Cyber Security: Issues, Applications, Solutions

Dr. S. Rakshit

CAIR(DRDO)

Computer Security Day

SETS & IEEE-ACM

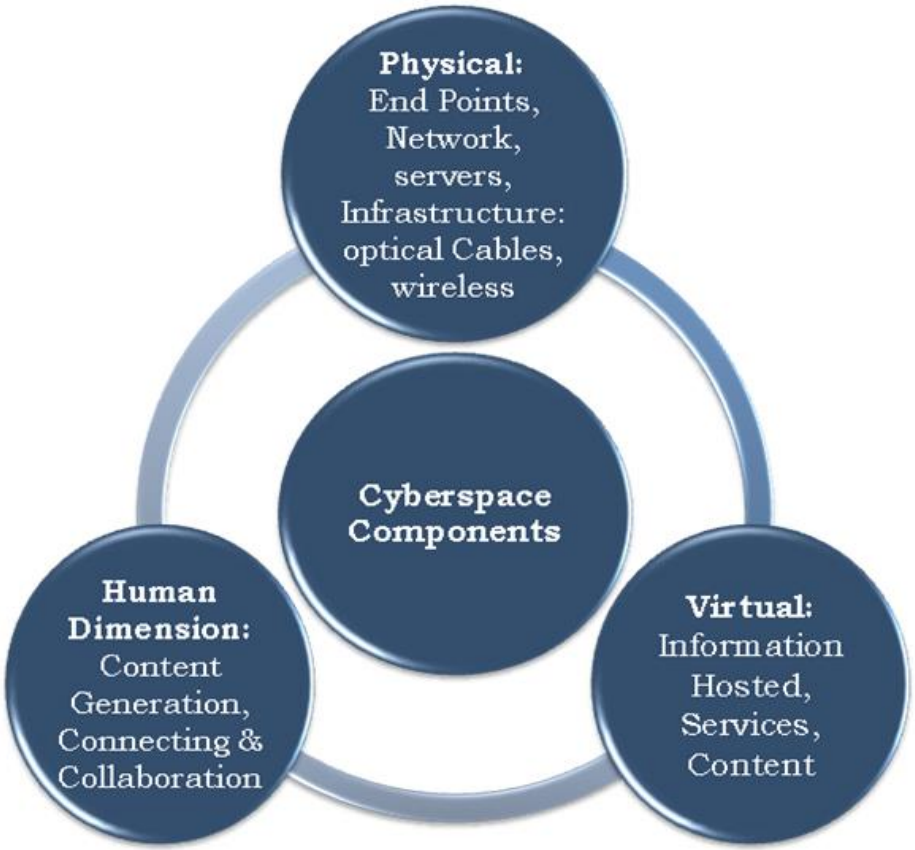
29 November 2019

Asset Model for Cyberspace

"The notional environment in which communication over computer networks occurs" oxford dictionary

Importance of Cyberspace

- Provides **Connectivity** to critical infrastructure
- Exchange of **Information**
- 24/7 On-demand **Services**
- Peer to peer **Content** and **Collaboration**



Unbounded Growing Domain

Exploitation of Cyberspace

Public's Exploitation

- Access to information
- Access to Services
- Distributed groups
- Protected communication

Adversary Exploitation

- Access to technical info, location info, background on VVIPs
- Form & coord distributed groups
- Securely exchange plans

LEA/GoI Exploitation

- Visitors to sites
- Intercept keywords
- Track groups
- Monitor comm

Cyberspace is an advantage if we have technical edge

What do we imply by ‘Security’

- A set of Do’s and Don’t’s
- **Preserving a set of properties**
- **Enforcing a desired user policy**

Security: Military vs LEA

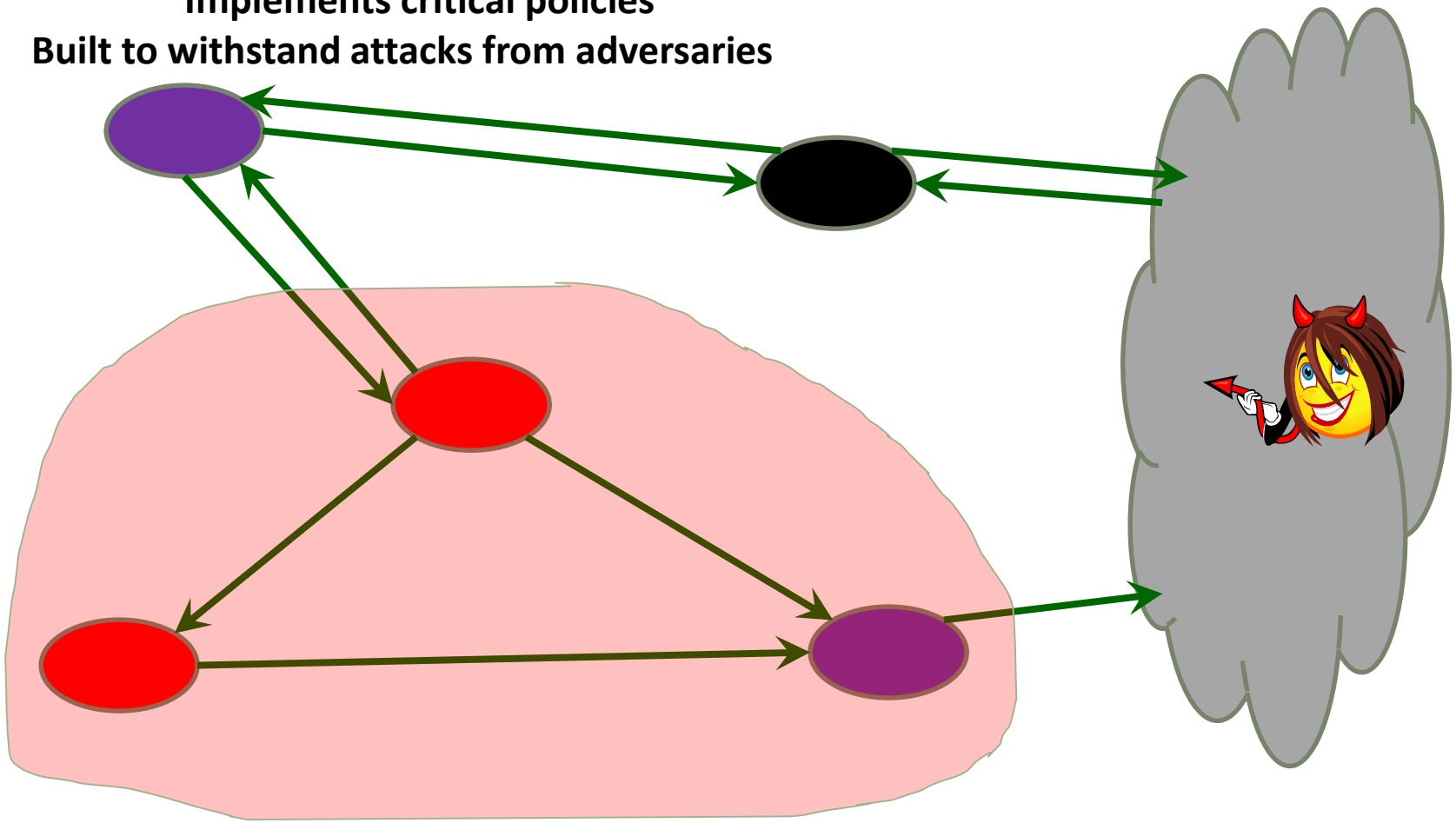
1. LEA paradigm
   a. Legal framework draws the line
   b. Security is in context of governance cover
   c. Best effort, deters attacks by aiding LEA
   d. Economics of feature set
2. Military paradigm
   a. All is fair in love and war …
   b. Security is in context of avowed adversary
   c. High assurance, prevents sec failure
   d. Technical evaluation of assurance

CyberSec: Military or LEA?

1. Characteristic Differences
   a. Military:
      i. No governance. Perimeter exists. High cost
   b. LEA:
      i. No perimeter. Governance exists. Low cost
2. Based on geometry of boundary
   a. LEA: The contact surface is fractal
3. Based on existence of governance
   a. Military: Non-attribution, inadequate laws
4. Based on cost
   a. LEA: ‘China price’ is the norm for ICT

System Design for Security

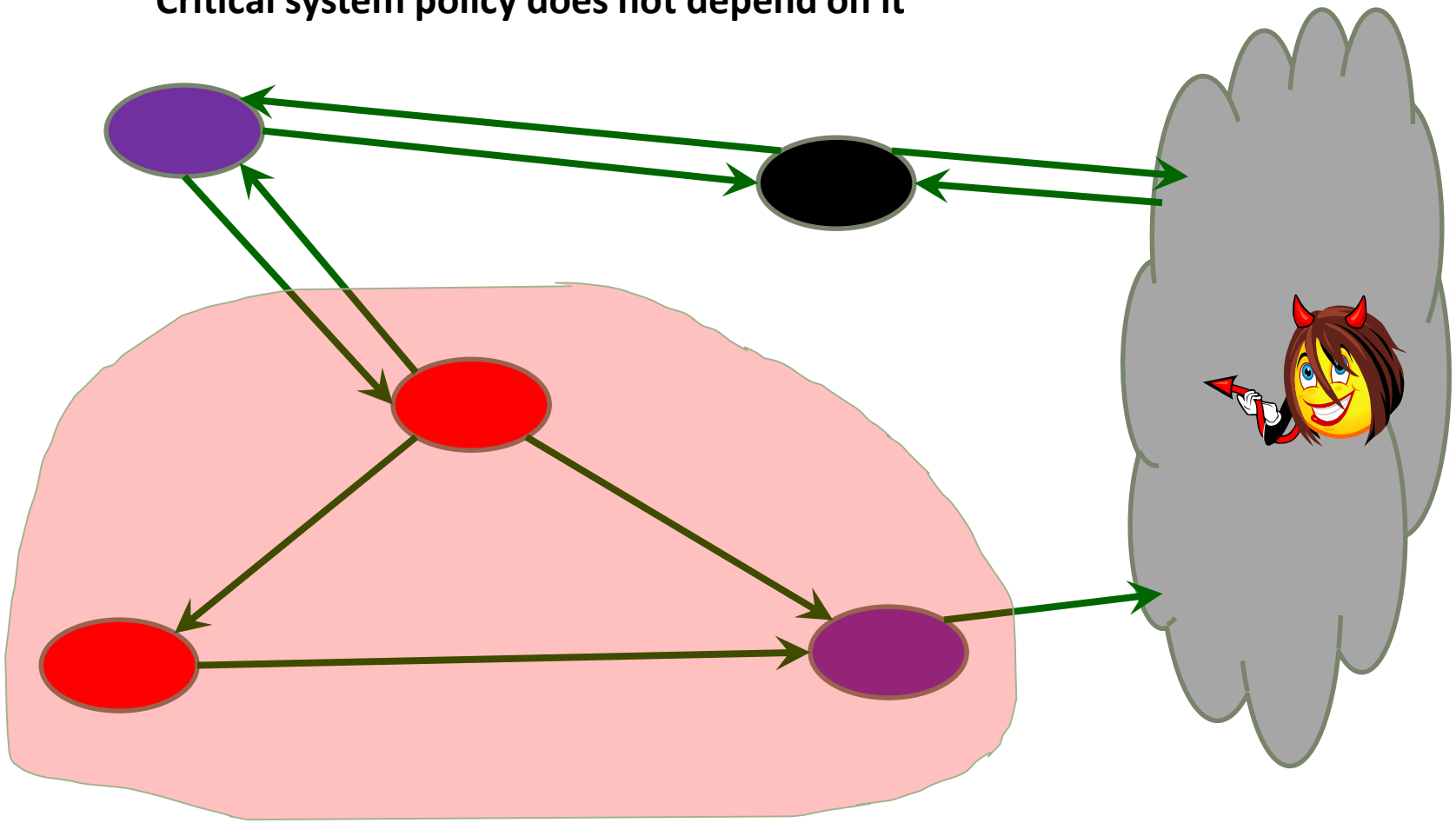
Purple: Simple enough to be sure of correctness. From trusted source
Implements critical policies
Built to withstand attacks from adversaries



Placing / Selecting the Purple for Red / Black

System Design for Security

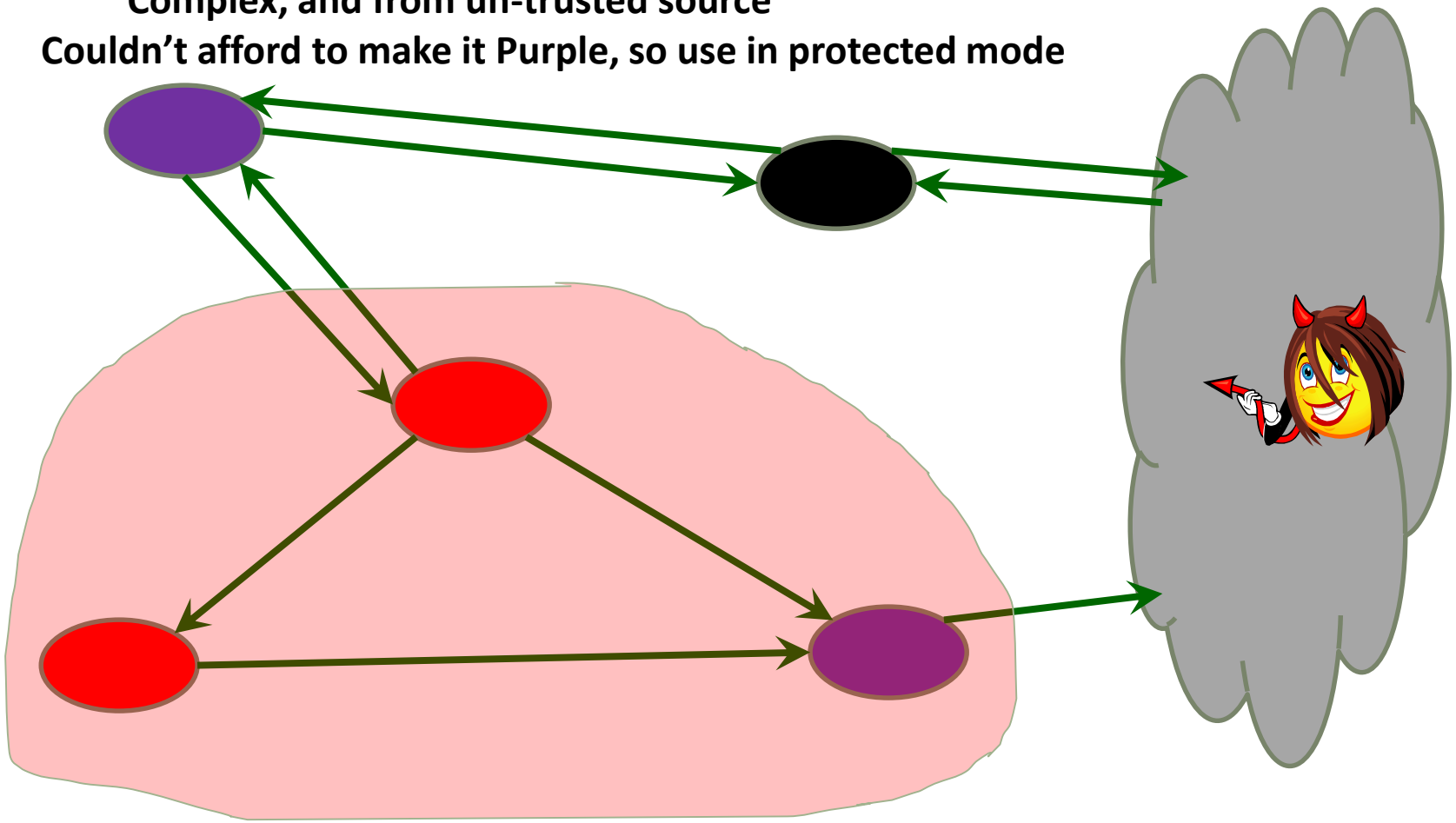
Black: Too big and co0mplex. From un-trusted source
Critical system policy does not depend on it



Placing / Selecting the Purple for Red / Black

System Design for Security

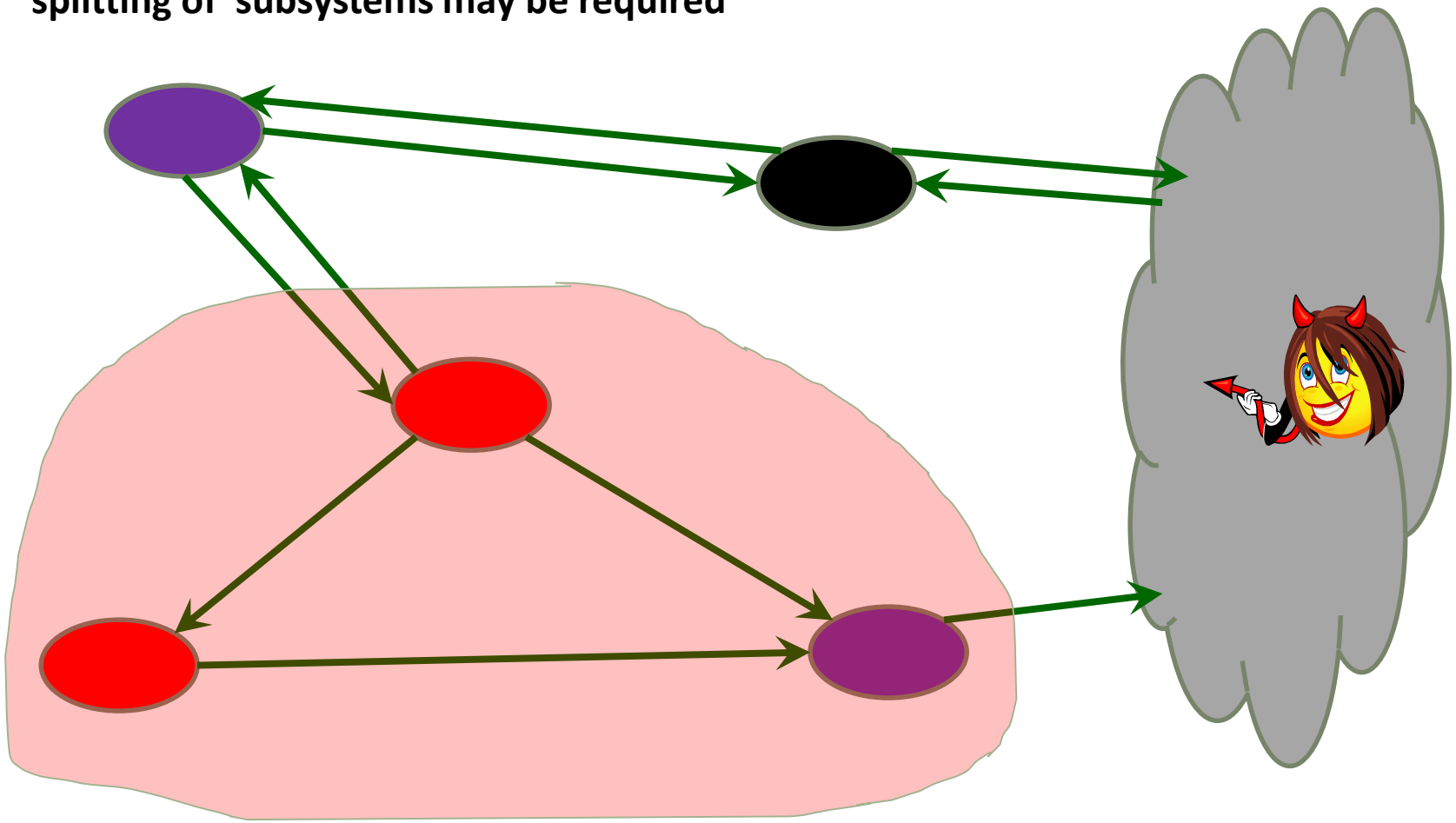
Red: Needed for critical functions and some policies
Complex, and from un-trusted source
Couldn't afford to make it Purple, so use in protected mode



Placing / Selecting the Purple for Red / Black

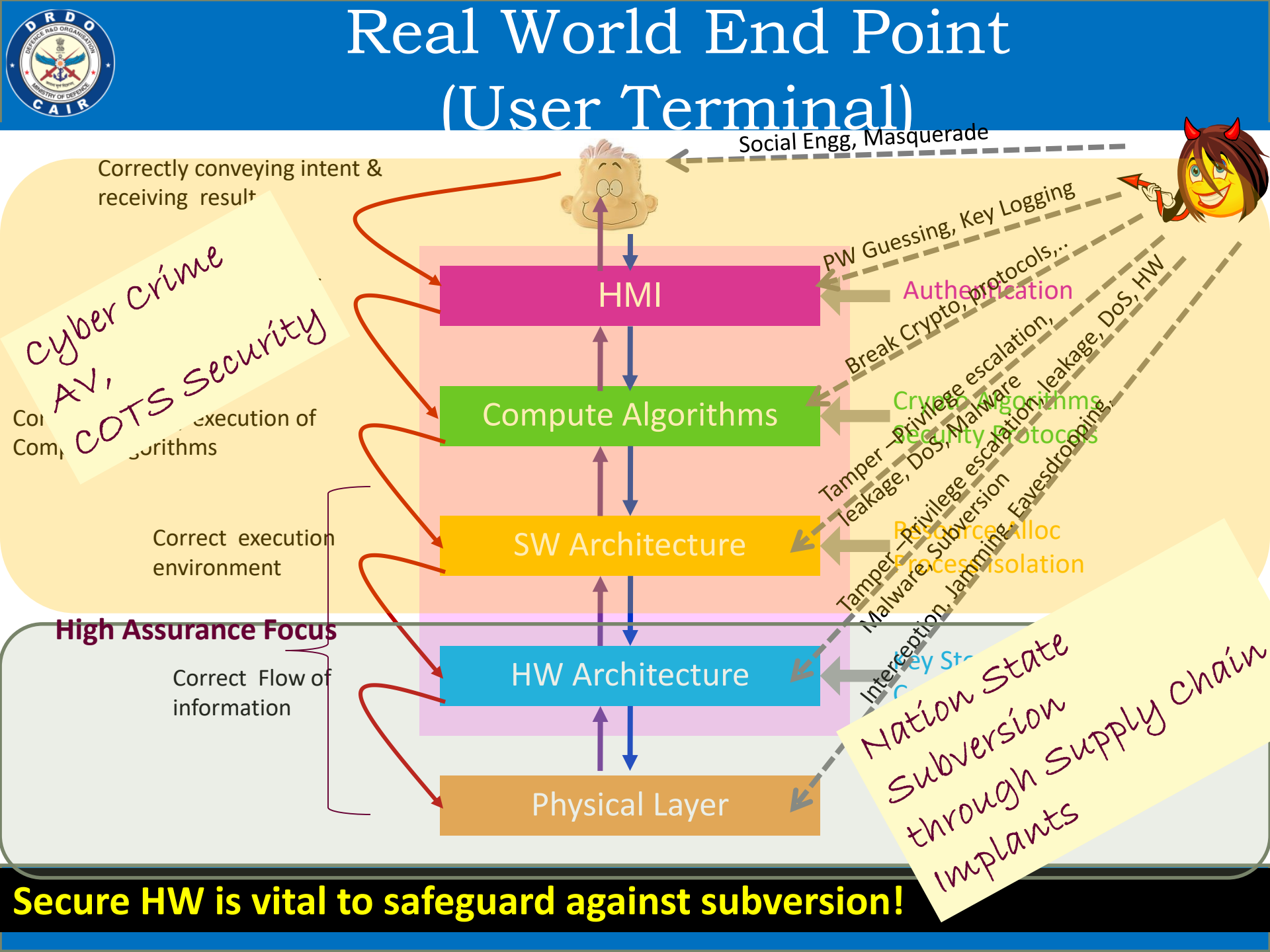
System Design for Security

Some re-allocation of functions, elimination of paths, splitting of subsystems may be required



Security analysis and top level design to be done early

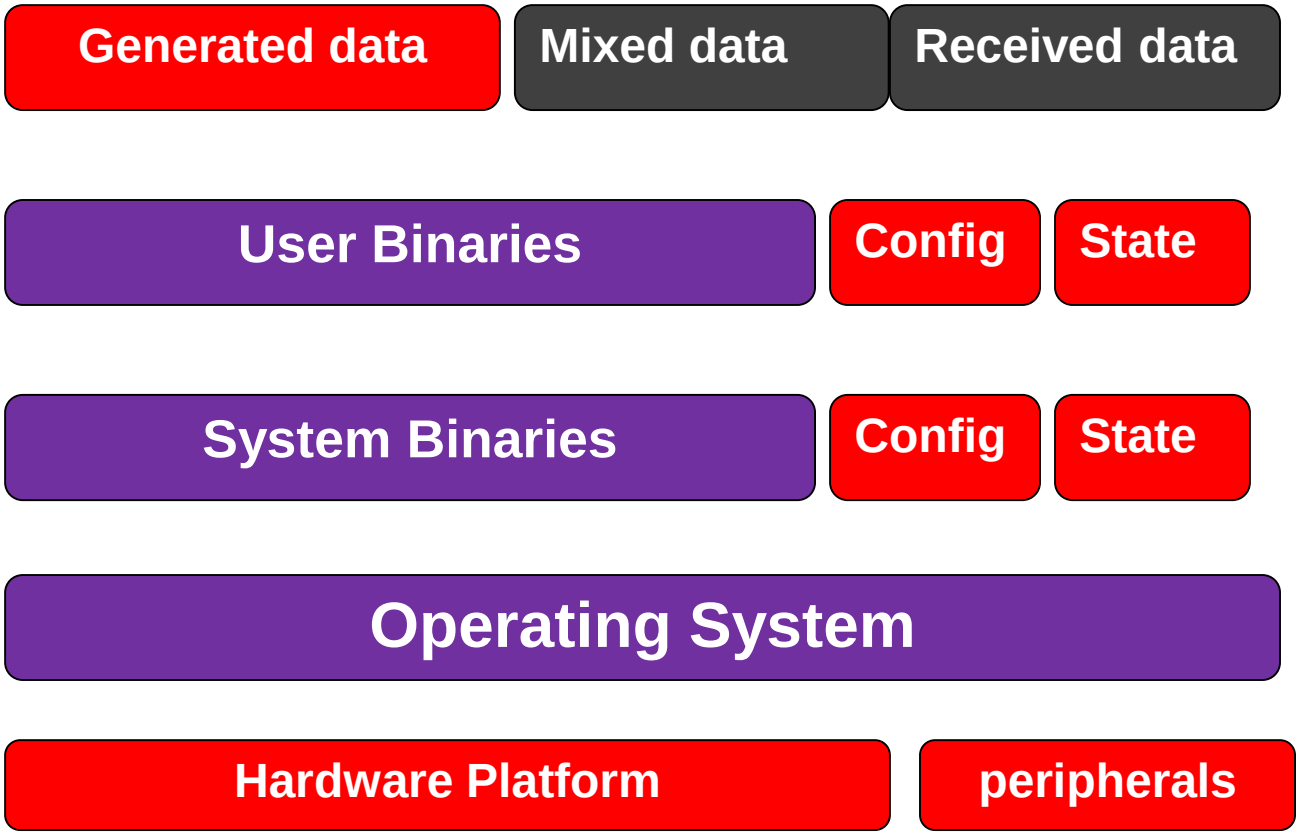
Real World End Point (User Terminal)



Secure HW is vital to safeguard against subversion!

Computation - 1

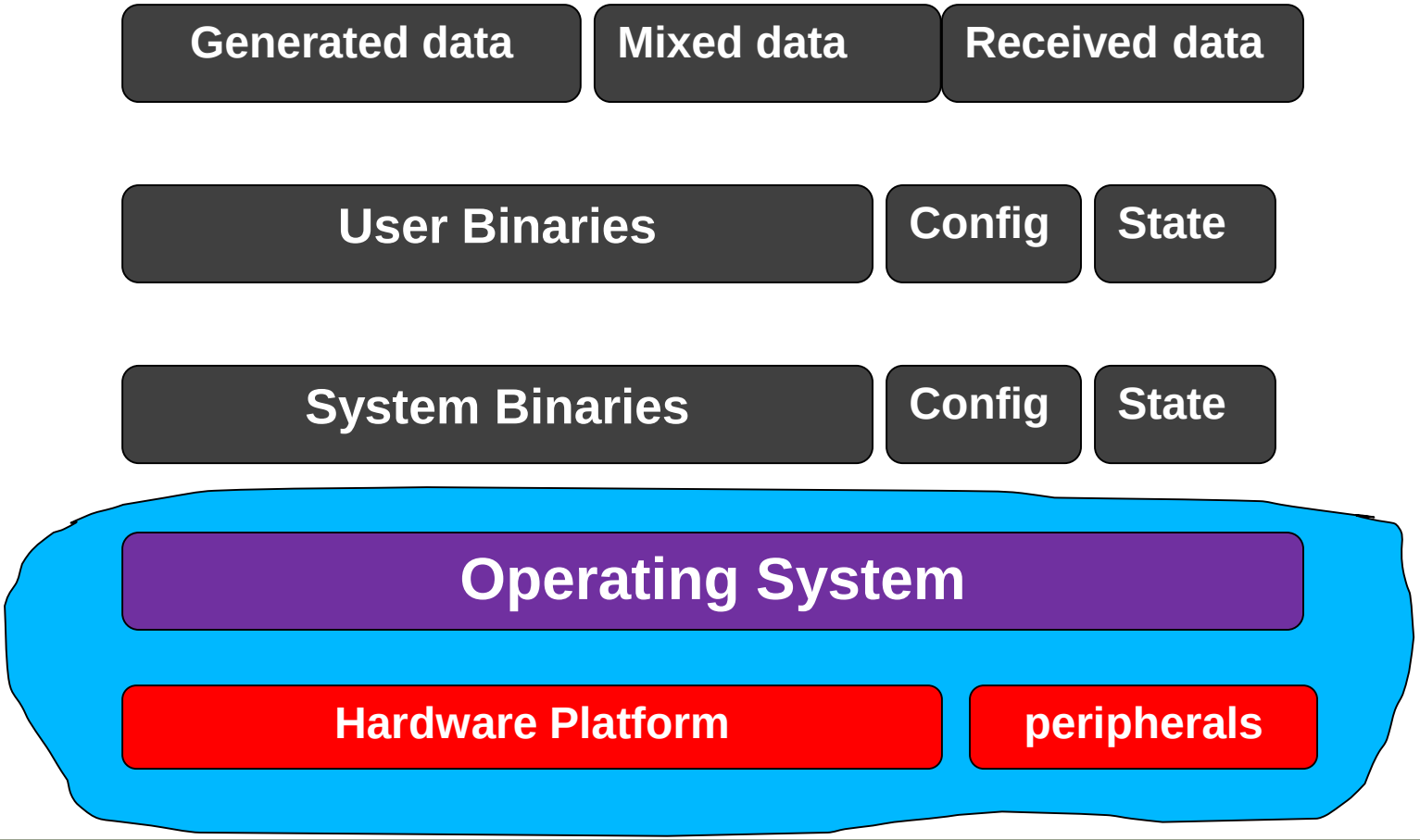
IDEAL STATE: SAFE EXCHANGE OF DATA POSSIBLE



Would require ideal ‘correct’ binaries and OS: perfect isolation of executables from input data

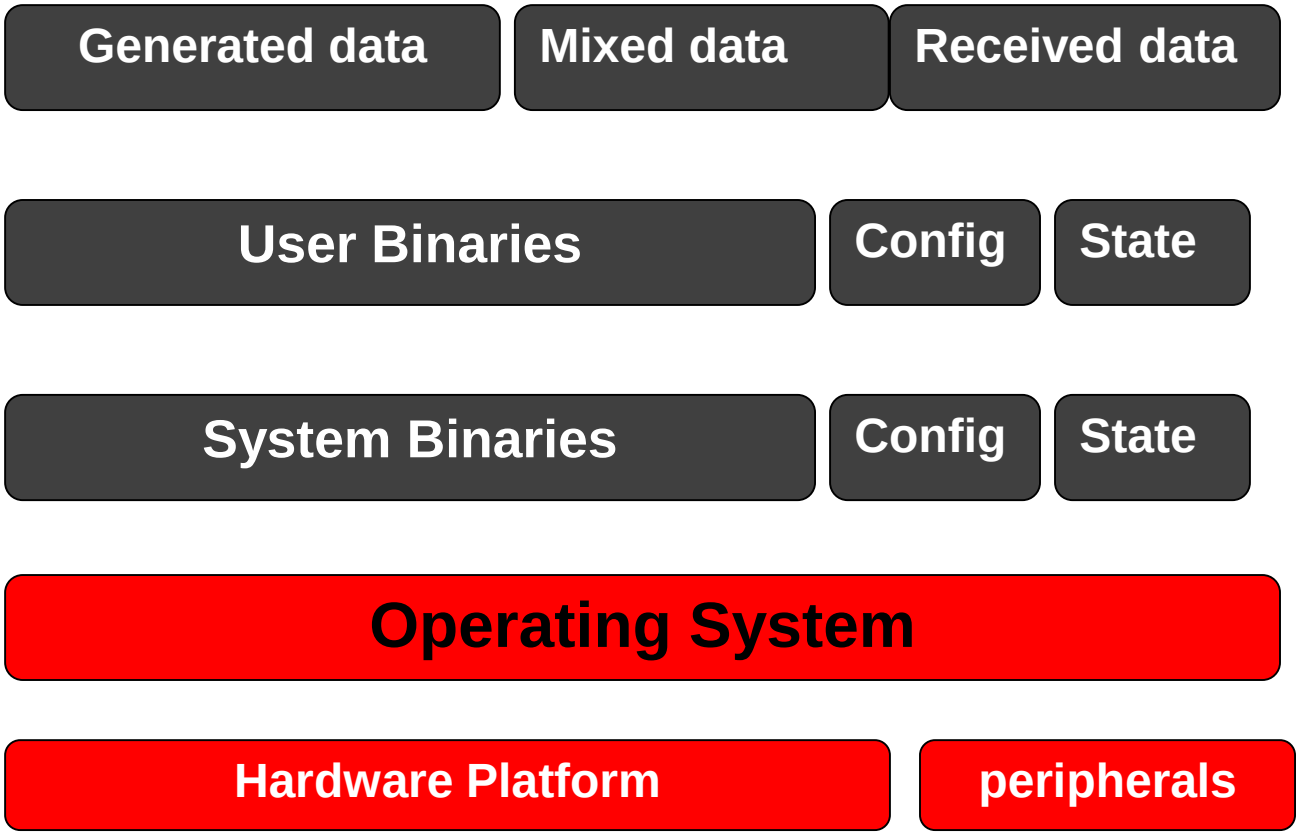
Computation - 2

Users composition of binaries by installation from media or network



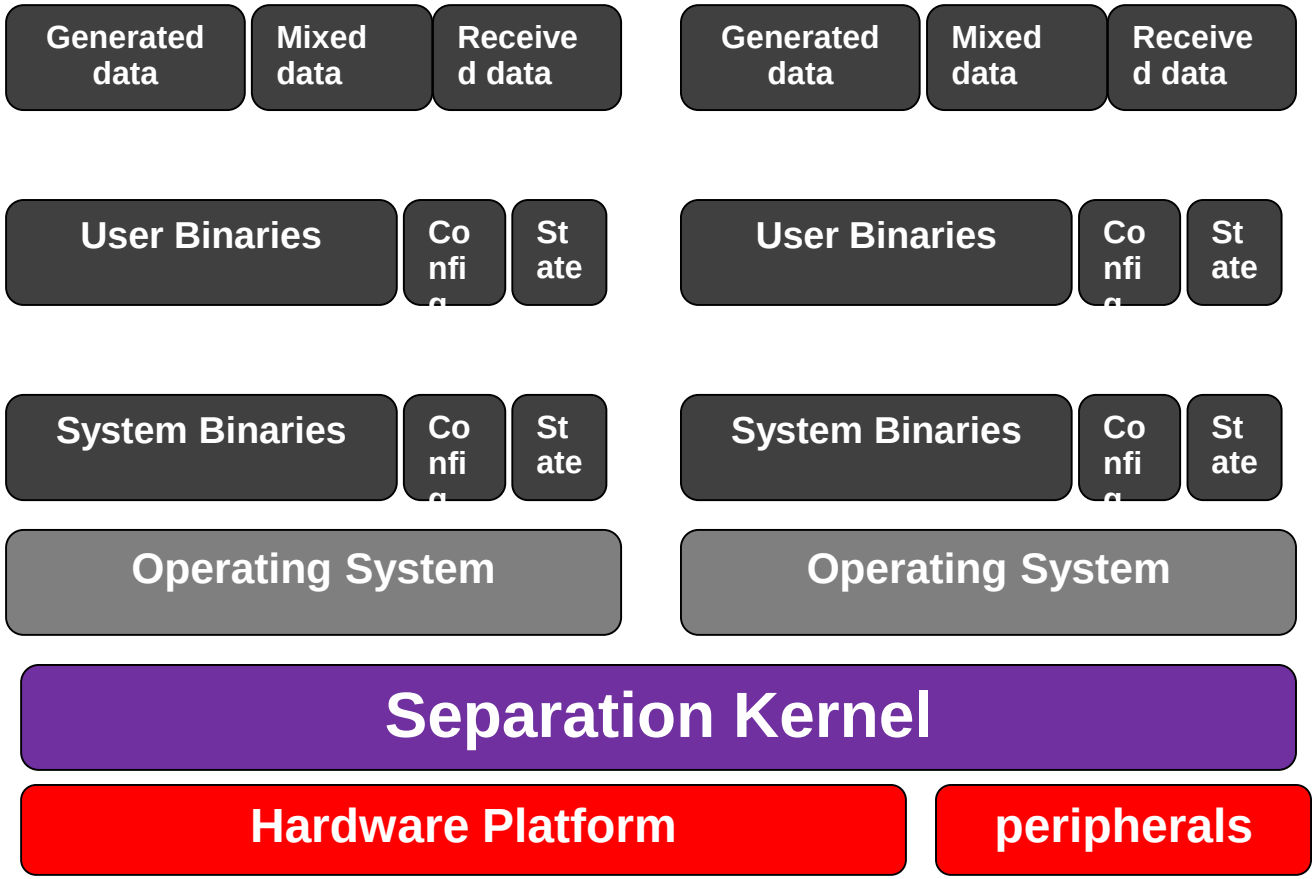
Coarse level policies only. Availability of services suspect.

Component View for Analysis



Mainstream OS become too complex to be security-bug-free

Computation - 3



Policies only wrt separation and confinement

Platform Integrity levels

Level	Description
1	Regular COTS OS on COTS HW in default configurations
2.	Regular COTS OS on COTS HW with OS hardening based on configuration
3.	Regular COTS OS on COTS HW with OS hardening and Security Applications
4	*Creation of Secure Execution Environment (SEE) to ensure immutability and security policy enforcement at OS, HAL and Application levels on COTS HW*
5	*Creation of Secure Execution Environment to ensure immutability and security policy enforcement at OS, HAL & Appl levels on HW augmented to provide roots of trust (evaluate-able TCB) and support to SEE*
6	SEE and HW designed for high assurance rather than functionality or run-time re-configurability

The Demise of 'Secure' OS

Hardware abstraction for applications

- Open set: Plug and play, active devices, wireless communications

Common system services for easy application development

- High diversity, constant evolution

Resource arbitration between processes

- Concurrency, statefulness, time & resource slicing

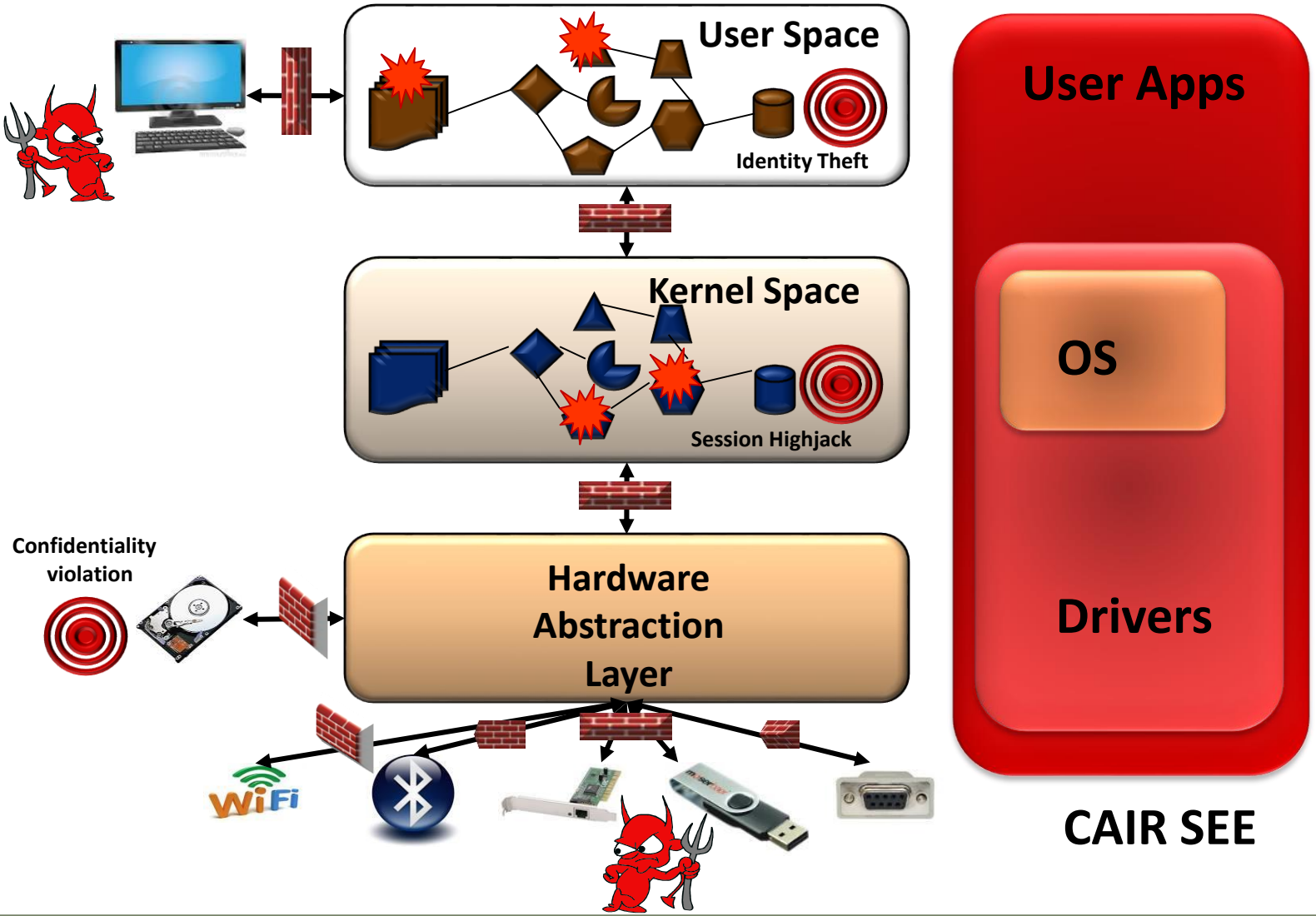
Policy enforcer between multiple processes and between multiple users

- Too many loopholes, too many fixes

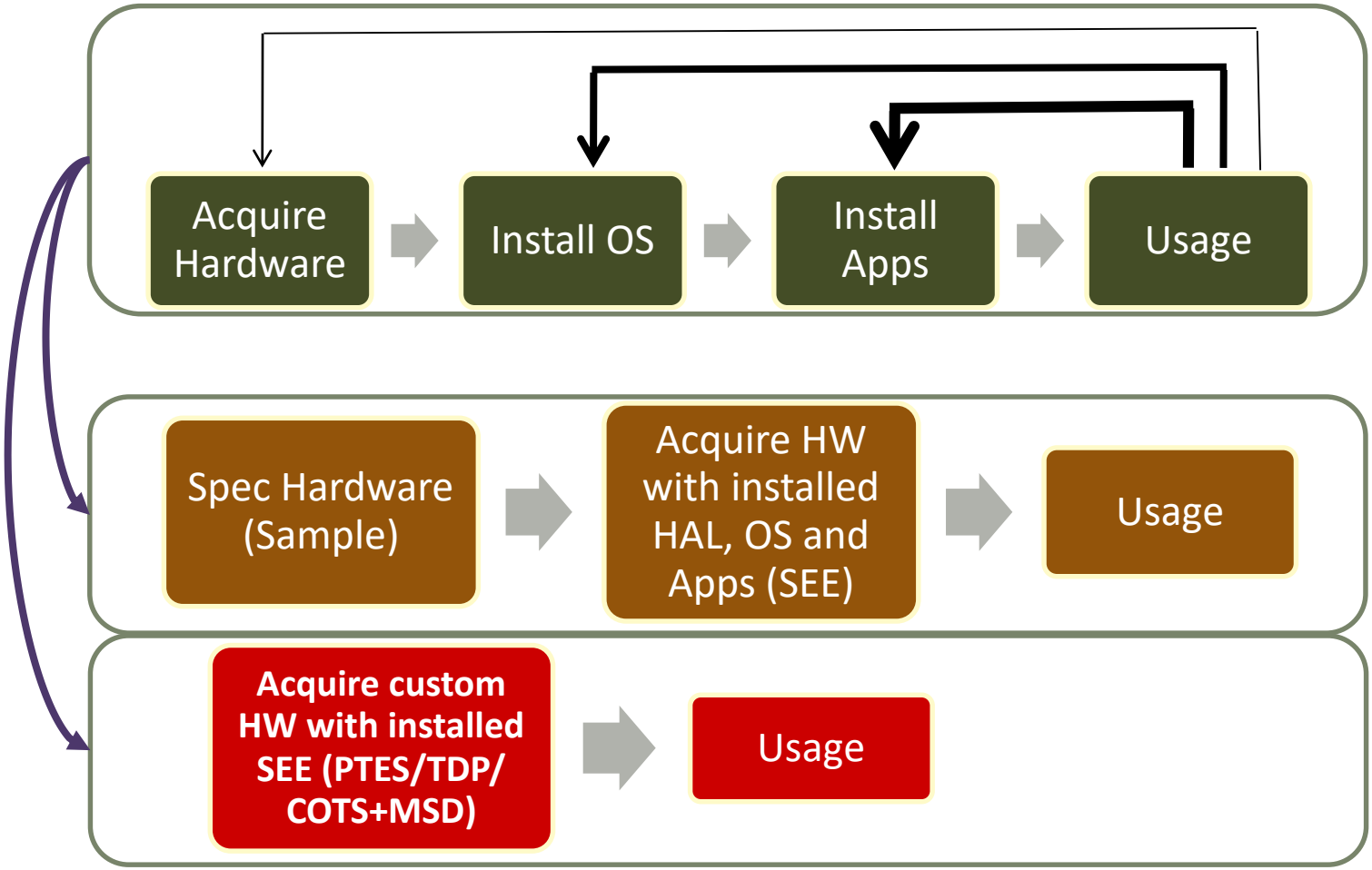
The Evolving ICT Security State

Usage Model	The Red	The Black	Security
Stand Alone, Dedicated, Scientific	H/W, OS, Applications, Data	Humans	(Physical) access
Stand Alone, Document sharing	H/W, OS, Applications	Data	Applications, OS to separate data from execution
User installs applications	H/W, OS,	Data, Appilcations	OS to enforce control of appls
User installs, upgrades OS	H/W, Hypervisor	OS, Data and Applications	Contain the OS. HW protects hyp.

Providing SEE to End-User



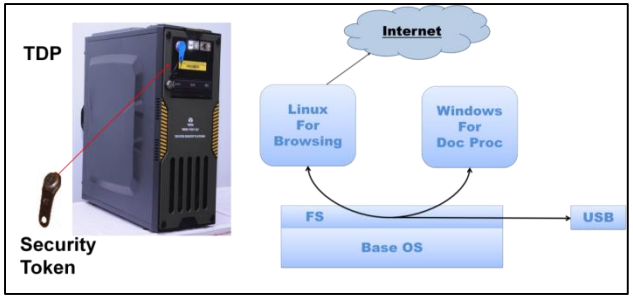
Design, Acquisition and Usage Model



For high assurance against capable adversaries
For deployments on homogenous platforms with fixed functionality

High Assurance Solutions

Desktop



Windows + MS Office: For user functionality

Virtual Machine Manager: Interface to Windows

CAIR's Secure Execution Environment(SEE): Protects

COTS computer hardware for easy deployment

Logging of read/write media access

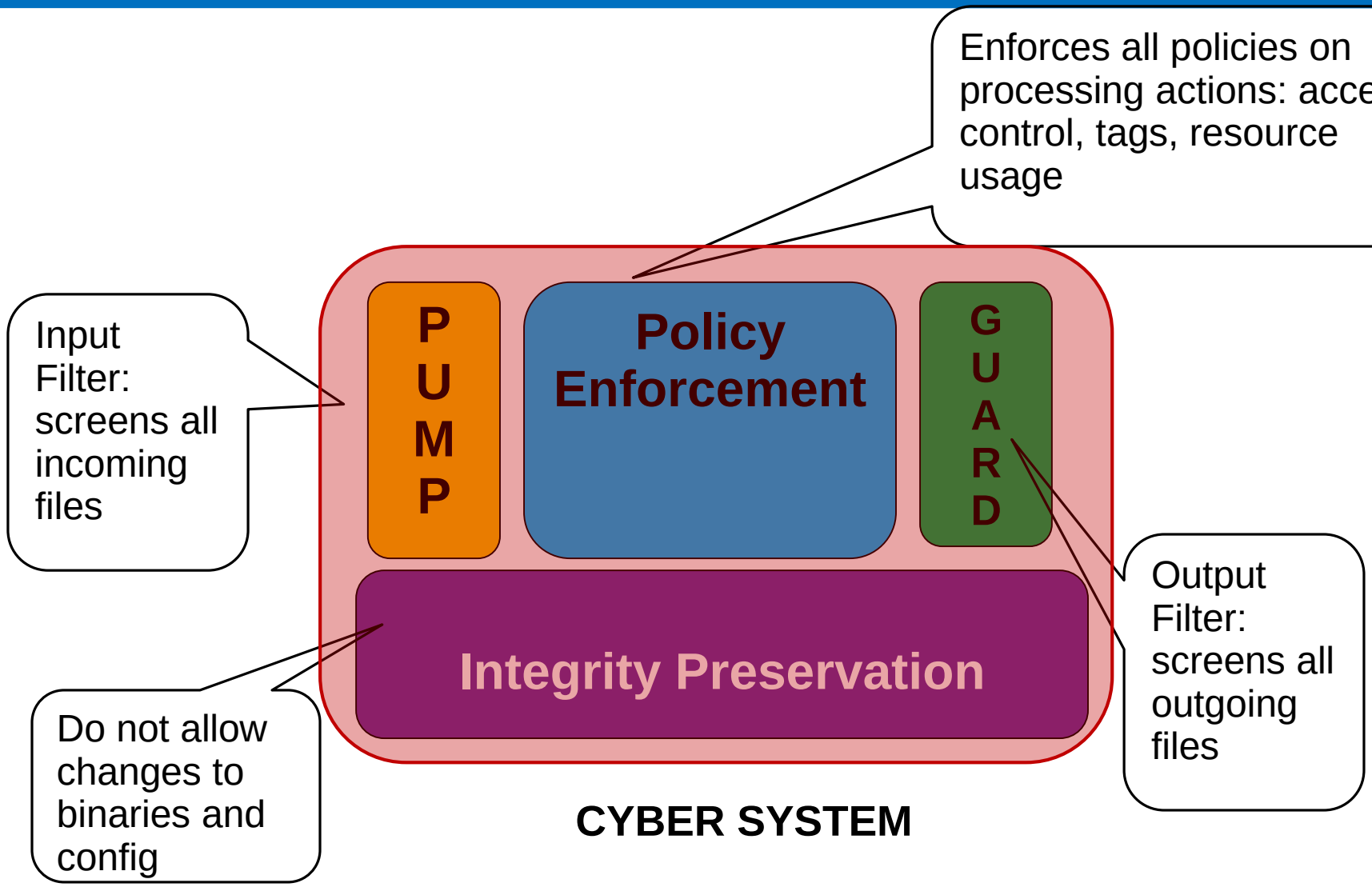
Removable media (USB/CD/DVD)

SDPS-CS

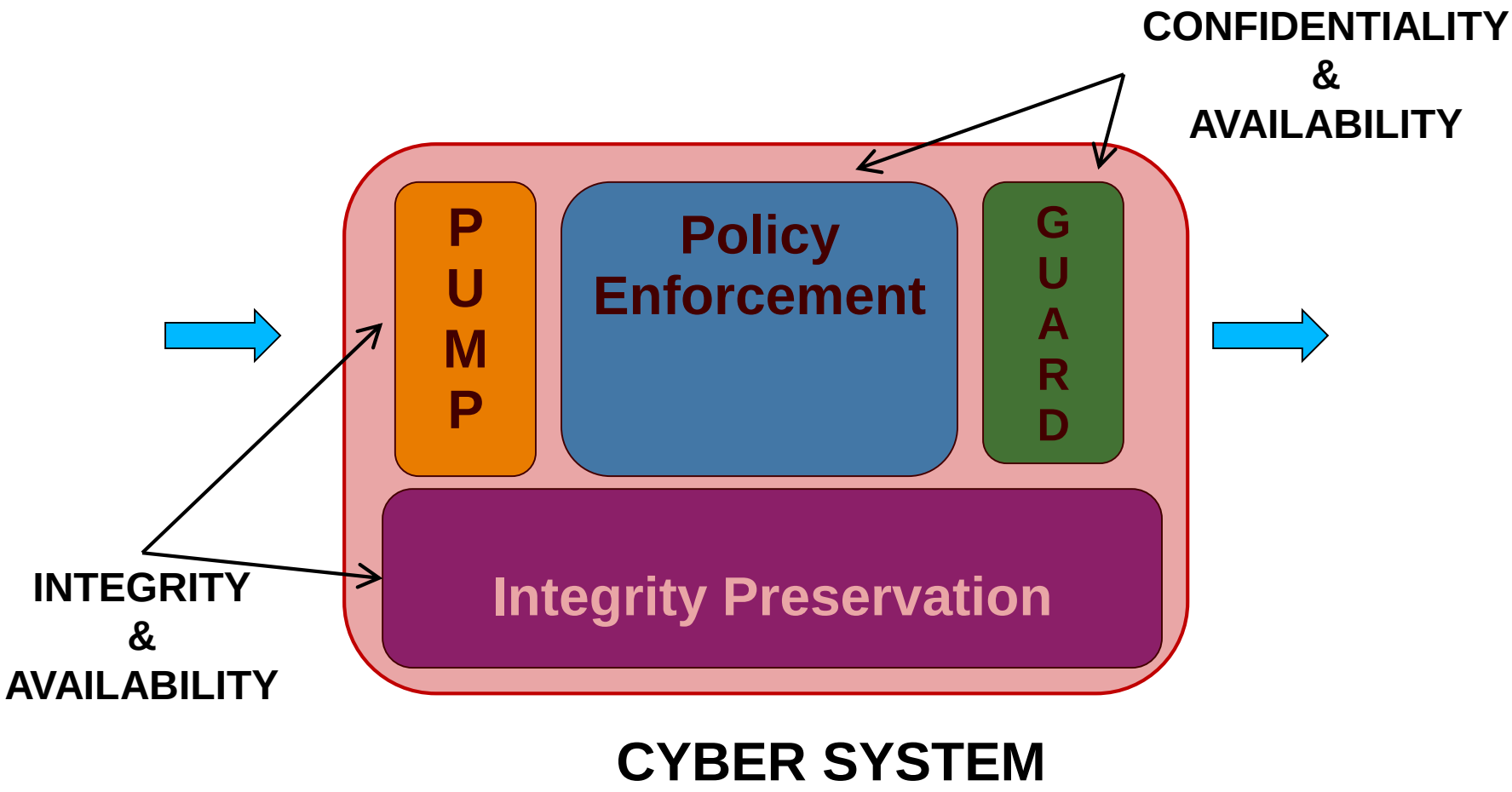
Terminal

Secure NIC

The Canonical View



Cyber Utility vs Cyber Sec



IDENTITY & PRIVACY

1. Identity

a. Must be unique across the domain

b. Must be public (no need for secrecy)

c. Must be (correctly) verifiable

Else, need for authentication using secrets

2. Privacy

a. Right to withhold information

b. Not a ‘Right to Lie’

c. Cannot be used to control others

d. Must be paid for, in opportunity costs

Brokering the Peace

OK, Boomer

OK, Google

???????