ExecVus

Alexandru Totolici
a way to visualize control-flow in software execution
planned for:

- code collapse
- code reordering
- improved selection of exec. path
- “zoom” in/out of control flow
- scented graph for func. callers/callees
about then now delta demo future q+a

managed:
✓ code collapse
✗ code reordering
...
improved selection of exec. path
...
“zoom” in/out of control flow
✓ scented graph for func. callers/callees
about then **now** delta demo future q+a

bonus, learned:

```html
<canvas>
javascript / jquery
mercurial / hgweb
python
mice in IKBLC!
```
about then now delta demo future q+a

improved selection of exec. path
rewrite to underlying engine required
animating on <canvas> might be slow
“zoom” in/out of control flow
some backend code written
had issues because of dataset
demo
about then now delta demo future q+a

finish zooming functionality
improve scents
performance tweaks
questions? comments.
(answers optional)
EXPORT_SYMBOL(key_instantiate_and_link);

int key_negate_and_link(struct key *key,
 EXPORT_SYMBOL(key_negate_and_link);
 static void key_cleanup(struct work_struct *work)
 void key_put(struct key *key)
 {
 if (key) {
 key_check(key);

 if (atomic_dec_and_test(&key->usage))
 schedule_work(&key_cleanup_task);

 }
int security_audit_rule_init(u32 field, u32 op, char *rulestr, void **lsmrule)

int security_key_getsecurity(struct key *key, char **_buffer)

int security_xfrm_state_pol_flow_match(struct xfrm_state *x,
int security_xfrm_policy_lookup(struct xfrm_sec_ctx *ctx, u32 fl_secid, u8 dir)
EXPORT_SYMBOL(security_xfrm_state_delete);
int security_xfrm_state_delete(struct xfrm_state *x)
EXPORT_SYMBOL(security_xfrm_policy_alloc);
void security_inet_conn_established(struct sock *sk,
EXPORT_SYMBOL(security_req_classify_flow);
void security_req_classify_flow(const struct request_sock *req, struct flowi *fl)
EXPORT_SYMBOL(security_sk_classify_flow);
EXPORT_SYMBOL(security_socket_getpeersec_dgram);
int security_socket_getpeersec_dgram(struct socket *sock, struct sk_buff *skb, u32 *secid)
int security_socket_getsockname(struct socket *sock)
int security_socket_recvmsg(struct socket *sock, struct msghdr *msg,
int security_socket_sendmsg(struct socket *sock, struct msghdr *msg, int size)
int security_socket_bind(struct socket *sock, struct sockaddr *address, int addrlen)
int security_unix_may_send(struct socket *sock, struct socket *other)
int security_unix_stream_connect(struct socket *sock, struct socket *other,
EXPORT_SYMBOL(security_ipc_permission);
int security_ipc_permission(struct kern_ipc_perm *ipcp, short flag)
void security_task_to_inode(struct task_struct *p, struct inode *inode)
int security_task_movememory(struct task_struct *p)
int security_task_getioprio(struct task_struct *p)
EXPORT_SYMBOL(security_task_getsecid);
void security_task_getsecid(struct task_struct *p, u32 *secid)
int security_prepare_creds(struct cred *new, const struct cred *old, gfp_t gfp)
int security_dentry_open(struct file *file, const struct cred *cred)
int security_file_set_fowner(struct file *file)
int security_file_lock(struct file *file, unsigned int cmd)
int security_file_mmap(struct file *file, unsigned long reqprot,
int security_file_alloc(struct file *file)
void security_inode_delete(struct inode *inode)
int security_inode_follow_link(struct dentry *dentry, struct nameidata *nd)
int security_inode_rmdir(struct inode *dir, struct dentry *dentry)
int security_inode_create(struct inode *dir, struct dentry *dentry, int mode)
int security_path_rename(struct path *old_dir, struct dentry *old_dentry,
int security_path_rmdir(struct path *path, struct dentry *dentry)
int security_inode_init_security(struct inode *inode, struct inode *dir,
EXPORT_SYMBOL(security_sb_parse_opts_str);
EXPORT_SYMBOL(security_sb_clone_mnt_opts);
EXPORT_SYMBOL(security_sb_set_mnt_opts);
int security_sb_pivotroot(struct path *old_path, struct path *new_path)
void security_sb_post_addmount(struct vfsmount *mnt, struct path *mountpoint)
void security_sb_post_remount(struct vfsmount *mnt, unsigned long flags, void *data)
void security_sb_umount_close(struct vfsmount *mnt)
int security_sb_mount(char *dev_name, struct path *path,
int security_sb_statfs(struct dentry *dentry)
EXPORT_SYMBOL(security_sb_copy_data);
int security_bprm_secureexec(struct linux_binprm *bprm)
int security_vm_enough_memory_mm(struct mm_struct *mm, long pages)
int security_vm_enough_memory(long pages)
int security_acct(struct file *file)
int security_real_capable_noaudit(struct task_struct *tsk, int cap)
int security_capset(struct cred *new, const struct cred *old,
static void __init do_security_initcalls(void)
static inline int verify(struct security_operations *ops)