



Compass Security

Argus PitBull Research

August 3, 2001

Document name:	Summary-Argus.doc
Version:	V 1.0
Author:	Ivan Buetler, Compass Security AG
References:	PitBull Basic Training
Date of delivery:	August 3, 2001
Document state:	PUBLIC

GLÄRNISCHSTR. 7
POSTFACH 1671
CH-8640 RAPPERSWIL

Tel. +41 55-214 41 60
Fax +41 55-214 41 61
info@csnc.ch www.csnc.ch



CONTENT

1	INTRODUCTION.....	1
1.1	<i>About the Author</i>	2
1.2	<i>Why this Article?</i>	2
1.3	<i>Little Notice</i>	2
1.4	<i>My Summary</i>	3
2	THEORIE	4
2.1	<i>What is a PitBull?</i>	4
2.1.1	Building Restrictions	4
2.1.2	MAC	5
2.1.3	DAC	6
2.1.4	TCB	6
2.1.5	Authorization	6
2.2	<i>Bypass Restrictions</i>	7
2.2.1	Privileges	7
2.2.2	Authorization	8
2.2.3	SecurityGate and ExecEnv	8
3	BASICS.....	9
3.1	<i>Sensitivity Labels</i>	9
3.1.1	Command Overview	9
3.1.2	Directory SL's	9
3.1.3	File SL's	9
3.1.4	Partition Directories	9
3.1.5	Change Label Encoding File (LEF)	10
3.1.6	SL Rules	10
3.1.7	Process SL	11
3.2	<i>Clearance</i>	11
3.2.1	Command Overview	11
3.2.2	See and Set Process Clearance	11
3.3	<i>Kernel</i>	12
3.3.1	Commands Overview	12
3.3.2	setkat	12
3.3.3	netrules	12
3.3.4	Kernel Flags (getseconf / setseconf)	12
3.3.5	Set Kernel Max and Min Label (setsyslab)	12
3.3.6	Show Kernel Security Flags	13
3.3.7	Set Kernel Security Flags	14
3.4	<i>Privileges</i>	15
3.4.1	Command Overview	15
3.4.2	Rules	15
3.4.3	Checking Privileges	15
3.4.4	Inheritance	16
3.4.5	Privileges Directive	16
3.4.6	Life Time Privileges	16
3.4.7	See Process Privileges	16
3.5	<i>Authorization</i>	17
3.5.1	Command Overview	17



3.5.2	Access Authorization	17
3.5.3	Privilege Authorization	17
3.5.4	Checks	17
3.5.5	Exit Codes	18
3.6	<i>Networking (MSN)</i>	18
3.6.1	Command Overview	18
3.6.2	RIPSO/CIPSO	18
3.6.3	Rules	18
3.6.4	Commands	19
3.7	<i>User Management</i>	19
3.7.1	Command Overview	19
3.7.2	Add User	19
3.8	<i>Auditing</i>	20
3.8.1	Command Overview	20
3.8.2	Events	20
3.8.3	Catch the Audit	20
3.8.4	Audit Related File Security Flags	20
3.9	<i>Administration</i>	21
3.9.1	Device Management	21
3.9.2	Operation Modes	21
3.9.3	File Security Flags	21
3.9.4	Trusted Library Path	22
3.9.5	Filesystem Conversion	22
3.10	<i>Integrity Checking</i>	22
3.10.1	Command Overview	22
3.10.2	Backup/Restore	23
3.10.3	Startup	23
3.11	<i>Utilities</i>	24
4	COMMANDS.....	26
4.1	<i>/sbin/*</i>	26
4.2	<i>getlicense -d</i>	27
4.3	<i>getrunmode</i>	27
4.4	<i>See Binary Security Flags</i>	27
4.5	<i>tracepv</i>	28
5	APPENDIX SYSTEM FILES.....	29
5.1	<i>LabelEncodings</i>	29
5.2	<i>azdb (Authorisation DB)</i>	31
5.3	<i>Clearance</i>	34
6	APPENDIX LAB	35
6.1	<i>Sensitivity Labels</i>	35
6.2	<i>SL Inheritance</i>	37
6.3	<i>MLS / Secure Networking</i>	38
6.4	<i>Authorizations</i>	39
7	APPENDIX APACHE EXAMPLE	43
7.1	<i>Compartment Design for Apache</i>	43
7.2	<i>Least Privilege</i>	43



7.3	<i>Consult Privileges</i>	43
7.4	<i>Authorization</i>	44
7.5	<i>Execenv</i>	44
7.6	<i>Secure Gate</i>	44
7.7	<i>Step by Step Procedure</i>	44
7.7.1	Create Restrictions	44
7.7.2	Apply Privileges (bypass the restrictions)	45
7.7.3	Apply Authorization (bypass the restrictions)	45
7.7.4	Start Apache by ExecEnv	48
7.7.5	Secure Gate Example	49



1 Introduction

This article briefly summarizes my personal highlights during the PitBull course at ARGUS Systems. It won't allow you to understand everything, but might give you a short introduction to the product

PitBull offers the following features:

- Build Restrictions MAC, DAC, TCB, Authorization
- Bypass Restrictions (processes) Privileges
- Bypass Restrictions (users) Authorization

New Themes:

- Inheritance of privileges and authorizations
- Kernel settings and influences
- Network SL labelling for incoming and outgoing packets
- Auditing
- Intrusion Detection on files (like Tripwire) -> check integrity
- SecurityGate (communications between 2 compartments)
- ExecEnv (start process at a certain SL even with an non-author. user)

This course bases on:

```
maquette ibuetler > uname -a
SunOS maquette 5.7 Argus_Enhanced_Security_3.0_MU4PLUS sun4u sparc
SUNW,Ultra-5_10
maquette ibuetler >
```

Important ARGUS files :

- /sbin
- /etc/security
- /opt/gibraltar
- /etc/asn



1.1 About the Author

Ivan Buetler is working for Compass Security, Switzerland. Compass is focussed on Pen-Tests and Security Reviews, also called “Security Assessments”.

Ivan Buetler
ivan.buetler@csnc.ch

<http://www.csnc.ch>

1.2 Why this Article?

Compass Security was asked to perform a B1 PitBull security check. Therefore we got a test machine from Argus Switzerland and I played around with this magic device. After a couple of weeks the real client project started, but I felt uncomfortable to perform the check by my own. I asked Argus for a technical consultant and we did the pen-test on PitBull together.

After this first touch with PitBull, I decided to follow the Argus training to gain a better understanding on what is going on.

This article describes my notices in a very short form. It's not intended to be a fully and nice to read document about PitBull.

If you think I missed some important aspects, wrote something wrong or any other feedback is really appreciated.

Happy reading

Ivan

1.3 Little Notice

PitBull offers his product with additional tools. They are not necessary to understand the behaviour of PitBull. Personally I am not really convinced of all the tools. I like the SecurityGate and ExecEnv very much. These tools are needed and cool.

But I don't like the idea, Argus offers a webentry infrastructure with reverse proxy, central authentication service and talks about HTTP, cookies, URL redirection during the training. This is an application you can run on PitBull if you like (even there are more professional services available in the market out there).

If you want to understand PitBull, don't be confused by themes like UDE, SecureCGI, Secure Authentication Module. These additional Modules from PitBull try to enable Argus per default to a webentry component.



1.4 My Summary

I like the PitBull very much. I am convinced, you will be able to configure PitBull to be very secure, even the hacking contest was lost a couple of weeks/months ago and Argus had to pay the USD 100'000.

When I was talking to a RACF specialist, I recognized tremendous amount of theory and mechanisms almost similar. I will go forward to really better understand the difference between PitBull and RACF. In every case, it sounds quite interesting.

I would recommend using PitBull for high confidential data applications. It could be configured high secure.

The major disadvantage in my eyes is the missing trace capabilities. If something hangs, you better tell your wife not being with here during the weekend.

Some people take it a disadvantage to have a complex system. Personally I don't agree. You need some time to understand the rules within this new matrix. But everyone with in depth unix knowledge should easy understand inheritance on fork and exec. PitBull enhanced the security attributes you can set to a file and therefore some more inheritance tasks will be performed than in a standard unix. But everyone who understands the matrix will love this box.

I want to thank ARGUS for giving me the chance to play with their product in my lab.

At the end, I just want to confirm: if you don't take the time to use least privileges, you have little security more than a standard Unix box. Keep in mind, you need to configure B1 before production.



2 Theorie

2.1 What is a PitBull?

Standard Unix knows about 1 major restriction – file permissions. If the process belongs to a certain user and group, file permissions will control whether the process will be able to read/write/execute his task or not.

PitBull offers more flexibility (and complexity) of setting up restrictions. It's not only the file-permission which decides.

2.1.1 Building Restrictions

PitBull offers the following methods to set up restrictions:

- MAC (mandatory access control)
- DAC (discretionary access control == file permissions)
- TCB (trusted computing base)
- Auth (authorization)

Once you have set up your restrictions, you need to bypass them by least privileges. You can do this by:

- Privileges (enable processes to overwrite restrictions)
- Authorization (allow users to run processes with select sets of privileges)

Now you have heard the most important “words” in PitBull. Do you understand? Probably not, because you don't know what MAC, DAC, TCB is. You have no feeling about this technology. Therefore I will just follow up with very little description in that. But before so, I want to summarize something very important:

Where in a standard Unix you need to bypass the DAC restrictions, you need to bypass in PitBull

1. MAC
2. DAC
3. TCB
4. Authorization

in this order!!!! before access to something will be allowed. It could happen, you have proper DAC, TCB and Authorization, but you are not allowed to access your file, because MAC is not properly set up. This makes it very heavy to trace errors. Unfortunately the error messages are more than “little”. If your application doesn't run, only god will help you to find out why.

I suggested Argus to control the error message behaviour by a flag, the administrator will be able to enable when it's needed. Because you have no clue, why something is works or not, if you inherit your conclusion from the error message.

But now to the more fancy part. What are this MAC, DAC, etc?

2.1.2 MAC

MAC is a matrix. If you place your file somewhere on your filesystem, it will be saved somewhere within the matrix. (sounds like the matrix movie)

Proper B1 applications will have a concept in where you want to place your files to, because the matrix follows certain rules.

This means, you cannot only set uid/gid attributes to your file, you will be able to set this matrix attributes to the file as well. Argus has expanded the attributes of files, if you want.

Such a MAC attribute is identified as sensitivity label (SL). A sensitivity label contains 2 values

- a) compartment (A-F in the example below)
- b) classification (IMPL_LOW – TOPSEC in the example below)

Classification	A	B	C	D	E	F
TOPSEC						
SEC		apache				
CONF				oracle		
RESTRICT						
SENS						
PUBLIC						
UN_CL						
IMP_LOW						

ARGUS offers 1024 compartments, not limited to A-F. ARGUS offers to rename and create new classifications as well. The classifications above are standard.

There are a couple of rules within the matrix. I don't want to copy/paste the training, but at least you should understand that:

- files or processes running in compartment A can't see other files from other compartments (something like chroot)

- its only allowed to read files within the same compartment, if the process runs in a higher or equal classification. The correct syntax would be: the process SL (sensitivity label) must dominate file SL
- Its only allowed to read files within the same compartment, if the process runs in the equal classification. The correct syntax is: the process SL must equal file SL
- its only allowed to execute a file, if the starting process has a higher or equal classification as the file. The correct syntax is: to execute a file, process SL must dominate file SL (same as for read)

If you keep this rules in place, how would you configure your apache webserver?

- binaries
- conf
- logs
- htdocs

This was an example of the course. See 7.1 for the lab.

2.1.3 DAC

DAC is the same as file permissions in standard Unix. I wont explain it here furthermore.

2.1.4 TCB

The „Trusted Computing Base“ is a flag you will be able to apply to a file. If this flag is set (FSF_TCB), it cannot be created, altered, deleted during normal operation.

the TCB protects system components that enforce security.

As in other Unix known, you have multiple modes. Single-User mode, Run-Level modes etc. It is necessary to drive into “operation mode”, before the files containing the FSF_TCB flag will be changeable, deleteable, etc.

2.1.5 Authorization

Files have 2 sets of authorization

Access Authorization	(who can execute which binary)
Privilege Authorization	(by adding & filtering privileges on executed binaries)



2.2 Bypass Restrictions

I will add my statement below again: If you have set up properly your restrictions, you need to open certain restrictions afterwards making your application run.

Keep in mind, the restrictions will be checked in the following order:

1. MAC
2. DAC
3. TCB
4. Authorization

It's might a good idea (if you play with your system) to have no DAC, no TCB and no Authorization restrictions in place first and only play with MAC. As mentioned above, the error messages don't inform, why a certain task will fail. This makes it very hard to trace.

You have 2 options to bypass restrictions

- Privileges
- Authorization

2.2.1 Privileges

Privileges enable processes to bypass security. If you want to enable a service to bind a port, the binary needs the "PV_ASN_PORT" privilege.

Privileges are hard-coded into PitBull. You can't customize privileges

There are groups of privileges already pre-defined in PitBull. These are called "categories"

Privileges are sorted within a hierarchy.

Privileges are inherited by fork(); from parent

Privileges are gained from binary settings on exec();

In a standard Unix, you have only 1 privilege == root. In PitBull, this privilege was split into multiple (about 130) privileges.

Examples of privileges:

- to execute the command "secls" you need PV_LEF privileges
- to bind a port "PV_ASN_PORT" is required

There are a couple of more handy mechanisms, how a process will gain privileges. As the matrix in MAC, the privilege inheritance follows a rule-based system you need to know in the very detail, in order to perform configuration with least privileges. (or you open too many privileges)

2.2.2 Authorization

I introduced into Authorization in chapter 2.1.5 as a method to build restrictions. Now its suddenly a bypass restriction?

I'll try to explain

Authorizations are associated to users.

Users might have multiple authorizations

It is possible to create new authorizations (privileges are hard-coded)

Files have 2 sets of authorizations

- Access Authorization APS
- Privilege Authorization PAS

And that's where the aspect of bypass restrictions comes in (PAS).

The process will not only inherit privileges from his father. The new process might gain additional privileges, because the user who starts the executable has the authorization to do so.

Example:

Authorization of the user ibuetler
ibuetler: LOGIN, MKFS

mkfs binary settings:
PAS: MKFS
APS: PV_DAC_R, PV_DAC_W, PV_MAC_R, PV_MAC_W

The shell of ibuetler has not the privilege mkfs needs to perform his tasks. The PV_DAC_W and PV_MAC_W are powerful privileges you won't apply to a simple users shell. So you can't inherit this privileges from the parent process at the time the user wants to perform a mkfs.

Therefore the mkfs process reads the APS settings from the binary and if the user has the required MKFS authorization, the mkfs process will have the required privilege.

From this point of view, authorizations are not only needed for building up restrictions. They are very important when you want to bypass the restrictinos based on authorizations.

2.2.3 SecurityGate and ExecEnv

This 2 tools are very cool from Argus.

SecurityGate will help you to let your application talking to another component in another compartment (which is not possible by default)

ExecEnv will allow you to start a process you don't have privileges and authorizations to. It's a mechanism you will enable the dummy user just to start and stop the application and nothing more.

3 Basics

3.1 Sensitivity Labels

3.1.1 Command Overview

chsl	change sensitivity label on a file
getsl	show sl of process
getsyslab	show kernel max and min sl
setsl	change sl of a process
setsyslab	set kernel max and min labels
slop	sl comparisation
ttygmt	modify sl of tty
devseconfig	"devseconfig: manage MIN;MAX;IL SL on device files"
getclear	get clearance of a user default login SL
netrule	manage network interface behavior (SL,...)
setclear	change users clearance and default login SL

3.1.2 Directory SL's

Directories have 2 SLs

- min SL
- max SL

all files in dir must be within directory SL range

```
/tbin/secsl
/tbin/secsl -s

/tbin/chsl

z.B.
    /tbin/chsl -l <MIN-SL> -h <MAX-SL> DirName
```

3.1.3 File SL's

Files have only 1 SLs

- min SL
- max SL (equal max SL)

min SL and max SL are **always equal**

3.1.4 Partition Directories

pdir: kernel creates hidden sub-dirs



- if not already existing
- called psdir
- MIN SL and MAX SL equal process ESL

kernel redirects process to psdir

- cannot be seen by regular process

Process Mode

Attribute of process could be

- virtual mode
- real mode

```
pdmode - returns the current partitioned directory access mode or runs a command with a specified partitioned directory access mode
```

```
/tbin/pdmode -v sh  
/tbin/pdmode -r /bin/cpio
```

3.1.5 Change Label Encoding File (LEF)

- /etc/security/clear
- /etc/security/tty
- /etc/security/integrity/*.*
- /etc/security/device_levels

```
Check /etc/security/LabelEncodings
```

```
chklef - verify a LabelEncodings file
```

3.1.6 SL Rules

Rule	Description
read access to the directory	process ESL must dominate dir MIN SL
write access to the directory	process ESL must dominate dir MIN SL && dir MAX SL must dominate process ESL
cread directory	process ESL inherited to MIN SL and MAX SL
init process SL	hardcoded ESL = IMPL_LO MaxCL = TS ALL MinCL = IMPL_LO



shell process SL	/etc/security/clear DEFAULT_CLEAR:IMPL_LO:TS ALL:IMPL_LO isso:IMPL_LO:TS ALL:IMPL_LO sa:IMPL_LO:TS ALL:IMPL_LO so:IMPL_LO:TS ALL:IMPL_LO
------------------	--

3.1.7 Process SL

```
/tbin/getsl
/tbin/getsl -s

/tbin/setsl

login: ibuetler
Password:
Last login: Thu Jun 21 09:49:53 from pts/14
maquette ibuetler > getsl $$
25997:
    EFFECTIVE SL:      IMPLEMENTATION LOW INTERNET
    MINIMUM CLEARANCE: IMPLEMENTATION LOW
    MAXIMUM CLEARANCE: TOP SECRET ALL
maquette ibuetler >
maquette ibuetler > setsl -a "SEC C" $$
25997:
    EFFECTIVE SL:      SECRET COMP_C
    MINIMUM CLEARANCE: SECRET COMP_C
    MAXIMUM CLEARANCE: SECRET COMP_C
maquette ibuetler >
```

3.2 Clearance

3.2.1 Command Overview

getclear	get clearance of a user default login SL
setclear	change users clearance and default login SL
setps -l	clearance of process

3.2.2 See and Set Process Clearance

```
/tbin/secps -l $$
/tbin/setsl -M „TS A B“ $$
/tbin/setsl -m “IMPL_LOW” $$
```

3.3 Kernel

3.3.1 Commands Overview

asninit	save/restor ASN (netrule) settings from kernel
getseconf	show kernel security flags
setkat	load azdb into kernel
setseconf	set kernel security flags
setsyslab	set kernel max and min labels
getsyslab	show kernel max and min sl

3.3.2 setkat

set kernel (load azdb changes into kernel)

3.3.3 netrules

All changes with the command /tbin/netrule will immediatly loaded into the kernel

3.3.4 Kernel Flags (getseconf / setseconf)

/tbin/getseconf	get current kernel flags
/tbin/setseconf	set kernel flags

There is a ASN (network and packet checker) kernel security flag to:

enable/disable ASN on a machine

Show Flag: /tbin/getseconf

3.3.5 Set Kernel Max and Min Label (setsyslab)

NAME
setsyslab - set the kernel maximum and minimum labels

SYNOPSIS
setsyslab

AVAILABILITY
ASGec2sf

DESCRIPTION
setsyslab is used to set the kernel minimum sensitivity label (SL), maximum SL, minimum information label (IL), minimum integrity label (TL), maximum integrity label, and

```

default integrity label. These labels are used by the kernel for assigning actual SL/IL/TL classification and compartment information to SLs, ILs, and TLs that use a format field of 1 or 2. The values of the SLs, IL, and TLs are taken from the label encodings file. setsyslab is run automatically during the system boot sequence.

FILES
/etc/security/LabelEncodings ASCII label encodings file.

AUTHORIZATIONS
Only users with the SETSYSLAB authorization can run setsyslab.

NOTES
TL operations are not currently supported.

SEE ALSO
setsyslab(1a)

SunOS 5.7      Last change: 21 July 1999      1

```

Label names not loaded into kernel
 SLSL and SHSL are loaded into kernel

```

setsyslab - set the kernel maximum and minimum labels

$ /sbin/setsyslab
System minimum SL: IMPLEMENTATION LOW
System maximum SL: TOP SECRET ALL
System minimum IL: IMPLEMENTATION LOW
$

Only users with the SETSYSLAB authorization can run setsyslab

```

getsecconf == show kernel security flag

3.3.6 Show Kernel Security Flags

```

maquette tbin > getsecconf

Current Security Flag Status      (operational mode)
ASN [A]:                ENABLED
AZROOT [Z]:             DISABLED
MLS [B]:                ENABLED
CAPABILITIES [C]:      DISABLED
GIBRALTAR [K]:          DISABLED
IL_FLOAT [I]:           DISABLED
SL_ENFORCEMENT [J]:    ENABLED
PRIV [P]:               ENABLED
PVROOT [V]:             ENABLED
STRPUSH [U]:            ENABLED
SU_EMUL [R]:            DISABLED
TCB [E]:                ENABLED
TDE [D]:                DISABLED

```



```
TL_ENFORCEMENT [G]:  DISABLED
XIL [H]:             DISABLED
XPRIVS [T]:         DISABLED
XSL [X]:            DISABLED
```

3.3.7 Set Kernel Security Flags

Check out Appendix F of PutBull script

```
NAME
    setseconf - change the Argus kernel security flags

SYNOPSIS
    setseconf { -M | -O } [ -FLNuv ] [ -f <ffile> ] [ -Ax ] [
    -Bx ] [ -Cx ] [ -Dx ] [ -Ex ] [ -Gx ] [ -Hx ] [ -Ix ] [ -Jx
    ] [ -Px ] [ -Rx ] [ -Tx ] [ -Ux ] [ -Vx ] [ -Xx ] [ -Zx ]

AVAILABILITY
    ASGec2sf

DESCRIPTION
    setseconf loads kernel security flag settings into the
    running kernel.  Some flags only have meaning if specific
    Argus security modules are installed, so care must be taken
    to manipulate only those flags for which modules have been
    installed.  Each Argus module properly enables its required
    security flags as part of its installation script, so secu-
    rity flags should not be changed unless the security policy
    of the site allows the modification.

    setseconf is run as part of the system startup scripts at
    each reboot.

    Either the -M or -O option must be specified.
```

```
maquette ibuetler > getsyslab
System minimum SL: IMPLEMENTATION LOW
System maximum SL: TOP SECRET ALL
System          IL: IMPLEMENTATION LOW
maquette ibuetler >
```

3.4 Privileges

3.4.1 Command Overview

chpv	change privilege on a file
findpv	find required privilege of a process
getpv	show privilege of a process
setpv	change privilege of a process
tracepv	trace privileges on a process

- Associated with processes
- Enables process to perform otherwise restricted actions

The privilege enables a process to go over MAC restrictions. DAC will still apply.

4 process privileges	
3 file privileges	

EPS	effective privilege set
MPS	maximum privilege set
LPS	limiting privilege set, also called the absolute privilege set

LPS -> MPS -> EPS (privilege inheritance)

Its required to have the necessary privilege inherited from LPS, if this is used in EPS.

3.4.2 Rules

Process may remove a privilege from any privilege set

- if not in lower level priv.set

Process may never add to its LPS

Process may add to MPS if

- Privilege in LPS
- Process has PV_PV_PROC in EPS

copy: privileges are empty (unless destination file exists and has privileges)

move: privileges are inherited from source

3.4.3 Checking Privileges

PV_DAC_W_NS there?

PV_DAC_W there?

PV_DAC there?



Privileges check continues up the hierarchy.

3.4.4 Inheritance

On fork:

- child process inherits all privileges from parent

On exec:

- settings on binary affect process

3.4.5 Privileges Directive

PROXY PRIVILEGES	proxy privs to post-exec (filter)
INNATE PRIVILEGES	added privs to post-exec MPS
AUTHORIZED PRIVILEGES	added privs to MPS, if authorized

FSF_EPS, automatically placed MPS to EPS

3.4.6 Life Time Privileges

- in memory only
- goes away if process dies

3.4.7 See Process Privileges

```
login: ibuetler
Password:
Last login: Tue Jun 19 11:48:41 from pts/2
maquette: ~ > getpv $$

maquette: ~ > su - isso
Sun Microsystems Inc. SunOS 5.7 Generic October 1998
$ /tbin/getpv $$

$ /tbin/setpv +a pv_root $$
$ /tbin/getpv $$
----EFFECTIVE PRIVILEGES----
PV_ROOT
----MAXIMUM PRIVILEGES----
PV_ROOT

$
```



3.5 Authorization

3.5.1 Command Overview

azcheck	check if user has authorization (true/false respond)
azlist	background infos about authorizations (azdb)
bootauth	used during startup to verify if user has boot auth
chauth	change authorization on a file
chazdb	utility to change authorization (front end to azdb)
setauth	change authorization on a process

Files have 2 sets of authorizations

- access authorization (control execute access to command)
- privilege authorization (control feature provided by command)

Used to implement roles on a system.

Unlike privileges, you can edit authorizations on your system

Configure authorizations in /etc/security/azdb and load it by /sbin/setkat (set kernel). It is required to have SETKAT authorization

It is possible (no business case known) where you do additional rules to /etc/security/las and azdb won't be used any more.

3.5.2 Access Authorization

decides who can execute which binary

if the bin has more auths in ASS, user must have at least one of those auths

3.5.3 Privilege Authorization

by adding and filtering privileges on executed binary. This means, the privilege authorization decides what functions the user might be able to use on a binary. Let's assume the useradd has 4 options, where 2 options are for everybody and the other 2 options are only for a restricted number of users. This could be done by privilege authorizations

3.5.4 Checks

AAS (Access Authorization) and PAS (Privilege Authorization) checks are external to code

1. read user name of process



2. search azlist to see, if user has required authorization
3. if not found, search if user has dominating authorization
4. continue up to hierarchy

3.5.5 Exit Codes

Scripts have the following (to unix opposite) values

success:	exit code 1
fail:	exit code 0

3.6 Networking (MSN)

3.6.1 Command Overview

asninit	save/restor ASN (netrule) settings from kernel
netrule	manage network interface behavior (SL,...)

3.6.2 RIPSO/CIPSO

ASN = Advanced Secure Networking

1. Income traffic will assigns label
2. Outgoing traffic will held SL info, if configured so

ASN is based on:

RIPSO (Revised IP Security Options)
CIPSO (Commercial IP Security Option) / also called CSL Common Security Label

3.6.3 Rules

Host Rules:

IP address / subnet AND Protocol AND Port
Host rules applies to all interfaces

Network Rules:

Network Interface (physical or virtual)

Both have same parameters



Host rules take precedence

3.6.4 Commands

```
/tbin/netrule hl Host list
/tbin/netrule il Interface List
/tbin/netrule h-i Remove Host Rule
```

```
/tbin/asninit save <filename> Save current rules from kernel into a database
/tbin/asinit load <filename> Load rules from database
```

3.7 User Management

3.7.1 Command Overview

azcheck	check if user has authorization (true/false respond)
bootauth	used during startup to verify if user has boot auth
getclear	get clearance of a user default login SL
loginblock	modify blocked users
setclear	change users clearance and default login SL

3.7.2 Add User

SA	authorized to create/delete users
ISSO	modify security settings
SA	cannot change password

```
$ /usr/sbin/useradd -u 8888 -g staff -d /export/home/i999 -s
/usr/local/bin/bash -c cool -m i8888
6 blocks
$
```

```
usage: useradd [-u uid [-o] | -g group | -G group[[,group]...] | -d dir |
-s shell | -c comment | -m [-k skel_dir] | -f inactive | -e expire ] login
useradd -D [-g group | -b base_dir | -f inactive | -e expire ]
```

1. Login as SA and type useradd
2. Login as ISSO and add user to /etc/security/azdb
3. Load new azdb to kernel by setkat
4. Try to login by the new assigned user (it should work, if LOGIN auth for user given)

3.8 Auditing

3.8.1 Command Overview

auctlmod	/sbin/auctlmod - modify audit system files (vi)
auditd	/usr/sbin/auditd
auditdconfig	/usr/sbin/auditconfig
auditreduce	/usr/sbin/auditreduce
auditstat	/usr/sbin/auditstat
praudit	/usr/sbin/praudit

3.8.2 Events

Predefined audit events in /etc/security/audit_event

Predefined audit classes in /etc/security/audit_class

Predefined audit preselection mask in /etc/security/audit_control

1. Create Audit Events
2. Create Audit Groups, also called Classes
3. Configure preselection mask

Events can be in more groups/classes. A class can be specified for

- successful events
- failed events
- either

3.8.3 Catch the Audit

Each process has an audit preselection mask

- define which class to audit
- specify whether to fail, success or either
- configured in /etc/security/audit_control

3.8.4 Audit Related File Security Flags

FSF_AUDIT Mark file as part of the audit subsystem

FSF_MONITOR Audit, regardless of process audit mask



3.9 Administration

- device management
- system modes
- security flags
- trusted library path
- file system conversion
- important argus directories
- backup and restore
- startup and shutdown

3.9.1 Device Management

Device attribute database: /etc/security/device_levels

/tbin/devlvmgmt to update device
/tbin/devsecconfig set device to default

example: /tbin/devlvmgmt -I SLS -h SHSL /dev/console

3.9.2 Operation Modes

System operates in 2 modes:

- operation mode
- maintenance mode

In op-mode it is not possible to change kernel security flags, trusted library path and TCP objects
in m-mode it is allowed to do the tasks above and at and cron are disabled

Switch Mode by:
/sbin/init m Maintenance Mode
/sbin/init o Operation Mode

3.9.3 File Security Flags

Associated with files. Check out Appendix E of the PitBull Scripts

/tbin/secls -f
/tbin/chfsf

3.9.4 Trusted Library Path

Check for trusted libraries in

`/etc/security/libpath.txt`

```
maquette ~ > tlibadmin
tlibadmin: Library path table does not exist
maquette ~ > cat /etc/security/lib
lib      libpath.txt
maquette ~ > cat /etc/security/libpath.txt
#
# /etc/security/libpath.txt
#
# Trusted library paths should be entered below.  These paths will be loaded
# into the kernel when the system boots.  Path entries can later be updated by
# executing the following command:
#
# /sbin/tlibadmin -s
#
# All entries must begin with a forward slash ('/') or will be discarded.
# Blank lines or lines beginning with a '#' are completely ignored, while any
# other line that is not a path entry causes tlibadmin to generate an error
# message.
maquette ~ >
```

3.9.5 Filesystem Conversion

Add Argus UFS to a standard UFS (if standard disk will be included)

```
/usr/lib/fs/ufs/ufs2mufs <filesystem>
/usr/lib/fs/ufs/mufs2ufs <filesystem>
```

3.10 Integrity Checking

3.10.1 Command Overview

```
chkintegrity    verify crypto checksum on a file
makeidb         create integrity entry
```

Located in `/etc/security/integrity`

Stored into integrity database. This is all within `/etc/security/integrity/*`

Create Integrity Database

```
maquette ~ > find . -print | makeidb ibuetler.integrity
maquette ~ > head 20 ibuetler.integrity
20: No such file or directory
==> ibuetler.integrity <==
/export/home/peter|D|700|peter|other|*|*|*|IMPL_LO|TS ALL|IMPL_LO|IMPL_LO|||0 0 0 0|||
```

```

/export/home/peter/.profile|F|644|peter|other|74ea780e55498fcf500f0a736399059d0000006f|*|*|I
MPL_LO|IMPL_LO|IMPL_LO|IMPL_LO|||0 0 0 0|||
/export/home/peter/local.cshrc|F|644|peter|other|d4058893857a1aa3da4b6eb38f35c82c00000098|*|
*|IMPL_LO|IMPL_LO|IMPL_LO|IMPL_LO|||0 0 0 0|||
/export/home/peter/local.login|F|644|peter|other|c5625a07702954869e85c87477ec165500000245|*|
*|IMPL_LO|IMPL_LO|IMPL_LO|IMPL_LO|||0 0 0 0|||
/export/home/peter/local.profile|F|644|peter|other|dc584145f2df9818b1763be87795dd8d00000232|
*|*|IMPL_LO|IMPL_LO|IMPL_LO|IMPL_LO|||0 0 0 0|||
/export/home/peter/demo|D|777|peter|other|*|*|*|IMPL_LO|TS ALL|IMPL_LO|IMPL_LO|||0 0 0
0|||
/export/home/peter/demo/config|D|777|peter|other|*|*|*|IMPL_LO|TS ALL|IMPL_LO|IMPL_LO|||0
0 0 0|||
/export/home/peter/demo/config/execenv.conf|F|777|peter|other|1259bad16d0a98669c2244e518f509
43000003cf|*|*|IMPL_LO|IMPL_LO|IMPL_LO|IMPL_LO|||0 0 0 0|||
/export/home/peter/demo/config/gibraltar.conf|F|777|peter|other|6b345a1ca7eb33065dbe59de8526
ccb3000013d8|*|*|IMPL_LO|IMPL_LO|IMPL_LO|IMPL_LO|||0 0 0 0|||
/export/home/peter/demo/log|D|777|peter|other|*|*|*|IMPL_LO|TS ALL|IMPL_LO|IMPL_LO|||0 0 0
0|||
maquette ~ >

```

3.10.2 Backup/Restore

Normal Backup.

ufsdump/ufsrestore Attributes will be stored

cpio/tar: SL will not be stored to destination

Solution: restore (SL are lost) and perform integrity checking in fixing mode

3.10.3 Startup

Actions in rc* Scripts

- | | |
|----------------------------------|---------------------------|
| • set kernel authorization table | setkat |
| • load display security flags | setsecconf and getsecconf |
| • load sys_low && sys_hi labels | setsyslab |
| • load display KSF | setsecconf |
| • change to operation mode | init \$ |

Actions in rc*.d Directories

- | | |
|-------------------------------|--------------|
| • apply sec attributes to dev | devsecconfig |
| • set trusted lib path | tlibadmin |
| • get boot authorization | bootauth |
| • check integrity | chkintegrity |
| • initialise ASN | asninit |

3.11 Utilities

Tool	Description
EXECENV	start foreign processes the user has no priv and auth to Config /etc/security/execenv.conf
SLOP	compares 2 SL or give min or max SL of a file
SCRIPTS	Hardening scripts for apache and netscape webserver
SECURITY GATE	Gate between 2 compartments [CON A] <-> [SEC B] The security gate needs to dominate both labels Definition Security Gate in /etc/security/gibraltar.conf
Secure CGI	CGId Requires ARGUS plug-in modules Secure execution of CGI programs Server and scripts belong to different compartments Communication by Security Gate CGId dominates the scripts
UDE Secure Com. Enforcer	UDE = Upgrade / Downgrade Enforcer Enable communications between 2 or more processes with disjoint SLs Rules apply to: <ul style="list-style-type: none"> - rules in config file - SL of request (assigned by ASN = netruler) - URL of request (cookie, ...) - cookie (if set by Authd) HTTP Mode (SL and URL) HTTP Mode (cookie set) UDE == software router
AUTHD	supports any auth. programm



	session handled by encrypted md5 session cookie
TSSH	multilevel ssh connection based on openssh no changes to ssh client required
IPSEC	not implemented

4 Commands

4.1 /tbin/*

asninit	save/restor ASN (netrule) settings from kernel
auctlmod	auctlmod - modify audit system files
azcheck	check if user has authorization (true/false respond)
azlist	background infos about authorizations (azdb)
bootauth	used during startup to verify if user has boot auth
chauth	change authorization on a file
chazdb	utility to change authorization (front end to azdb)
chfsf	change file security flag on a file
chkintegrity	verify crypto checksum on a file
chklef	verify LabelEncodings syntax
chpv	change privilege on a file
chsl	change sensitivity label on a file
devlvlmgmt	get or set device's security flags
devseconfig	devseconfig: manage MIN;MAX;IL SL on device files
findpv	find required privilege of a process
foureyes	set up access rule with 4 eyes rules
getclear	get clearance of a user default login SL
getip	gets info back like "192.168.0.211 maquette hme0"
getlicense	show/write license info
getpv	show privilege of a process
getrunmode	show current run mode
getseconf	show kernel security flags
gets1	show sl of process
getsyslab	show kerlen max and min sl
led	label encoding daemon
loginblock	modify blocked users
makeidb	create integrity entry
netrule	manage network interface behavior (SL,...)
pdlink	link files across partintioned directories
pdmkdir	create partinioned directory
pdmode	show current partinioned directory settings
pdrmdir	delete partinioned directory
pdset	add directory to partitioned directory
rvi	restricted vi
secipcs	show IPC (interprocess communication)
secls	show file security flags
secps	show process security flags
seels	show binary security flags
setauth	change authorization on a process
setclear	change users clearance and default login SL
setkat	load azdb into kernel
setpv	change privilege of a process
setseconf	set kernel security flags
sets1	change sl of a process
setsyslab	set kernel max and min labels
slop	sl comparisation



sumargus	determine the checksum for a file
tlibadmin	tlibadmin manages the trusted paths (for linker)
tracepv	trace privileges on a process
ttymgmt	modify sl of tty

4.2 getlicense -d

```
maquette tbin > getlicense -d
expiry=20010714
products=ff3fffe3
company="Argus"
license=9c6d0ec1
maquette tbin >

-d == display
```

4.3 getrunmode

```
maquette tbin > getrunmode
system is currently in OPERATIONAL MODE
maquette tbin >
```

4.4 See Binary Security Flags

```
maquette bin > seels httpd
httpd
  FILE SECURITY FLAGS: 0x20
  SENSITIVITY LABEL
F:0x0, C:0x78, P:0x0
CMP: 10000000 0 0 0 0 0 0 0
CMP: 0 0 0 0 0 0 0 0
CMP: 0 0 0 0 0 0 0 0
CMP: 0 0 0 0 0 0 0 0
  INFORMATION LABEL
F:0x1, C:0x0, P:0x0
CMP: 0 0 0 0 0 0 0 0
CMP: 0 0 0 0 0 0 0 0
CMP: 0 0 0 0 0 0 0 0
CMP: 0 0 0 0 0 0 0 0
MRK: 0 0 0 0 0 0 0 0
  INTEGRITY LABEL
F:0x3, C:0x0, P:0x0
  PROXY PRIVILEGES: 0x0 0x0 0x0 0x0
  INNATE PRIVILEGES: 0x0 0x0 0x60 0x0
  AUTHORIZED PRIVILEGES: 0x0 0x0 0x0 0x0
  ACCESS AUTHORIZATIONS: 8888 0 0 0
  PRIVILEGE AUTHORIZATIONS 0 0 0 0
  CAPABILITY SET
  flags : 0x0
  read  : 0x0
  write : 0x0
  exec  : 0x0
maquette bin >
```



4.5 tracepv

```
maquette bin > tracepv
usage: tracepv [options] command [args...]
where options is any of the following
    -a          Apply discovered privileges to files.
    -e          Follow the exec system call.
    -f          Follow the fork system call.
    -g group    Group to run as.
    -h homedir  Home dir of process.
    -l          List opened files.
    -o file     Output to file, not stderr.
    -s label    Sensitivity Label to run at.
    -t          Add discovered library paths to /etc/security/libpath.txt
               and kernel.
    -T file     Add discovered library paths only to file.
    -u user     User to run as.
    -v          Verbose output.
maquette bin >
```



5 Appendix System Files

5.1 LabelEncodings

FILE	DIRECTORY
\$ /sbin/secls -s /etc/security/LabelEncodings	\$ /sbin/secls -s /etc/security
LabelEncodings	/etc/security
SENSITIVITY LABEL	MAXIMUM SENSITIVITY LABEL
TOP SECRET ALL	TOP SECRET ALL
\$	MINIMUM SENSITIVITY LABEL
	IMPLEMENTATION LOW
	\$

files have only 1 SL

```

$ pwd
/etc/security
$ cat LabelEncodings
*
* Comments can be placed in the encodings file any place a keyword can start.
* Comments begin with a * and continue to the end of the line.
*

*****
VERSION= ARGUS GIBRALTAR VERSION
*****

*****
CLASSIFICATIONS:
*****

    name= IMPLEMENTATION LOW;      sname= IMPL_LO; value= 0;
    name= UNCLASSIFIED;           sname= U;       value= 20;
    name= PUBLIC;                 sname= PUB;    value= 40;
    name= SENSITIVE;              sname= SEN;    value= 60;
    name= RESTRICTED;             sname= RES;    value= 80;
    name= CONFIDENTIAL;           sname= CON;    value= 100;
    name= SECRET;                 sname= SEC;    value= 120;    initial markings=
19;

    name= TOP SECRET;             sname= TS;     value= 140;    initial markings=
19;

*****
INFORMATION LABELS:
*****

WORDS:

    name= ALL;          sname= ALL;    compartments= 0-89;
    name= GIBRALTAR;   sname= GIB;    compartments= 1-5;
    name= COMP_A;      sname= A;      compartments= 1;
    name= COMP_B;      sname= B;      compartments= 2;
    name= COMP_C;      sname= C;      compartments= 3;
    name= COMP_D;      sname= D;      compartments= 4;
    name= INTERNET;    sname= INET;   compartments= 67;
    name= TADMIN;      sname= TADMIN; compartments= 68;
    name= *** PROTECT WITH GREAT CARE ***; sname= PROTECT; markings= 19;

REQUIRED COMBINATIONS:

```



```
COMBINATION CONSTRAINTS:

*****
SENSITIVITY LABELS:
*****

WORDS:

    name= ALL;          sname= ALL;          compartments= 0-89;
    name= GIBRALTAR;    sname= GIB;          compartments= 1-5;
    name= COMP_A;       sname= A;            compartments= 1;
    name= COMP_B;       sname= B;            compartments= 2;
    name= COMP_C;       sname= C;            compartments= 3;
    name= COMP_D;       sname= D;            compartments= 4;
    name= INTERNET;     sname= INET;         compartments= 67;
    name= TADMIN;       sname= TADMIN;        compartments= 68;

REQUIRED COMBINATIONS:

COMBINATION CONSTRAINTS:

*****
CLEARANCES:
*****

WORDS:

    name= ALL;          sname= ALL;          compartments= 0-89;
    name= GIBRALTAR;    sname= GIB;          compartments= 1-5;
    name= COMP_A;       sname= A;            compartments= 1;
    name= COMP_B;       sname= B;            compartments= 2;
    name= COMP_C;       sname= C;            compartments= 3;
    name= COMP_D;       sname= D;            compartments= 4;
    name= INTERNET;     sname= INET;         compartments= 67;
    name= TADMIN;       sname= TADMIN;        compartments= 68;

REQUIRED COMBINATIONS:

COMBINATION CONSTRAINTS:

*****
CHANNELS:
*****

WORDS:

*****
PRINTER BANNERS:
*****

WORDS:

*****
ACCREDITATION RANGE:
*****

    classification= IMPL_LO;          only valid compartment combinations:

    impl_lo

    classification= U;                only valid compartment combinations:

    u

    classification= PUB;              all compartment combinations valid;
    classification= SEN;              all compartment combinations valid;
    classification= RES;              all compartment combinations valid;
```

```

classification= CON;    all compartment combinations valid;
classification= SEC;    all compartment combinations valid;
classification= TS;     all compartment combinations valid;

minimum clearance= impl_lo;
minimum sensitivity label= impl_lo;
minimum protect as classification= impl_lo;

```

5.2 azdb (Authorisation DB)

```

$ /tbin/secls -s azdb
azdb
    SENSITIVITY LABEL
    TOP SECRET ALL
$ ls -al azdb
----- 1 sys      sys      4808 Jun 19 09:26 azdb
$

```

```

cat: cannot open /etc/security/azdb
$ id
uid=128(isso) gid=1(other)
$ /tbin/setpv +a pv_root $$
$ cat /etc/security/azdb
#
# PitBull 3.0 Authorization Database
#
# The format of database file is as follows:
#  AZNAME:index:[flags]:[Admin AZ]:[AZ Role]:[Dominating AZ]:username,...
#
# Authorization names are CaSe sensitive
#
# High Level Authorizations
#
AUTH:1::::isso          # Authorization Administration
ISSO:2::SA:::isso,richard
SA:3::::sa
SO:4::::so
#
# Medium Level Authorizations
#
AUDITSYS:20::::ISSO:    # Audit Analysis
BOOT:21::::ISSO:       # Trusted Startup
DEBUG:22::::ISSO:      # Application Debugging
FSADMIN:23::::SA:       # File System Admin
FSVIEW:24::::SECFSVIEW: # View All FS Attributes
SECFSVIEW:25::::SECFSADMIN: # View All FS Security Attributes
SECFSADMIN:26::::ISSO:  # File System Security Admin
PROCVIEW:27::::SECROCVIEW: # View All Process Attributes
SECROCVIEW:28::::SECROCVIEW: # View All Process Security Attributes
SECROCVIEW:29::::ISSO:  # Process System Security Admin
NETCONFIG:30::::ISSO:   # Network Configuration and Settings
SECNETCONFIG:31::::ISSO: # Network Security Configuration and Settings
OPERATIONS:32::::SO:    # System Operator
SECSSADMIN:33::::ISSO:  # General Security System Administrator
SECUSERADMIN:34::::ISSO: # User Security Administrator
#
#
# Command Level Authorizations
#
# Audit

```



```
AUCTIONMOD:100::::AUDITSYS: # AAS
AUDIT:101::::AUDITSYS: # AAS
AUDITCONFIG:102::::AUDITSYS: # AAS
AUDITD:103::::AUDITSYS: # AAS
AUDITREDUCE:104::::AUDITSYS:ccuser
AUDITSTAT:105::::AUDITSYS: # AAS
PRAUDIT:106::::AUDITSYS: # PAS
#
# Authorization
AZLIST:150::::AUTH:
CHAZDB:151::::AUTH: # AAS
GETKAT:152::::AUTH: # PAS
SETKAT:153::::AUTH: # AAS
#
# Cron Subsystem
CRON:200::::SA:
#
# Device Security
DEVLVLMGMT:225::::SECSYSADMIN: # AAS
DEVSECCONFIG:226::::SECSYSADMIN: # AAS
TTYMGMT:227::::SECSYSADMIN: # AAS
#
# Network Configuration
IFCONFIG:250::::NETCONFIG: # PAS
INETD:251::::NETCONFIG: # PAS
NDD:252::::NETCONFIG: # PAS
ROUTE:253::::NETCONFIG: # PAS
#
# Network Security Configuration
ASNINIT:275::::SECNETCONFIG: # PAS
NETRULE:276::::SECNETCONFIG: # PAS
#
# File System Attributes
LS:300::::FSVIEW: # PAS
SECLS:301::::SECFVIEW: # PAS
SEELS:302::::SECFVIEW: # AAS
#
# File System Administration
CHGRP:325::::FSADMIN: # PAS
CHMOD:326::::FSADMIN: # PAS
CHOWN:327::::FSADMIN: # PAS
FSCK:328::::FSADMIN,OPERATIONS: # PAS
MKFS:329::::FSADMIN: # PAS
MOUNT:330::::FSADMIN,OPERATIONS: # PAS
RM:331::::FSADMIN: # PAS
RMDIR:332::::FSADMIN: # PAS
SETFACL:333::::FSADMIN: # PAS
SETTIME:334::::FSADMIN: # PAS
UMOUNT:335::::FSADMIN,OPERATIONS: # PAS
UNLINK:336::::FSADMIN: # PAS
FSTYP:337::::FSADMIN: # PAS
#
# File System Security Administration
CHAUTH:350::::SECFADMIN: # AAS
CHFSF:351::::SECFADMIN: # AAS
CHPV:352::::SECFADMIN: # AAS
CHSL:353::::SECFADMIN: # PAS
MKDIR:354::::SECFADMIN: # PAS
MKNOD:355::::SECFADMIN: # PAS
CHCAP:356::::SECFADMIN: # AAS
CHIL:357::::SECFADMIN: # AAS
CHTL:358::::SECFADMIN: # AAS
FUSER:359::::SECFADMIN: # PAS
CLRI:360::::SECFADMIN: # PAS
#
# Process Attributes
GETPV:400::::SECROCVIEW: # PAS
```



```
GETSL:401::::SECROCVIEW: # PAS
PS:402::::PROCVIEW: # PAS
SECPS:403::::SECROCVIEW: # PAS
#
# Process Security Settings
SETAUTH:425::::SECROCVIEW: # AAS
SETPV:426::::SECROCVIEW: # AAS
SETSL:427::::SECROCVIEW: # PAS
SETCAP:428::::SECROCVIEW: # AAS
SETIL:429::::SECROCVIEW: # AAS
SETTL:430::::SECROCVIEW: # PAS
#
# System Security Configuration
SETSECONF:450::::SECSYSADMIN: # AAS
SWAP:451::::SECSYSADMIN: # AAS
TLIBADMIN:452::::SECSYSADMIN: # AAS
UNAME:453::::SECSYSADMIN: # PAS
#
# System Operations Changes
DATE:500::::SECSYSADMIN: # PAS
HALT:501::::OPERATIONS: # AAS
INIT:502::::OPERATIONS: # PAS
REBOOT:503::::OPERATIONS: # AAS
SHUTDOWN:504::::OPERATIONS: # AAS
UADMIN:505::::OPERATIONS: # AAS
UFSDUMP:506::::OPERATIONS: # AAS
UFSRESTORE:507::::OPERATIONS: # AAS
#
# System Analysis
FINDPV:525::::DEBUG: # PAS
TRUSS:526::::DEBUG: # PAS
#
# System Tools
CHKLEF:550::::ISSO: # AAS
MAKEIDB:551::::ISSO: # AAS
#
# User I&A Commands
SU:575::::SECUSERADMIN: # PAS
LOGINBLOCK:576::::SECUSERADMIN: # AAS
GETCLEAR:577::::SECUSERADMIN: # PAS
SETCLEAR:578::::SECUSERADMIN: # PAS
LOGIN:579::::SECUSERADMIN:html0,admin0,richard,html1,admin1
SULOGIN:580::::SECUSERADMIN:
PASSWD:581::::SECUSERADMIN:
#
# SL Auths
DOWNGRADE:600::::CHSL:
OUTSIDEACCRED:601::::CHSL:
UPGRADE:602::::CHSL:
GETSYSLAB:603::::SECSYSADMIN:
SETSYSLAB:604::::SECSYSADMIN:
#
# IL Auths
ILMODIFY:650::::CHIL:
WEBADMIN:651::::ISSO:root
EXECENV:652::::ISSO:root
#
# Authorizations added by patch 106541-11
#
LINK:653::::FSADMIN: # PAS
SECIPCS:654::::SECROCVIEW: # PAS
PDMKDIR:655::::SECFSADMIN:
PDRMDIR:656::::SECFSADMIN:
PDREAL:657::::SECROCVIEW,SECFSADMIN:
PDLINK:658::::SECFSADMIN:
SAVECORE:659:::::
GETFACL:660:::::
```



```
KEYSERV:661::::  
DOMAINNAME:662::::  
ALLOCATE:663::::  
KILLALL:664::::  
COREADM:665::::  
FIND:666::::  
FSCONV:667::::
```

5.3 Clearance

```
$ /tbin/secls -s clear  
clear  
    SENSITIVITY LABEL  
    TOP SECRET ALL  
$ ls -al clear  
----- 1 sys      sys          210 Jun 15 14:17 clear
```

```
$ cat clear  
#  
# User Clearance Database  
#  
# Format: username:min clearance:max clearance:default label  
#  
DEFAULT_CLEAR:IMPL_LO:TS ALL:IMPL_LO  
isso:IMPL_LO:TS ALL:IMPL_LO  
sa:IMPL_LO:TS ALL:IMPL_LO  
so:IMPL_LO:TS ALL:IMPL_LO
```



6 Appendix LAB

6.1 Sensitivity Labels

```
maquette: ~/5.1 > pwd
/export/home/ibuetler/5.1
maquette: ~/5.1 > gets1 $$
25499:
    EFFECTIVE SL:      IMPLEMENTATION LOW INTERNET
    MINIMUM CLEARANCE: IMPLEMENTATION LOW
    MAXIMUM CLEARANCE: TOP SECRET ALL
maquette: ~/5.1 > touch file
maquette: ~/5.1 > secls -s file
file
    SENSITIVITY LABEL
    IMPLEMENTATION LOW INTERNET
maquette: ~/5.1 > chsl -a "IMPL_LO" file
CAUTION: NEW SL of "file" is not within parent directory's SL range
file: file
    SL : IMPLEMENTATION LOW
maquette: ~/5.1 > secls -s file
file
    SENSITIVITY LABEL
    IMPLEMENTATION LOW
maquette: ~/5.1 > chsl -a UNCLASSIFIED file
CAUTION: NEW SL of "file" is not within parent directory's SL range
file: file
    SL : UNCLASSIFIED
maquette: ~/5.1 > secls -s file
file
    SENSITIVITY LABEL
    UNCLASSIFIED
maquette: ~/5.1 > chsl -a PUBLIC file
CAUTION: NEW SL of "file" is not within parent directory's SL range
file: file
    SL : PUBLIC
maquette: ~/5.1 > secls -s file
file
    SENSITIVITY LABEL
    PUBLIC
maquette: ~/5.1 > rm file
file: No such file or directory
maquette: ~/5.1 > ls -al
total 4
drwx-----  2 ibuetler other      512 Jun 19 10:41 .
drwx-----  5 ibuetler other      512 Jun 19 10:40 ..
-rw-----  1 ibuetler other         0 Jun 19 10:42 file
```



If you want to delete the file, you have 2 options:

- change the SL of the file to the SL of the shell process (chsl)
- change the SL of the process to the SL of the file (setsl)

DELETE FILE BY CHANGE THE SL OF THE FILE TO THE ESL OF THE PROCESS

```
maquette: ~/5.1 > secls -s file
file
  SENSITIVITY LABEL
  PUBLIC
maquette: ~/5.1 > chsl -a "SECRET" file
CAUTION: NEW SL of "file" is not within parent directory's SL range
file: file
  SL : SECRET
maquette: ~/5.1 > rm file
file: No such file or directory
maquette: ~/5.1 > getsl $$
3427:
  EFFECTIVE SL:      IMPLEMENTATION LOW INTERNET
  MINIMUM CLEARANCE: IMPLEMENTATION LOW
  MAXIMUM CLEARANCE: TOP SECRET ALL
maquette: ~/5.1 > chsl -a "IMPLEMENTATION LOW INTERNET" file
file: file
  SL : IMPLEMENTATION LOW INTERNET
maquette: ~/5.1 > rm file
maquette: ~/5.1 >
```

DELETE FILE BY CHANGE THE ESL OF THE PROCESS TO THE SL OF THE FILE

```
maquette: ~/5.1 > getsl $$
3427:
  EFFECTIVE SL:      SECRET
  MINIMUM CLEARANCE: IMPLEMENTATION LOW
  MAXIMUM CLEARANCE: TOP SECRET ALL
maquette: ~/5.1 > secls -s file
file
  SENSITIVITY LABEL
  SECRET
maquette: ~/5.1 >
maquette: ~/5.1 > del file
bash: del: command not found
maquette: ~/5.1 > rm file
file: No such file or directory

I am not able to delete the file, because the 5.1 dir does have the following SL

maquette: ~ > secls -s 5.1/
5.1/
  MAXIMUM SENSITIVITY LABEL
  IMPLEMENTATION LOW INTERNET
  MINIMUM SENSITIVITY LABEL
  IMPLEMENTATION LOW INTERNET
maquette: ~ >

If I move the directories SL to SECRET as well, I will be able to delete the file

222 chsl -h SECRET -l SECRET 5.1/
223 cd 5.1
224 setsl -e SECRET $$
225 cd 5.1
229 secls -s file
230 touch ivan
231 secls -s *
232 rm ivan
```

```
233 ls -al
234 history
235 rm file
236 history
```

After the file is between MIN and MAX SL of the directory, I can delete the file

6.2 SL Inheritance

File and directory inherit the SL of the process that created it

```
maquette: ~ > getsl $$
1668:
    EFFECTIVE SL:      IMPLEMENTATION LOW INTERNET
    MINIMUM CLEARANCE: IMPLEMENTATION LOW
    MAXIMUM CLEARANCE: TOP SECRET ALL
maquette: ~ > touch myfile
maquette: ~ > secls -s myfile
myfile
    SENSITIVITY LABEL
    IMPLEMENTATION LOW INTERNET
maquette: ~ > mkdir mydir
maquette: ~ > rm myfile
maquette: ~ > secls -s mydir
mydir
    MAXIMUM SENSITIVITY LABEL
    IMPLEMENTATION LOW INTERNET
    MINIMUM SENSITIVITY LABEL
    IMPLEMENTATION LOW INTERNET
maquette: ~ >
maquette: ~ >

myfile and mydir inherit the "IMPL_LO INTERNET" SL form the shell
```

6.3 MLS / Secure Networking

MLS is equal host based authentication and inherits the privileges to the daemon.

```

maquette ibuetler > netruler h+io 192.168.3.66 =tcp :ftp -ts:l:p +impl_lo +ts +con
192.168.3.66 IN OUT tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS | CON |

maquette ibuetler > netruler hl
192.168.3.66 OUT tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS | CON |
192.168.3.66 IN tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS | CON |

-----

maquette ibuetler > netruler h+io 192.168.3.66 =tcp :ftp -ts:l:p +impl_lo +ts a +con a
192.168.3.66 IN OUT tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS A | CON A |

maquette ibuetler > netruler hl
192.168.3.66 OUT tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS A | CON A |
192.168.3.66 IN tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS A | CON A |

-----

maquette ibuetler > netruler h+io 192.168.3.66 =tcp :telnet -ts:l:p +impl_lo +res a +res a
192.168.3.66 IN OUT tcp:23 s:l:p:i:i:i:i:
  | IMPL_LO | RES A | RES A |

maquette ibuetler > netruler hl
192.168.3.66 OUT tcp:23 s:l:p:i:i:i:i:
  | IMPL_LO | RES A | RES A |
192.168.3.66 OUT tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS A | CON A |
192.168.3.66 IN tcp:23 s:l:p:i:i:i:i:
  | IMPL_LO | RES A | RES A |
192.168.3.66 IN tcp:21 s:l:p:i:i:i:i:
  | IMPL_LO | TS A | CON A |

```

next Step:

- 1) load c:\autoexec.bat to ftp directory by ftp [CON A]
- 2) telnet to server and try to view the file

It is not possible to view the file. The command getsl returns

```

maquette ~ > getsl $$
20515:
EFFECTIVE SL:   RESTRICTED COMP_A
MINIMUM CLEARANCE: IMPLEMENTATION LOW
MAXIMUM CLEARANCE: TOP SECRET ALL

```

But this is not enough to gain access to the [CON A] uploaded file

6.4 Authorizations

```

$ id
uid=128(isso) gid=1(other)
$ /tbin/azlist -l -aruf

FSCONV                flags:
FIND                   flags:
COREADM                flags:
KILLALL                flags:
ALLOCATE               flags:
DOMAINNAME             flags:
KEYSERV                flags:
GETFACL                flags:
SAVECORE               flags:
PDLINK                 flags:
PDREAL                 flags:
PDRMDIR                flags:
PDMKDIR                flags:
SECIPCS                flags:
LINK                   flags:
EXECENV                flags:
    root

WEBADMIN                flags:
    root

ILMODIFY                flags:
SETSUSLAB              flags:
GETSYSLAB              flags:
UPGRADE                flags:
OUTSIDEACCRED          flags:
DOWNGRADE              flags:
PASSWD                 flags:
SULOGIN                flags:

LOGIN                   flags:
    html0
    admin0
    richard
    html1
    admin1
    sepp
    ibuetler
    html2
    html3
    admin2
    peter
    admin3
    Christian
    html5
    admin5
    html6
    admin7
    admin6
    mike
    html8
    admin8
    cug
    html9
    admin9
    jenslehm

```



html10	
admin10	
andy	
html12	
admin12	
lehmann	
html13	
admin13	
sidney	
html14	
admin14	
sid	
html15	
admin15	
richard2	
html16	
admin16	
SETCLEAR	flags:
GETCLEAR	flags:
LOGINBLOCK	flags:
SU	flags:
MAKEIDB	flags:
CHKLEF	flags:
TRUSS	flags:
FINDPV	flags:
UFSRESTORE	flags:
UFSDUMP	flags:
UADMIN	flags:
SHUTDOWN	flags:
REBOOT	flags:
INIT	flags:
HALT	flags:
DATE	flags:
UNAME	flags:
TLIBADMIN	flags:
SWAP	flags:
SETSECCONF	flags:
SETTL	flags:
SETIL	flags:
SETCAP	flags:
SETSL	flags:
SETPV	flags:
SETAUTH	flags:
SECPS	flags:
PS	flags:
GETSL	flags:
GETPV	flags:
CLRI	flags:
FUSER	flags:
CHTL	flags:
CHIL	flags:
CHCAP	flags:
MKNOD	flags:
MKDIR	flags:
CHSL	flags:
CHPV	flags:
CHFSF	flags:
CHAUTH	flags:
FSTYP	flags:
UNLINK	flags:
UMOUNT	flags:
SETTIME	flags:
SETFACL	flags:
RMDIR	flags:
RM	flags:



MOUNT	flags:
MKFS	flags:
FSCK	flags:
CHOWN	flags:
CHMOD	flags:
CHGRP	flags:
SEELS	flags:
SECLS	flags:
LS	flags:
NETRULE	flags:
ASNINIT	flags:
ROUTE	flags:
NDD	flags:
INETD	flags:
IFCONFIG	flags:
TTYMGMT	flags:
DEVSECCONFIG	flags:
DEVLVLMGMT	flags:
CRON	flags:
SETKAT	flags:
GETKAT	flags:
CHAZDB	flags:
AZLIST	flags:
PRAUDIT	flags:
AUDITSTAT	flags:
AUDITREDUCE ccuser	flags:
AUDITD	flags:
AUDITCONFIG	flags:
AUDIT	flags:
AUCTLMOD	flags:
SECUSERADMIN	flags:
SECSYSADMIN	flags:
OPERATIONS	flags:
SECNETCONFIG	flags:
NETCONFIG	flags:
SECPROCADMIN	flags:
SECPROCVIEW	flags:
PROCVIEW	flags:
SECFSADMIN	flags:
SECFSVIEW	flags:
FSVIEW	flags:
FSADMIN	flags:
DEBUG	flags:
BOOT	flags:
AUDITSYS	flags:
SO so	flags:
SA sa	flags:
ISSO isso richard sepp ibuetler peter Christian mike cug jenslehm andy	flags:



```
lehmann  
sidney  
sid  
richard2
```

```
AUTH  
isso
```

```
flags:
```



7 Appendix Apache Example

How to configure an application being secure

- set restrictions up (compartments, SL)
- give apache PV_ROOT privilege
- find necessary privileges by findpv
- restrict privileges of application
- test application
- set FSF flags
- set up authorizations

7.1 Compartment Design for Apache

bin: secret, apache
conf: confidential, apache
htdocs: public, apache
logs: secret, apache (write access from bin)

FSF_EPS privilege is required on binary. This enables the copy of the MPS to the EPS

7.2 Least Privilege

- Put PV_ROOT on shell
- run binary
- Consult used privilege set
- stop apache
- put those privilege set of binary
- set FSF_EPS flag (make sure) (chfsf -e FSF_EPS myApp)

7.3 Consult Privileges

- findpv
- tracepv (follows forks)

tracepv -a myApp -q

-a == apply used privileges to Innate Privilege Set of myApp binary
-q == whatever arguments the binary needs

7.4 Authorization

The lab includes giving applying a required access authorization to the bin file in order to create a role concept.

/tbin/chauth +p SO <file>	change auth
/tbin/azlist	view users auth
/tbin/setkat	load azlist into kernel

7.5 Execenv

Start the process by another user or from far away of the appropriate SL will be possible by the execenv

7.6 Secure Gate

Data stream link between application components stored on different compartments. The apache example shows up the cgi-bin part within another compartment and allow apache to start printenv there.

7.7 Step by Step Procedure

7.7.1 Create Restrictions

```
cd ~ibuetler/demo  
find . -type d -print -exec chsl -l "IMPL_LO COMP_C" -h "TS COMP_C" {} \;
```

this gives all diectories within the demo directory MIN and MAX SL

```
cd ~ibuetler/demo  
find . -type f -print -exec chsl -a "IMPL_LO COMP_C" {} \;
```

this gives all files within the demo directory the SL

```
chsl -a "SEC COMP_C" bin  
chsl -a "SEC COMP_C" log  
chsl -a "SEC COMP_C" libexec  
chsl -a "CON COMP_C" conf  
chsl -a "PUB COMP_C" htdocs
```

```
chsl -a "SEC COMP_C" bin/*  
chsl -a "SEC COMP_C" log/*
```

```
chsl -a "SEC COMP_C" libexec/*
chsl -a "CON COMP_C" conf/*
chsl -a "PUB COMP_C" httdocs/*
```

```
give shell pv_root
setpv +a pv_root $$
```

```
adapt PORT in httpd.conf; start apache
./bin/apachectl start
```

7.7.2 Apply Privileges (bypass the restrictions)

```
find necessary privileges
findpv <pid of apache daemon>
```

```
set privileges to binary
267 chpv +i PV_ASN_IOCTL,PV_ASN_PORT httpd
268 chpv +i PV_ASN_IOCTL,PV_ASN_PORT apachectl
```

```
set file security flag (fsf)
335 chfsf -e FSF_EPS httpd
```

The problem of not allowing to connect to the machine belongs to the situation, that the PitBull rejected my connection tries.

An additional netrule command enabled my IP address.

7.7.3 Apply Authorization (bypass the restrictions)

```
Authorization Lab

Create new authorization entry into /etc/security/azdb [CSNC]
Promote new entry to kernel by /etc/security/setkat [ok]
Add Access authorization to httpd [chauth]
Try to start apache (ibuetler has not CSNC auth) [error, http does not start]
Add CSNC auth to ibuetler in /etc/security/azdb [vi]
Load new azdb into kernel [/tbin/setkat]
Check if new auth is applied to user [azlist]
Try to start apachectl again [works]
Exclude CSNC in /etc/security/las file [vi file]
Login with a new shell to the machine [login again]
Try to start apache again [error]
```

CREATE NEW AUTHORIZATION AND APPLY IT TO ACCESS AUTH OF BINARY

```
maquette bin > grep CSNC /etc/security/azdb
CSNC:8888:::::
```

```
maquette bin > azlist
LOGIN, ISSO
```

```
maquette bin > /tbin/setkat
```

```
chauth +a CSNC httpd
```



```
maquette bin > secls httpd
60439 -rwxr-xr-x 1 ibuetler other 630520 Jun 20 14:10 httpd
FILE SECURITY FLAGS
FSF_EPS
SENSITIVITY LABEL
SECRET COMP_C
INFORMATION LABEL
IMPLEMENTATION LOW
INTEGRITY LABEL
IMPLEMENTATION LOW
PROXY PRIVILEGES
None
INNATE PRIVILEGES
PV_ASN_IOCTL PV_ASN_PORT
AUTHORIZED PRIVILEGES
None
ACCESS AUTHORIZATIONS
CSNC
PRIVILEGE AUTHORIZATIONS
None
CAPABILITY SET
flags :
read : -----
write : -----
exec : -----
maquette bin >

maquette bin > azlist
LOGIN, ISSO
```

TRY TO START HTTP BINARY (USER IBUETLER DOES NOT HAVE THE REQUIRED AUTH)

```
maquette bin > ./apachectl start
./apachectl: /export/home/ibuetler/demo/apache/default/bin/httpd: cannot execute
./apachectl start: httpd could not be started
```

APPLY THE REQUIRED AUTHORIZATION TO THE USER

```
add ibuetler to /etc/security/azdb
load /tbin/setkat

maquette bin > id
uid=138(ibuetler) gid=1(other)
maquette bin > azlist
CSNC, LOGIN, ISSO
```

TRY TO START THE HTTPD DAEMON

```
maquette bin > ./apachectl start
./apachectl start: httpd started

maquette bin > ps -ef|grep ibuetler
ibuetler 23250 4264 0 14:19:32 pts/2 0:00 -bash
ibuetler 4264 15572 0 13:46:17 pts/2 0:01 -bash
ibuetler 9677 1 1 14:19:28 ? 0:00
/export/home/ibuetler/demo/apache/default/bin/httpd -d /export/home/ibuetler/de
ibuetler 14797 9677 0 14:19:29 ? 0:00
/export/home/ibuetler/demo/apache/default/bin/httpd -d /export/home/ibuetler/de
maquette bin >
```

EXCLUDE THE USER BY THE LAS FILE

```
bash-2.05$ pwd
/etc/security
bash-2.05$ tail las
```



```
#
# username:EXCEPT:
#
# The following would cause login to leave the LAS in the most restrictive state
# for username.
#
# username:ONLY:
#
DEFAULT_LAS:EXCEPT:
ibuetler:ONLY:SA UPGRADE DOWNGRADE
bash-2.05$
```

LOGIN AGAIN WITH A NEW SHELL TO THE MACHINE (LAS WILL BE APPLIED DURING LOGIN)

```
login: ibuetler
Password:
Last login: Wed Jun 20 14:30:15 from pts/13
maquette ibuetler > setsl -a "SEC C" $$
22547:
    EFFECTIVE SL:      SECRET COMP_C
    MINIMUM CLEARANCE: SECRET COMP_C
    MAXIMUM CLEARANCE: SECRET COMP_C
maquette ibuetler > cd demo/apache/default/bin/
maquette bin > ./apachectl start
./apachectl: /export/home/ibuetler/demo/apache/default/bin/httpd: cannot execute
./apachectl start: httpd could not be started
maquette bin > getpv $$

maquette bin > azlist
CSNC, LOGIN, ISSO
maquette bin > secls httpd
    60439 -rwxr-xr-x   1 ibuetler other      630520 Jun 20 14:10 httpd
    SENSITIVITY LABEL
    SECRET COMP_C
    INFORMATION LABEL
    IMPLEMENTATION LOW
    INTEGRITY LABEL
    IMPLEMENTATION LOW
    PROXY PRIVILEGES
    None
    INNATE PRIVILEGES
    PV_ASN_IOCTL          PV_ASN_PORT
    AUTHORIZED PRIVILEGES
    None
    ACCESS AUTHORIZATIONS
    None
    PRIVILEGE AUTHORIZATIONS
    None
    CAPABILITY SET
    flags :
    read  : -----
    write : -----
    exec  : -----
maquette bin >
```

even I would have the right authorization, I am not allowed to use the httpd. I can check the las settings by

7.7.4 Start Apache by ExecEnv

Create the following directive to `/etc/security/execenv.conf`

```
<butweb>
Authorization CSNC
User ibuetler
Group other
SL "SEC C"
Delay 2
Command /export/home/ibuetler/demo/apache/default/bin/apachectl
Options AllowArgs
</butweb>
```

Make sure the required user with EXECENV privileges has CSNC authorization

```
maquette ~ > /opt/gibraltar/bin/execenv butweb
usage: /export/home/ibuetler/demo/apache/default/bin/apachectl
(start|stop|restart|fullstatus|status|graceful|configtest|help)

start      - start httpd -d /export/home/ibuetler/demo/apache/default -R
/export/home/ibuetler/demo/apache/default/libexec"
stop       - stop httpd -d /export/home/ibuetler/demo/apache/default -R
/export/home/ibuetler/demo/apache/default/libexec"
restart    - restart httpd if running by sending a SIGHUP or start if
not running
fullstatus - dump a full status screen; requires lynx and mod_status enabled
status     - dump a short status screen; requires lynx and mod_status enabled
graceful   - do a graceful restart by sending a SIGUSR1 or start if not running
configtest - do a configuration syntax test
help      - this screen

maquette ~ >
maquette ~ >
maquette ~ > /opt/gibraltar/bin/execenv butweb start
/export/home/ibuetler/demo/apache/default/bin/apachectl start: httpd (pid 16512) already
running

maquette ~ > /opt/gibraltar/bin/execenv butweb stop
/export/home/ibuetler/demo/apache/default/bin/apachectl stop: httpd stopped

maquette ~ > /opt/gibraltar/bin/execenv butweb start
/export/home/ibuetler/demo/apache/default/bin/apachectl start: httpd started

maquette ~ > id
uid=143(peter) gid=1(other)

maquette ~ > ps -ef|grep ibuetler
ibuetler  224      1  0 11:32:36 ?          0:00
/export/home/ibuetler/demo/apache/default/bin/httpd -d /export/home/ibuetler/de
ibuetler  9433    224  0 11:32:38 ?          0:00
/export/home/ibuetler/demo/apache/default/bin/httpd -d /export/home/ibuetler/de
maquette ~ >
```

This example adds CSNC authorization to the user peter in `/etc/security/azdb` and enter `/sbin/setkat`. This will allow peter to execute the binary

7.7.5 Secure Gate Example

Apply the CGI's to another compartment and try to use them by an Secure Gate

```
Enable interface for daemon

maquette ibuetler > netrule h+io 192.168.3.66 =tcp :301 -ts:l:p +impl_lo +ts all
+impl_lo
192.168.3.66 IN OUT tcp:301 s:l:p:i:i:i:i:
| IMPL_LO | TS ALL | IMPL_LO |
maquette ibuetler >
```

BEFORE ANY CHANGE

<http://192.168.0.211:301/cgi-bin/test-cgi>

```
ScriptAlias /cgi-bin/ "/export/home/ibuetler/demo/apache/default/cgi-bin/"
```

```
maquette default > secls -s cgi-bin/
cgi-bin/
  MAXIMUM SENSITIVITY LABEL
  SECRET COMP_C
  MINIMUM SENSITIVITY LABEL
  SECRET COMP_C
maquette default > cd cgi-bin/
maquette cgi-bin > secls -s test-cgi
test-cgi
  SENSITIVITY LABEL
  SECRET COMP_C
maquette cgi-bin >
```

In this configuration, the test-cgi works !!

CONFIGURE TEST-CGI TO ANOTHER COMPARTMENT [SECRET COMP_D]

```
maquette default > chsl -a "SEC D" cgi-bin/
CAUTION: NEW SL of "cgi-bin/" is not within parent directory's SL range
file: cgi-bin/
  MIN SL : SECRET COMP_D
  MAX SL : SECRET COMP_D
maquette default > cd cgi-bin/
maquette cgi-bin > chsl -a "SEC D" *
file: printenv
  SL : SECRET COMP_D
file: test-cgi
  SL : SECRET COMP_D
maquette cgi-bin > cd ..
maquette default >
```

The refresh (new request) to the test-cgi fails now.