Unidad 2 / Escenario 3

Lectura fundamental

Modelos tradicionales de seguridad de la información

Contenido

Palabras clave: Bell-LaPadula, Muralla China, Clark Wilson, Biba.

En la unidad anterior revisamos algunos términos importantes en torno a la seguridad de la información. Esta terminología es la base del conocimiento acerca de la seguridad de la información y de ahí surgen nuevos conceptos que deben ser considerados para continuar con este tema.

El siguiente paso es el conocimiento de los modelos utilizados para implementar sistemas de seguridad de la información. Un modelo es una representación que sirve de referencia o guía en una determinada área del saber. En este caso, un modelo de seguridad de la información es un instrumento para la implementación de sistemas de seguridad de la información. Para el modelo es importante la estructura a seguir, los conceptos que maneja y las posibles aplicaciones. No es lo mismo un modelo para un entorno físico que para un entorno de desarrollo.

En este escenario estudiaremos los modelos tradicionales que se han utilizado, su evolución y aplicación en diferentes ambientes.

1. Modelo Bell – LaPadula, 1972-1975

El modelo de seguridad de Bell-LaPadula produjo herramientas conceptuales para el análisis y diseño de sistemas informáticos seguros. Para comprender el modelo, es importante realizar el análisis en su propio tiempo, los conceptos que identifica, su estructura, reglas y aplicaciones.

1.1. Historia

En la época de 1960, el costo de procesamiento de las llamadas supercomputadoras era bastante elevado y en temas de seguridad era necesario utilizar computadoras separadas para cada nivel de seguridad existente. Sin embargo, surgió el desarrollo de compartido, que implicaba la posibilidad de compartir esos sistemas informáticos en todos los niveles de seguridad, con una importante condición: era crucial que los artefactos de procesamiento de cada nivel de seguridad (archivos, registros, datos) se mantuvieran rigurosamente separados con un alto grado de confianza. Por tal motivo, en 1972, The MITRE Corporation inició su tarea de producir un informe titulado *Secure Computer Systems*. El informe debía describir un "modelo matemático de seguridad en los sistemas informáticos". Esta actividad fue una de varias en un proyecto de seguridad y recayó en Len LaPadula y David Elliott Bell, quienes a partir de un modelo matemático desarrollaron el Modelo de Seguridad Bell-LaPadula (BLP). El proyecto fue más allá de la primera iniciativa, puesto que el objetivo final era formalizar la política de seguridad multinivel del Departamento de Defensa de los Estados Unidos (Bell, 2005).

1.2. Elementos

El modelo Bell-LaPadula utiliza diferentes elementos para su funcionamiento. Estos elementos son:

- Sujetos: elementos que realizan el acceso.
- Objetos: elementos a los cuales quiere acceder el sujeto.
- Niveles de seguridad: características del objeto que determinan su acceso.
- Operaciones de acceso: lectura, escritura y lectura/escritura.
- Matriz: elemento en el que se determina si un sujeto puede acceder al objeto, de acuerdo con el nivel de seguridad.

Para realizar el proceso de control de acceso, la autorización del sujeto se compara con la clasificación del objeto y luego se aplican las reglas específicas para controlar cómo pueden tener lugar las interacciones entre los objetos.

Como se aprecia en la figura 1, los sujetos y objetos pueden residir en diferentes niveles de seguridad y tienen relaciones y reglas que dictan las actividades aceptables entre ellos. Por esta razón, se considera que es un modelo de seguridad de flujo de información que se enfoca en asegurar que los sujetos estén debidamente autenticados antes de acceder al objeto.

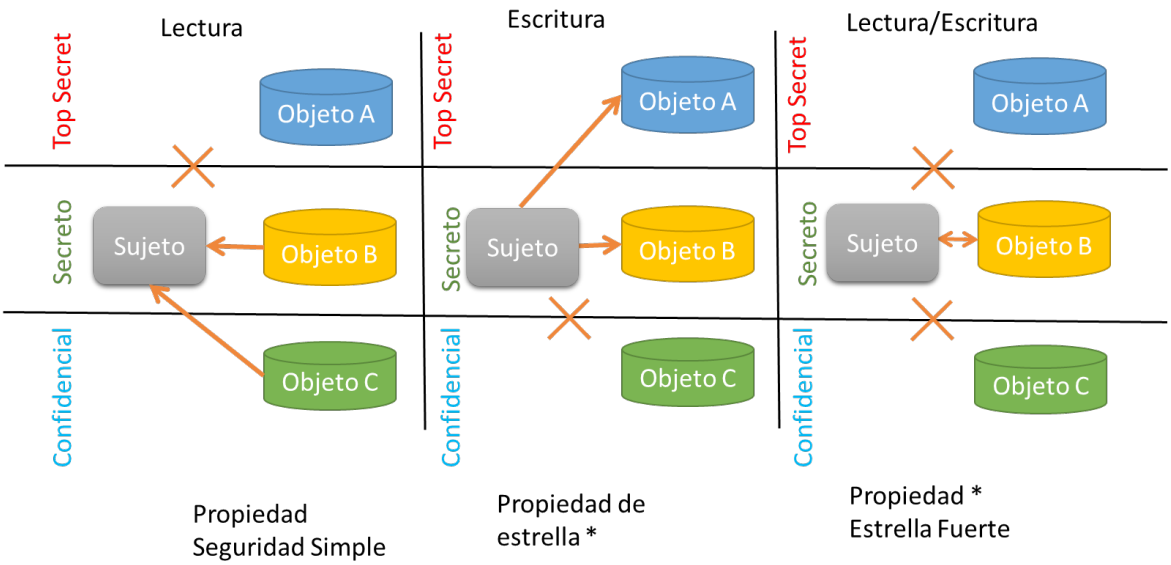

Figura 1. Estructura modelo Bell-LaPadula

Fuente: elaboración propia

1.3. Estructura del modelo

Es un modelo de máquina de estado utilizado para el control de acceso en aplicaciones gubernamentales y militares. La idea general consiste en modelar la transición de estado de la política de seguridad de la información, que describe un conjunto de reglas de control de acceso que utilizan etiquetas de seguridad en objetos y autorizaciones para los sujetos. Las etiquetas de seguridad abarcan desde las más delicadas (por ejemplo, "Máximo secreto") hasta las menos vulnerables (por ejemplo, "Sin clasificar" o "Pública") (Bell, 2005).

El objetivo principal del modelo es evitar que se acceda a información secreta de forma no autorizada. Un sistema que emplea el modelo Bell-LaPadula se denomina sistema de seguridad multinivel porque los usuarios con distintas autorizaciones usan el sistema y este procesa los datos en diferentes niveles de clasificación. El nivel en el que se clasifica la información determina los procedimientos de manejo que se deben usar. Este modelo aplica los aspectos de confidencialidad para el control de acceso (Harris, 2013).

1.4. Consideraciones generales del modelo

En el modelo de Bell-LaPadula se utilizan y aplican tres reglas principales: de seguridad simple, propiedad de estrella y propiedad de estrella fuerte.

La regla de seguridad simple establece que un sujeto en un nivel de seguridad determinado no puede leer datos que residen en un nivel de seguridad superior. Por ejemplo, si al sujeto se le otorga la autorización de seguridad del secreto, esta regla establece que no puede leer los datos clasificados como de *top secret*.

La regla de propiedad * (regla de propiedad de estrella) establece que un sujeto en un nivel de seguridad determinado no puede escribir información en un nivel de seguridad inferior. La regla de seguridad simple se conoce como la regla "sin lectura", y la regla de propiedad * se conoce como la regla "sin anotar".

La tercera regla, la regla de propiedad de estrella fuerte, establece que un sujeto que tiene capacidades de lectura y escritura solo puede realizar esas funciones en el mismo nivel de seguridad; nada más alto y nada más bajo. Entonces, para que un sujeto pueda leer y escribir en un objeto, la separación y la clasificación deben ser iguales. Estas tres reglas indican a qué estados puede entrar el sistema.

El estado de un sistema cambia a medida que se realizan diferentes operaciones. El modelo Bell-LaPadula define un estado seguro, es decir, un entorno informático seguro y las acciones adecuadas, que son operaciones que preservan la seguridad. Esto implica que no importan las entradas a las cuales se someta el sistema, pues, al final del día, este es solo tan seguro como lo fue al comienzo del día.

Esta es la definición del Teorema de Seguridad Básica utilizado en ciencias de la computación, que establece que si un sistema se inicializa en un estado seguro y todas las transiciones de estado permitidas son seguras, entonces cada estado subsiguiente será seguro sin importar qué entradas se produzcan (Harris, 2013).

¿Sabía que...?

Un estado son los valores de las variables en un instante de tiempo dentro de un programa. Si un sujeto ha realizado una operación de lectura en un objeto con un nivel de seguridad inferior, el sujeto ahora tiene una variable con los datos que se leyeron o copiaron. Si un sujeto ha escrito a un objeto con un nivel de seguridad más alto, el sujeto ha modificado ese objeto.

1.5. Aplicaciones

Hemos visto la importancia del control de acceso de niveles inferiores a niveles superiores, sin embargo, ¿qué pasa con el acceso de niveles superiores a inferiores?

Asegurar que la información no fluya desde un nivel de seguridad más alto a un nivel inferior se denomina control de degradación no autorizada de la información, que se lleva a cabo mediante una operación de "anotación". Es por esto que una de las aplicaciones del modelo se da en los sistemas de control de acceso obligatorio (MAC) y en algunos sistemas de control de acceso discrecional (DAC). Todos los sistemas MAC se basan en el modelo Bell-LaPadula, ya que se trata de un mínimo de seguridad multinivel para integrarse en el código. Los sujetos y objetos tienen etiquetas asignadas. La etiqueta del sujeto contiene su etiqueta de autorización (*top secret*, secreto o confidencial) y la etiqueta del objeto contiene su etiqueta de clasificación (*top secret*, secreto o confidencial).

Cuando un sujeto intenta acceder a un objeto, el sistema compara la etiqueta de aprobación del sujeto y la etiqueta de clasificación del objeto y revisa la matriz para validar si se trata de una actividad legal y segura. Para el ejemplo presentado en la figura 1, si la etiqueta de autorización del sujeto es *secret* y la etiqueta de clasificación del objeto es confidencial, el sujeto no puede escribir en este objeto, debido a la regla * *property* (propiedad de estrella), que asegura que los sujetos no pueden accidental o intencionalmente compartir información confidencial escribiendo a un objeto en un nivel de seguridad inferior.

Las bases de datos pueden seguir estas reglas mediante el uso de la poli-instanciación. Esta hace referencia al recurso utilizado para brindar seguridad de las bases de datos, debido a que en un modelo relacional estándar cada registro es determinado por los valores de su clave primaria. Al introducir clases de acceso a una relación puede existir la necesidad de conservar varios registros con el mismo valor de clave primaria, diferenciándose en la clase de acceso asociada; es decir, cuando un usuario de nivel bajo de seguridad inserta datos en un campo ya contiene datos de un nivel mayor o lo contrario (Harris, 2013).

2. Modelo Biba, 1975-1977

Este modelo fue desarrollado poco después del modelo Bell-LaPadula con el fin de abordar la problemática de seguridad desde el principio de integridad.

2.1. Historia

El Modelo Biba o el Modelo de Integridad Biba fue elaborado por Kenneth J. Biba, en 1975, para el proyecto de computadores seguros de la división de sistemas electrónicos de la fuerza aérea. El propósito del proyecto fue el diseño, construcción y validación de un entorno seguro para aplicaciones militarles. Fue creado como complemento del modelo BLP, por lo cual comparte similitudes en estructura y elementos que se utilizan (Biba, 1975).

2.2. Elementos

El modelo Biba utiliza diferentes elementos para su funcionamiento. Estos elementos son:

- Sujetos: elementos que realizan el acceso.
- Objetos: elementos a los cuales quiere acceder el sujeto.
- Niveles de seguridad: características del objeto que determinan su acceso.
- Operaciones de acceso: lectura, escritura y lectura/escritura.
- Matriz: elemento en el que se determina si un sujeto puede acceder al objeto de acuerdo con el nivel de seguridad.

Para realizar el proceso de control de acceso, la autorización del sujeto se compara con la clasificación del objeto y luego se aplican las reglas específicas para controlar cómo pueden tener lugar las interacciones entre los objetos.

Como se aprecia en la figura 2, los sujetos y objetos pueden residir en diferentes niveles de seguridad y tienen relaciones y reglas que dictan las actividades aceptables entre ellos. Por esta razón, se considera que es un modelo de seguridad de flujo de información: se enfoca en asegurar que los sujetos estén debidamente autenticados antes de acceder al objeto.

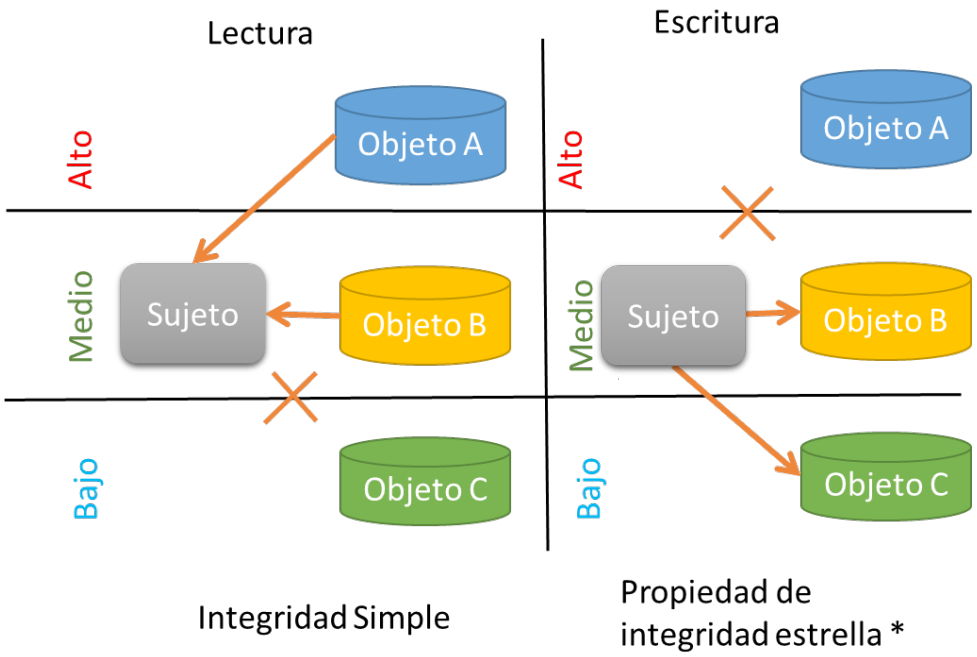

Figura 2. Modelo Biba

Fuente: elaboración propia

2.3. Estructura del modelo

Es un modelo de máquina de estado similar al modelo de Bell-LaPadula. Como ya lo vimos, este último utiliza la estructura de niveles de seguridad (*top secret*, secreto, sensible, etc.) para garantizar que los datos confidenciales solo sean accedidos por sujetos autorizados, mientras que el modelo Biba aborda la integridad de los datos dentro de las aplicaciones y para eso utiliza niveles de seguridad con el fin de evitar que los datos de cualquier nivel de integridad fluyan a un nivel de integridad más alto.

2.4. Consideraciones generales del modelo

Como se ha mencionado a lo largo del texto, el objetivo del modelo Biba es ocuparse de la integridad de los datos, por lo cual se manejan un par de consideraciones. La regla de propiedad estrella * indica que un sujeto no puede escribir datos en un objeto con un nivel de integridad más alto; la etiqueta asignada es "sin escritura". La regla de propiedad simple indica que un sujeto no puede leer datos de un nivel de integridad inferior; su etiqueta es "sin lectura". Esta regla protege los datos en un nivel de integridad más alto para que no se corrompan con los datos en un nivel de integridad inferior; se trata de confiar en la fuente de la información. Otra forma de verlo es que los datos confiables son datos "limpios" y los datos que no son de confianza (desde un nivel de integridad inferior) son datos "sucios". Los datos sucios no deberían mezclarse con datos limpios, ya que eso podría arruinar la integridad de los datos limpios. El axioma de integridad simple se aplica no solo a los usuarios que crean los datos, sino también a los procesos. Un proceso de menor integridad no debería estar escribiendo en datos confiables de un nivel de integridad más alto (Harris, 2013).

3. Modelo Clark-Wilson, 1987

El modelo Clark-Wilson fue desarrollado después de Biba y toma diferentes enfoques para proteger la integridad de la información.

3.1. Historia

El modelo fue descrito en un documento, en 1987 (*Una comparación de las políticas de seguridad informática comerciales y militares*), elaborado por David D. Clark y David R. Wilson, que explica el principio de integridad de la información comparado con los requisitos de los sistemas de seguridad multinivel del departamento de Defensa del Gobierno de los Estados Unidos de Norteamérica. El modelo es aplicable a procesos comerciales e industriales (Clark y Wilson, s.f.).

3.2. Elementos

El modelo utiliza los siguientes elementos:

- Sujetos: usuarios.
- Procedimientos de transformación (TP): operaciones abstractas programadas, como leer, escribir y modificar .
- Elementos de datos restringidos (CDI): pueden ser manipulados solo por TP.

- Elementos de datos no restringidos (UDI): pueden ser manipulados por los usuarios a través de operaciones básicas de lectura y escritura.
- Procedimientos de verificación de integridad (IVP): verifica la consistencia de los CDI con la realidad externa.

3.3. Estructura del modelo

En términos generales, cuando una aplicación utiliza el modelo de Clark-Wilson debe separar los datos que están altamente protegidos o que son restringidos, conocidos como CDI, y los datos que no requieren un alto nivel de protección, denominados UDI. Los usuarios no pueden modificar datos críticos (CDI) directamente; en cambio, el sujeto (usuario) debe estar autenticado con un nivel de permisos suficiente para que el *software* realice procedimientos (TP) en nombre del usuario. Por ejemplo, cuando un sujeto necesita actualizar la información contenida en la base de datos de su compañía, no se le permitirá hacerlo sin un *software* que controle estas actividades, como se muestra en la figura 3. Los pasos a seguir para la operación son:

1. El sujeto debe autenticarse en un programa que actúa como *front-end* para la base de datos.
2. El programa controlará lo que el sujeto puede o no puede realizar con la información contenida en la base de datos. Esto se conoce como triple acceso: sujeto (usuario), programa (TP) y objeto (CDI).

Un usuario no puede modificar CDI sin usar un TP. Si el sujeto intenta ingresar datos para sobrescribir en la base de datos original, el *software* (TP) debe validar si este tipo de operación es segura y llevar a cabo el procedimiento de escritura a nombre del sujeto. El CDI debe tener su integridad protegida por los TP.

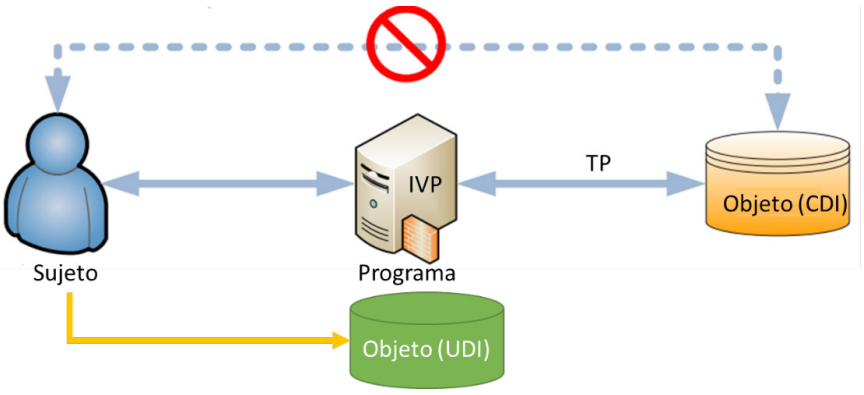

Figura 3. Modelo Clark-Wilson

Fuente: elaboración propia

3.4. Consideraciones generales del modelo

El modelo Clark–Wilson es un modelo de integridad, por lo cual se debe garantizar que se utilicen reglas específicas de integridad. Este trabajo lo realiza el IVP de acuerdo con la definición realizada previamente. Es decir, el modelo indica qué se debe realizar brindando un marco de referencia para construir una determinada característica en el *software* (confidencialidad, integridad), pero no indica las reglas específicas a implementar, pues esto depende de las necesidades de la aplicación, los conocimientos del desarrollador, el enfoque de seguridad de la empresa, etc. Un usuario puede tener acceso a datos de UDI sin el uso de un TP, pero cuando el usuario necesita acceder a CDI debe usar TP. Entonces, el proveedor que desarrolla el producto determinará qué tipo de datos se consideran UDI y qué tipo de datos son CDI y desarrollará los TP para controlar y asignar los valores de integridad del CDI (Harris, 2013).

El uso de TP para modificar los CDI se conoce como una transacción bien formada, la cual consiste en una serie de operaciones que se llevan a cabo para transferir los datos de un estado consistente a otro.

Otra manera de realizar modificaciones al CD es por medio de la separación de tareas en la arquitectura de la aplicación. Por ejemplo, una aplicación que requiere la adquisición de elementos en línea para su desarrollo (un videojuego); si inicialmente contaba con 2 elementos y se adquieren 2 más, el CDI debe tener un valor de 4. El IVP garantiza la coherencia de datos. Después de que el sujeto realiza la actividad (comprar, ganar, adquirir, etc.) y el IVP valida la integridad de CDI (el nuevo valor de elementos es correcto), se considera que el CDI se encuentra en un estado consistente. Si el sujeto decide transferir elementos a otro usuario (videojuego en línea), se realizan dos operaciones: resta de elementos y agregar elementos a un usuario diferente. Al asegurarse que los datos son correctos en ambos sentidos, se mantiene la consistencia interne y externa.

Si continuamos con nuestro ejemplo del videojuego y se requiere adquirir elementos utilizando pago para ello, la aplicación necesita que se realice alguna validación para efectuarlo. Esta validación garantiza que se tenga el nivel de permisos para acceder a esa parte del *software*.

En conclusión, el modelo proporciona las reglas que los desarrolladores deben seguir para implementar y hacer cumplir adecuadamente la separación de funciones a través de procedimientos de *software*.

3.5. Aplicaciones

Los modelos de integridad tienen los siguientes objetivos:

- Evitar que usuarios no autorizados realicen modificaciones.
- Evitar que los usuarios autorizados realicen modificaciones incorrectas (separación de tareas).
- Mantener la coherencia interna y externa (transacción bien formada).

Por lo anterior, los modelos de integridad se aplican para desarrollo de software seguro en diferentes niveles.

El modelo Clark-Wilson cumple con los tres objetivos, mientras que el modelo Biba abarca tan solo el primero. La consistencia interna y externa es provista por el IVP, que asegura que lo que está almacenado en el sistema como CDI corresponda adecuadamente con el valor de entrada que modificó su estado; utiliza el sistema de triple acceso (sujeto, *software* [TP], objeto), separación de tareas y auditoría y usa el concepto de transacciones bien formadas para completar la validación de integridad (Harris, 2013).

4. Modelo Muralla China, 1989

Es también llamado el modelo Brewer y Nash y fue creado con el fin de permitir control de acceso dinámico de acuerdo con las actividades previas del usuario.

4.1. Historia

El modelo Brewer–Nash, o Muralla China, es un modelo de seguridad de la información que fue creado en 1989 por el Dr. David F.C Brewer y el Dr. Michael J. Nash, en el Reino Unido y con el patrocinio de Gamma Secure Systems Limited (Nash, s.f.).

4.2. Elementos

A diferencia de los otros modelos, no se establecen niveles de seguridad y objetos propiamente dichos. Un sujeto o usuario cuenta con acceso a la información de manera dinámica de acuerdo con la información que accedió previamente, con el fin de evitar conflicto de intereses.

4.3. Estructura del modelo

El modelo de Brewer y Nash, también llamado modelo de la Muralla China, se creó para proporcionar control de acceso dinámico de acuerdo con las acciones del usuario.

A diferencia de los modelos anteriormente estudiados en los cuales se identificaban niveles de seguridad, en este se valida previamente si el sujeto ha ingresado anteriormente a datos diferentes. El objetivo principal del modelo es protegerse contra los conflictos de intereses que pueden existir por acceso no controlado. Como se muestra en la figura 4, una empresa que ofrece un servicio en particular, por ejemplo, *outsourcing* de nómina, cuenta con dos clientes. Un usuario que trabaje para el cliente A no debería ver la información del cliente B, puesto que tal acción podría crear un conflicto de intereses si, por ejemplo, los clientes son competencia, hacen parte de una fusión etc.

Estos controles de acceso cambian dinámicamente dependiendo de las autorizaciones, actividades y solicitudes de acceso previas del usuario. El modelo de la muralla china es considerado también un modelo de flujo de información porque esta no puede fluir entre sujetos y objetos, de manera que resulte en un conflicto de intereses.

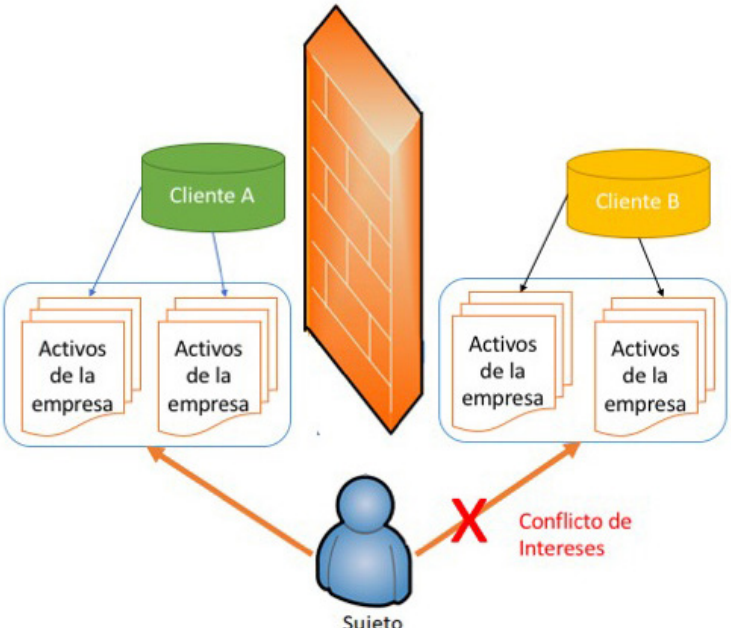

Figura 4. Modelo Brewer-Nash

Fuente: elaboración propia

4.4. Consideraciones generales del modelo

El modelo establece que un sujeto puede escribir en un objeto si, y solo si, el sujeto no puede leer otro objeto que está en un conjunto de datos diferente. Entonces, si nos quedamos con nuestro ejemplo, el usuario no podría escribir en ningún objeto dentro del conjunto de datos del cliente A si actualmente tiene acceso de lectura a cualquier objeto en el conjunto de datos del cliente B.

4.5. Aplicaciones

Este modelo es aplicable a cualquier empresa que presente diferentes tipos de conflictos de intereses en el flujo de información. La mayoría de las aplicaciones son de tipo comercial, más que militar, como los modelos anteriores.

Referencias

Bell, D. E. (2005). *Looking Back at the Bell-La Padula Model.* Recuperado de http://seclab.cs.ucdavis.edu/projects/history/papers/biba75.pdfhttps://www.acsac.org/2005/papers/Bell.pdf

Biba, K. J. (1975). *Integrity Considerations for Secure Computer Systems.*

Clark, D. y Wilson, D. (s.f.). *A Comparison of Comercial and Military Computer Security Policies.* Recuperado de http://theory.stanford.edu/~ninghui/courses/Fall03/papers/clark_wilson.pdf

Harris, S. (2013). *All in One CISSP Exam Guide.* New York: McGraw Hill.

Nash, D. D. (s.f.). *The Chinese Wall Security Policy.* Recuperado de https://www.cs.purdue.edu/homes/ninghui/readings/AccessControl/brewer_nash_89.pdf

INFORMACIÓN TÉCNICA

FACULTAD DE
INGENIERÍA, DISEÑO
E INNOVACIÓN

Módulo: Teoría de la Seguridad

Unidad 2: Modelos de seguridad de la información

Escenario 3: Modelos tradicionales de seguridad de la información

Autor: Alexandra Peña Daza

Asesor Pedagógico: Angie Viviana Laitón Fandiño

Diseñador gráfico: Kelly Valencia

Asistente: Alejandra Morales

Este material pertenece al Politécnico Grancolombiano.

Prohibida su reproducción total o parcial.