RULE SET BASED ACCESS CONTROL

Paweł Bylina
(pako@overflow.pl)

Plan wykładu.

- Czym jest RSBAC?
- Moduł AUTH.
- Moduł UM.
- Moduł RC.
- Moduł ACL.
- Moduł CAP.
- Co jeszcze?
- Implementacja polityki bezpieczeństwa dla Apache'a.
- Podsumowanie.

Czym jest RSBAC?

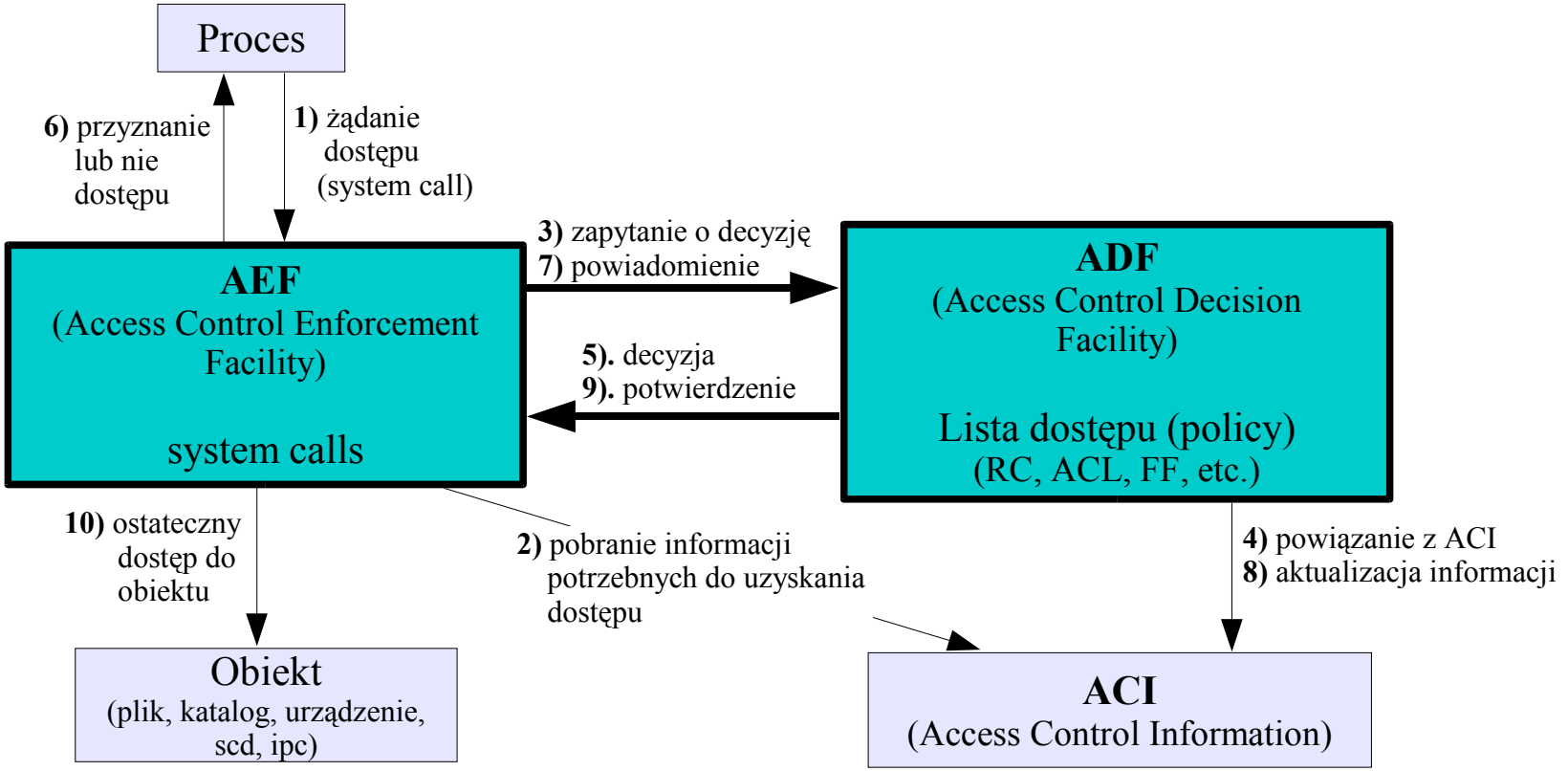
- Krótka historia.
- Czym jest RSBAC?
  - **Cechy pozytywne:**
    - modularność,
    - szybkość,
    - wysoka konfigurowalność,
    - przenośność kodu (FreeBSD, Solaris, AIX) oraz jego czystość,
    - niezależność od filesystemów.
  - **Cechy negatywne:**
    - czasochłonna konfiguracja,
    - brak dokumentacji (!).

Czym jest RSBAC?

- Secofficer.
- Softmode.
- Switch module.
- ADF debug dla modułów.
- Logowanie dostępu:
  - access,
  - denied.
- Narzędzia administracyjne.
- Wydajność.

Czym jest RSBAC?

Architektura RSBACa



Moduł AUTH

- Co to AUTH?
  - **Kontrola zmiany właściciela procesu:**
    - blokady autoryzacyjne (sshd),
    - kontrola aplikacji SUIDowych.
  - **auth_learn,**
  - **rsbac_auth_learn.**

Moduł AUTH

Przykład próby zmiany UIDu procesu

```c
#include <unistd.h>
#include <stdlib.h>


int main(void)
{
      if (setuid(0) < 0) {
            exit(EXIT_FAILURE);
      }
      return EXIT_SUCCESS;
}
```

```
rsbac-host:~# ./setuid
0000000133|rsbac_adf_request(): request CHANGE_OWNER, pid 1759, ppid 1345,
prog_name setuid, prog_file /root/setuid, uid 0, target_type PROCESS, tid 1759,
attr
owner, value 0, result NOT_GRANTED (Softmode) by AUTH
rsbac-host:~#
```

Moduł UM (User Management)

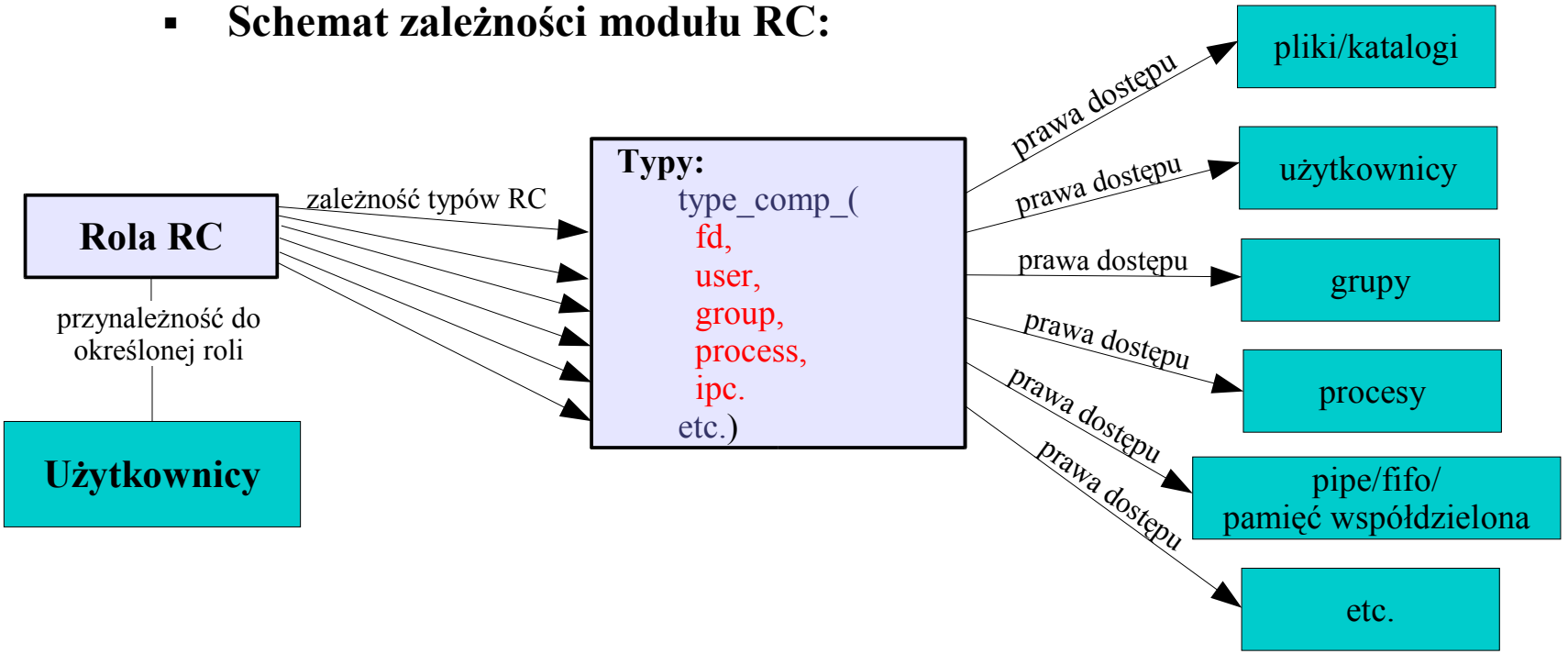
- Co to UM?
  - **Co zyskujemy dzięki zastosowaniu UM?:**
    - wykluczenie z systemu plików /etc/passwd, shadow, group,
    - szyfrowanie haseł algorytmem SHA1,
    - separacje kont administratorów,
    - wykluczenie dostępu aplikacjom do haseł systemowych (biblioteki libpam oraz libnss).

zaimplementowany UM

```
rsbac-host:~# ls /etc/passwd /etc/shadow
/etc/passwd  /etc/shadow
rsbac-host:~# rm /etc/passwd /etc/shadow
rsbac-host:~# ls /etc/passwd /etc/shadow
ls: /etc/passwd: Nie ma takiego pliku ani katalogu
ls: /etc/shadow: Nie ma takiego pliku ani katalogu
rsbac-host:~# rsbac_login
rsbac-host login: pako
pako's RSBAC password:
pako@rsbac-host:~$
```

Moduł RC (Role Compatibility)

- Co to RC?
  - **Schemat zależności modułu RC:**

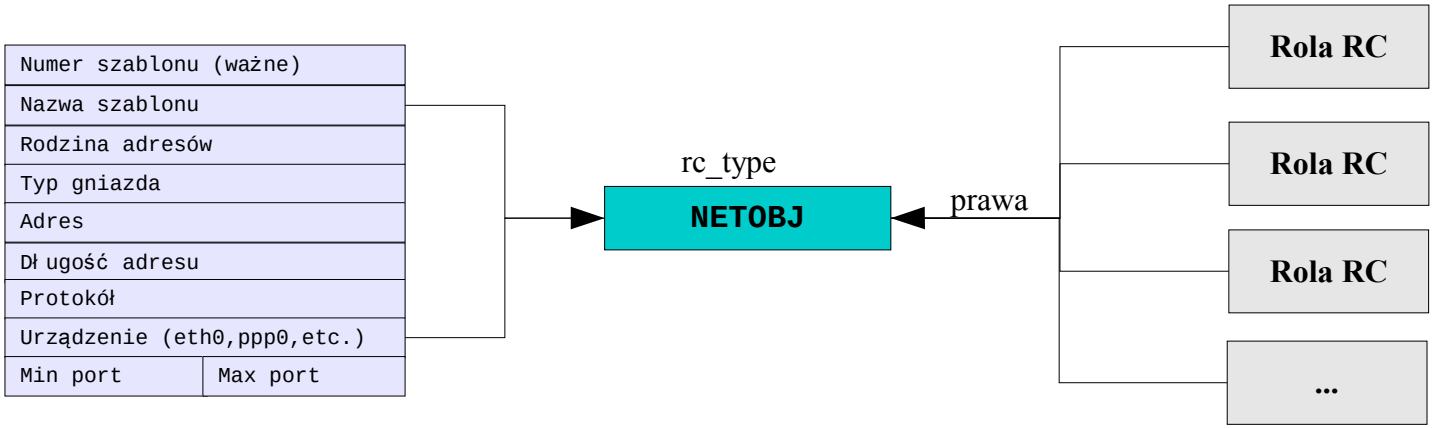


Moduł RC (Role Compatibility)

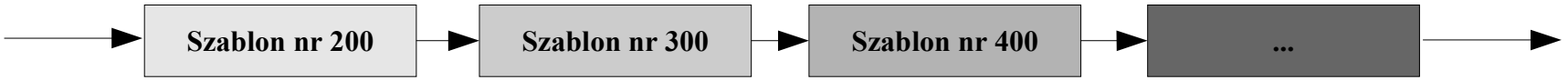
- Możliwości konfiguracyjne modułu RC:
  - Dostęp do plików/katalogów/dowiązań symbolicznych/urządzeń,
  - Prawa względem modułu UM (zarządzanie użytkownikami, grupami),
  - Zarządzanie obiektami IPC, SCD,
  - Zarządzanie urządzeniami internetowymi (ppp0, eth0, itd.) oraz obiektami i szablonami dostępu internetowego.
  - **Konfiguracja ról/typów dla nowo utworzonych:**
    - obiektów FD (plików/katalogów etc.)
    - użytkowników,
    - grup,
    - procesów,
    - obiektów IPC.

Moduł RC (Role Compatibility)

- NetTemplates (szablony internetowe)



- Hierarchia szablonów



Moduł RC (Role Compatibility)

- Domyślne role i typy modułu RC.
- Przykład nadania roli oraz typu RC:
  - Skopiowanie i nadanie użytkownikowi istniejącej roli General_User:

```
secoff@rsbac-host:~$ rc_copy_role 0 4
secoff@rsbac-host:~$ rc_set_item ROLE 4 name 'rola_test'
secoff@rsbac-host:~$ attr_set_user jasio rc_def_role 4
```

  - Utworzenie i nadanie typu obiektowi FD:

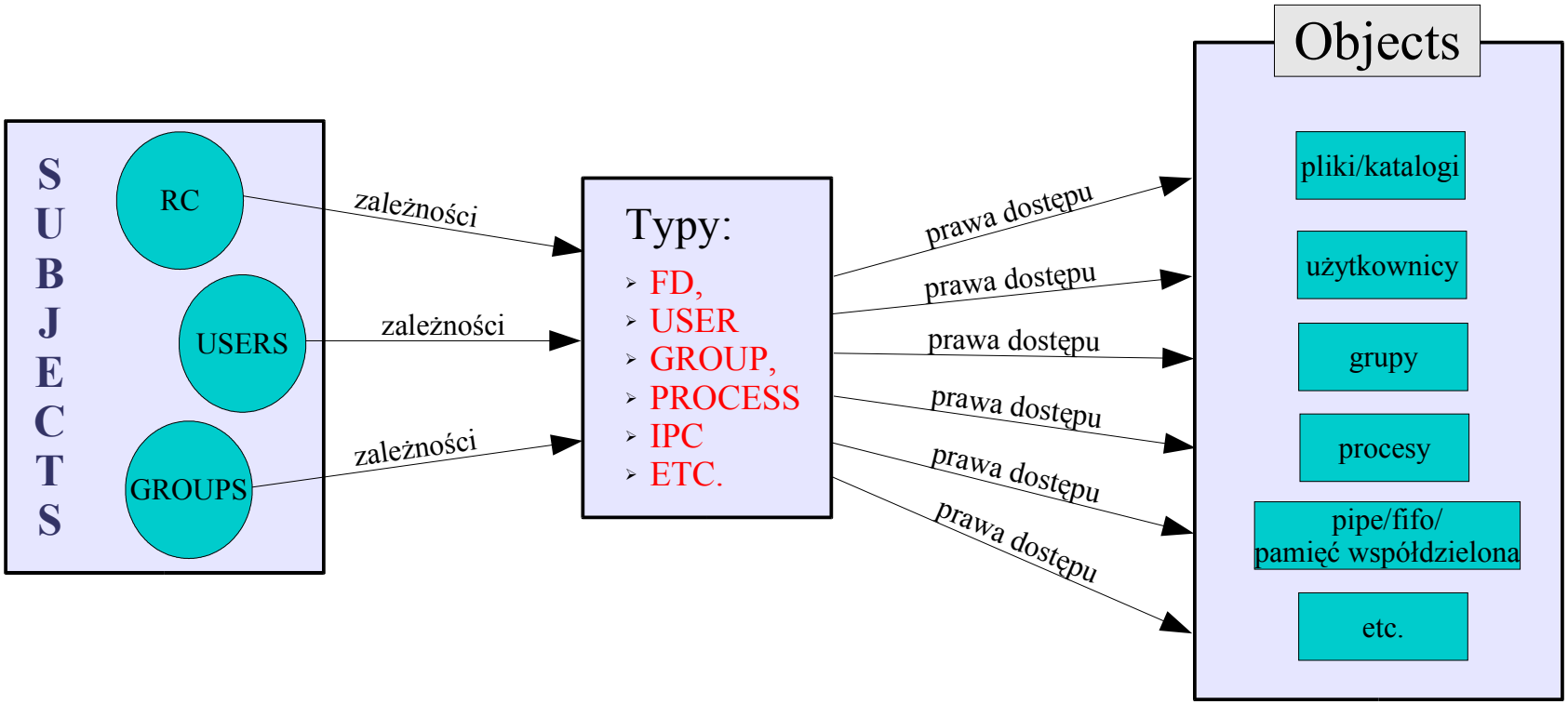
```
secoff@rsbac-host:/home/jasio$ rc_set_item TYPE 3 type_fd_name 'typ_fd'
secoff@rsbac-host:/home/jasio$ echo 'test' > plik
secoff@rsbac-host:/home/jasio$ attr_set_file_dir FD plik rc_type_fd 3
```

  - Nadanie praw **do odczytu** roli 'rola_test' (4) dla typu FD 'typ_fd' (3):

```
secoff@rsbac-host:/home/jasio$ rc_set_item ROLE 4 type_comp_fd 3
READ_OPEN READ GET_STATUS_DATA CLOSE
```

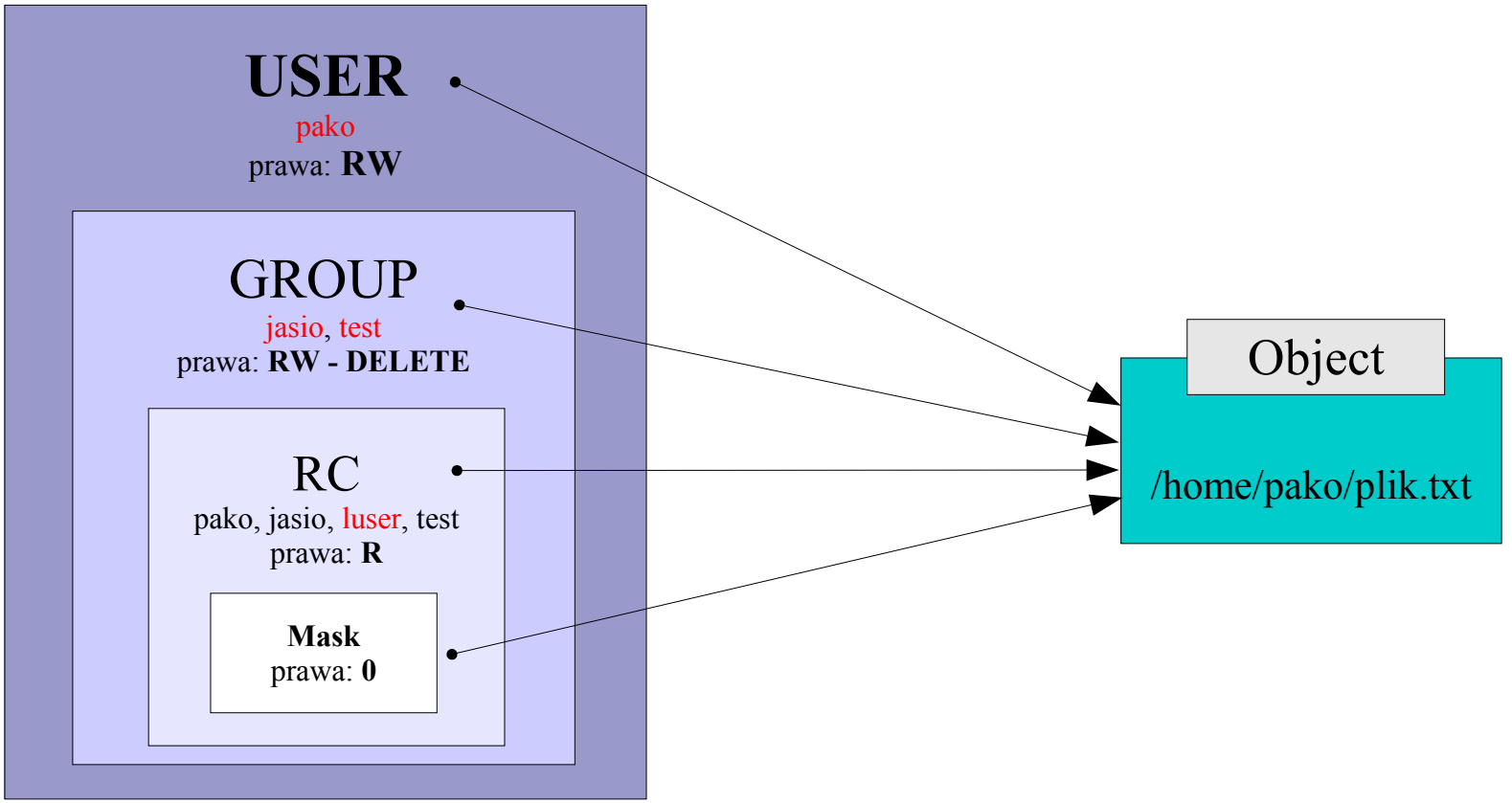
Moduł ACL (Access Control Lists)

- ACL jako uzupełnienie/rozwinięcie RC.
  - **Schemat zależności modułu ACL:**



Moduł ACL (Access Control Lists)

Hierarchia praw dostępu



Moduł ACL (Access Control Lists)

- Rozwinięcia względem modułu RC:
  - **Możliwość nadawania praw dla:**
    - pojedynczych użytkowników,
    - grup ACL,
    - grup RC.
  - **Tworzenie własnych grup ACL:**
    - private,
    - global.
  - **Możliwość określenia TTL ustawionych praw.**
  - **linux2acl (z DAC disable).**

Moduł CAP

- Istota linuksowych CAPabilities.
  - **Wykorzystanie:**
    - restrykcja praw programów uruchomionych przez roota,
    - nadanie wybranych praw administratora określonym plikom lub użytkownikom,
    - możliwość pozbycia się aplikacji SUIDowych.
- Wybrane prawa: DAC_OVERRIDE.

```
pako@rsbac-host:~$ cd /root
-/bin/bash: cd: /root: Brak dostępu
```

```
secoff@rsbac-host:~$ attr_set_user -a pako min_caps DAC_OVERRIDE
```

```
pako@rsbac-host:~$ cd /root
pako@rsbac-host:/root$
```

Moduł CAP

Wybrane prawa: **NET_BIND_SERVICE**

```
int main(void)
{
      struct sockaddr_in serv;
      int sock;

      memset(&serv, 0, sizeof(serv));
      serv.sin_family = PF_INET;
      serv.sin_port = htons(80);
      inet_aton("127.0.0.1", &serv.sin_addr);
      sock = socket(PF_INET, SOCK_STREAM, 0);

      if (bind(sock, (struct sockaddr *)&serv, sizeof(serv)) < 0) {
            puts("Potrzebne prawa roota do bindowania portów poniżej 1024!");
            exit(EXIT_FAILURE);
      }
      printf("Zbindowane..");
      /* listen(), accept(), close(sock) .. */
      return EXIT_SUCCESS;
}
```

```
pako@rsbac-host:~$ ./con
Potrzebne prawa roota do bindowania portów poniżej 1024!
```

```
secoff@rsbac-host:~$ attr_set_user -a pako min_caps NET_BIND_SERVICE
```

```
pako@rsbac-host:~$ ./con
Zbindowane..
```

Moduł CAP

Wybrane prawa: **SYS_ADMIN**

```
#include <sys/mount.h>
#include <stdlib.h>

int main(void)
{
      if (mount("/dev/hda2", "hda", "ext2", 0, NULL) < 0) {
            puts("Nie może zamontować /dev/hda2 (wymagane prawa roota)!");
            exit(EXIT_FAILURE);
      }

      puts("Zamontowane.");

      return EXIT_SUCCESS;
}
```

```
pako@rsbac-host:~$ ./mount
Nie można zamontować /dev/hda2 (wymagane prawa roota)!
```

```
secoff@rsbac-host:~$ attr_set_user -a pako min_caps SYS_ADMIN
```

```
pako@rsbac-host:~$ ./mount
Zamontowane.
```

Co jeszcze?

- Moduł Jail.
- Moduł PAX.
- Moduł FF.
- Moduł Dazuko.
- Moduł RES.
- Opcje GENeral:
  - **Fake Root UID,**
  - **Linux DAC Disable,**
  - **rsbac_cap_process_hiding.**

Implementacja polityki bezpieczeństwa dla Apache'a

```
#!/bin/bash

APACHE="/usr/sbin/apache"
APACHE_CFG="/etc/apache"
APACHE_WWW="/var/www"
APACHE_LOG="/var/log/apache"
APACHE_USER="www-data"

# Tworzymy nową rolę (initial) dla Apache'a
I_ROLE_APACHE=`rc_get_item list_unused_role_nr`
rc_copy_role 0 $I_ROLE_APACHE
rc_set_item ROLE $I_ROLE_APACHE name "Apache_initial"

# Tworzymy nową rolę (force) dla Apache'a
F_ROLE_APACHE=`rc_get_item list_unused_role_nr`
rc_copy_role 0 $F_ROLE_APACHE
rc_set_item ROLE $F_ROLE_APACHE name "Apache_force"

# Tworzymy typ NETOBJ dla szablonu Apache'a
APACHE_OBJ=`rc_get_item list_unused_netobj_type_nr`
rc_set_item TYPE $APACHE_OBJ type_netobj_name "Apache_NETOBJ"

# Tworzymy typy FD dla plików konfiguracyjnych, www i logów Apache'a
APACHE_FD_CFG=`rc_get_item list_unused_fd_type_nr`
rc_set_item TYPE $APACHE_FD_CFG type_fd_name "Apache_conf_FD"
APACHE_FD_WWW=`rc_get_item list_unused_fd_type_nr`
rc_set_item TYPE $APACHE_FD_WWW type_fd_name "Apache_www_FD"
APACHE_FD_LOG=`rc_get_item list_unused_fd_type_nr`
rc_set_item TYPE $APACHE_FD_LOG type_fd_name "Apache_log_FD"
```

Implementacja polityki bezpieczeństwa dla Apache'a

```
# Tworzymy nettemporary (szablon) dla połączeń z Apachem
net_temp new_template 300
net_temp set_address_family 300 INET
net_temp set_name 300 "Apache_nettemp"
net_temp set_type 300 STREAM
net_temp set_valid_len 300 0
net_temp set_protocol 300 TCP
net_temp set_min_port 300 80
net_temp set_max_port 300 80
# Przypisujemy szablonowi netobiekt Apache'a (Apache_NETOBJ)
attr_set_net NETTEMP rc_type $APACHE_OBJ 300

# Przypisujemy roli Apache'a (initial) prawa dla typu General_FD, plików konfiguracyjnych i logów
rc_set_item ROLE $I_ROLE_APACHE type_comp_fd 0 APPEND_OPEN CHDIR CLOSE CREATE DELETE GET_STATUS_DATA READ
READ_OPEN SEARCH WRITE WRITE_OPEN MAP_EXEC
rc_set_item ROLE $I_ROLE_APACHE type_comp_fd $APACHE_FD_CFG CHDIR CLOSE GET_STATUS_DATA READ READ_OPEN
SEARCH
rc_set_item ROLE $I_ROLE_APACHE type_comp_fd $APACHE_FD_LOG APPEND_OPEN GET_STATUS_DATA SEARCH WRITE

# Przypisujemy roli Apache'a (force) prawa dla typu General_FD, plików www i logów
rc_set_item ROLE $F_ROLE_APACHE type_comp_fd 0 CHDIR GET_STATUS_DATA SEARCH
rc_set_item ROLE $F_ROLE_APACHE type_comp_fd $APACHE_FD_WWW CHDIR CLOSE GET_STATUS_DATA DELETE READ
READ_OPEN READ_WRITE_OPEN RENAME SEARCH WRITE WRITE_OPEN
rc_set_item ROLE $F_ROLE_APACHE type_comp_fd $APACHE_FD_LOG APPEND_OPEN CHDIR CLOSE CREATE DELETE
GET_STATUS_DATA READ READ_OPEN SEARCH WRITE WRITE_OPEN
```

Implementacja polityki bezpieczeństwa dla Apache'a

```
# Przypisujemy prawa roli (initial) Apache'a dla typów General_NETOBJ i Apache_NETOBJ
rc_set_item ROLE $I_ROLE_APACHE type_comp_netobj 0 CREATE MODIFY_SYSTEM_DATA
rc_set_item ROLE $I_ROLE_APACHE type_comp_netobj $APACHE_OBJ BIND LISTEN

# Przypisujemy prawa roli (force) Apache'a dla typów General_NETOBJ i Apache_NETOBJ
rc_set_item ROLE $F_ROLE_APACHE type_comp_netobj 0 READ WRITE ACCEPT SEND RECEIVE
rc_set_item ROLE $F_ROLE_APACHE type_comp_netobj $APACHE_OBJ GET_STATUS_DATA MODIFY_SYSTEM_DATA NET_SHUTDOWN

# Nadajemy użytkownikom (General_FD) możliwość łączenia się z Apachem po localhoście
rc_set_item ROLE 0 type_comp_netobj 3 CONNECT SEND RECEIVE

# Nadajemy typy katalogom z plikami konfiguracyjnymi, www oraz logami
attr_set_file_dir FD $APACHE_CFG rc_type_fd $APACHE_FD_CFG
attr_set_file_dir FD $APACHE_WWW rc_type_fd $APACHE_FD_WWW
attr_set_file_dir FD $APACHE_LOG rc_type_fd $APACHE_FD_LOG

# Przypisujemy Apacheowi utworzone role initial i force
attr_set_file_dir FD $APACHE rc_initial_role $I_ROLE_APACHE
attr_set_file_dir FD $APACHE rc_force_role $F_ROLE_APACHE

# Dodajemy prawo do zmiany właściciela procesu Apache'a na użytkownika www (www-data)
auth_set_cap FD add "/usr/sbin/apache" $APACHE_USER
# Ustawiamy rolę domyślną dla użytkownika www na rolę force Apache'a
attr_set_user $APACHE_USER rc_def_role $F_ROLE_APACHE
```

Implementacja polityki bezpieczeństwa dla Apache'a

Przykładowe testy utworzonej struktury bezpieczeństwa (1,2)

```
pako@rsbac-host:~$ telnet localhost 80
Trying 127.0.0.1...
Connected to debian-selinux.
Escape character is '^]'.
GET /
Witamy!
Connection closed by foreign host.
pako@rsbac-host:~$ cat /var/www/index.html
0000000156|rsbac_adf_request(): request SEARCH, pid 1461, ppid 1345, prog_name cat, prog_file /bin/cat, uid
1000, target_type DIR, tid Device 03:65 Inode 473210 Path /var/www, attr none, value none, result NOT_GRANTED
by RC
cat: /var/www/index.html: Operacja niedozwolona
```

/var/www/test3.php:

```
<?php
      exec("ls");
?>
```

```
pako@rsbac-host:~$ telnet localhost 80
Trying 127.0.0.1...
Connected to debian-selinux.
Escape character is '^]'.
GET /test3.php
0000000178|rsbac_adf_request(): request EXECUTE, pid 1522, ppid 1434, prog_name apache, prog_file /
usr/sbin/apache, uid 33, remote ip 127.0.0.1, target_type FILE, tid Device 03:65 Inode 798167 Path /
bin/bash, attr none, value none, result NOT_GRANTED by RC
Connection closed by foreign host.
```

Implementacja polityki bezpieczeństwa dla Apache'a

Przykładowe testy utworzonej struktury bezpieczeństwa (3)

/var/www/test.php

```
<?php
      readfile("/proc/version");
?>
```

```
pako@rsbac-host:~$ telnet localhost 80
Trying 127.0.0.1...
Connected to debian-selinux.
Escape character is '^]'.
GET /test.php
0000000192|rsbac_adf_request(): request READ_OPEN, pid 1435, ppid 1431, prog_name apache, prog_file /
usr/sbin/apache, uid 33, remote ip 127.0.0.1, target_type FILE, tid Device 00:03 Inode 4026531843 Path /
proc//version, attr open_flag, value 1, result NOT_GRANTED by RC
<br />
<b>Warning</b>:  readfile(/proc/version): failed to open stream: Operation not permitted in
<b>/var/www/test.php</b> on line <b>2</b><br />
Connection closed by foreign host.
```

Podsumowanie

- Gdzie używać RSBACa?
- Gdzie szukać pomocy?
  - **IRC – freenode, kanał #rsbac,**
  - **mailista:** https://www.rsbac.org/mailman/listinfo/rsbac,
  - **www.rsbac.org.**
- Adamantix (http://adamantix.org).
- www.rsbac.org - get it!
- Pytania?