Bodacion Technologies

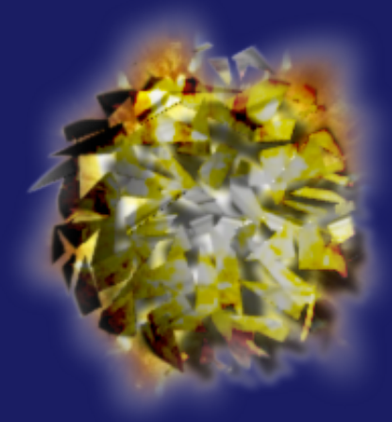
Presents a brief overview on the building of a new security technology:

HYDRA

ACSA
Applied Computer Security Associates

Introduction

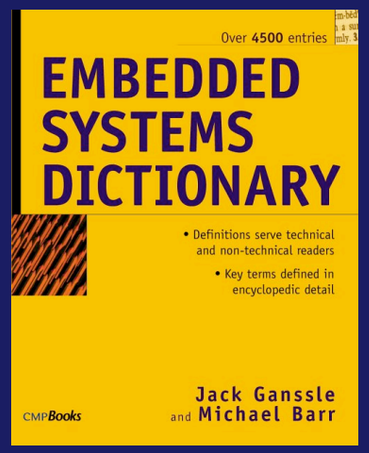
- HYDRA is touted as “unhackable” and “invulnerable”



- Today I will show you:
  - What an embedded system actually is
  - Why we created one over using more popular methods
  - The steps in creating HYDRA that make it singularly secure

What Is "Embedded?"

- From the book "Embedded Systems Dictionary" by Jack Ganssle and Michael Barr



embedded system

n. A combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a dedicated function. In some cases, embedded systems are part of a larger system or product, as in the case of an antilock braking system in a car.

What Isn't "Embedded?"

- Confusing market-speak abounds:
  - "We rolled our own OS"
  - "It's an embedded appliance"
  - "We use the (product-name) OS"
  - These are typically modified Linux or BSD kernels

- Back to the book:

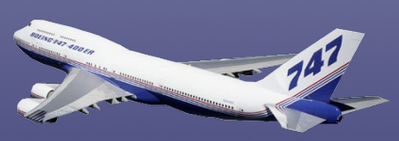
general-purpose computer

n. A combination of computer hardware and software that serves as a general-purpose computing platform. PCs, Macs, and Unix workstations are the most popular modern examples.

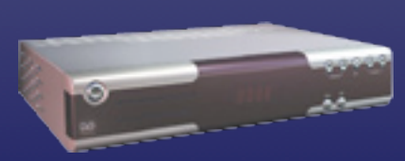
The Operating System Spectrum

Exec Loop	µC/OS VRTX	VxWorks WinCE QNX	RT Linux Embedded Linux	Linux BSD Windows

Hard Real-time — Some, or Soft Real-time — No Real-time



Embedded is Often Talked About...

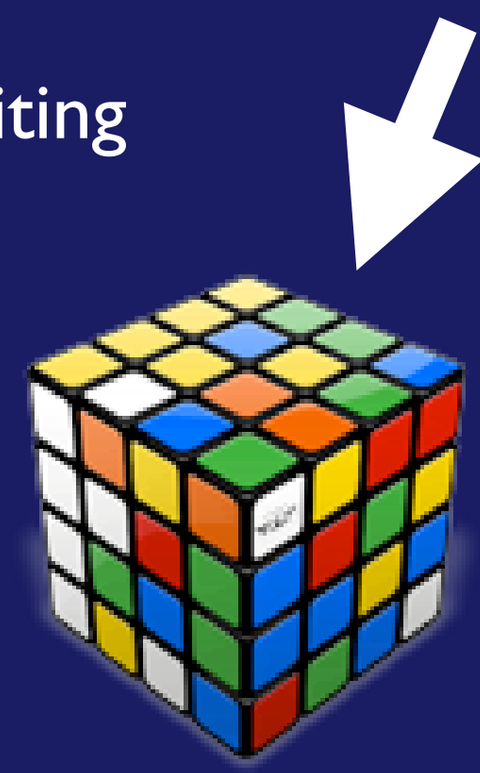
- Marketing knows it's a great idea
- It is not dishonesty or subterfuge
- It is a simple misunderstanding
- "Embedded" builds an association in a customer's or user's mind between headless systems and simplicity
- A recent example of the danger of this misunderstanding:

Nachi worm

SANS: "Bad design decisions here: Running a single purpose appliance on a general purpose OS."

... But Embedded is Rarely Done

- Disparate skill sets
- Much longer road, requires a different discipline
  - Example - "favorite number" addressing
  - Example - Memory deallocation
- Think writing an OS, not modifying it or writing an application for it
- Tool and library availability
- Not typically considered enterprise scale



Why We Chose the Path

1

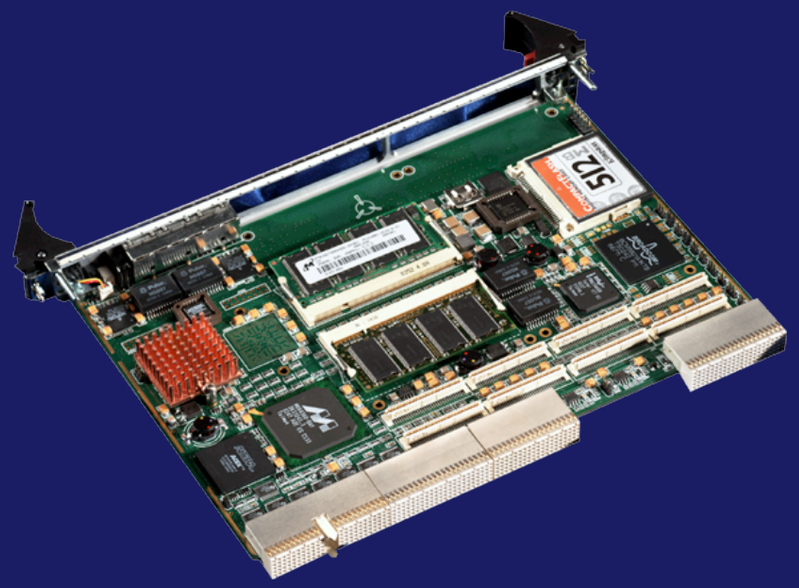
No other viable choices given our requirements and the state of current technology

2

We already had the skills, tools, and experience required to design and implement

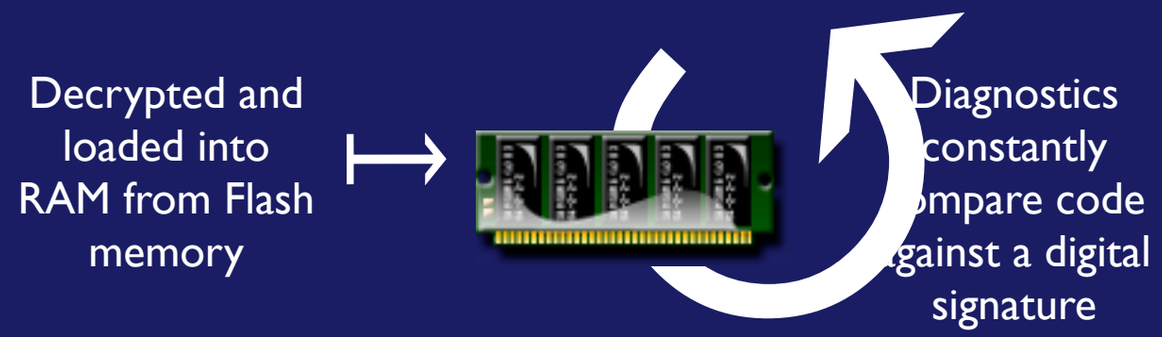
Hardware

- Selected solutions used in Military and other demanding environments
- cPCI - an industry standard card/backplane ("blade") available from numerous vendors
- PowerPC card for low power and heat without sacrificing performance

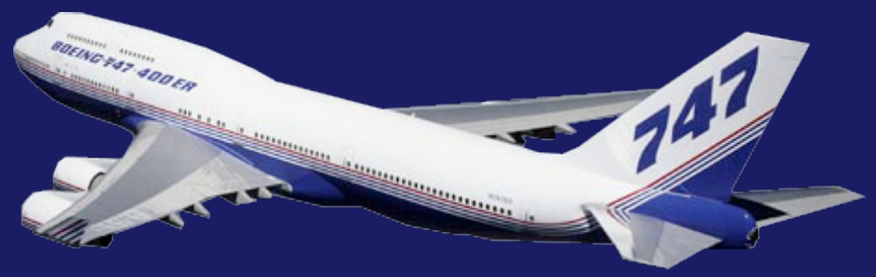


BSP

- Board Support Package
  - Loads code from firmware
  - Virus proof!
  - Encrypted
  - At about 1.5 MB, it would fit on an old floppy

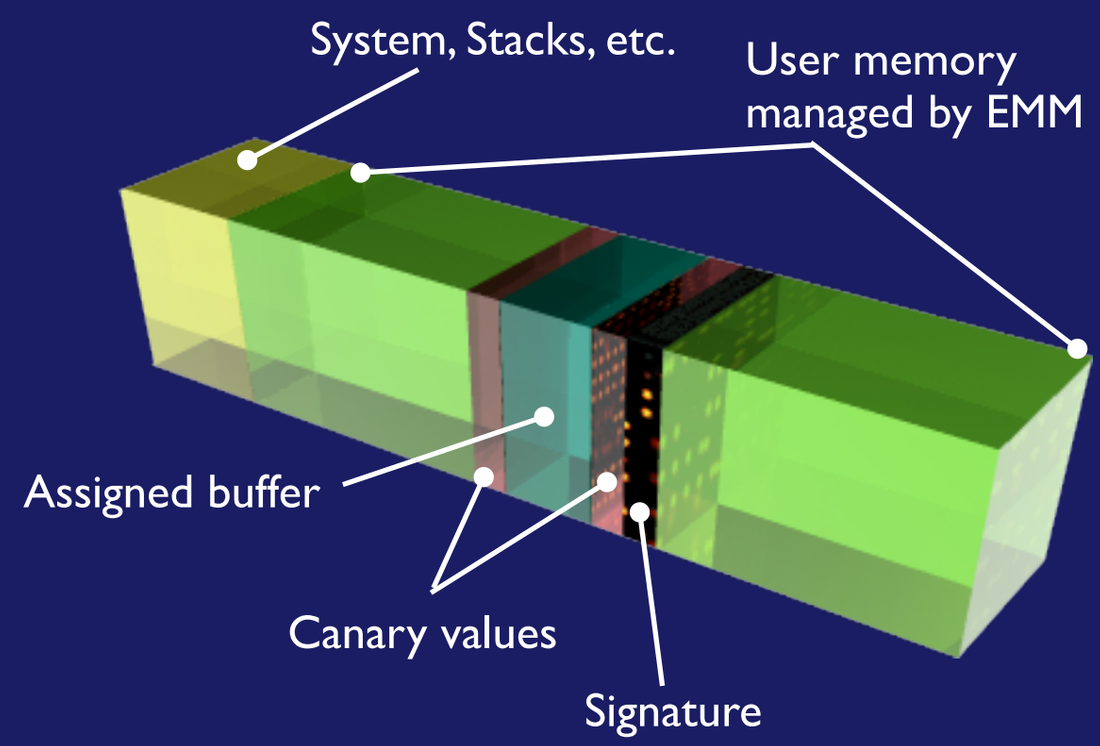


- Started with a commercial RTOS
  - Hard-real time operating system
  - Deterministic scheduling
  - DO-178B Level A certified reliability



Memory Management

- Monolithic memory model
  - Faster data sharing
  - Reduced fragmentation
- Proprietary memory model and manager features:
  - Canary values to prevent buffer overruns
  - Signed blocks for data caches
  - Constantly scanned by internal diagnostics
  - Hard real-time performance



Crypto Functions

- FIPS approved ciphers in HW and SW
- Biomorphics
  - The foundation of all the pseudo-random numbers for HYDRA
  - Produces non-correlated, non-repeating sequences

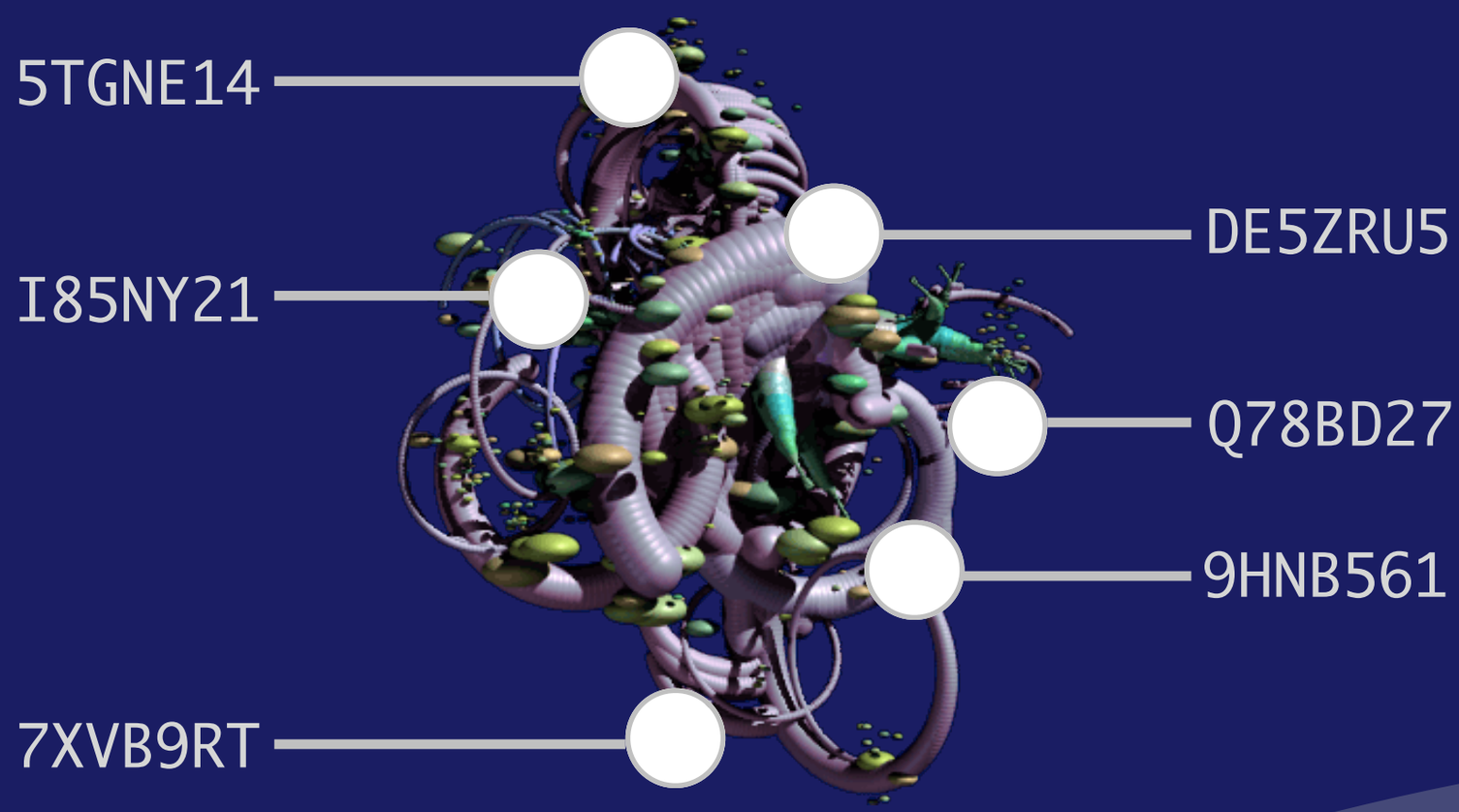
Virus Scanning

Random Number Generation

Content Signatures

TCP Initial Sequence Numbers

SSL/TLS Sessions and Certificates



Cryptographic Key Material

Session ID's

Access Verification

Logging Signatures

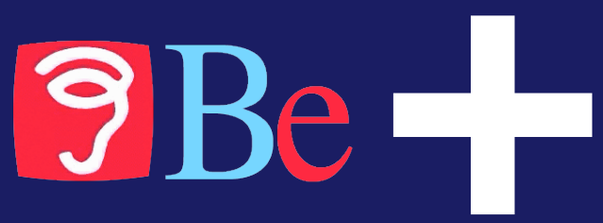
Password Protection

Faster Hashing

More...

Network and File Systems

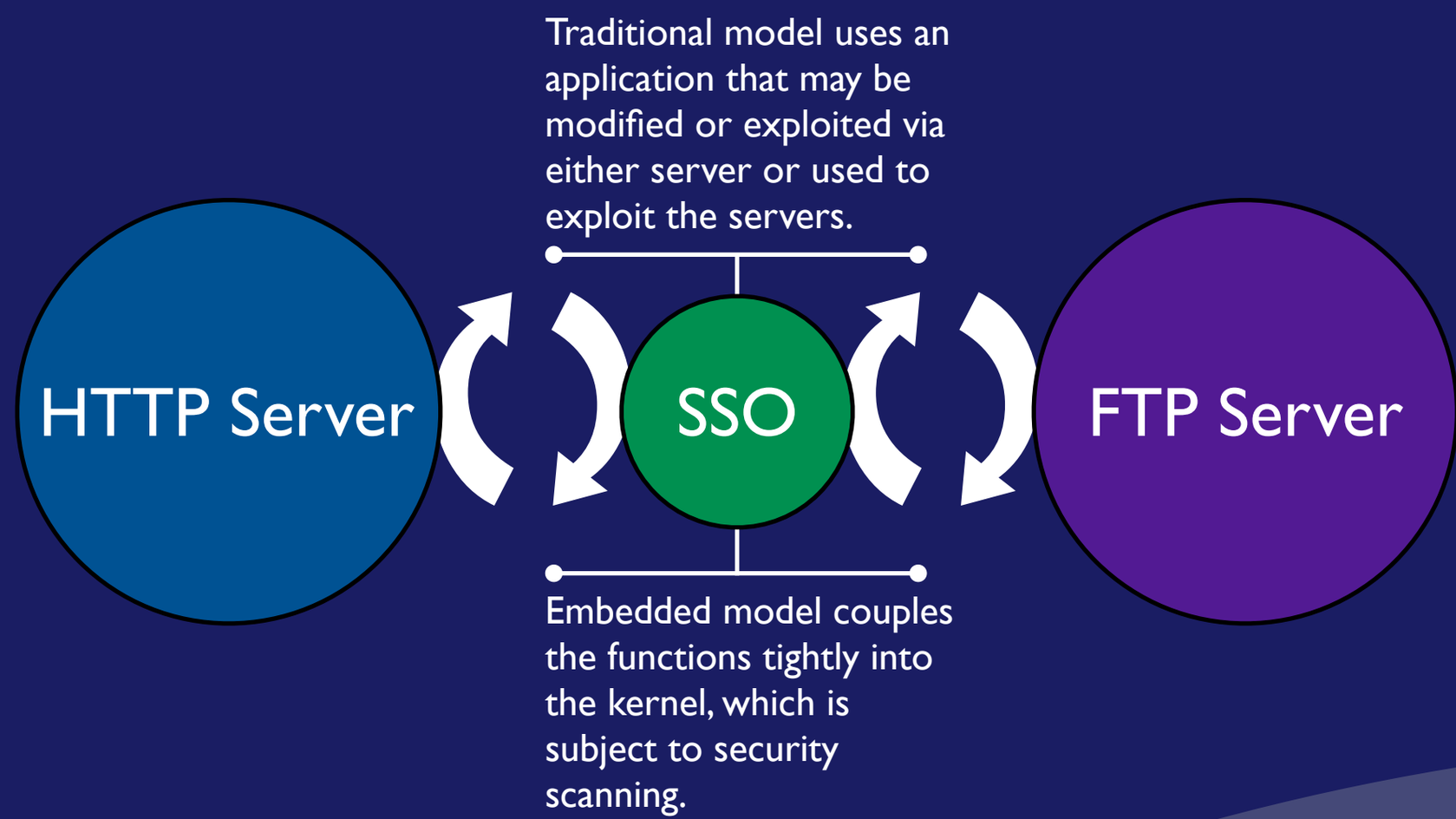
- Created a TCP/IP stack
  - Commercial libraries had too many bugs and inadequate performance
  - Most are designed for a trickle of packets, we needed a fire hose
  - Allowed us to add security features
- Created a file system
  - Commercial options were typically DOS-based
  - We needed performance (which means anything but DOS)
  - We needed robustness
  - We needed security



User-to-user separation
Journaling
Caching

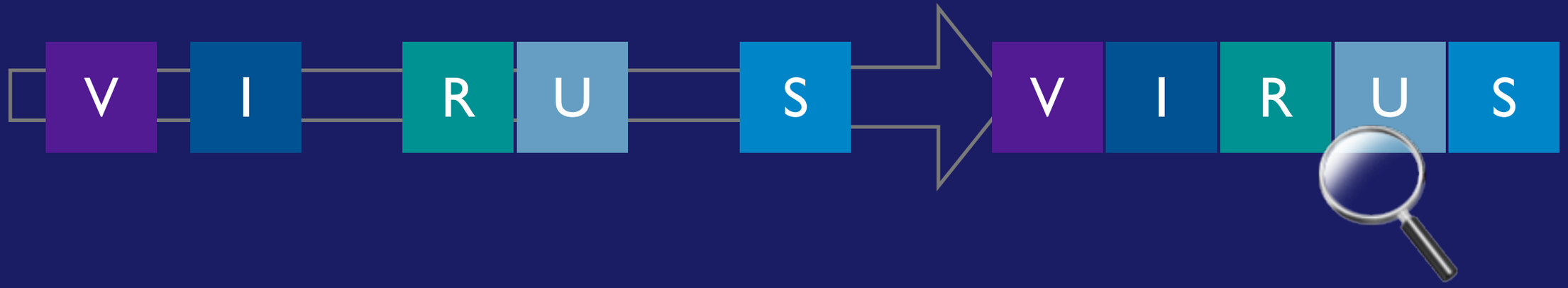
FTP Server

- FTP(S) Server
  - Grab the RFC, then start the state machine design
  - Added features such as single-sign on



HTTP Client and Server

- Grab the RFC, then start the state machine design
  - All the typical server features (e.g. virtual servers)
  - Proxying and load-balancing
  - Connectors to Application Servers
  - Buffers entire request and scans for ANY anomaly



SQL Injection • Intel Byte Codes • Source File Viewing
Cross Site Scripting • Illegal Navigation • Many More...

Embedded Only

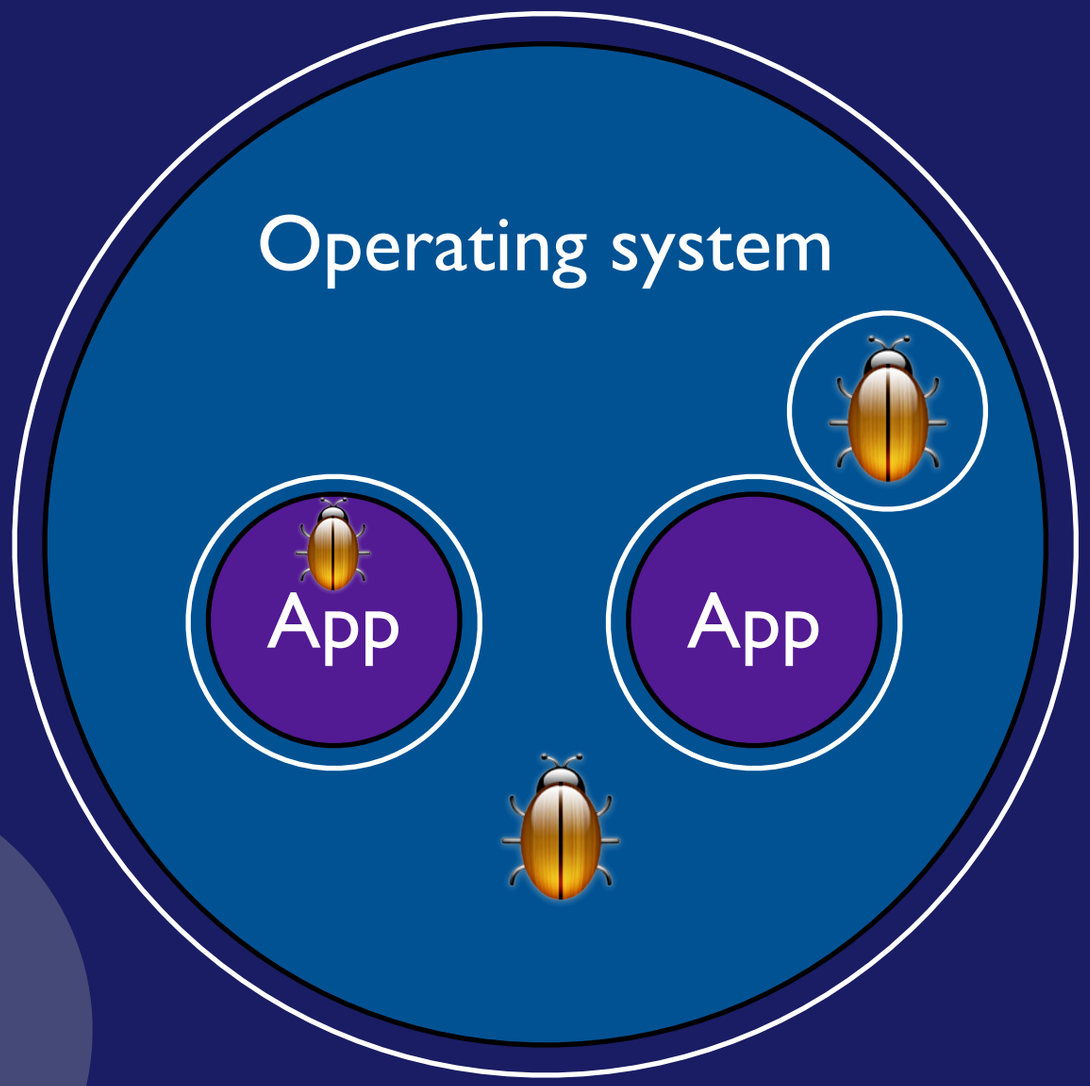
- Real-time rate monotonic analysis
  - Optimal task priority assignment in which higher priorities are accorded to tasks that execute at higher rates
  - Allows a processor as slow as 333 MHz to saturate a 100 Mb network
- True Zero-copy from Multiple Sources
  - Tasks just pass pointers around
  - Buffers can be check-summed or signed
- Startup takes about four seconds
- Watchdog Timers

Success!

- Enterprise performance
- Successful hacker challenge
- NSA SPOCK program
- Successful deployments
- Surprise and disbelief at the security and simplicity

Embedded is not Automatically Better

- HYDRA is embedded because we needed it to be better, it is not better because it is embedded



Internal IDS

No complex settings (e.g. NFS vs. Web server)

Anything that's not "perfect" is considered an attack (e.g. Intel byte codes)

We know we're a dishwasher, put in a TV dinner and we'll complain! That's what an appliance does

Key Points

- Embedded is not a cure-all
  - HYDRA incorporates defenses from the driver level to the Web server
  - Many of these defenses have nothing to do with being embedded
- Development is far more difficult and requires more time
  - Developer skill set is fundamentally different from Linux or Windows
  - Applicable libraries are few and either buggy or (ahem) mature, so you'll end up doing all the work yourself
- Rewards are worth the effort if security is a concern
  - There is simply no other way to get the same level of security
- Appliance ≠ Embedded
- Usually Embedded ≠ Embedded

Thank You!

Contact: Eric Uner
uner@bodacion.com
Bodacion Technologies
http://www.bodacion.com
(847) 842-9008