

Forum:[Hack Forums](#)

Thread: [\[v1.0.0.1\] RunPE Shield - detect processes created with a RunPE](#)

debug - December 1, 2015, 4:53 pm

Summary

I created this project to detect malware executed by RunPE's.

This is a work in progress, no where near its final stage. That being said, it's probably very buggy, so use with caution.

In Action

OldSpoiler [\(Click to View\)](#)

Notes

In the gif you can see that the first crypted.exe process is declared as clean, as it is not a process created by a runPE. But if it uses a runPE to create a new process, it will kill both processes.

TODOMove away from WMI; it's slow.

Monitor all processes, existing and new. Do so by scanning existing processes first and log their PID's. Run a constant thread to search for new processes by PID by referencing the PID list.

Clear closed processes from PID list every so often.

Handle processes protected by security access control.

Handle persistence.

Source

Github: <https://github.com/debug-hf/RunPE-Shield>

Change Log

v1.0.0.1Removed WMI method. Looping all processes now.

Scans all currently running 32-bit processes at the start, and continues to scan.

Improved speed of detection.

Remove old processes from active list

Added documentation.

Color coded log.

Credits

Almost all credits go to [BahNahNah](#) for his [PEToolKit](#).

I modified his code to work in this fashion, so shout out to him for most of the hard work.

If you find any bugs or have any suggestions let me know. I think this will be a fun project to work on.

Hope you like it!

-debug

beat you to it Lapan, suck it

Business Crypt - December 1, 2015, 5:29 pm

Hi DeBuG :^)

Glad to see you've moved onto something worthwhile.

So if I gather correctly - you're scanning for mismatching fields from the PE on disk and the PE in memory? I believe DarkCoderSC's tool does the same thing.

Nice work, it's a clever idea. I can think of at least a couple problems this could run into off the top of my head. Nonetheless, cool.

debug - December 1, 2015, 6:12 pm

(Dec 1, 2015 06:29 PM)Business Crypt Wrote: [\[CITE\]](#) Hi DeBuG :^)

Glad to see you've moved onto something worthwhile.

So if I gather correctly - you're scanning for mismatching fields from the PE on disk and the PE in memory? I believe DarkCoderSC's tool does the same thing.

Nice work, it's a clever idea. I can think of at least a couple problems this could run into off the top of my head. Nonetheless, cool. Correct. And If there are 5 or greater mismatches, it's flagged.

I will probably later turn this into more general Anti-Malware program, but for now, wiping out runpe's takes care of alot. ;)

ucsandaasu - December 2, 2015, 3:20 am

(Dec 1, 2015 05:53 PM)debug Wrote: [\[CITE\]](#) Summary

I created this project to detect malware executed by RunPE's.

This is a work in progress, no where near its final stage. That being said, it's probably very buggy, so use with caution.

In Action

Notes

In the gif you can see that the first crypted.exe process is declared as clean, as it is not a process created by a runPE. But if it uses a runPE to create a new process, it will kill both processes.

TODOMove away from WMI; it's slow.

Monitor all processes, existing and new. Do so by scanning existing processes first and log their PID's. Run a constant thread to search for new processes by PID by referencing the PID list.

Clear closed processes from PID list every so often.

Handle processes protected by security access control.

Handle persistence.

Source

Github: <https://github.com/debug-hf/RunPE-Shield>

Credits

Almost all credits go to [BahNahNah](#) for his [PEToolKit](#).

I modified his code to work in this fashion, so shout out to him for most of the hard work.

If you find any bugs or have any suggestions let me know. I think this will be a fun project to work on.

Hope you like it!
-debug

beat you to it Lapan, suck it
mate , this is insane! thank you alot and keep it up

Cecilia - December 2, 2015, 3:30 am

Eh yours is less advanced than mine, but you have done a good job in making it available to everyone ;)

log dot zip - December 2, 2015, 8:03 am

Finally someone who isn't showing off generic shit malware. Cool idea - reminds me of this demonstration, by Malwarebytes:

[Video: https://www.youtube.com/watch?v=s667v_TS-4A]

I'm guessing they're using the same basic flagging principle.

+rep for not being a shitter.

Only3 - December 2, 2015, 8:46 am

u can use async event

[https://msdn.microsoft.com/en-us/library/system.management.managemeventwatcher.stop\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/system.management.managemeventwatcher.stop(v=vs.100).aspx)

properties [https://msdn.microsoft.com/en-us/library/windows/desktop/aa394372\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa394372(v=vs.85).aspx)

or simply loop the process anyway theres better and simple ways to do it

debug - December 2, 2015, 8:53 am

(Dec 2, 2015 09:03 AM)codeconfusion Wrote: [\[CITE\]](#) Finally someone who isn't showing off generic shit malware. Cool idea - reminds me of this demonstration, by Malwarebytes:

[Video: https://www.youtube.com/watch?v=s667v_TS-4A]

I'm guessing they're using the same basic flagging principle.

+rep for not being a shitter.thanks!

(Dec 2, 2015 09:46 AM)Only3 Wrote: [\[CITE\]](#) u can use async event

[https://msdn.microsoft.com/en-us/library/system.management.managemeventwatcher.stop\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/system.management.managemeventwatcher.stop(v=vs.100).aspx)

properties [https://msdn.microsoft.com/en-us/library/windows/desktop/aa394372\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa394372(v=vs.85).aspx)

or simply loop the process anyway theres better and simple ways to do itYes, I plan to move away from WMI. See: todo.

Updated with significant improvements. See changelog.

Wardow - June 10, 2016, 9:41 am

Many ways to bypass this, I can leave the original PE loaded or, remap it to another place and make so the first module base address points to it, then create another row to the module list with the same info as the first one and use my new base address and patch the program base address in the PEB with that one too, this way your program checks the original application against the disk copy and does not detect anything

debug - June 10, 2016, 10:27 am

(Jun 10, 2016 09:41 AM)Wardow Wrote: [\[CITE\]](#) Many ways to bypass this, I can leave the original PE loaded or, remap it to another place and make so the first module base address points to it, then create another row to the module list with the same info as the first one and use my new base address and patch the program base address in the PEB with that one too, this way your program checks the original application against the disk copy and does not detect anything I'm sure you can bypass it Wardow, but for 99% of the malware skiddies, this will work just fine.

The-One - June 10, 2016, 11:42 am

(Jun 10, 2016 10:27 AM)debug Wrote: [\[CITE\]](#) I'm sure you can bypass it Wardow, but for 99% of the malware skiddies, this will work just fine.

In C# one can easily install EasyHook and write a simple hook for ReadProcessMemory to redirect your calls to ReadProcessMemory and return invalid information.

Wardow - June 11, 2016, 4:39 am

(Jun 10, 2016 11:42 AM)The-One Wrote: [\[CITE\]](#) In C# one can easily install EasyHook and write a simple hook for ReadProcessMemory to redirect your calls to ReadProcessMemory and return invalid information.

Not the subject, nor the point, anyone can get over anything using hooking, it's all about integrity level, if you have a driver you can even do that on Anti-Viruses, again, not the point

The-One - June 13, 2016, 2:24 am

(Jun 11, 2016 04:39 AM)Wardow Wrote: [\[CITE\]](#) Not the subject, nor the point, anyone can get over anything using hooking, it's all about integrity level, if you have a driver you can even do that on Anti-Viruses, again, not the point

But that is the point I wanted to make; I tried to explain that this tool could easily be fooled and one shouldn't rely on it. Also, what's wrong with hooking? Why should I want to write something that remaps my assembly or messes with the PEB while I can easily hook rpm?

Wardow - June 13, 2016, 6:27 am

(Jun 13, 2016 02:24 AM)The-One Wrote: [\[CITE\]](#) But that is the point I wanted to make; I tried to explain that this tool could easily be fooled and one shouldn't rely on it. Also, what's wrong with hooking? Why should I want to write something that remaps my assembly or messes with the PEB while I can easily hook rpm?

"Hooking" is universal,

When hooking is only way to bypass you can consider it secure; as is, integrity is what will stop your program from operating, so get it higher when you run it.

The-One - June 13, 2016, 7:20 am

(Jun 13, 2016 06:27 AM)Wardow Wrote: [\[CITE\]](#) "Hooking" is universal,

When hooking is only way to bypass you can consider it secure; as is, integrity is what will stop your program from operating, so get it higher when you run it.

Okay, I get that hooking is universal and that doesn't affect the integrity but still; why would I write loads of code when I could apply a simple rpm hook, just out of curiosity?

Wardow - June 13, 2016, 8:37 am

(Jun 13, 2016 07:20 AM)The-One Wrote: [\[CITE\]](#) Okay, I get that hooking is universal and that doesn't affect the integrity but still; why would I write loads of code when I could apply a simple rpm hook, just out of curiosity?

Actually my solution is less code..

Cecilia - June 14, 2016, 2:36 am

Can you take your bickering to another section (users above me)?

It's kinda annoying after a while.

I don't care whether you hook x y or z. This is a nice piece of software. Way better than AV's for the most part on crypted apps.

Nick (Name) - March 21, 2017, 5:25 pm

Hey debug, I was recently looking at this project and was wanting it to be compiled for .NET 3.5, I fixed the integer pointer issue and made this available for the lower framework. Thought I show you what I did to downgrade

Modified your vLoad method in PEloader.cs:

Spoiler ([Click to View](#))

To a .NET 3.5 working version:

Spoiler ([Click to View](#))

zalm - March 24, 2017, 9:25 am

(Jun 14, 2016 02:36 AM)LaPanthere Wrote: ([CITE](#)) Can you take your bickering to another section (users above me)?

It's kinda annoying after a while.

I don't care whether you hook x y or z. This is a nice piece of software. Way better than AV's for the most part on crypted apps.

How is it anywhere near as good as AV's? Legit (packed/protected) apps also change PE headers at runtime

w4zt3d - April 24, 2017, 4:16 am

Did some one bypass it just for fun?

stefsot - April 25, 2017, 6:25 am

(Apr 24, 2017 04:16 AM)w4zt3d Wrote: ([CITE](#)) Did some one bypass it just for fun?
just patch the PE header

EZLib - April 29, 2017, 2:25 pm

Thanks debug for this, sad to see Net Seal go ugh.