Self Hosting vs. Cloud Hosting: Accounting for the security impact of hosting in the cloud

David Molnar, Stuart Schechter
Microsoft Research
{DMolnar,StuS}@microsoft.com

Abstract

As more and more organizations consider moving their applications and data from dedicated hosting infrastructure, which they own and operate, to shared infrastructure leased from 'the cloud', security remains a key sticking point. Tenants of cloud hosting providers have substantially less control over the construction, operation, and auditing of infrastructure they lease than infrastructure they own. Because cloud-hosted infrastructure is shared, attackers can exploit the proximity that comes from becoming a tenant of the same cloud hosting provider. As a result, some have argued that that cloud-hosted infrastructure is inherently less secure than the self-hosted infrastructure, and that it will never be appropriate for high-stakes applications such as health care or financial transaction processing.

We strive to present a more balanced treatment of the potential security impacts of transitioning to cloud-hosted infrastructure, surveying both the security costs and security benefits of doing so. The costs include exposure to new threats, some of which are technological, but many others of which are contractual, jurisdictional, and organizational. We also survey potential countermeasures to address these threats, which are also as likely to be contractual or procedural as technological. Transitioning to a cloud-hosted infrastructure may also have security benefits; some security measures have high up-front costs, may become affordable when amortized at cloud scale, and impact threats common to both cloud- and self-hosted infrastructures.

1 Introduction

Behind the hype surrounding 'cloud computing', and competing definitions of the term, are compelling economic forces driving changes in the infrastructure used to host organizations' applications and data. Instead of owning and operating infrastructure themselves, organizations may now lease shared resources from 'clouds', effectively becoming infrastructure *tenants* rather than owners. The resource-elasticity offered by cloud providers eliminates the up-front costs of building a self-hosted infrastructure and removes delays by allowing tenants to scale up their resources on demand. Cloud-hosting also offers cost savings achieved through economies of scale: cloud providers receive bulk prices for components, can better utilize specialized staff, use lower aggregate spare-capacity through sharing, and amortize of the up-front costs of building and administering data centers over many tenants [5].

Standing in the way of the potential savings achievable through cloud-hosting are concerns about security. In April 2009, Cisco CEO John Chambers called the security implications of cloud hosting "a nightmare", explaining that "you'll have no idea what's in the corporate data center" [33]. Ron Rivest suggested that the phrase "swamp computing" might better represent the correct mindset in which to examine the security implications of moving to the cloud [45]. Among Bruce Schneier's many cloud computing concerns was that critical data could end up "on some cloud that abruptly disappears because its owner goes bankrupt" [44]. Others fear that as competing providers rush to grab early market share, which is especially valuable given the high switching costs and large scale economies of the cloud hosting business, they will be tempted to adopt a ship-first-secure-later strategy.

Most of these security concerns surrounding cloud hosting are not new, but are already endemic to existing hosting offerings, such as those that offer accounts on shared servers or virtual private servers that run on shared hardware.[1] Other threats, such as the risk that an attack on one tenant will impact another, are already endemic to content distribution networks. What differentiates cloud-hosting providers from traditional hosting providers is their ability to offer scalable resources, purchasable in small time units of time and offered at prices made possible through economies of scale. Whereas virtual private servers target customers seeking to establish a basic web presence or basic email service, cloud-hosting target applications and data would have previously required dedicated data centers. Prospective tenants of cloud-hosting providers thus often have much higher security requirements than those of traditional web hosting providers.

Despite numerous concerns about the security of cloud-hosted infrastructure that are both legitimate and significant, it would be unfair to assume that cloud-hosted infrastructure is inherently less secure than self-hosted infrastructure. Those who argue cloud hosting is inherently less safe inevitably compare it to a security ideal in which organizations that operate and own their own infrastructure have unlimited resources to secure it properly. In reality, securing a hosting infrastructure is expensive and replete with costs that must be expended regardless of scale. A balanced treatment must recognize not only new threats introduced by moving to cloud hosting but also the economies of scale in addressing existing threats endemic to both cloud- and self-hosting. Operating at cloud scale opens the design space for security measures to include solutions not previously feasible: those with up-front costs that are prohibitively expensive below cloud scales, but that achieve net savings over competing solutions by reducing the marginal per-tenant and per-machine costs.

Contributions and scope

We strive to survey the long-term security implications of cloud hosting independent of the constraints of today's implementations.

Our first contribution is to survey and catalog the new threats that are introduced when applications and data are moved to leased/shared (cloud-hosted) infrastructure from owned/dedicated (self-hosted) infrastructure. Many of these threats pertain less to technology than to matters of human resources, incentive alignment, and jurisdiction. While many of these threats have been raised elsewhere, we assemble them together in an accessible manner. We also explore existing technological, organizational, and legal avenues to address cloud computing threats. Finally, we identify security measures that may benefit from the economies of cloud scale, potentially enabling tenants of cloud hosting providers to get more security for their dollar than could be achieved by hosting their own infrastructure.

We have intentionally restricted the scope of this survey to cloud hosting of tenants' applications and data, and not cloud applications in which the the hosting and application infrastructure are built entirely by a third party (e.g. Google's Docs, Office Live, DropBox, Flickr). While cloud hosting and cloud applications are often treated alongside each other in discussions of 'cloud computing' trends and security threats, the services and their security implications are quite different.

We have also intentionally chosen *not* to build mathematical formulas or models for the decision to move to cloud hosting. This choice is cost/benefit decision, and while we seek to provide insight by enumerating and examining these costs and benefits, once these factors are quantified the accounting itself is straightforward. We believe there is little further to be gained (and a great deal of clarity and generality to be lost) from the introduction of mathematical decision models and the simplifying assumptions required to make general claims about these decisions.

While we enumerate numerous threats, countermeasures, and sources of economies of scale in cloud-infrastructure security, omissions are sure to be discovered in each of these sets. This is a working document, and one that we expect to change both in response to feedback from the workshop, the realization of unforseen threats, and the development of new security schemes.

[1] The WEIS conference websites have run on a virtual private server since the 2006 conference.

2 Threats introduced by cloud hosting

We examine two sets of threats that arise from moving from self-hosted to cloud-hosted infrastructure: threats introduced because resources that were once owned and operated by the tenant will now be leased from a cloud provider, and threats introduced because leased resources are shared with other – potentially malicious – tenants. As mentioned in the introduction, many of these threats are not new but are endemic to existing shared hosting products such as virtual private servers.

2.1 From owning to leasing

A tenant who formerly owned its infrastructure will now rely on the cloud provider to provide and secure the physical plant, hardware, software, and the administrative infrastructure. The following threats are those in which the cloud hosting provider fails to protect the tenant's infrastructure as well as the tenant would like.

Threats to infrastructure assembly

Physical infrastructure
: The physical plant (building, power, backup, air conditioning, etc.), computing hardware, and network selected by the cloud provider may not be sourced or provisioned to the tenant's standards, may be maliciously designed to subvert security, or may be compromised by a third party. Sourcing is already a problem as evidence by a case in which counterfeit chips were been sold to the U.S. Navy [34]. In 2009 the FBI disrupted a counterfeiting ring for Cisco equipment [43], which shows this happens in practice. An example of why sourcing is a security concern, King et al. have demonstrated how back doors can be hidden in hardware via small modifications [28].

Software infrastructure
: The software infrastructure - including the OS or cloud infrastructure services such as database services - may fail to meet the standards promised to the tenant, may be intentionally designed to subvert security, or be compromised by a third party.

Human infrastructure
: The employees selected by the cloud provider to administer the infrastructure, and who may thus be able access to the tenant's resources, may not be screened or audited to ensure their security skills or trustworthiness meets the standards promised to the tenant. Regardless of screening, they pose an insider threat in that they may use their administrative rights to compromise the tenant's security.

Contractual threats

Cost-overrun attacks
: While the ample resources of cloud hosting may prevent some denial-of-service attacks from taking tenants' services offline, tenants may still face orders-of-magnitude larger bills if stuck paying for the resources required to respond to attackers' requests. Thus, a thwarted denial-of-service attack may become a cost-overrun attack.

Deceptive billing
: The cloud provider may bill the tenant for more resources than are actually consumed or, more subtly, may cause its infrastructure to run less efficiently to increase consumption of billable resources.

Captivity
: High switching costs may make tenants captives of the cloud provider, which may use its increased bargaining power to the detriment of its tenants.

Bankruptcy
: The cloud service provider may go out of business. Tenants will lose access to applications and data that they are unable to move before the cloud provider's infrastructure goes offline. Infrastructure that stores tenant data may become the property of the cloud provider's creditors.

Legal and jurisdictional threats

Indirect legal coercion

Cease and desist letters and injunctions that would previously been directed to the tenant may now be directed at the cloud provider, threatening to hold the cloud provider accountable for the tenant's actions. The cloud provider may yield to such pressures more easily than the tenant would, in part because the cloud provider may have more assets at risk and in part because yielding may result in smaller direct losses for the provider than the tenant.

Secret search

A sealed search warrant served at the cloud provider may allow law enforcement to search the tenant's systems while forbidding the cloud provider from notifying the tenant that a search took place. For example, in the United States the Federal Bureau of Investigation may perform secret search by issuing a National Security Letter request. The authority to issue these requests is used frequently, as the Justice Department's Office of the Inspector General estimated that $47,000$ National Security Letter requests were issued in 2005 alone [49].

Direct jurisdictional exposure

The cloud provider may move data or applications into jurisdictions that expose the tenant to new opportunities for surveillance or interference in such forms as regulation or taxation.

Indirect jurisdictional exposure

Even if applications and data remain within the tenant's preferred jurisdiction, governments from other jurisdictions with power over the cloud provider may attempt to exert this power to the detriment of the tenant. For example, a government could pressure a cloud provider with a heavy investment in its country to spy on a tenant regardless of whether the tenant or its data resided within the government's jurisdiction. For an analog in banking, consider the impact that the sizable U.S. business presence of Swiss bank UBS on the firm's decision [48] to comply with the demands of U.S. law enforcement to reveal information about funds held in the Swiss accounts of U.S. customers.

2.2 From dedicated to shared infrastructure

When infrastructure is shared, tenants must be confident that mechanisms are in place to protect tenants from one another. Furthermore, sharing resources may negatively impact availability and, when tenants may be identified by the resources they share, reputation as well.

Threats from other tenants

Direct breach

Malicious tenants may pierce hardware, software, or network isolation boundaries to compromise the confidentiality or integrity of another tenant's data, code, or communications.

Side channel attack

Malicious tenants may use side channel attacks, such as those that examine cache behavior[47], to read other tenants' private data.

Denial of resources

Malicious tenants may compromise availability by consuming too many resources or exploiting vulnerabilities exposed through tenant-accessible APIs. For example, Vadhat describes one such vulnerability in the Xen hypervisor [20, 50].

Resource theft

Malicious tenants may steal shared resources (e.g. compute time), or find ways to charge resources to other tenants.

Collateral damage to shared reputation

The actions of malicious tenants may damage the reputation of infrastructure components or other tenants, such as could occur when shared IP ranges are used to send spam. For example, in 2009 the spam blacklist provider Spamhaus added all nodes on Amazon EC2 to its e-mail blacklist [8]; EC2 users who wish to send e-mail must now apply specially to Amazon to use a node not on the blacklist [3].[2]

Legal and jurisdictional threats

Jurisdictional collateral damage

Law enforcement may shut down a data center or cloud provider to target one or more tenants, causing all other tenants to lose service. For example, the US Federal Bureau of Investigation shut down an entire co-location facility and seized equipment as part of an investigation of a single tenant [54].

Threats to the availability and cost of shared resources

Underprovisioning

When infrastructure is shared, tenants must rely on the cloud provider to properly estimate their collective peak resource needs and provision appropriately. Underprovisioning may cause performance degradation or outright failure at times of peak demand. Cloud providers that fail to account for correlated usage bursts (economies of scale in provisioning require assuming some level of independence in resource needs) might unintentionally underprovision. So might a provider who fails to anticipate the exponential growth of a tenant; Imagine a cloud provider that discovers too late that it is hosting the Internet's next Google or Facebook. Competitive pressures might lead cloud providers to intentionally provision fewer spare resources than their tenants expect, as every unused resource cuts into profit margins. Customers and monitoring firms are already measuring performance degradation in Amazon's EC2 service and interpreting it as evidence of underprovisioning [52, 11].

Collateral denial of shared resources

A denial-of-service attack on one tenant, or on the cloud provider itself, may impact other tenants who rely on shared resources targeted in the attack.

Diminished audit, detection, and incident response capabilities

Forensic restrictions

If an outside party breaches a tenant via cloud-level infrastructure components that the tenant is forbidden from monitoring, it may be impossible for the tenant to identify the cause of the breach and prevent it from happening again. Because infrastructure is shared with other mutually-suspicious tenants, tenants may be expressly forbidden to investigate infrastructure components as doing so might violate the security guarantees made to other tenants.

3 Countermeasures

We now summarize available countermeasures to the threats in Section 2. For leasing-induced threats, in which tenants fear that a cloud provider will not be sufficiently diligent in protecting infrastructure, it should come as no surprise that countermeasures focus on auditing mechanisms and, where possible, restoration of policy control to tenants. For sharing-induced threats, technology issues are more prevalent but policy issues remain pervasive.

In evaluating the costs of these countermeasures, it is important to consider which portions are fixed and which increase with the number of applications, tenants, or machines. Cloud providers can tolerate countermeasures with high up-front costs, because these costs can be amortized over many customers. On

[2]Our reliance on Amazon EC2 for real-world examples of cloud-hosting problems results entirely from EC2's share of the market. We in no way intend to imply that these problems are exclusive to EC2.

the other hand, countermeasures with high and unavoidable per-tenant variable costs may be prohibitively expensive, especially if most tenants are unwilling to pay a premium for them.

3.1 From owning to leasing

Threats to infrastructure assembly
Physical and Software infrastructure

For self-hosted infrastructure, the quality, safety, and integrity of owned hardware and software is often managed via perimeter security. Where possible, *intake controls* verify the integrity of equipment and software as they arrive. Sample components may be tested before purchase and purchased components may be tested again at random. To implement properly, intake-control processes are expensive to set up and have high marginal costs that grow with the amount of hardware and software entering the system.

Tenant auditing of cloud providers via intake-controls would be impractical, as tenants use only a small fraction of cloud providers' total equipment and would thus be forced to spend most of their time auditing equipment they will never use. What's more, it would be hard to set up a process which a tenant could have access to audit hardware without introducing the risk that this tenant, if malicious, could use this access to compromise the hardware. One workaround is to rely on a trusted third party to perform audits, but this may still not be an attractive option one given the sheer scale of cloud data centers.

An attractive alternative is to support auditing of hardware and software via remote hardware/software attestation, a hardware feature already shipping in today's Trusted Platform Modules (TPMs). These modules, paired with certificates from hardware manufacturers that attest to the source of the hardware, allow the hardware to cryptographically sign statements about its properties and the properties of the software it is running. Using these statements, a tenant or auditor could verify that an application is indeed running on the contractually specified hardware and software (bios, hypervisor, OS, on up). To protect tenants' data from malicious hardware components with access to memory or disks, data can be stored in memory and on disk in encrypted form, with keys leveraging the 'sealed storage' feature of TPMs. For a survey on technologies for remotely verifying properties of hardware and software, see Parno et al. [38].

The use of TPMs and similar hardware mechanisms to automate the audit of the integrity of hardware and software has attractive economies of scale. The cost to verify the software stack, and to build the attestation process, is fixed in the number of machines and tenants. Each tenant incurs only a small marginal computational cost (a handful of signature verifications) to audit the hardware and software infrastructure each machine it uses.

Alas, memory encryption is not yet available on commodity CPUs and today's TPMs provided limited protection against hardware attacks, such as the sand-and-scan attack used by Tarnovsky to extract secret keys from a TPM [46]. Further research is needed to make TPMs more resilient to attack without making them prohibitively expensive.

Still, TPMs may remain attractive when compared to intake controls, which are also unlikely to detect the presence of a single compromised machine. Even if detection were straightforward, inspecting every incoming component would likely be impractical. In the long run, solutions that leverage attestation, sealed storage, and memory encryption may well provide a level security that is neither technically nor economically feasible with intake controls.

Even if TPMs or other hardware mechanisms can provide reasonable assurances as to the origin and condition of the hardware and software being provided, they alone cannot ensure that this software is secure. Testing of the design- and code-level security of a hypervisor is expensive, and security-sensitive tenants may demand testing by one or more third parties. The cost of auditing code is high and varies in the size of the code and frequency with which it is updated. However, the cost of verifying that a code region matches an audited code region is negligible. Thus, the cost of code audits are fixed with respect to the number of machines and tenants, and can thus be amortized over all the machines running the audited code.

Human infrastructure

Human-level infrastructure must also be audited. Administrators and other employees in positions of trust may be required to undergo background checks that go beyond simple criminal and financial background checks. Pre-employment verification processes are expensive to design and have high marginal costs. However, cloud providers will be able to amortize the fixed cost of establishing hiring practices over a larger number of employees than providers of owned infrastructure. While the costs of screening increase with the number of employees to be screened, cloud hosting providers require fewer trusted staff per machine than those operating at smaller scale.

To protect against insider attacks, cloud-infrastructure may be designed to limit administrators' access to potentially-confidential tenant data. Since such restrictions could impede administrators ability to perform important tasks, such as penetration testing or forensic investigation, procedures would still be required to deactivate them.

Another approach to protecting against administrative malice or incompetence is to put into place administrative controls that require administrative actions to be approved by a second administrator, or even be performed by multiple administrators. These control mechanisms might also randomize the assignment of administrative tasks to administrators, preventing administrators from seeking out tasks that would confer access to a targeted tenant. Such technical mechanisms to support support multiple-person control for computer system administration have been explored by Potter, Bellovin, and Nieh [40].

The design, implementation, and testing of administrative controls will certainly incur up-front costs. Once controls are in place the variable costs associated with administrative tasks will also increase. However, these added variable administrative costs are likely to be more than offset by the savings attained from the economies of scale of administering a large, relatively homogenous, infrastructure. Oracle reports that their cloud data centers require one tenth as many administrative staff per machine as are required for typical data centers in higher education [36]. Even if all administrative tasks had to be duplicated by two administrators (doubling the administrative costs) a cloud-hosted infrastructure might still come out ahead.

Contractual threats

Cost-overrun attacks

One way to address cost-overrun attacks is to allow tenants to set quotas to bound the rate at which an application or tenant can consume billable resources. This allows the tenant to choose the lesser of two evils, losing service availability or consuming resources it cannot afford.

Another redress to tenants' concerns of cost-overrun attacks is for cloud-providers to absorb the bulk of the resource costs incurred by attacks. The costs of spare compute and storage resources are, after all, already sunk. The economic viability of a large cloud provider like Amazon, Google, or Microsoft is unlikely to be greatly impacted by operating at peak resource consumption during an attack, and so they are well positioned to insure their tenants against cost-overrun attacks in this way. Cloud providers who insure their tenants' against cost-overruns will have an extra incentive to build infrastructure that uses resources efficiently when under attack.

Alas, with insurance comes adverse selection. Insuring against cost-overrun attacks will attract a disproportionate share of denial-of-service-prone tenants (e.g. gambling or auction websites). Insured tenants will have less incentive to engineer their application-level infrastructure to use resources efficiently when under attack. Cloud providers that insure tenants against cost-overrun attacks may thus require more spare resources than those that do not. What's more, each attack will incur direct costs that result from increased power consumption, bandwidth utilization, and incident response time. Cloud providers may not want to provide full insurance to tenants, and may want to adjust the cost of insurance based on the business of the tenant and its performance under load testing.

Deceptive billing

Deceptive billing is an attractive target for auditing, either directly or via attestation-based auditing solutions.

If the cloud provider offered tenants the ability to run their own test infrastructure, tenants would be able to evaluate the performance of their applications to estimate how much they should cost to run at cloud scale. Auditors may run similar tests to ensure that resource usages claimed by the cloud provider match local tests.

An attestation-based approach to resource accounting could leverage a third party to audit the software used to account for tenants' resource consumption, rather than auditing the resource usage bills. Tenants would require that all bills be itemized by machine, with each machine signing its billing statement along with a attestation chain that verifies which billing software was running. If the auditor and tenant are confident that the accounting software honestly reports resource usage, they can have greater confidence in the integrity of the bills they receive.

Captivity

The most direct way to lower captivity concerns is to reduce switching costs. Switching costs can be reduced by making cloud infrastructure more homogenous, such as by using the same cloud hosting APIs and tools as other providers. Alas, such a solution is unlikely as cloud providers use their infrastructures to differentiate themselves from competitors and keep their businesses from becoming commodities. What's more, providers who purchased software infrastructure would risk becoming captives of the software developer. What is more likely to occur is that certain components of cloud providers solutions (e.g. VMs, databases) will become semi-standardized, lowering but not removing switching costs. Alternately, intermediate layers may be used to abstract away provider-specific APIs, lowering switching costs but sacrificing access to provider-specific features. One example of such a layer is the Simple Cloud API, which currently abstracts blob storage mechanisms [53].

Given that many cloud infrastructures will inevitably have high switching costs, captivity concerns may be best addressed through the use of long term contracts renewable far in advance of expiration. One complication to writing such contracts is that it is impossible to predict the cost of providing service, or the competitive cost of service, over a long time horizon. Thus, long-term contracts will need to be written in terms of indexed storage and computation costs. This will help ensure that tenants will benefit from technological progress without putting cloud providers' businesses at risk should technology fail to progress at the expected pace.

Bankruptcy

Bankruptcy poses a threat to both the availability of the infrastructure required to run an application and the data within the application itself. Whereas data back-up may be challenging for certain data-intensive applications, backing up application infrastructure poses an even greater challenge.

Cloud providers may need to structure tenant contracts to guarantee that, in the event of bankruptcy, tenants will automatically acquire the rights to any custom software required to replicate the provider's infrastructure. Cloud providers must also structure contracts with tenants, investors, and creditors, to give each tenant priority rights to a proportional share of the provider's hardware and other physical infrastructure.

Another route to addressing the threat of bankruptcy is to obtain an insurance contract that will guarantee the funds required to operate the infrastructure for a period following bankruptcy. To ensure greater warning in advance of bankruptcy, contracts could also require regular financial audits that track the viability risks to the cloud providers' business. Insurance policies might also be used to ensure that a tenant's switching costs would be covered in the event of bankruptcy. The cost of such policies will surely depend on the expected size of these switching costs.

Legal and jurisdictional threats
Indirect legal coercion, Secret search, Direct and Indirect jurisdictional exposure

Indirect legal coercion is a problem that ISPs and traditional hosting companies are already familiar with, as they are already exposed to takedown notices and injunctions that relate to client data. For example, hosting providers in the United States regularly receive requests under the Digital Millennium Copyright Act (DMCA) to remove allegedly copyright-infringing material. One way to address this threat is to clarify, within the provider-tenant contract, the process for responding to injunctions, what legal costs will be incurred by the cloud provider, and how disagreements regarding these terms will be arbitrated.

Cloud providers can deploy software tools to enable tenants to manage where their applications and data are hosted and how data is stored, helping to manage direct jurisdictional exposure. Technologies such as memory and disk encryption, as well as sealed storage, can be deployed to help tenants comply with existing regulations.

Laws and international agreements may be crafted to reduce the vulnerability of cloud hosting legal coercion and indirect jurisdictional threats (indirect interference, secret search) while providing governments recourse should tenants attempt to use cloud hosting to escape applicable laws. The DMCA, for example, already specifies the steps a hosting provider is required to take to avoid being held liable for a tenant's actions. The prospect of jobs, tax revenue, and lower-latency access to cloud-hosted services would be among the incentives that might lead countries to participate in agreements friendly to cloud-hosting providers.

3.2 From dedicated to shared infrastructure

Threats from other tenants
Direct breach, Side channel attack

Hypervisor protections on the confidentiality and integrity of tenant software and data have been the subject of a great deal of research, both offensive and defensive [30]. Current directions include making hypervisors smaller [31] and easier to verify using formal tools [27, 24]. Microprocessor manufacturers are adding support to simplify and improve the design of hypervisors [13, 12].

One of Intel's motivations for including AES instructions in its 2010 Core series of microprocessors is to address side-channel attacks on cryptographic keys [19], such as the timing attack of Tromer et al. [47] and other timing attacks [29].

One reason for cautious optimism for hypervisor-based isolation is that a key cause of isolation failure, the introduction of buggy communication mechanisms demanded by parties that wish to break isolation boundaries, is unlike to apply. When tenants do communicate, they are more likely to be using existing network protocols than to communicate via the hypervisor. Similarly, shared database or storage services are likely to be accessed via network protocols as well. However, shared networks, storage, and other resources will also be potential sources of isolation failure.

IPsec and virtual private networks are already heavily-relied upon in order to protect data in transit on potentially-unfriendly networks. Since these tools are insufficient to prevent availability attacks on the infrastructure itself, control networks may also be necessary. Building authentication, encryption, and integrity protection into intra-tenant and intra-cloud communications could actually simplify the work of application developers, who today must implement myriad authentication solutions to support different services.

Finally, for those systems that do not use hypervisors or want to add defense-in-depth, application-level sandboxing and API-limitations may further aid isolation, at the cost of interfering with existing tools that may assume access to resources now restricted.

Denial of resources, Resource theft

The threat of other tenant's hogging or stealing resources will also challenge technologists to develop appropriate fairness algorithms and accounting mechanisms. This poses many challenges. For example, if tenants control the TCP stack they will also control the network's fairness mechanisms and the code

that determines whether a transmission was in fact completed. Accounting imprecisions due to coarse measurement units or externalities such as process swap time may be the target of arbitrage by malicious tenants.

Collateral damage to shared reputation

Cloud providers can assist external reputation systems by providing a mapping between communications and tenants. There are various ways to accomplish such a goal. For example, for email a cloud provider could restrict all outgoing port 25 connections, require outgoing mail to go through its own server, and then augment all outgoing mail with a header identifying the tenant. The approach currently taken by Amazon is more restrictive, requiring tenants apply for special permission to make outgoing connections on port 25 [3]. Alteratively, the cloud provider could provide a protocol that enabled outsiders to look up the identity of a tenant by providing the IPs, ports, and times associated with a current (or recent) communication.

Legal and jurisdictional threats

Jurisdictional collateral damage

Cloud-hosting providers benefit from the opportunity to build relationships with law enforcement through recurring interactions. A record of compliance with search warrants, providing law enforcement access to audit logs and data snapshots which can be obtained without disrupting tenants, will reduce the likelihood that law enforcement will attempt to take infrastructure offline.

Threats to the availability and cost of shared resources

Underprovisioning

Attestation-based audit mechanisms could also be used to verify that spare capacity has been provisioned to the levels promised in a service level agreement contract. Resource accounting modules on each system could report total usage over time without revealing anything about the tenants who used those resources. Again, attestation could be used to safeguard against cloud providers' attempts to tamper with resource usage reports. Availability reporting poses a greater challenge than usage reporting. Attestation-based mechanisms to prove that resources were available if needed, at times when they were powered down, are an open research problem.

Attestation-based capacity auditing may not be necessary if spare capacity levels are not made verifiable, but instead guaranteed via penalties paid by the cloud provider to the tenant in the event that resources are unavailable. However, large outages could result in penalties so large as to require the cloud provider to obtain insurance. Insurers would then likely find it necessary to audit spare capacity.

Collateral denial of shared resources

Collateral damage from denial-of-service attacks may be limited through the use of resource quotas. For example, a tenant might have as a default quota that is equal to its average resource consumption plus the resources available when the load on the cloud is at its 99.9^{th} percentile.

Alternatively, when resources are scarce, the cloud provider could calculate ratio of current resource consumption to average resource consumption for each application or tenant. Scarce resources could be granted to those with the lowest utilization ratios. Resources could even be moved away from those with the highest ratios.

Diminished audit, detection, and incident response capabilities

Forensic restrictions

In order to accommodate mutually suspicious tenants, we see few alternatives to relying on cloud providers or collectively trusted third parties to perform any auditing, monitoring, or security response tasks that could impact the security of more than one tenant. The potential risks of relying on cloud service providers

to disclose infrastructure-level problems is illustrated by a data corruption bug in Amazon S3 that was discovered by its users [26]. This particular incident was customer-detectable and verifiable. Amazon thus had little choice but to accept responsibility and respond, identifying the cause as a flaky border router. If the problem had only affected a single customer and could not be so easily reproduced and isolated the response may have been different.

Contractual solutions to overcome forensic restrictions are difficult. No matter how many resources a cloud provider is willing to dedicate to investigating its potential culpability for each tenant's breaches, there may always be tenants who claim that it is not doing enough. Even given the utmost good faith on the part of the provider, it may simply not be economically viable to investigate every issue that customers claim are infrastructure bugs.

Tenants could put contract clauses into place that would allow them to demand cloud providers, or collectively-trusted third parties, perform a forensic investigation of a breach until the source is determined or a maximum budget is reached. If the audit proves the breach occurred within the tenants' infrastructure the bulk of the cost of the investigation would be borne by the tenant. If the cloud provider could not identify the source of the breach the bulk of the cost would be borne by the cloud provider. One appealing feature of such contracts is that they would provide a strong incentive for cloud providers to implement extensive logging capabilities and powerful analysis tools, as these tools we be needed to limit the portion of forensic investigations that it must pay for. Regardless of whether the cloud provider or a third party does the auditing, a tenant would need to trust the that party not to overbill for the service.

4 Security benefits of building infrastructure at cloud scale

Though self-hosted infrastructure may be free from threats specific to cloud-hosted infrastructure, meeting the security expectations of those who who depend on it can prohibitively expensive. Securing a hosting infrastructure has significant costs that are fixed with respect to the number of machines to be secured. Examples of these fixed costs include:

- Assembling a host and network security strategy
- Training staff on the full range of tasks required by the security strategy
- Keeping abreast of new threats and countermeasures
- Developing a relationship with law enforcement

Cloud-infrastructure operators can amortize these fixed costs over a much larger infrastructure than self-hosting organizations can. Staff in cloud hosting providers can become more specialized than their counterparts administering self-hosted infrastructure, allowing them to develop expertise that increases productivity while receiving lower per-employee training.

Managed security solutions already allow owners of self-hosted infrastructure to achieve some of these scale benefits. These managed offerings range from solutions in a box – these boxes may provides firewalls, backup, or spam filtering – to full service security consulting and system monitoring. Alas, managed security solutions may expose their clients to many of the same threats that cloud providers' tenants face. For example, a spam filtering box will have access to the client's network infrastructure and all incoming email, and is susceptible to secret search.

Economics will likely drive cloud-infrastructure operators to provide many of the solutions offered by managed security solutions today. Since the cloud provider must already be trusted with tenants' applications and data, tenants can obtain these services without growing their trusted employee and organization base. For example, a cloud-hosting operator, who already controls your network, needs no additional privileges to filter incoming traffic on port 25. What's more, security features built into the infrastructure can be cheaper to integrate into an application than those that require new components to be installed or that have APIs that may not be customized to the infrastructure. Once a cloud-infrastructure provider incurs the cost

to develop a managed security solution for a security-conscious customer, the marginal cost to deploy the feature to other tenants is often negligible.

Some examples of security features that could be built into clouds, some of which are already present in hosting tools such as CPanel [14], are:

- Network and operating system auditing tools
- Tracking of all installed software, publishers, versions, and patch levels
- Credit card storage and fraud detection
- Public/private key generation, certificate generation, and storage
- Automatic authentication and protection of intra-tenant network communications
- Secure (append-only) logging of system events
- Spam filtering
- Password hashing and storage
- CAPTCHA generation and verification
- Software widgets such as password-strength meters

Many of these features would not be affordable if tenants had to cover the up-front costs, but become affordable if tenants only have to cover their share of the marginal costs. The leads to positive externalities whenever a security-conscious prospective tenant demands a new security feature.

Another benefit of building security features into the cloud infrastructure is to leverage data from multiple tenants. For example, when monitoring tools detect a new attack against one tenant the monitoring team and system will be more alert to similar attacks against other clients. Such systems must be designed not to restrict undesirable information from leaking from one tenant to the other. Still, reputation systems that identify bots, spammers, and other malicious activity can benefit from a wealth of data and few tenants would have a reason to opt out of providing it. Employees of the cloud provider entrusted to perform forensics on one tenant's compromised system may leverage what they learned from inspecting others' systems without leaking data. Bundling managed security into the cloud helps to overcome the free-riding problem in security data sharing identified by Gordon, Loeb, and Lucyshyn [17].

Tracking jurisdictional threats and keeping up with myriad laws and regulations is an expensive task, but one that has economies of scale. If infrastructure within the cloud providers' purview can be certified to provide compliance with security or privacy regulations, cloud providers may be able to assist with compliance at cloud scale. Cloud providers may also be able to assist in disseminating information that allows tenants to evaluate jurisdictional risks and keep up with local laws.

The economies of scale exhibited by these security solutions explain why existing managed security solutions are a big business, despite scale limitations that result from having clients in distributed locations with heterogenous infrastructures. Gartner estimates the total managed security service provider market had revenues of roughly $500 million in 2009. Major telecommunications carriers such as BT (via its acquisition of Counterpane) and Verizon now offer these services [41].

As we noted previously, cloud-hosting providers benefit from the opportunity to build relationships through their recurring interactions with regulators and law enforcement. If law enforcement officials know the cloud provider can guarantee them access to audit logs and data snapshots even if a tenant turns out to be malicious, they are less likely to take a tenant – or an entire data center! – offline in order to protect an investigation. More strategically, cloud providers can take an active role in shaping compliance and legal regimes to favor their tenants. The sheer scale of cloud-hosting providers may make their security practices defacto best practices. Since liability law faults those who fail to take precautions that other reasonable parties would take, joining the herd that has put its security in the hands of the cloud may actually provide protection against liability suits.

5 Which security measures will go into the cloud

Many of the cloud-hosting concerns we've discussed are the evolutionary descendants of threats that predate cloud hosting. Regardless of whether hosting infrastructure is owned or entrusted to a cloud provider, critical hardware and software components will be sourced from third parties and staff may not be trustworthy. Reliance on component APIs may make tenants captives of these component providers. While component makers may not be able to overbill consumers directly, components may be designed to intentionally underperform to increase sales. Secret search already happens at third-party data centers. Underprovisioning is already a common concern of web hosting clients, who accuse providers of being unable to deliver the "unlimited" resources that are advertised [35].

Given that a change in hosting strategy has myriad impacts on how organizations develop and administer application-level infrastructure, as well as how they staff themselves, security alone is unlikely to motivate a move to cloud hosting. More likely, security will be a factor that will prevent, but not incite, a prospective tenant from leasing space from a cloud-hosting infrastructure. Low up-front costs and elasticity are more likely to drive prospective tenants. In developing a security strategy, cloud hosting providers are unlikely to set an initial goal of achieving a net increase in security over self-hosting, and more likely to seek to remove any real or perceived net security loss that could act as so called 'deal-breakers'. For example, disk encryption may be given top priority to assure prospective tenants with concerns related to data breach notification laws.

Cloud providers will implement those sets of security measures that cost less to put into place than the value of the new tenants these features will attract. These total cost of these measures includes not only the up-front costs and the variable costs of servicing those tenants who demand these features, but also any per-tenant or per-machine variable costs that are incurred serving tenants who do not consider these features essential. Tenants who consider a security feature valuable, but not essential, still benefit when low per-tenant costs make it attractive for cloud providers to bundle them in.

If there are security measures that are not desired by the bulk of the cloud provider's customer base, but have large per-tenant variable costs, cloud providers may find it necessary to abandon the features. Alternatively, they may partition the cloud so that it need not deploy the feature universally and price discriminate—essentially creating two clouds and sacrificing some economies of scale.

The most promising research opportunity in cloud security is thus to identify security solutions that may have higher up-front costs but reduce per-tenant and per-machine variable costs—solutions that may have previously been unaffordable because their up-front costs could not be sufficiently amortized. For example, if tenants will rely on the cloud provider to investigate breaches, the cloud provider may invest in tools to reduce per-investigation costs. It may build forensic tools for inspecting VMs or logging tools that make it easier to record, search, and analyze log data. Solutions for remotely verifying the integrity of a machine's operating environment been frequently proposed but rarely implemented; the engineering costs for getting them right may yet be affordable at cloud scale.

6 Related Work

The idea of providing computing as a utility is far from new, as are security issues with shared computing infrastructure, but recent developments have catalyzed explosive interest and growth of what we now call 'cloud computing'. Karger and Schell discuss lessons learned from the security evaluation of Multics, which was one of the first systems to tackle the problems of secure shared computing [25]. Ambrust et al. discuss the reasons for the cloud computing's recent popularity growth and outline key features that make it different from prior shared computing systems, such as the ability to scale down to small pilot projects or up to large projects [5].

Many others have discussed threats arising from cloud computing. Talbot's article in MIT's *Technology Review* provides a high-level examination of cloud security issues, covering both cloud applications (e.g. Facebook and Gmail) and cloud-hosting [45]. Schneier observes many potential threats of cloud hosting and notes similarities between cloud hosting and traditional timesharing computing [44], while Balding and Hoff

each discuss problems with compliance in today's cloud hosting regimes [6, 21]. The Cloud Security Alliance enumerates technological threats to cloud providers and tenants [2]. Varia describes best practices such as frequent patching for virtual machines as part of a white paper on architecting for cloud computing [51].

Many of the threats we have enumerated have origins in real events. Amazon S3 suffered data corruption due to a flaky border gateway router [26]. The experience highlighted the difficulty today's cloud customers have in verifying the integrity of cloud infrastructure and isolating the source of failures. Underprovisioning is already a concern of some cloud tenants [52] and third-party monitors [11].

Amazon, Microsoft, and other cloud providers rely heavily on hypervisor-based virtual machines to isolate tenants, thus making their security a key area of concern. While virtual-machine level isolation provided by hypervisors is easier to reason about than most OS-level isolation, it is not immune to security flaws. The Cloudburst exploit found by Kortchinsky demonstrated how a specially crafted guest video driver could take control of a host machine running VMWare Workstation or ESX Server [30]. The flaw exploited by Cloudburst was a failure by VMWare to properly bounds check certain calls from the guest video card driver to VMWare emulated 3D hardware. Ormandy found that simple random fuzzing of common virtualization software, including QEMU and VMWare, uncovered potentially exploitable bugs [37]. Like the Cloudburst exploit, several of these bugs were also located in hardware emulation code. Garfinkel and Rosenblum discuss further issues with security in virtualized environments, such as the challenge of patching virtual machine images or the potential for re-use of randomness in cryptographic operations [16].

The drive towards features has pushed commodity virtual machine monitors to include more code, which increases the risk that a serious bug will appear. Recent academic work has pushed back against this trend by focusing on smaller, easier to verify hypervisors [27, 24]. For example, Flicker and Trustvisor reduce the size of their hypervisors by exploiting new CPU features designed to make writing hypervisors easier [31, 32].

The timing attacks that may impact tenant-shared CPUs in the cloud have their roots in cryptosystems. Kocher demonstrated timing attacks on smart cards [29] and later Boneh and Brumley showed that timing attacks could be carried out over the network [9]. Tromer et al. showed that cache effects could lead to timing attacks even on symmetric encryption schemes such as AES [47], which could potentially be used to attack a tenant sharing a CPU. Bortz and Boneh show how timing attacks can reveal information about web applications as well [7].

Ristenpart et al. demonstrate side channel attacks on the Amazon Elastic Compute Cloud and Xen hypervisor that allow them to determine whether their tenant VM is co-located with a VM belonging to a target web service and, if so, to learn keystroke timing information [42].

In the area of audit, the CloudAudit working group is currently drafting a specification for an API focused on "audit, assertion, assessment, and assurance" for cloud providers [18]. The goal of the API is to generate machine-readable assertions that detail which security features and certifications a provider does and does not have. Prospective tenants can then programatically decide whether to purchase resources from a provider for their application given their security needs.

Kelsey and Schnier introduce the concept of secure audit logs, a possible mechanism for implemented the audit countermeasures we discuss in Section 3. Iliev and Smith propose logs that utilize a security coprocessor, such as the IBM 4758, to achieve tamper evidence [23]. Their work followed on the Packet Vault project, which aimed at capturing and recording every packet over a 10 MBps link indefinitely on commodity disk storage [4].

For new security features that could be deployed to cloud tenants, Cui's work shows how to detect malware from scanning memory images, and more generally how to identify specific objects in a memory dump [10]. Cloud providers could use this functionality as part of a cloud infrastructure to audit tenant execution with modest overhead. Garfinkel et al. describe an architecture for embedding intrusion detection directly inside a hypervisor [15].

Gordon et al. model the optimal amount of information sharing between different entities [17]. Their analysis reveals a free rider problem that leads to systematic under investment in security when each firm is free to choose its level of sharing. A cloud provider can avoid this free riding problem by bundling a given level of information sharing with the cloud service.

7 Conclusion

Cloud hosting has desirable features including low up-front costs, elasticity of resources, and cost savings that result from economies of scale. Self hosting provides greater direct control over infrastructure than can be achieved when leasing shared infrastructure from the cloud. However, achieving the benefits of cloud infrastructure by transferring infrastructure control to a third party needn't necessarily result in a net loss of security—security may also benefit from scale economies.

In particular, cloud providers can afford security measures with up-front costs that would be unaffordable in self-hosting environments, amortizing these costs over myriad machines or tenants. A key research opportunity is to develop security measures that reduce marginal costs even if they incur greater up-front costs. With three new workshops on cloud security emerging in the past year[1, 39, 22], we hope to see new technical solutions that exploit the economics of deploying security in cloud-hosting infrastructures.

References

[1] ACM. CCSW 2010: the ACM cloud computing security workshop, 2010. `http://crypto.cs.stonybrook.edu/ccsw10`.

[2] Cloud Security Alliance. Security guidance for critical areas of focus in cloud computing v2.1, December 2009. http://cloudsecurityalliance.org.

[3] Amazon. Request to remove email sending limitations, February 2010. `http://aws.amazon.com/contact-us/ec2-email-limit-request/`.

[4] C.J. Antonelli, M. Undy, and P. Honeyman. The packet vault: Secure storage of network data. In *Proceedings USENIX Workshop on Intrusion Detection and Network Monitoring*, April 1999.

[5] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds: A berkeley view of cloud computing. Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley, Feb 2009.

[6] Craig Balding. What everyone ought to know about cloud security, 2009. http://www.slideshare.net/craigbalding/what-everyone-ought-to-know-about-cloud-security.

[7] Andrew Bortz and Dan Boneh. Exposing private information by timing web applications. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 621–628, New York, NY, USA, 2007. ACM.

[8] Carl Brooks. Amazon EC2 email blocked by antispam group spamhaus. `http://searchcloudcomputing.techtarget.com/news/article/0,289142,sid201%_gci1371369,00.html`.

[9] David Brumley and Dan Boneh. Remote timing attacks are practical. In *Proceedings of the 12th USENIX Security Symposium*, pages 1–14. Usenix, 2003.

[10] Martim Carbone, Weidong Cui, Wenke Lee, Marcus Peinado, and Xuxian Jiang. Mapping kernel objects to enable systematic integrity checking. In *ACM Conference on Computer and Communications Security*, 2009.

[11] Cloudkick. Visual evidence of amazon EC2 network issues, January 2010. `https://www.cloudkick.com/blog/2010/jan/12/visual-ec2-latency/`.

[12] AMD Corporation. AMD presidio extensions, 2010.

[13] Intel Corporation. Intel®trusted execution technology, 2009. http://www.intel.com/technology/security/.

[14] cPanel. cPanel, 2010. http://www.cpanel.net/.

[15] Tal Garfinkel and Mendel Rosenblum. A virtual machine introspection based architecture for intrusion detection. In *Proc. Network and Distributed Systems Security Symposium*, pages 191–206, 2003.

[16] Tal Garfinkel and Mendel Rosenblum. When virtual is harder than real: Security challenges in virtual machine based computing environments. In *In 10th Workshop on Hot Topics in Operating Systems*, 2005.

[17] L.A. Gordon, M.P. Loeb, and W. Lucyshyn. Sharing information on computer systems: An economic analysis. *Journal of Accounting and Public Policy*, 22(6):461–485, 2003.

[18] CloudAudit Working Group. A6 - the audit, assertion, assessment, and assurance API, 2010. http://www.cloudaudit.org.

[19] Shay Gueron. Intel®advanced encryption standard (aes) instructions set - rev. 3.0. http://software.intel.com/file/24917, January 26, 2010.

[20] Diwaker Gupta, Ludmila Cherkasova, and Amin Vadhat. Enforcing performance isolation across virtual machines in xen. In *Proceedings of the ACM/IFIP/USENIX Middleware Conference*, September 2007.

[21] Chris Hoff. Please help me: I need a QSA to assess PCI/DSS compliance in the cloud..., 2008. http://rationalsecurity.typepad.com/blog/2008/10/please-help-me-i-need-a-qsa-to-assess-pcidss-compliance-in-the-cloud.html.

[22] IEEE. International workshop on security in cloud computing, 20102. http://bingweb.binghamton.edu/~ychen/SCC2010.htm.

[23] A. Iliev and S. W. Smith. Protecting client privacy with trusted computing at the server. *IEEE Security and Privacy*, 3(2):153–186, 2005.

[24] Rhanjit Jhala, Michael Emmi, Eddie Kohler, and Rupak Majumdar. Verifying reference counting implementations. In *TACAS 2009: 15th Conference on Tools and Algorithms for the Construction and Analysis of Systems*, 2009.

[25] Paul A. Karger and Roger R. Schell. Thirty years later: Lessons from the multics security evaluation. In *in Annual Computer Security Applications Conference (ACSAC*, pages 119–126, 2002.

[26] Amazon Kathrin@AWS. Re: S3 data corruption?, June 2008. http://developer.amazonwebservices.com/connect/thread.jspa?threadID=22709.

[27] N. Kidd, T. Reps, J. Dolby, and M. Vaziri. Finding concurrency-related bugs using random isolation. In *In Proc. Verification, Model Checking, and Abstract Interpretation*, 2009.

[28] Samuel T. King, Joseph Tucek, Anthony Cozzie, Chris Grier, Weihang Jiang, and Yuanyuan Zhou. Designing and implementing malicious hardware. In *First USENIX Workshop on Large-Scale Exploits and Emergent Threats*, April 2008.

[29] Paul C. Kocher. Timing attacks on implementations of diffie-hellman, RSA, DSS, and other systems. In *CRYPTO*, 1998.

[30] Kostya Kortchinsky. CLOUDBURST: A VMware guest to host escape story. In *Black Hat USA*, 2009. http://www.blackhat.com/presentations/bh-usa-09/KORTCHINSKY/BHUSA09-Kortchinsky-Cloudburst-SLIDES.pdf.

[31] Jonathan M. McCune, Bryan Parno, Adrian Perrig, Michael K. Reiter, and Hiroshi Isozaki. Flicker: An execution infrastructure for TCB minimization. In *Proceedings of the ACM European Conference on Computer Systems (EuroSys)*, April 2008.

[32] Jonathan M. McCune, Ning Qu, Yanlin Li, Anupam Datta, Virgil D. Gligor, and Adrian Perrig. Efficient TCB reduction and attestation. In *IEEE Symposium on Security and Privacy*, 2010.

[33] Robert McMillan. Cloud computing a 'security nightmare,' says Cisco CEO. `http://www.pcworld.com/businesscenter/article/163681/cloud\_computing\_%a\_security\_nightmare\_says\_cisco\_ceo.html`, 2009.

[34] Robert McMillan. Man pleads guilty to selling fake chips to US navy, November 2009. `http://pcworld.about.com/od/securit1/Man-Pleads-Guilty-to-Selling-F.htm%`.

[35] Rich Miller. GoDaddy goes unlimited: resistance is futile, February 2009. `http://www.datacenterknowledge.com/archives/2009/02/23/go-daddy-goes-un%limited-resistance-is-futile/`.

[36] Oracle. Oracle keynote cloud expo 11-04-2009, November 2009. `http://www.slideshare.net/wrecks/oracle-keynote-cloud-expo-110409`.

[37] Tavis Ormandy. An empirical study into the security exposure to hosts of hostile virtualized environments, 2007. http://taviso.decsystem.org/virtsec.pdf.

[38] Bryan Parno, Jonathan M. McCune, and Adrian Perrig. Bootstrapping trust in commodity computers. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2010.

[39] Wolter Pieters. SPCC 2010 workshop on security and privacy in cloud computing, 2010. `http://www.spcc2010.info/`.

[40] Shaya Potter, Steven M. Bellovin, and Jason Nieh. Two person control administration: Preventing administration faults through duplication. In *LISA '09*, November 2009.

[41] Gartner Research. Gartner magic quadrant for managed security service providers 2009, 2009.

[42] Thomas Ristenpart, Eran Tromer, Hovav Shacham, and Stefan Savage. Hey, you, get off of my cloud! Exploring information leakage in third-party compute clouds. In Somesh Jha and Angelos Keromytis, editors, *Proceedings of CCS 2009*, pages 199–212. ACM Press, November 2009.

[43] Paul Roldan. FBI Criminal Investigation: Cisco Routers, January 2008. `http://www.networkworld.com/community/node/27858`.

[44] Bruce Schneier. Schneier on security: Cloud computing. http://www.schneier.com/blog/archives/2009/06/cloud_computing.html, June 4 2009.

[45] David Talbot. Security in the ether, February 2010. http://www.technologyreview.com/web/24166/.

[46] Christopher Tarnovsky. Deconstructing a 'secure' processor. In *Black Hat Briefings Federal*, February 2010. `http://www.blackhat.com/presentations/bh-dc-10/Tarnovsky_Chris/BlackHat%-DC-2010-Tarnovsky-DASP-slides.pdf`.

[47] Eran Tromer, Dag Arne Osvik, and Adi Shamir. Efficient cache attacks on AES and countermeasures. *Journal of Cryptology*, 23(1):37–71, 2009.

[48] UBS. Formal signing of settlement agreement relating to the John Doe summons, August 2009. http://www.ubs.com/1/e/investors/releases?newsId=170330.

[49] Office of the Inspector General United States Department of Justice. A review of the federal bureau of investigation's use of national security letters, 2009. http://www.justice.gov/oig/special/s0703b/final.pdf.

[50] Amin Vadhat. The achilles' heel of performance isolation in the cloud, 2010. http://idleprocess.wordpress.com/2010/01/17/the-achilles-heel-of-perfor%mance-isolation-in-the-cloud/.

[51] Jinesh Varia. Architecting for the cloud: Best practices, 2010. http://jineshvaria.s3.amazonaws.com/public/cloudbestpractices-jvaria.pd%f.

[52] Alan Williamson. Has amazon EC2 become over subscribed?, 2009. http://alan.blog-city.com/has_amazon_ec2_become_over_subscribed.htm.

[53] Zend. Simple cloud API, 2010. http://www.simplecloud.org.

[54] Kim Zetter. FBI defends disruptive raids on texas data centers, April 2009. http://www.wired.com/threatlevel/2009/04/data-centers-ra/.