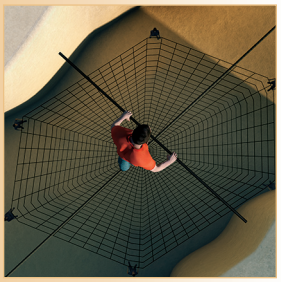

Trusted Cloud Computing with Secure Resources and Data Coloring

Trust and security have prevented businesses from fully accepting cloud platforms. To protect clouds, providers must first secure virtualized data-center resources, uphold user privacy, and preserve data integrity. The authors suggest using a trust-overlay network over multiple data centers to implement a reputation system for establishing trust between service providers and data owners. 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.

Kai Hwang
University of Southern California

Deyi Li
Tsinghua University, China

Cloud computing enables a new business model that supports on-demand, pay-for-use, and economies-of-scale IT services over the Internet. The Internet cloud works as a service factory built around virtualized data centers.[1] Cloud platforms are dynamically built through virtualization with provisioned hardware, software, networks, and datasets. The idea is to migrate desktop computing to a service-oriented platform using virtual server clusters at data centers. However, a lack of trust between cloud users and providers has hindered the universal acceptance of clouds as outsourced computing services. To promote multitenancy, we must design the cloud ecosystem to be secure, trustworthy, and dependable.[2]

In reality, trust is a social problem, not a purely technical issue. However, we believe that technology can enhance trust, justice, reputation, credibility, and assurance in Internet applications. To increase the adoption of Web and cloud services, *cloud service providers* (CSPs) must first establish trust and security to alleviate the worries of a large number of users. A healthy cloud ecosystem should be free from abuses, violence, cheating, hacking, viruses, rumors, pornography, spam, and privacy and copyright violations. Both public and private clouds demand "trusted zones" for data, *virtual machines* (VMs), and user identity, as VMware and EMC[3] originally introduced.

Data integrity issues in the cloud

 1089-7801/10/$26.00 © 2010 IEEE

*Table 1. Cloud platforms, reported services, and security features.**

Model	IBM	Amazon	Google	Microsoft	Salesforce.com
Platform as a service	BlueCloud, Websphere CloudBurst Appliance, Research Compute Cloud (RC2)		Google App Engine	Windows Azure	Force.com
Infrastructure as a service	Ensembles	Elastic Compute Cloud, Simple Storage Service, Simple Queue Service, SimpleDB			
Software as a service	Lotus Live		Gmail, Docs	.NET service, dynamic customer relationship management (CRM)	Online CRM, Gifttag
Reported services	Service-oriented architecture, B2, Tivoli Service Automation Manager, Rational Application Developer, Web 2.0	Amazon Web Services, Hadoop	GFS, BigTable, MapReduce	Live, Structured Query Language, Azure, Hotmail	Apex, Visualforce, record security
Security features	WebSphere2 and PowerVM tuned for protection	Public-key infrastructure and VPN for security, Elastic Block Store to recover from failure	Some HW security in data centers	Replicated data, rule-based access control	Administrative record security, metadata API

*Blank table entries refer to unknown services or cloud applications that are still under development.

differ from those in traditional database systems. Cloud users are most concerned about whether data-center owners will abuse the system by randomly using private datasets or releasing sensitive data to a third party without authorization. Cloud security hinges on how to establish trust between these service providers and data owners. To address these issues, we propose a reputation-based trust-management scheme augmented with data coloring and software watermarking. Information about related trust models is available elsewhere.[2,4]

Cyber-Trust Demands in Cloud Services

The Cloud Security Alliance[5] has identified a few critical issues for trusted cloud computing, and several recent works discuss general issues on cloud security and privacy.[1,6,7] Public and private clouds demand different levels of security enforcement. We can distinguish among different *service-level agreements* (SLAs) by their variable degree of shared responsibility between cloud providers and users. Critical security issues include data integrity, user confidentiality, and trust among providers, individual users, and user groups. The three most popular cloud service models have varying security demands, which we detail in Table 1.

The *infrastructure-as-a-service* (IaaS) model sits at the innermost implementation layer, which is extended to form the *platform-as-a-service* (PaaS) layer by adding OS and middleware support. PaaS further extends to the *software-as-a-service* (SaaS) model by creating applications on data, content, and metadata using special APIs. This implies that SaaS demands all protection functions at all levels. At the other extreme, IaaS demands protection mainly at the networking, trusted computing, and compute/storage levels, whereas PaaS embodies the IaaS support plus additional protection at the resource-management level. Figure 1 characterizes the various security, privacy, and copyright protection measures these models demand.

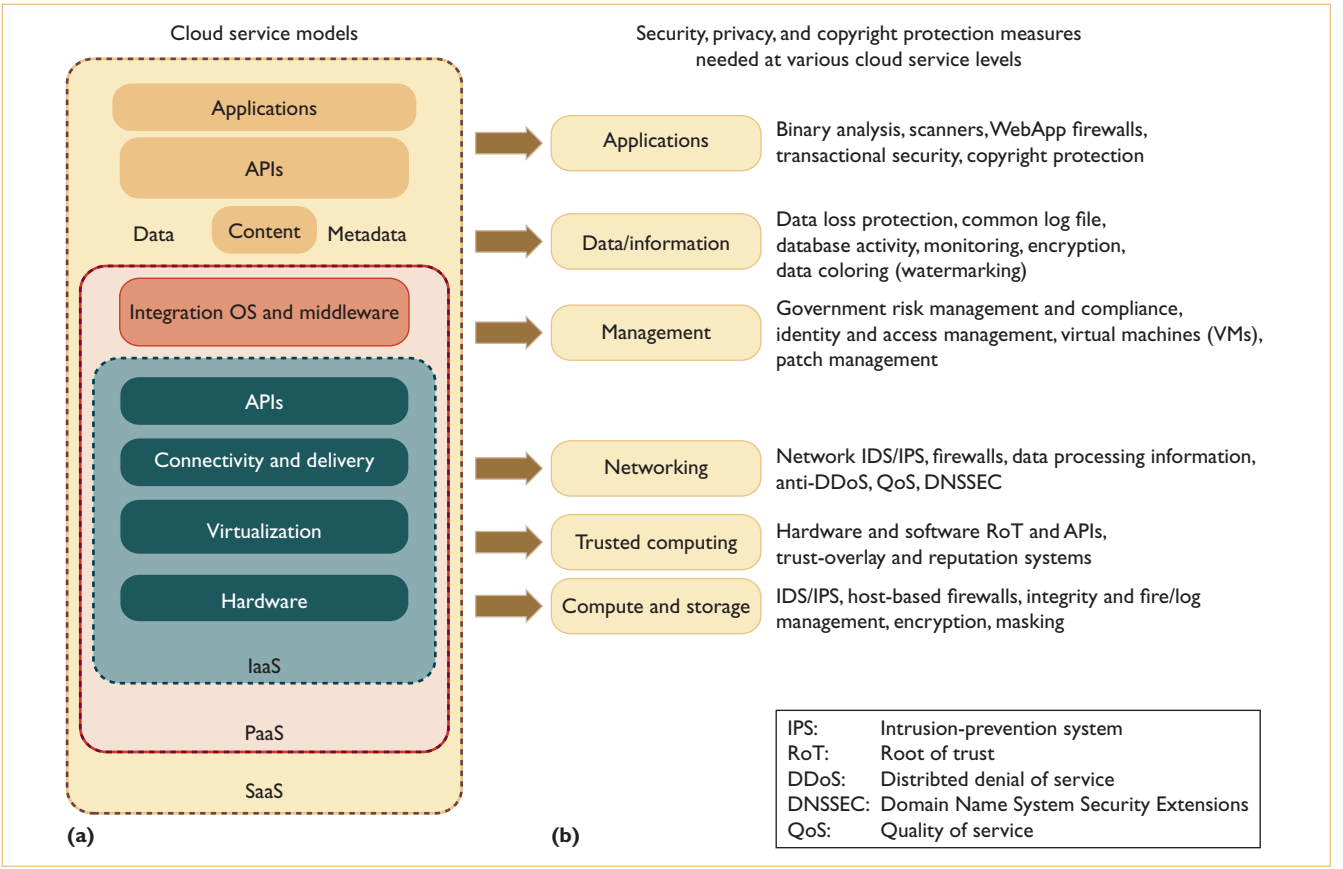

Figure 1. Three cloud service models. (a) Infrastructure as a service (IaaS) is built on top of virtualized compute, storage, and network resources, platform as a service (PaaS) at the OS/middleware level, and software as a service (SaaS) at the user application level. Each service level requires (b) different security, privacy, and copyright protection measures.

Many of the protection features Figure 1 lists are well established in grid and network-based computing systems — we can apply them to protecting clouds as well. The new features we suggest (bolded in the figure) include securing cloud computing with copyrighted content, data coloring (watermarking), VM management, trust-overlay construction, and reputation systems specifically designed for protecting data centers. We detail these new features in later sections, but first let's examine the existing models and their security features.

Securing Infrastructure as a Service

The IaaS model lets users lease compute, storage, network, and other resources in a virtualized environment. The user doesn't manage or control the underlying cloud infrastructure but has control over the OS, storage, deployed applications, and possibly certain networking components. Amazon's Elastic Compute Cloud (EC2) is a good example of IaaS. At the cloud infrastructure level, CSPs can enforce network security with intrusion-detection systems (IDSs), firewalls, antivirus programs, distributed denial-of-service (DDoS) defenses, and so on.

Securing Platform as a Service

Cloud platforms are built on top of IaaS with system integration and virtualization middleware support. Such platforms let users deploy user-built software applications onto the cloud infrastructure using provider-supported programming languages and software tools (such as Java, Python, or .NET). The user doesn't manage the underlying cloud infrastructure. Popular PaaS platforms include the Google App Engine (GAE) or Microsoft Windows Azure. This level requires securing the provisioned VMs, enforcing security compliance, managing potential risk, and establishing trust among all cloud users and providers.

Securing Software as a Service

SaaS employs browser-initiated application software to serve thousands of cloud customers, who make no upfront investment in servers or software licensing. From the provider's perspective, costs are rather low compared with conventional application hosting. SaaS — as heavily pushed by Google, Microsoft, Salesforce.com, and so on — requires that data be protected from loss, distortion, or theft. Transactional security and copyright compliance are designed to protect all intellectual property rights at this level. Data encryption and coloring offer options for upholding data integrity and user privacy.

Cloud Providers and Reported Services

Table 1 lists the major cloud providers and summarizes the services they provide. For example, GAE offers PaaS for upgraded Web-scale cloud services. The best SaaS applications are IBM Lotus Live, Google's Gmail and Docs, and online customer relationship management (CRM) services from Salesforce.com. The Research Compute Cloud (RC2) now supports eight IBM Research Centers, and Amazon Web Services (AWS) includes EC2 for running virtual servers, Simple Storage Service (S3) for online storage, and Simple Queue Service (SQS) for communication services. Microsoft Windows Azure also supports PaaS and SaaS applications.

Cloud security involves hardware and software facilities, networking and platforms, and large datasets. Cloud computing demands three primary security requirements: *confidentiality, integrity,* and *availability*. As we move from SaaS to PaaS to IaaS, providers gradually release control over security to cloud users. The SaaS model relies on the cloud provider to perform all security functions, whereas, at the other extreme, the IaaS model expects users to assume almost all security functions except availability. The PaaS model relies on providers to maintain data integrity and availability but burdens users with confidentiality and privacy control.

Data Integrity and Privacy Protection

Users desire a cloud software environment that provides many useful tools for building cloud applications over large datasets. Let's look at some security and privacy features these users desire:

- cloud resources they can access with security protocols such as HTTPS or Secure Sockets Layer (SSL), as well as security auditing and compliance checking;
- fine-grained access control to protect data integrity and deter intruders or hackers, as well as single sign-on or sign-off;
- shared datasets that are protected from malicious alteration, deletion, or copyright violations;
- a method to prevent ISPs or CSPs from invading user privacy;
- CSPs that fight against spyware and Web bugs; and
- personal firewalls and shared datasets protected from Java, JavaScript, and ActiveX Applets, as well as established VPN channels between resource sites and cloud clients.

We can enhance some of these features with cloud reputation systems and more efficient identity management systems, which we discuss in subsequent sections.

Trusted Cloud Computing over Data Centers

Malware-based attacks such as worms, viruses, and DoS exploit system vulnerabilities and give intruders unauthorized access to critical information. Risky cloud platforms can cause businesses to lose billions of dollars and might disrupt public services. We propose a security-aware cloud architecture and identify the protection mechanisms needed.

Security-Aware Cloud Architecture

Figure 2 shows the security-aware cloud architecture we propose. This architecture helps insulate network attacks by establishing trusted operational zones for various cloud applications. Security compliance demands that CSPs protect all data-center servers and storage areas. Our architecture protects VM monitors (or *hypervisors*) from software-based attacks and safeguards data and information from theft, corruption, and natural disasters. It provides strong authentication and authorized access to sensitive data and on-demand services. We had several design objectives for a trusted and dependable cloud when creating our architecture.

Virtual network security and trust negotiation. Virtual network security protects VMs in vir-

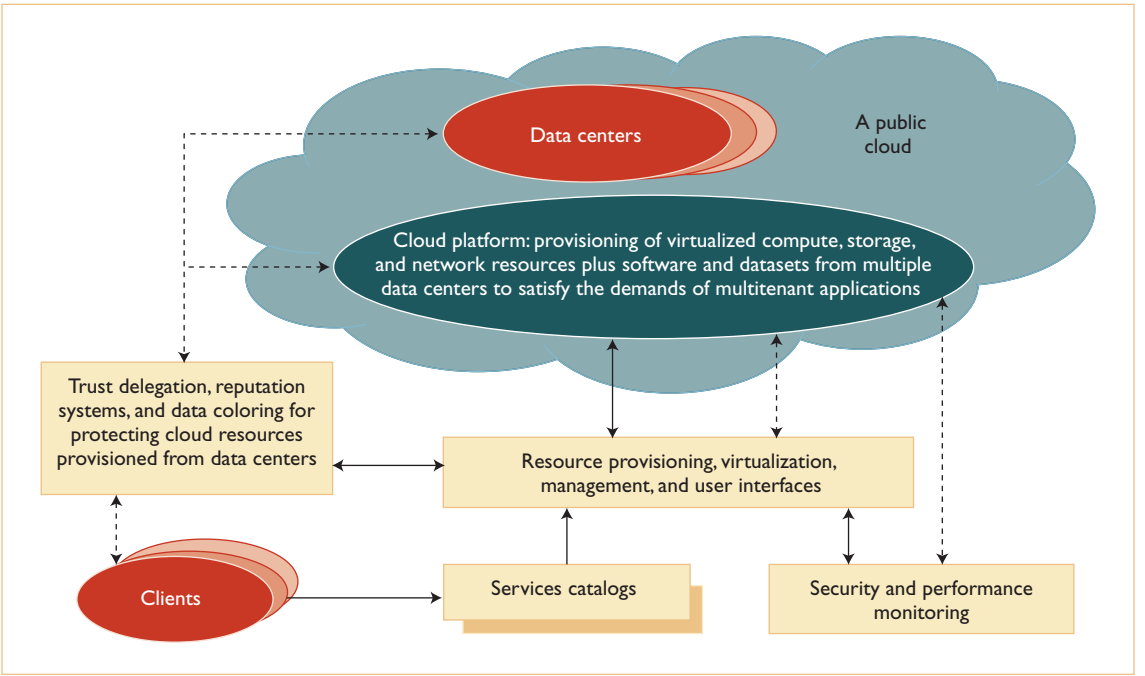

Figure 2. A trusted and dependable cloud architecture. Our architecture has secure resources and protected data access at data centers. Solid lines represent data or service flows and dashed lines control flows in trust management and security enforcement operations.

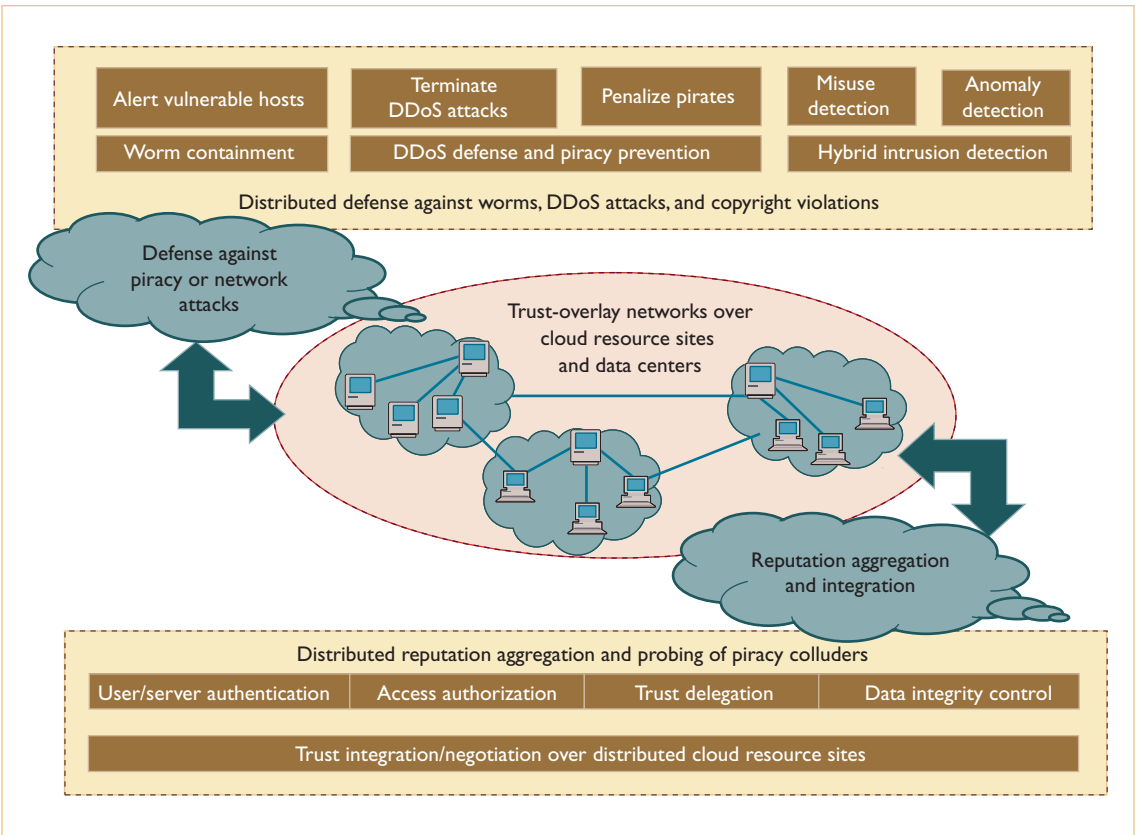

Figure 3. Distributed-hash-table (DHT)-based trust-overlay networks. We build these networks over cloud resources provisioned from multiple data centers for trust management and distributed security enforcement.

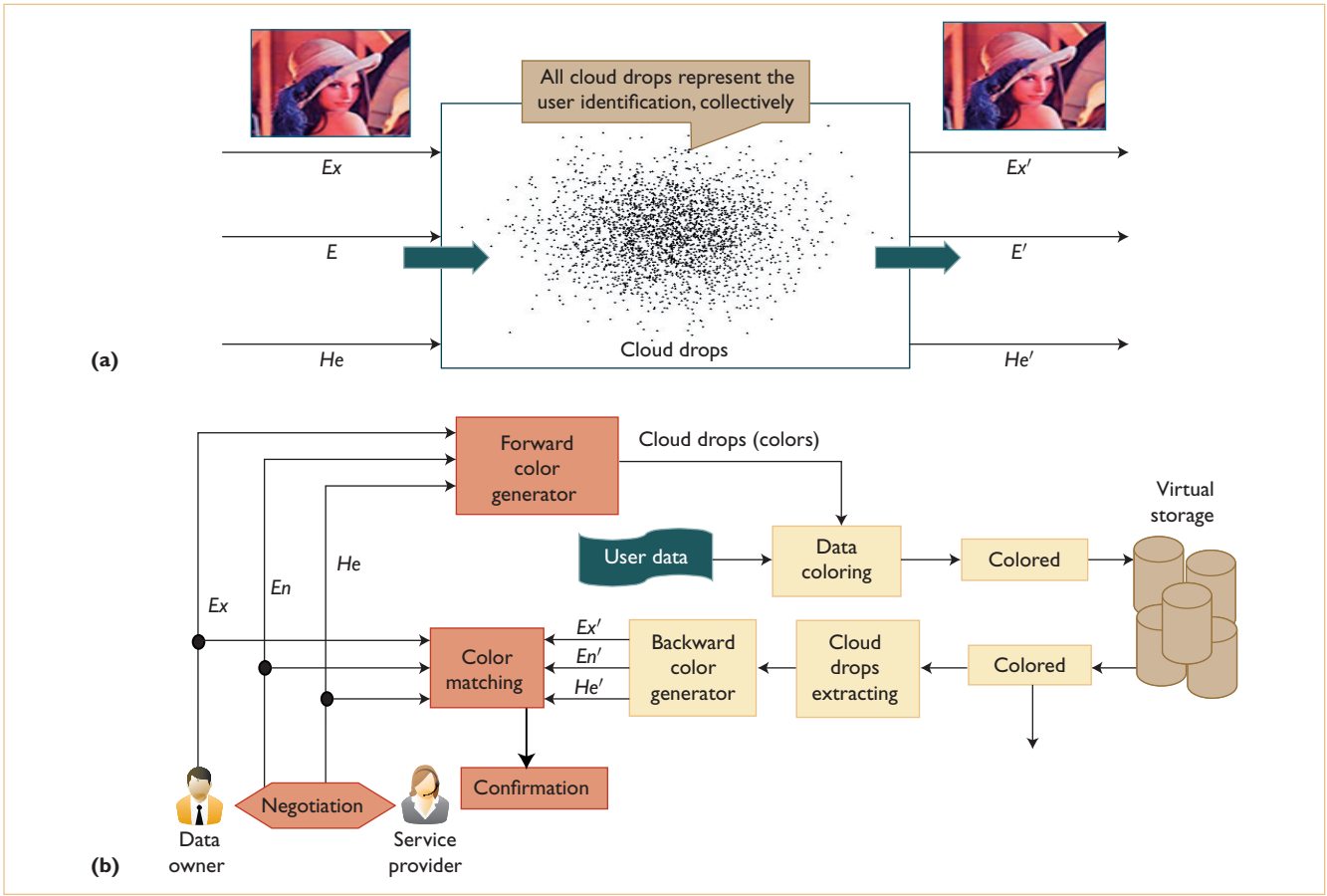

Figure 4. Data coloring using type-2 fuzzy logic. This coloring method enables trust management at various security clearance levels in an open data center. We can see (a) forward and backward data coloring processes by adding or removing unique cloud drops (colors) in data objects. We also demonstrate (b) data coloring and user identification color matching through trust negotiation.

tualized data centers and prevents data loss for other tenants. Users must use cross certificates to delegate trust across public-key infrastructure (PKI) domains for data centers. Trust negotiation among different certificate authorities (CAs) resolves policy conflicts.

Worm containment and DDoS defense. Internet worm containment and distributed defense against DDoS attacks are necessary to insulate infrastructure from malware, trojans, and cyber criminals. This demands that we secure federated identities in public clouds.

Reputation systems for data centers. We can build reputation systems using peer-to-peer (P2P) technology or a hierarchy of reputation systems among virtualized data centers and distributed file systems (see Figure 3). In such systems, we can protect intellectual copyright using proactive content poisoning to prevent piracy.[8] We discuss using reputation systems in more detail shortly.

Data coloring. Our architecture uses data coloring at the software file or data object level. This lets us segregate user access and insulate sensitive information from provider access, as Figure 4 shows. We discuss this method in more detail shortly.

Defense of Virtualized Resources

Virtualization enhances cloud security. First, VMs add an additional layer of software that could become a single point of failure. That is, virtualization lets us divide or partition a single physical machine into multiple VMs (as with server consolidation), giving each VM better security isolation and protecting each partition from DDoS attacks by other partitions. Security attacks in one VM are isolated and contained — VM failures don't propagate

to other VMs. A hypervisor provides the same visibility as the guest OS but with complete guest isolation. This fault containment and failure isolation VMs provide allows for a more secure and robust environment.

Furthermore, a sandbox provides a trusted zone for running programs.[5] It can provide a tightly controlled set of resources for guest OSs, which lets us define a security testbed on which to run untested code and programs from untrusted third-party vendors. With virtualization, the VM is decoupled from the physical hardware; we can represent it as a software component and regard it as binary or digital data. This implies that we can save, clone, encrypt, move, or restore the VM with ease. VMs also enable higher availability and faster disaster recovery.

Live Migration and Open Virtual Format

Live migration occurs when we move a VM from one host to another with minimum downtime. Christopher Clark and his colleagues suggested using live migration of VMs for securing cloud platforms to recover from failures or disasters.[9] We suggest live migration of VMs specifically designed for building distributed IDSs (DIDSs). CPSs can deploy multiple IDS VMs at various resource sites, including data centers. DIDS design demands trust negation among PKI domains. Providers must resolve security policy conflicts at design time and update them periodically. A defense scheme is needed to protect user data from server attacks. Additionally, users' private data must not be leaked to other users without permission. To address these issues, Google's platform essentially applies in-house software, whereas Amazon EC2 applies the HMEC standard and X.509 certificates to secure resources.

Once users move data into the cloud, they can't easily extract their data and programs from one cloud server to run on another. This leads to a data lock-in problem. One possible solution is to use standardized cloud APIs. This requires building standardized virtual platforms that adhere to the *Open Virtual Format* (OVF) — a platform-independent, efficient-to-implement, extensible, and open format for VMs. Adhering to OVF would enable efficient security software distribution, facilitating VM mobility. Using OVF, users can move data from one application to another with much reduced risk of data loss.

Reputation-Guided Data-Center Protection

In the past, most reputation systems were designed for P2P social networking or online shopping services.[10,11] We can convert such systems to protect cloud platform resources or user applications on the cloud. A centralized reputation system is easier to implement but demands more powerful and reliable server resources. Distributed reputation systems are more scalable and reliable for handling failures. The reputation system we propose can help providers build content-aware trusted zones using the VMware vShield and the RSA DLP package for data traversing monitoring.[6]

Reputation represents a collective evaluation by users and resource owners. Researchers have proposed many reputation systems in the past for P2P, multi-agent, or e-commerce systems. To support trusted cloud services, we suggest building a *trust-overlay network* to model the trust relationships among data-center modules. Runfang Zhou and Kai Hwang first introduced the idea of a trust overlay for e-commerce.[11] We can structure the overlay with a distributed hash table (DHT) to achieve fast aggregation of global reputations from numerous local reputation scores. Here, we extend the design to have two layers of trust overlays (see Figure 3).

At the bottom layer is the trust overlay for distributed trust negotiation and reputation aggregation over multiple resource sites. This layer handles user or server authentication, access authorization, trust delegation, and data integrity control. The upper trust overlay deals with worm signature generation, intrusion detection, anomaly detection, DDoS defense, piracy prevention, and so on. These two layers facilitate worm containment and IDSs to protect against virus, worm, and DDoS attacks. The content-poisoning technique Xiaosong Lou and Hwang present for copyright protection in P2P networks[8] is also reputation-based. We can easily extend this protection scheme to stop copyright violations in a cloud environment surrounding multiple data centers.

Data Coloring and Software Watermarking

Given cloud computing's use of shared files and datasets, an adversary could compromise privacy, security, and copyright in a cloud computing environment. We want to work in

a trusted software environment that provides useful tools for building cloud applications over protected datasets. In the past, watermarking was mainly used for digital copyright management. Christian Collberg and Clark Thomborson have suggested using watermarking to protect software modules.[12] The trust model Deyi Li and his colleagues propose offers a second-order fuzzy membership function for protecting data owners.[13] We extend this model to add unique data colors to protect large datasets in the cloud. We consider cloud security a community property. To guard it, we combine the advantages of secured cloud storage and software watermarking through data coloring and trust negotiation. Figure 4 illustrates the data-coloring concept. The woman's image is the data object being protected.

Figure 4a shows the forward and backward color-generation processes. We add the cloud drops (data colors) into the input photo (left) and remove color to restore the original photo (right). The coloring process uses three data characteristics to generate the color: the expected value (*Ex*) depends on the data content, whereas *entropy* (*En*) and *hyperentropy* (*He*) add randomness or uncertainty, which are independent of the data content and known only to the data owner. Collectively, these three functions generate a collection of cloud drops to form a unique "color" that providers or other cloud users can't detect. Additional details about this cloud watermark scheme are available elsewhere.[13,14]

We can use data coloring at varying security levels based on the variable cost function applied. We can apply the method to protect documents, images, video, software, and relational databases. Figure 4b shows the details involved in the color-matching process, which aims to associate a colored data object with its owner, whose user identification is also colored with the same *Ex*, *En*, and *He* identification characteristics. The color-matching process assures that colors applied to user identification match the data colors. This can initiate various trust-management events, including authentication and authorization. Virtual storage supports color generation, embedding, and extraction.

Combining secure data storage and data coloring, we can prevent data objects from being damaged, stolen, altered, or deleted. Thus, legitimate users have sole access to their desired data objects. The computational complexity of the three data characteristics is much lower than that performed in conventional encryption and decryption calculations in PKI services. The watermark-based scheme thus incurs a very low overhead in the coloring and decoloring processes. The *En* and *He* functions' randomness guarantees data owner privacy. These characteristics can uniquely distinguish different data objects.

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. In the future, we expect that *security as a service* (SECaaS) and *data protection as a service* (DPaaS) will grow rapidly. These are crucial to the universal acceptance of Web-scale cloud computing in personal, business, finance, and digital government applications. Internet clouds demand that we globalize operating and security standards. The interoperability and mesh-up among different clouds are wide-open problems. Cloud security infrastructure and trust management will play an indispensable role in upgrading federated cloud services.

Acknowledgments

We thank Sameer Kulkarni, Zhongyuan Qin, and Kaikun Dong of the University of Southern California and Yuchao Liu, Wen He, and Zhiwei Yu of Tsinghua University for their assistance in plotting Figure 1 and Figures 4a and 4b, respectively.

References

1. K. Hwang, G. Fox, and J. Dongarra, *Distributed Systems and Cloud Computing: Clusters, Grids/P2P, and Internet Clouds*, Morgan Kaufmann, to appear, 2010.
2. K. Hwang, S. Kulkarni, and Y. Hu, "Cloud Security with Virtualized Defense and Reputation-Based Trust Management," *IEEE Int'l Conf. Dependable, Autonomic, and Secure Computing* (DASC 09), IEEE CS Press, 2009.
3. J. Nick, "Journey to the Private Cloud: Security and Compliance," tech. presentation, EMC, Tsinghua Univ., 25 May 2010.
4. S. Song et al., "Trusted P2P Transactions with Fuzzy Reputation Aggregation," *IEEE Internet Computing*, vol. 9, no. 6, 2005, pp. 24–34.
5. "Security Guidance for Critical Areas of Focus in Cloud Computing," Cloud Security Alliance, Apr. 2009; www.cloudsecurityalliance.org/guidance/csaguide.v2.1.pdf.

6. T. Mather, S. Kumaraswamy, and S. Latif, *Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance*, O'Reilly Media, 2009.
7. J. Rittinghouse and J. Ransome, *Cloud Computing: Implementation, Management and Security,* CRC Publisher, 2010.
8. X. Lou and K. Hwang, "Collusive Piracy Prevention in P2P Content Delivery Networks," *IEEE Trans. Computers,* July 2009, pp. 970–983.
9. C. Clark et al., "Live Migration of Virtual Machines," *Proc. Symp. Networked Systems Design and Implementation*, 2005, pp. 273–286.
10. L. Xiong and L. Liu, "PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities," *IEEE Trans. Knowledge and Data Eng.,* July 2004, pp. 843–857.
11. R. Zhou, and K. Hwang, "PowerTrust: A Robust and Scalable Reputation System for Trusted Peer-to-Peer Computing," *IEEE Trans. Parallel and Distributed Systems*, Apr. 2007, pp. 460–473.
12. C. Collberg and C. Thomborson, "Watermarking, Tamper-Proofing, and Obfuscation-Tools for Software Protection," *IEEE Trans. Software Eng.*, vol. 28, 2002, pp. 735–746.
13. D. Li, C. Liu, and W. Gan, "A New Cognitive Model: Cloud Model," *Int'l J. Intelligent Systems*, Mar. 2009, pp. 357–375.
14. D. Li and Y. Du, *Artificial Intelligence with Uncertainty*, Chapman & Hall, 2008.

Kai Hwang is a professor of computer engineering at the University of Southern California and an IV-endowed visiting professor at Tsinghua University, China. He specializes in computer architecture, parallel processing, Internet security, and cloud computing. Hwang has a PhD from the University of California, Berkeley. He's the founding editor in chief of the *Journal of Parallel and Distributed Computing* and a fellow of IEEE. Contact him at kaihwang@usc.edu.

Deyi Li is a professor of software engineering at Tsinghua University, China, and heads the Information Science Directorate of the Natural Science Foundation of China. His current research interests include networked data mining, artificial intelligence with uncertainty, and cloud computing. Li has a PhD in computer science from Heriot-Watt University, UK. He's a member of the Chinese Academy of Engineering. Contact him at lidy@cae.cn.