



# AdSec: A System for Adaptive Network Security

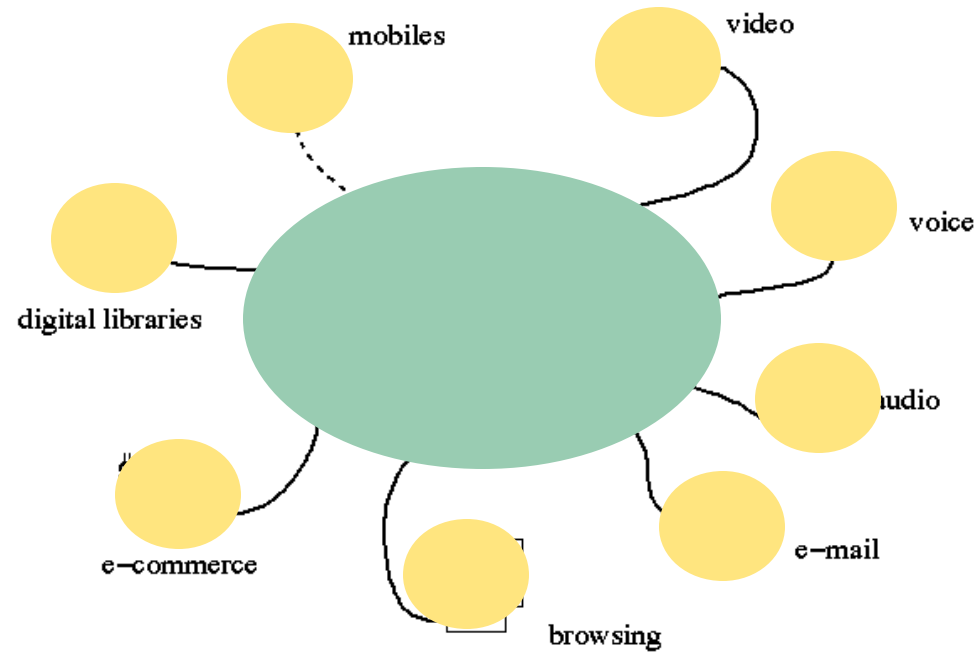
Kihong Park  
Network Systems Lab &  
CERIAS  
Dept. of Computer Sciences  
Purdue University  
park@cs.purdue.edu

Joint: E. Spafford (co-PI), M. Tripunitara, G. Nalawade

Network Systems Lab & CERIAS

# Motivation

- ◆ Network environment → Secure QoS





## Motivation (cont.)

Goal Facilitate *Secure QoS*

- ◆ User plane protection
- ◆ Control plane protection
- ◆ Minimal footprint



# Technical Challenges

- ◆ Integrate QoS & security architectures
- ◆ Minimize security footprint
- ◆ Fault-tolerance
- ◆ Programmability
- ◆ Interoperability



# Approach

- ◆ User plane protection: end-to-end
  - Security services: confidentiality, integrity, authentication, access control
  - Property of cryptographic protocols

→ well-understood



## Approach (cont.)

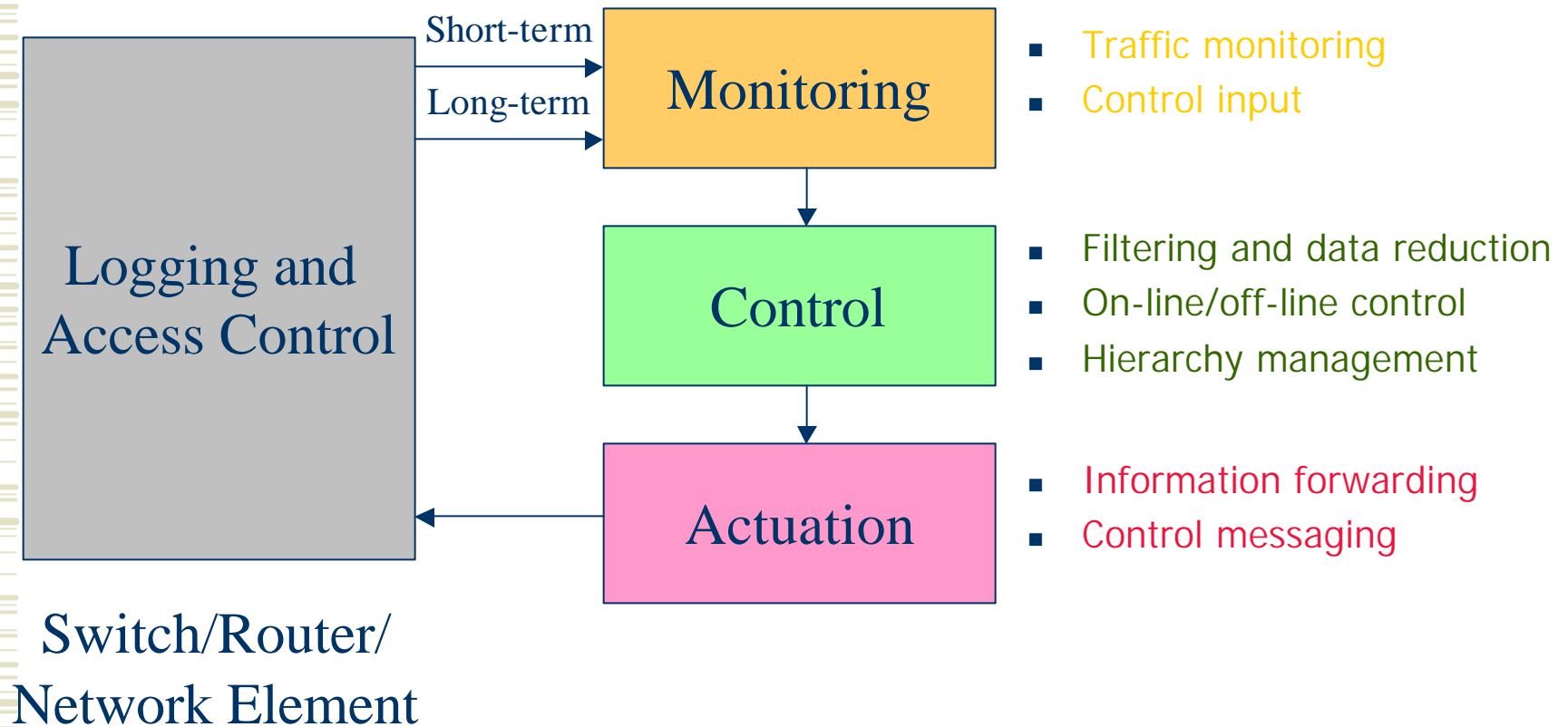
- ◆ Control plane protection
  - Proactive: authentication/integrity of certain signalling
  - Reactive:
    - too costly to make 100% proactive (e.g., Gbps switching)
    - security-QoS trade-off
    - selective, controlled introduction of “security holes”
    - reactive management



## Approach (cont.)

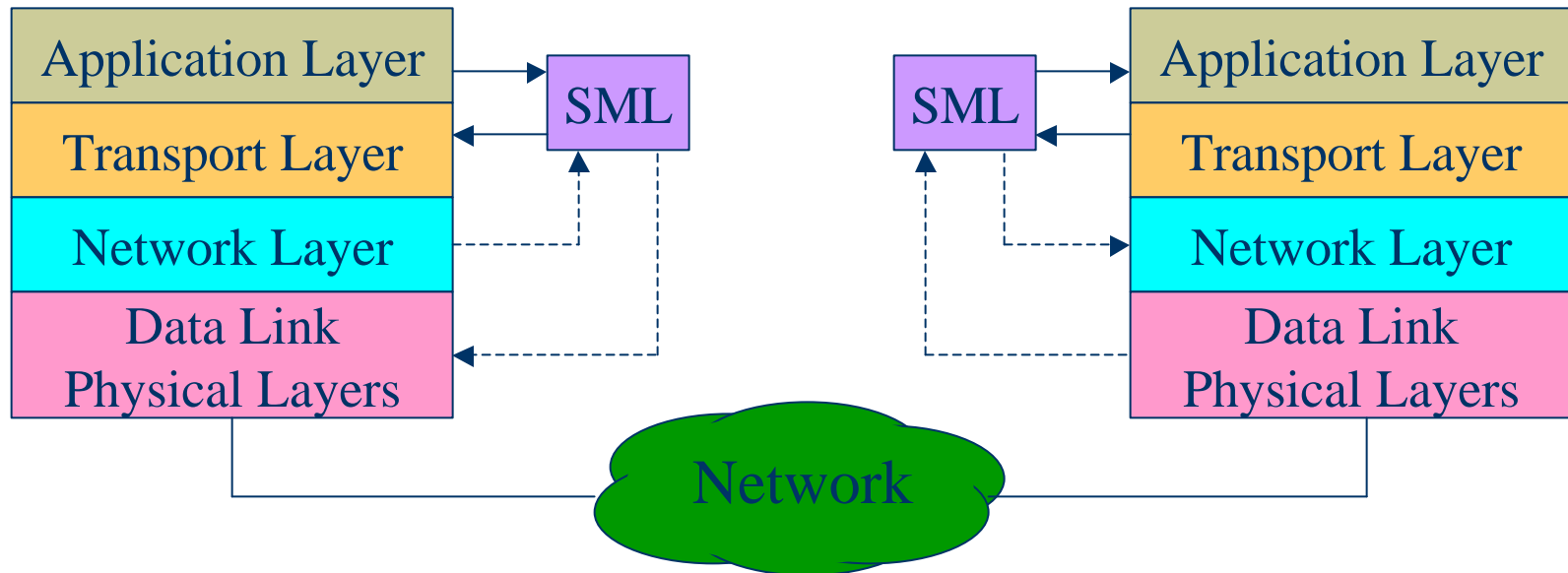
- ◆ Adaptive security
  - Combination of proactive/reactive rendering of security services
  - Multiple levels of service
  - Monitoring, control, and actuation
  - Active resource management: control/scheduling
  - Deterrent effect

# Adaptive Security Architecture

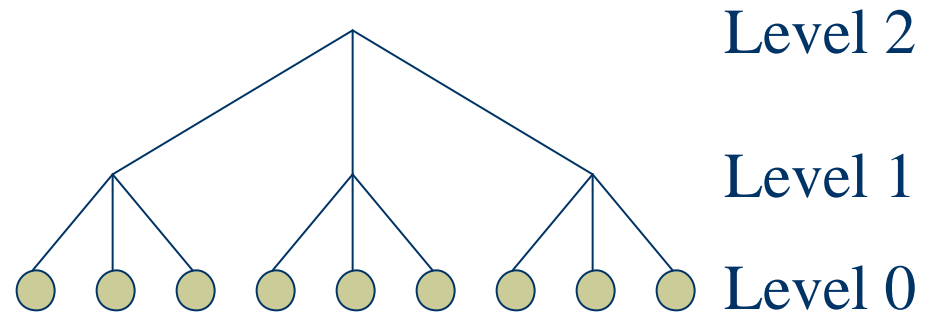
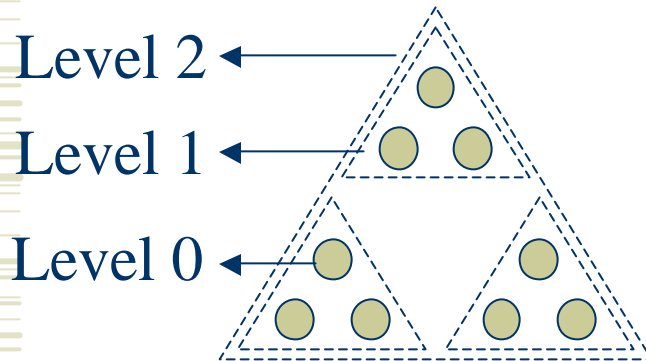




# Security Middleware Layer



# Monitoring & Control Hierarchy



- ◆ Hierarchical structure for resiliency and scalability
- ◆ Byzantine agreement protocols (2/3 majority) for isolating compromised nodes



# Overview of AdSec

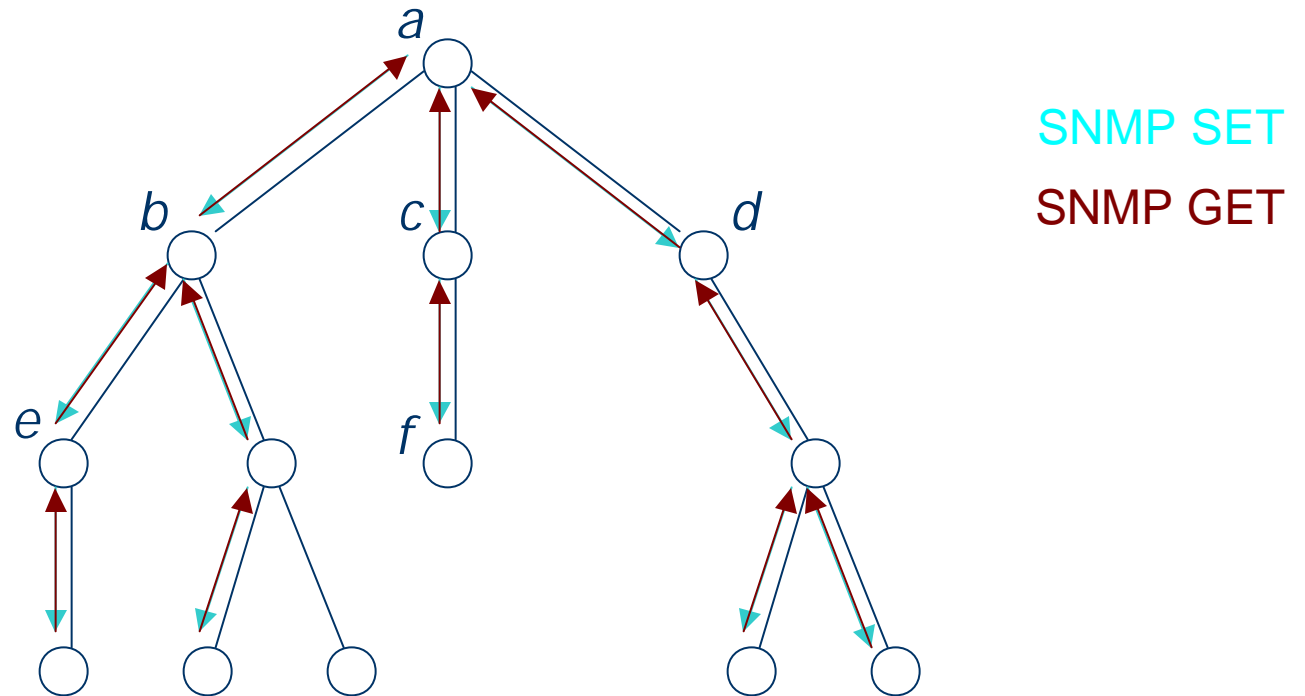
Function Programmable network management system  
for adaptive network security

- ◆ Implementation of any logical hierarchy
- ◆ Monitoring, control, and actuation
- ◆ SNMP based
- ◆ User programmable

# AdSec: Components

- ◆ MIB Dependency Relation (MDR) → **config.**
- ◆ Managers → **snmpd**
- ◆ Agents
- ◆ Monitor function → **wrapper**
- ◆ Control function → **wrapper**
- ◆ System libraries

# MIB Dependency Relation



# Monitor & Control Functions

- ◆ Monitor function  $F$  : **SNMP GET**
- ◆ Control function  $G$  : **SNMP SET**

→ user programmable

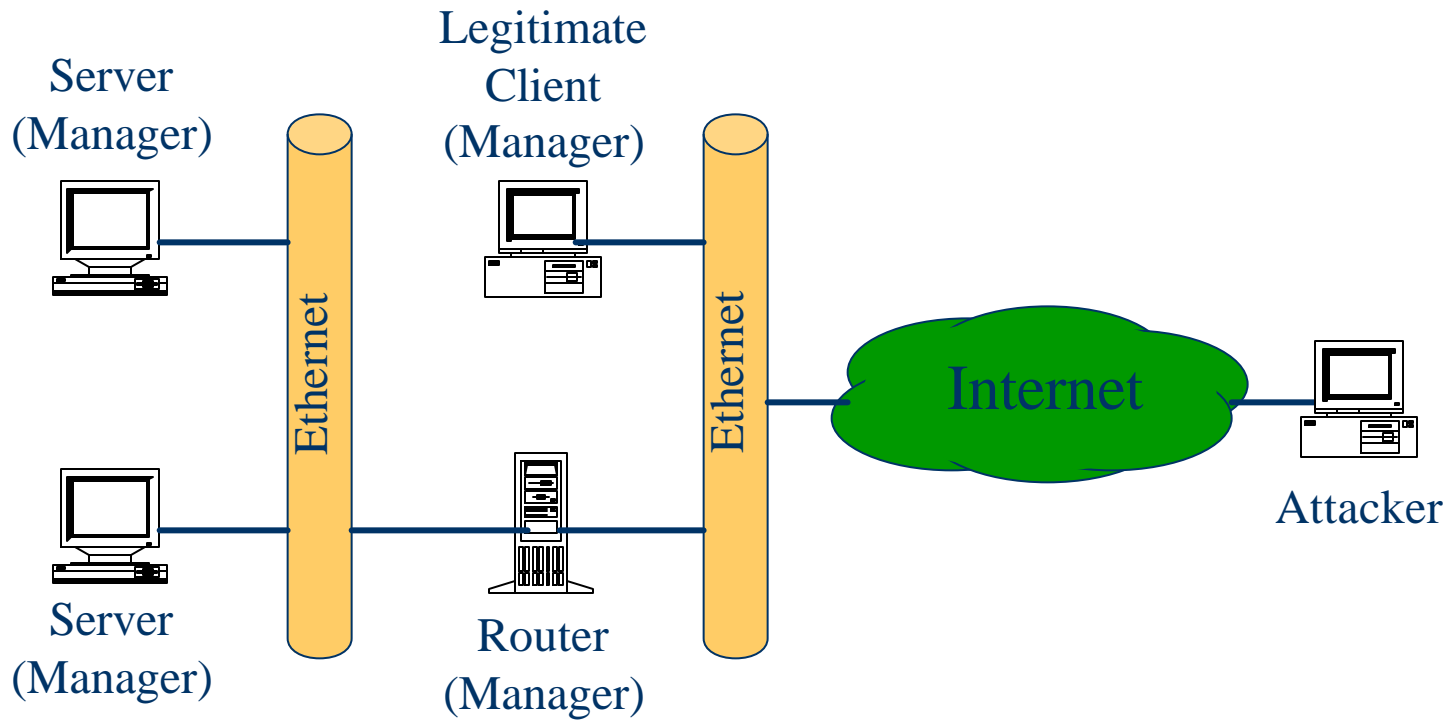
- ◆ Function wrappers
- ◆ Transparent read/write
  - local vs. remote MIB
  - $\text{OID} \leftrightarrow (\text{IP}, \text{Port})$  → physical mapping



# AdSec: Implementation Details

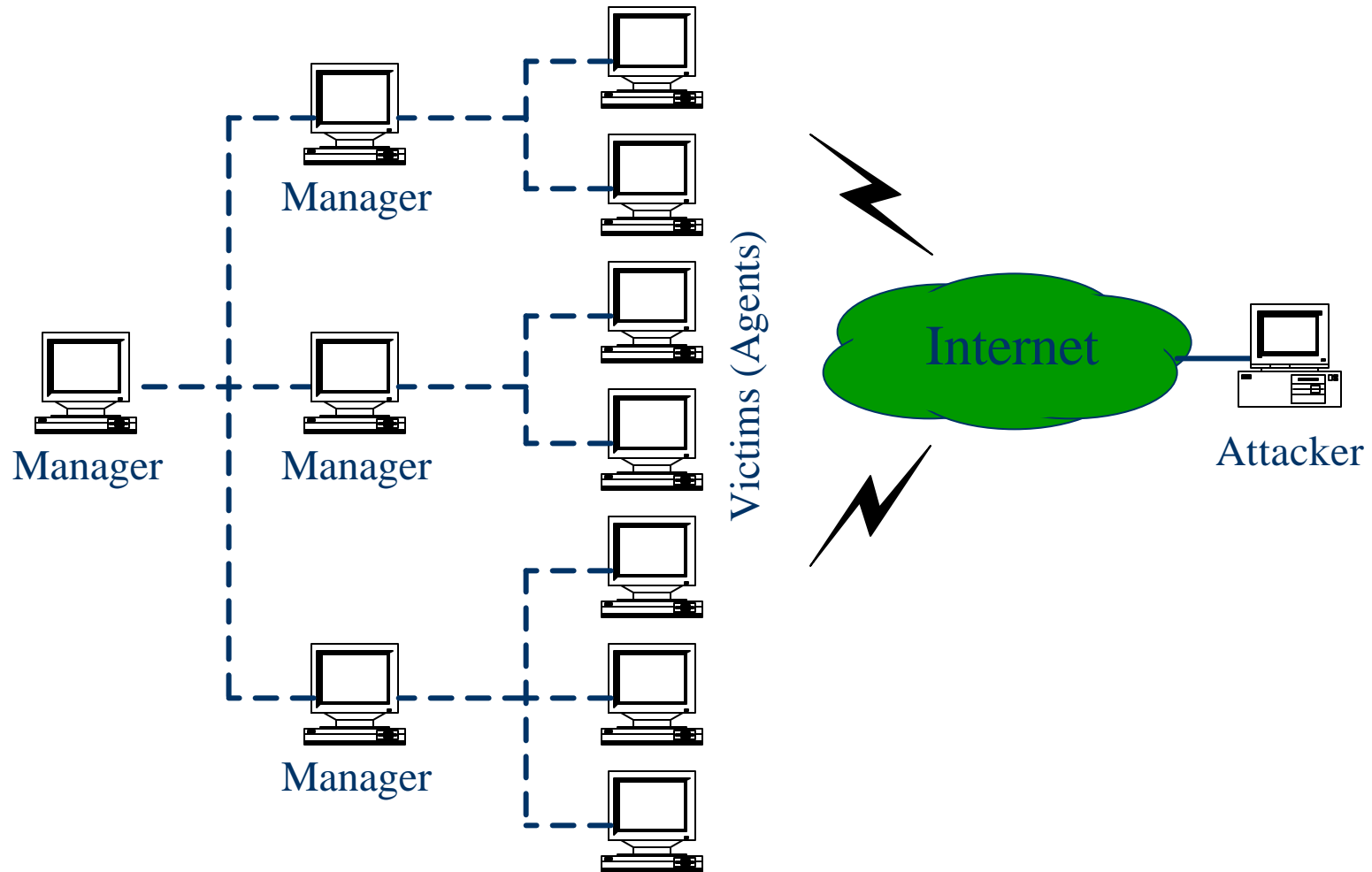
- ◆ Prototype system SNMP version 1
- ◆ UC Davis snmpd implementation
  - Others (e.g., Sun Microsystems) possible
- ◆ CMU SNMP library
- ◆ Functions:
  - Intrusion detection
  - Anomaly detection
  - Automatic control

# Benchmark: Set-up A





# Benchmark: Set-up B





## On-going & Future Work

- ◆ Asynchronous extension: SNMP Trap
- ◆ Extensive monitor/control function libraries
- ◆ Integration with network management
- ◆ Active resource scheduling
- ◆ Integration with QoS management