[v3] selinux: restrict kernel module loading

Archived: 2026-04-05 21:24:32 UTC

Message ID	1459886787-19858-1-git-send-email-jeffv@google.com (mailing list archive)
State	Accepted
Headers	

Commit Message

```
  Utilize existing kernel_read_file hook on kernel module load.
Add module_load permission to the system class.

Enforces restrictions on kernel module origin when calling the
finit_module syscall. The hook checks that source type has
permission module_load for the target type.
Example for finit_module:

allow foo bar_file:system module_load;

Similarly restrictions are enforced on kernel module loading when
calling the init_module syscall. The hook checks that source
type has permission module_load with itself as the target object
because the kernel module is sourced from the calling process.
Example for init_module:

allow foo foo:system module_load;

Signed-off-by: Jeff Vander Stoep <jeffv@google.com>
---
v2: The target type for init_module changed from SECINITSID_KERNEL
to the same type as the source.
v3: Use inode_security() to ensure inode's label is revalidated.

 security/selinux/hooks.c            | 46 ++++++++++++++++++++++++++++++++++++
 security/selinux/include/classmap.h |  2 +-
 2 files changed, 47 insertions(+), 1 deletion(-)
```

Comments

On Tuesday, April 05, 2016 01:06:27 PM Jeff Vander Stoep wrote:
> Utilize existing kernel_read_file hook on kernel module load.
> Add module_load permission to the system class.
>
> Enforces restrictions on kernel module origin when calling the
> finit_module syscall. The hook checks that source type has
> permission module_load for the target type.
> Example for finit_module:
>
> allow foo bar_file:system module_load;
>
> Similarly restrictions are enforced on kernel module loading when
> calling the init_module syscall. The hook checks that source
> type has permission module_load with itself as the target object
> because the kernel module is sourced from the calling process.
> Example for init_module:
>
> allow foo foo:system module_load;
>
> Signed-off-by: Jeff Vander Stoep <jeffv@google.com>
> ---
> v2: The target type for init_module changed from SECINITSID_KERNEL
> to the same type as the source.
> v3: Use inode_security() to ensure inode's label is revalidated.

Merged, thanks for your patience. I had to do one minor fixup to resolve a
problem at compile time, see below.

> diff --git a/security/selinux/hooks.c b/security/selinux/hooks.c
> index 3fa3ca5..231c897 100644
> --- a/security/selinux/hooks.c
> +++ b/security/selinux/hooks.c

...

> +static selinux_kernel_read_file(struct file *file, enum kernel_read_file_id
> id)

You're missing the return type :) No need to resend, I fixed it when merging
your patch, see the selinux#next branch.

>
> You're missing the return type :) No need to resend, I fixed it when
> merging
> your patch, see the selinux#next branch.

>

Thanks for catching that.

Hello Jeff,

We are a Wireless Consulting Firm conducting research for a U.S. Health and
Human Services Grant. We require Secure Mobile Devices. How much is a
license for Google Android SELinux and where do i go for the config?

Dennis Sherrell
Sherrell Consulting
Company #136601
Wireless Security Consultant
Cisco Certified Wireless Specialist
DISA Mobile Device Administartor

On Tue, Apr 5, 2016, 1:33 PM Jeffrey Vander Stoep <jeffv@google.com> wrote:

> You're missing the return type :) No need to resend, I fixed it when
>> merging
>> your patch, see the selinux#next branch.
>>
>
> Thanks for catching that.
>
> _____
> Selinux mailing list
> Selinux@tycho.nsa.gov
> To unsubscribe, send email to Selinux-leave@tycho.nsa.gov.
> To get help, send an email containing "help" to
> Selinux-request@tycho.nsa.gov.

On Apr 6, 2016 03:01, "Dennis Sherrell" <sherrellconsulting@gmail.com>
wrote:
>
> Hello Jeff,
>
> We are a Wireless Consulting Firm conducting research for a U.S. Health
and Human Services Grant. We require Secure Mobile Devices. How much is a
license for Google Android SELinux and where do i go for the config?

It's all open source in the aosp (Android Open Source Project) project.

There is essentially two paths you can go, Android branding and non Android branding. An example of non-Android branded devices would be Amazon's Kindle line or the Silent circle black phone. Also there are popular aftermarket software ROMs based on aosp, like Cyanogenmod.

If you want branding, then you have to go through Google since they own the brand, they have various programs for that. This page might help provide more detail: https://source.android.com/compatibility/index.html

The only parts that are generally not available in the open are the proprietary drivers that bridge Android to the hardware.

You can download aosp at https://source.android.com/source/downloading.html

>
> Dennis Sherrell
> Sherrell Consulting
> Company #136601
> Wireless Security Consultant
> Cisco Certified Wireless Specialist
> DISA Mobile Device Administartor
>
>
> On Tue, Apr 5, 2016, 1:33 PM Jeffrey Vander Stoep <jeffv@google.com>
wrote:
>>>
>>> You're missing the return type :) No need to resend, I fixed it when
merging
>>> your patch, see the selinux#next branch.
>>
>>
>> Thanks for catching that.
>>
>> _____
>> Selinux mailing list
>> Selinux@tycho.nsa.gov
>> To unsubscribe, send email to Selinux-leave@tycho.nsa.gov.
>> To get help, send an email containing "help" to
Selinux-request@tycho.nsa.gov.
>
>
> _____
> Selinux mailing list
> Selinux@tycho.nsa.gov
> To unsubscribe, send email to Selinux-leave@tycho.nsa.gov.
> To get help, send an email containing "help" to

Selinux-request@tycho.nsa.gov.

On Apr 6, 2016 5:42 AM, "William Roberts" <bill.c.roberts@gmail.com> wrote:
>
>
> On Apr 6, 2016 03:01, "Dennis Sherrell" <sherrellconsulting@gmail.com>
wrote:
> >
> > Hello Jeff,
> >
> > We are a Wireless Consulting Firm conducting research for a U.S. Health
and Human Services Grant. We require Secure Mobile Devices. How much is a
license for Google Android SELinux and where do i go for the config?
>
> It's all open source in the aosp (Android Open Source Project) project.
There is essentially two paths you can go, Android branding and non Android
branding. An example of non-Android branded devices would be Amazon's
Kindle line or the Silent circle black phone. Also there are popular
aftermarket software ROMs based on aosp, like Cyanogenmod.
>
> If you want branding, then you have to go through Google since they own
the brand, they have various programs for that. This page might help
provide more detail: https://source.android.com/compatibility/index.html
>
> The only parts that are generally not available in the open are the
proprietary drivers that bridge Android to the hardware.
>
> You can download aosp at
https://source.android.com/source/downloading.html
>

FYI This question is off topic to the thread and mailing list. In the
future post a new topic to the seandroid mailing list.

> >
> > Dennis Sherrell
> > Sherrell Consulting
> > Company #136601
> > Wireless Security Consultant
> > Cisco Certified Wireless Specialist
> > DISA Mobile Device Administartor
> >
> >
> > On Tue, Apr 5, 2016, 1:33 PM Jeffrey Vander Stoep <jeffv@google.com>
wrote:

> >>>
> >>> You're missing the return type :) No need to resend, I fixed it when
merging
> >>> your patch, see the selinux#next branch.
> >>
> >>
> >> Thanks for catching that.
> >>
> >> _____
> >> Selinux mailing list
> >> Selinux@tycho.nsa.gov
> >> To unsubscribe, send email to Selinux-leave@tycho.nsa.gov.
> >> To get help, send an email containing "help" to
Selinux-request@tycho.nsa.gov.
> >
> >
> > _____
> > Selinux mailing list
> > Selinux@tycho.nsa.gov
> > To unsubscribe, send email to Selinux-leave@tycho.nsa.gov.
> > To get help, send an email containing "help" to
Selinux-request@tycho.nsa.gov.

```
@@ -3719,6 +3719,51 @@  static int selinux_kernel_module_request(char *kmod_name)
                            SYSTEM__MODULE_REQUEST, &ad);
 }

+static int selinux_kernel_module_from_file(struct file *file)
+{
+       struct common_audit_data ad;
+       struct inode_security_struct *isec;
+       struct file_security_struct *fsec;
+       u32 sid = current_sid();
+       int rc;
+
+       /* init_module */
+       if (file == NULL)
+               return avc_has_perm(sid, sid, SECCLASS_SYSTEM,
+                                   SYSTEM__MODULE_LOAD, NULL);
+
+       /* finit_module */
+       ad.type = LSM_AUDIT_DATA_PATH;
```

```
+	ad.u.path = file->f_path;
+
+	isec = inode_security(file_inode(file));
+	fsec = file->f_security;
+
+	if (sid != fsec->sid) {
+		rc = avc_has_perm(sid, fsec->sid, SECCLASS_FD, FD__USE, &ad);
+		if (rc)
+			return rc;
+	}
+
+	return avc_has_perm(sid, isec->sid, SECCLASS_SYSTEM,
+				SYSTEM__MODULE_LOAD, &ad);
+}
+
+static selinux_kernel_read_file(struct file *file, enum kernel_read_file_id id)
+{
+	int rc = 0;
+
+	switch (id) {
+	case READING_MODULE:
+		rc = selinux_kernel_module_from_file(file);
+		break;
+	default:
+		break;
+	}
+
+	return rc;
+}
+
 static int selinux_task_setpgid(struct task_struct *p, pid_t pgid)
 {
 	return current_has_perm(p, PROCESS__SETPGID);
@@ -6022,6 +6067,7 @@  static struct security_hook_list selinux_hooks[] = {
 	LSM_HOOK_INIT(kernel_act_as, selinux_kernel_act_as),
 	LSM_HOOK_INIT(kernel_create_files_as, selinux_kernel_create_files_as),
 	LSM_HOOK_INIT(kernel_module_request, selinux_kernel_module_request),
+	LSM_HOOK_INIT(kernel_read_file, selinux_kernel_read_file),
 	LSM_HOOK_INIT(task_setpgid, selinux_task_setpgid),
 	LSM_HOOK_INIT(task_getpgid, selinux_task_getpgid),
 	LSM_HOOK_INIT(task_getsid, selinux_task_getsid),




@@ -32,7 +32,7 @@  struct security_class_mapping secclass_map[] = {
 	    "setsockcreate", NULL } },
```

```
        { "system",
          { "ipc_info", "syslog_read", "syslog_mod",
-           "syslog_console", "module_request", NULL } },
+           "syslog_console", "module_request", "module_load", NULL } },
        { "capability",
          { "chown", "dac_override", "dac_read_search",
            "fowner", "fsetid", "kill", "setgid", "setuid", "setpcap",
```

Source: https://patchwork.kernel.org/patch/8754821/