Resource Provisioning and Platform Deployment

PartII of Chapter 4 from Kai Hwang

Provisioning of Compute Resources (VMs)

- Providers supply cloud services by signing SLAs with end users. *The SLAs must commit sufficient resources such as CPU, memory, and bandwidth that the user can use for a present period.*
- *Under provisioning of resources will lead to broken SLAs and penalties*. Overprovisioning of resources will lead to resource underutilization, and consequently, a decrease in revenue for the provider.
- *Deploying an autonomous system to efficiently provision resources to users is a challenging problem*.
- The difficulty comes from the unpredictability of consumer demand, software and hardware failures, heterogeneity of services, power management, and conflicts in signed SLAs between consumers and service providers.

Conclusion:-

- Efficient VM provisioning depends on the cloud architecture and management of cloud infrastructures.
- Resource provisioning schemes also demand fast discovery of services and data in cloud computing infrastructures

Resource Provisioning Methods

- In case (a), overprovisioning with the peak load causes heavy resource waste (shaded area).
- In case (b), underprovisioning (along the capacity line) of resources results in losses by both user and provider in that paid demand by the users (the shaded area above the capacity) is not served and wasted resources still exist for those demanded areas below the provisioned capacity.
- In case (c), the constant provisioning of resources with fixed capacity to a declining user demand could result in even worse resource waste.

The user may give up the service by canceling the demand, resulting in reduced revenue for the provider. Both the user and provider may be losers in resource provisioning without elasticity.

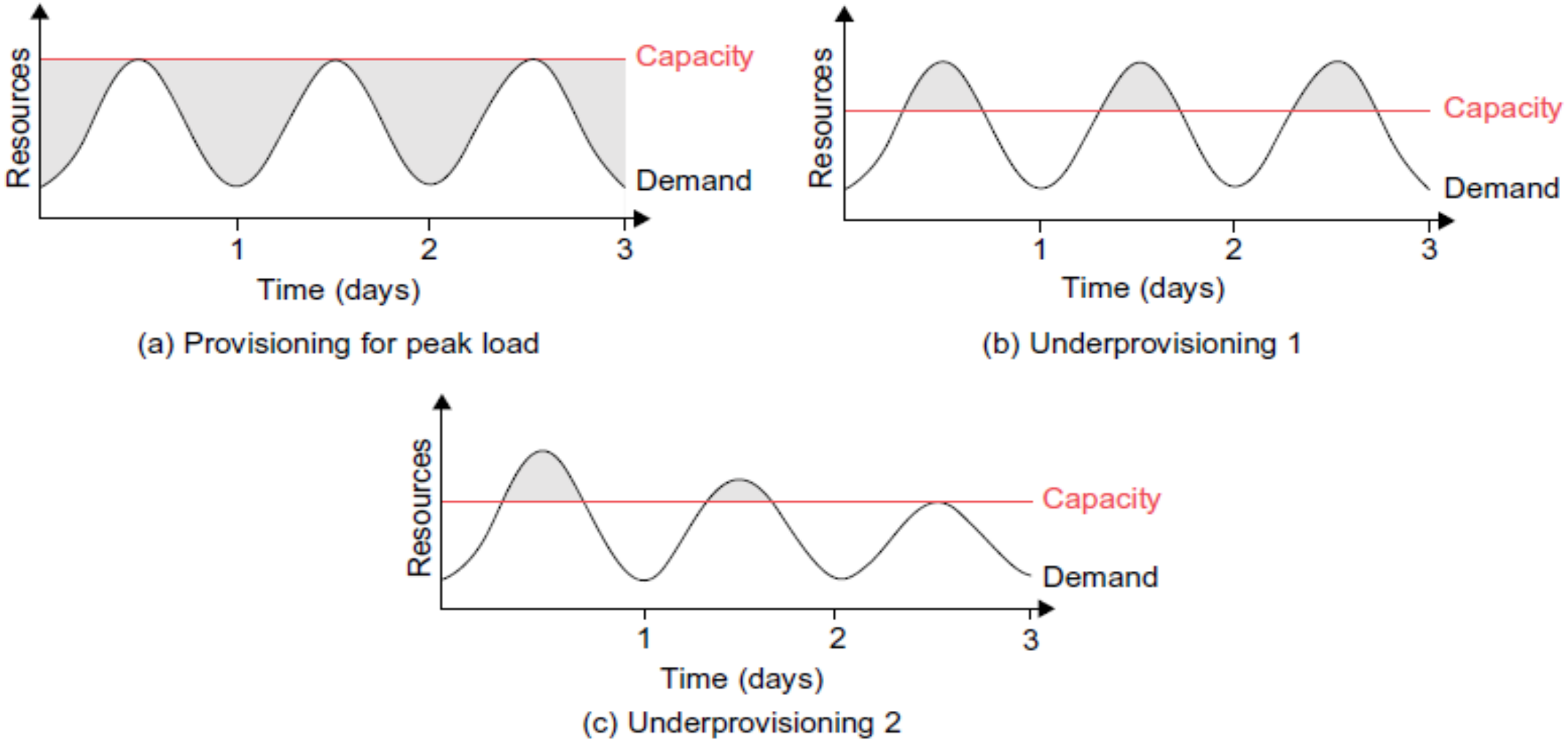


FIGURE 4.24

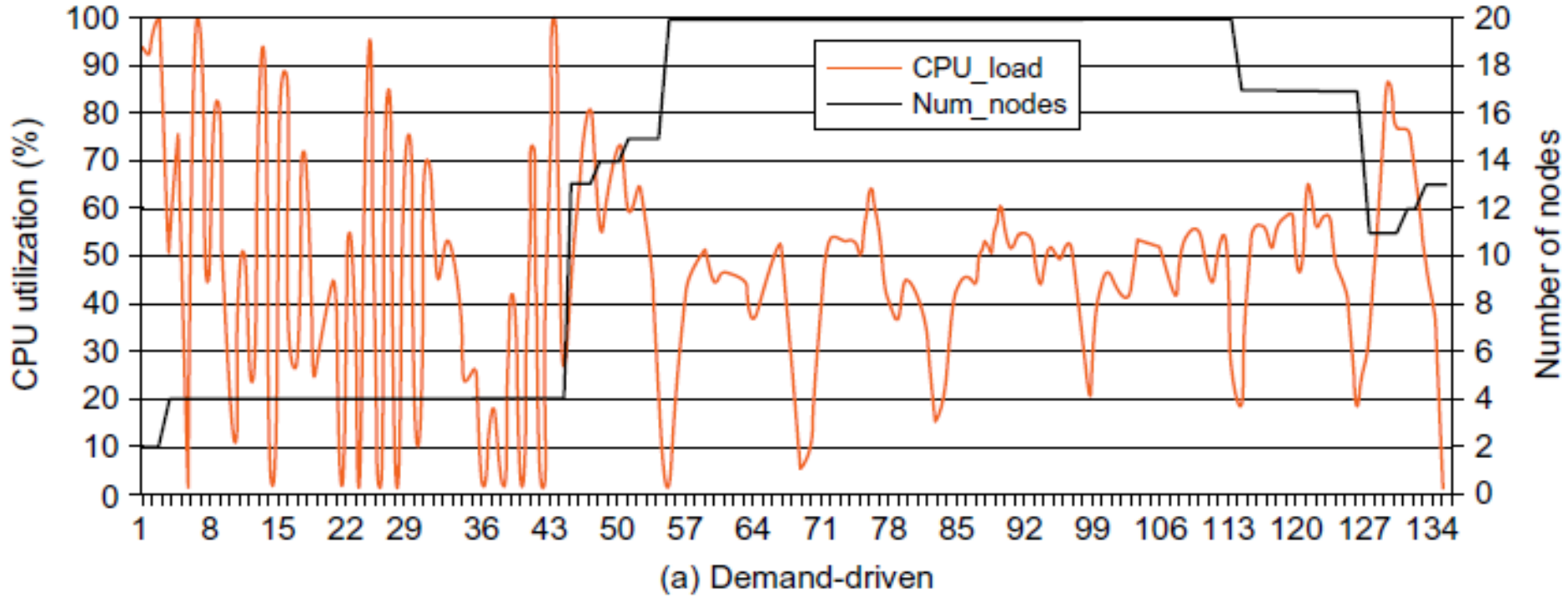
Three cases of cloud resource provisioning without elasticity: (a) heavy waste due to overprovisioning, (b) underprovisioning and (c) under- and then overprovisioning.

- Three resource-provisioning methods are presented in the following sections. ***The demand-driven method*** provides static resources and has been used in grid computing for many years. The ***event driven method*** is based on predicted workload by time. The ***popularity-driven method*** is based on Internet traffic monitored.

Demand-Driven Resource Provisioning (Auto scaling feature)

- This method adds or removes computing instances based on the current utilization level of the allocated resources.
- The demand-driven method automatically allocates two Xeon processors for the user application, when the user was using one Xeon processor more than 60 percent of the time for an extended period.
- In general, when a resource has surpassed a threshold for a certain amount of time, the scheme *increases that resource* based on demand.
- When a resource is below a threshold for a certain amount of time, that *resource could be decreased accordingly.*
- Amazon implements such an auto-scale feature in its EC2 platform. This method is easy to implement. The scheme does not work out right if the workload changes abruptly.

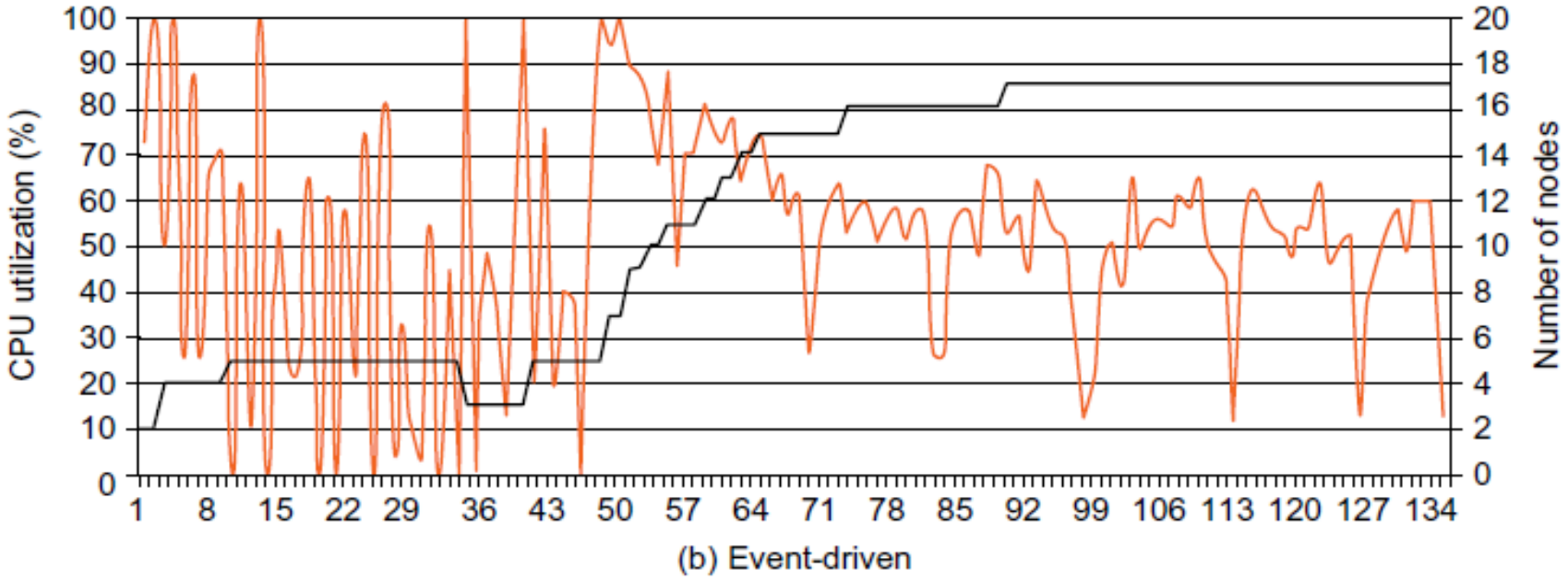
- The x-axis in Figure 4.25 is the time scale in milliseconds. In the beginning, heavy fluctuations of CPU load are encountered. All three methods have demanded a few VM instances initially. Gradually, the utilization rate becomes more stabilized with a maximum of 20 VMs (100 percent utilization) provided for demand-driven provisioning in Figure 4.25(a).



(a) Demand-driven

Event-Driven Resource Provisioning

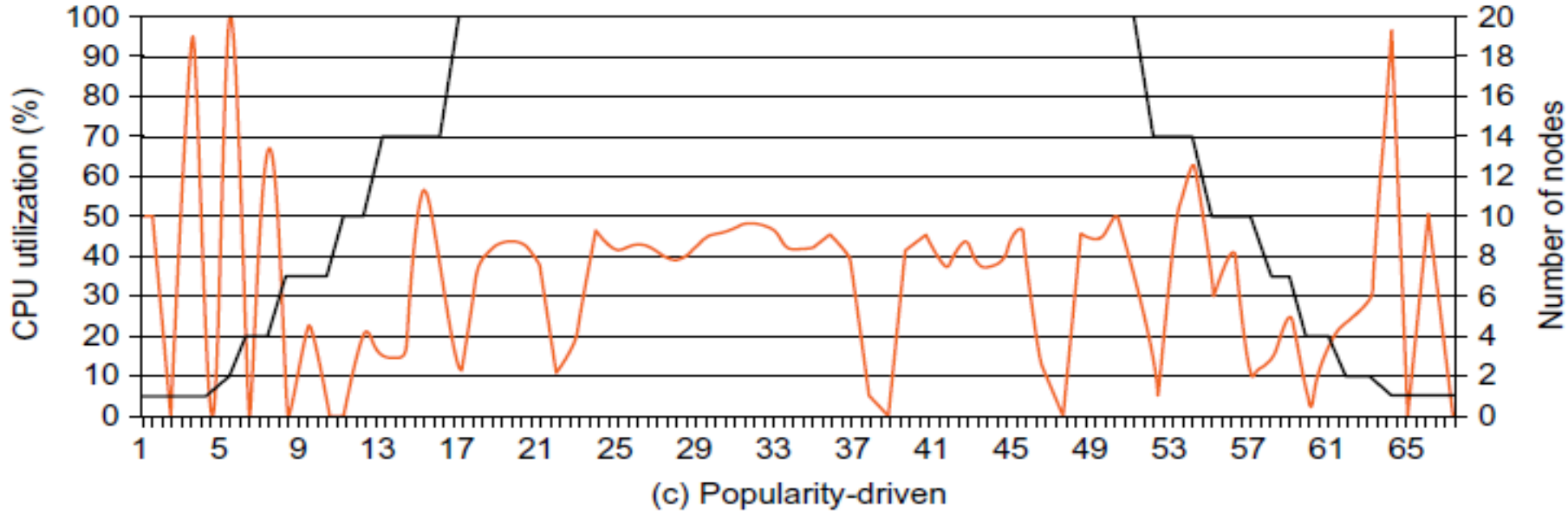
- This scheme adds or removes machine instances based ***on a specific time event***. The scheme works better for seasonal or predicted events such as Christmastime in the West and the Lunar New Year in the East. During these events, the number of users grows before the event period and then decreases during the event period. This scheme anticipates peak traffic before it happens. The method results in a minimal loss of QoS, if the event is predicted correctly. Otherwise, wasted resources are even greater due to events that do not follow a fixed pattern.



(b) Event-driven

Popularity-Driven Resource Provisioning [240]

- In this method, the Internet searches for popularity of certain applications and creates the instances by popularity demand. The scheme anticipates increased traffic with popularity. Again, the scheme has a minimal loss of QoS, if the predicted popularity is correct. Resources may be wasted if traffic does not occur as expected. In Figure 4.25(c), EC2 performance by CPU utilization rate (the dark curve with the percentage scale shown on the left) is plotted against the number of VMs provisioned (the light curves with scale shown on the right, with a maximum of 20 VMs provisioned).



(c) Popularity-driven

Global Exchange of Cloud Resources

- In order to support a large number of application service consumers from around the world.
- Cloud infrastructure providers (i.e., IaaS providers) have established data centers in ***multiple geographical locations to provide redundancy and ensure reliability in case of site failures.***
- ***Amazon example:-***For example, Amazon has data centers in the United States (e.g., one on the East Coast and another on the West Coast) and Europe also.
- However, currently Amazon expects its cloud customers (i.e., SaaS providers) to express a preference regarding where they want their application services to be hosted.
- Amazon does not provide seamless/automatic mechanisms for scaling its hosted services across ***multiple geographically distributed data centers***.

High-level components of the Melbourne group's proposed InterCloud architecture: Exchange of cloud resources.

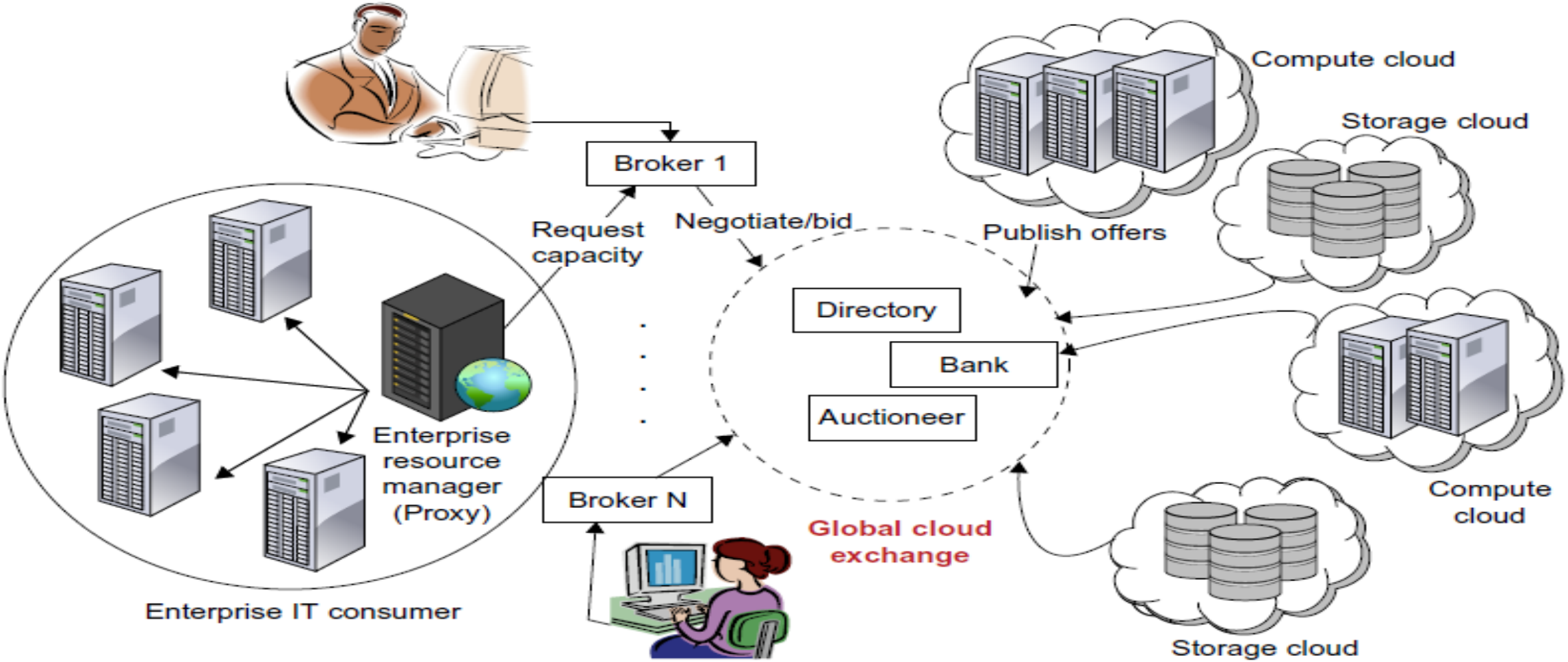


FIGURE 4.30

Inter-cloud exchange of cloud resources through brokering.

- They consist of ***client*** ***brokering*** and ***coordinator*** ***services*** that support utility-driven federation of clouds, application scheduling, resource allocation, and migration of workloads.
- The architecture cohesively couples the administrative and topologically distributed storage and compute capabilities of clouds as part of a single resource leasing abstraction.
- The system will ease the cross domain capability integration for on-demand, flexible, energy-efficient, and reliable access to the infrastructure based on virtualization technology.

- The Cloud Exchange (CEx) acts as a market maker for bringing together service producers and consumers.
- It aggregates the infrastructure demands from application brokers and evaluates them against the available supply currently published by the cloud coordinators.
- CEx allows participants to locate providers and consumers with fitting offers.
- This will pave the way for creation of dynamic market infrastructure for trading based on SLAs.
- ***An SLA specifies the details of the service to be provided in terms of metrics agreed upon by all parties(why????)***, and incentives and penalties for meeting and violating the expectations, respectively.

CLOUD SECURITY AND TRUST MANAGEMENT

- For web and cloud services, trust and security become even more demanding, ***because leaving user applications completely to the cloud providers has faced strong resistance by most PC and server users.***
- Cloud platforms become worrisome to some users for lack of privacy protection, security assurance, and copyright protection.
- Trust is a social problem, not a pure technical issue. However, the social problem can be solved with a technical approach (Example: Semantic Analysis).

- Technology can be enhance through trust, justice, reputation, credit, and assurance in Internet applications.

Cloud Security Defense Strategies

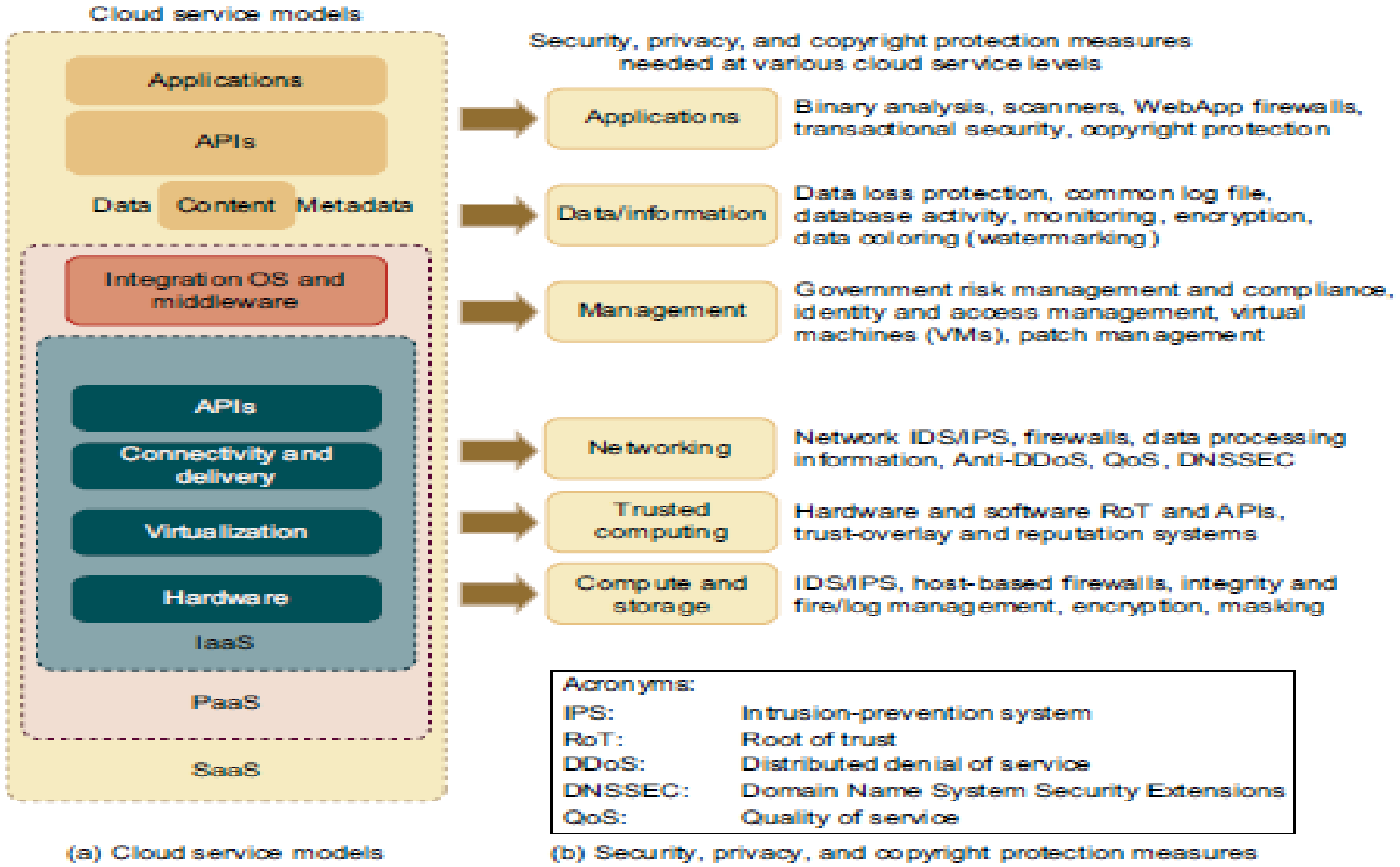
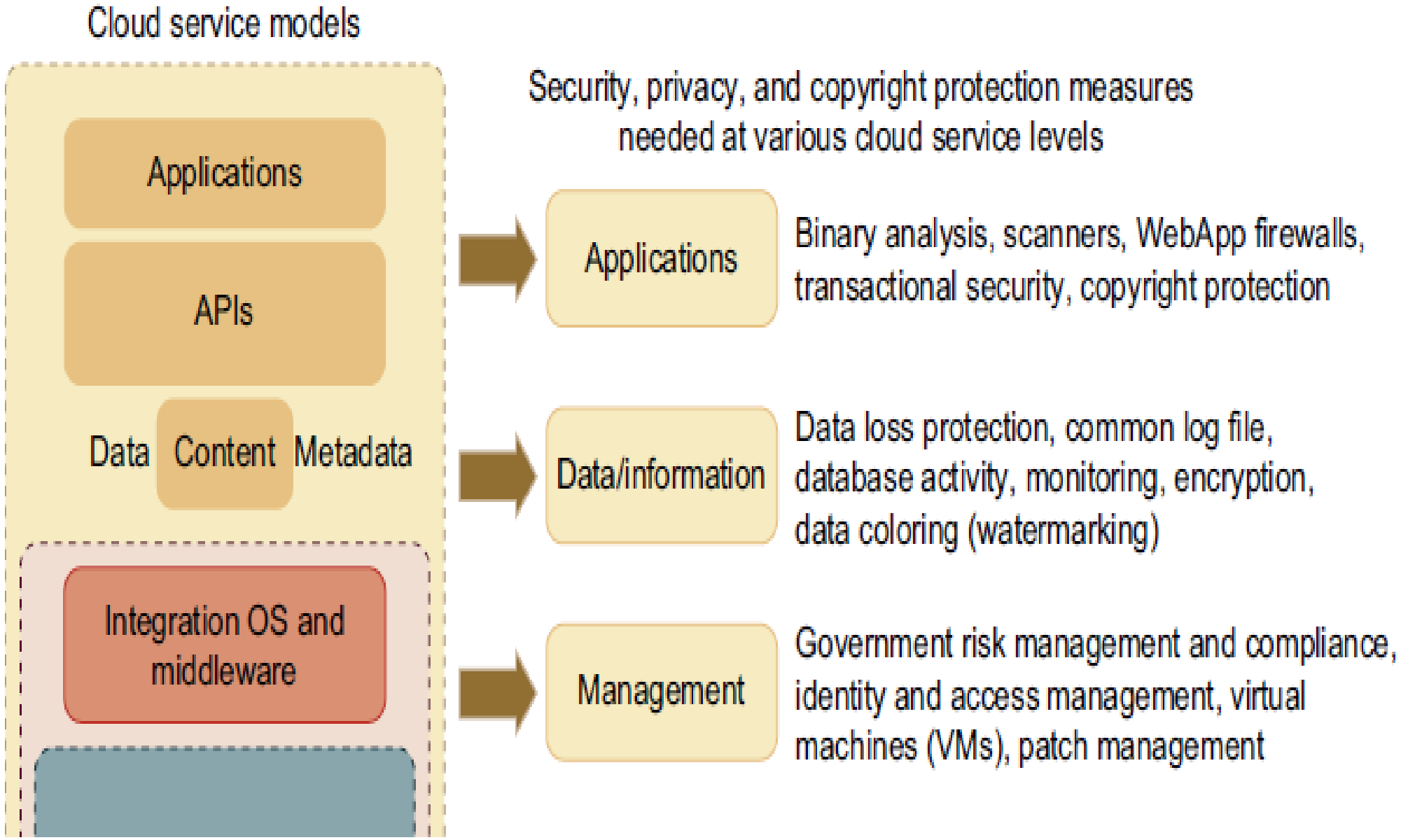
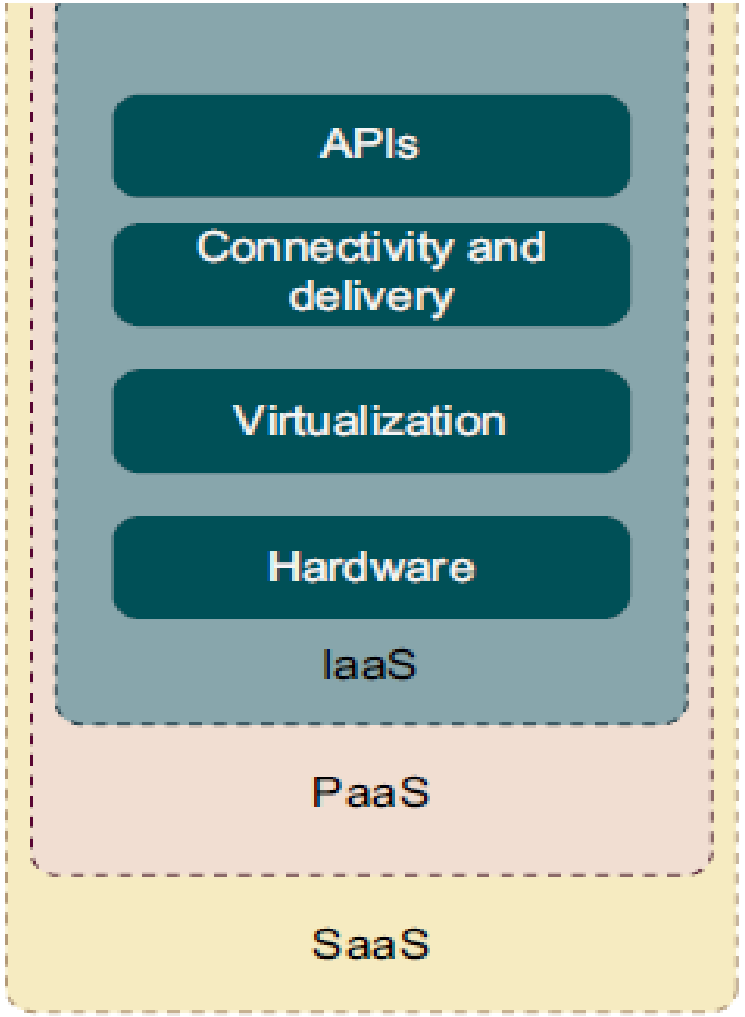


FIGURE 4.31

Cloud service models on the left (a) and corresponding security measures on the right (b); the IaaS is at the innermost level, PaaS is at the middle level, and SaaS is at the outermost level, including all hardware, software, datasets, and networking resources.

(Courtesy of Hwang and Li [36])


Cloud service models
Applications
APIs
Data
Content
Metadata
Integration OS and middleware
Security, privacy, and copyright protection measures needed at various cloud service levels
Applications
Binary analysis, scanners, WebApp firewalls, transactional security, copyright protection
Data/information
Data loss protection, common log file, database activity, monitoring, encryption, data coloring (watermarking)
Management
Government risk management and compliance, identity and access management, virtual machines (VMs), patch management



(a) Cloud service models

Networking	Network IDS/IPS, firewalls, data processing information, Anti-DDoS, QoS, DNSSEC
Trusted computing	Hardware and software RoT and APIs, trust-overlay and reputation systems
Compute and storage	IDS/IPS, host-based firewalls, integrity and fire/log management, encryption, masking

Acronyms:

IPS:	Intrusion-prevention system
RoT:	Root of trust
DDoS:	Distributed denial of service
DNSSEC:	Domain Name System Security Extensions
QoS:	Quality of service

(b) Security, privacy, and copyright protection measures

- The Domain **Name** System Security Extensions (**DNSSEC**) is a suite of Internet Engineering Task Force (IETF) specifications for securing certain kinds of information provided by the Domain **Name** System (DNS) as used on Internet Protocol (IP) networks.

Root of Trust:

https://whatis.techtarget.com/definition/Roots-of-Trust-RoT

DEFINITION

Roots of Trust (RoT)

Roots of Trust (RoT) is a set of functions in the trusted computing module that is always trusted by the computer's operating system (OS). The RoT serves as separate compute engine controlling the trusted computing platform cryptographic processor on the PC or mobile device it is embedded in.

The RoT provides the functionality behind trusted computing features including

- On the fly drive encryption.
- Detection and reporting of unauthorized changes to the operating system or programs.
- Detection of rootkits.
- Memory curtaining to prevent programs from inappropriately reading from or writing to another program's memory.
- Hardware-based digital rights management (DRM) support.

Security defenses are needed to protect all cluster servers and data centers. Here are some cloud components that demand special security protection:

- Protection of servers from malicious software attacks such as **worms, viruses, and malware**.
- Protection of hypervisors or VM monitors from software-based attacks and vulnerabilities.
- Protection of VMs and monitors from service disruption and DoS attacks.
- Protection of data and information from theft, corruption, and natural disasters
- Providing authenticated and authorized access to critical data and services

Security Challenges in VMs

- Traditional network attacks include ***buffer overflows, DoS attacks, spyware, malware, rootkits, Trojan horses, and worms.***
- A computer **worm** is a type of **malware** that spreads copies of itself from computer to computer. A **worm** can replicate itself without any human interaction, and it does not need to attach itself to a software program in order to cause damage.
- **Spyware** is unwanted software that infiltrates your computing device, stealing your internet usage data and sensitive information.
- A **rootkit** is a collection of computer software, typically malicious, designed to enable access to a computer or an area of its software that is not otherwise allowed (for example, to an unauthorized user) and often masks its existence or the existence of other software.

- A **Trojan horse**, or **Trojan**, is a type of malicious code or software that looks legitimate but can take control of your computer. *A **Trojan** is designed to damage, disrupt, steal, or in general inflict some other harmful action on your data or network.*
- In a cloud environment, newer attacks may result from **hypervisor malware, guest hopping and hijacking, or VM rootkits**. Another type of attack is the man-in-the-middle attack for VM migrations.
- In general, passive attacks steal sensitive data or passwords. Active attacks may manipulate kernel data structures which will cause major damage to cloud servers. An IDS can be a N(Network-based)IDS or a H(Host based)IDS.

Cloud Defense Methods

Table 4.9 Physical and Cyber Security Protection at Cloud/Data Centers

Protection Schemes	Brief Description and Deployment Suggestions
Secure data centers and computer buildings	Choose hazard-free location, enforce building safety. Avoid windows, keep buffer zone around the site, bomb detection, camera surveillance, earthquake-proof, etc.
Use redundant utilities at multiple sites	Multiple power and supplies, alternate network connections, multiple databases at separate sites, data consistency, data watermarking, user authentication, etc.
Trust delegation and negotiation	Cross certificates to delegate trust across PKI domains for various data centers, trust negotiation among certificate authorities (CAs) to resolve policy conflicts

Worm containment and DDoS defense	Internet worm containment and distributed defense against DDoS attacks to secure all data centers and cloud platforms
Reputation system for data centers	Reputation system could be built with P2P technology; one can build a hierarchy of reputation systems from data centers to distributed file systems
Fine-grained file access control	Fine-grained access control at the file or object level; this adds to security protection beyond firewalls and IDSes
Copyright protection and piracy prevention	Piracy prevention achieved with peer collusion prevention, filtering of poisoned content, nondestructive read, alteration detection, etc.
Privacy protection	Uses double authentication, biometric identification, intrusion detection and disaster recovery, privacy enforcement by data watermarking, data classification, etc.

Defense with Virtualization

- The VM is decoupled from the physical hardware. The entire VM can be represented as a software component and can be regarded as binary or digital data.
- The VM can be saved, cloned, encrypted, moved, or restored with ease. VMs enable HA and faster disaster recovery.
- Live migration of VMs was suggested by many researchers for building ***distributed intrusion detection systems*** (DIDSes).
- Multiple IDS VMs can be deployed at various resource sites including data centers.
- DIDS design demands trust negation among PKI domains. Security policy conflicts must be resolved at design time and updated periodically.

Privacy and Copyright Protection

- Here are several security features desired in a secure cloud:

Dynamic web services with full support from secure web technologies
Established trust between users and providers through SLAs and reputation systems

- Effective user identity management and data-access management
  - Single sign-on and single sign-off to reduce security enforcement overhead
  - Auditing and copyright compliance through proactive enforcement
  - Shifting of control of data operations from the client environment to cloud providers
  - Protection of sensitive and regulated information in a shared environment

Distributed Intrusion/Anomaly Detection

- Data security is the weakest link in all cloud models. Many IT companies are now offering cloud services with no guaranteed security.

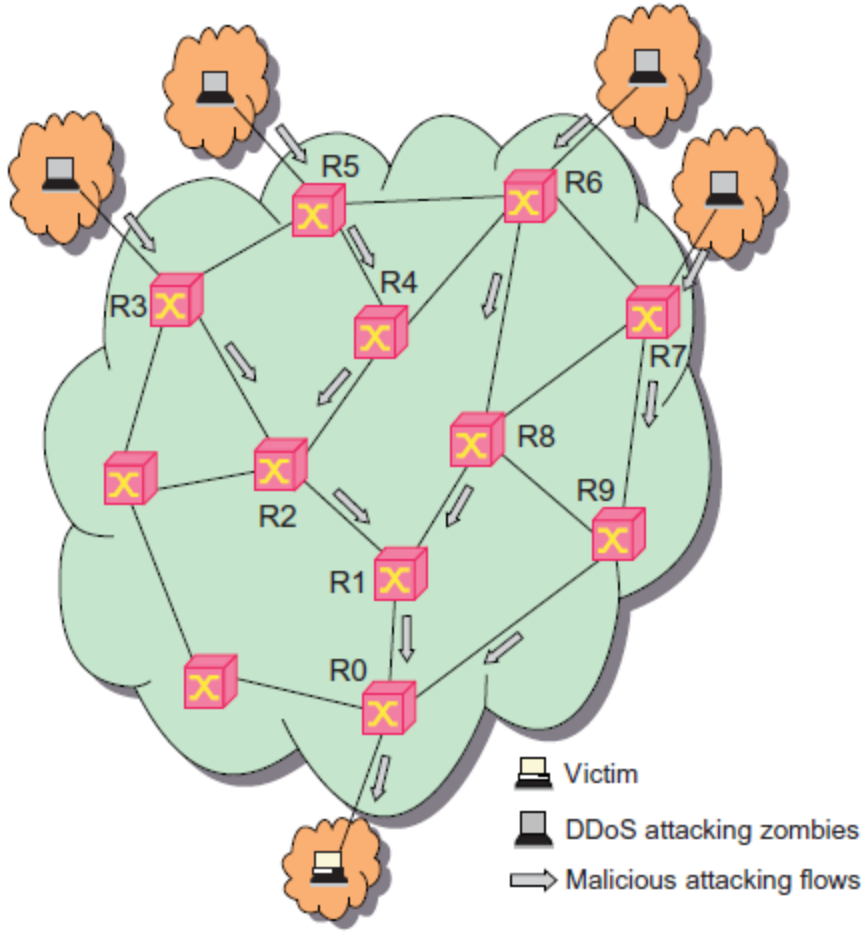
"***Security threats may be aimed at VMs, guest OSes, and software running on top of the cloud***. IDSes attempt to stop these attacks before they take effect. Both signature matching and anomaly detection can be implemented on VMs dedicated to building IDSes"

- Signature-matching IDS technology is more mature, but require frequent updates of the signature databases.
- Network anomaly detection reveals abnormal traffic patterns, such as unauthorized episodes of TCP connection sequences, against normal traffic patterns. Distributed IDSes are needed to combat both types of intrusions.

Distributed Defense against DDoS Flooding Attacks

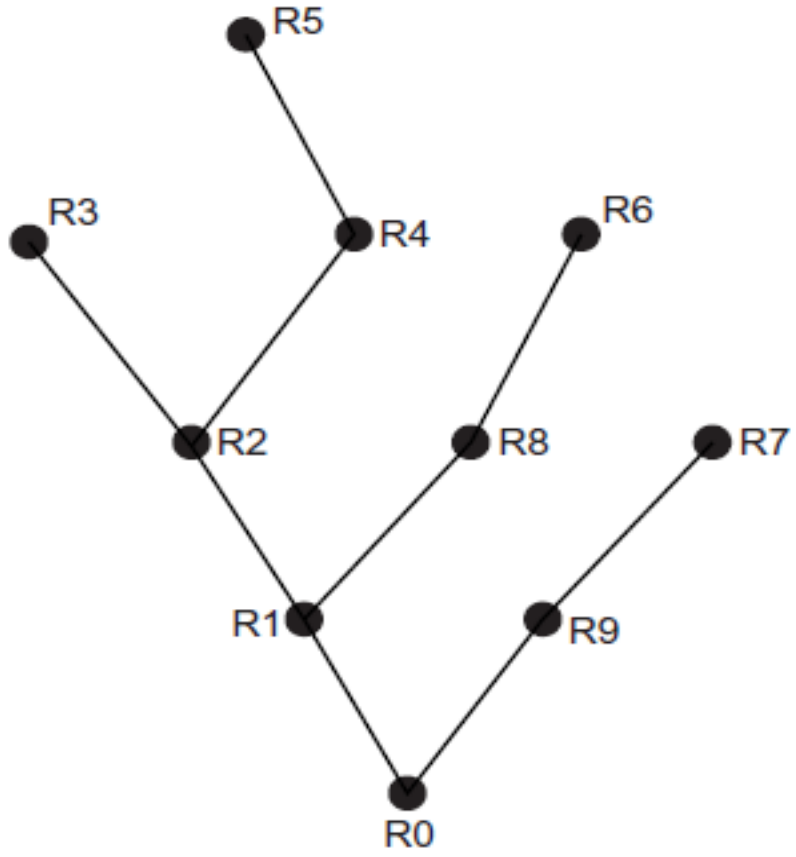
A DDoS defense system must be designed to cover multiple network domains spanned by a given cloud platform. These network domains cover the edge networks where cloud resources are connected.

DDoS attacks come with widespread worms. The flooding traffic is large enough to crash the victim server by buffer overflow, disk exhaustion, or connection saturation. shows a flooding attack pattern. Here, the hidden attacker launched the attack from many zombies toward victim server at the bottom router R0.



(a) Traffic flow pattern of a DDoS attack

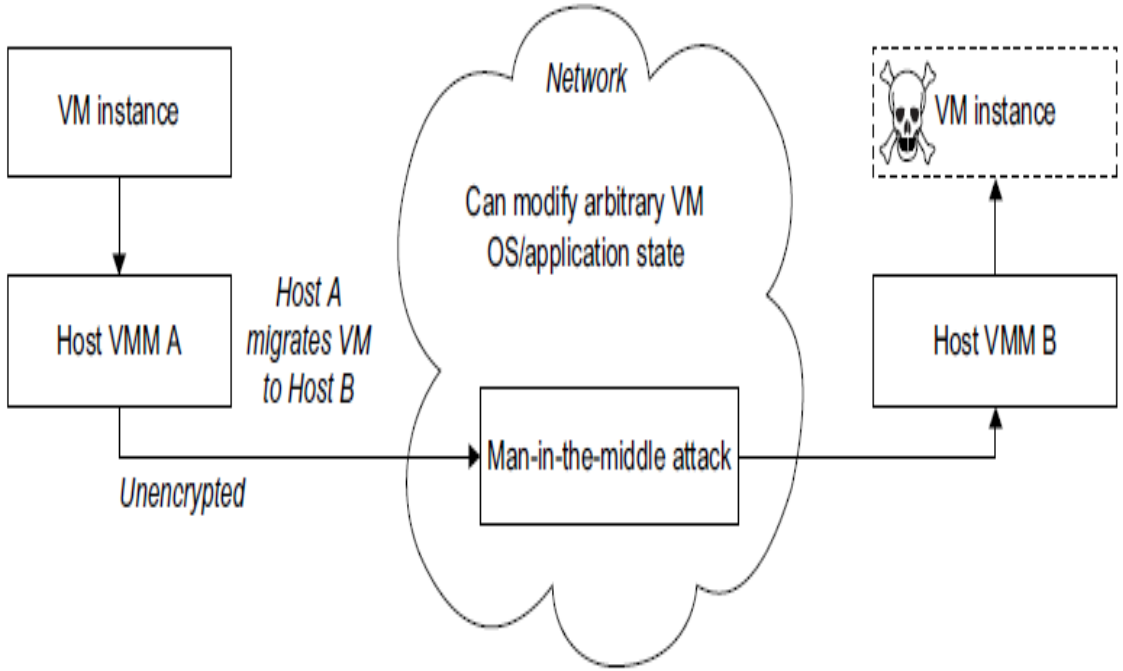
The flooding traffic flows essentially with a tree pattern shown in Figure 4(b). Successive attack-transit routers along the tree reveal the abnormal surge in traffic. This DDoS defense system is based on change-point detection by all routers. Based on the anomaly pattern detected in covered network domains, the scheme detects a DDoS attack before the victim is overwhelmed. The detection scheme is suitable for protecting cloud core networks. The provider-level cooperation eliminates the need for intervention by edge networks



(b) The attack traffic flow tree over 10 routers

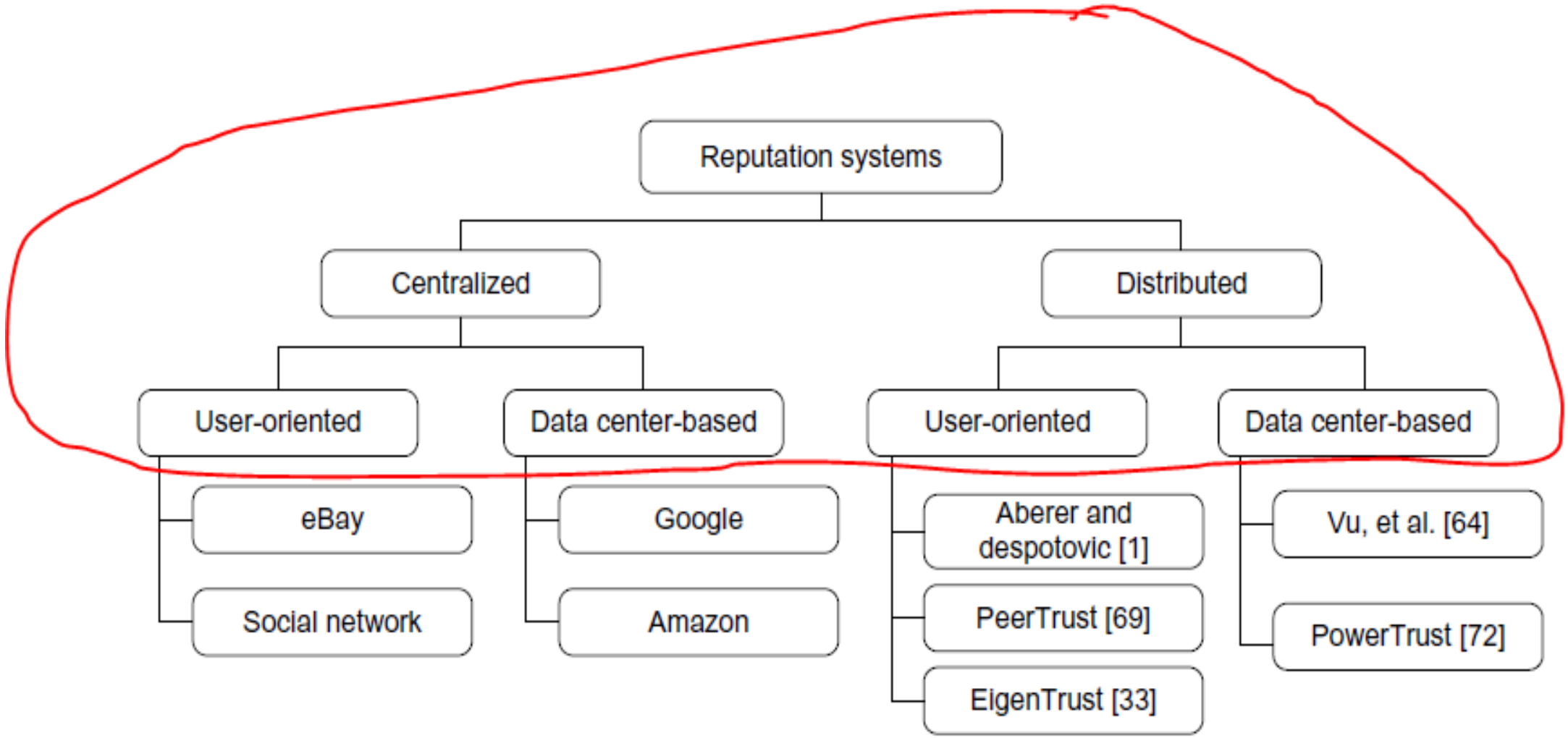
Man-in-the-Middle Attacks

- Shows VM migration from host machine VMM A to host machine VMM B, via a security vulnerable network. In a man-in-the-middle attack, the attacker can view the VM contents being migrated, steal sensitive data, or even modify the VM-specific contents including the OS and application states. An attacker posing this attack can launch an active attack to insert a VM-based rootkit into the migrating VM, which can subvert the entire operation of the migration process without the knowledge of the guest OS and embedded application.



Reputation System Design Options

- Trust is a personal opinion, which is very subjective and often biased.
- Trust can be transitive but not necessarily symmetric between two parties.
- Reputation is a public opinion, which is more objective and often relies on a large opinion aggregation process to evaluate. Reputation may change or decay over time.
- Recent reputation should be given more preference than past reputation.


Reputation systems
Centralized
Distributed
User-oriented
Data center-based
User-oriented
Data center-based
eBay
Social network
Google
Amazon
Aberer and despotovic [1]
PeerTrust [69]
EigenTrust [33]
Vu, et al. [64]
PowerTrust [72]

- A centralized reputation system is easier to implement, but demands more powerful and reliable server resources; a distributed reputation system is much more complex to build
- Distributed systems are more scalable and reliable in terms of handling failures.
- Distributed reputation systems are mostly developed by academic research communities

Trust Overlay Networks

- Reputation represents a collective evaluation by users and resource owners.
- Many reputation systems have been proposed in the past for P2P, multiagent, or e-commerce systems.
- To support trusted cloud services, Hwang and Li have suggested building a trust overlay network to model trust relationships among data-center modules.
- This trust overlay could be structured with a distributed hash table (DHT) to achieve fast aggregation of global reputations from a large number of local reputation scores.
- Here(Figure in next slide), the designer needs to have two layers for fast reputation aggregation, updating, and dissemination to all users.

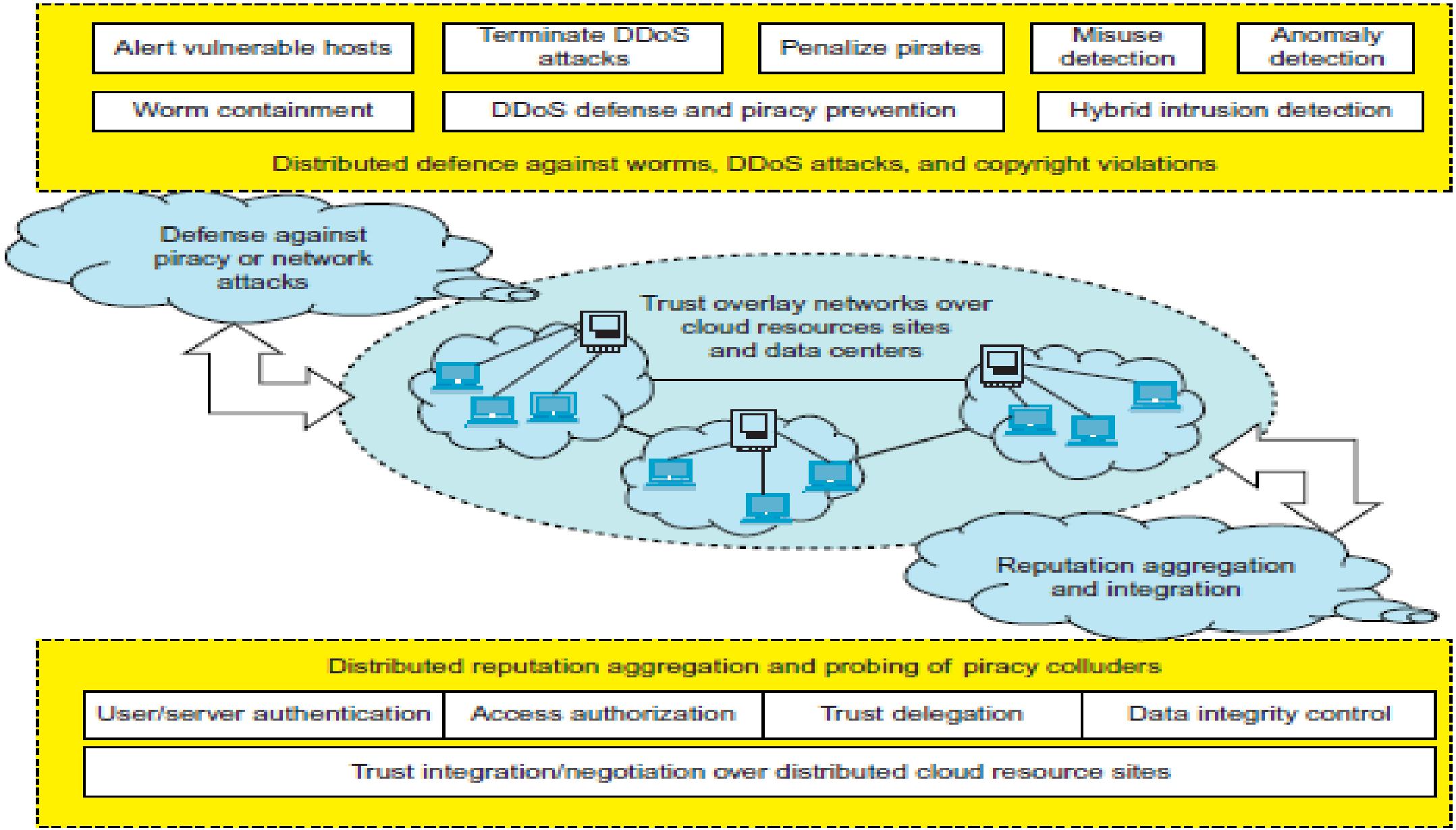


FIGURE 4.37

DHT-based trust overlay networks built over cloud resources provisioned from multiple data centers for trust management and distributed security enforcement.

(Courtesy of Hwang and Li [36])

Different layers of trust.

- At the bottom layer is the trust overlay for distributed trust negotiation and reputation aggregation over multiple resource sites.
- This layer handles user/server authentication, access authorization, trust delegation, and data integrity control.
- At the top layer is an overlay for fast virus/worm signature generation and dissemination and for piracy detection.
- This overlay facilitates worm containment and IDSes against viruses, worms, and DDoS attacks.
- The content poisoning technique is reputation-based.
- This protection scheme can stop copyright violations in a cloud environment over multiple data centers.

- The reputation system enables trusted interactions between cloud users and data-center owners.
- Privacy is enforced by matching colored user identifications with the colored data objects.
- The use of content poisoning was suggested to protect copyright of digital content.

"The design is aimed at a trusted cloud environment to ensure high-quality services, including security."

Conclusion:-The cloud security trend is to apply virtualization support for security enforcement in data centers.

- Both reputation systems and data watermarking mechanisms can protect data-center access at the coarse-grained level and to limit data access at the fine-grained file level.
- In the long run, a new Security as a Service is desired.
- This “SaaS” is crucial to the universal acceptance of web-scale cloud computing in personal, business, community, and government applications.

About Data coloring and watermarking techniques

- Data coloring and software watermarking techniques protect shared data objects and massively distributed software modules.
- These techniques safeguard multi-way authentications, enable single sign-on in the cloud, and tighten access control for sensitive data in both public and private clouds.
- Defense against tampering is tamper-proofing, so that unauthorized modifications to software (for example, to remove a watermark) will result in nonfunctional code.
- Providers can implement our proposed reputation system and data-coloring mechanism to protect data-center access at a coarse-grained level and secure data access at a fine-grained file level