

Forum:[Hack Forums](#)

Thread: [What happened to Process Hollowing?](#)

Phantom_Services - March 2, 2025, 3:42 am

Hello everyone,

I recently discovered some old process hollowing code that I developed a few years ago. I tested it, but it no longer works. I also checked on GitHub and encountered the same issue. Does anyone know what exactly has changed in Windows or why it doesn't work anymore?

Phantom_Services - March 2, 2025, 4:17 am

(Mar 2, 2025 05:13 AM)Lowkey Wrote: [\[CITE\]](#) Do you have the project files or any specific line item code errors you are receiving?

Wouldn't mind taking a look with/for you to see if I can help!
Tested on win11 btw.

0x00000004
0x00000005
0x0000141

these mostly. on every script i tested.

Here is a .Net runpe x64 where i get the error:

<https://paste.ee/p/0gLeYNv7>

Phantom_Services - March 2, 2025, 4:26 am

(Mar 2, 2025 05:25 AM)Lowkey Wrote: [\[CITE\]](#) Alright so

For 0x00000005 (Permission Issues)

- You might need to run your program as administrator since you're modifying memory, injecting into a process, and using VirtualAllocEx and WriteProcessMemory functions.

For 0x00000141 (Issues with DLL Loading)

- If kernel32.dll or ntdll.dll fails to load properly, LoadLibrary may return an invalid handle.
- So basically make sure that the DLLs are present and accessible.
- Instead of storing kernel32Handle and ntdllHandle as static members, try calling LoadLibrary in LoadFunctions() to ensure they are properly loaded.

You also have an incorrect function delegate(s)

The CreateProcessDelegate is incorrect:

It should be

CreateProcessA uses STARTUPINFO and PROCESS_INFORMATION, not byte arrays.

The function signature must match exactly.

Thanks will check that

Phantom_Services - March 2, 2025, 4:39 am

(Mar 2, 2025 05:35 AM)Incog Wrote: [\[CITE\]](#) The issue is due to the new windows update (11 24H2)

Classic runPE uses MEM_PRIVATE instead of normal processes using MEM_IMAGE

<https://windowsforum.com/threads/windows-11-24h2-breaking-process-hollowing-and-malware-defense.351197/>

https://github.com/hasherezade/transacted_hollowing

Here is an example of a runPE that circumvents this, you can also patch ntdll however that can lead to issues with stability.

Uff, is there another good alternative?

boomboom99 - March 2, 2025, 9:31 pm

Amazing how crypt services are only now starting to understand what started happening 3 months ago. Just shows you the quality of the crypts that are sold here.

There is a much easier way to bypass it without needing to change any "runpe" original code.. but of course you expect people to just give you the solution.

Open a debugger.. Use your brain. Where does the Hotpatch check comes from? Protip: Not in kernel. So you can simply remove it.

Phantom_Services - March 3, 2025, 12:14 am

(Mar 2, 2025 10:31 PM)boomboom99 Wrote: [\[CITE\]](#) Amazing how crypt services are only now starting to understand what started happening 3 months ago. Just shows you the quality of the crypts that are sold here.

There is a much easier way to bypass it without needing to change any "runpe" original code.. but of course you expect people to just give you the solution.

Open a debugger.. Use your brain. Where does the Hotpatch check comes from? Protip: Not in kernel. So you can simply remove it.

You know my Crypt works perfectly on win11 with the Update, because i dont use process hollowing :)

But thanks, will take a look

boomboom99 - March 3, 2025, 3:04 am

(Mar 3, 2025 01:14 AM)Phantom_Services Wrote: [\[CITE\]](#) You know my Crypt works perfectly on win11 with the Update, because i dont use process hollowing :)

But thanks, will take a lookFirst of all the issue is not only "Process hollowing" But merely anyone who "Allocates" MEMORY in a remote suspended process.

If you are so good, why open a thread? You can't open a debugger? Find the issue yourself? Or is your only way of fixing things is by being spoon fed with information? Is that only way you know? People forgot how to use their brain.

People who are good, don't need to open a thread begging for information. Minimum you could of done is google searching... You didn't even do the minimum.

But okay "Mr CRYPTER 1337"

ServersFxcked - March 3, 2025, 11:14 pm

(Mar 3, 2025 04:04 AM)boomboom99 Wrote: [\[CITE\]](#) First of all the issue is not only "Process hollowing" But merely anyone who "Allocates" MEMORY in a remote suspended process.

If you are so good, why open a thread? You can't open a debugger? Find the issue yourself? Or is your only way of fixing things is by being spoon fed with information? Is that only way you know? People forgot how to use their brain.

People who are good, don't need to open a thread begging for information. Minimum you could of done is google searching... You didn't even do the minimum.

But okay "Mr CRYPTER 1337"

Mate his in a forum for god sake his allowed to ask hahah, damn we got Fun Police over here

Phantom_Services - March 3, 2025, 11:32 pm

(Mar 4, 2025 12:14 AM)ServersFxcked Wrote: [\[CITE\]](#) Mate his in a forum for god sake his allowed to ask hahah, damn we got Fun Police over here
Fr

boomboom99 - March 4, 2025, 12:19 am

(Mar 4, 2025 12:14 AM)ServersFxcked Wrote: [\[CITE\]](#) Mate his in a forum for god sake his allowed to ask hahah, damn we got Fun Police over here

It is just funny to me that a CRYPT owner that demands 100\$ a crypt asks a question that he can answers himself with a debugger in under 2 minutes. Shows you the quality of the crypters that are sold here.

Copy pasted garbage.

Phantom_Services - March 4, 2025, 1:41 am

(Mar 4, 2025 01:19 AM)boomboom99 Wrote: [\[CITE\]](#) It is just funny to me that a CRYPT owner that demands 100\$ a crypt asks a question that he can answers himself with a debugger in under 2 minutes. Shows you the quality of the crypters that are sold here.

Copy pasted garbage.

When i would copy paste stuff, i wouldnt have 0-2 Detections Runtime :)) Just use Loadpe for a long time now and was interested what microsoft did. When you have a problem with me asking questions, ignore them. Thank you

loomixx - March 4, 2025, 5:01 am

(Mar 2, 2025 04:42 AM)Phantom_Services Wrote: [\[CITE\]](#) Hello everyone,

I recently discovered some old process hollowing code that I developed a few years ago. I tested it, but it no longer works. I also checked on GitHub and encountered the same issue. Does anyone know what exactly has changed in Windows or why it doesn't work anymore?

If you have this issue on Win11 24H2, you have to patch RtlpInsertOrRemoveScpCfgFunctionTable function (ntdll.dll) with this:

```
const BYTE Patch[4] =  
{  
0x48, 0x31, 0xC0, // xor rax, rax  
0xC3 // ret  
};
```

boomboom99 - March 4, 2025, 9:01 am

(Mar 4, 2025 06:01 AM)loomixx Wrote: [\[CITE\]](#) If you have this issue on Win11 24H2, you have to patch RtlpInsertOrRemoveScpCfgFunctionTable function (ntdll.dll) with this:

```
const BYTE Patch[4] =
```

```
{  
0x48, 0x31, 0xC0, // xor rax, rax  
0xC3 // ret  
};
```

RtlpInsertOrRemoveScpCfgFunctionTable IS not exported so this comment is useless.

loomixx - March 4, 2025, 10:00 am

(Mar 4, 2025 10:01 AM)boomboom99 Wrote: [\[CITE\]](#) RtlpInsertOrRemoveScpCfgFunctionTable IS not exported so this comment is useless.

wdym lol

boomboom99 - March 4, 2025, 11:10 am

(Mar 4, 2025 11:00 AM)loomixx Wrote: [\[CITE\]](#) wdym lol

It means u can't find it by walking the export table like you can find any normal NT function because it is not exported.

So it is not a reliable way.

loomixx - March 4, 2025, 11:40 am

(Mar 4, 2025 12:10 PM)boomboom99 Wrote: [\[CITE\]](#) It means u can't find it by walking the export table like you can find any normal NT function because it is not exported.

So it is not a reliable way.

Oh, of course, you must know better. I'm sure you thoroughly tested this on the fresh Windows 11 24H2 machine - naturally after creating a suspended process, ensuring the function is mapped into memory before patching it, right? Didn't you? Because I actually did, and, surprise, it works exactly as intended. Next time, please try verifying something before making useless statements.

boomboom99 - March 4, 2025, 12:01 pm

(Mar 4, 2025 12:40 PM)loomixx Wrote: [\[CITE\]](#) Oh, of course, you must know better. I'm sure you thoroughly tested this on the fresh Windows 11 24H2 machine - naturally after creating a suspended process, ensuring the function is mapped into memory before patching it, right? Didn't you? Because I actually did, and, surprise, it works exactly as intended. Next time, please try verifying something before making useless statements.

Rtlp Functions are not exported, what are you fighting about? Showing a messagebox proves what exactly? This is not a reliable way. You need to byte search for it or patch the inner ZwQueryVirtualMemory calls.

Don't talk unless you walk. Copy and pasting code from git will get you no-where.

loomixx - March 4, 2025, 12:16 pm

(Mar 4, 2025 01:01 PM)boomboom99 Wrote: [\[CITE\]](#) Rtlp Functions are not exported, what are you fighting about? Showing a messagebox proves what exactly? This is not a reliable way. You need to byte search for it or patch the inner ZwQueryVirtualMemory calls.

Don't talk unless you walk. Copy and pasting code from git will get you no-where.

Yes, RtlpInsertOrRemoveScpCfgFunctionTable isn't exported, but that doesn't mean it's inaccessible. Once ntdll.dll is mapped into the suspended process, the function exists in memory

and can be patched directly - no need for export resolution or byte searching. That's why this method works.

As for ZwQueryVirtualMemory, sure, you could patch the calls inside, but why overcomplicate it when disabling the function entirely achieves the same result?

boomboom99 - March 4, 2025, 12:21 pm

(Mar 4, 2025 01:16 PM)loomixx Wrote: [\[CITE\]](#) Yes, RtlpInsertOrRemoveScpCfgFunctionTable isn't exported, but that doesn't mean it's inaccessible. Once ntdll.dll is mapped into the suspended process, the function exists in memory and can be patched directly - no need for export resolution or byte searching. That's why this method works.

As for ZwQueryVirtualMemory, sure, you could patch the calls inside, but why overcomplicate it when disabling the function entirely achieves the same result?

????????????????????

"the function exists in memory and can be patched directly - no need for export resolution or byte searching. That's why this method works."

????????????

In order to PATCH a function YOU FIRST NEED TO FIND IT. IF IT IS NOT EXPORTED, HOW DO YOU PLAN TO DO THAT? Spitting useless garbage you found in the internet does not make you smarter.

loomixx - March 4, 2025, 12:44 pm

(Mar 4, 2025 01:21 PM)boomboom99 Wrote: [\[CITE\]](#) ??????????????????????

"the function exists in memory and can be patched directly - no need for export resolution or byte searching. That's why this method works."

????????????

In order to PATCH a function YOU FIRST NEED TO FIND IT. IF IT IS NOT EXPORTED, HOW DO YOU PLAN TO DO THAT? Spitting useless garbage you found in the internet does not make you smarter.

The function resides at a fixed offset inside ntdll.dll for a given Windows version. Since ntdll.dll is mapped at a predictable base in the suspended process, the function can be reliably patched using a known RVA (NtdllBase + 0x7BE0 in this case).

daveso666 - March 16, 2025, 9:23 am

There is a reliable way that involves only patching the functions that are exported. It has been described in this blog:

[hxxps://hshrzd.wordpress.com/2025/01/27/process-hollowing-on-windows-11-24h2/](https://hshrzd.wordpress.com/2025/01/27/process-hollowing-on-windows-11-24h2/) and the linked thread: [hxxps://github.com/hasherezade/libpeconv/issues/59](https://github.com/hasherezade/libpeconv/issues/59)

There are two functions to patch: ZwQueryVirtualMemory and NtManageHotPatch:

[hxxps://github.com/hasherezade/libpeconv/blob/master/run_pe/patch_ntdll.cpp](https://github.com/hasherezade/libpeconv/blob/master/run_pe/patch_ntdll.cpp)

sooGg - August 25, 2025, 3:21 am

(Mar 16, 2025 09:23 AM)daveso666 Wrote: [\[CITE\]](#) There is a reliable way that involves only patching the functions that are exported. It has been described in this blog:

[hxxps://hshrzd.wordpress.com/2025/01/27/process-hollowing-on-windows-11-24h2/](https://hshrzd.wordpress.com/2025/01/27/process-hollowing-on-windows-11-24h2/) and the linked thread: [hxxps://github.com/hasherezade/libpeconv/issues/59](https://github.com/hasherezade/libpeconv/issues/59)

There are two functions to patch: ZwQueryVirtualMemory and NtManageHotPatch:

[hxxps://github.com/hasherezade/libpeconv/blob/master/run_pe/patch_ntdll.cpp](https://github.com/hasherezade/libpeconv/blob/master/run_pe/patch_ntdll.cpp)

Transacted Hollowing or Process Overwriting, which use proper MEM_IMAGE mapping or disable CFG (Control Flow Guard), still work on Win11 24H2!

Cryptify - August 25, 2025, 5:37 am

windows 24h2 patched x64 bit injection but obviously it can be fixed with some tweaks

ItsMehRAWR - September 1, 2025, 7:14 pm

RtlpInsertOrRemoveScpCfgFunctionTable now calls ZwQueryVirtualMemory

MemoryImageExtensionInformation only accepts MEM_IMAGE sections

Traditional hollowing uses MEM_PRIVATE -> FAILS with 0xC0000141

EliteCipher21 - November 13, 2025, 3:17 pm

I read that recent versions of Windows Block Process Hollowing techniques. IDK if thats true.