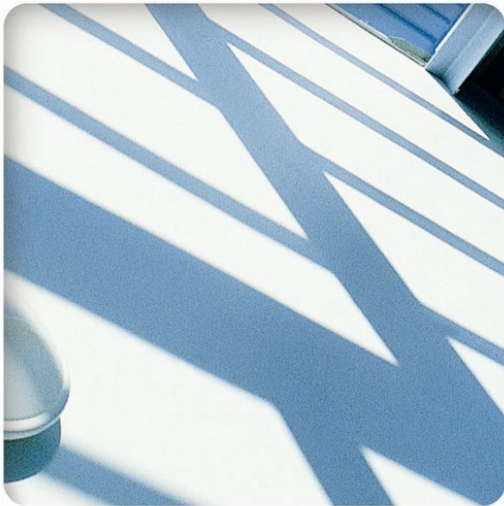




# Symbian OS v9 and Platform Security



**Mark Shackman**

Developer Consultant  
Symbian Developer Network

# Agenda

- Symbian OS v9 Overview
- EPOC Kernel Architecture 2
- New Tool Chain for Symbian OS v9
- Platform Security

# Symbian OS v9 – Overview

# Symbian OS v9 objectives

- Adoption in mid-tier, higher volume phones
  - ... Decrease build costs of devices
  - ... Enable shipping into the mass market
- Meet needs of industry stakeholders
  - ... Symbian OS licensees and network operators
  - ... Symbian partners, application developers and content providers
- Strategic market context:
  - ... Secure, robust & efficient phones
  - ... Secure deployment of DRM content and m-commerce
  - ... Platform for mass-market revenue-earning services

# Symbian OS v9.1 and Series 60 3<sup>rd</sup> Edition

Series 60 Core Applications

Developer Applications

Series 60 3<sup>rd</sup> Edition

Symbian OS v9.1

# Symbian OS v9: Key Features

- Hard Real Time Kernel
  - ...EPOC Kernel Architecture 2 (EKA2)
- Move to ARM's Application Binary Interface (ABI) binary standard
  - ...take advantage of latest ARM architecture
    - Improved performance
- Evolution of Platform Security
  - ...Platform Security evolves from perimeter security model
  - ...facilitates network operator deployment of m-commerce
  - ...protects networks, handsets, user data from malware
  - ...strengthens network operator confidence in open phones

# Symbian OS v9

- Major Binary Compatibility break
  - ...re-tooling to ARM's Application Binary Interface (ABI)
  - ...Platform Security
- Some Source Compatibility breaks
  - ...new Kernel
  - ...Platform Security
- Incorporate major changes in one major release
- Ensure stability going forward

# Symbian OS v9 changes for Developers

- **EKA2 architectural enhancements**  
... kernel enhancements and new IPC “publish & subscribe” mechanism
- **ISO C++ enablement**  
... moving towards ISO Standard C++
- **Updated tools**  
... RVCT, CodeWarrior and Eclipse changes
- **Capability-based platform security model**  
... detailed discussion of PlatSec

# EPOC Kernel Architecture 2 (EKA2)

# EPOC Kernel Architecture 2 (EKA2)

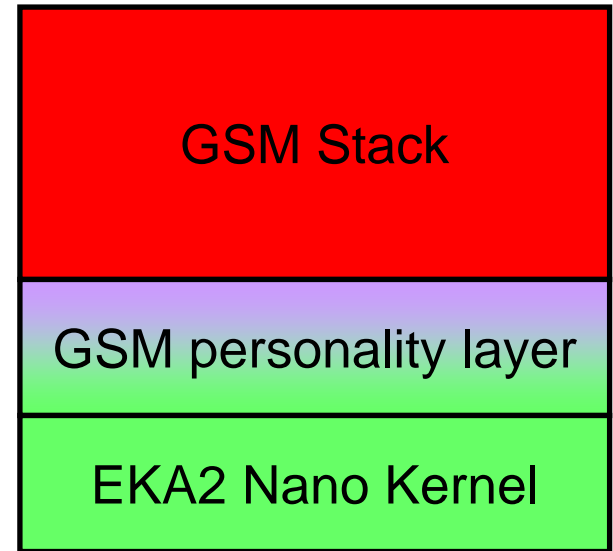
- Enable single core architecture
  - ... instead of separate baseband and application processors
  - ... through real time kernel
  - ... reduce phone manufacturing costs
- Tidy up kernel
  - ... simply base porting
  - ... simplify device driver writing
  - ... improve Emulator
- Increase robustness
  - ... in face of malware for example

# Architectural Evolution – EKA2

- New multi-threaded, pre-emptible Real Time Kernel
- A Symbian OS personality on top of a Nano kernel – **many** personalities can co-exist
- System calls are all pre-emptible; use dual stacks
- Deterministic ISR, thread response, latencies etc
- More IPC mechanisms such as message queues, publish-and-subscribe, shared I/O buffers

# Nano Kernel and personalities

- EKA2 splits the kernel in two layers: Nano Kernel & Symbian OS Kernel
- Nano Kernel is responsible for the very basic synchronisation, timing, initial interrupt handling and scheduling services
  - ... Just enough functionality to run GSM signalling stack
- Many “personalities” simultaneously run on top of Nano Kernel
  - ... Each can be run pre-emptively
- Symbian OS kernel is a personality
- GSM stack is a personality



# Kernel pre-emptibility

- The EKA2 Symbian OS Kernel is multi-threaded
  - ... device drivers are much easier to write
- It is completely pre-emptible
  - ... even the memory allocations and context switch can be pre-empted
- User side threads have a user mode and supervisor mode stack
  - ... executive calls run on user thread's supervisor stack
  - ... executive calls can thus all be pre-empted

# EKA2 emulator

- The EKA2 emulator does not rely on the host OS for the thread scheduling
  - ... Win32 thread for each Symbian OS thread
  - ... “freezes” all threads and uses it’s CPU time to schedule (in terms of priorities) exactly as the Nano Kernel scheduler would
  - ... debugging is so much better this way, since the relevant priorities will always be correct !!
- It has been redesigned to be easily portable to other host OSes

# More IPCs

EKA2 has seen the addition of three new Inter Process Communication mechanisms:

- ✓ Publish & Subscribe
- ✓ Message Queues
- ✓ Shared Buffer I/O – between driver and user space

EKA1's only IPC mechanism was Client-Server

...some back-porting of these in the latest v7.0s and v8.x products. See [www.symbian.com/developer/techlib/papers/cpp\\_sysarch.asp#ipc\\_mech](http://www.symbian.com/developer/techlib/papers/cpp_sysarch.asp#ipc_mech) and in v8.0 Developer Library

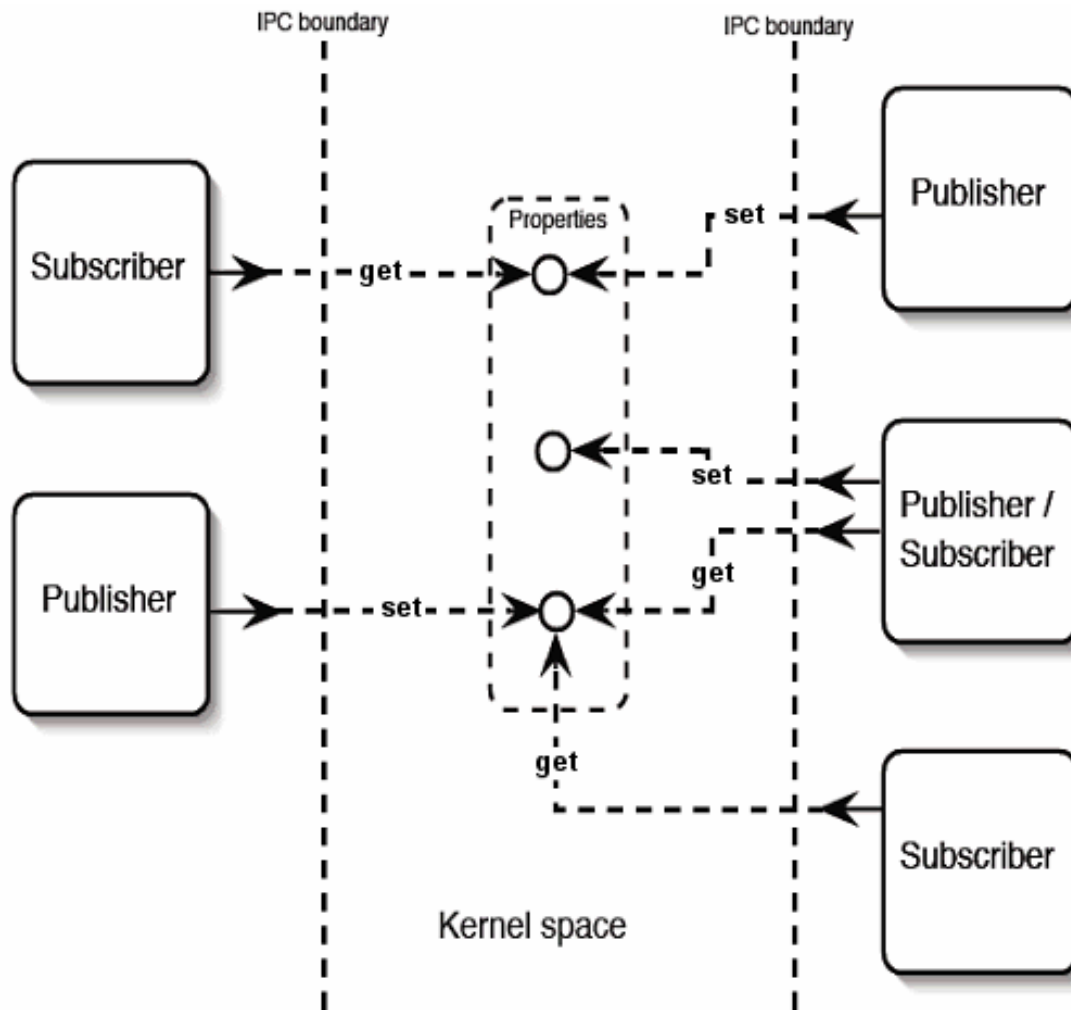
# Publish & Subscribe (a.k.a. Properties)

- Define and publish system-wide properties
- Both user and kernel side (via similar APIs)
- Either publisher or subscriber may define a property
- Connection-less paradigm (between P – S)
- Publishers and subscribers don't need to know about each other or link to special client APIs etc.
- Properties are communicated to *many peers* asynchronously

# Property characteristics

- Properties are single data values, uniquely identified by an integral key
- Properties have *identity* and *type*
  - ... The identity and type are the only things that need to be shared between publisher(s) and subscriber(s)
- Definition and deletion coupled in the same thread
  - ... Properties persist until reboot or they are deleted
- Properties are read and written atomically

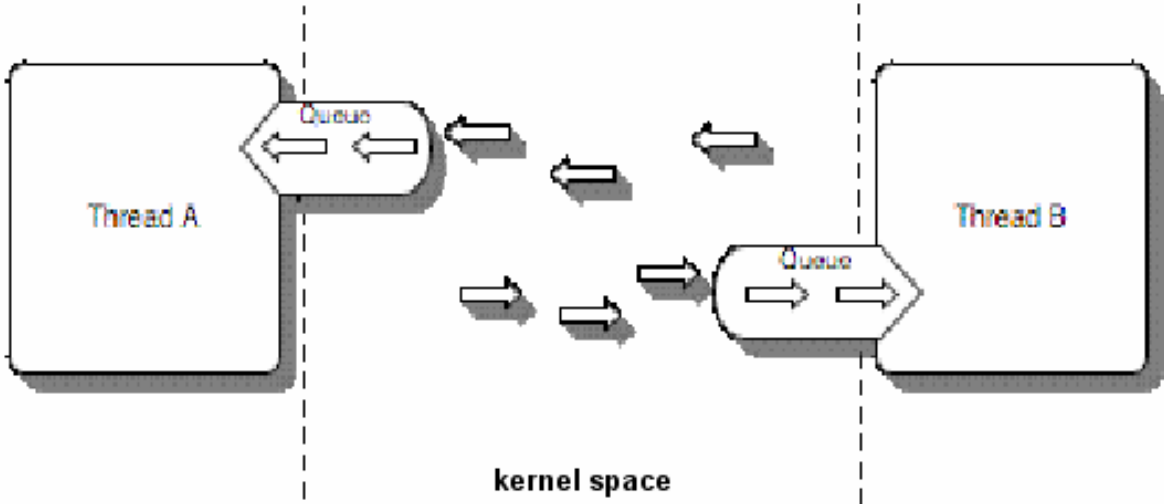
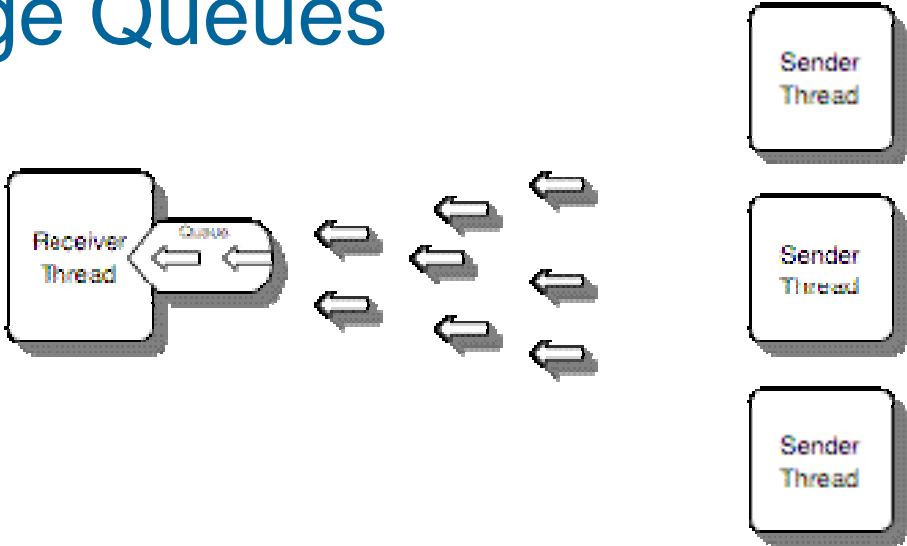
# Publish & Subscribe



# Message Queues

- Peer-to-peer
- Many-to-many
- Fire-and-forget communication semantics
- Guaranteed delivery of messages to queues
  - ... but final delivery to reader isn't
- Queues are dimensioned at the point of creation, messages are short and fixed size
- Lowest overhead IPC in EKA2 - Symbian OS v9

# Message Queues



# Moving closer to ISO C++

- TRAP and User::Leave() implemented internally in terms of catch and throw
- Ported 3<sup>rd</sup> party code can use standard C++ exception mechanisms : try/catch/throw
  - ... But they cannot mix these with Symbian OS system APIs
  - ... Leaving functions must not throw, unless they also catch internally
- Standard C++ exception specifications are supported
- Writeable static data for DLLs is finally here
  - ... Eases porting applications to Symbian OS
  - ... Use with caution as large overhead

# Symbian OS v9 – New tool chain

# Symbian OS v9 tools

- Performance optimisations available in latest ARM v5 / v6
  - giving licensees ability to exploit ARM processors
  - performance gains, hardware choice
- Supporting ABI standard for ARM architecture
  - enabling competitive market for compilers for Symbian OS
  - optimized specifically to get maximum performance from current and future ARM architectures
- Free entry-level offerings
  - GCC-E compiler
  - Eclipse IDE

# Symbian OS v9 – compilers

- ARM RVCT 2.2 Compiler
  - ... Takes advantage of ARMv5, ARMv6 architectures & beyond (no BC breaks on ARM roadmap)
  - ... Optimisations reduce ROM size (~5-10%) and increase performance of devices – especially games & multimedia
  - ... ABI standard for ARM architecture
- CodeWarrior for Symbian OS 3.0
  - ... Integrates IDE, emulator, RVCT compiler, EABI debug
- GCC-E
  - ... Free compiler, available now
  - ... EABI standard, based on GCC 3.4

# Symbian OS v9 – IDEs

- CodeWarrior for Symbian OS v3.0
  - ... Integrates
    - IDE
    - Emulator
    - RVCT compiler
    - EABI debug
- Eclipse IDE
  - ... Free download from H2 2005
  - ... Entry level IDE
    - initially targets Symbian OS v9
  - ... Build/debug on Windows host
    - build only for ARM target

# Symbian OS v9 – Platform Security

# Platform Security Agenda

- Existing security pre-Symbian OS v9
- What is Symbian OS v9 Platform Security?
- New concepts for Platform Security
- Capabilities and practicalities

# Pre Symbian OS v9 – Perimeter Security

- Security check at install time
  - ... Checks origin of application →
  - ... Full access to files and APIs once installed
- Symbian Signed for SIS files:
  - ... C++ binaries
  - ... Multimedia content
  - ... VB applications
  - ... Themes
- Java MIDlets covered separately
  - ... “Java Verified” scheme



# What is Symbian OS v9 Platform Security ?

It is a fine-grained way to efficiently restrict or completely prevent unauthorised access to sensitive APIs and data on the mobile phone while keeping the device open to developers.

- ✓ It follows a per-process capability-based model
- ✓ It compartmentalises the system, according to access capabilities, to APIs and files
- ✓ It makes sure that the users can make policy decisions they understand
- ✓ It is Kernel mediated but server enforced

# Why a finer-grained Platform Security model ?

- Phones are open, networked & data communication devices
- Users expect their phones to be highly reliable
- Users care about their privacy – and their phone bills
- Mobile networks are not like the internet – they can restrict access
- Existing “Perimeter Security” model enables unrestricted access to all phone capabilities once installed

# Platform Security – user centric view

PlatSec means for users that:

- they have
  - ... no unexpected items in their phone bill
  - ... their phone working when needed
  - ... no virus
  - ... their private data staying private
- they do not have
  - ... to take security decisions they do not understand
  - ... to take security decisions too often

# When we talk about Platform Security...

- it is about
  - ... protecting phone integrity
  - ... protecting sensitive data
  - ... controlling access to sensitive operations
- it is not about
  - ... encrypting data
  - ... scanning for viruses
  - ... managing public key infrastructure

# Scope of Platform Security

- Includes:
  - ... Symbian OS & device drivers
  - ... User interface
  - ... Applications
- Excludes:
  - ... Hardware
  - ... Network infrastructure
  - ... Remote servers

# Benefits of Platform Security

- For developers
  - ... Maintains network operator & user confidence in open phone environment
  - ... Grows opportunity for mass market applications, content & services
  - ... enables m-commerce applications & high value DRM content
- For network operators
  - ... Protects network & handsets from malware
  - ... Protects customer data & privacy

# New Symbian OS Concepts

# New Symbian OS Concept – Capabilities

A capability is a statement of trust

- Every executable is tagged at build time with some capabilities, this applies for both EXEs and DLLs
- At run time, every process is granted a set of capabilities
- Capabilities are assigned based on which APIs a process needs and therefore is authorised to use
- Capabilities of a process never change
- Capabilities, and policing of, is transparent to API users

# New Symbian OS concept – Data Caging

- Separating code from data
- File-system structure changes
  - ... \sys, \resource, \private\<process specific>, \<other>
  - ... Executables will be placed in and only run from \sys\bin
- Processes are confined to their own part of the file-system
- Access rules based on directory path
  - ... Single user, no access control list required
  - ... No extra storage needed
- Support for removable media file systems
  - ... tamper evidence for binaries

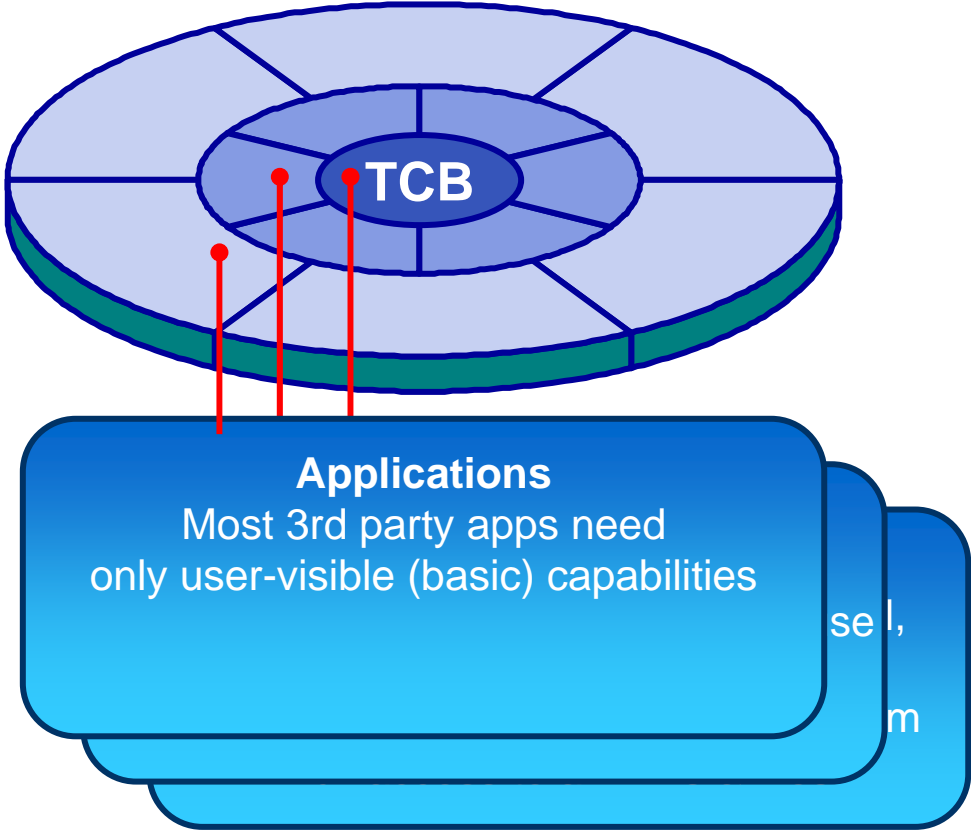
# New Symbian OS Concept – Process Identification

- Each executable now contains a Secure ID (SID)
- Secure IDs are guaranteed to be locally unique
  - ... Hence `\private\<<Secure_ID>\`
- SIDs will come from the lower part of the UID range
- SID is specified by the `SECUREID` keyword in an MMP file
  - ... If not given, UID3 is used, otherwise KNullID

# New Symbian OS Concept - Trusted Computing

- Trusted Computing Base (TCB)
  - ... New Kernel, EKA2
  - ... New Software Install
  - ... File server & Loader
- Trusted Computing Environment (TCE)
  - ... All important system servers (e.g. ETel, ESock, WServ etc)

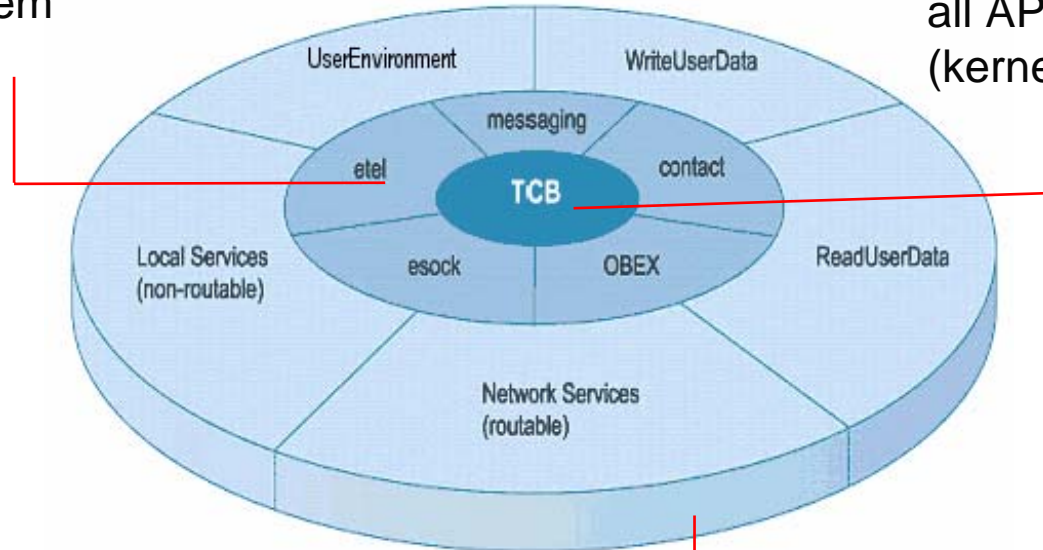
# Capabilities Model enables Compartmentalisation



# Capabilities & Trusted Computing Platform

Trusted Computing Environment:  
Servers run at different restricted system privileges.

Trusted Computing Base:  
Runs at full file system - permission to modify executables. Full access to all APIs and files (kernel, s/w install, F32)



User can grant these capabilities at install time OR applications can be signed for them.

# Capability assignment - syntax

- Capabilities are defined in MMP files

```
// program123.mmp
TARGET                program123.exe
TARGETTYPE            exe
UID                   0x00000000 0x00000123
SOURCEPATH            ..\mysource
SOURCE                myfile.cpp
USERINCLUDE           ..\include
SYSTEMINCLUDE         \epoc32\include
...
CAPABILITY         ReadUserData WriteUserData
```

- Applications requesting capabilities must generally be signed



# Signed and Unsigned Capabilities

	One-shot	Blanket
Unsigned - Sandboxed	NetworkServices Location	LocalServices UserEnvironment
Signed	Premium Billable events. (Not capability related)	LocalServices UserEnvironment NetworkServices Location ReadUserData WriteUserData ReadDeviceData WriteDeviceData SWEvent ProtSrv PowerMgmt SurroundingsDD

# Policing Capabilities

When calling a capability-protected API, verification of the capability may be dependent on the arguments passed to the API:

For example, a call to CFi I eMan: : Copy():

- **is checked** for **AllFiles** if it is accessing a protected folder (e.g. \private\- is **not checked** for any capabilities if it is accessing the application's protected data directory (in \private\

Therefore, most applications will probably not require AllFiles capability.

Details of possible capability requirements are in the DevKit under:

Symbian OS -> Symbian OS Guide -> Platform Security -> Capability report

# Capabilities at application load time

- Rule 1: The capabilities of a process never change
  - ... No way to add or remove capabilities to a process
  - ... Loading a DLL never changes the process' capabilities

# Capabilities at DLL load time

- Rule 2: A process cannot load a DLL with less capabilities than itself
  - ... DLL capabilities do *only* reflect a level of trust
  - ... DLL capabilities do not authorise anything
  - ... DLL code runs at process' capabilities level
  - ... DLL can have more capabilities than process

# Implications for static interface DLLs

- Shared libraries that export a static interface will need to have capabilities such that all its users may load them
- This means that even a simple DLL that does for example some signal processing calculations will need to have capabilities such that a telephony application may use it.
- A DLL that is loaded by another DLL will need to have the same or greater capabilities as the calling executable

# Implications for Plug-in DLLs

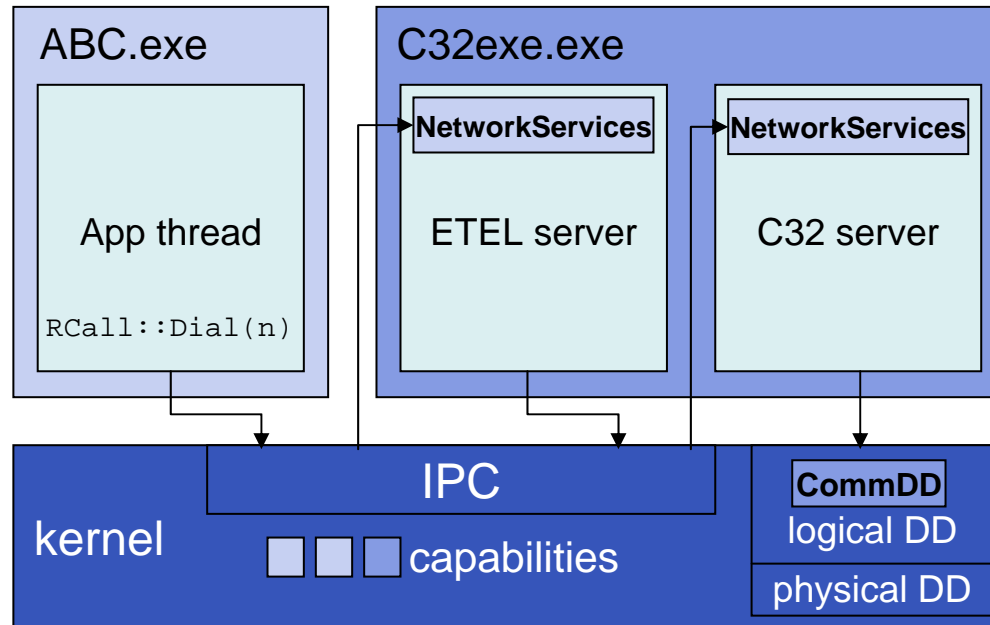
- Plug-in DLLs limited to what the host process can do
  - ... Implementers do not have to implement capability checking
- Plug-in DLLs as trusted as the host process
  - ... Recognisers, same trust level as Apparac server
  - ... MTMs same trust level as Messaging server

# Implications for Plug-in Frameworks

- Polymorphic DLLs as plug-ins to be avoided
- Frameworks used to search a directory (or directories) to find polymorphic DLLs.
- With Platform Security, all binaries are located in `\sys\bin\`
  - ... only processes with the AllFiles capability are able to read from that location.
- This means that most processes are not able to scan for binaries themselves
- To avoid, replace with ECOM plug-in

# Capabilities at run time – calls to a server API

- Capabilities protect APIs
  - Capabilities checked when crossing the process boundary
  - Calling process **must** have capability required to call server API
- ... ABC.exe must have NetworkServices capability for example



# Implications for servers

- Servers will need to police access to their resources accordingly (use of CPolicyServer)
- Policing must occur at IPC boundaries
- Servers which are trusted by the TCE and others, should be careful not to 'leak' such trust
- Changes necessary to secure the Client Server Architecture for v9:
  - ... Server Interface
    - Replace CServer with CServer2
    - Replace CShareableSession with CSession2
    - Replace RMessage with RMessage2 derives from RMessagePtr2
  - ... Client Interface
    - RSessionBase / RSubSessionBase - post v9 APIs package all arguments in a TlpcArgs object; this has templated constructors taking between 0 and 4 objects.

# What happens to applications then ?

- ABC.app becomes ABC.exe
  - ... To assign ABC.exe the capabilities it needs
  - ... To protect ABC's private data
  - ... Only a few code lines to change
- Application files need to be relocated

\System\Apps\ABC\ABC.app	\Sys\Bin\ABC.exe
\System\Apps\ABC\ABC.mbm	\Resource\Apps\ABC.mbm
\System\Apps\ABC\ABC.rsc	\Private\10003a3f\ABC.rsc

# Quiz - Loading DLLs and capabilities

- **Information**

- ... A Bluetooth game application Game.exe, with LocalServices capability, wants to load and call functions in GameEngine.dll which only computes game state and has no capabilities.

- **Question**

- ... Can Game.exe load GameEngine.dll?

- **Opinions**

- ... A - I think it can

- ... B - I don't think it can

- The situation is “**B** – cannot”

# Data Caging

- Access rules based on directory path
    - ... avoids overhead of Access Control Lists
  - \sys
    - ... read/write access only for TCB
    - ... binaries may only be executed from \sys\bin
    - ... tamper-evidence for binaries on removable media
  - \resource
    - ... readable by all, writable only by TCB
    - ... e.g. fonts, bitmaps, help files, ...
  - \private\  - ... read/write access only for the specified process and TCB
- other directories are public, with shared r/w for compatibility

# Impact on Developers

- Determining what capabilities are needed
- Deciding how to store and share data
- Changed and new APIs
- Impact on plug-in frameworks
- New features provide new opportunities
  - ... Programmatic access control (e.g. SIDs)

# Getting more information

## *Where can I go when I have questions?*

- Searchable SDK documentation
  - ...download most recent version from Symbian
- Searchable knowledge base (“FAQ”)
- Technical papers
- Example code
  - ...on SDKs & Web sites
- Free discussion forums
  - ...[www.symbian.com/developer/support.html](http://www.symbian.com/developer/support.html)
- Professional support services

# Questions

“Symbian OS Internals: Real Time Kernel Programming”, by Jane Sales

“Symbian OS Explained”, by Jo Stichbury

“Symbian OS for Mobile Phones vol2” , by Richard Harrison

...and of course visit the Symbian Developer Network  
to sign up for the newsletter at [www.symbian.com/developer/newsletter.html](http://www.symbian.com/developer/newsletter.html)