
Subject: Security risks in ve_allow_kthreads

Posted by [Jakob Goldbach](#) on Tue, 08 Jan 2008 19:19:40 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

What security risks do I impose on myself when enabling kernel threads inside the VE ?

/Jakob

Subject: Re: Security risks in ve_allow_kthreads

Posted by [vaverin](#) on Wed, 09 Jan 2008 07:27:27 GMT

[View Forum Message](#) <> [Reply to Message](#)

When VE is stopped all processes should be finished. Userspace processes can be killed, but kernel threads cannot be stopped by this way.

You should have some guarantee that your kernel threads will be stopped by some way when VE is stopped. Otherwise you will have at least memory leakage.

thank you,
Vasily Averin

Subject: Re: Re: Security risks in ve_allow_kthreads

Posted by [Jakob Goldbach](#) on Wed, 09 Jan 2008 07:39:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Wed, 2008-01-09 at 10:27 +0300, vaverin wrote:

>

> When VE is stopped all processes should be finished. Userspace processes can be killed, but kernel threads cannot be stopped by this way.

Okay.

> You should have some guarantee that your kernel threads will be stopped by some way when VE is stopped. Otherwise you will have at least memory leakage.

>

My kernel threads is from the Lustre filesystem which I mount inside the VE after the VE has started. Unmounting this should stop the thread so I should be in the clear.

Thanks
/Jakob

Subject: Re: Security risks in ve_allow_kthreads
Posted by [dev](#) on Wed, 09 Jan 2008 07:52:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

Jakob Goldbach wrote:

> Hi,

>

> What security risks do I impose on myself when enabling kernel threads

> inside the VE ?

1. if kernel thread doesn't terminate on VE stop - VE stop will be blocked.
2. security implications can be that kernel threads usually can do things which user space applications can't. So security implications depend on what thread in question does.
3. if your system is quite trusted (2) is not an issue at all.
only (1) must be concerned.

Kirill
