

# Nftables-gui

Una interfaz gráfica para el nuevo cortafuegos de Linux.

José María Caballero Alba  
caballeroalba@gmail.com

May 6, 2015

- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones

# Problema

Nftables es un nuevo framework que sustituye al antiguo iptables. Este nuevo software aun no esta desarrollado al 100 % pero ya es operativo en gran parte de sus funciones.

# Problema

- Nueva sintaxis (Desinterés del usuario)
- Tiene una curva de apredizaje.
- Aún en desarrollo
- Migración en servidores

# Problema

- nft add rule filter input ip saddr 192.168.1.0/24 ct established accept
- nft add rule ip nat preroute dnat tcp dport map { 80 : 192.168.1.2, 21 : 192.168.1.3 }

```
root@LUNA:/home/caballeroalba# nft list table filter -nn
table ip filter {
    chain input {
        type filter hook input priority 0; policy accept;
        ip saddr 192.168.1.0/24 ct state established accept
    }

    chain preroute {
        type filter hook prerouting priority 0; policy accept;
        dnat tcp dport map { 21 : 92.168.1.3, 80 : 192.168.1.2}
    }
}
root@LUNA:/home/caballeroalba#
```

- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones

# Solución aportada

- La solución aportada consistirá en una interfaz gráfica escrita en c y usando ncurses para poder manejar nftables y que de esta manera sea mas fácil su uso.
- Esto implica una mejora para aquellas personas que quieran dejar de usar iptables y puedan utilizar nftables con todas las características nuevas haciendo que la pendiente de la curva sea mas liviana.

# Solución aportada

- Esto nos da la solución a:
  - Curva de aprendizaje
  - Migración final
  - Desinterés de usuario



- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones

# Estudio del arte

Software	Facilidad de uso	Vida del proyecto	Tipo interfaz	Soporte	Configuración
vuurmuur	3	2009	ncurses	si	5
ipmenu	5	2001	curl	no	2
fwbuilder	4	2013	gtk	si	5
easy firewall generator	5	2005	web	no	1
turtlefirewall	5	2011	web	si	3

La puntuación va desde 1 a 5, siendo 1 la peor y 5 la mejor

# Estudio del arte

El problema viene dado por que ninguno de ellos esta hecho para nftables.

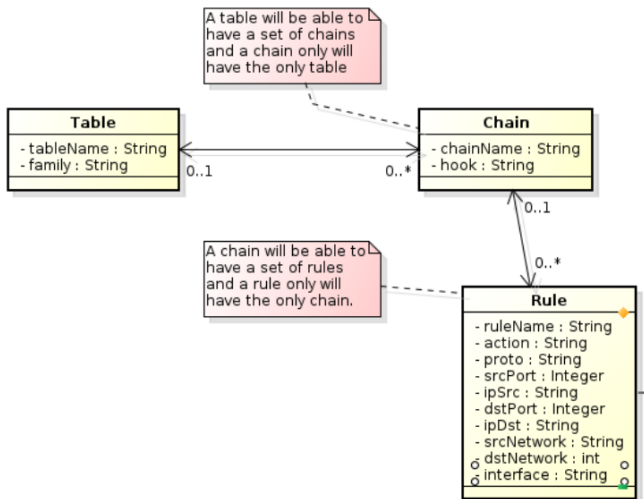
- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones

# Requisitos

- Crear/borrar tablas de tipo ip, ip6, arp, bridge
- Crear/borrar cadenas de tipo base o personalizadas
  - Hook (INPUT, OUTPUT, PREROUTING, FORWARD, POSTROUTING)
- Crear/borrar reglas permitiendo personalizar el filtrado teniendo en cuenta:
  - Procotolo
  - Ip/red/puerto/interfaz origen y destino
  - Diccionarios
  - Zonas
  - NAT
  - Mapas

- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones

# Modelo conceptual



# Diseño

- Este software se ha llevado a cabo usando el modelo arquitectónico MVC, separando la lógica del software con la presentación al usuario en ncurses.
- Esto se ha llevado a cabo creando una api que permite mediante una serie de parámetros crear una vista sin que esta sepa de la lógica de negocio de la aplicación.



- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones

# Implementación

- Utilidad de linea de comandos de espacio de usuario nft ( utilidad para interactuar con nftables)
- GNU Build System (Autotools ) para la construcción del software
- Git como control de versiones

# Implementación

```

enum {
    NFTGUI_TABLE_TABLE_NAME,
    NFTGUI_TABLE_FAMILY,
    NFTGUI_TABLE_CHAIN,
    NFTGUI_TABLE_NUM_CHAINS,
    __NFTGUI_TABLE_MAX
};

#define NFTGUI_TABLE_MAX (__NFTGUI_TABLE_MAX -1)

#define xfree(ptr)    free((void *)ptr);

struct table {
    struct list_head head;
    struct list_head chains;
    const char *table_name;
    const char *family;
    uint32_t num_chains;

    uint32_t flags;
};

```

```

enum {
    NFTGUI_CHAIN_CHAIN_NAME,
    NFTGUI_CHAIN_HOOK,
    NFTGUI_CHAIN_RULE,
    NFTGUI_CHAIN_NUM_RULES,
    __NFTGUI_CHAIN_MAX
};

#define NFTGUI_CHAIN_MAX (__NFTGUI_CHAIN_MAX -1)

#define xfree(ptr)    free((void *)ptr);

struct chain {
    struct list_head head;
    struct list_head rules;
    const char *chain_name;
    const char *hook;
    uint32_t num_rules;

    uint32_t flags;
};

```

# Implementación

```
enum {
    NFTGUI_RULE_NAME,
    NFTGUI_RULE_ACTION,
    NFTGUI_RULE_PROTO,
    NFTGUI_RULE_SRCPORT,
    NFTGUI_RULE_DSTPORT,
    NFTGUI_RULE_IPSRC,
    NFTGUI_RULE_IPDST,
    NFTGUI_RULE_SRCNETWORK,
    NFTGUI_RULE_DSTNETWORK,
    NFTGUI_RULE_INTERFACE,
    NFTGUI_RULE_ID,
    __NFTGUI_RULE_MAX
};

#define NFTGUI_RULE_MAX (__NFTGUI_RULE_MAX - 1)

#define xfree(ptr)      free((void *)ptr);

struct rule {
    struct list_head head;
    uint32_t id;
    const char *rule_name;
    const char *action;
    const char *proto;
    uint32_t srcport;
    uint32_t dstport;
    const char *ipsrc;
    const char *ipdst;
    const char *srcnetwork;
    const char *dstnetwork;
    const char *interface;

    uint32_t flags;
};

#define MAX_BUFFER_SIZE 4096

extern char buf_screen[MAX_BUFFER_SIZE];

/* prototypes */
void create_table(struct table *t);
//void list_tables(struct table_list *list);

struct table_list {
    struct list_head list;
    int elements;
};

void delete_all_tables();
void list_tables(struct table_list *list);
void list_table_details(int ntable, struct table_list *list);
void list_chain_details(int ntable, int nchain, struct table_list *list);
void list_chains(int ntable, struct table_list *list);
void list_rules(struct chain *ch);
void list_rule_details(struct chain *ch, int nrule);
struct rule * get_rule(struct chain *ch, int nrule);
void create_chain(int ntable, struct table_list *list);
void delete_chain(int ntable, struct table_list *list);
void delete_table(int ntable, struct table_list *list);
void create_rule(struct table *t, struct chain *ch);
struct chain * get_chain(struct table *t, int nchain);
struct table * get_table(int ntable, struct table_list *list);
char *trim(char *s);
```

# Implementación

```
int print_menu(int highlight, char *choices[], int n_choices, char *message, char *title)
{
    int x,y,i;
    int startx,starty=0;
    WINDOW *menu_win;
    startx=(40-WIDTH) / 2;
    starty=(24-HEIGHT) /2;

    initscr();

void form_create(int n, char *opts[], char *opts_values[])
{
    FIELD *field[n];
    FORM *my_form;
    int ch;

    /* Initialize curses */
    initscr();
    cbreak();
    noecho();
    keypad(stdscr, TRUE);

    /* Initialize the fields */
```

- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones

# Trabajo futuro

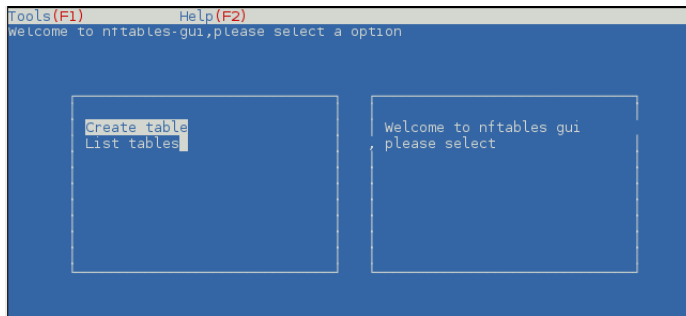
Este proyecto no termina aquí. En el futuro se pretende:

- Traducciones a otros lenguajes.
- Mejoras en la funcionalidad
- Inclusión en los repositorios oficiales de las distribuciones habituales
- Libnftnl para la comunicación con el subsistema del kernel `nf_tables`
- Libmnl para la validación, construcción y parseo de las estructuras de tablas, cadenas y reglas.

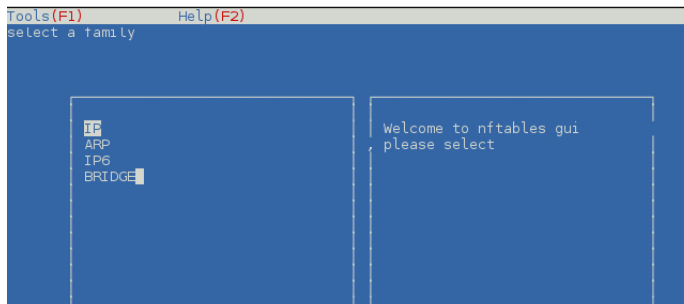
- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones



# Resultados



# Resultados



# Resultados

Please fill values and press F2 to exit

Table name filter

Select a table for details

filter

# Resultados

You are in filter table, please select a option

```
List chains
Create chain
Delete this table
Back
```

Please fill values and press F2 to exit

```
Chain name  Internet
Hook        input
```

# Resultados

You are in filter table chain list,  
please select a chain for details

Internet

Please fill values and press F2 to exit

Rule name: \_\_\_\_\_  
Action: \_\_\_\_\_  
Protocol: \_\_\_\_\_  
Src port: \_\_\_\_\_  
Dst port: \_\_\_\_\_

# Resultados

Please fill values and press F2 to exit

Rule name: test  
Action: drop  
Protocol: tcp  
Src port: 80  
Dst port: 80

# Resultados

```
root@LUNA:/home/caballeroalba# nft list table filter
table ip filter {
    chain Internet {
        type filter hook input priority 0; policy accept;
        tcp sport http drop
    }
}
root@LUNA:/home/caballeroalba#
```

- nft create table filter
- nft create chain filter Internet {type filter hook input priority 0 \; }
- nft add rule filter Internet tcp tcp sport 80 drop

- 1 Problema
- 2 Solución aportada.
- 3 Estudio del arte
- 4 Requisitos
- 5 Diseño
- 6 Implentación
- 7 Trabajo futuro
- 8 Resultados
- 9 Conclusiones



# Conclusions



# Gracias

THANK YOU

GRACIAS

ARIGATO

SHUKURIA

GOZAIMASHITA

EFCHARISTO

JUSPAXAR

DANKSCHEEN

TASHAKKUR ATU

YAQHANYELAY

SUKSAMA

EKHMET

MEHRBANI

PALDIES

BIYAN

SHUKRIA

BOLZIN

MERCI