Graphical Malware Actuation with 🐼 and ⚡

Brendan Dolan-Gavitt
NYU School of Engineering
PANDA: Built for RE

- Based on QEMU 1.0.1
- Deterministic record/replay
- Translation to LLVM for all QEMU architectures (extended from S2E code)
- Android (ARM) emulation support
- Plugin architecture – easy to extend to new analyses
Record/Replay

CPU

Outside World

== Friday?

== 0x45?

>= 0x80?
Record/Replay

CPU

outside world

Get Current Date

Fri May 23 11:33:27

== Friday?

== 0x45?

>= 0x80?

Fri May 23 11:33:27
Record/Replay

CPU

Get Current Date

Outside World

Fri May 23 11:33:27

== Friday?

== 0x45?

>= 0x80?
Record/Replay

CPU

== Friday?

>= 0x45?

Outside World

Get Current Date

Fri May 23 11:33:27

Recv Packet

0x0000:  4500 002c 0000 4000
0x0008:  4006 6b48 127e 0021
0x0010:  5dae 5f37 01bb bed4
0x0018:  fc0d 820f d690 0847
0x0020:  6012 3908 cfa2 0000
0x0028:  0204 05b4

CPU

Outside World
**Record/Replay**

**CPU**

- == Friday?
- == 0x45?
- >= 0x80?

**Outside World**

- Fri May 23 11:33:27

**Recv Packet**

- 0x0000:  4500 002c 0000 4000
- 0x0008:  4006 6b48 127e 0021
- 0x0010:  5dae 5f37 01bb bed4
- 0x0018:  fccd 820f d690 0847
- 0x0020:  6012 3908 cfa2 0000
- 0x0028:  0204 05b4
Record/Replay

CPU

== Friday?

Get Current Date

Fri May 23 11:33:27

Recv Packet

0x0000:  4500 002c 0000 4000
0x0008:  4006 6b48 127e 0021
0x0010:  5dae 5f37 01bb bed4
0x0018:  fccd 820f d690 0847
0x0020:  6012 3908 cfa2 0000
0x0028:  0204 05b4

Outside World
Record/Replay

CPU

Get Current Date

Recv Packet

>= 0x80?

== 0x45?

== Friday?

Outside World

Fri May 23 11:33:27

Record Log
Malware Reproducibility

• Big problem in malware research: *dynamic analysis is not reproducible*

• E.g. if C&C servers go down, malware does not behave the same way

• Combined with lack of access to samples, creates barriers to entry for research, problems with replication
Malrec: Reproducible Malware Analysis for All

- Idea: run malware samples in PANDA and share recordings

- Since December we have been running 100 samples every day

- [http://panda.gtisc.gatech.edu/malrec/](http://panda.gtisc.gatech.edu/malrec/)

- As of 5/28/2015: 17,200 recordings, PCAPs, movies
Replay Subsumes Other Artifacts

- PCAPs
- Reports
- Memory Dumps
- Screenshots
- Movies
Malware Pipeline

- Ingest Malware Feed
- Malware Recorders
- Log Compressors
- Compressed Malware Feed
Graphical Malware

- Lots of malware has some UI: Fake AV, trojaned installers, adware, etc.

- We will ignore the issue for now of whether it’s still malware if you have to click “OK” to run it…

- Back of envelope calculation: ~60% of the malware we’ve seen has some GUI component
Fake AV
Trojaned Installer
SourceForge grabs GIMP for Windows’ account, wraps installer in bundle-pushing adware [Updated]

SourceForge says account was abandoned; they were just giving it some (revenue-enhancing) love.

by Sean Gallagher - May 27, 2015 3:30pm CDT
http://laredo-13.mit.edu/~brendan/malrec/movies/642a7a54-99cc-4756-98f9-ac81cdb912c2.mp4
• Super-powerful memory analysis framework

• Really needs no introduction ;)

• Important feature for our purposes: can parse kernel-mode objects that represent GUI
Administrator: Command Prompt

Windows Task Manager

Tab1

Processes

Show processes from all users

End Process
GUI View

Volatility View (wintree)

./vol.py -f mem.dd --profile=Win7SP1x86 wintree
GUI View

Volatility View (wintree)

```
./vol.py -f mem.dd --profile=Win7SP1x86 wintree
```

Windows Task Manager (visible) taskmgr.exe:1516 -

Users taskmgr.exe:1516 -

&Send Message... (visible) taskmgr.exe:1516 6.0.7601.17514!Button

&Logoff (visible) taskmgr.exe:1516 6.0.7601.17514!Button

&Disconnect (visible) taskmgr.exe:1516 6.0.7601.17514!Button

Users (visible) taskmgr.exe:1516 6.0.7601.17514!SysListView32

... #70038 (visible) taskmgr.exe:1516 6.0.7601.17514!SysHeader32

Networking taskmgr.exe:1516 -

... #40156 (visible) taskmgr.exe:1516 6.0.7601.17514!ScrollBar

... No Active Network Adapters Found. (visible) taskmgr.exe:1516 6.0.7601.17514!Static

... Totals (visible) taskmgr.exe:1516 6.0.7601.17514!SysListView32

... #201a8 (visible) taskmgr.exe:1516 6.0.7601.17514!SysHeader32

Performance taskmgr.exe:1516 -

... &Resource Monitor... (visible) taskmgr.exe:1516 6.0.7601.17514!Button

... Kernel Memory (MB) (visible) taskmgr.exe:1516 DavesFrameClass

... Physical Memory (MB) (visible) taskmgr.exe:1516 DavesFrameClass

... Tab1 (visible) taskmgr.exe:1516 6.0.7601.17514!SysTabControl32

... #50162 taskmgr.exe:1516 6.0.7601.17514!msctls_updown32

... Processes (visible) taskmgr.exe:1516 -

... &End Process (visible) taskmgr.exe:1516 6.0.7601.17514!Button

... Processes from all users (visible) taskmgr.exe:1516 6.0.7601.17514!Button

... #801a4 (visible) taskmgr.exe:1516 6.0.7601.17514!SysListView32

... Processes (visible) taskmgr.exe:1516 6.0.7601.17514!SysHeader32
```
What Do We Click On?

What Do We Click On?


...in what languages?
Implementation

• Build on work done by Bryan Payne and Tamas Lengyel in libvmi

• On the QEMU/PANDA side, spawns a thread that responds to memory access requests over a local socket

• Adds new address space to Volatility that has access to raw memory of PANDA
Implementation

- runmal.py
- Volatility (as library)
- panda_addrspace

- Emulated Keyboard
- Guest RAM
Implementation

runmal.py

\[\text{list\_windows()}\]

Volatility (as library)

panda_addrspace

Emulated Keyboard

Guest RAM
Implementation

runmal.py

Volatility
(as library)

panda_addrspace

Emulated Keyboard

Guest RAM

read memory 0x123450 4

local socket
Implementation

runmal.py

Volatility (as library)
panda_addrspace

list_windows()

read memory 0x123450 4

41 63 63 65 70 74

local socket

Emulated Keyboard

Guest RAM
Implementation

```
runmal.py

Volatility (as library)
panda_addrspace

mouse_move x y
mouse_button 1
QEMU Monitor

read memory 0x123450 4
41 63 63 65 70 74
local socket

Emulated Keyboard

Guest RAM
```
Implementation Details

• Because we are recording in PANDA, we have to be careful about how we access guest memory

  • Standard QEMU memory access functions are not thread-safe & interfere with recording

• Instead, locate raw buffer used to represent guest RAM by QEMU and memcpy from it
Complications

- Mouse acceleration
  - Disable it in the images
- Mouse movement coalescing
  - Insert delay between each movement
- How do we pick between multiple matching buttons?
  - Pick at random from the choices
http://panda.gtisc.gatech.edu/malrec/movies/576f7d4ac43b-4d30-9df3-8237135bc66f.mp4
Did We Improve Coverage?

• Hypothesis: average number of instructions executed per sample increased once we started clicking on buttons

• This is a testable scientific hypothesis!
Looking at the Data
Hypothesis Testing

• Doesn’t look like much visually, so we turn to a $t$-test

• We can do this easily using **scipy** and **pandas**:

```python
scipy.stats.ttest_ind(
    data['2015-04-28':]['count'],
    data['2015-04-27']['count']
)
```
Result

- The samples since 4/28 do have a higher mean number of instructions executed
- The difference is very close to significant ($p = .053$)
- …which is another way of saying “not significant”
...and this is where we put the non-significant results.

Credit: Matthew Hankins
https://mchankins.wordpress.com/2013/04/21/still-not-significant-2/
Why Not?

- Number of samples we have actually found buttons to click on is still fairly low (500 out of the last 3000)

- Not an apples to apples comparison

  - We would like to measure whether we improved coverage for GUI malware

  - This signal may be drowned out by the non-GUI samples

- More careful measurements are needed!
Limitations

- Not all GUI elements are managed by Windows kernel
Welcome to RadioHoops installation

The download and installation of RadioHoops is run by Trovi Download Manager.

Troví Download Manager is a stand-alone application that optimizes and manages the installation of your chosen software and, in the process, presents you with a number of additional offers. Any additional offers selected will be installed automatically.

Troví Download Manager does not endorse third-party software.

By clicking “Next”, you confirm that you have read and agree to Trovi’s Terms and Privacy Policy and, if applicable, authorize Trovi to access your account on Google Play™ store in order to install applications on your Android™ mobile devices. Google Play and Android are trademarks of Google Inc.
Limitations

• Not all GUI elements are managed by Windows kernel

• No support for other languages (Russian and Chinese are highest priority)
Limitations

- Not all GUI elements are managed by Windows kernel
- No support for other languages (Russian and Chinese are highest priority)
- Currently do not attempt to check for hidden / occluded windows (i.e. Alt-Tab)
Future Work

• Better coverage analysis – what does clicking on these buttons buy us? How much more of the program do we see?

• Can we find ways to see inside user-managed UI? OCR? Static analysis of message loop?

• More complex click patterns (“I agree” then “Next”; “Install”, then run the program)
Conclusion

• Combining Volatility & PANDA gives some powerful results!

• Code for malrec is open source: https://github.com/moyix/panda-malrec/

• GUI code not merged yet but will be posted soon