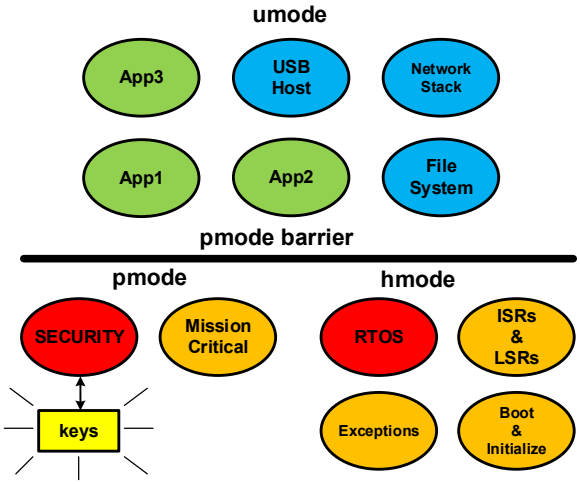
Protect Your Devices

for OEM Managers

What is the Problem?

Currently cyber criminals are exploiting the low-hanging fruit of email phishing and similar methods to break into computer systems. However these vulnerabilities are being closed off. Soon these bad actors will be forced to start breaching unprotected embedded and IoT devices. This may not become a big problem for a few more years. However now is the time for forward-looking companies to take action. Once your device has been hacked it is too late to avoid the negative consequences.



What is Our Solution?

We have recently released a new RTOS which provides a high-level of security for embedded and IoT devices. It is called SecureSMX, and it is aimed at microcontroller systems based upon the Arm Cortex-M architecture v7 and v8. It contains many innovative, patented solutions.

How Does It Work?

SecureSMX enables dividing an application into fully isolated partitions. Should a bad actor gain access to one partition, he or she cannot access other partitions. In addition, strong limits are applied to partitions such that bad actors cannot bring down the rest of the system through stratagems such as infinite loops or using up system resources. Security is further strengthened by putting critical resources below the *pmode barrier* (see diagram) and keeping vulnerable resources above the barrier.

Does It Support Existing Systems?

Yes. SecureSMX is specifically designed to enable moving vulnerable code into isolated partitions above the pmode barrier. A series of demos showing this process are posted at www.smxrtos.com/securesmx. Mission-critical and other code continue to run, as is, with little or no modification. Code moved into isolated partitions also requires little modification. SecureSMX fosters an iterative process where device security can be slowly improved over a period of time. Even if a device cannot be upgraded, once shipped, if it has a long lifetime ahead, it makes sense to start shipping less vulnerable devices.

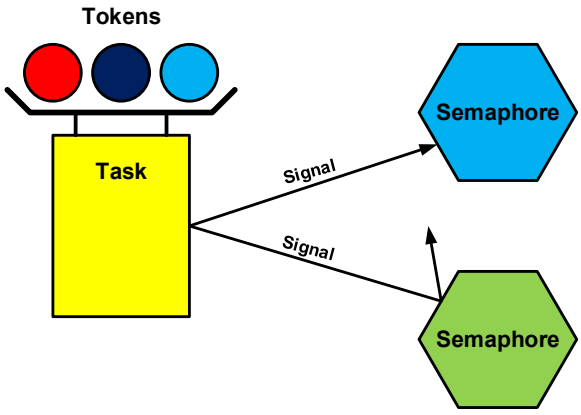
What About New Systems?

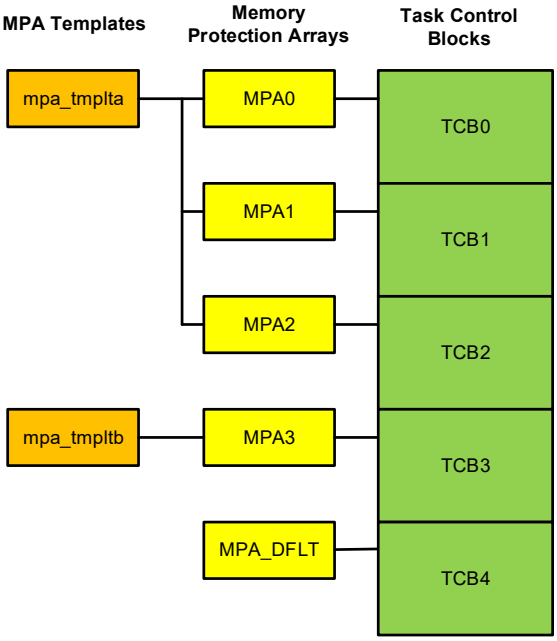
SecureSMX supports *frameworks*. These start with determining what modules are needed and how they must interconnect. Then a framework is built where each

module is placed inside of an isolated partition, its estimated size is emulated with an array, and its estimated processor usage is emulated with a loop. Interconnections are emulated with generic portals and some stub code. The entire framework will run by itself, emulating the final system. Individual developers can work on their modules and continuously test them within the full framework environment. The framework approach supports modern programming techniques such as Agile programming and CI/CD. As portals are fleshed out, misunderstandings are ironed early in the project. The net result is a rugged system with built-in security and delivered on time!

What If I Am Not Using SMX?

We provide FreeRTOS and ThreadX ports; other ports are being developed. Hence, your application with the porting layer can be easily moved over to the SMX engine where it will run as well or better than it did before. Once this is accomplished, the security features of SecureSMX can start to be employed.

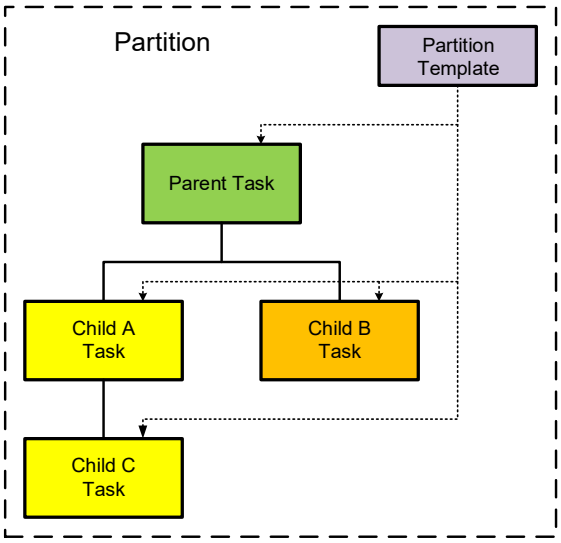




What Is Included In SecureSMX?

- SMX is a rich RTOS with considerable functionality and with many security and reliability features built in, such as parameter testing, event monitoring, error management, function callbacks, etc. It is not a new RTOS. It has been used in hundreds of devices since 1989.
- SecureSMX runs on top of SMX and includes innovative features to efficiently utilize the v7 and v8 MPUs and Cortex-M in order to enable truly isolated partitions, runtime limiting, resource control via tokens, moving ISR code into umode partitions, and numerous other features. SecureSMX is designed for high flexibility, which allows applying security features only where needed, and it provides alternative methods for achieving security objectives.

- smxAware is an RTOS plug-in for the IAR C-SPY debugger. It not only provides in-depth support for SMX, but it also permits viewing MPAs and MPUs conveniently.
- MpuMapper creates a map showing what partitions variables and functions are in. This is very helpful during debugging.
- MpuPacker facilitates getting the most efficient ordering of region blocks in memory for Cortex-v7M processors.
- FreeRTOS and ThreadX ports facilitate moving applications from these RTOSs to SMX in order to utilize SecureSMX security features.
- Partition demos that show the step-by-step process for moving a module into an isolated partition above the pmode barrier.
- smx User's Guide, smx Reference Manual, and SecureSMX User's Guide. Each of these 200+ page, carefully-written manuals provides a wealth of accurate information. In addition manuals are available for smxAware, smxBase, eheap, target guide, and others. These manuals can be freely downloaded from www.smxrtos.com.



Pricing

Please contact sales@smxrtos.com for licensing and pricing information.

Micro Digital

www.smxrtos.com/securesmx