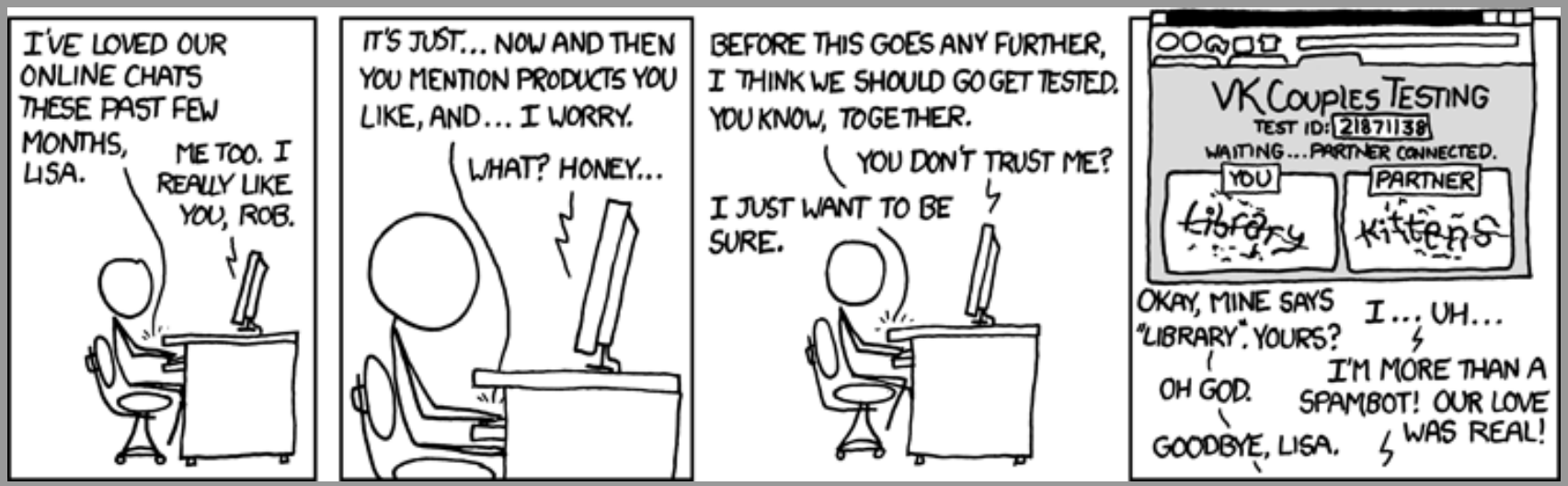
Sistemas de Cofianza

Miguel Angel Astor Romero

11 de octubre de 2019

Agenda


I'VE LOVED OUR ONLINE CHATS THESE PAST FEW MONTHS, LISA.
ME TOO. I REALLY LIKE YOU, ROB.
IT'S JUST... NOW AND THEN YOU MENTION PRODUCTS YOU LIKE, AND... I WORRY.
WHAT? HONEY...
BEFORE THIS GOES ANY FURTHER, I THINK WE SHOULD GO GET TESTED. YOU KNOW, TOGETHER.
YOU DON'T TRUST ME?
I JUST WANT TO BE SURE.
VK COUPLES TESTING
TEST ID: 21871138
WAITING... PARTNER CONNECTED.
YOU
PARTNER
library
kittens
OKAY, MINE SAYS "LIBRARY". YOURS?
I... UH...
OH GOD.
I'M MORE THAN A SPAMBOT! OUR LOVE WAS REAL!
GOODBYE, LISA.

A que Llamamos "Confianza"

Según el Diccionario de la RAE

de confianza

1. *loc. adj.* Dicho de una persona: Con quien se tiene trato íntimo o familiar.
2. *loc. adj.* Dicho de una persona: En quien se puede confiar.
3. *loc. adj.* Dicho de una cosa: Que posee las cualidades recomendables para el fin a que se destina.

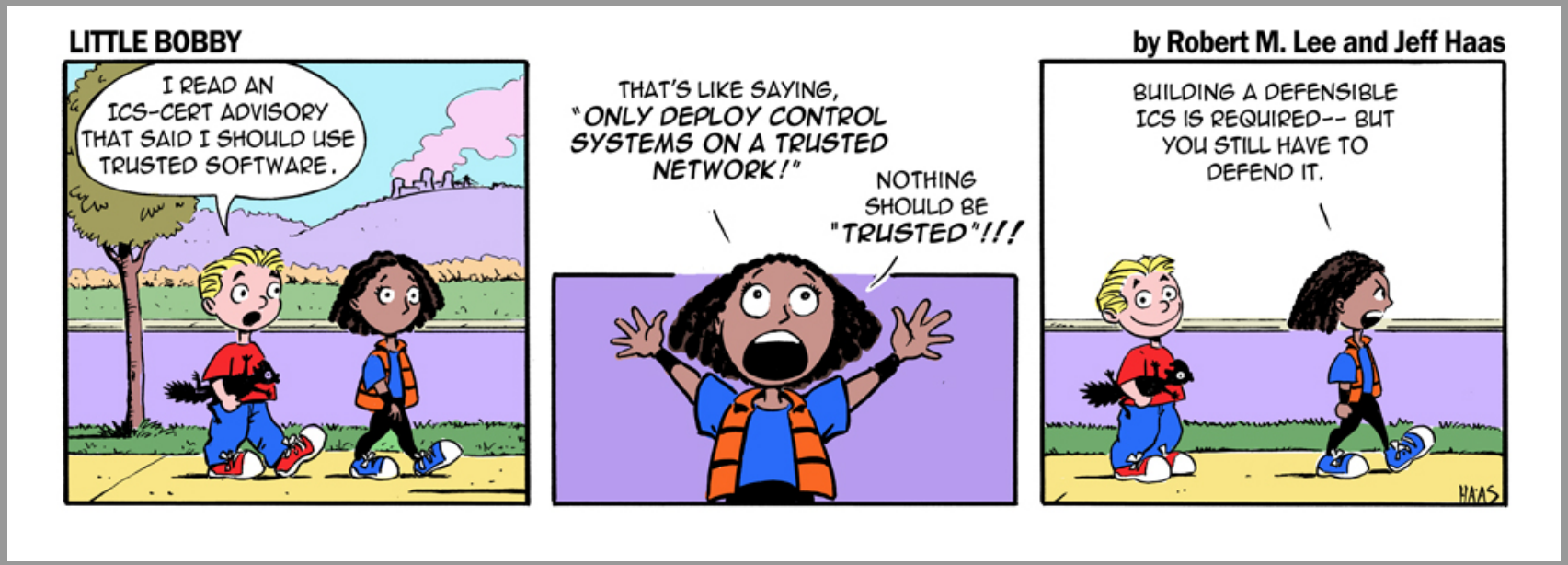
Sistemas de Confianza y Confiabilidad de Sistemas

Confianza no es lo mismo que confiabilidad.

Confiabilidad (del inglés *reliability*)

- Propiedad de un sistema de ser capaz de funcionar de manera continua sin fallas por un intérvalo de tiempo.
- Es la probabilidad de que un sistema este disponible durante un intervalo de tiempo específico.

Por otro lado, un sistema de confianza es un mecanismo de control de acceso y gestión digital de derechos/restricciones (DRM).


LITTLE BOBBY
by Robert M. Lee and Jeff Haas
I READ AN ICS-CERT ADVISORY THAT SAID I SHOULD USE TRUSTED SOFTWARE.
THAT'S LIKE SAYING, "ONLY DEPLOY CONTROL SYSTEMS ON A TRUSTED NETWORK!"
NOTHING SHOULD BE "TRUSTED"!!!
BUILDING A DEFENSIBLE ICS IS REQUIRED-- BUT YOU STILL HAVE TO DEFEND IT.
HAAS

Bases del Control de Acceso

- Dado un sistema que almacena datos sensibles, es necesario entonces controlar quien tiene acceso a estos datos.
- Un mecanismo elemental de control de acceso son los permisos en UNIX.

```
total 2160
-rw-rw-r-- 1 miky miky ... confianza.bbl
-rw-rw-r-- 1 miky miky ... confianza.org
-rw-rw-r-- 1 miky miky ... confianza.pdf
-rw-rw-r-- 1 miky miky ... confianza.tex
```

Modelo General de Control de Acceso

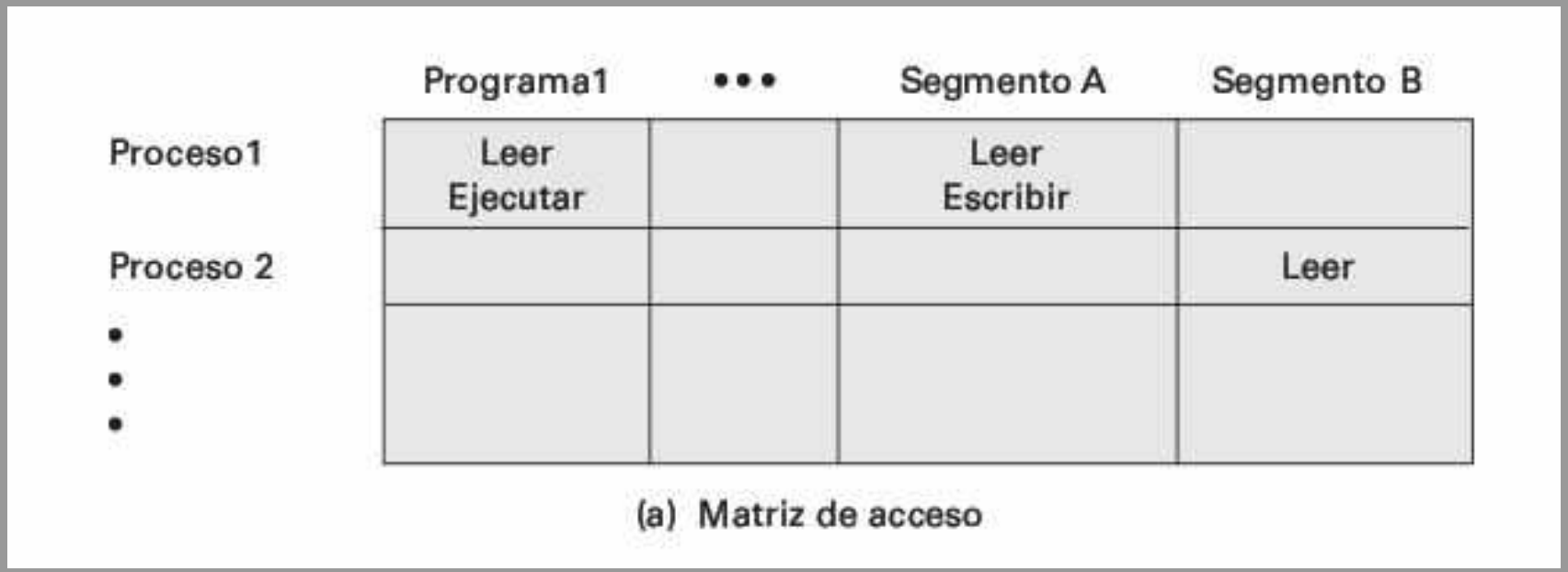
Elementos Básicos

Sujeto entidad capaz de acceder a objetos.

Objeto entidad cuyo acceso y uso deben ser controlados.

Derecho (de acceso) formas en que un sujeto puede acceder a un objeto.

Matrices de Acceso

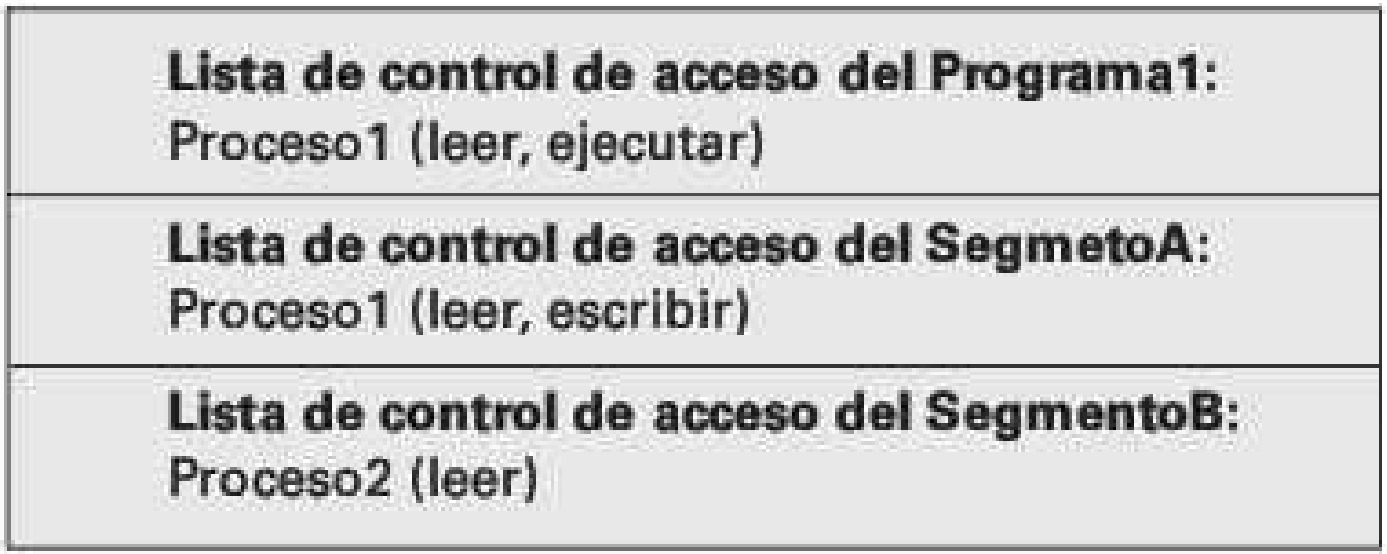


	Programa1	•••	Segmento A	Segmento B
Proceso1	Leer Ejecutar		Leer Escribir	
Proceso 2				Leer
•				
•				
•				

(a) Matriz de acceso

Listas de Control de Acceso

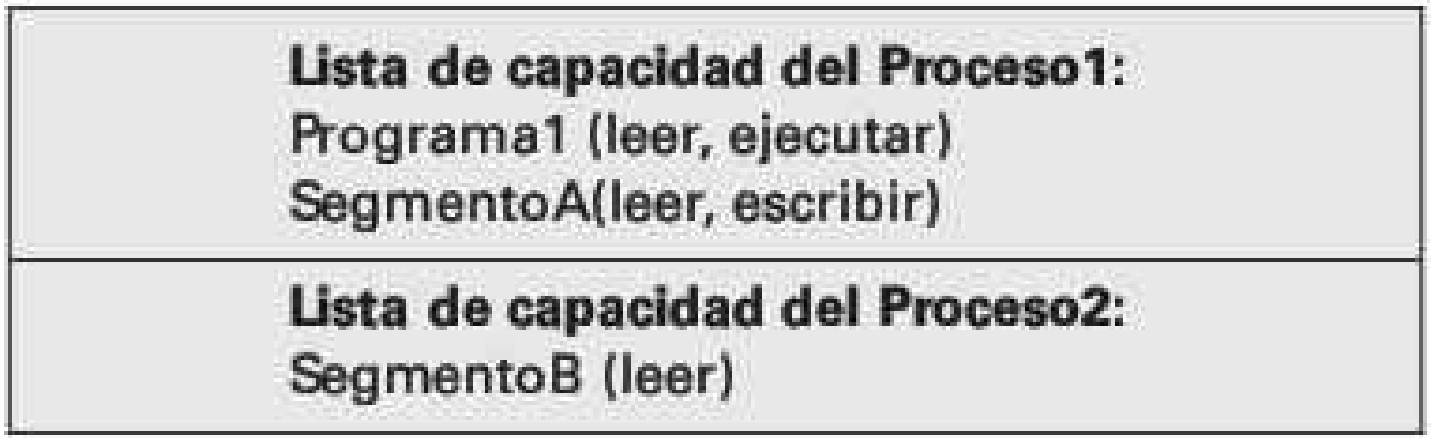
Descomposición por columnas.



(b) Lista de control de acceso

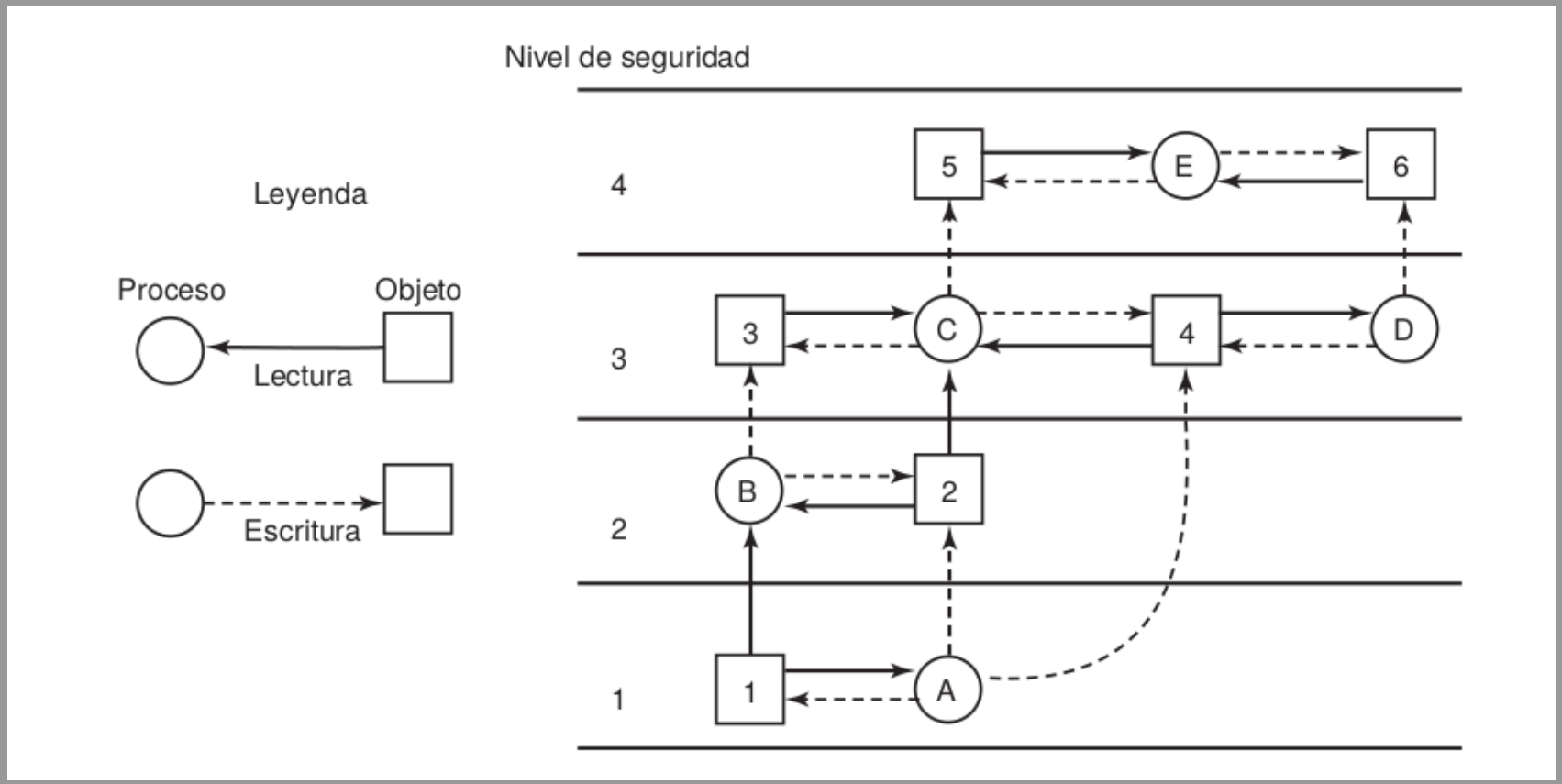
Tickets o Listas de Capacidades

Descomposición por filas.



(c) Lista de capacidad

Concepto


Nivel de seguridad
Leyenda
Proceso
Objeto
Lectura
Escritura
4
3
2
1
5
E
6
3
C
4
D
B
2
1
A

Modelo Bell-La Padula

Provee confidencialidad pero no integridad de datos.

No leer hacia arriba *(propiedad de seguridad simple)* un sujeto solamente puede acceder a objetos de su mismo nivel de seguridad o menor.

No escribir hacia abajo *(propiedad de seguridad *)* un sujeto solo puede escribir en objetos de su mismo nivel de seguridad o superior.

Propiedad * fuerte

Un sujeto solo puede escribir a objetos de su nivel de seguridad.

- Provee integridad de datos pero limita que se puede realizar.

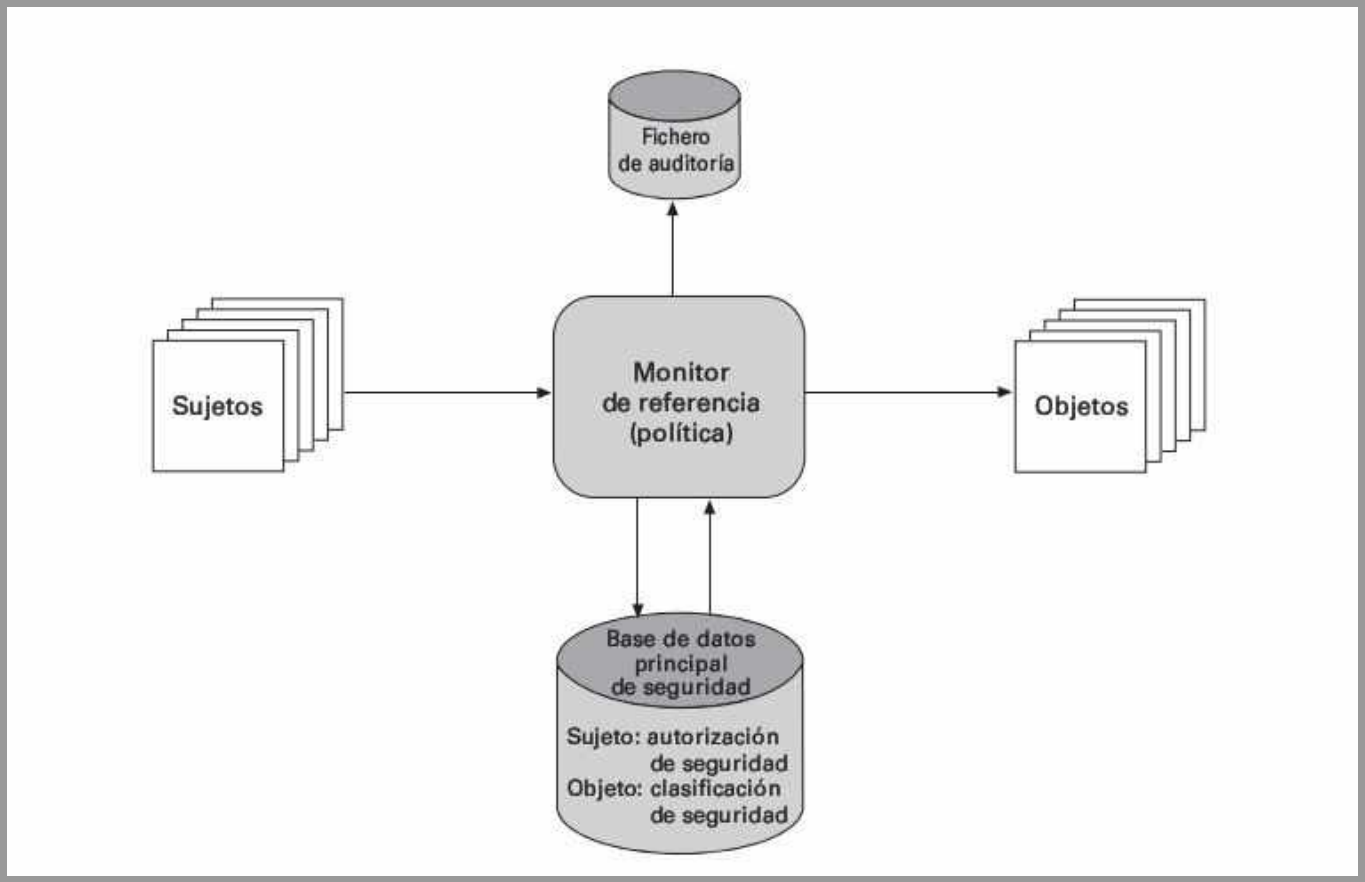
Modelo Biba

Provee integridad de datos pero no confidencialidad.

No escribir hacia arriba *(propiadad de integridad simple)* un sujeto solamente puede escribir en objetos de su mismo nivel de seguridad o menor.

No leer hacia abajo *(propiedad de integridad *)* un sujeto solo puede acceder a objetos de su mismo nivel de seguridad o superior.

Monitor de Referencia



Propiedades de un Monitor de Referencia

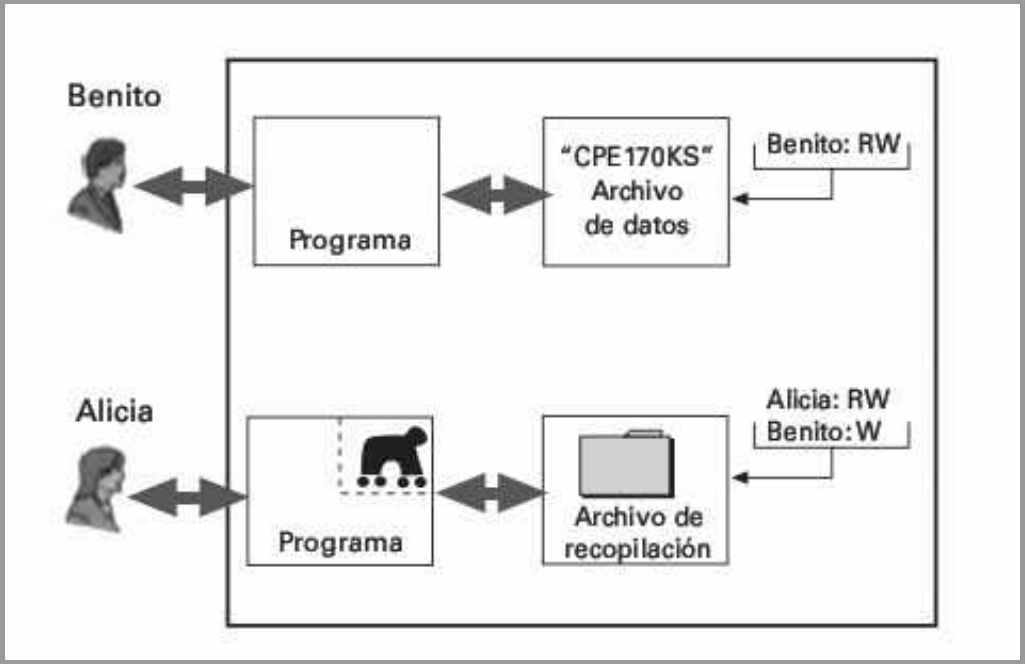
Mediación completa las reglas de seguridad se aplican en todos los accesos.

Aislamiento el monitor de referencia y la base de datos fundamental de seguridad están debidamente protegidos.

Verificabilidad es demostrable matemáticamente que el monitor de referencia funciona correctamente.

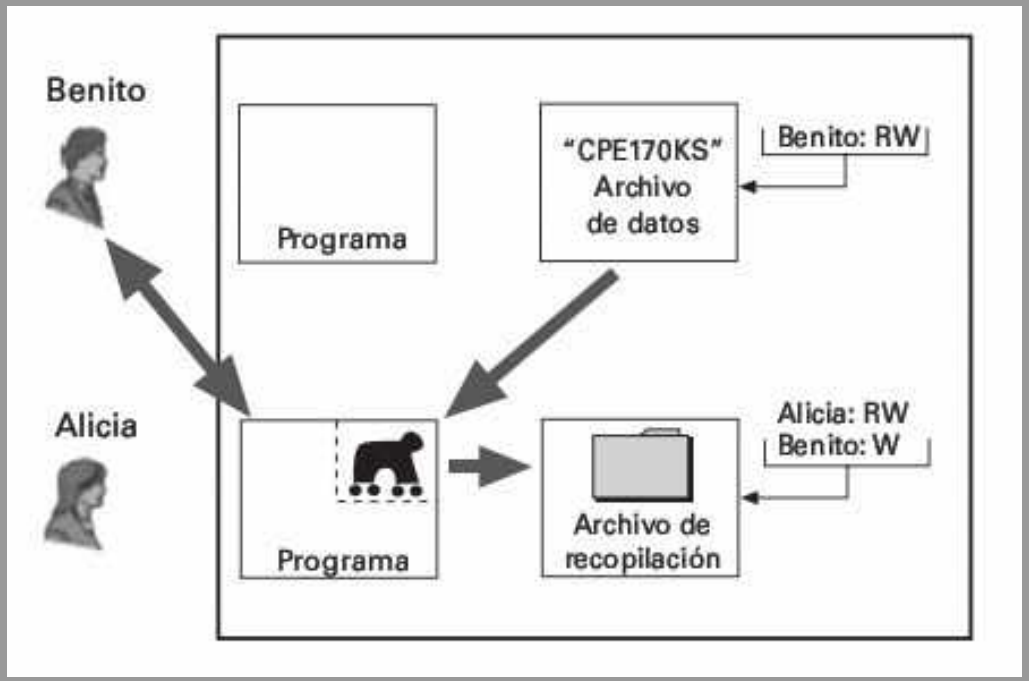
Funcionamiento del Monitor de Referencia

Acceso Legítimo sin Monitor de Referencia



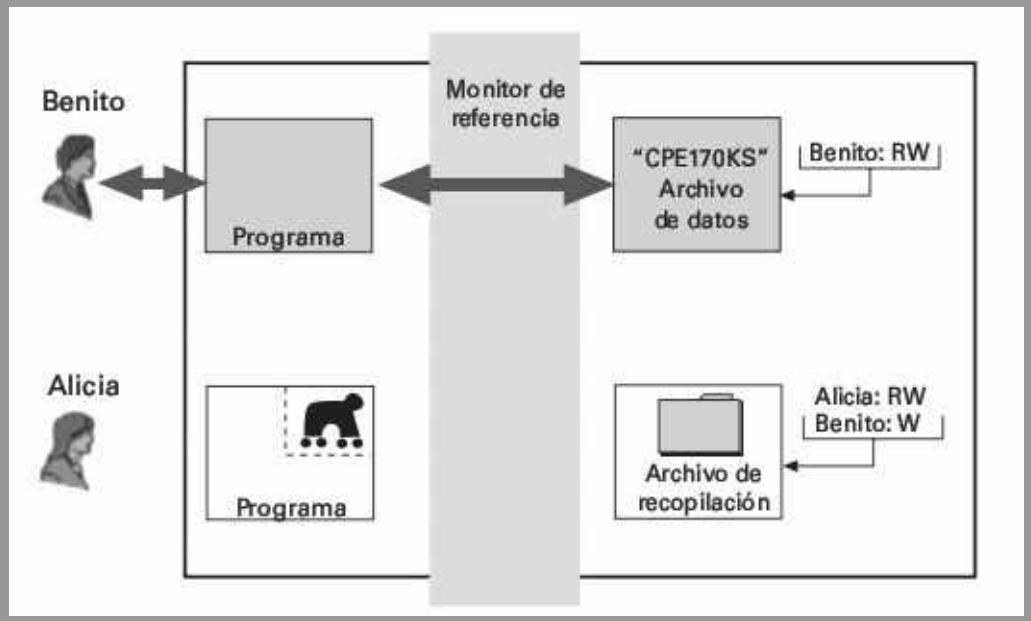
Funcionamiento del Monitor de Referencia

Troyano sin Monitor de Referencia



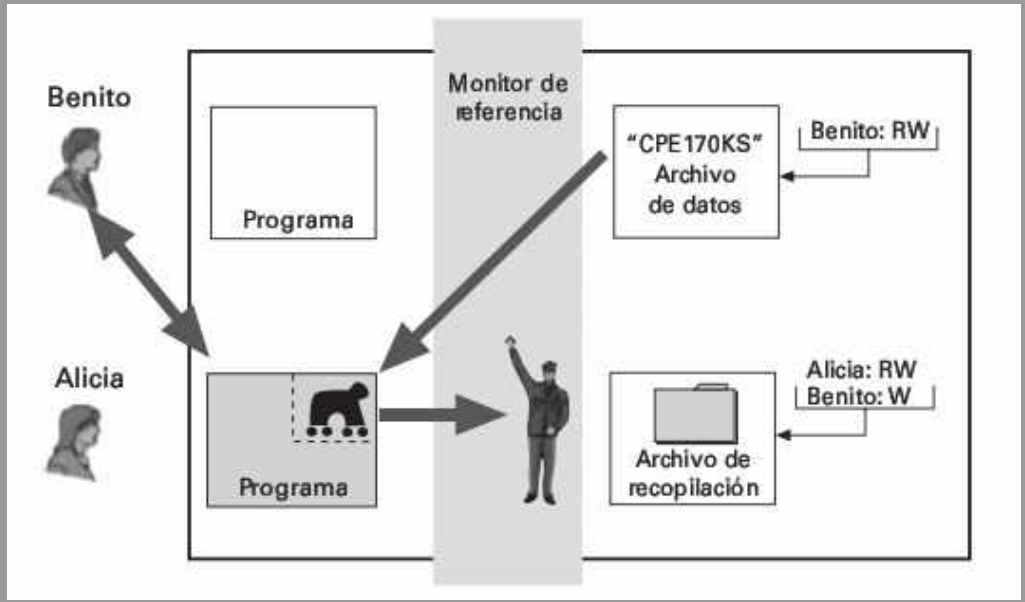
Funcionamiento del Monitor de Referencia

Acceso Legítimo con Monitor de Referencia



Funcionamiento del Monitor de Referencia

Troyano con Monitor de Referencia



Este funcionamiento asume el modelo Bell-La Padula

Módulo de Plataforma Confiable (TPM)

- *Trusted Platform Module* en inglés.
- Coprocesador criptográfico que permite:
  - Almacenar claves de cifrado.
  - Cifrar y descifrar.
  - Verificar firmas.
- Gestionado por el Sistema Operativo.

Funciones de un TPM

- Controlar que sistemas operativos pueden correr en la computadora:
  - Windows 10 y Linux pueden hacer uso de esto con UEFI.
- Permitir la ejecución solamente de programas autorizados.
- Controlar el acceso a archivos.

Los TPM son Espectacularmente Controversiales

¿Quien debería gestionar el TPM?

- El usuario.
- El desarrollador del sistema operativo.
- El fabricante de la máquina.

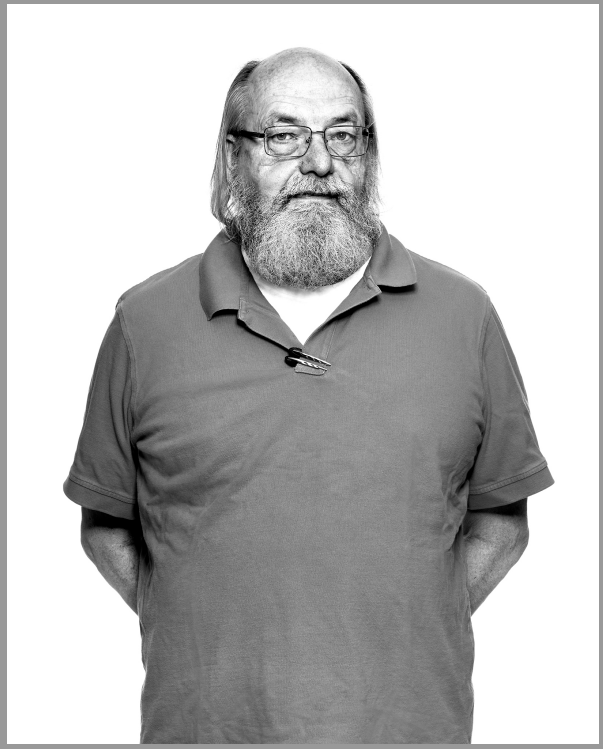
Gestión Digital de Derechos/Restricciones

"To what extent should one trust a statement that a program is free of Trojan horses? Perhaps it is more important to trust the people who wrote the software."

Reflections on Trusting Trust
Ken Thompson

Ken Thompson

- Creador de UNIX, junto a Dennis Ritchie, Brian Kernighan y otros.
- Creador del lenguaje de programación B.
- Inventor de la codificación UTF-8 y el lenguaje Go con Bob Pike.
- Premio Turing de la ACM en 1983.



Quines o Programas Auto-Repoducibles

- Nombrados en honor a Willard Van Orman Quine.
- Ejercicio de programación que consiste en diseñar un programa que imprima su própio código fuente sin examinar el archivo con su código fuente.
- La misma idea se puede usar para construir un compilador malicioso.

```
char s[] = {
        ...
        0,
};
main() {
  int i;
  printf("char s[] = {\n");
  for (i = 0; i < s[i]; i++)
    printf("\t%c,\n", s[i]);
  printf("%s", s);
}
```

Paso por Paso

1. Se agrega código al compilador de C para que inserte una puerta trasera en el programa `login` de UNIX al detectar que se está compilando este programa específico.
2. Se agrega código al compilador para que inserte el código anterior y este mismo cuando se detecta que se está compilando el compilador.
3. Se instala el binario del compilador adulterado como el compilador estándar del sistema.

Supongamos un Compilador de C Cualquiera

```
void compile(char * s) {
   ...
}
```

Paso 1: se Inserta un Troyano en el Comando login

```
void compile(char * s) {
  if (match(s, "pattern") {
    compile_login_troyan();
    return;
  }
  ...
}
```

Paso 2: Compilador Auto-Modificable

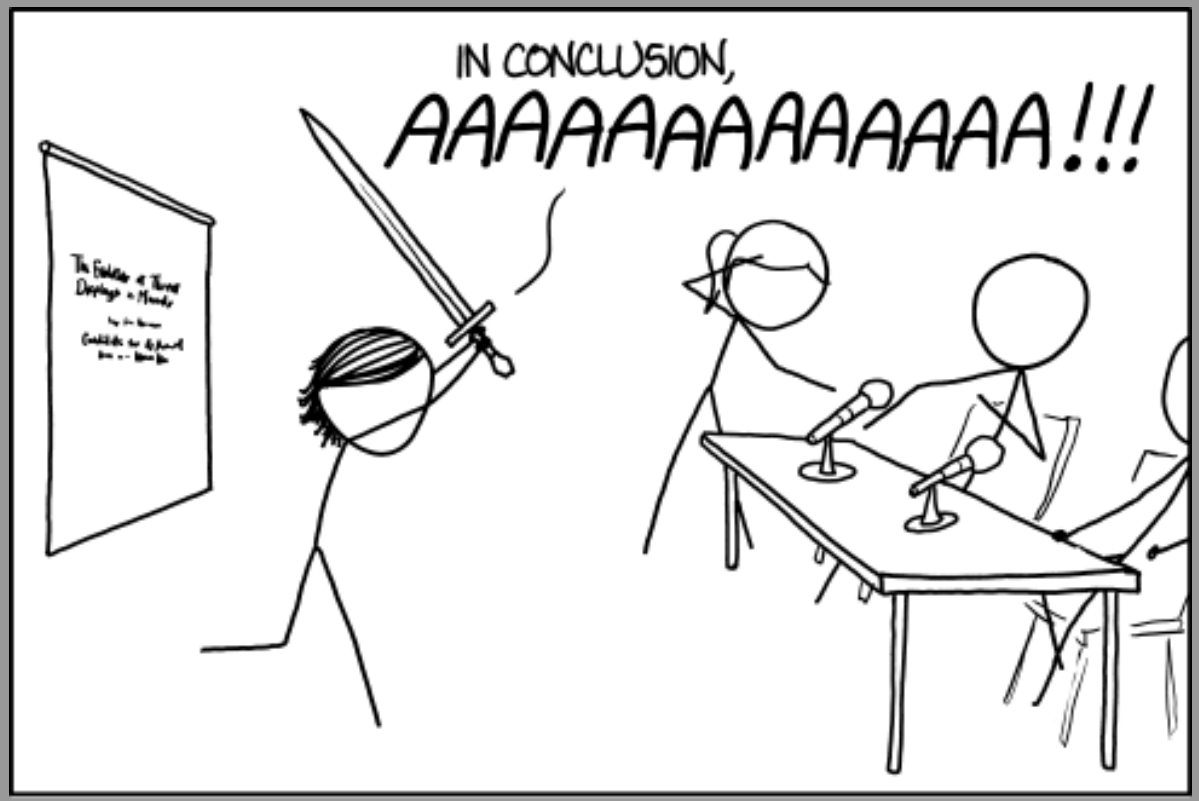
```
void compile(char * s) {
  if (match(s, "pattern 1") {
    compile_login_troyan();
    return;
  }
  if (match(s, "pattern 2") {
    compile_cc_troyan();
    return;
  }
  ...
}
```

Paso 3: Comienza la Paranoia . . .

La Moraleja de la Historia

"You can't trust code that you did not totally create yourself. (Especially code from companies that employ people like me.) No amount of source-level verification or scrutiny will protect you from using untrusted code."

Reflections on Trusting Trust
Ken Thompson


IN CONCLUSION,
AAAAAAAAAAAAA!!!
THE BEST THESIS DEFENSE IS A GOOD THESIS OFFENSE.

Conclusiones

- "Sistema de confianza" tiene dos significados, dependiendo de si estamos hablando de la academia o la industria.
- Academicamente, un sistema de confianza es un mecanismo de control de acceso multinivel para el sistema operativo.
- Industrialmente, un sistema de confianza es un mecanismo de gestión de digital de derechos/restricciones.
- No existe tal cosa como un sistema 100 % seguro o confiable.
- Como dicen en Rusia: confien, pero verifiquen.

¿Preguntas?