

Always on the GO with FreeSWITCH

OpenSIPS Summit - Amsterdam 2019

HELLO

my name is

M I C H A E L

MAVROUDIS

- Founder / CEO
VoiceTel
 - Community Liaison
FreeSWITCH
-

ESL

What is ESL?



ESL

Event Socket Library - Allows a two-way communication link between FreeSWITCH and an application. There are over 200 valid commands in a standard installation including:

...	is_lan_addr	status	uuid_media	vm_fsdh_pref_password_set
acl	json	stop_tdd_detect	uuid_media_ip	vm_fsdh_pref_rename_set
alias	limit_hash_usage	stopch	uuid_media_send	vm_inject
amr_debug	limit_interval_reset	strftime	uuid_morp_send	vm_list
banner	limit_reset	strftime.1z	uuid_outgoing_answer	vm_prefs
bg_system	limit_status	stmicorepoch	uuid_park	vm_read
bgapi	limit_usage	stun	uuid_pause	voicemail
break	list_users	switchname	uuid_phone_event	voicemail_inject
chat	load	system	uuid_pre_answer	xml_flash_cache
coalesce	local_stream	time_test	uuid_preprocess	xml_locate
complete	log	time_test	uuid_record	xml_wrap
cond	lua	tolower	uuid_recovery_refresh	
conference	luarun	tone_detect	uuid_recv_dtmf	
console_complete	md5	trapper	uuid_redirect	
console_complete_xml	module_exists	unload	uuid_ring_ready	
create_uuid	msleep	unsched_api	uuid_send_dtmf	
db	msrp	uptime	uuid_send_info	
db_cache	nat_map	url_decode	uuid_send_message	
domain_data	nvpa_debug	url_encode	uuid_send_tdd	
domain_exists	originate	user_data	uuid_send_text	
echo	page	user_exists	uuid_session_heartbeat	
escape	pause	uuid_answer	uuid_set_media_stats	
eval	presence	uuid_audio	uuid_setvar	
event_channel_broadcast	quote_shell_arg	uuid_break	uuid_setvar_multi	
expand	req_url	uuid_bridge	uuid_simplifi	
expr	regex	uuid_broadcast	uuid_transfer	
file	reload	uuid_buylist	uuid_video_bandwidth	
file_add_outbound	reloadacl	uuid_capture_text	uuid_video_bitrate	
file_check_bridge	reloadxml	uuid_chat	uuid_video_refresh	
file_member	replace	uuid_codec_debug	uuid_write_png	
file_exists	say_string	uuid_codec_param	uuid_xfer_zombie	
find_user_xml	sched_api	uuid_debug_media	uuid_zombie_exec	
fscl	sched_broadcast	uuid_deflect	uuid_info	
getcpulime	sched_del	uuid_display	version	
getenv	sched_hangup	uuid_display	verto	
gethost	sched_transfer	uuid_dtmf_dtmf	verto_contact	
global_getvar	show	uuid_dtmf_transfer	vm_baccout	
global_setvar	shutdown	uuid_dump	vm_delete	
group	signalwire	uuid_early_ok	vm_fsdh_auth_login	
group_call	sofia	uuid_exists	vm_fsdh_msig_count	
hash	sofia_contact	uuid_filename	vm_fsdh_msig_delete	
hash_dump	sofia_count_reg	uuid_flush_dtmf	vm_fsdh_msig_email	
hash_remote	sofia_dig	uuid_getvar	vm_fsdh_msig_forward	
help	sofia_gettray_data	uuid_hold	vm_fsdh_msig_get	
host_lookup	sofia_presence_data	uuid_jitterbuffer	vm_fsdh_msig_list	
hostname	sofia_username_of	uuid_kill	vm_fsdh_msig_merge	
htapi	spandsp_start_tone_detect	uuid_limit	vm_fsdh_msig_save	
hupall	spandsp_stop_tone_detect	uuid_limit_release	vm_fsdh_msig_undelete	
in_group	sql_escape	uuid_loglevel	vm_fsdh_pref_greeting_get	
interface_ip	start_tdd_detect		vm_fsdh_pref_greeting_set	

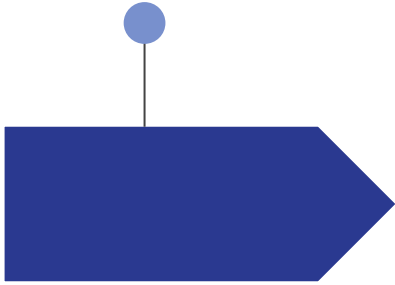
Questions

- When is it listening?
- Where is it listening?
- How is it secured?

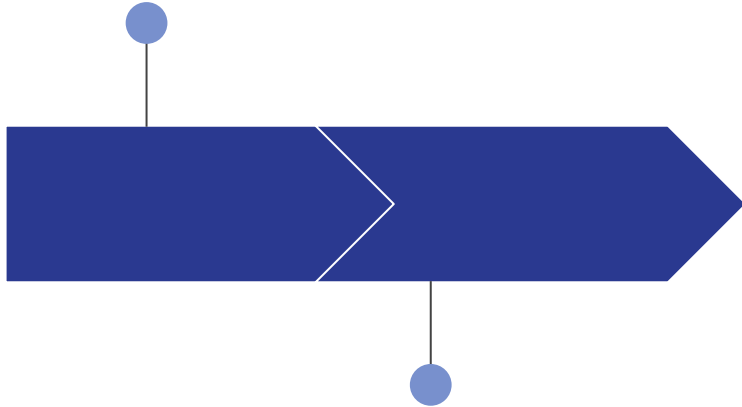


Connection brokering middleware
to restrict user/application
access to ESL commands.

External Request
containing user
authentication and
command



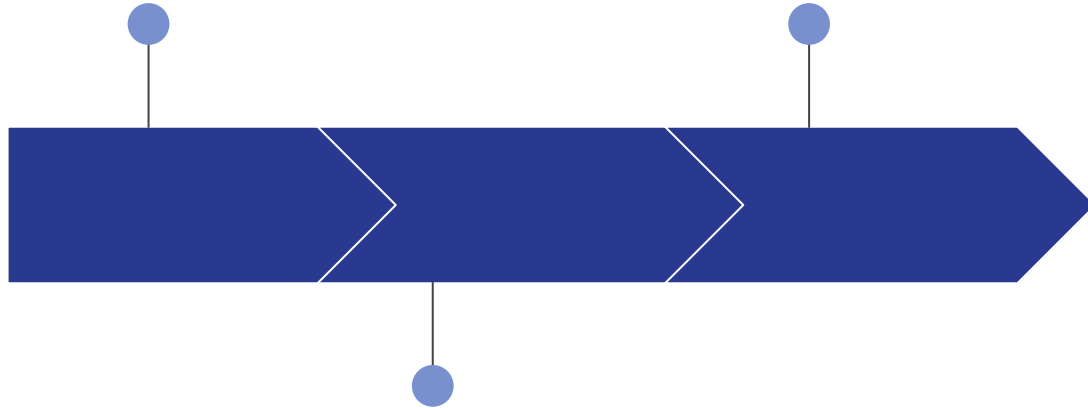
External Request
containing user
authentication and
command



Check directory and
verify credentials

External Request
containing user
authentication and
command

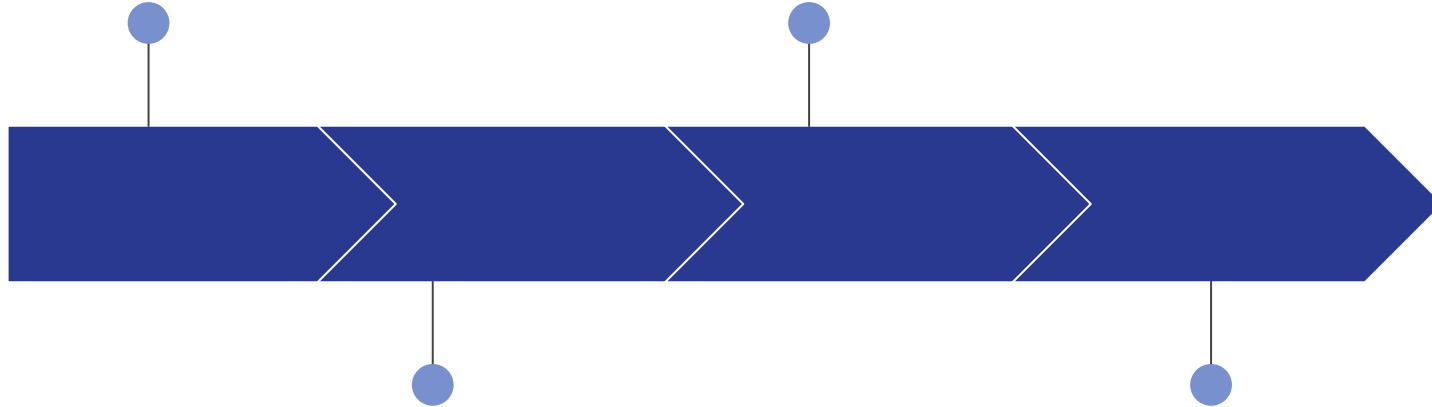
Verify command
against per user
whitelist.



Check directory and
verify credentials

External Request
containing user
authentication and
command

Verify command
against per user
whitelist.



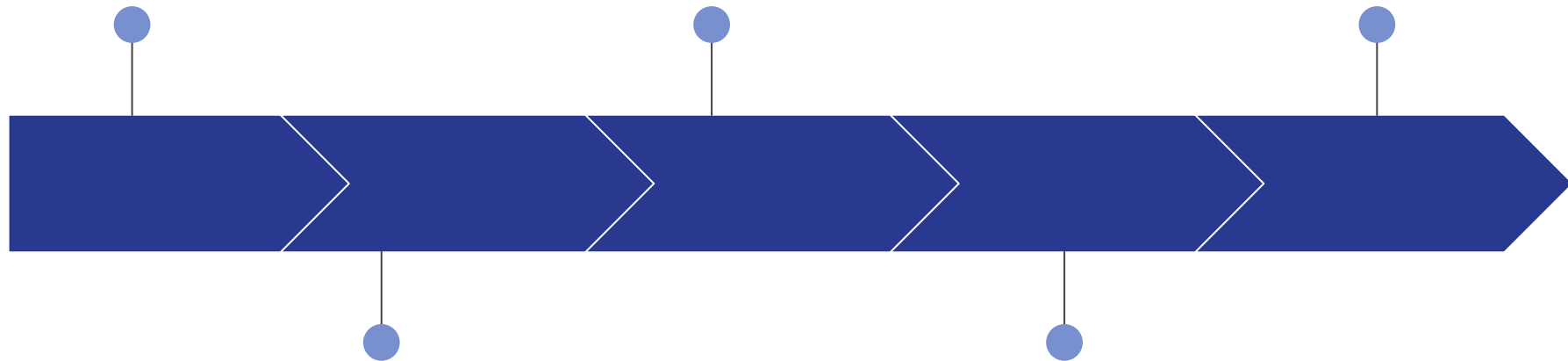
Check directory and
verify credentials

Process ESL
command

External Request
containing user
authentication and
command

Verify command
against per user
whitelist.

Receive ESL response



Check directory and
verify credentials

Process ESL
command

Questions

- Is this a good idea?
- Is there a better way?
- How can this be used?

Simple Go Implementation

```
1. import (  
2.     "fmt"  
3.     "strings"  
4.     "net/http"  
5.     "encoding/json"  
6.  
    "github.com/fiorix/go-eventsocket/events  
    ocket"  
7. )
```