

Forum:[Hack Forums](#)

Thread: [JWorm: An Open-Source Java Worm](#)

nullifiedvoid - June 19, 2015, 10:39 pm

Hello Forum,

If any of you are interested in Java bytecode injection, [here](#) is a link to an open-source Java Worm. It uses bytecode injection to infect other files with its replication method, without affecting the functionality of the target program.

Best Regards,
NullifiedVoid

JavaProfit - June 19, 2015, 11:05 pm

When you showed your browser bot I checked this project out and it looks pretty cool. Nice job working on it.

MrOrang - June 20, 2015, 1:04 am

Why are you extending to Object? I really do not get that.

Anyways I made one once and mine just added all class files from one jar into another one and changed the main method to call the other main.

JavaProphet - June 20, 2015, 1:28 am

Looks nice.

You should also look into Java Agents, as they are very powerful. Especially on windows, where I believe you can privilege escalate if a Java daemon is running as an Admin. On Linux, however, they are secure.

+Immediate execution

Injecting into the jar file is a good long term solution though.

JavaProfit - June 20, 2015, 1:28 am

(Jun 20, 2015 01:04 AM)Looka Wrote: [\[CITE\]](#) Why are you extending to Object? I really do not get that.

Anyways I made one once and mine just added all class files from one jar into another one and changed the main method to call the other main.

I don't really understand why he is using injection as writing in extra class files works perfectly well, but of course they both work. Maybe it's less suspicious if no extra classes are added?

nullifiedvoid - June 20, 2015, 4:24 am

(Jun 20, 2015 01:04 AM)Looka Wrote: [\[CITE\]](#) Why are you extending to Object? I really do not get that.

Anyways I made one once and mine just added all class files from one jar into another one and changed the main method to call the other main.

Well even if it were not specified, when the compiler compiles the source if no extending class was placed, then it will automatically extend Object. I guess this is done because it makes the class look better graphically? Is it pointless? Of course. Lol.

(Jun 20, 2015 01:28 AM)Systemic Wrote: [\[CITE\]](#) I don't really understand why he is using injection as writing in extra class files works perfectly well, but of course they both work. Maybe it's less suspicious if no extra classes are added?

Yes sir, that is exactly why. To make it look less suspicious.

I used this same technique for bukkit plugins a while back and I learned that some devs open the jar files up in winrar to see if anything "odd" appears in them.

(Jun 20, 2015 01:28 AM)JavaProphet Wrote: [\[CITE\]](#) Looks nice.

You should also look into Java Agents, as they are very powerful. Especially on windows, where I believe you can privilege escalate if a Java daemon is running as an Admin. On Linux, however, they are secure.

+Immediate execution

Injecting into the jar file is a good long term solution though.

Will do, thank you for the information.

EDIT: You mean Java Instrumentation?

JavaProfit - June 20, 2015, 10:39 am

(Jun 20, 2015 01:28 AM)JavaProphet Wrote: [\[CITE\]](#) Looks nice.

You should also look into Java Agents, as they are very powerful. Especially on windows, where I believe you can privilege escalate if a Java daemon is running as an Admin. On Linux, however, they are secure.

+Immediate execution

Injecting into the jar file is a good long term solution though.

Why would he need to use a Java agent instead of what he already has? Java agents just hook at "pre runtime" (technically runtime of course). I suppose the only real advantage to this would be if you somehow were able to make all jars run with -javaagent.

MalikDz - June 20, 2015, 12:14 pm

(Jun 20, 2015 10:39 AM)Systemic Wrote: [\[CITE\]](#) Why would he need to use a Java agent instead of what he already has? Java agents just hook at "pre runtime" (technically runtime of course). I suppose the only real advantage to this would be if you somehow were able to make all jars run with -javaagent. Look into the attach api made by oracle , you can even use it to attach native agents.

JavaProphet - June 20, 2015, 12:27 pm

Yes, you can attach at runtime with oracle's API.

What's interesting, is I think it has no way to check permissions on windows, or I am mistaken, privilege escalation is possible.

JavaProfit - June 20, 2015, 12:34 pm

(Jun 20, 2015 12:14 PM)MalikDz Wrote: [\[CITE\]](#) Look into the attach api made by oracle , you can even use it to attach native agents.

I've used that API before actually in a private MC client. Thanks for informing me that native agents work too.

Offtopic: Also, haven't seen you in many months. Good luck on your RS botting

Sakawa - June 20, 2015, 1:42 pm

I don't get this. Where the hell is the source? All I see is a .jar file.

badvibes - June 20, 2015, 2:12 pm

(Jun 20, 2015 01:42 PM)Sakawa Wrote: [\[CITE\]](#) I don't get this. Where the hell is the source? All I see

is a .jar file.

<https://github.com/DMoneigh/JWorm/tree/master/src>

press the com/jworm

christ...

JavaProfit - June 20, 2015, 3:06 pm

(Jun 20, 2015 02:12 PM)badvibes Wrote: [\[CITE\]](#) <https://github.com/DMoneigh/JWorm/tree/master/src>

press the com/jworm

christ...

its okay not everyone is good at computers.

Sakawa - June 21, 2015, 8:13 am

lol. I didn't see that folder.

I'm good at computers and use git without any problem, but I never used GitHub :\

btw, the code seems interesting

JavaProfit - June 21, 2015, 11:43 am

(Jun 21, 2015 08:13 AM)Sakawa Wrote: [\[CITE\]](#) lol. I didn't see that folder.

I'm good at computers and use git without any problem, but I never used GitHub :\

btw, the code seems interesting

haha it's fine i was joking.