Multiple Independent Layers of Security (MILS) Network Subsystem Protection Profile (MNSPP)

An Approach to High Assurance Networking Rationale

The MILS Network Subsystem (MNS) is

A *class* of subsystem that:

- runs on MILS separation kernels
- is developed for environments requiring medium to high robustness (EAL4 - EAL6+)

… is intended to solve the problem:

- to provide reliable and secure network services
- to be resistant to sophisticated attacks

… and:

- ranges over configurations defined by the MILS Network Subsystem PP
- is not skewed toward a particular vendor approach
- is a “pluggable” MILS component
- interoperates with other MILS and non-MILS peers
- gives precedence to security considerations over other considerations (e.g. throughput, simplicity, code space, etc)

The MILS Network Subsystem is also

- Is scalable over a range of configurations, e.g.:
  - Large-scale MILS servers and MILS clusters
  - MILS workstation hosts
  - Custom networks of MILS components
  - MILS-based high-robustness network appliances
- Provides flexible options for product developers
  - MSL or MLS realizations are possible
  - Interoperable with existing protocols / devices
- Balances Robustness / Performance / Interoperability to achieve
  - (any)MNS-to-(any)MNS may lead to additional features (RFCs)
  - MNS-to-hostile-network must be interoperable and robust
- Provides for growth and evolution
  - E.g., developers may implement IPv4 and/or IPv6 products

MILS Network System Key Concepts

- **Range of Features**
  - **Protocols and Services - e.g., TCP, IP, UDP,**
  - **“Profiles” - Functional packages defined by other parties, e.g. DISA, SRI, The Open Group**
- **Diverse Implementation Techniques**
  - **“Virtualization of Stacks” (Vanfleet) - degree of isolation of data from different clients**
  - **Strength of isolation is a factor in robustness -- use SK resources for highest robustness**
- **Degrees of Assurance**
  - *Sub-Profiles* **defined by PP as “chunks” of functionality and a sub-profile type**
  - *Sub-profile types* **(A, B, C) - like EALs or DO-178B levels, applied to Sub-Profiles**
  - **CC assurance levels EAL4+, EAL5+ and EAL6+; DCID* 6/3 protection levels PL 3, 4 and 5**
  - **Formal description of network stack components based on a protocol component model**
- **Protocol component model for specification and implementation analyses**
  - **Layered interfaces**
  - **Service provider (SP) / service user (SU)**
  - **Service primitives - abstract, atomic, implementation-independent interaction between SP-SU**
  - **Protocol entity / (its) Peer**
  - **Protocol specification**

*** Director of Central Intelligence Directive**

MNSPP Security Environment

- Enumerate the Assumptions, Organizational Policies and Threats
  - Assumptions concerning external factors
    - 2 network types:
      - Closed networks – protected from intrusion by physical security
      - Open networks – unknown and potentially malicious entities may have access to network resources
    - Open networks present more security challenges and require more complex assurance scrutiny
  - Organizational Policies concern
    - Functional – Address security in layers, just as networking is implemented
    - Provides Defense in Depth
    - Allows for flexibility in protocol implementation at upper layers
    - Identify and secure interaction between layers
  - Threats occur in every phase of the life cycle:
    - Development -- failure to avoid or eliminate flaws
    - Configuration -- delivery, installation, and configuration
    - Interaction (malicious) -- action of malicious subject, user, or external agent
    - Interaction (non-malicious) -- human user or administrator error
    - Physical -- intentional / unintentional physical compromise

MILS Network Stack Validation

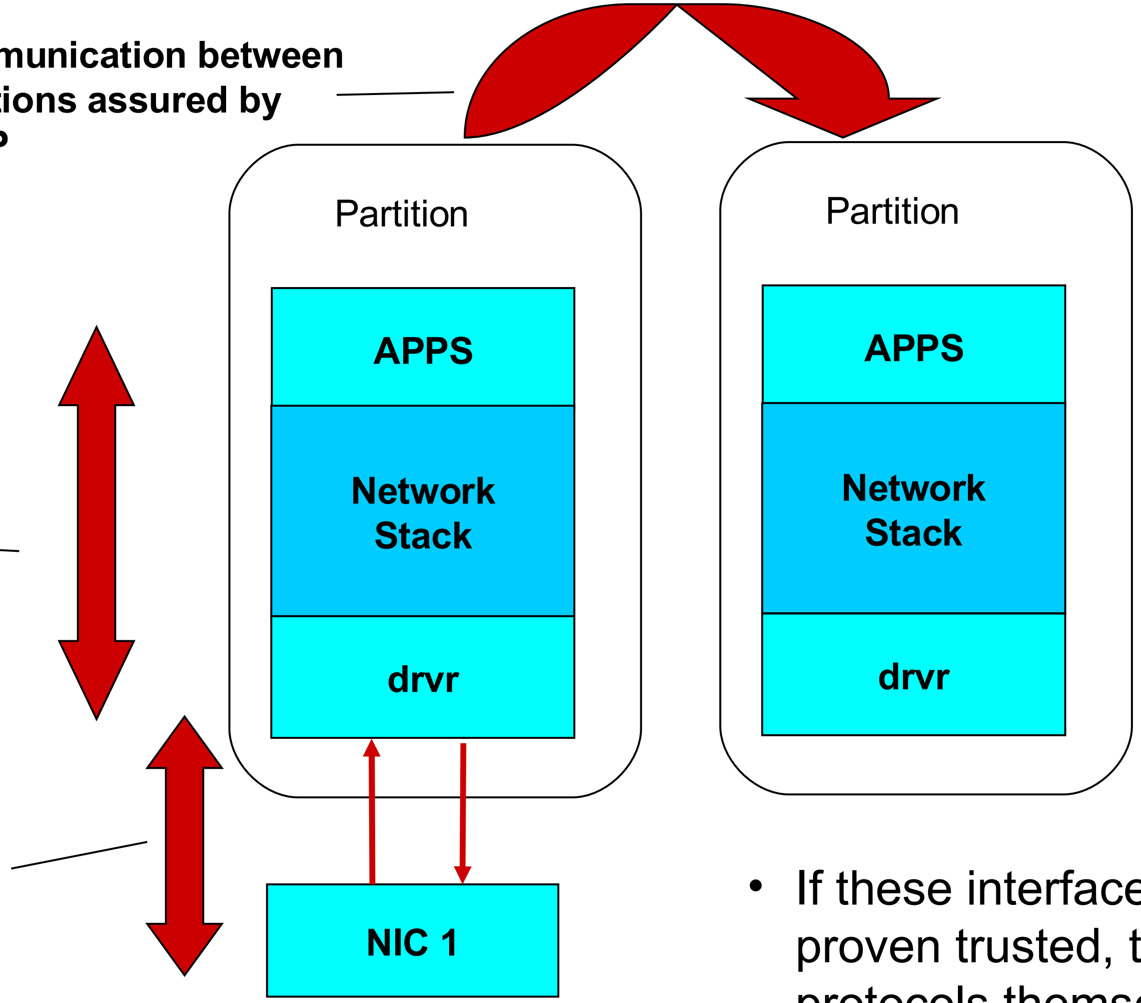
- Address security in layers, just as networking is implemented
  - Provides Defense in Depth
  - Allows for flexibility in protocol implementation at upper layers
  - Identify and secure interfaces between layers
- Approach to classify networks as closed or open
  - Closed Network: A network in which physical security prevents unauthorized access to the nodes and media of the network.
  - Open Network: A network in which one or more ‘vulnerable’ points are accessible, potentially by malicious entities.
  - Open Networks require much more attention to threats and policies
    - intruders will attempt to exploit vulnerable points
    - We cannot know a-priori what types of systems/nodes will attach to vulnerable points
    - Information at all security levels must be protected until nodes are authenticated and authorized

The Layered Approach

Communication between partitions assured by SKPP

Communication within Partition assured by SKPP

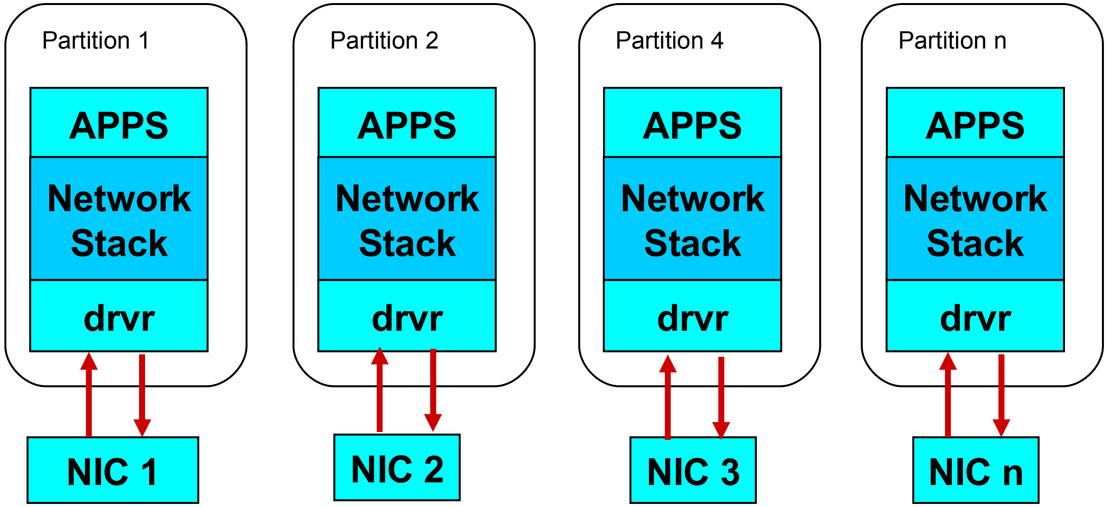
Communication with Network drivers/chip/media assured by MNSPP



- If these interfaces are proven trusted, the protocols themselves become less relevant

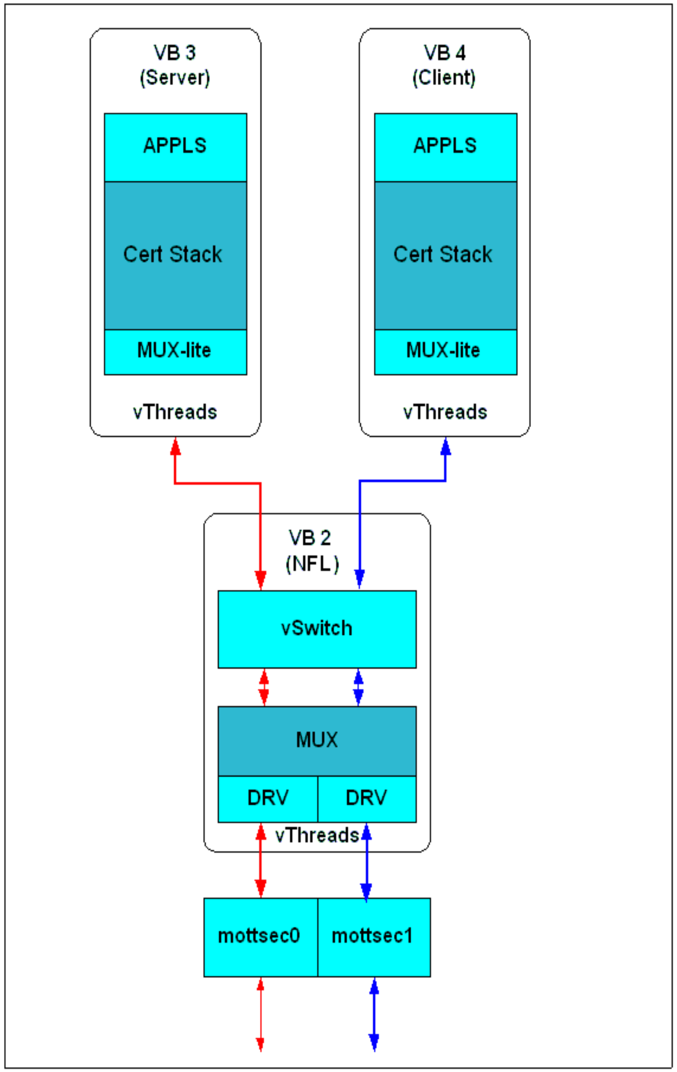
1 Partition Model

- Single-level secure
- Each partition has full network stack and network interface (multiple NICs)
- Separation is guaranteed via SK
- Pros: simplicity, high leverage of SK
- Cons: requires lots of redundant code, memory space, multiple network interfaces



Multi-partition Network Stack Models

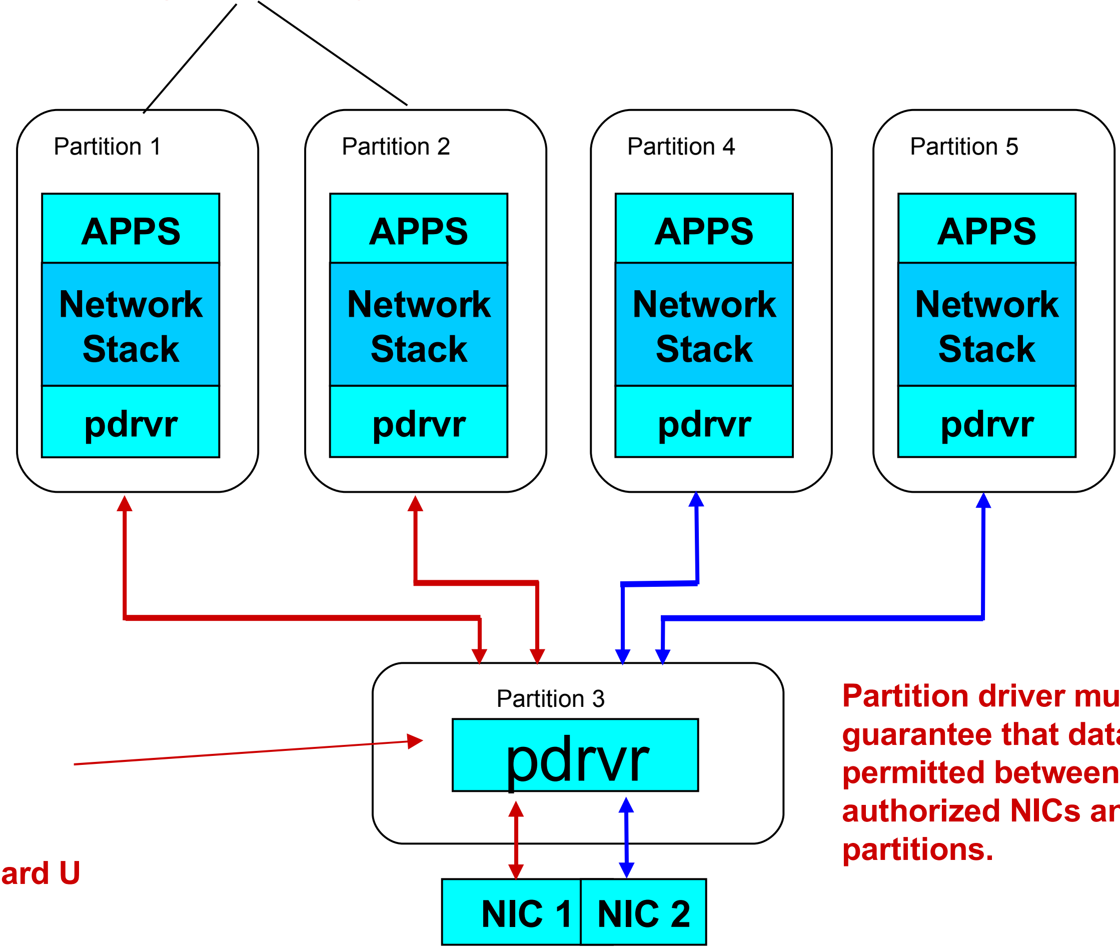
- Divide the network stack between secure partitions and a common network driver (HA) partition.
- To the extent possible, make the HA code protocol agnostic
  - Allows the most flexibility in protocol implementation
  - Keep certification costs lower by moving protocol stacks outside of HA
  - Rely on SK to securely deliver data to the HA network partition
- **Pros:**
  - reuse of common HA partition
- **Cons:**
  - still redundant network stack code in partitions
  - Greater security burden on common networking partition



WIND RIVER

2 Partition (2p) implementation example

Separation
Between partitions
Is guaranteed by SKPP



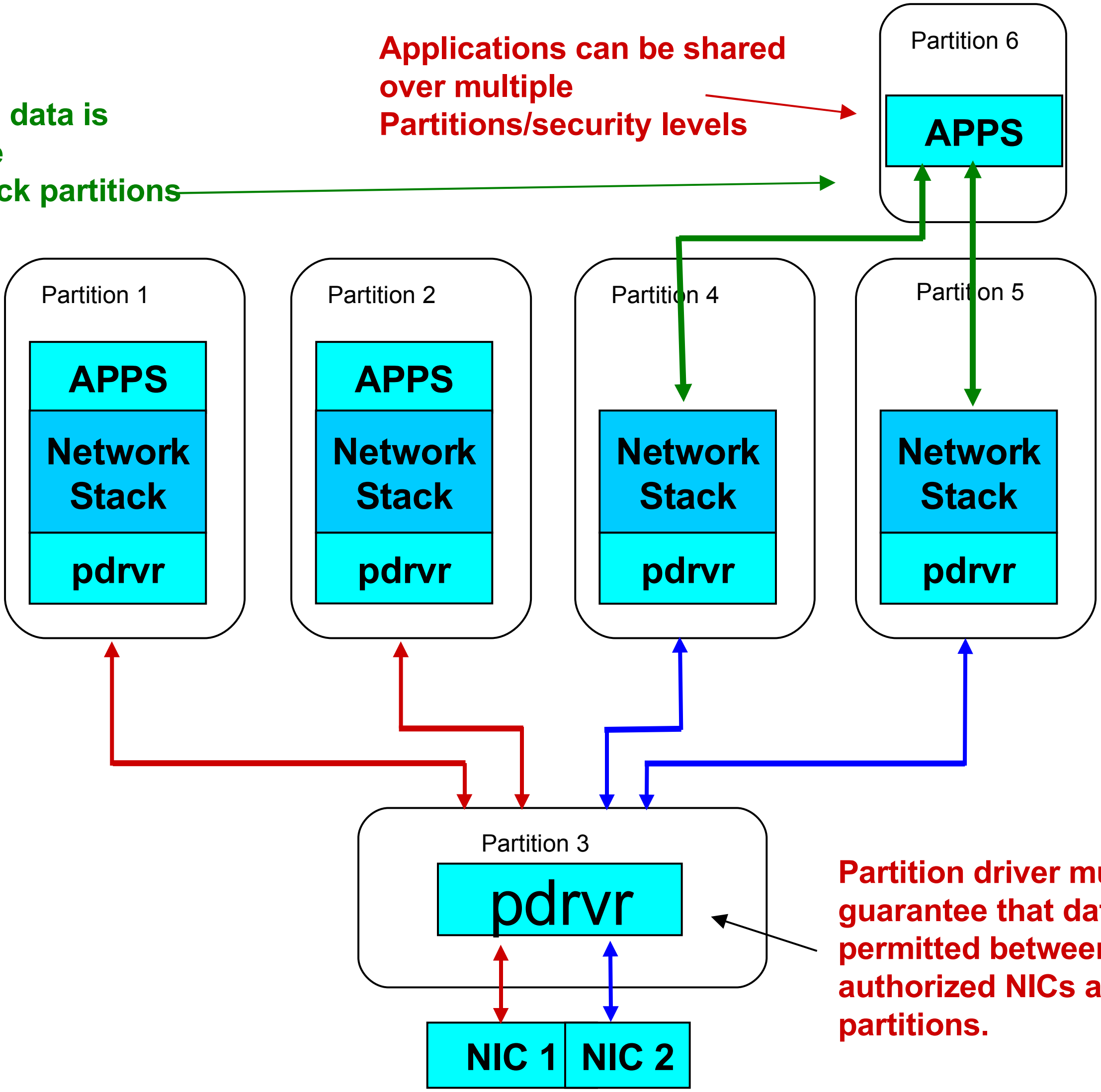
Partition driver must guarantee that data cannot be incorrectly routed to the wrong partition (illegal onboard U turns).

Partition driver must guarantee that data is only permitted between authorized NICs and partitions.

3 Partition Network Stack Model

Applications can be shared over multiple Partitions/security levels

Must guarantee data is Sent only to the appropriate stack partitions

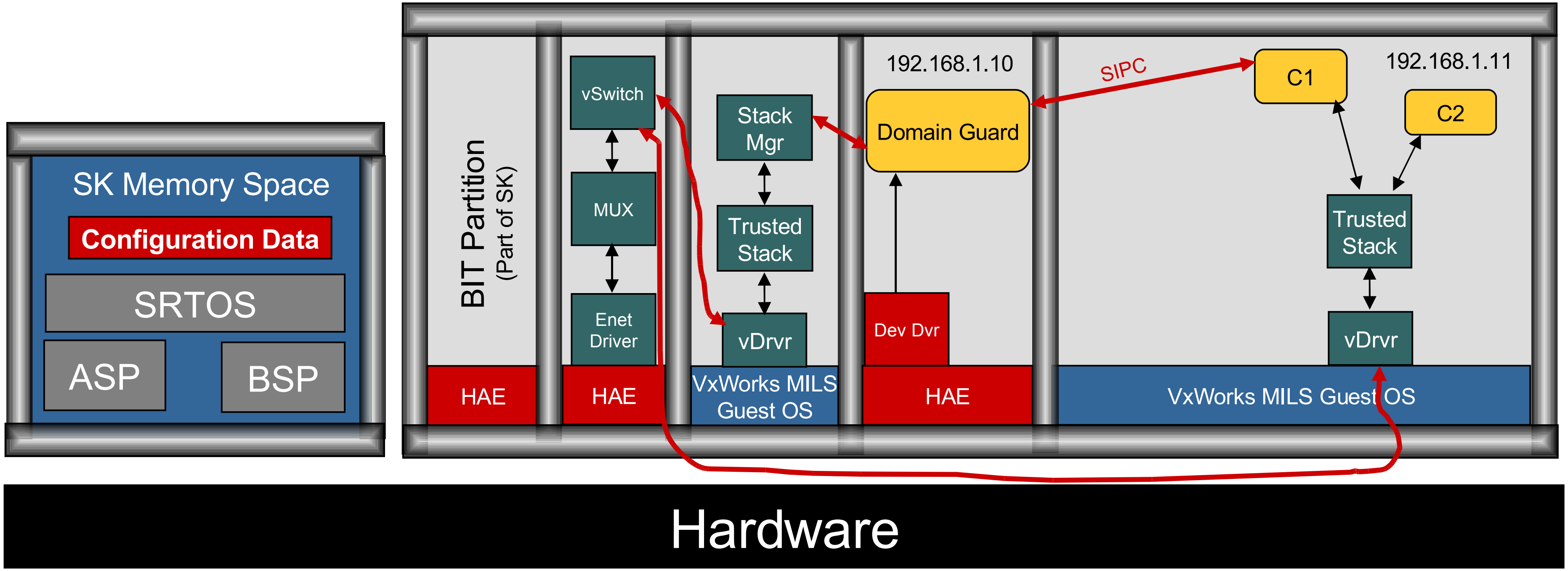


Partition driver must guarantee that data is only permitted between authorized NICs and partitions.

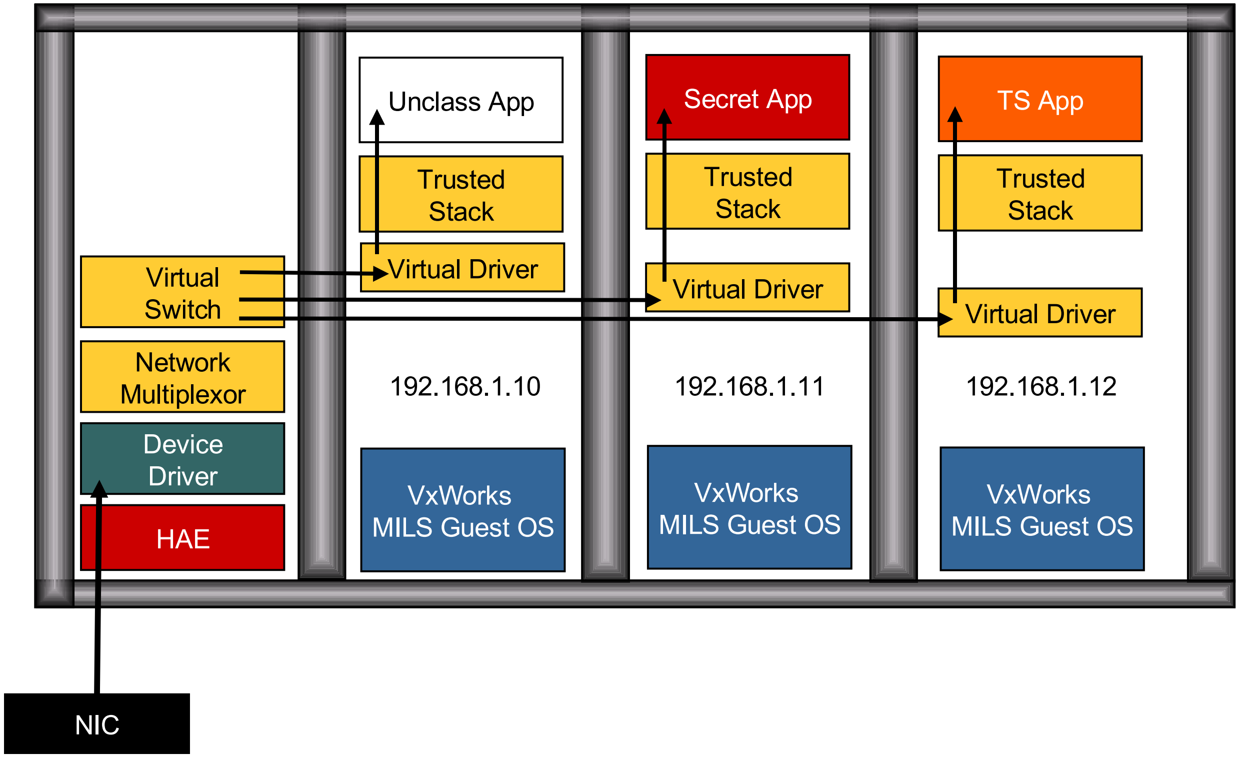
Encryption can help

- If communications to the network are encrypted, accidental/malicious interception is not harmful
- Must guarantee secure establishment environment
  - IPsec security associations
  - What about layer 2?
- Encryption can be expensive
  - CPU cycles
  - Crypto coprocessors
  - Need to provide secure environment for unencrypted traffic also

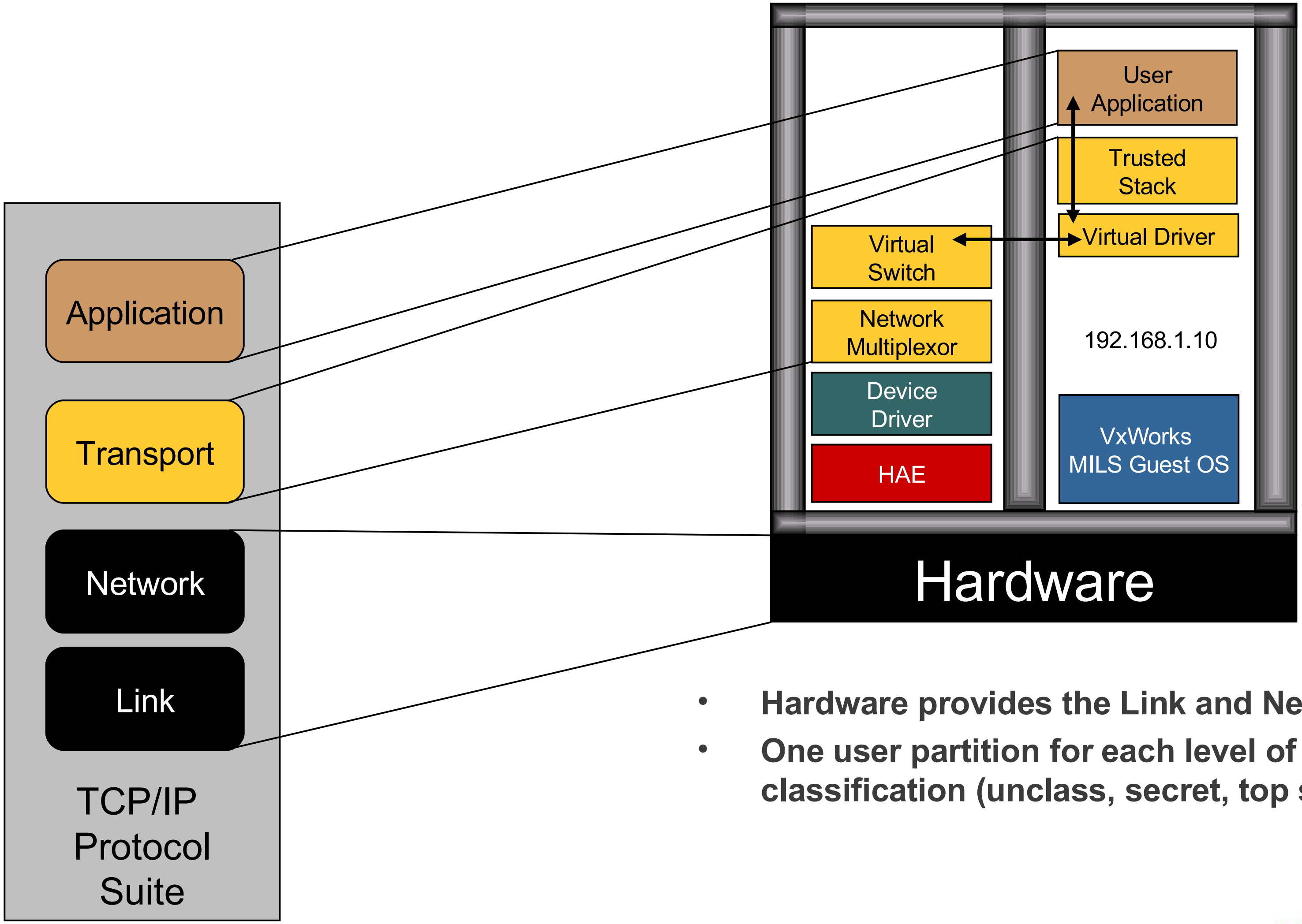
Customer #1


SK Memory Space
Configuration Data
SRTOS
ASP
BSP
BIT Partition
(Part of SK)
HAE
vSwitch
MUX
Enet Driver
HAE
Stack Mgr
Trusted Stack
vDrvr
VxWorks MILS Guest OS
192.168.1.10
Domain Guard
Dev Dvr
HAE
SIPC
C1
192.168.1.11
C2
Trusted Stack
vDrvr
VxWorks MILS Guest OS
Hardware

Customer #2


Unclass App
Secret App
TS App
Trusted Stack
Trusted Stack
Trusted Stack
Virtual Switch
Virtual Driver
Virtual Driver
Virtual Driver
Network Multiplexor
192.168.1.10
192.168.1.11
192.168.1.12
Device Driver
HAE
VxWorks MILS Guest OS
VxWorks MILS Guest OS
VxWorks MILS Guest OS
NIC

Customer #3 - Notional



- Hardware provides the Link and Network layers
- One user partition for each level of data classification (unclass, secret, top secret, etc.)

Summary of Wind River Progress

- High-assurance systems can be built without requiring the entire stack to be EAL-6+
  - Evaluate network interface code to High-Assurance
  - Rely on SK to protect stack code within a partition
  - Results in far less code to be evaluated
- Smaller set of *Threats, Policies* and *Assumptions* to identify
  - Shorter evaluation time
  - Lower certification costs
  - Can accelerate market adoption without compromising existing MNSPP design
- Design getting favorable reviews from prospects
- Experience with MILS SK has helped form perspective on network stack requirements

Milestones for end of November

- Work with SRI to match SKPP assumptions with MNSPP assumptions for 2 partition stack model
- High-level design of HA stack code enabling:
  - Code size estimates (ELOC)
  - Certification cost estimates
  - EAL4 and EAL6+
- Get validation for 2 partition model from at least 5 prospects
  - Suitability of design
  - Timeframe
  - Certification costs

Further work

- Offload co-processors
  - Cryptography
  - IP forwarding
  - Checksum calculators
- How much information can be gained before the system blocks intrusion?
  - Addresses
  - Network size
  - Vendor Ids
- What authentication mechanisms can be used for high-assurance?
  - IPsec, X.509
  - Layer 2?
  - Other methods?
- Ensure that buffers are not reused
  - Memory protection
  - Scrub buffers when freed
  - Assure no unintended access
- Denial of Service/resource exhaustion issues
  - External firewall to isolate ‘open’ ports
- Layer 2 broadcast/discovery issues
  - How to distinguish valid from invalid discovery
- Performance considerations
  - Copying data = performance hit, but sharing buffers = security risk

WIND RIVER