Tratamiento seguro de llaves criptográficas hacia KeyBlock y TR-31

10/10/2023

utimaco®

Índice

- La normartiva
- La criptografía en la banca
- ¿Qué es un HSM?
- Custodia de llaves (Variant vs KeyBlock)
- ¿Por qué “variant” no es considerado seguro?
- La Solución

La normativa

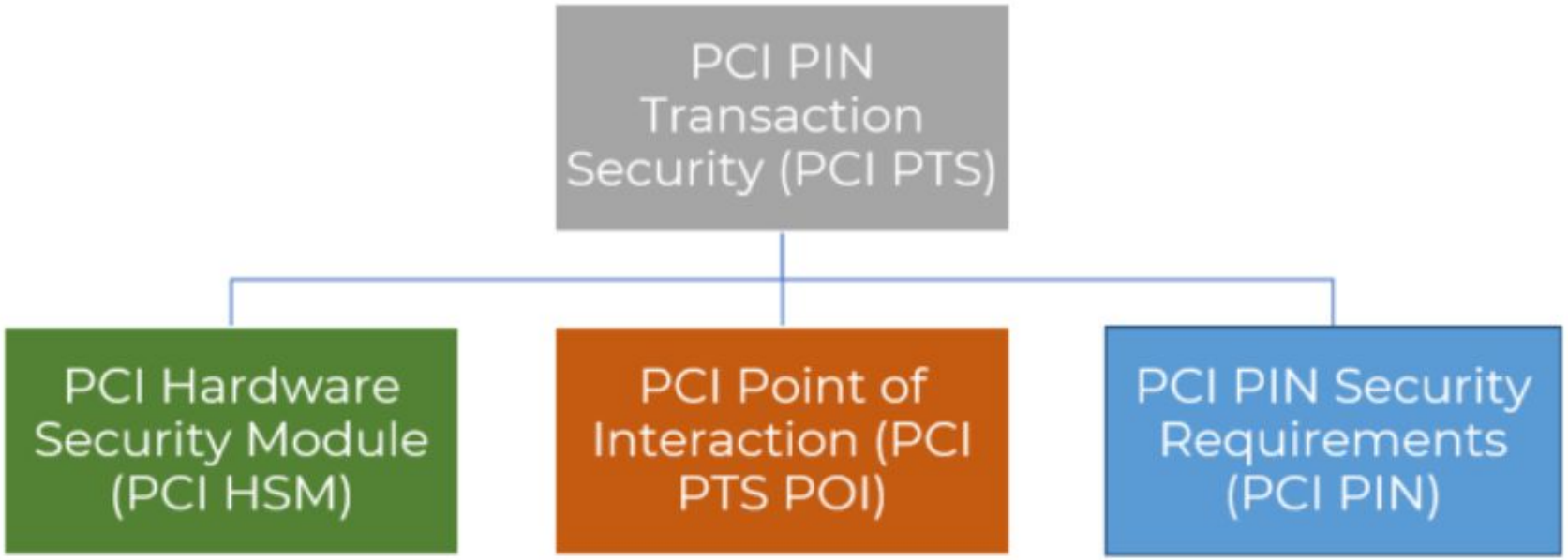
utimaco®

Normativa

Payment Card Industry (PCI)
PIN Security

Normativa PCI PIN - ¿Qué es?

El estándar Payment Card Industry (PCI) PIN Security (o PCI PIN) es un estándar de seguridad que establece los requerimientos para la gestión segura, procesamiento y transmisión del PIN durante el procesamiento de transacciones de pago en línea y fuera de línea en cajeros electrónicos (ATM) y en terminales de punto de venta (POS).



*Normativa PCI PIN – ¿**Qué** está afectado?*

Afecta a todas las llaves criptográficas utilizadas en relación con la adquisición y el procesamiento de datos de PIN.

- ✔ Zone Master Keys (ZMKs).
- ✔ Key Encipherment Keys (KEKs).
- ✔ Base Derivation Keys (BDKs).
- ✔ Terminal Master Keys (TMKs).
- ✔ PIN Encryption Keys (PEKs).
- ✔ Zone Pin Keys (ZPK).

*Normativa PCI PIN – ¿**Quién** debe cumplir?*

El estándar PCI PIN es de **obligatorio cumplimiento** para todas las instituciones adquirientes y agentes responsables del procesamiento de transacciones con PIN de las tarjetas de las marcas del PCI SSC (VISA, MasterCard, AMEX, Discover y JCB) incluyendo servicios de inyección de claves y distribución de claves simétricas usando claves asimétricas (RKL).

*Normativa PCI PIN – ¿**Cómo** se ha de cumplir?*

Objetivo 5: Requisito 18-3:

Las llaves simétricas cifradas deben gestionarse en estructuras llamadas Bloques de Llaves (**Key Blocks**). El uso de la llave debe estar unido criptográficamente a la llave mediante métodos aceptados*, de tal manera que debe ser inviable que la llave sea utilizada si los atributos de uso han sido alterados.

*Normativa PCI PIN - ¿**Cómo** se ha de cumplir?*

1. Los PIN utilizados en las transacciones se han de procesar con equipo y metodologías que garantizan su seguridad. *Solución: HSM (PCI PTS HSM)*
2. Las llaves criptográficas utilizadas para el cifrado y descifrado de PIN y la administración de llaves se han de crear mediante procesos que garanticen que no sea posible predecir ninguna llave ni determinar que ciertas llaves sean más probables que otras. *Solución: HSM con generador números aleatorios certificado PCI PTS HSM.*
3. Las llaves se transmiten de forma segura. *Solución: TR-31*
4. La carga de llaves en los dispositivos HSM y POI con aceptación de PIN se maneja de forma segura. *Solución: Trusted Path (KLD)*
5. Las llaves se utilizan de forma que se evite o detecte su uso no autorizado. *Solución: KeyBlock*
6. Las llaves se administran de forma segura. *Solución: KeyBlock*
7. El equipo utilizado para procesar los PIN y las llaves se administra de forma segura. *Solución: HSM (PCI PTS HSM)*

Normativa PCI PIN - ¿Cuándo?

utimaco®

- **Fase 1:** Implementación de KeyBlock para conexiones internas y almacenamiento de llaves en entornos de proveedores de servicios; esto incluiría todas las aplicaciones y bases de datos conectadas a módulos de seguridad de hardware (HSM).

  Fecha de entrada en vigor: 1 de junio de 2019. (Completo).

- **Fase 2:** Implementación de Keyblock para conexiones externas a asociaciones y redes.

  Fecha de entrada en vigor: 1 de enero de 2023 .

- **Fase 3:** Implementación del Keyblock para extenderlo a todos los hosts de comerciantes, dispositivos de punto de venta (POS) y cajeros automáticos.

  Fecha de entrada en vigor: 1 de enero de 2025.*

*** Marzo 2021 - PCI Security Standards Council (PCI SSC);** suspendió las fechas para implementar Pinblock formato ISO 4. Las fechas de vigencia se indicarán en un comunicado posterior.

Criptografía en la banca

utimaco®

Criptografía Simétrica

Aquella en la que se utiliza **la misma** llave para cifrar el mensaje en el origen y para descifrar el mensaje cifrado en el destino.

Criptografía Simétrica

Usos más frecuentes:

1. Transporte de llaves.
2. Cifrado de información.

Características principales:

1. Algoritmos MUY rápidos.
2. Garantizan la confidencialidad.
3. Garantizar la integridad del mensaje.

Principales tipos:

1. DES o TDEA
2. AES

Llaves criptográficas

Parece que las llaves son muy importantes, ¿verdad?, entonces……

……¿dónde las guardamos de manera segura?

HSM

Hardware Security Module

¿Qué es un HSM?

utimaco®

¿Qué es un HSM?

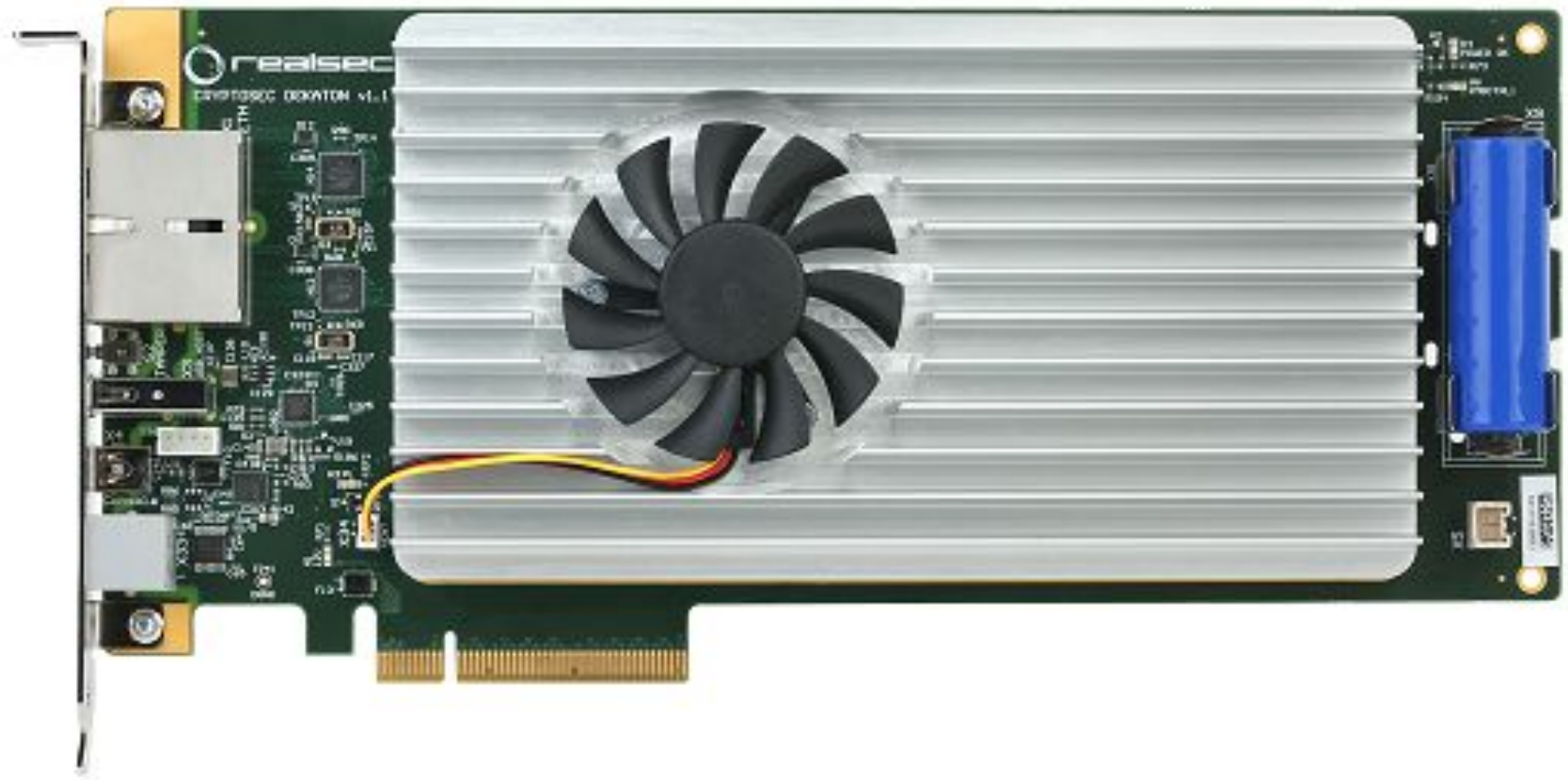
Un HSM es un dispositivo criptográfico basado en hardware que **genera**, **almacena** y **protege** llaves criptográficas y suele aportar aceleración hardware para operaciones criptográficas.

Genera: es el HSM quien, utilizando un generador de números aleatorios, crea de manera totalmente aleatoria aquellas llaves de trabajo (simétricas o asimétricas) que posteriormente utilizará.

Almacena: el HSM guarda internamente las llaves de trabajo cifradas por una llave maestra (MasterKey) o una LMK (Local Master Key) que NUNCA salen fuera del perímetro de seguridad del HSM.

Protege: el HSM mantiene su Master Key en una zona de memoria de un chip dedicado a tal efecto.... + malla metálica + resina epoxy + carcasa metálica + contramedidas frente ataques (voltaje o temperatura).

Esto es un HSM



Formato HSM más conocido

Custodia de llaves en HSMs (Variant vs KeyBlock)

utimaco®

Premisas

1. Las llaves criptográficas NUNCA se utilizan en texto plano (en claro), siempre se han de almacenar y usar encriptadas bajo la una LMK o la MasterKey del HSM que va a hacer uso de ellas. A esas llaves cifradas se las conoce como "*criptograma*".

2. Las llaves criptográficas son creadas para un único propósito y para evitar sean usadas para cualquier otra cosa, una vez se generan son cifradas con una LMK en concreto. Hay tantas LMKs como propósitos (cifrar PINes, cifrar datos, generar MAC, calcular PIN, etc.).

¿Qué es variant y KeyBlock?

Variant y KeyBlock son dos formatos de criptogramas, es decir, llaves que están protegidas por la MasterKey o una LMK de un HSM y que se encuentran almacenadas en una BB.DD a la espera de ser utilizadas en alguna transacción financiera.

Variant es el formato que se ha venido utilizando hasta la fecha y que poco a poco por normativa PCI, están siendo migradas a un formato más robusto....el que se conoce como “Bloque de llave” o “KeyBlock”.

¿Cómo es una llave variant?

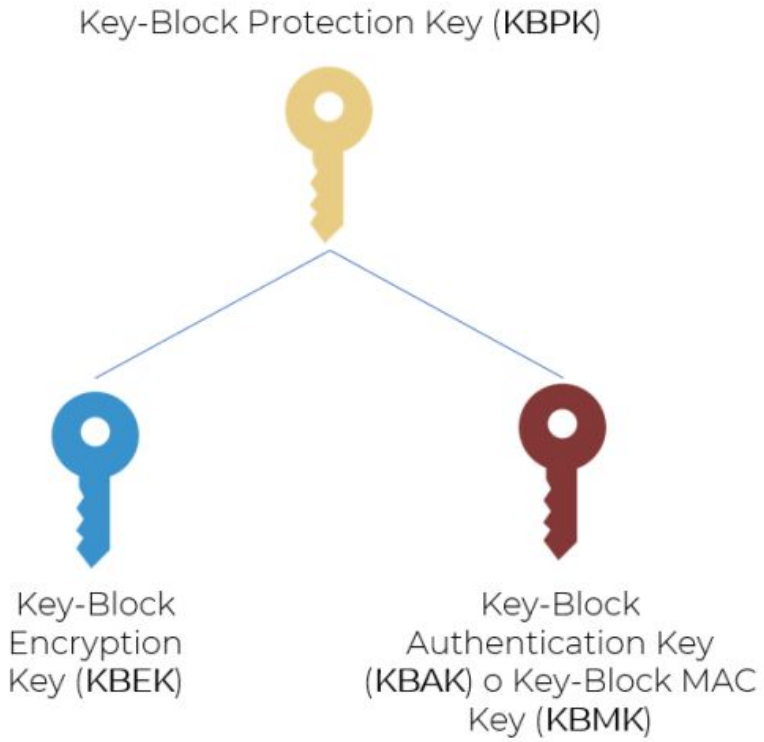
- Una llave formato “variant” es aquella llave a la que se le ha aplicado una máscara (XOR) antes de ser protegida por la LMK del HSM. Esta máscara es diferente según el fabricante del HSM. No lleva ningún metadato que expliquen sus limitaciones de uso.
- Al ser un método propietario de cada fabricante de HSM, no se puede utilizar para distribuir llaves entre HSMs de diferentes marcas o hacia TPVs o ATM (a menos que lo implementen expresamente).
- Por lo tanto, el método de exportación más utilizado hasta ahora es el ANSI, en el que exclusivamente la llave en plano se cifra con la llave de transporte, perdiendo entonces toda traza del propósito para el que se creó la llave.
- Este método no provee ninguna funcionalidad para verificación de la integridad o autenticación de la clave. *Se le considera un bloque de llave* ***inseguro***.

¿Cómo es una llave KeyBlock?

utimaco®

- Una llave formato “KeyBlock” es aquella a la que además de la propia llave, se le han añadido:
  - Una cabecera (*header*) la cual DEBE indicar el USO y limitaciones de la llave.
  - Un MAC para poder comprobar la integridad.
- Una llave en formato KeyBlock es protegida por dos llaves diferentes que se derivan de una llave común (KBPK – Key Block Protection Key):
  - una llave para el cifrado de la llave (KBEK – Key Block Encryption Key)
  - una llave para el cálculo del MAC de todo el bloque (KBAK - Key Block Authentication Key)
- Este formato de criptogramas *se le considera un bloque de llave* ***seguro***.

¿Cómo es una llave KeyBlock?



Variant vs KeyBlock

Ejemplo de una llave (3DES2) en formato:

- Variant:

94B420C80BA3461F86FE26EFC4A3B8E4

- KeyBlock:

B0080P0TE00E000094B420079CC80B89A89B6756D098E08C098A54243DE0983461
F86FE26EFC4A3B8E4FA4C5F534116EED7B727B8A248E

Donde:

B0080...E000 □ Cabecera
094B4.....FC4A □ Llave criptográfica
3B8E....248E □ MAC (Autenticación)

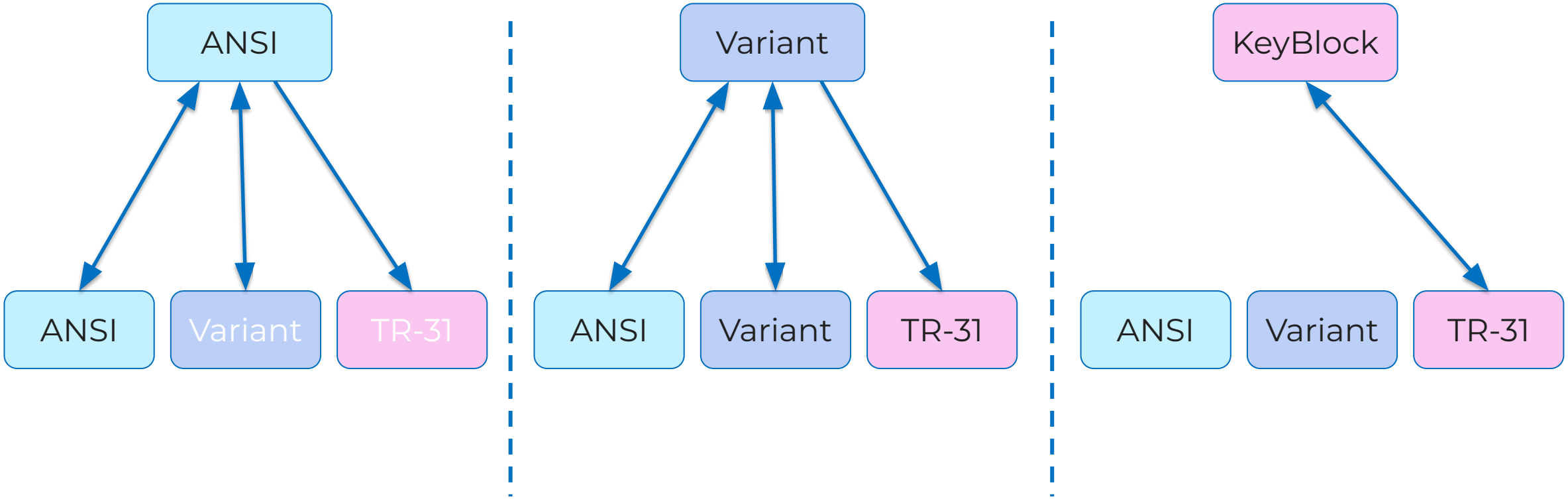
Distribución de llaves (Variant vs KeyBlock)

- Existen principalmente tres formatos para distribución de llaves simétricas en entornos bancarios:
  1. Variant (propietario e inseguro)
  2. ANSI (estándar pero inseguro)
  3. TR-31 (estándar y seguro)

- IMPORTANTE : **No permitir el transportar una llave que se encuentre en un formato considerado seguro a uno que se considere inseguro. Esto se hace con el campo "Exportability" que se encuentra en la cabecera del KeyBlock.**

Distribución de llaves (Variant vs KeyBlock)

Alternativas de exportación/importación de llaves son:



TR-31 – Ejemplo

Ejemplo de una llave exportada en TR-31

```
B0080P0TE00E0000 94B420079CC80BA3461F86FE26EFC4A3B8E4FA4C5F534117
6EED7B727B8A248E
```

	Field	Description
B	Version ID	Key block protegido usando TDEA Key Derivation Binding Method (TDEA-CMAC).
0080	Longitud del Key Block	Longitud en bytes de todo el bloque de llave (KBH + lave cifrada + MAC)
P0	Propósito	'P0' = Llave para cifrado de PINes
T	Algoritmo	T = Triple DEA.
D	Modo de uso	D = Sólo para descifrar or importar.
00	Número de versión	Número de version que le podemos dar a la llave
E	Exportabilidad	Puede ser exportado o no y hacia que tipo de formatos.
00	Numero de bloques opcionales	Cero
00	RFU	Cero
94B4.. 4117	Criptograma de la llave	Criptograma con la llave
6EED..24 8E	MAC	Valor de MAC para verificar la integridad del bloque

¿Por qué “Variant” no es seguro?

utimaco®

Consideraciones previas

Para poder explotar la vulnerabilidad de las llaves en formato Variant, hay que entender tres consideraciones:

1. Siempre hay una necesidad de exportar llaves para trasladarlas a un tercero (banco, ATM, TPV, etc).

2. El formato Variant es propietario de cada marca de HSM.

3. Al ser un formato propietario, en la mayoría de los casos se opta por exportar las llaves en formato ANSI. Por ser el método más sencillo y que todos los fabricantes soportan.

¡¡ Brecha de seguridad !!

Diagrama de flujo de una transacción

1. El tarjetahabiente introduce el PIN de su tarjeta (ej. 2309)

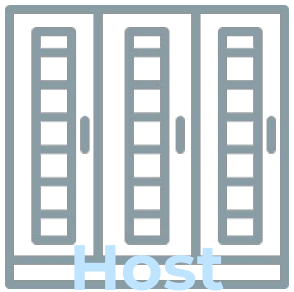
2. El EPP cifra el PIN introducido con la TPK (ej. 866A6BE7487A1846)

3. El PIN cifrado junto con otros datos son enviados al Host del Banco o Switch

8098098b&09f'9867f878e978c98546
35scba789a**866A6BE7487A1846**6a5s
4f78y8b978jñlk*a6547f65as8798498#
45987be0897f2B

4. El Host le envía una petición de verificación de PIN al HSM enviando el PIN y la TPK

5. El HSM responde con un OK si la verificación es satisfactoria.



6. Host envía la respuesta de PIN OK al ATM

PIN CORRECTO

Puntos claves

Los PINes de los tarjetahabientes viajan cifrados bajo una llave simétrica (TPK).

Esa misma TPK es la que utiliza el HSM para descifrar el PIN internamente y verificar si es el correcto.

Demo (toma los datos para el ataque)

utimaco®

- Trabajo en un banco en el que las llaves se encuentran almacenadas en una BBDD en formato Variant.

- Tengo acceso de lectura a la BBDD donde se encuentran esas llaves.

- Obtengo todo lo que necesito:

  1 - Llave de transporte (ZMK): UC974A661196426B4C195C04C8248C4DC
  2 - Llave de transporte de PIN (TPK): U861CC102FCDBD155B3461FE2ECE49555

Demo (preparando el ataque)

- Como tambien tengo acceso al HSM:

1 – Envío un comando de exportación al HSM para exportar la TPK en formato ANSI (porque está permitido) utilizando la ZMK que he obtenido de la BBDD:

[COMMAND]
HEADA870DUC974A661196426B4C195C04C8248C4DCU861CC102FCDBD155B3461FE2ECE49555X
[RESPONSE]
HEADA900XDB6D6E1C68F04C55348B83C45523E3074A2CF9

2 – Aprovechando que la llave exportada ha perdido toda información en cuanto a su propósito, envío un comando de importación al HSM para importar la que era una TPK pero ahora convirtiéndola en una llave que sirva para cifrar/descifrar datos (DEK):

[COMMAND]
HEADA600BUC974A661196426B4C195C04C8248C4DCXDB6D6E1C68F04C55348B83C45523E3074U
[RESPONSE]
HEADA700**U7E46F6DDB0DCA3772B985D90B4D1D089**4A2CF9

Demo (el ataque)

1 – Consigo una trama (mediante sniffing o similar) de una transacción exitosa de retiro de efectivo y localizo el:

- PAN de la tarjeta (4978539460000200)
- El bloque de PIN cifrado con la TPK (866A6BE7487A1846)

2 – Envío un comando para descifrar el bloque de PIN :

```
[COMMAND]
HEADM2001000BU7E46F6DDB0DCA3772B985D90B4D1D0890010866A6BE7487A1846
[RESPONSE]
HEADM3000010004238CC6B9FFFFDF
```

3 – Deshago la máscara haciendo un XOR con el PAN y .....

```
      04238CC6B9FFFFDF
XOR   000853946000020
      ----------------
      042309FFFFFFFFFF
```

La Solución: KeyBlock y TR-31

utimaco®

La solución

Migrar las llaves que actualmente se almacenan en la/s BBDD de formato variant a KeyBlock marcando que **solo sea exportable a otro formato considerado seguro.**

De esta forma estaremos obligados a utilizar un método de exportación de llaves (comúnmente TR-31) en las que el uso o propósito esté ligado en todo momento a la llave exportada, por lo que los HSMs donde se vayan a importar nuevamente esa llave EXCLUSIVAMENTE se podrán hacer bajo el mismo propósito original.

Thank you

for your attention!

UTIMACO GmbH
Germanusstraße 4
52080 Aachen
Germany
Phone +49 241 1696-0
Web hsm.utimaco.com
E-Mail academy@utimaco.com

UTIMACO Inc.
900 E Hamilton Ave #400,
Campbell, CA 95008
Phone (844) 884 6226