Integrity Verification of a Trusted Computing Base for the SELinux Example Policy

Trent Jaeger Reiner Sailer Xiaolan Zhang
IBM T. J. Watson Research Center
Hawthorne, NY 10532 USA
Email: {jaegert,sailer,cxzhang}@us.ibm.com

February 27, 2003

1 Integrity Analysis

The idea is the we choose a set of subject types to belong to a the trusted computing base (TCB) of the SELinux example policy and determine whether the integrity of these TCB subject types is protected by the SELinux example policy. The discussion below is for the SELinux example policy for 2.4.19.

As an initial estimate of integrity protection, we use the Biba integrity policy. The basis of Biba integrity is that high integrity processes may not use (e.g., read, execute) low integrity data. Thus, high integrity processes (i.e., our TCB subject types) may not use integrity data (i.e., data written by other subject types).

Since this is a pretty restrictive policy, we expect conflicts and hope to find means to resolve them. Options include: (1) excluding the subject type from the system; (2) exclude the object type from the system; (3) find some means to believe that the low integrity data can be sanitized; (4) assume that conflicting assignments are denied (denials take precedence); (5) use audit and/or IDS to track use of low integrity data and its effects; and (6) change the SELinux policy specification to remove the conflict. Of course, adding the conflicting type to the TCB is also an option. We are investigating the effectiveness of all options.

One controversial option is sanitization. LSM provides hooks that could be used for sanitization in some cases (e.g., socket read, file read, but not for mmapped files). Also, we are interested in program analysis of the use of low integrity data by high integrity programs.

2 Proposed TCB Subject Types

We propose an initial set of TCB subject types. System authentication, initialization, and administration services compromise the initial TCB shown in Figure 1. The graph shows the transition relations between some basic subject types. Since these can transition to many other subject types or define data used by one or more of the other TCB types they are included.

Rather than the more extensive core services proposed on the SELinux mailing list, we are trying to find a near-minimal TCB. However, analysis dictates that we add to the TCB. Other services we added include `dpkg_t` and `devfsd_t` which people have indicated should not be in the TCB.

3 Biba Conflicts

Table 1 shows the integrity conflicts between the TCB subject types and the remainder of the system. The conflicts indicate a subject type and write permission (under object type and op) that impacts an object type used by the trusted type.

The last two columns indicate the resolution to this conflict. *Class* is an automated estimate of the least complex resolution and *resolution* is our best guess at the time. The dicussion on the mailing list will cause me to refine these resolutions.

I should make it clear that this table does not express all Biba integrity conflicts, but rather, one conflict per permission assignment. In this case, we collect one instance of a write permission to an object to which a high

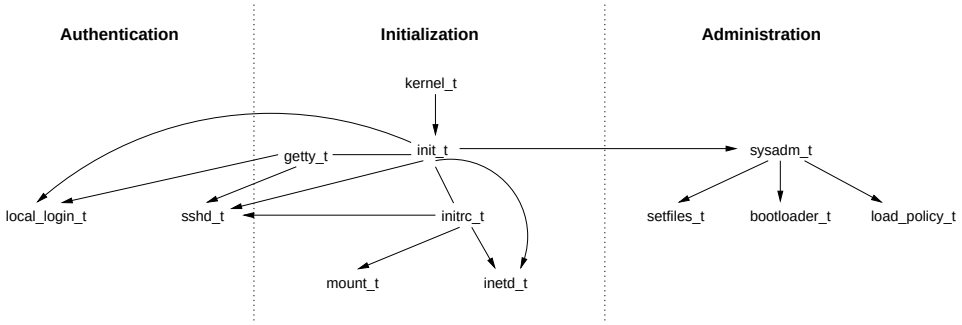

Figure 1: SELinux Example Policy's type transition hierarchy for our proposed TCB subject types.

integrity process has read permission. Also, we only collect one instance of a unique read permission. Therefore, other conflicts between subject types may need to be resolved, but these are the set of conflicting permissions. I hope that made sense.

Also, a couple of permissions like `netif_type:netif` and `node_type:node` are not listed.

4 Trusted Computing Base Subject Types

In Table 2, we express our proposal for a minimal TCB. Based on comments from Stephen, Russell, and others this list could be pruned further.

5 Excluded Subject Types

In Table 3, the set of subjects that must be excluded in order for this set of TCB subjects to be supported is listed. This combination of trusted and excluded subjects depends on the sanitizations and policy changes under the resolution column above. Further investigation is needed to determine the exact changes/sanitizations.

Trusted Type	*Conflict Type*	*Object Type & Op*	*Class*	*Resolution*
devfsd_t	many	file_type:blk/chr/file	change	change
dpkg_t	tmpreaper_t	tmp_dpkg_t:file all	trust	exclude
initrc_t	useradd_t	etc_t:file write	trust	trust
initrc_t	gpm_t	psaux_t:chr write	exclude	trust
initrc_t	hwclock_t	clock_device_t:chr/blk write	trust	trust
initrc_t	sound_t, xdm_t	sound_device_t:chr write	trust	exclude
initrc_t	httpd_admin_xserver_t	framebuf_device_t:chr write	change	sanitize
initrc_t	many	initrc_t:fifo write	exclude	sanitize
kernel_t	slapd_t, squid_t, +	*:*_socket sendto	sanitize	sanitize
kernel_t	dhcpc_t	resolv_conf_t:file write	trust	trust
kernel_t	dhcpd_t	var_run_dhcpd_t:file write	trust	trust
kernel_t	quota_t	file_t:file quotaon	trust	trust
local_login_t	many	proc_t:file write	sanitize	sanitize
local_login_t	insmod_t	local_login_t:process signal	exclude	exclude
local_login_t	logrotate_t	local_login_t:process signal	trust	trust
mount_t	automount_t	autofs_t:dir all	exclude	trust
mount_t	bootloader_t, fsadm_t	fixed_disk_device_t:* all	trust	trust
sysadm_t	user_t	misc_device_t:* all	sanitize	exclude obj
sysadm_t	many	sysadm_devpts_t/ptyfile:* all	change	change
sysadm_t	sysadm_*_t	sysadm_home_t:* write	change	change/sanitize one file
sysadm_t	sysadm_*_t	sysadm_tmp_t:file exec	exclude	change
sysadm_t	sysadm_irc_t	sysadm_irc_t:file all	exclude	change/sanitize
sysadm_t	sysadm_xserver_t	sysadm_xserver_t:shm all	exclude	exclude
sysadm_t	sysadm_xauth_t	sysadm_home_xauth_t:file all	exclude	exclude
sysadm_t	admin	kernel_t:system avc_toggle	trust	trust
sshd_t	many	sshd_devpts_t/userpty:* all	change	change

Table 1: Biba integrity conflicts of proposed TCB for SELinux example policy.

kernel_t	init_t	initrc_t	sysadm_t	getty_t
mount_t	fsadm_t	load_policy_t	dpkg_t	devfsd_t
setfiles_t	dhcpc_t	dhcpd_t	automount_t	sshd_t
sshd_login_t	local_login_t	quota_t	gpm_t	useradd_t
hwclock_t	apt_t	install_menu_t	ipsec_mgmt_t	admin_passwd_exec_t
bootloader_t	logrotate_t	newrole_t	snmpd_t	passwd_t
syslogd_t	checkpolicy_t	cardmgr_t	ldconfig_t	klogd_t

Table 2: Final trusted computing base subject types.

insmod_t	rlogind_t	remote_login_t	sysadm_xserver_t	xdm_t
sysadm_xauth_t	sound_t	tmpreaper_t		kmod_t
lpd_t	xdm_xserver_t	vmware_user_t	sendmail_t	procmail_t
hotplug_t	traceroute_t	update_modules_t	gatekeeper_t	smbd_t

Table 3: Final excluded subject types.