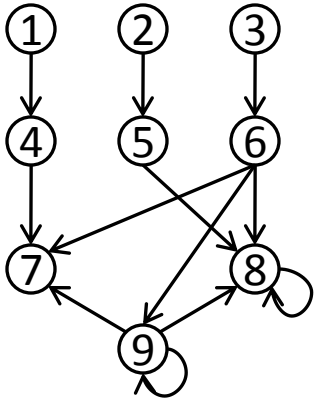
(h) Subroutine outlining

Figure 1: Example of Polymorphism

Malware Detection as an Optimization Problem

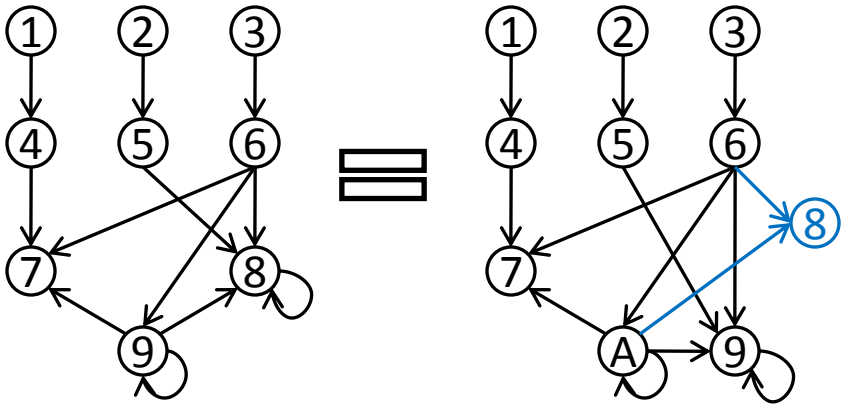
- Malware into Dependency Graph
  - Only appearance changes
  - Variable dependency remains
    - Logic and relation among variables remain
- Maximum Subgraph Isomorphism

```
Dim n,p,i
n=5
p=1
For i=1 to n
   p = p * i
Next
```



Malware Detection as an Optimization Problem

- Malware into Dependency Graph
- Maximum Subgraph Isomorphism
  - If two dependency graphs share much, they can be polymorphic
  - To detect unknown polymorphic malware, find maximum subgraph with the known one's



Dependency Graph

- Generate dependency graph based on semantics

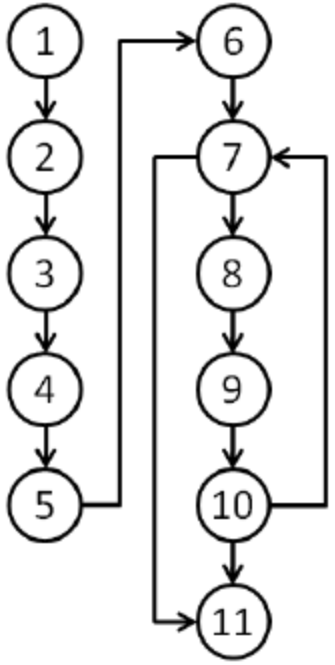
Original Code

```
1: dim n, p, i
2: n = 5
3: p = 1
4: for i = 1 to n do
5:     p = p * i
6: end for
```

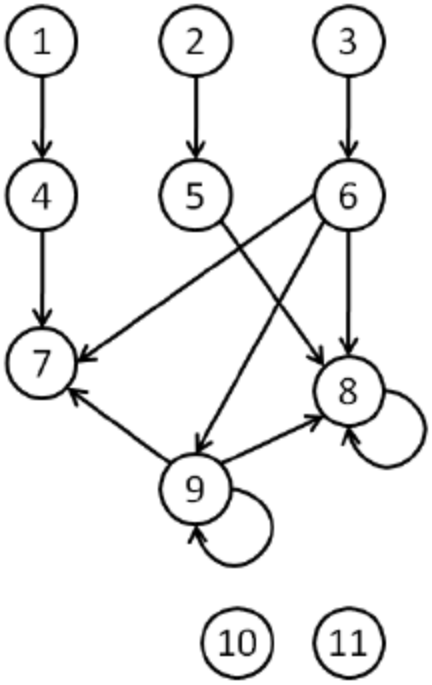
Semantic Code

```
1: dim n
2: dim p
3: dim i
4: n = 5
5: p = 1
6: i = 1
7: if i ≤ n then
8:     p = p * i
9:     i = i + 1
10:    goto 7:
11: end if
```

Control Flow Graph



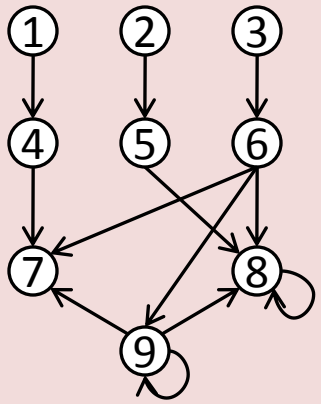
Dependency Graph



Polymorphism and Dependency Graph

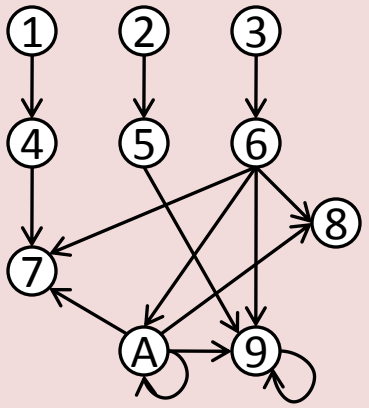
Original code

```
Dim n,p,i
n=5
p=1
For i=1 to n
   p = p * i
Next
```



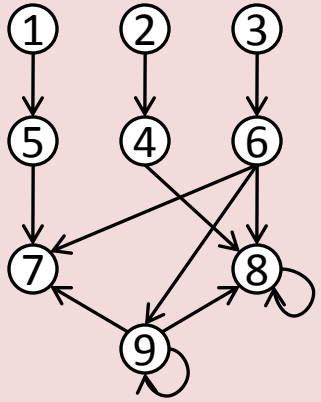
Junk code insertion

```
Dim n,p,i
n=5
p=1
For i=0 to n
      if i<>0 then
                  p = p * i
      end if
Next
```



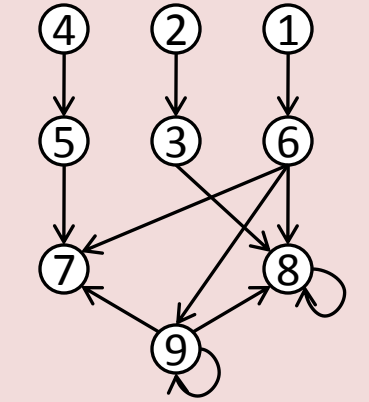
Variable renaming

```
Dim a,b,c
b=1
a=5
For c=1 to a
      b = b*c
Next
```



Statement reordering

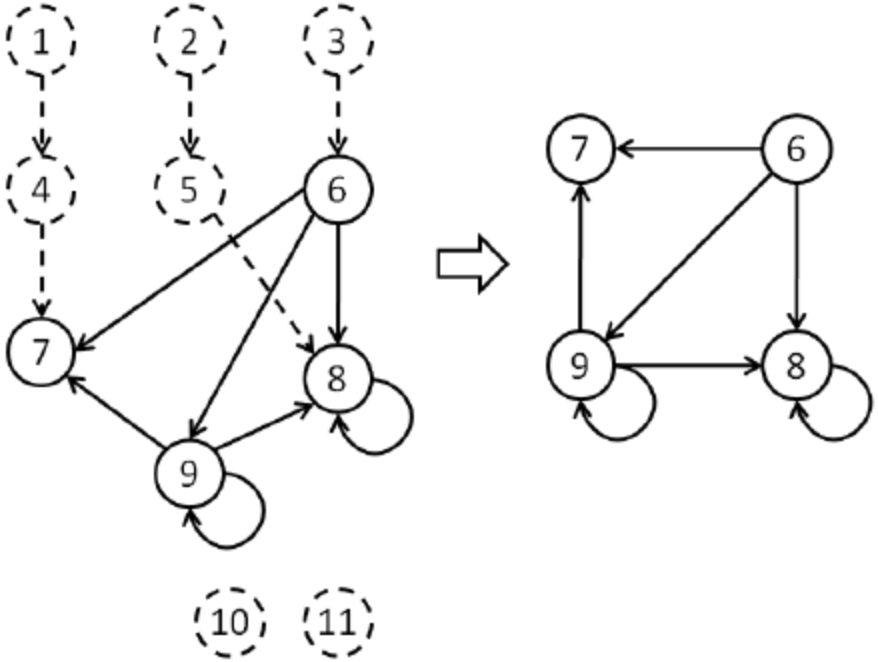
```
Dim i,p
p=1
Dim n
n=5
For i=0 to n
      p = i * p
Next
```



Graph Reduction

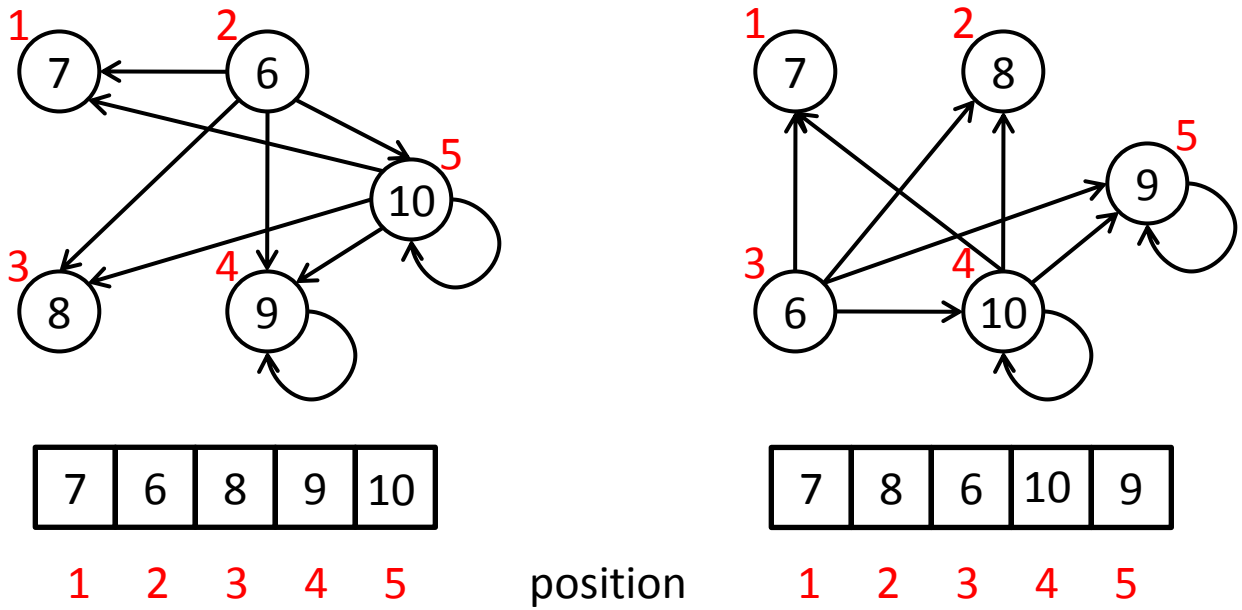
- Increase performance (speed)
- Eligibility for reduction

# of outgoing	# of incoming
1	0
Declaration of a variable	
0	1
Just use value in a variable	
1	1
Convey value from one to another	
0	0
Non-necessary redundant part	



Genetic Algorithm

- Representation
  - Linear encoding
  - Each gene represents location of the vertex in permutation



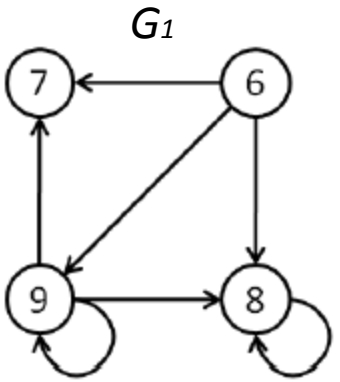
Genetic Algorithm

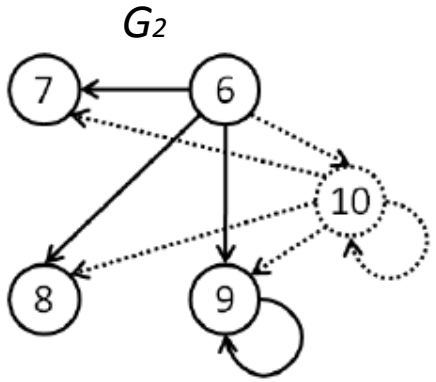
- Fitness Function

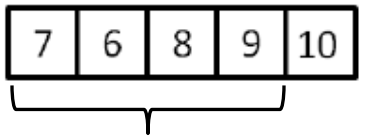
$$I(e,E) = \begin{cases} 0 & \text{if } e \in E \\ 1 & \text{otherwise.} \end{cases}$$

$$d = \frac{\sum_{e \in E_1} I(e,E_2) + \sum_{e \in E_2} I(e,E_1)}{|E_1|}$$

  - Smaller *d*, higher fitness
  - If *d=0*, a complete subgraph exists







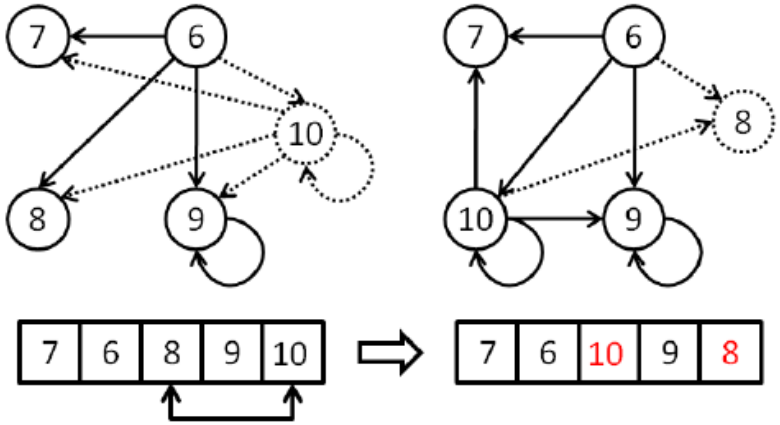
d=0.429

Genetic Algorithm

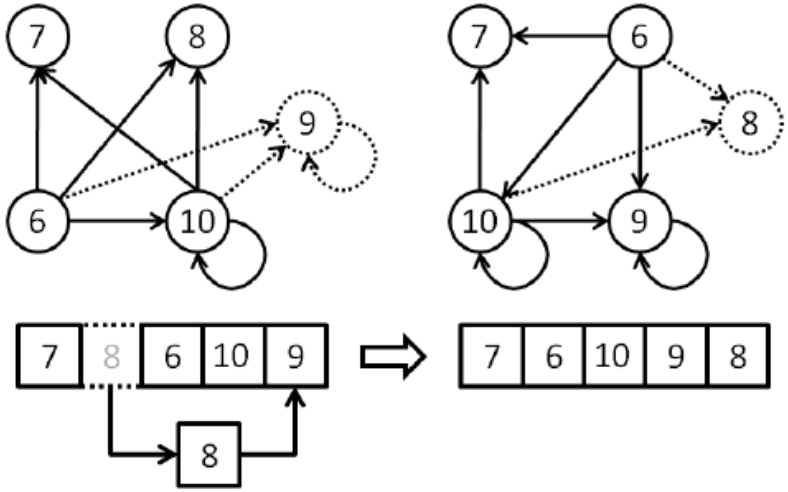
- Initialization
  - 1 solution in increasing order
  - 99 solutions randomly
- Selection
  - Roulette-wheel-based proportional selection (*k=3*)
- Crossover
  - Better one from Cycle and PMX (P_C = 0.9)
- Mutation
  - Exchange randomly selected two genes (P_m = 0.2)
- Replacement
  - 20 worst members to new offspring every generation
- Stop condition
  - When finding a complete subgraph isomorphism (*d=0*)
  - Maximum generation: 100 in Hybrid GA

Heuristics

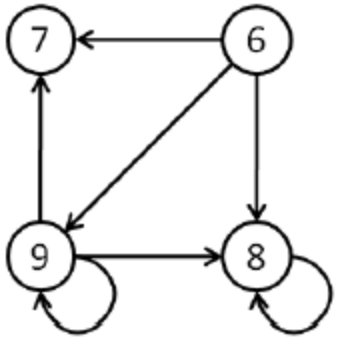
- Two-vertex Exchange Heuristic



- Vertex Relocation Heuristic



Target Graph



Dataset

- Malwares used for test

Malware Series	Variants
Hello	Hello, Hello.v1, Hello.v2, Hello.v3, Hello.v4
Neves	Neves.a, Neves.b, Neves.c, Neves.d
Rabbit	Rabbit.a, Rabbit.b
Internal	Internal.a, Internal.b, Internal.c, Internal.f, Internal.g
Small	Small.a, Small.b

Malware Source: http://vx.netlux.org

Experimental Result – Heuristic Only

Virus	Hello	Neves	Rabbit	Internal	Small
Hello	0	81.25	144.44	72.72	60
Hello.v1	0	81.25	144.44	72.72	60
Hello.v2	0	81.25	144.44	72.72	60
Hello.v3	36.84	65.62	88.88	72.72	80
Hello.v4	36.84	65.62	88.88	72.72	80
Neves.a	81.25	0	111.11	86.36	60
Neves.b	92.1	34.37	122.22	86.36	60
Neves.c	81.25	0	111.11	86.36	60
Neves.d	97.36	106.25	88.88	77.27	80
Rabbit.a	144.44	111.11	0	188.88	60
Rabbit.b	180	130	11.11	160	60
Internal.a	118.42	121.87	155.55	113.63	80
Internal.b	65.78	106.25	155.55	113.63	80
Internal.c	72.72	59.37	188.88	0	60
Internal.f	65.78	87.17	155.55	113.63	80
Internal.g	102.63	81.25	111.11	81.81	80
Small.a	60	60	60	80	0
Small.b	92.86	57.14	100	85.71	80

Experim…ult – Heuristic Only

8 out of 18 malwares are detected as having a complete subgraph by Heuristics!

12 out of 18 malwares are detected by Heuristics if $\alpha = 40$.

			Rabbit	Internal	Small
			144.44	72.72	60
			144.44	72.72	60
		81.25	144.44	72.72	60
		65.62	88.88	72.72	80
	36.84	65.62	88.88	72.72	80
Neves.a	81.25	0	111.11	86.36	60
Neves.b	92.1	34.37	122.22	86.36	60
Neves.c	81.25	0	111.11	86.36	60
Neves.d	97.36	106.25	88.88	77.27	80
Rabbit.a	144.44	111.11	0	188.88	60
Rabbit.b	180	130	11.11	160	60
Internal.a	118.42	121.87	155.55	113.63	80
Internal.b	65.78	106.25	155.55	113.63	80
Internal.c	72.72	59.37	188.88	0	60
Internal.f	65.78	87.17	155.55	113.63	80
Internal.g	102.63	81.25	111.11	81.81	80
Small.a	60	60	60	80	0
Small.b	92.86	57.14	100	85.71	80

Experimental Result – Hybrid GA

Virus	Hello	Neves	Rabbit	Internal	Small
Hello	0	28.13	22.22	40.91	40
Hello.v1	0	28.13	22.22	36.36	40
Hello.v2	0	28.13	11.11	36.36	40
Hello.v3	0	31.25	22.22	31.82	40
Hello.v4	0	34.38	22.22	36.36	40
Neves.a	28.13	0	22.22	36.36	40
Neves.b	63.16	0	22.22	31.82	40
Neves.c	28.13	0	22.22	36.36	40
Neves.d	31.58	0	33.33	27.27	40
Rabbit.a	22.22	22.22	0	155.56	60
Rabbit.b	20	40	0	130	60
Internal.a	55.3	40.6	22.2	27.3	40
Internal.b	20	22	77.78	0	40
Internal.c	40.91	36.36	155.56	0	40
Internal.f	21	22	77.78	0	40
Internal.g	44.7	40.6	44.4	50	40
Small.a	40	40	60	40	0
Small.b	28.57	42.86	57.14	35.71	0

Experi…esult – Hybrid GA

16 out of 18 malwares are detected by Hybrid Genetic Algorithm!

	Neves	Rabbit	Internal	Small
	0	28.13	22.22	40.91	40
Hello.v1	0	28.13	22.22	36.36	40
Hello.v2	0	28.13	11.11	36.36	40
Hello.v3	0	31.25	22.22	31.82	40
Hello.v4	0	34.38	22.22	36.36	40
Neves.a	28.13	0	22.22	36.36	40
Neves.b	63.16	0	22.22	31.82	40
Neves.c	28.13	0	22.22	36.36	40
Neves.d	31.58	0	33.33	27.27	40
Rabbit.a	22.22	22.22	0	155.56	60
Rabbit.b	20	40	0	130	60
Internal.a	55.3	40.6	22.2	27.3	40
Internal.b	20	22	77.78	0	40
Internal.c	40.91	36.36	155.56	0	40
Internal.f	21	22	77.78	0	40
Internal.g	44.7	40.6	44.4	50	40
Small.a	40	40	60	40	0
Small.b	28.57	42.86	57.14	35.71	0

Conclusion

- Summary
  - Malware detection as Maximum Subgraph Isomorphism Problem
  - Useful to detect unknown polymorphic malwares
  - Possible to be extended to related areas
  - Graph reduction and Heuristics enhance performance
- Future Work
  - Reduce computational cost
  - Extensive experiments

Comparison with Anti-Viruses

Virus Variant	Anti-Viruses	Proposed System
Hello	12/41 (29.27%)	Detected
Hello.v1	5/41 (12.20%)	Detected
Hello.v2	3/41 (7.32%)	Detected
Hello.v3	1/41 (2.44%)	Detected
Hello.v4	0/41 (0%)	Detected
Neves.a	32/40 (80%)	Detected
Neves.b	32/40 (80%)	Detected
Neves.c	25/41 (60.98%)	Detected
Neves.d	25/41 (60.98%)	Detected
Rabbit.a	36/41 (87.80%)	Detected
Rabbit.b	33/39 (84.62%)	Detected
Internal.a	14/41 (34.15%)	Undetected
Internal.b	19/41 (46.35%)	Detected
Internal.c	16/40 (40%)	Detected
Internal.f	13/41 (31.71%)	Detected
Internal.g	21/41 (51.22%)	Undetected
Small.a	35/41 (85.37%)	Detected
Small.b	7/41 (17.08%)	Detected

Table 3: Graph reduction on Node

Virus	# Nodes	# Nodes after Reduction	Ratio
Hello	53	25	47.17
Hello.v1	53	25	47.17
Hello.v2	53	25	47.17
Hello.v3	53	25	47.17
Hello.v4	50	25	50
Neves.a	60	21	35
Neves.b	72	25	34.72
Neves.c	100	21	21
Neves.d	122	39	31.97
Rabbit.a	13	0	0
Rabbit.b	14	0	0
Internal.a	65	31	47.69
Internal.b	40	18	45
Internal.c	35	13	37.14
Internal.f	39	18	46.15
Internal.g	160	65	40.63
Small.a	14	4	28.57
Small.b	18	10	55.56

Table 4: Graph reduction on Edge

Virus	# Edges	# Edges after Reduction	Ratio
Hello	59	38	64.41
Hello.v1	59	38	64.41
Hello.v2	59	38	64.41
Hello.v3	59	38	64.41
Hello.v4	55	38	69.09
Neves.a	65	32	49.23
Neves.b	81	43	53.09
Neves.c	71	32	45.07
Neves.d	134	73	54.48
Rabbit.a	9	0	0
Rabbit.b	10	0	0
Internal.a	86	68	79.07
Internal.b	47	39	82.98
Internal.c	32	22	68.75
Internal.f	47	39	82.98
Internal.g	163	110	67.48
Small.a	14	5	35.71
Small.b	17	14	82.35