TEMA 3

Seguridad en Sistemas Operativos

Índice

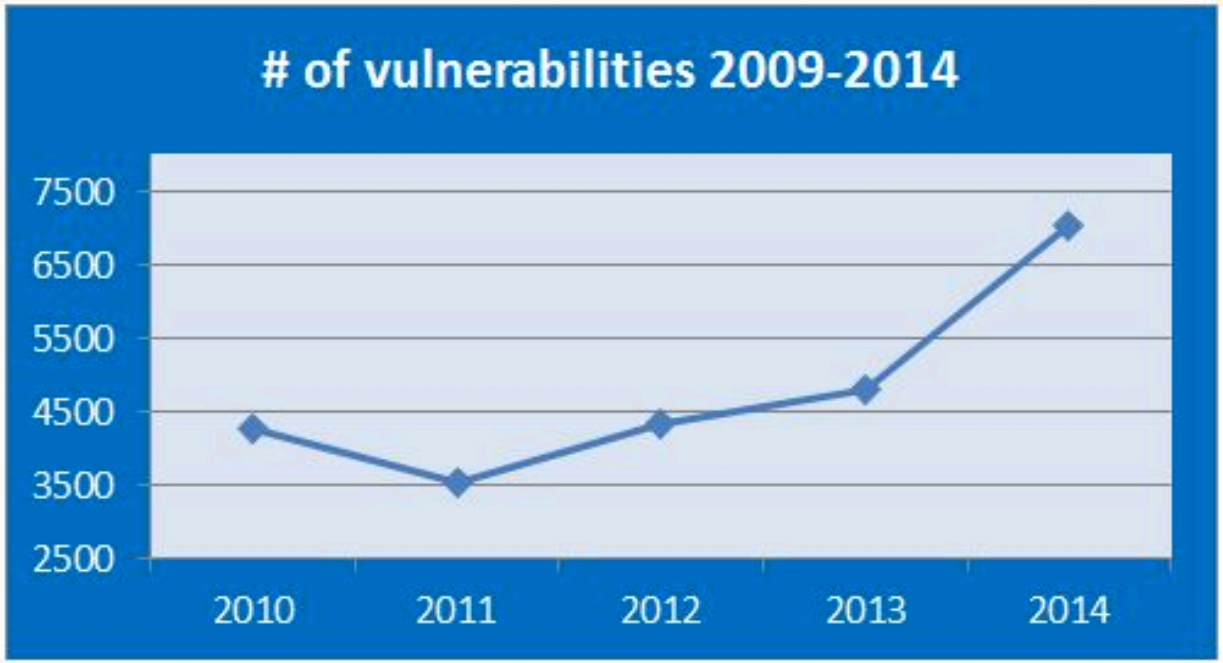
- Seguridad en Sistemas de Escritorio
- Seguridad en Plataformas Móviles
- Seguridad en Entornos Virtualizados

SEGURIDAD EN SISTEMAS DE ESCRITORIO

Vulnerabilidades en los S.O.

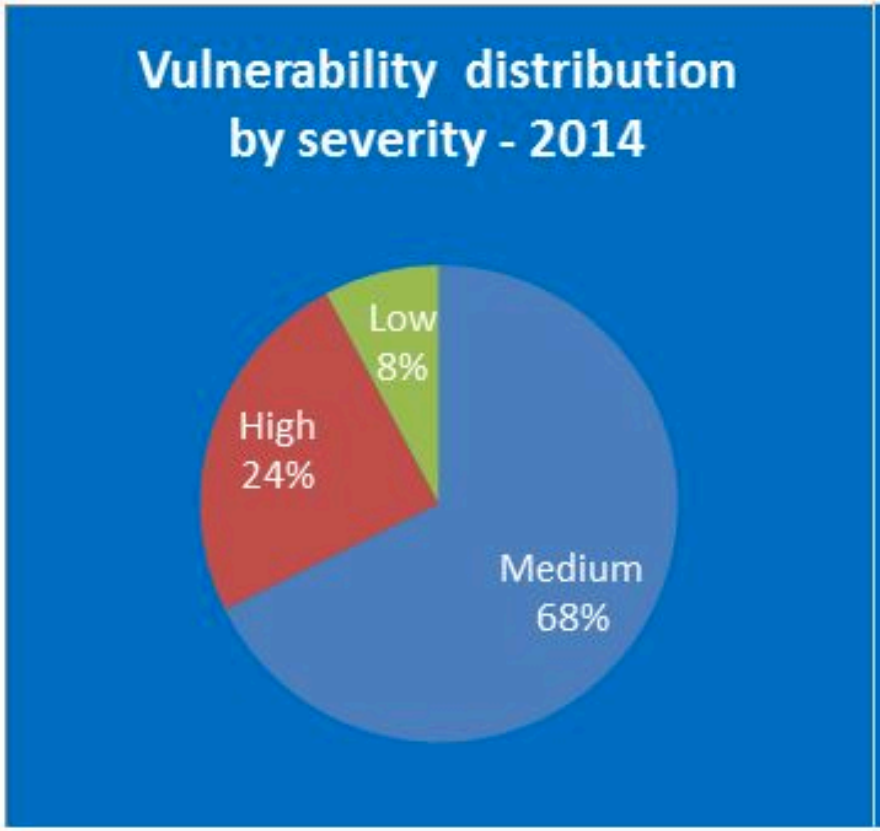
- En 2014 se publicaron una media de 14 vulnerabilidades diarias en la National Vulnerability Database (NVD). Esto hace un total de 7,038 en 2014. Un número significativamente mayor que en 2013, y sigue en progresión creciente.

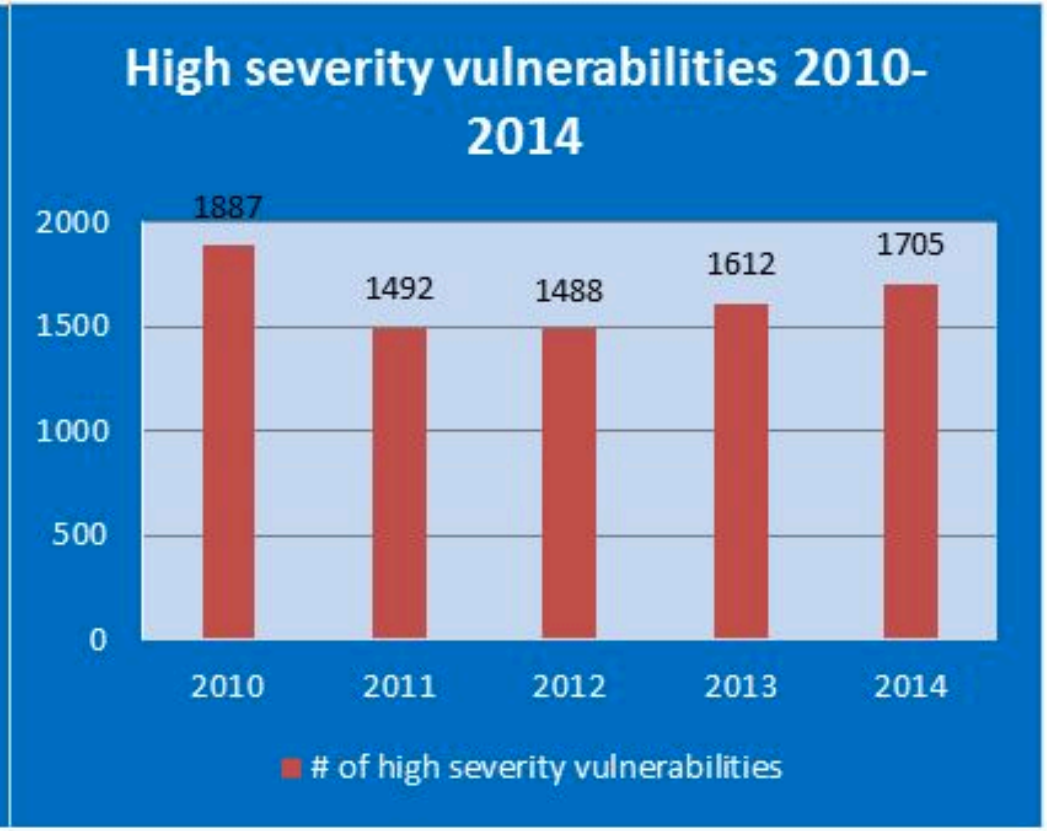
Year	# of vulnerabilities
2010	4,258
2011	3,532
2012	4,347
2013	4,794
2014	7,038



Criticidad de las Vulnerabilidades

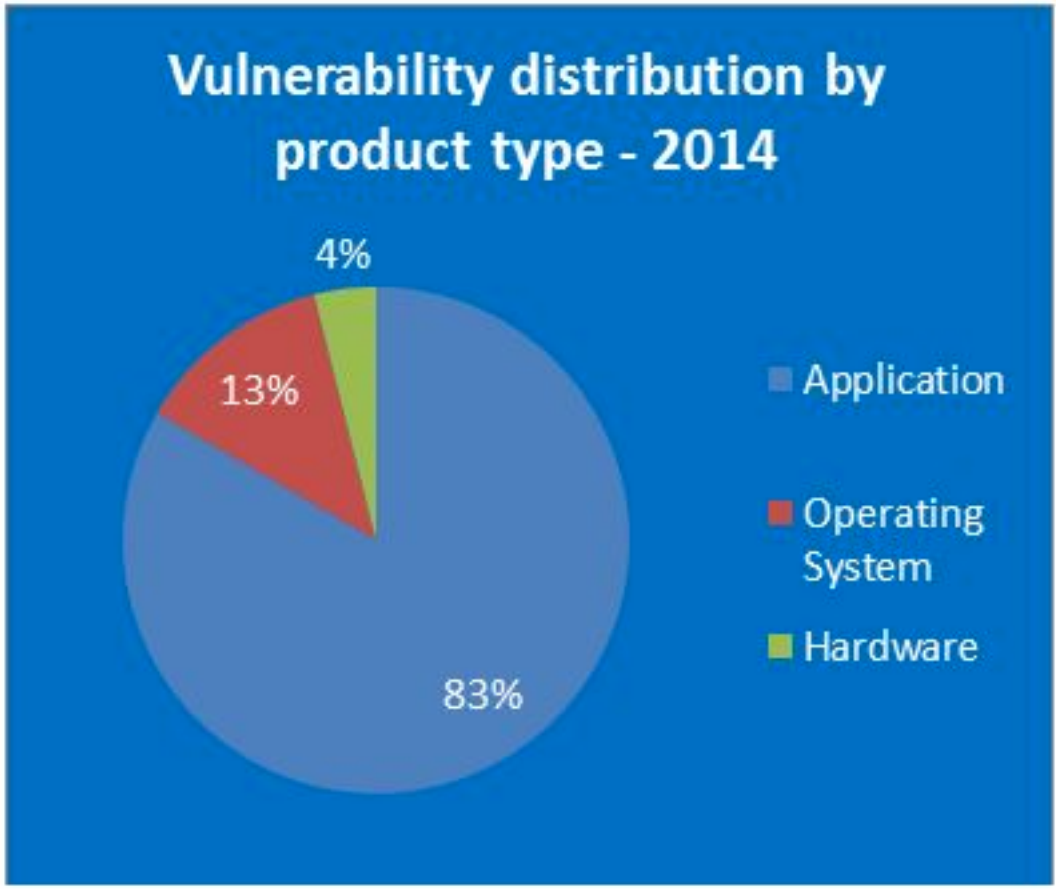
- El 24% de las vulnerabilidades de 2014 están descritas como de alta criticidad. En términos relativos se ha bajado con respecto a 2013, pero se ha subido en términos absolutos.





Fuentes de vulnerabilidad

- La fuente más importante de vulnerabilidades son las aplicaciones de terceros (80%). Los SO solo suponen el 13% y el hardware el 4%



Los 10 SO con más vulnerabilidades

Operating system	# of vulnerabilities	# of HIGH vulnerabilities	# of MEDIUM vulnerabilities	# of LOW vulnerabilities
Apple Mac OS X	147	64	67	16
Apple iOS	127	32	72	23
Linux Kernel	119	24	74	21
Microsoft Windows Server 2008	38	26	12	0
Microsoft Windows 7	36	25	11	0
Microsoft Windows Server 2012	38	24	14	0
Microsoft Windows 8	36	24	12	0
Microsoft Windows 8.1	36	24	12	0
Microsoft Windows Vista	34	23	11	0
Microsoft Windows RT	30	22	8	0

Las aplicaciones con más vulnerabilidades

Application	# of vulnerabilities	# of HIGH vulnerabilities	# of MEDIUM vulnerabilities	# of LOW vulnerabilities
Microsoft Internet Explorer	242	220	22	0
Google Chrome	124	86	38	0
Mozilla Firefox	117	57	57	3
Adobe Flash Player	76	65	11	0
Oracle Java	104	50	46	8
Mozilla Thunderbird	66	36	29	1
Mozilla Firefox ESR	61	35	25	1
Adobe Air	45	38	7	0
Apple TV	86	29	49	8
Adobe Reader	44	37	7	0
Adobe Acrobat	43	35	8	0
Mozilla SeaMonkey	63	28	34	1

¿Donde se publican las Vulnerabilidades?

- **CERT** proviene de las siglas en inglés
  Computer Emergency Response Team
- Common Vulnerabilities and Exposures (CVE®)
- SecurityFocus
- CCN-CERT

Bug Bounty

https://bugcrowd.com/list-of-bug-bounty-programs

openbugbounty Top Security Researchers Open Bug Bounty Full Disclosure Forum About Search

Our Security Community helped fix 23288 vulnerabilities

Report
Open Bug Bounty or Full Disclosure

Email Alerts
Get notifications about your website vulnerabilities

Open Bug Bounty: 8454 coordinated disclosures
Full Disclosure: 75699 vulnerabilities
Total Vulnerabilities Fixed: 23288
70735 vulnerable websites, 9288 VIP websites
1379 security researchers, 1961 notification subscribers

Quickest Patched

Website	Patched in	Patched on	Reported by
groupon.com	1 minute	27.05.2015	Brute
turkcebilgi.com	5 minutes	28.04.2016	MirSultan
biomed.brown.edu	13 minutes	27.04.2016	Dengar
4teachers.de	19 minutes	17.07.2015	MLT

Latest Submissions

Website	Date	Reported by
opednews.com	16.05.2016	crashdemons
careclimatechange.org	16.05.2016	crashdemons
cpj.org	16.05.2016	crashdemons
hibiny-land.ru	16.05.2016	amlnspqr

https://www.openbugbounty.org/

Exploits y Parches

- El ciclo de vida de las vulnerabilidades es el siguiente:
  1. Un experto en seguridad **descubre** una vulnerabilidad
  2. **Avisa** a la empresa que desarrolla el software describiendo el problema y opcionalmente proporciona información para replicar el fallo.
  3. La empresa saca un **parche** de seguridad y actualiza todo su software
  4. La vulnerabilidad se **publica** en los CERT para que los usuarios sepan que tienen que actualizarse en caso de que el proceso no sea automático
  5. Se publica un **exploit** para comprobar si tu sistema es vulnerable.
- El problema surge cuando la empresa no reacciona a tiempo y los pasos 4 y 5 se producen antes que el 3.

Las vulnerabilidades provienen de errores

- El instituto SANS junto con el MITRE y otros expertos de seguridad a nivel internacional definieron en 2011 una lista de los **25** Errores de Software más Peligrosos.
- Esta lista es un buen punto de inicio para aprender a prevenir muchas de las vulnerabilidades más comunes.
- Esta lista está relacionada con el SANS Top 20 y la lista Common Weakness Enumeration (CWE).
- La lista de 25 errores está mantenida por el MITRE y contiene la descripción de los errores así como información sobre como mitigarlos o evitarlos.
- El CWE contiene información sobre más de 800 errores de programación, diseño, y arquitecturales que pueden dar lugar a vulnerabilidades.

Buffer Overflow

- Se puede considerar como uno de los errores más frecuentes y más antiguos. Tiene muchas variantes relacionadas si bien nos vamos a centrar el concepto clásico.
- Responde al principio de que *si quieres meter en un recipiente más contenido de la capacidad disponible algo irá mal.*

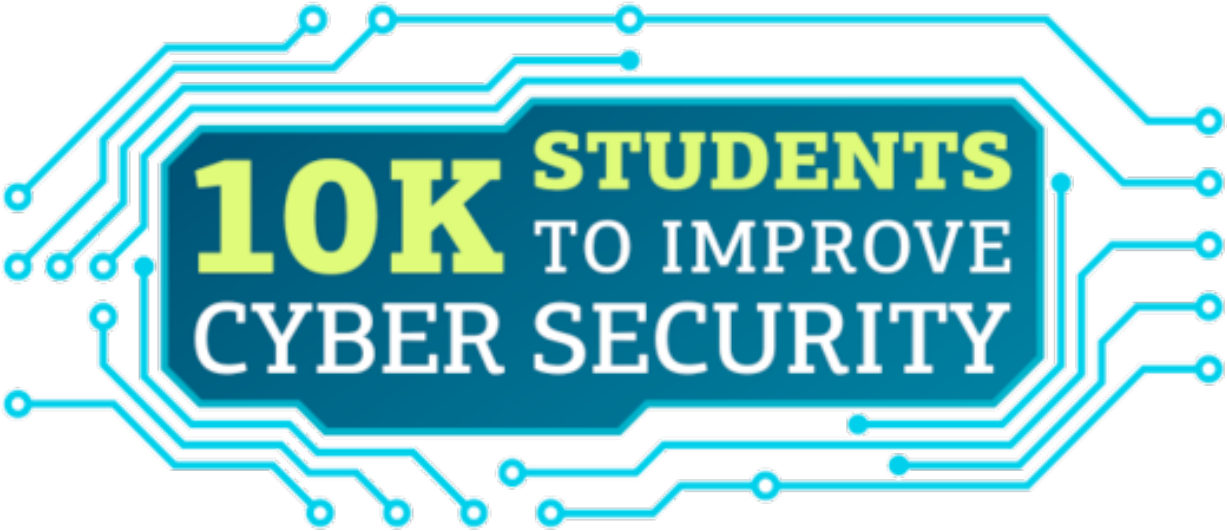
Buffer Overflow.

icanbarelydraw.com CC BY-NC-ND 3.0

Contenido On-line sobre Buffer Overflow

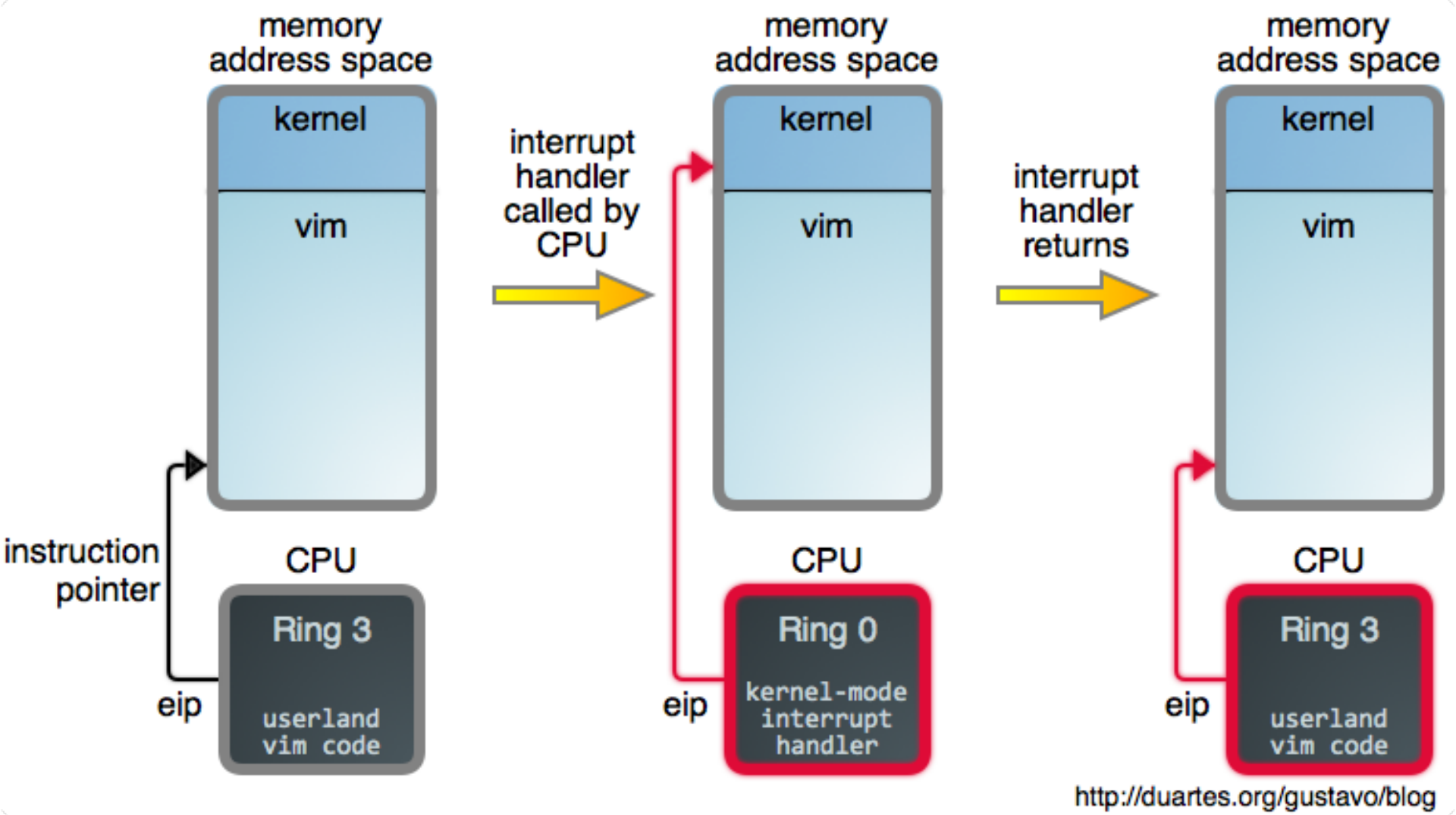
- General Introduction [pptx][pdf][YouTube]
- Part I - Basic Buffer Overflows (Everyone) [pptx][pdf][YouTube]
- Part II - Real Buffer Overflows (Computer Scientists) [pptx][pdf][YouTube]
- Part III - Countermeasures (Students in Security Courses) [pptx][pdf][YouTube]

http://10kstudents.eu



Interrupciones

http://duartes.org/gustavo/blog/post/when-does-your-os-run/



SEGURIDAD EN PLATAFORMAS MÓVILES

Arranque seguro y autenticación del software

- Para garantizar que el dispositivo se va a comportar como debe, es fundamental un mecanismo que gestione el arranque del dispositivo
  - Y autentique todo el software cargado, garantizando que es correcto y que no ha sido manipulado.
- Este proceso es responsable de establecer el estado inicial del sistema sobre el que se cargarán:
  - el sistema operativo
  - y las aplicaciones.
- La manera de implementar este mecanismo de arranque / autenticación suele ser mediante un software que debe ser implícitamente confiable,
  - por lo que suele ser un programa almacenado en ROM / firmware
  - capaz de verificar código firmado digitalmente.

Almacenamiento seguro de información

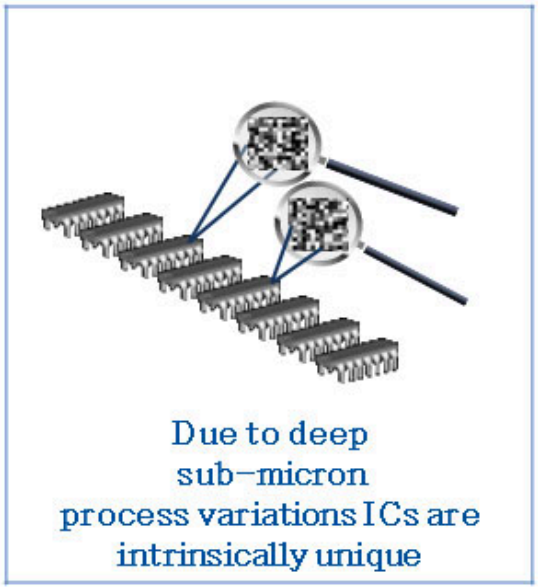
- La información, como claves criptográficas, credenciales de usuario, o datos financieros, debe poder ser protegida contra intrusos.

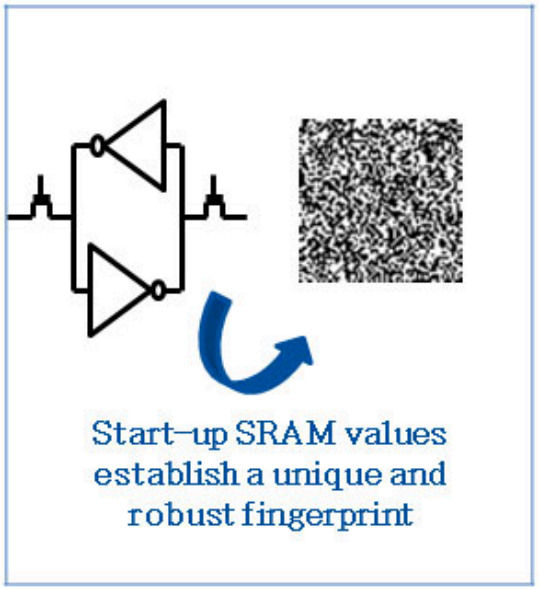
- Dos formas habituales:
  - Almacenamiento cifrado de la información en memoria no volátil (memoria FLASH o incluso pequeños discos duros)
    - lo que a su vez plantea el problema del almacenamiento seguro de las claves de cifrado;

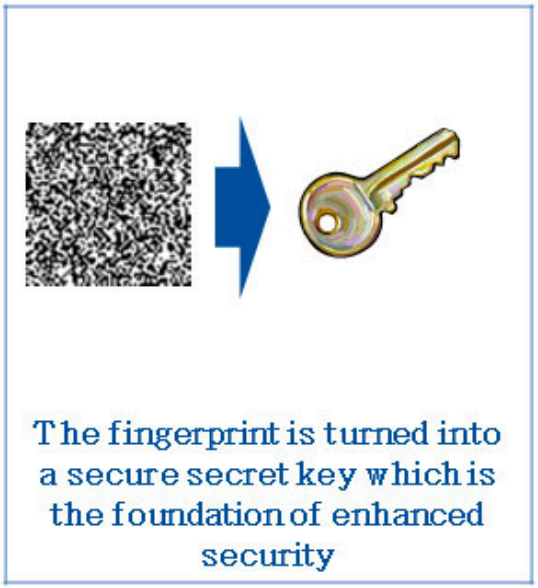
  - Utilizar la tarjeta SIM/USIM o algún tipo de smartcard que disponga de áreas de memoria seguras
    - y que puedan ser empleadas como almacenamiento seguro, especialmente para claves y certificados digitales

Physically Unclonable Functions

- En lugar de almacenar la clave en el dispositivo esta se genera al arrancar a partir de alguna “huella” electrónica del dispositivo, por ejemplo, usando la memoria SRAM.







Entorno de ejecución seguro

- Modo de ejecución dedicado que permita aislar al código y a los datos protegidos del sistema operativo y de las aplicaciones (que son vulnerables).

- Suele requerir de:
  - memoria RAM segura (en la forma de memoria RAM dedicada o como porción de la memoria RAM del sistema) que sólo debe poder ser accesible cuando la CPU se encuentre en Modo Seguro
  - periféricos y caminos de E/S seguros (que sólo puedan ser accedidos cuando el procesador se encuentre en Modo Seguro)
  - mecanismos de autenticación del código que vaya a ser ejecutado en el Modo Seguro.

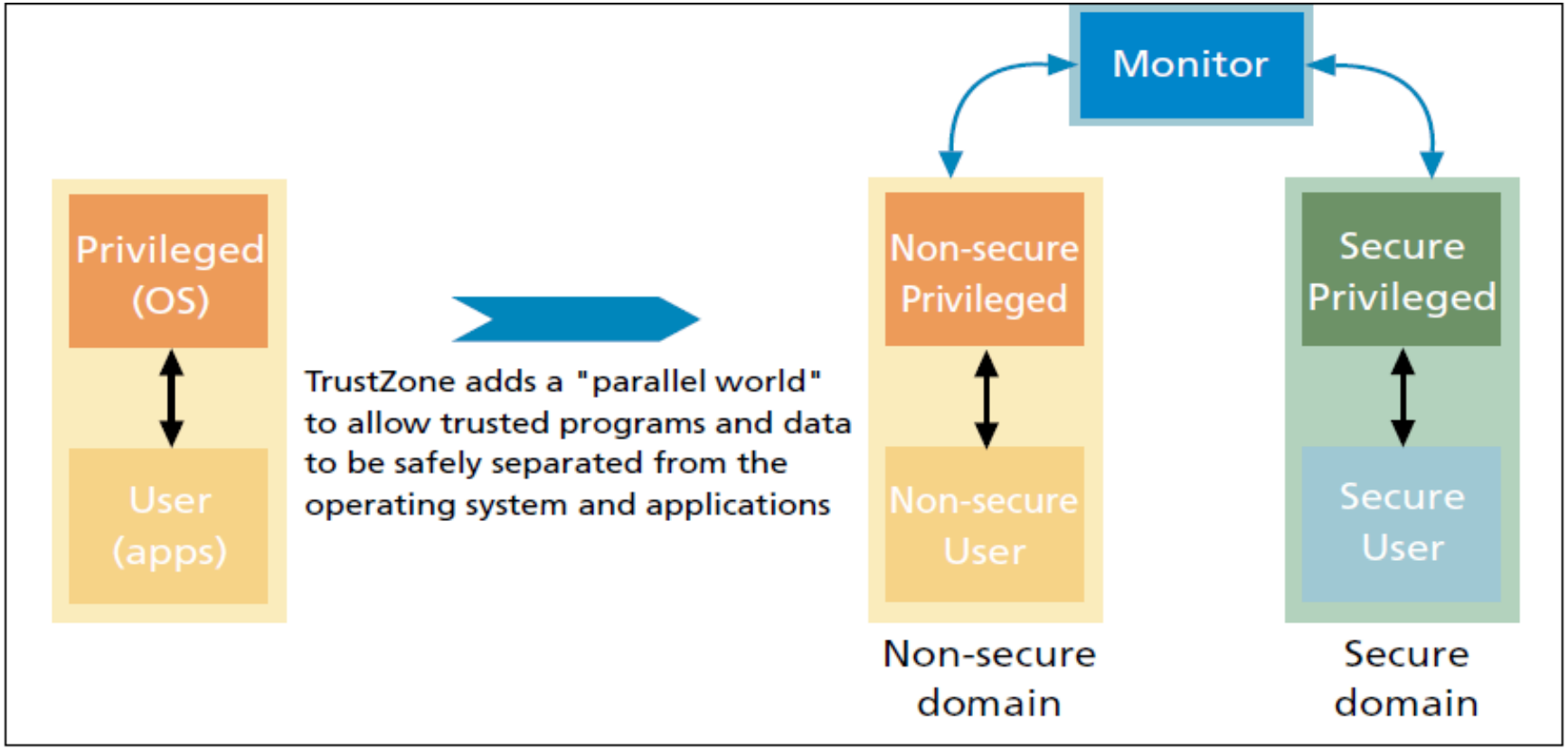
ARM

- ARM es la plataforma hardware más extendida en el sector de los dispositivos móviles
  - es capaz de proporcionar la potencia y las prestaciones necesarias en la mayoría de esos dispositivos
- Hay distintas versiones de la arquitectura ARM y de sus implementaciones derivadas
- La arquitectura ARM (originalmente Acorn RISC Machine) es una arquitectura RISC de 32 bits muy utilizada en un variado número de aplicaciones.
  - Desde 2012 está disponible una arquitectura ARM de 64 bits (ARMv8)

TrustZone

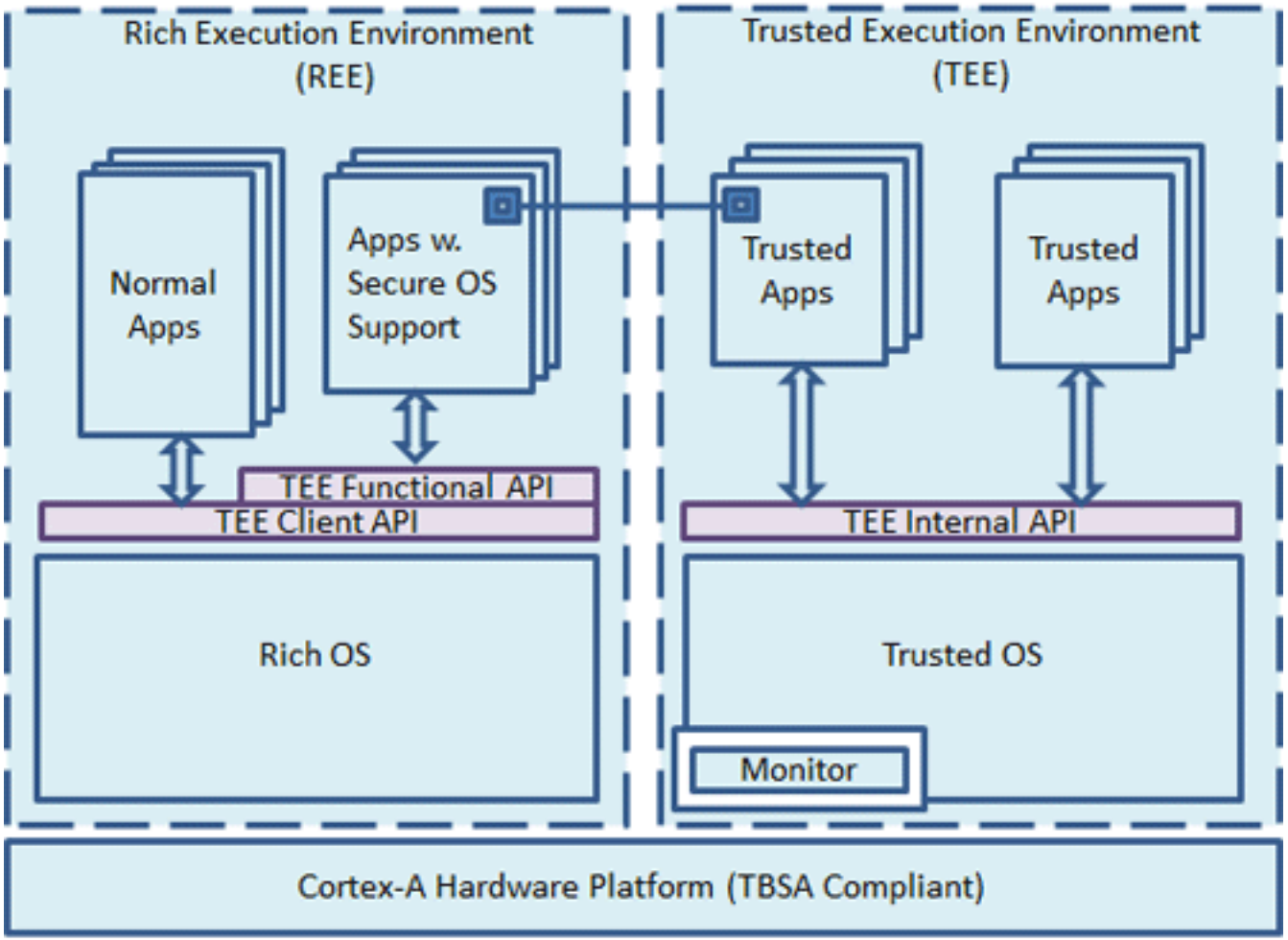
- TrustZone proporciona un entorno de ejecución seguro para plataformas ARM
  - permite a los fabricantes de dispositivos y a los desarrolladores implementar de manera rápida y fácil sus propios sistemas de seguridad orientados a cada aplicación particular, pudiendo utilizar para ello su propio hardware.
- Algunos de los servicios básicos de seguridad:
  - identificación y autenticación,
  - gestión de claves y certificados,
  - criptografía a bajo nivel,
  - control de acceso de E/S y a smart cards,
  - almacenamiento seguro de información
  - realización de chequeos para verificar la integridad de código

TrustZone



http://androidsecurity.weebly.com/mobile-security.html

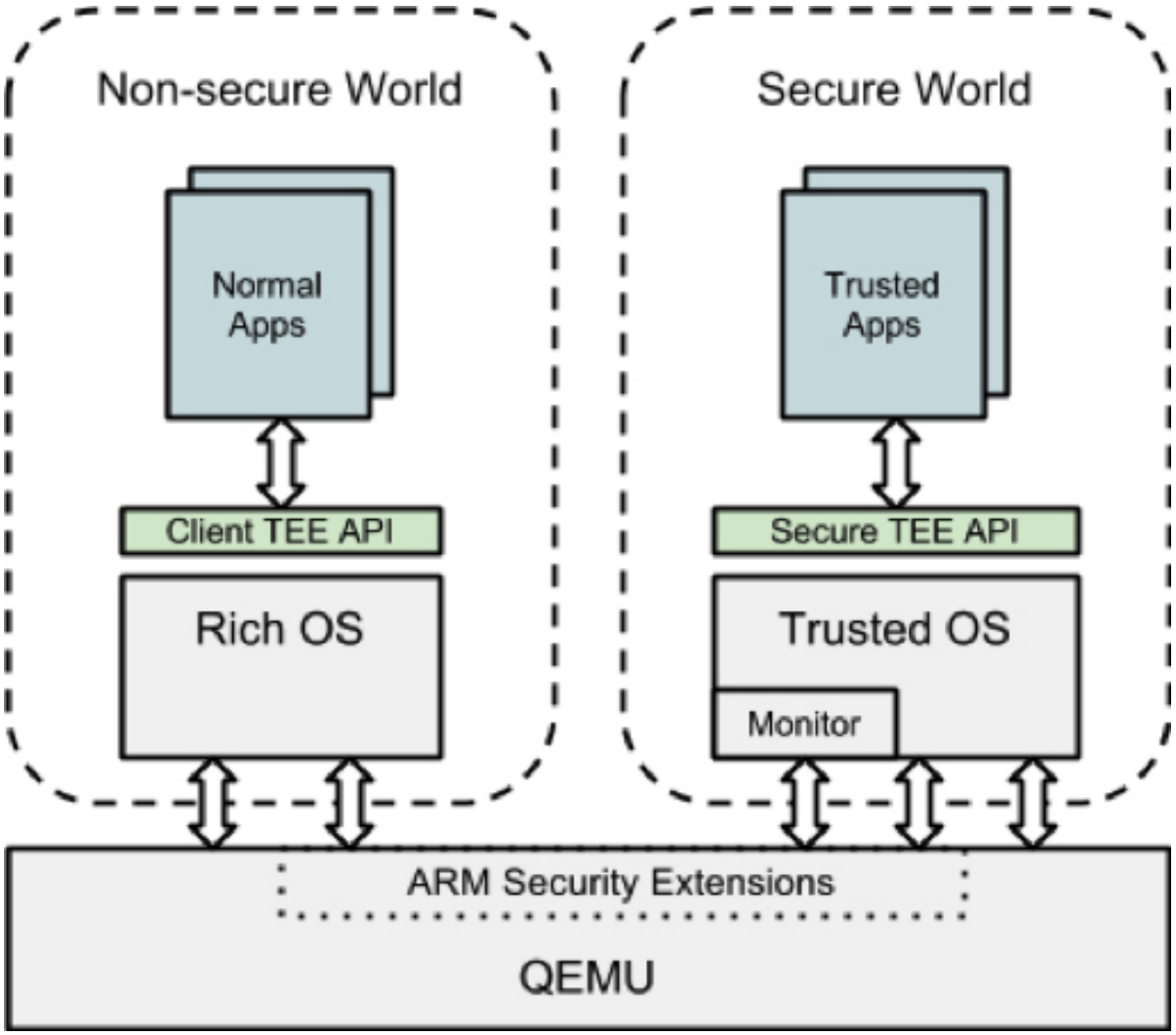
TEE (Trusted Execution Environment)



http://www.arm.com/products/processors/technologies/trustzone/tee-smc.php

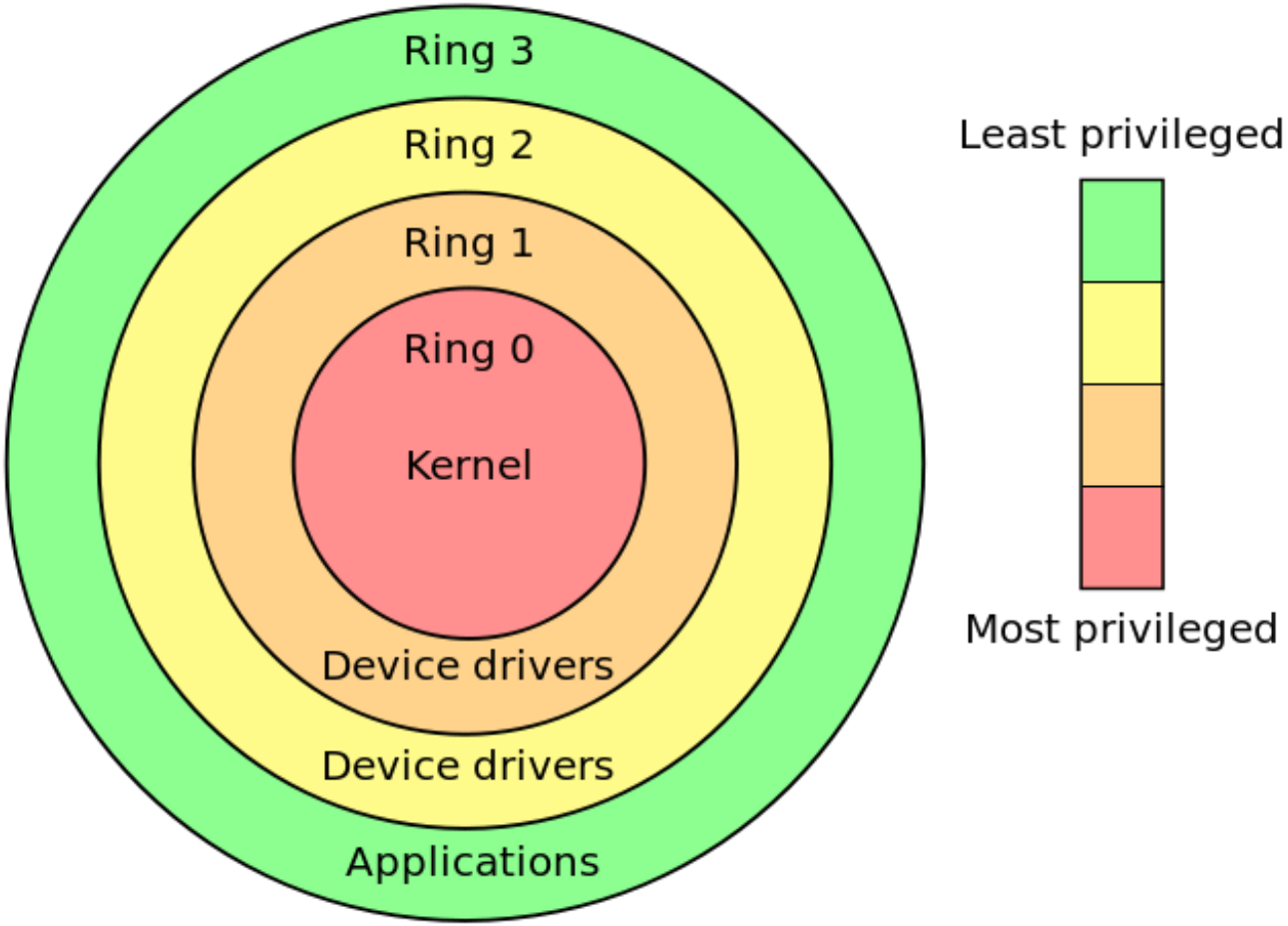
Secure Element
(SecurCore)

TrustZone en QEMU



http://www.linaro.org/blog/core-dump/arm-trustzone-qemu/

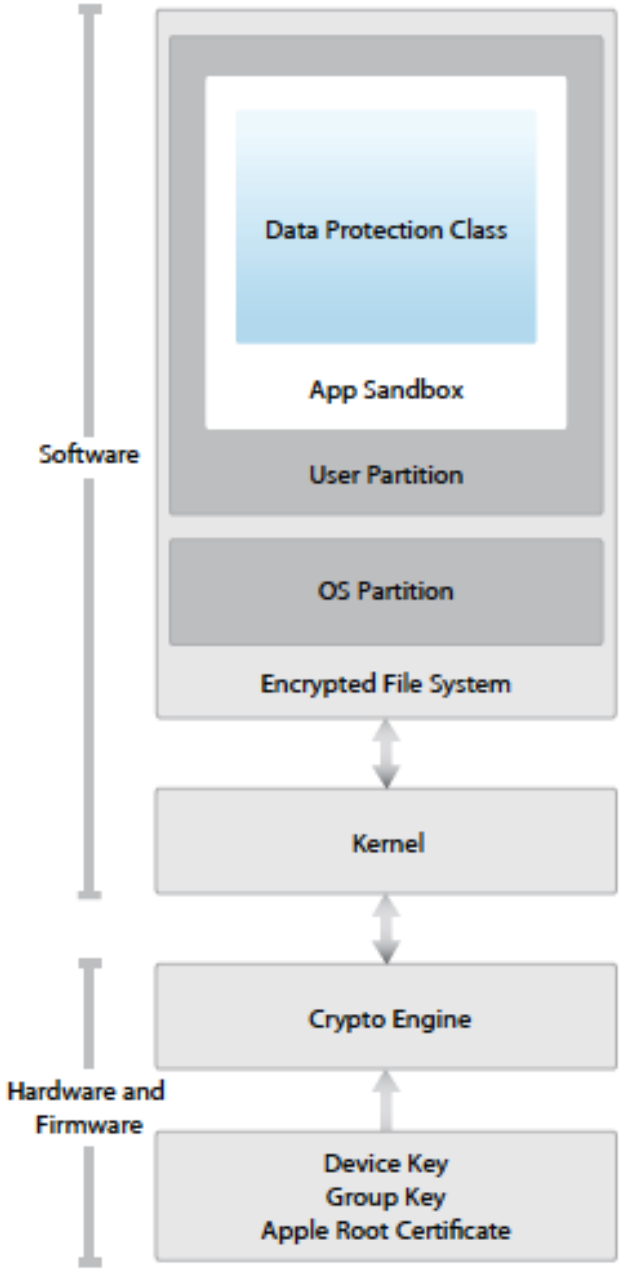
ARM vs. x86



http://duartes.org/gustavo/blog/post/cpu-rings-privilege-and-protection/

El caso de iOS

- La arquitectura de seguridad de iOS está compuesta de distintas tecnologías, incorporando muchas de las características vistas anteriormente



Perspectivas de seguridad en IOS

- Seguridad del sistema: El software y hardware seguro que forma la base de los iPhone, iPad, y iPod touch.
- Cifrado y protección de datos: La arquitectura y el diseño que protegen los datos del usuario si este se pierde o es sustraído, o si una persona no autorizada intenta usarlo o modificarlo.
- Seguridad en las Apps: El sistema que permite que las aplicaciones se ejecuten de forma segura sin comprometer la integridad de la plataforma.
- Seguridad en redes: Protocolos de comunicaciones estándar que proporcionan autenticación y confidencialidad en las transmisiones de datos.
- Servicios de Internet: La infraestructura de red de Apple para mensajería, sincronización y copias de seguridad.
- Controles del dispositivo: Métodos para prevenir el uso no autorizado del dispositivo así como el borrado remoto en caso de perdida o robo.
- Controles de privacidad: Capacidades de iOS que se pueden usar para para controlar el acceso a los servicios de localización y a los datos del usuario. https://www.apple.com/business/docs/iOS_Security_Guide.pdf

https://www.apple.com/es/business/docs/iOS_Security_Guide.pdf

Seguridad del Sistema

- La seguridad del sistema (system security) incluye el proceso de arranque, las actualizaciones de software, entre otros asuntos.
- Hay varios elementos implicados
  - Secure Boot Chain
  - System Software Authorization
  - Secure Enclave
  - Touch ID

Secure Boot Chain

- Cada paso del proceso de arranque contiene componentes firmados digitalmente por Apple.
- Cuando se arranca un dispositivo iOS, el procesador ejecuta código de la Boot ROM, que se inserta en el chip durante el proceso de fabricación
  - El código de la Boot ROM contiene la clave pública de la Autoridad de Certificación de Appl
    - Esa clave se usa para verificar la firma de Apple sobre el código
- Para el chip de telefonia y el coprocesador “Secure Enclave” se sigue in proceso de arranque similar

System Software Authorization

- iOS lo utiliza para prevenir que versiones anteriores de SW (con menor seguridad) se instalen en el dispositivo
  - partiendo de la base de que el SW actualmente instalado es seguro porque se ha seguido el proceso descrito en la transparencia anterior
- Durante el proceso de actualización de iOS, iTunes (o el propio dispositivo) se conecta al servidor de autorización de Apple
  - Le envía “huellas” criptográficas de los elementos a instalar, junto con un número aleatorio y el ID único de dispositivo (ECID).
  - El servidor comprueba que se pueden instalar esos elementos y devuelve esas mismas “huellas” junto al número aleatorio y el ECID firmados por Apple como elemento de autorización.
  - El dispositivo comprueba el la firma, el ECID y el número aleatorio antes de hacer la actualización.

Secure Enclave (procesadores >A6)

- Es un coprocesador que usa su propio secure boot independiente del procesador principal
- Proporciona las operaciones criptográficas necesarias para el proceso de administración de claves en la protección e integridad de los datos, incluso si el kernel está comprometido.
- Usa memoria cifrada e incorpora un generador de números aleatorios hardware.

- Para cifAES-256 y de SHA-1

Touch ID

- Si Touch ID está desactivado, las claves de protección de datos de IOS, que se mantienen de forma segura en el Secure Enclave, se descartan.
  - Hasta que no se introduzca la código de desbloqueo no vuelven a estar accesibles.
- Si Touch ID está activado, dichas claves se protegen con una clave que se almacena dentro del Touch ID.
  - Si el Touch ID reconoce la huella del usuario este devuelve la clave para recuperar las claves de protección de datos.
  - Si se reinicia el dispositivo, pasan más de 48 horas o se producen 5 intentos fallidos de lectura de huella, el Secure Enclave descarta las claves protegidas y pide la introducción del código de desbloquo.

Cifrado y Protección de Datos

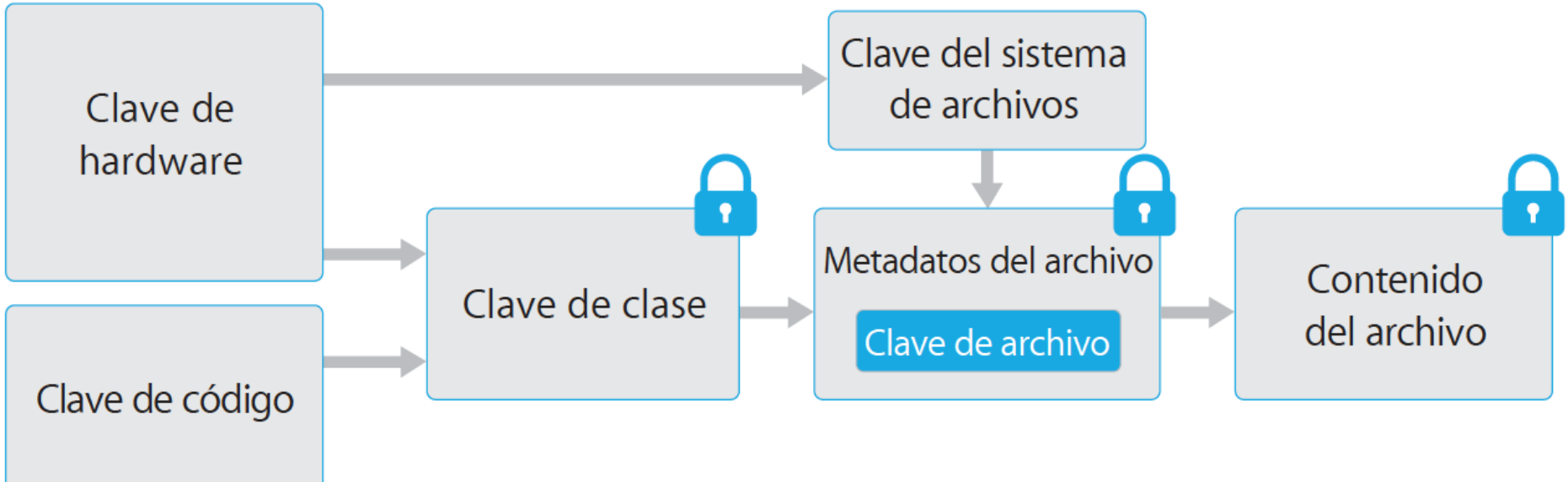
- Aparte de las medidas anteriores, iOS tiene unas capacidades adicionales de cifrado y protección de datos en caso de que se comprometa el sistema.
- En particular proporciona un Soporte Hardware para el cifrado y la gestión de claves.
- Además, las claves del sistema de almacenan de forma jerárquica para favorecer su mantenimiento.
- Los módulos criptográficos de iOS 8 se están evaluando bajo la norma FIPS 140-2 de los EE.UU.

Soporte Hardware

- Todos los dispositivos iOS tienen un coprocesador AES 256 entre el canal DMA que conecta el almacenamiento Flash y la memoria principal del sistema.
  - El objetivo es mejorar la eficiencia del cifrado, así como el consumo de energía
- Cada dispositivo dispone de dos claves AES 256 principales instaladas durante la fabricación.
  - UID (Device Unique ID). Única para cada dispositivo, no la conoce Apple ni sus suministradores, hace que los datos esté vinculados criptográficamente al dispositivo.
  - GID (Devide Group ID). Única para un grupo de dispositivos (p.e. Todos los dispositivos con A8), usada para para operaciones no críticas como actualizaciones de software.
- Salvo el UID y el GID, el resto de claves criptográficas se crean mediante un PRNG usando el esquema CTR_DRBG. La entropía se basa en mediciones de tiempo de arranque e interrupciones.
- Las claves generadas en el Secure Enclave usan un TRNG como fuente de entropía.
- iOS proporciona también mecanismos de borrado de claves bajo nivel.

Jerarquía de Claves

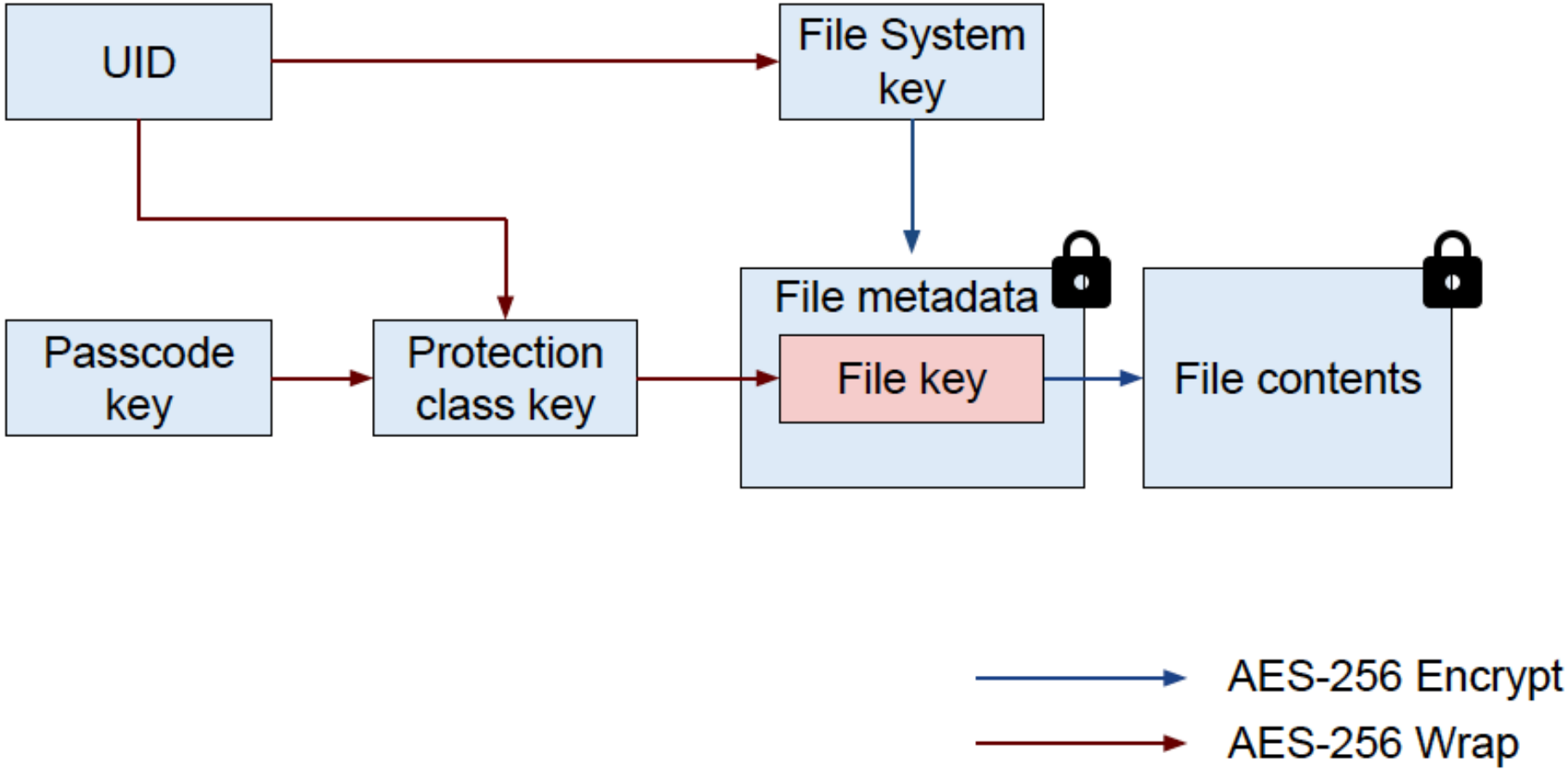
- Por cada archivo que se genera, se crea una clave (“per-file” key) que se utiliza para cifrarlo en flash usando AES CBC, el IV es una función del offset del fichero y el SHA1 de la clave.
- Es clave se protege usando diversas claves (RFC 3394) y se incluye entre los metadatos del fichero.
- La clave del sistema se almacena en Effaceable Storage, una zona dedicada de la memoria NAND usada para almacenar claves criptográficas, que se puede direccionar y borrar de forma segura.



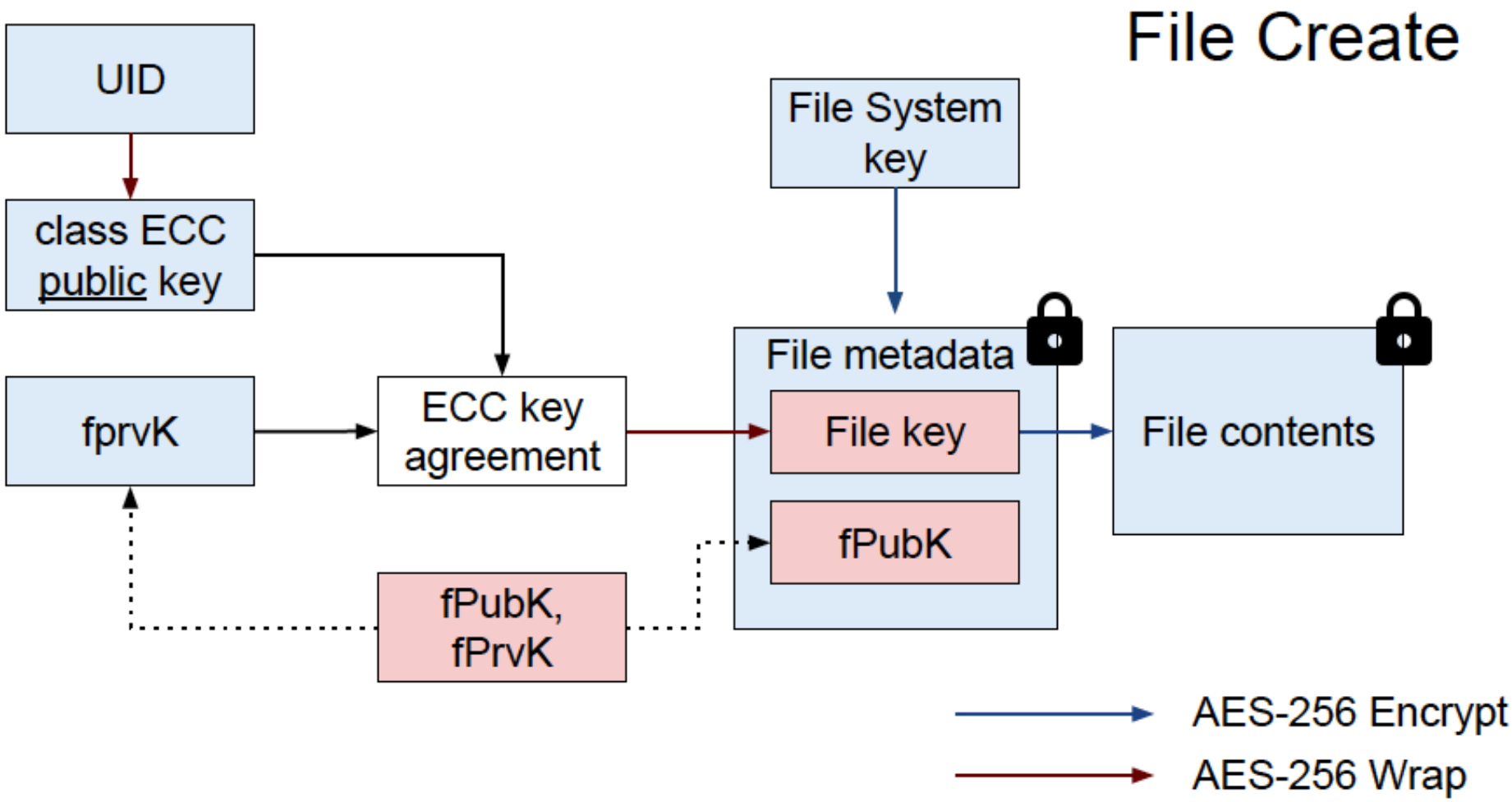
Clases de Protección de Datos

- Cuando una aplicación crea un nuevo archivo le asigna una clase.
- Cada clase usa diferentes políticas para determinar cuando se puede acceder a la información.
- Dos clases significativas son las siguientes:
  - Protección Completa. La clave de clase se deriva del código de desbloqueo y el UID y se elimina de memoria al poco tiempo de bloquearse el dispositivo.
  - Protección salvo abierto. La clave de clase es una clave privada DH y la clave de cifrado del fichero es una clave pública efímera DH.

NSFileProtectionComplete

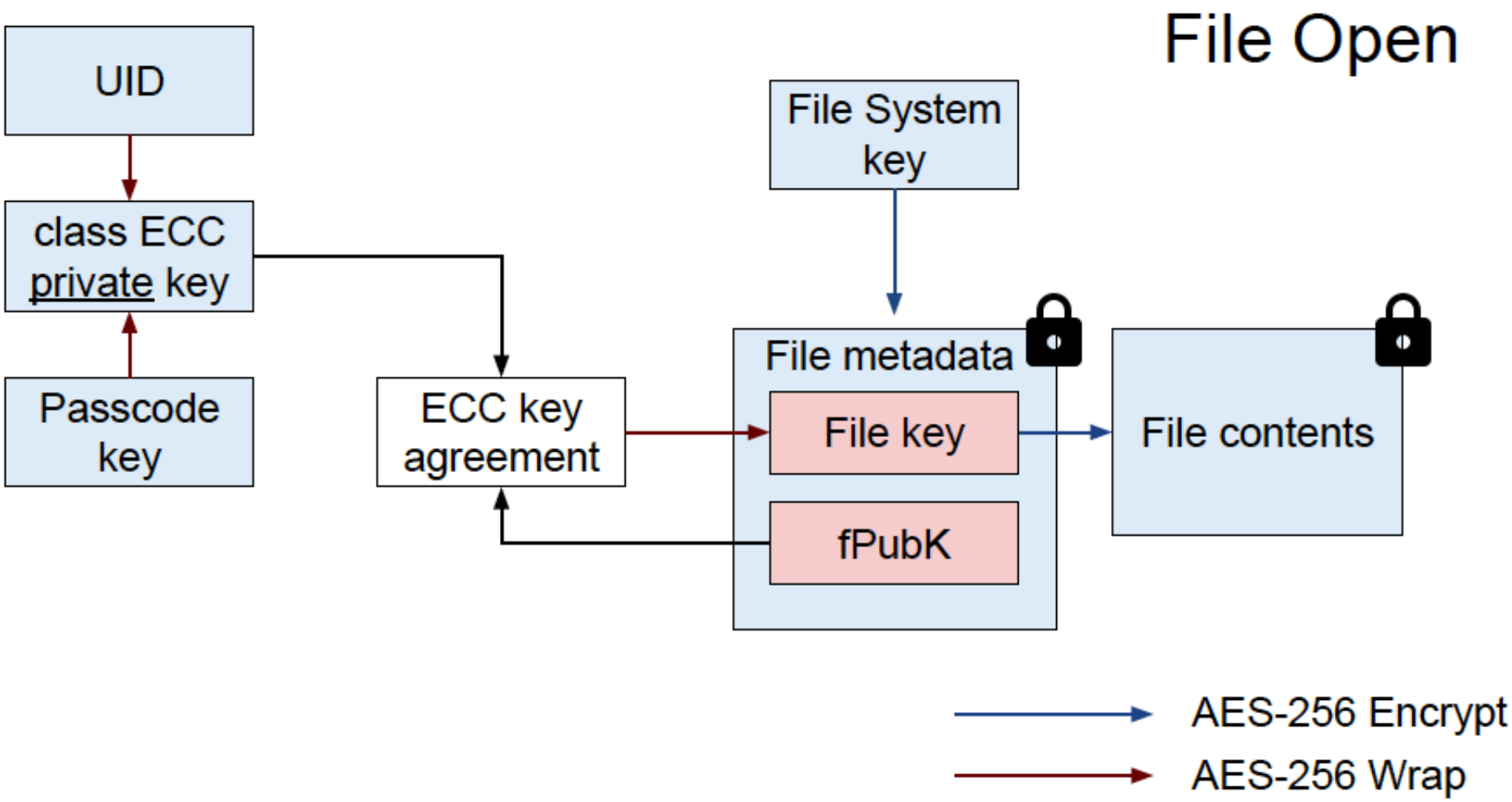


NSFileProtectionCompleteUnlessOpen



http://www.slideshare.net/DusanKlinec/ios-secure-app-development-60081091

NSFileProtectionCompleteUnlessOpen



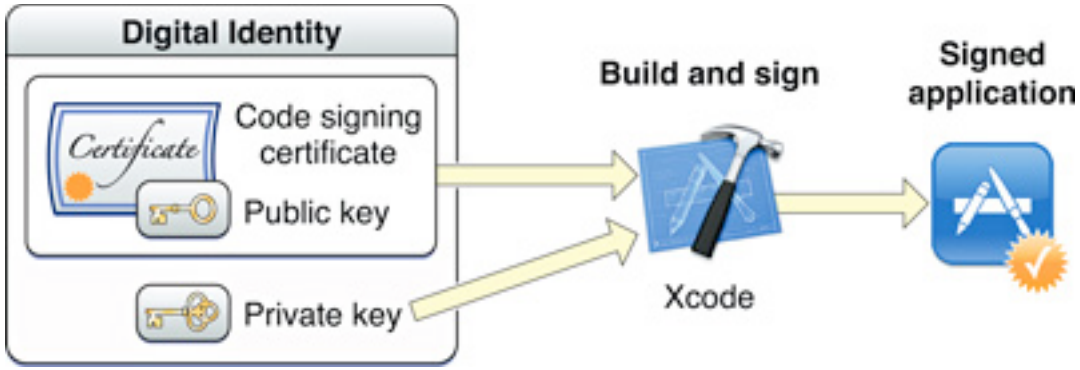
http://www.slideshare.net/DusanKlinec/ios-secure-app-development-60081091

Seguridad en las Apps

- En un nivel más alto tenemos la seguridad relativa a las aplicaciones. Tal y como vimos al principio del tema, la mayor parte de las vulnerabilidades provienen de código de terceros. Para mitigar estos problemas se definen mecanismos para,
  - firmar y verificar el código que se ejecute en los dispositivos
  - Aislar los datos de cada aplicación
  - Verificar la conexión con los accesorios.

Firma de código

- Las aplicaciones incluidas en iOS están firmadas por Apple directamente.
- Las aplicaciones de terceros están firmados por un desarrollador, que debe registrarse y conseguir un certificado firmado por Apple.
- Todas la librerías incluidas en el ejecutable deben ir firmadas, cuando se cargan en memoria se comprueba la firma para evitar la ejecución de código no verificado.

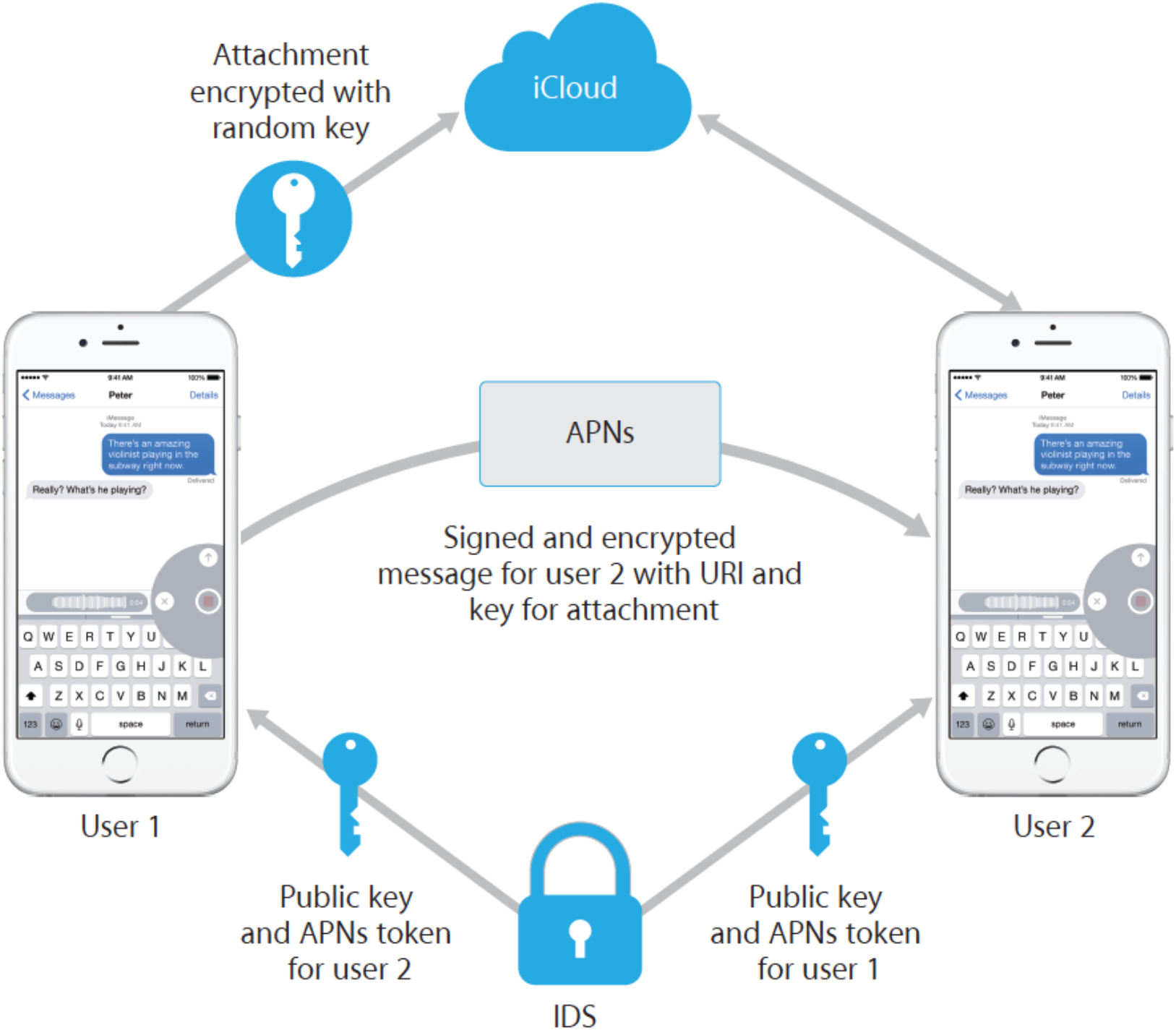


Seguridad en Redes

- iOS soporta los últimos estándares de seguridad en comunicaciones tanto en sus aplicaciones como en las APIs de desarrollo.
  - TLS 1.x, WPA2 802.1X (EAP-TLS, EAP-TTLS, …)
- Los últimos dispositivos de Apple (iPhone 5c, 5s y superiores; iPad Air y iPan mini retina) utilizan direcciones MAC aleatorias para el escaneo de redes WiFi siempre que el dispositivo no esté asociado a ninguna red wifi y esté en reposo.

Servicios de Internet

- Dentro de los diversos servicios de proporciona Apple nos vamos a centrar en iMessage y el servicio de notificaciones push (APNs).
- Apple no registra los mensajes ni los adjuntos y sus contenidos están cifrados extremo a extremo.
- Cuando se activa iMessage el dispositivo genera dos pares de claves publica y privada:
  - RSA 1280-bit para cifrado
  - ECDSA 256-bit para firma digital
- Las dos claves privadas se guardan en el dispositivo y las públicas se envían al directorio de Apple (IDS) donde se asocian con el número de teléfono o e-mail y las direcciones/token APNs.
- Cuando se quiere mandar un mensaje se recupera el token APNs del IDS.
- El APNs se encarga de enviar el mensaje firmado y cifrado de un dispositivo a otro.


Attachment encrypted with random key
iCloud
APNs
Signed and encrypted message for user 2 with URI and key for attachment
User 1
User 2
Public key and APNs token for user 2
Public key and APNs token for user 1
IDS

SEGURIDAD EN ENTORNOS VIRTUALIZADOS

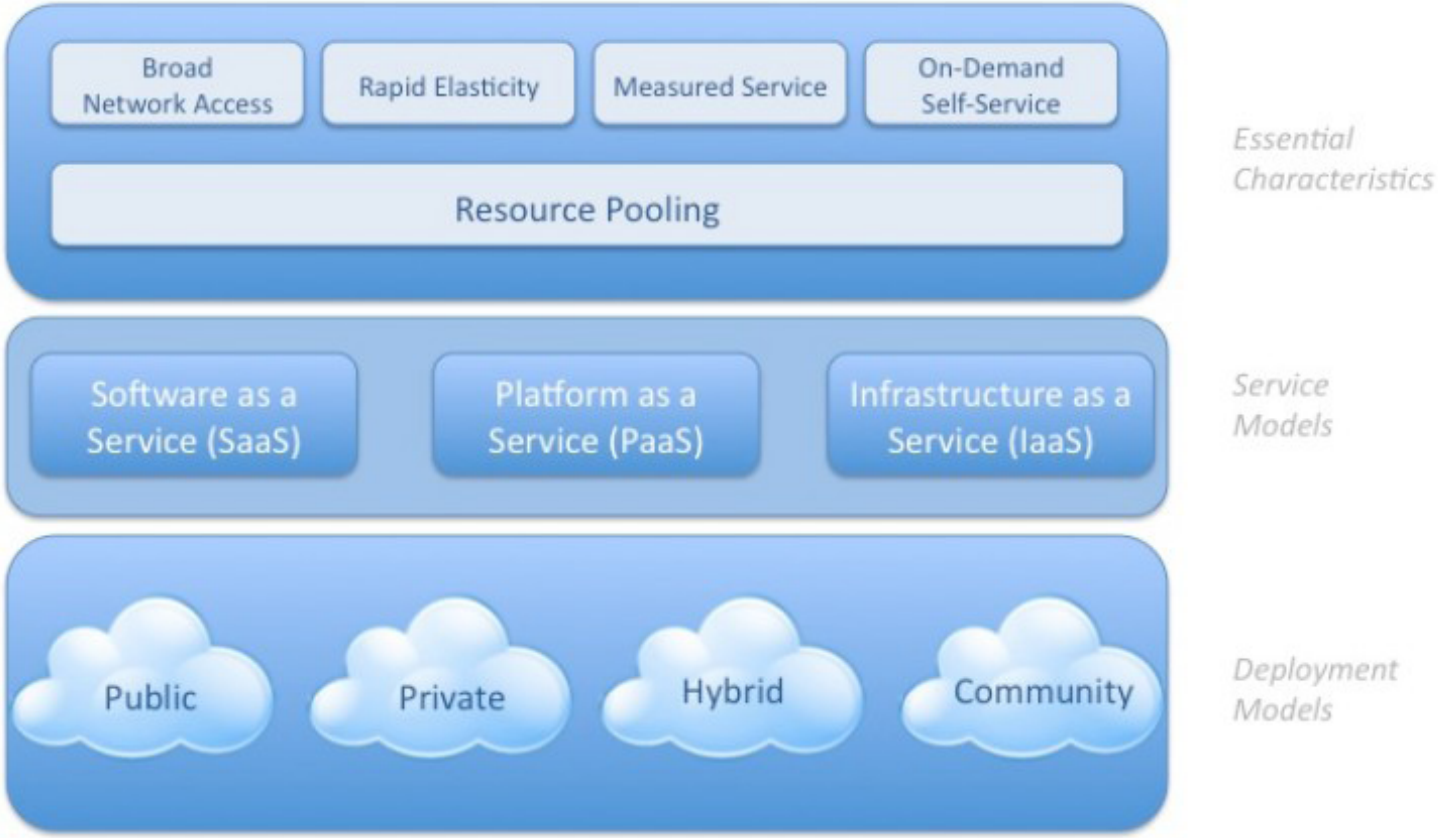
Concepto de cloud computing

- Cloud Computing es el conjunto de tecnologías que permiten, a través de la red, habilitar un conjunto de recursos TIC compartidos (servidores, almacenamiento, aplicaciones, etc) que presten servicio rápidamente, bajo demanda y con una mínima intervención por parte del proveedor.

- Hablar de cloud computing es hablar de disponibilidad y flexibilidad de recursos TIC.

Modelo del NIST de Cloud Computing

Visual Model Of NIST Working Definition Of Cloud Computing
http://www.csrc.nist.gov/groups/SNS/cloud-computing/index.html



Características esenciales

- Autoservicio bajo demanda
  - El cliente del servicio determina qué recursos y cuándo los necesita.
  - Aprovisionamiento automatizado.
- Acceso a través de la red desde múltiples dispositivos.
- Conjunto de recursos compartidos
  - El conjunto de recursos es compartido por distintos clientes.
  - El conjunto está compuesto de recurso físicos y virtuales.
  - Es distribuido y no tiene una localización física única (disponibilidad).
  - Infraestructuras descomunales.

Características esenciales (II)

- Elasticidad y rapidez
  - El cliente tiene la ilusión de disponer de recursos infinitos.
  - La respuesta a la petición aprovisionamiento es inmediata; pero la provisión no tiene por qué.

- Servicio monitorizado
  - Para satisfacer las condiciones de servicio.
  - Sistemas autónomos.
  - Posibilidad de pago por uso.

Niveles de servicio - IaaS

- Infraestructura como Servicio (IaaS, de sus siglas en Inglés Infraestructure as a Service)
  - Se trata del nivel más cercano al Hardware
  - Se encarga de entregar una infraestructura de procesamiento completa al usuario bajo demanda
  - El usuario dispone de una o varias máquinas virtuales en la nube con las que, por ejemplo, puede aumentar el tamaño de disco duro en unos minutos, obtener mayor capacidad de proceso o enrutadores y pagar solamente por los recursos utilizados
  - Este nivel puede ser visto como una evolución de los Servidores Privados Virtuales que ofrecen actualmente algunas empresas de Hosting, estamos en el nivel más cercano al Hardware.

Niveles de servicio - PaaS

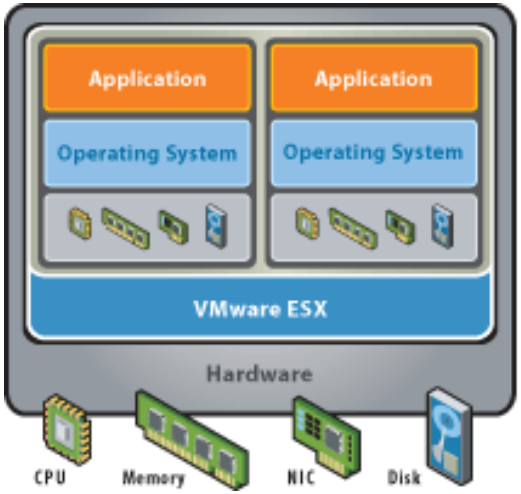
- Plataforma como Servicio (PaaS, de sus siglas en Inglés Platform as a Service)
  - Se trata del nivel intermedio
  - Se encarga de entregar una plataforma de procesamiento completa al usuario, plenamente funcional y sin tener que comprar y mantener el hardware y software
  - Por ejemplo, un desarrollador web necesita un servidor web que sirva sus páginas, un servidor de bases de datos y un sistema operativo
  - Este nivel se encarga de proporciona esos servicios

Niveles de servicio - SaaS

- Software como Servicio (SaaS, de sus siglas en Inglés Software as a Service)
  - Se trata del nivel más bajo
  - Se encarga de entregar el software a través de internet siempre que lo demande el usuario.
  - Por ejemplo una empresa necesita un servio online de edición de documentos como Google Docs o un servicio de correo electrónico
  - Este nivel se encarga de proporciona esos servicios

Conceptos Relacionados - Virtualización

- Un servidor con virtualización no es un Cloud privado.
- La virtualización es un componente del Cloud Computing.
- No hay aprovisionamiento inmediato y automatizado.
- No es distribuido y existen puntos únicos de fallo que no garantizan la disponibilidad.



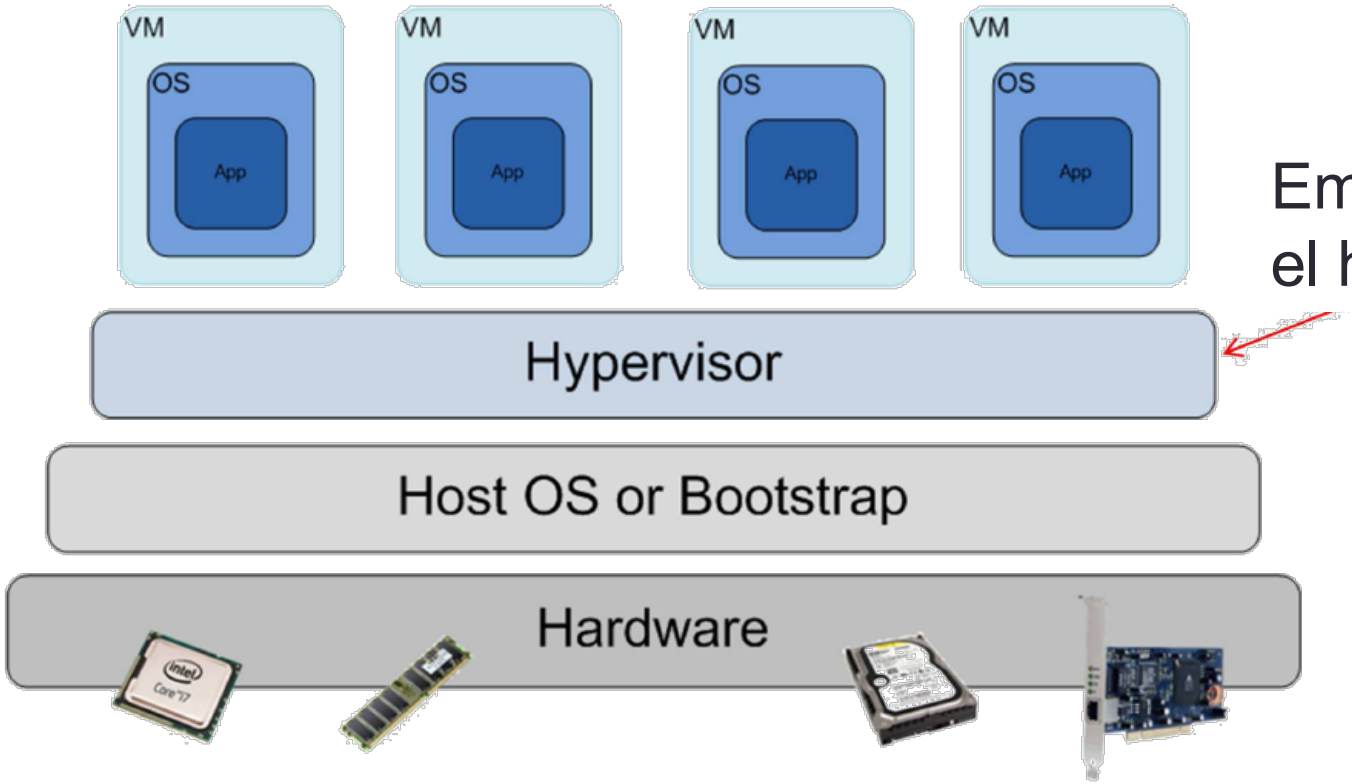
Virtualización

- Software usado para crear una plataforma hardware simulada
  - Simula video, red, discos duros, CPU, etc.
  - Se pueden crear múltiples copias
  - Está gestionado por el Hipervisor

- Permite desacoplar el hardware del software
  - Proporciona una capa de abstracción
  - Proporciona mayor flexibilidad de uso

Virtualización - Hipervisor

- El Hipervisor es el encargado de emular las configuraciones especificas de hardware para los sistemas operativos “huéspedes”.



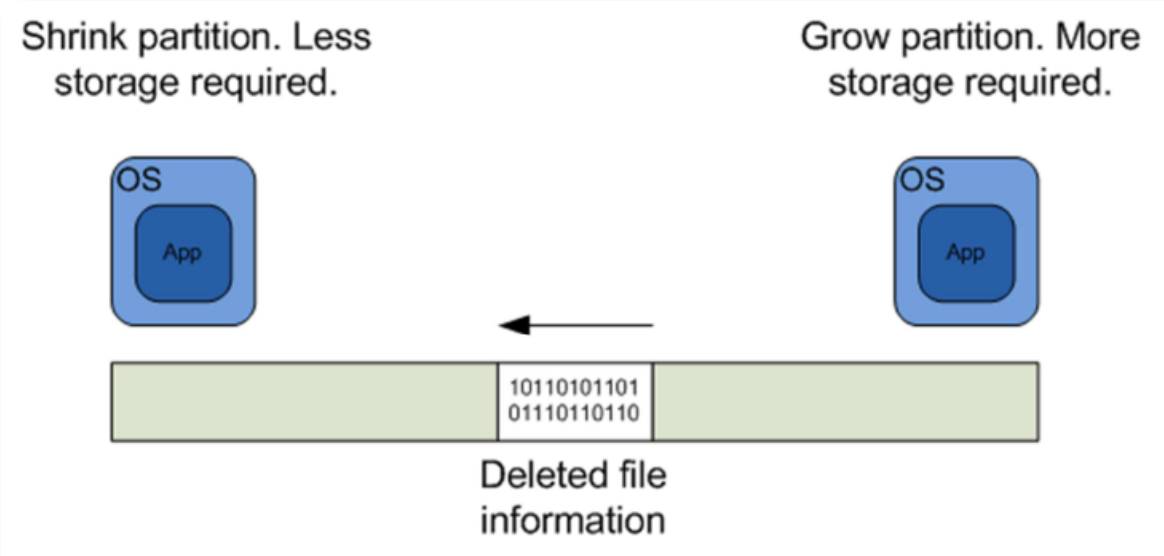
Emula todo el hardware

Virtualización - Configuraciones

- Sobre Sistema operativo o “Hosted”
  - Arranca desde el sistema operativo
  - El hipervisor se carga como una aplicación o servicio
  - Proporciona una mayor flexibilidad
  - Hereda los riesgos de seguridad del SO subyacente
  - Ejemplos: VirtualBox, VMware Workstation/Server/Player, QEMU, Microsoft Virtual PC
- Nativo o “Bare Metal”
  - Arranca directamente el hipervisor sobre un SO especifico minimalista
  - Normalmente es un derivado de Linux o BSD
  - Más complejo de actualizar
  - Elimina los riesgos inherentes de los SO tradicionales.
  - Proporciona mayor eficiencia
  - Ejemplos: VMware ESXi/ESX, Xen, Citrix XenServer, Microsoft Hyper-V Server.

Aislamiento

- VMs y virtualización de DMA
- Corpatición de recursos y almacenamiento
- Guest tools (cortapapeles, copia de ficheros entre host y guest, ect.)



Virtualización - Introspección

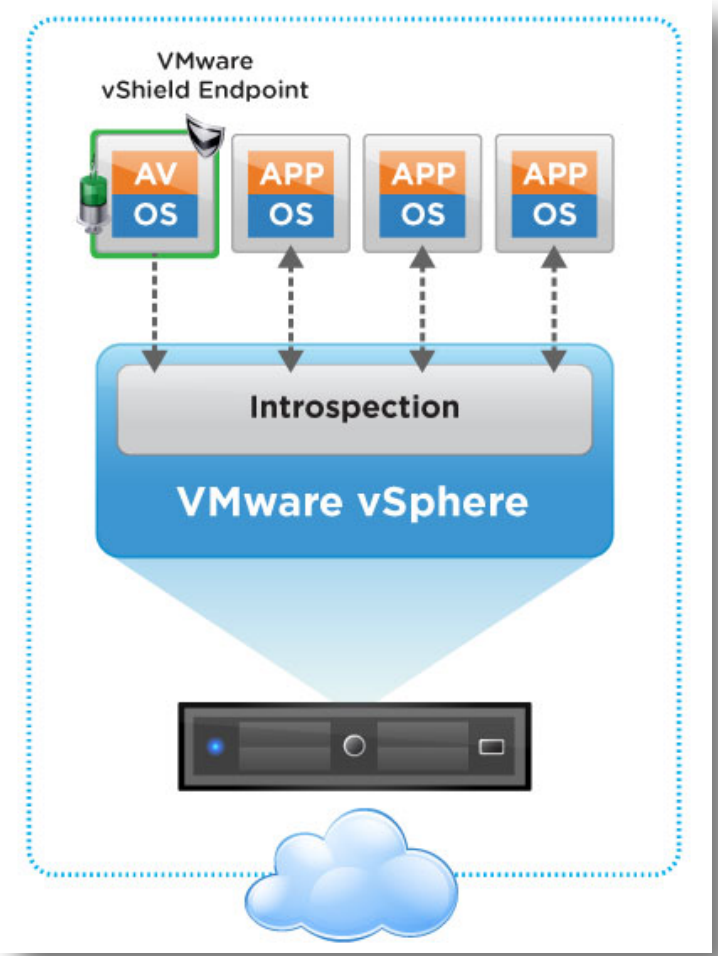
- Extiende las capacidades del Hipervisor
  - Está un nivel por encima de la simple monitorización de llamadas al sistema
- La Introspección permite monitorizar
  - La memoria y la ejecución del programa
  - Acceso a los ficheros almacenados en las maquinas virtuales
  - Tráfico de red
- Es una vía para un análisis más profundo de las maquinas virtuales.
  - Este análisis puede usarse para proporcionar características avanzadas de seguridad.

Virtualización – Introspección (II)

- El hipervisor se ejecuta con más privilegios que las VM
- Esto significa que dentro de la VM, el hipervisor lo puede controlar todo.
- Rootkit a nivel del kernel
  - Normalmente se pueden hacer invisibles para las herramientas de seguridad
  - Pero no pueden ocultar nada al hipervisor
  - La introspección permite validar la ejecución de código por tanto permite protegerse ante este tipo de amenzas.
- El código es más seguro cuanto menor y más eficiente es, pero la introspección añade líneas de código que pueden contener errores o vulnerabilidades.

Virtualización – Introspección y Seguridad

- Control de Software Malicioso y Detección de Intrusiones
  - La utilización de un Antivirus a nivel de Hipervisor permite que mediante introspección se puedan analizar todos los discos de todos las VM, de esta forma es más fácil encontrar patrones de infección y optimizar los análisis. Lo mismo ocurre con los IDS/IPS.
- Prevención de perdida de información
  - La introspección permite proteger ciertas porciones del disco evitando accesos destructivos por encima de las políticas de los sistemas operativos
- Cortafuegos
  - Entre VM, de forma que la comunicación entre VM se puede inspeccionar en el switch virtual para aplicar reglas con mayor prioridad que las de los SO de las VM
  - Entre VM
- Técnicas forenses

Antivirus en Entornos Virtualizados

- AV Storms
  - El análisis de virus programado en múltiples VM al mismo tiempo puede consumir los recursos del hardware subyacente.
- Se puede usar un solo AV por máquina física
  - Cada VM está conectada al AV mediante Introspección
- Ventajas
  - Integración de políticas de seguridad
  - Eficiencia
  - Fácil gestión
  - …