CFCC: Covert Flows Confinement For VM Coalitions

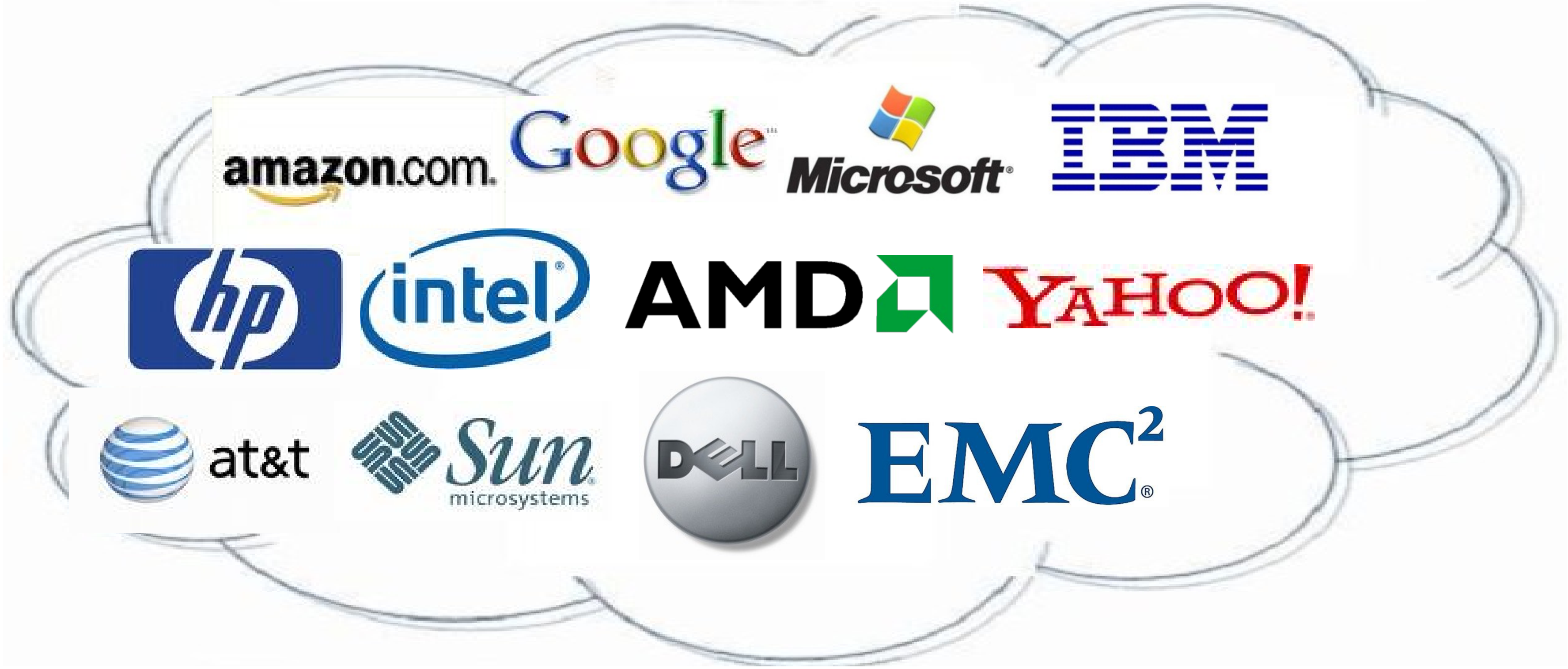
Ge Cheng, Hai Jin, Deqing Zou, Lei Shi, and Alex K. Ohoussou

服务计算技术与系统教育部重点实验室（SCTS）
集群与网格计算湖北省重点实验室（CGCL）

Outline

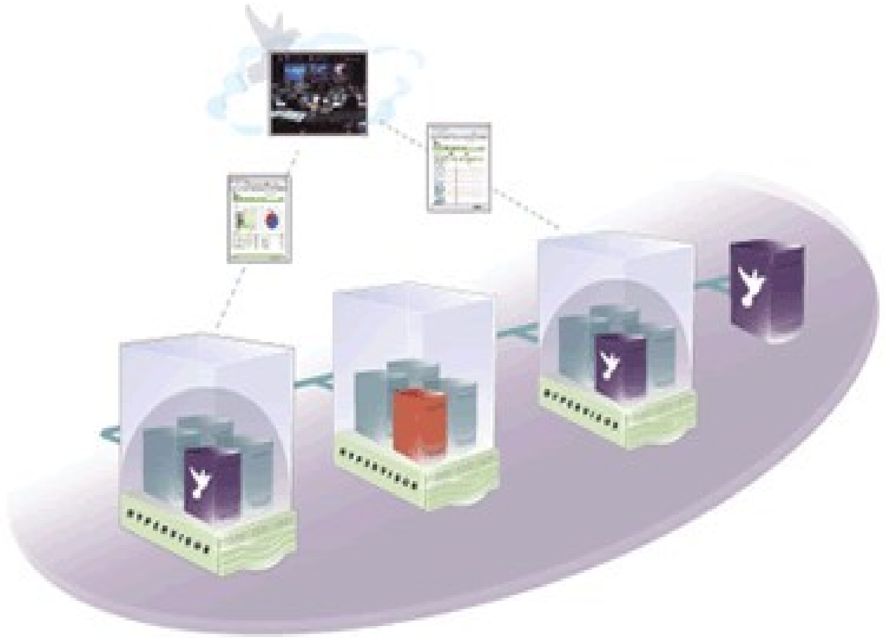
- Background
  - Cloud and Virtualization
  - Problems Statement
- Design
  - Requirement
  - Architecture
  - Algorithm
- Implementation and Experiment
  - Performance
- Conclusion and Further work

Background(Cloud and Virtualization)



Cloud computing currently emerges as a hot topic due to its abilities to enable companies to cut costs by outsourcing computations on-demand

Background(Cloud and Virtualization)

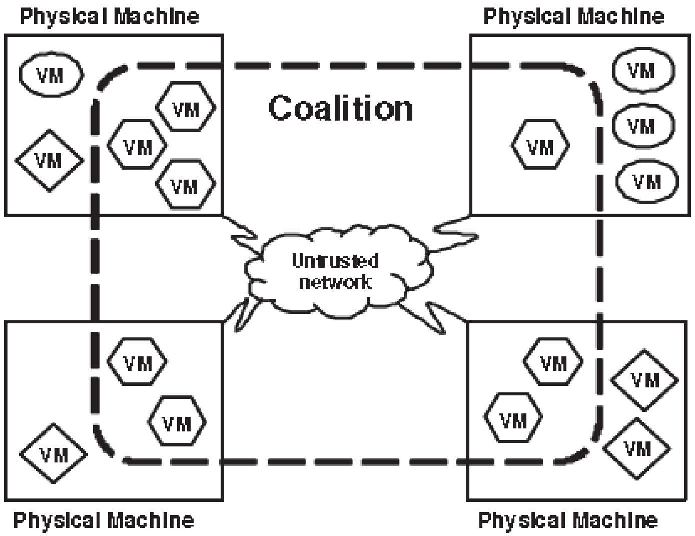


- Many cloud provider take Virtualization technology as the infrastructure, such as Elastic Compute Cloud of Amazon, Blue Cloud of IBM.
- So it is natural that resources in those cloud computing environment are allocated in VM granularity for cloud users.

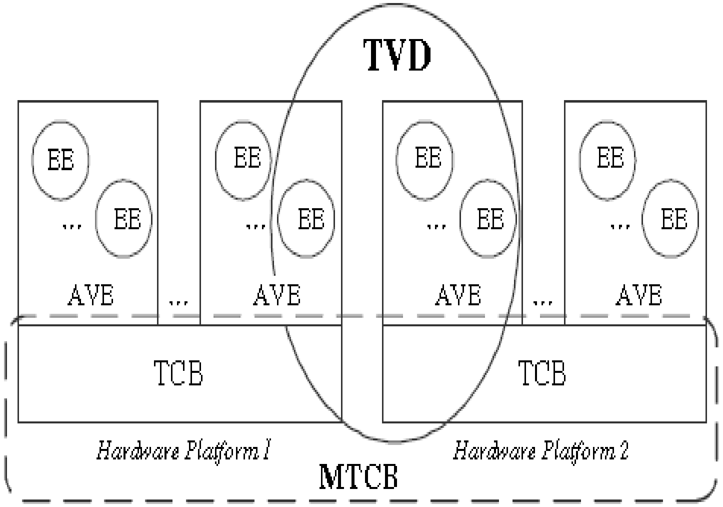
Background(Problem Statement)

- Although multiple VMs on the same hardware platform offer great benefits, it also raises the risk of information leakage between VMs belonging to different companies which may compete with each other.
- Enforcing MAC between VMs provides an attractive mechanism to improve the security of VM based cloud computing. Dynamic coalitions, also called domains in some papers, are used to organize VMs over nodes, and security policies differ in each coalition normally.

Background(Problem Statement)



Shamon



Trusted Virtual Domain

- **There are many VM coalition building approaches, which have been proposed in distributed VM systems, such as NetTop, Shamon, and Trusted Virtual Domain.**

Background(Problem Statement)

- However the existing VM coalition systems cannot eliminate covert channel, which are not the mechanism designed for implicitly communication controlling between VMs. For example, if both two VMs have the access to a disk, they may use it as a covert channel by controlling the exhaustion of the disk’s storage space.
- Although overt communication channels are enforced by explicit authorizations and we have some tools to check comprehensive coverage of authorizations to these channels, covert channels are difficult to identify and perhaps impossible to eliminate completely.

Background(Problem Statement)

- To address the above questions, we propose a covert flows confinement mechanism for VM coalitions (CFCC) in VM-based cloud computing.
- CFCC uses an effective but simplified alternative of the prioritized Chinese-Wall model[1], with a mandatory access control mechanism for all communication, migration, startup of VMs without changing current MAC policies inside the coalitions.
- Enforcing MAC to managing the covert flows by CFCC is not to eliminate covert channels by rewriting of hypervisor code but
  - (i) to prevent the covert flow through careful resource management.
  - (ii)to enable users through configuration options to mitigate covert channels

[1]Cheng, G., Jin, H., Zhou, D., Ohoussou, A.K., Zhao, F.: A Prioritized Chinese Wall Model for Managing the Covert Information Flows in Virtual Machine Systems. In: 9th International Conference for Young Computer Scientists, pp. 1481--1487. IEEE Press, Hunan (2008)

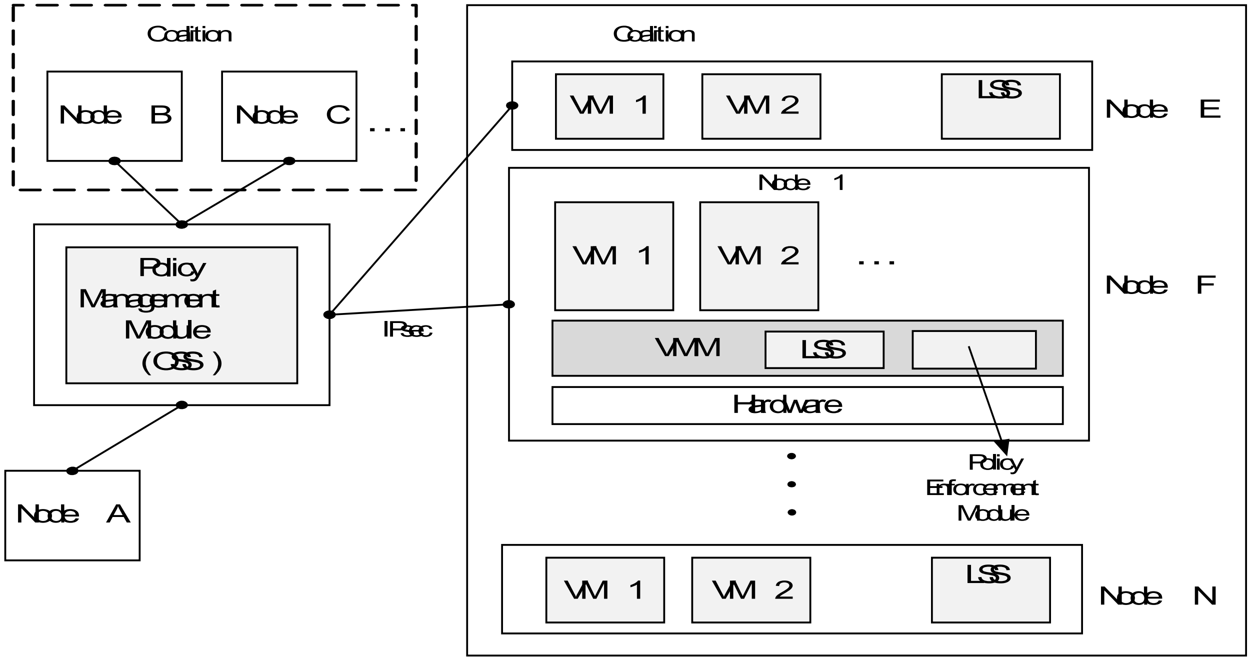
Design(Design Requirement)

- We use the conflicts of interest set of Chinese-Wall policy to describe the requirement of covert flows confinement between two VMs. The coalitions will be dynamically constructed. Both the subjects and objects of the Chinese-Wall policies used in our mechanism are VMs. A label defined by the system administrator is attached to a VM, and the following information flows between label-attached VMs will be controlled.
  - 1) covert information flows between label-attached VMs whose labels are the same are permitted;
  - 2) covert information flows between label-attached VMs whose labels belong to different conflicts of interest set are permitted;
  - 3) covert information flows between label-free VMs are permitted;
  - 4) covert information flows between label-attached VMs whose labels belong to the same conflicts of interest set are disallowed.

Design(Design Requirement)

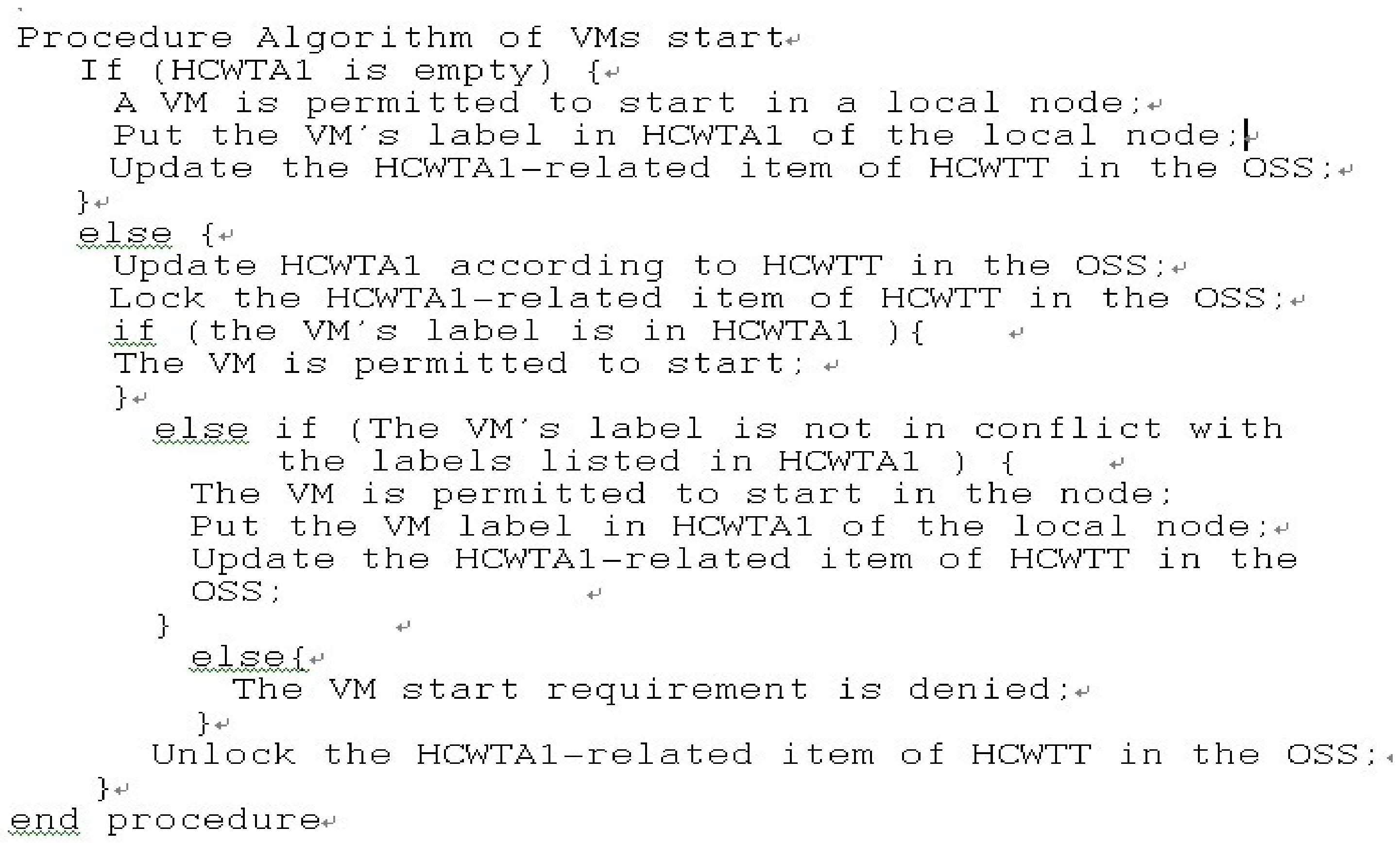
- The Chinese-Wall model is history-based, which needs to have the knowledge of the current system state to make decisions.
- Two features are needed in our architecture: distributed mandatory access control for all VMs and centralized information exchange. Both need to be implemented simultaneously based on the activity history of VMs.

Design(Architecture)



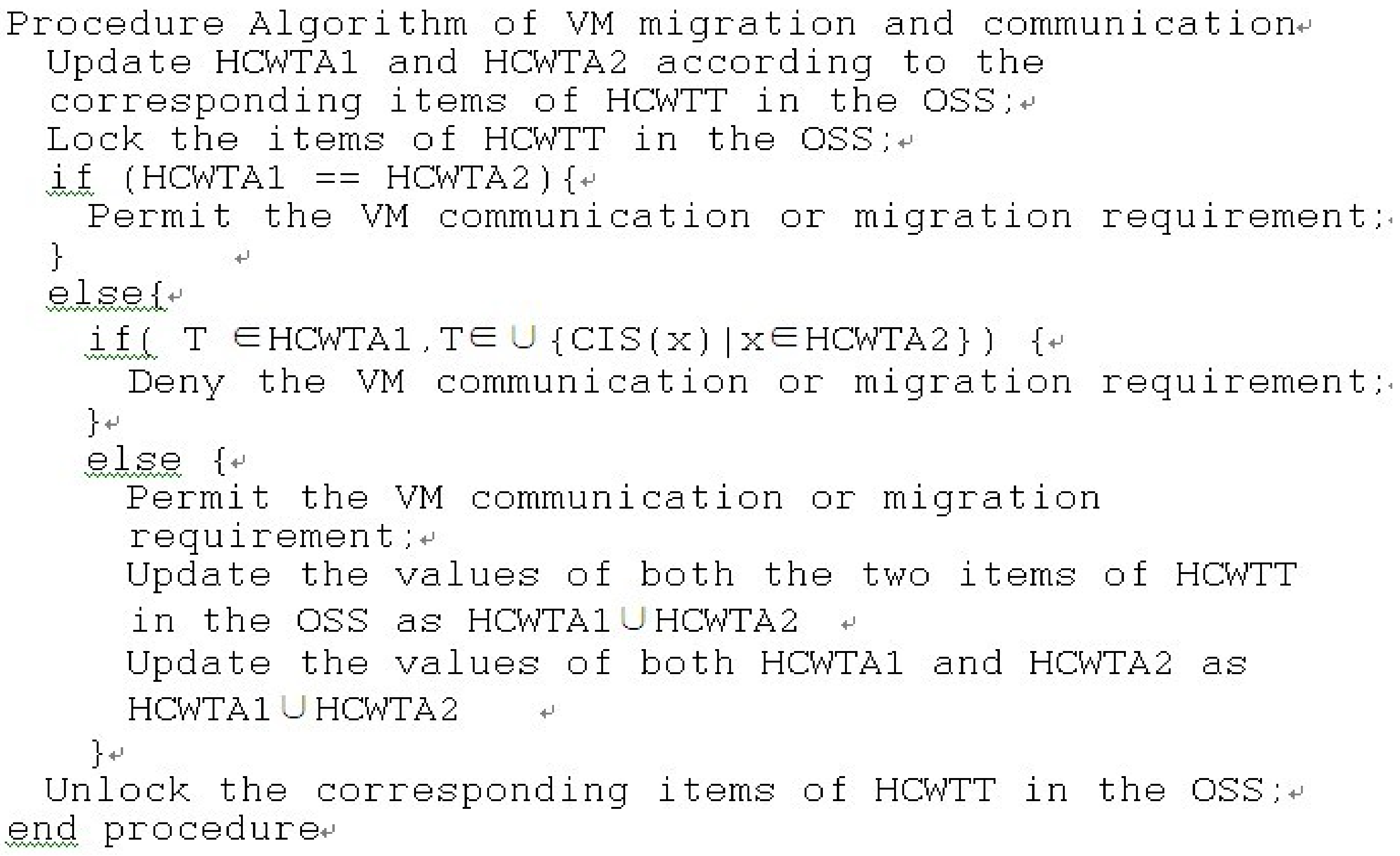
System Architecture of CFCC

Design(Algorithm)



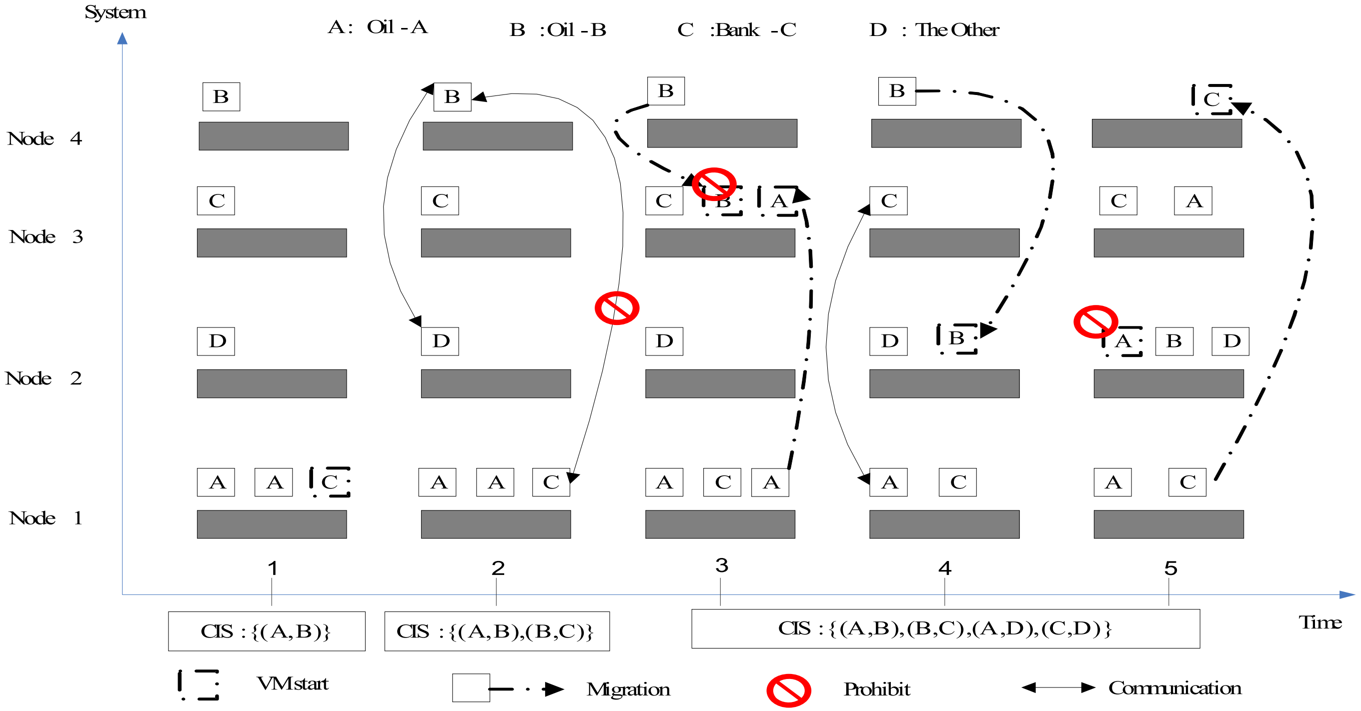
```
Procedure Algorithm of VMs start
   If (HCWTA1 is empty) {
     A VM is permitted to start in a local node;
     Put the VM's label in HCWTA1 of the local node;
     Update the HCWTA1-related item of HCWTT in the OSS;
   }
   else {
     Update HCWTA1 according to HCWTT in the OSS;
     Lock the HCWTA1-related item of HCWTT in the OSS;
     if (the VM's label is in HCWTA1 ){
     The VM is permitted to start;
     }
       else if (The VM's label is not in conflict with
             the labels listed in HCWTA1 ) {
         The VM is permitted to start in the node;
         Put the VM label in HCWTA1 of the local node;
         Update the HCWTA1-related item of HCWTT in the
         OSS;
       }
         else{
           The VM start requirement is denied;
         }
       Unlock the HCWTA1-related item of HCWTT in the OSS;
    }
end procedure
```

Design(Algorithm)



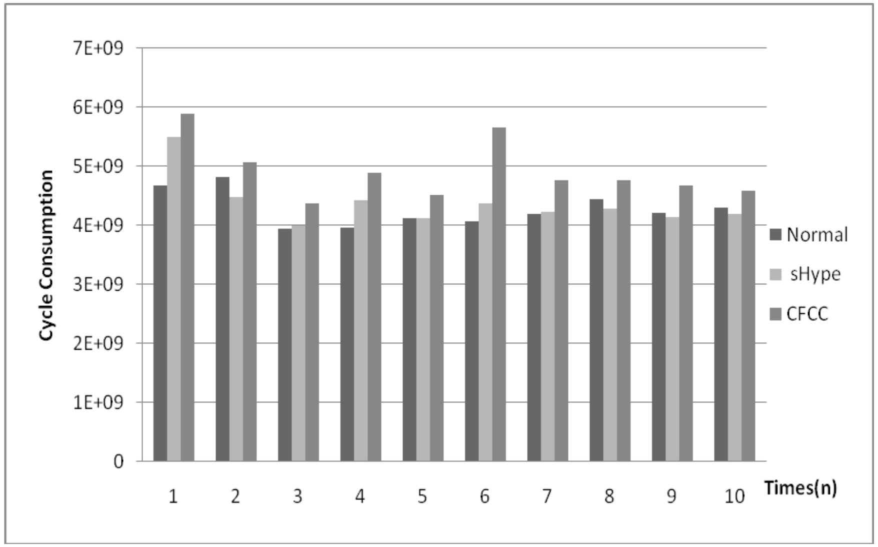
```
Procedure Algorithm of VM migration and communication
  Update HCWTA1 and HCWTA2 according to the
  corresponding items of HCWTT in the OSS;
  Lock the items of HCWTT in the OSS;
  if (HCWTA1 == HCWTA2){
    Permit the VM communication or migration requirement;
  }
  else{
    if( $T \in HCWTA1, T \in \cup \{CIS(x) | x \in HCWTA2\}$) {
      Deny the VM communication or migration requirement;
    }
    else {
      Permit the VM communication or migration
      requirement;
      Update the values of both the two items of HCWTT
      in the OSS as $HCWTA1 \cup HCWTA2$
      Update the values of both HCWTA1 and HCWTA2 as
      $HCWTA1 \cup HCWTA2$
    }
  Unlock the corresponding items of HCWTT in the OSS;
end procedure
```

Design(Case)

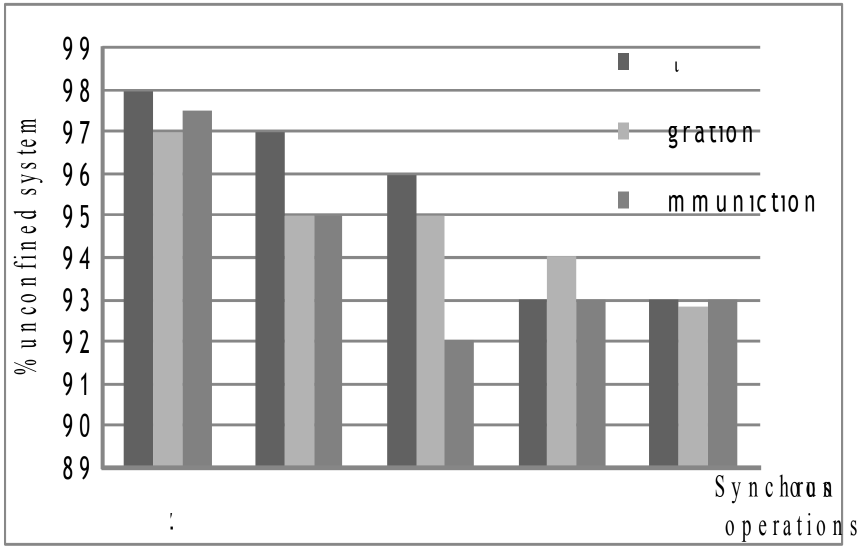


A scenario of covert flows confinement

Experiment (Performance)



Overhead of VMs startup in a single-node



Synchronization overhead

we implement a prototype, which consists of 4 machines connected with a 1000Mbit Ethernet. Three nodes used is a 2.33 GHz Intel Core Duo processor with 2 MB L2 cache, 2 GB RAM and an 80 GB 7200 RPM disk. The OSS is Pentium 4 machine with 2GHz, 2GB RAM and Federal Linux installed.

Conclusions and Future Work

- Our contribution aims to provide a mechanism to confine the covert flows (CFCC) which become a problem for VM-based cloud computing environments even enforced with mandatory access control (MAC).
- Enforcing MAC to managing the covert flows by CFCC is
  - (i) to prevent the covert flow through careful resource management.
  - (ii)to enable users through configuration options to mitigate covert channels
- Experimental results show that the performance overhead is acceptable.
- In our future work, we plan to add application level information flows control for virtual machine coalitions.

Thank You!

Any Question?