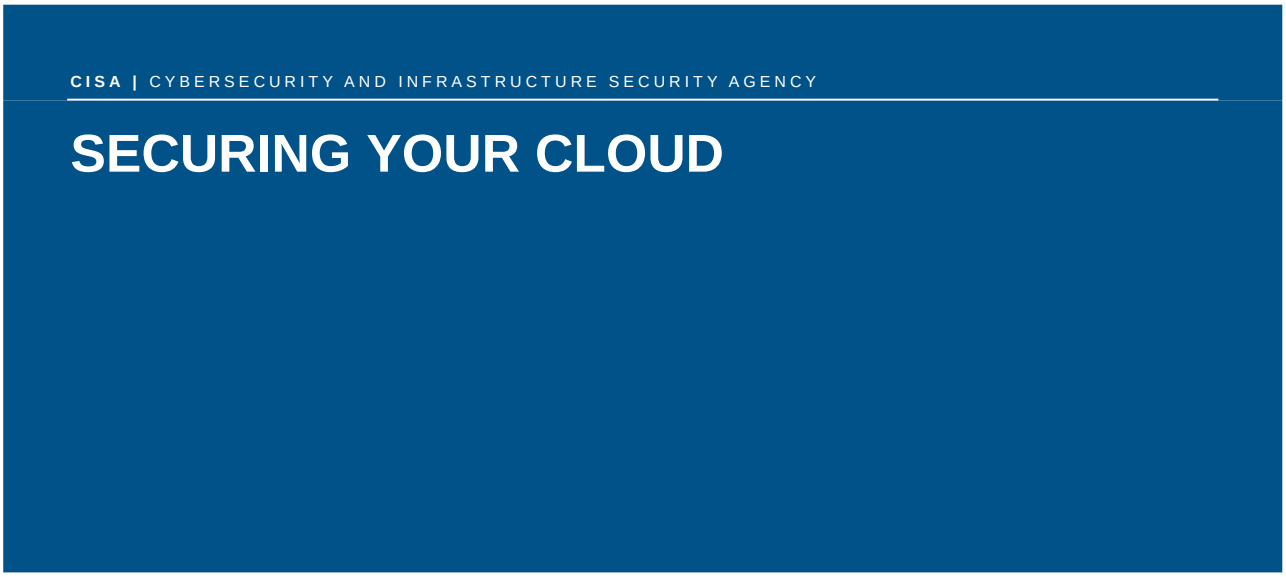
Securing Your Cloud

Table of Contents

Notices

**001 Instructor: Mastering identity and



**002 access management is probably one of the most

Identity and Access Management (IAM)

- Features of Cloud IAM
  - Unified view across the organization
  - Role-Based Access Control (RBAC)
  - Pre-defined policies
  - Integration with enterprise authentication (e.g., Windows Active Directory)
  - Multi-Factor Authentication
  - Built-in audit trails
- Best Practices
  - Don’t use Root account after initial setup
  - Don’t share credentials
  - Enable MFA
  - Use groups and RBAC
  - Least privilege
  - Manage service accounts
  - No credentials in code
  - Monitor activity and periodically audit permissions and policies

3

**003 important things your organization can do to protect your cloud computing environment, your data, and ultimately your business.

IAM is a core capability for each cloud service provider, but the depth and breadth of features varies between CSPs and their tiered offerings. Ultimately your goal is to have one unified view across your entire organization. You need to be fully aware of who has access to your resources, what permissions they have, and what actions have been taken.

In general, users are a physical person. Groups should be established for functions like admin or DevOps, or teams like engineering or design. Those groups then

contain lists of users that require specific permissions to resources based on their role within the organization. Azure recommends enabling single sign-on with Active Directory, which is not surprising as it is a mature Microsoft technology that is their core to traditional IAM.

Verizon's 2020 Data Breach Investigations Report observes that over 80 percent of breaches involve brute force or use of lost or stolen credentials. Enforcing multifactor authentication, or MFA, greatly mitigates this risk and it is recommended guidance for just about everything, not just cloud computing or cloud IAM.

Let's look at some best practices. First, don't use the root account. Once your subscription is set up, the master account needs to have multifactor authentication enabled right away. The next step is to create additional admin or owner accounts. These too should be limited and enforce MFA, and they should be used for daily admin duties instead of the root account. Those permissions, as well as the ones assigned to other roles, need to be carefully planned ahead of time. Your organization may have a group of application developers, IT folks, security analysts, DevOps engineers, and many others. These could even be spread across dozens of subunits within your organization.

If you have bad practices for managing IAM within your traditional

computing environment, here's an opportunity for you to clean those up. You absolutely do not want to transition bad habits to the cloud, or you could easily increase exposure to new threats if you implement IAM poorly. Do not use shared accounts and credentials. This aligns with traditional best practices, as you need to ensure accountability for actions taken within your cloud environment. And when possible, always enforce some kind of multifactor authentication.

I mentioned roles a moment ago. There are a number of built-in roles within each public cloud offering. It is important to research those and review documentation and know exactly what permissions are granted with each. In general, users should be assigned to groups and permissions defined for each of those roles, and along those lines, users should not have any permissions or rights available to them beyond what is necessary to do their job.

Even if trust is presumably high, this greatly minimizes the risk of accidental damage, information exposure, or spread of an attack if the user account is somehow compromised. When enabling certain cloud services, such as installing host-based agents or even configuring APIs, there's potential for a new service account to be created. It is critical that you understand and control the use of these accounts--what are they, when do they get installed, what password policies are

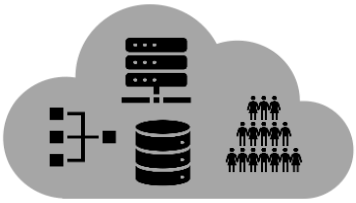
applied, what privileges do they have within your system, and, given the shared responsibility model, you need to be certain who is responsible for the security of these service accounts.

Cloud best practices further reflect general computing best practices when we say do not include credentials with code, especially when that code is stored in publicly accessible code repositories. This has become a popular attack vector, so proper use of variables and configuration files is paramount. Hard-coding credentials even for testing purposes can lead to unintended consequences.

And one of the biggest benefits to cloud computing is the built-in logging and monitoring capabilities for each platform. Some of these are on by default while others may need to be enabled, and in doing so you risk information overload. Take the time to learn and understand these features and tailor alerts to your environment. This could help identify unauthorized configuration changes, unintended expenses, and it could also be critical during an incident investigation.

Common Cloud Resources

- IaaS Basics
  - Virtual Machines
  - Networking
  - Shared Storage
  - Security Groups
- Top Risks
  - Misconfiguration
  - Brute force password guessing
- Mitigation
  - Time-based authentication
  - Account policies
  - Access Control Lists (ACLs)
  - Multi-factor authentication (MFA)
  - Monitoring/Alerting
    - Access attempts
    - Provisioning
  - Profile management



4

**004 Infrastructure as a service generally includes some basic components. A virtual machine, including its operating system and application, is fundamental. That server will need access to a network and presumably one or more services will need to be reachable from the internet. Having a virtual hard drive that can be attached to that server is another common asset, and depending on the application, that storage solution may end up exposing files to the internet as well, whether that is intentional or not. A security group or network security group is often used to identify the access control rules for resources and resource groups within a cloud service provider.

The top risks are some of the oldest and most common risks faced since the inception of the internet: misconfigured servers that unintentionally expose data, use weak or clear text protocols, or that expose unpatched services still exist in the cloud. That's why it is absolutely critical to have proper cloud platform training, strong identity and access management policies, change management processes, and enable activity monitoring.

Brute force password guessing is also very common. This often occurs on exposed administrative ports like TCP 3389 for a Microsoft remote desktop, or TCP 22 for SSH. Essentially bad guys automate thousands of login attempts trying to guess a username and password combination that will lead to system access.

Top mitigations include the use of time-based authentication. This feature allows a cloud resource administration to configure access control lists that would open up those SSH or RDP administrative ports for a limited time, like 30 minutes or an hour. The cloud-based firewall rules would automatically change or deny that access when the time limit was reached. Similar to non-cloud servers, enabling account policies that automatically lock accounts after a certain number of failed login attempts continues to be a recommendation for cloud servers. Similarly, multifactor authentication, which requires not only a username

and password but some other thing, such as a time-based token, digital certificate, or separate authentication device, is recommended, especially for any kind of privileged access.

Cloud accounts and profiles need to be configured with the minimum set of permissions required per job role. This mitigates the misconfiguration risk somewhat as users presumably may not be able to access or configure resources outside of their job scope.

Another way to mitigate these risks is to limit the devices that can configure, manage and/or monitor your cloud environment and resources. Some organizations choose to have dedicated laptops with very limited applications and tools installed that allow privileged users to do only the tasks they have been assigned. Other laptops are used for additional job functions and duties, but anything to do with their virtual private cloud requires a cloud-trusted desktop.

And like IAM, one of the biggest benefits to cloud computing is the built-in logging and monitoring capabilities for each platform. Some of these are on by default while others may need to be enabled. Proper configuration could provide early warnings of nefarious activities or help identify misconfigurations before they can be taken advantage of by threat actors.

Data Risks and Security

Unintended public release of information could lead to regulatory fines, lawsuits, lost revenue, investigation and response expenses, or even criminal charges.

- High-risk Information
  - Personal
  - Health
  - Payment/financial
  - Trade secrets
  - Passwords and keys
- Top Technical Risks
  - Application and DB vulnerabilities
  - Unsecured storage
  - Misuse/abuse of resources
- Mitigation
  - Data classification and management policies
  - Encryption
  - Authentication controls
  - Monitoring/alerting

5

**005 A data breach generally information that was not intended for public release. Exposure of this could lead to fines, lawsuits, lost reputation or revenue, and even additional expenses to support the investigation, pay for lawyers, or cover the cost of reissuing payment cards or paying for credit monitoring services for those involved.

High-risk information includes personal health information, PHI, financial data, trade secrets, intellectual property, and personally identifiable information, PII, which may include things like birthdays and social security numbers. The exposure of this sensitive data could lead to large fines, civil lawsuits, and even criminal charges.

The last one on the list, passwords and keys, is extremely important. Compromise of a password or perhaps an SSH or application programming interface, API, key could then lead to compromise of more critical information or disruption of production services.

The top technical risks in a cloud computing environment are listed here. If software developers and database administrators were perfect and technology was static, then perhaps this one would eventually go away, but for now, whether it is cloud or traditional computing, secure software development is required. The cloud provides easy access to storage resources that can be used to host application data, media assets, log files, or just about any other piece of information, and depending on exactly what those files are, controls should be put in place to encrypt that data or restrict access on a need-to-know basis. Misconfigured access control lists could provide attackers easy access to storage, and it is misconfiguration of storage application services, servers, and users that often leads to unintended compromise.

For mitigation, we must return to the basics. Start with knowing what data you have, why you have it, how it is collected, where it is stored, and based on that type of data, know what protections are required. This comes back to fundamentally classifying your data, which not only helps when identifying needed

controls, but helps guide the incident handling process if something bad does happen. Encryption of storage buckets may not be required if the files you are sharing are meant to be freely available to the world. But if they are not, then access controls must be put in place and encryption may also be required depending on the sensitivity that information.

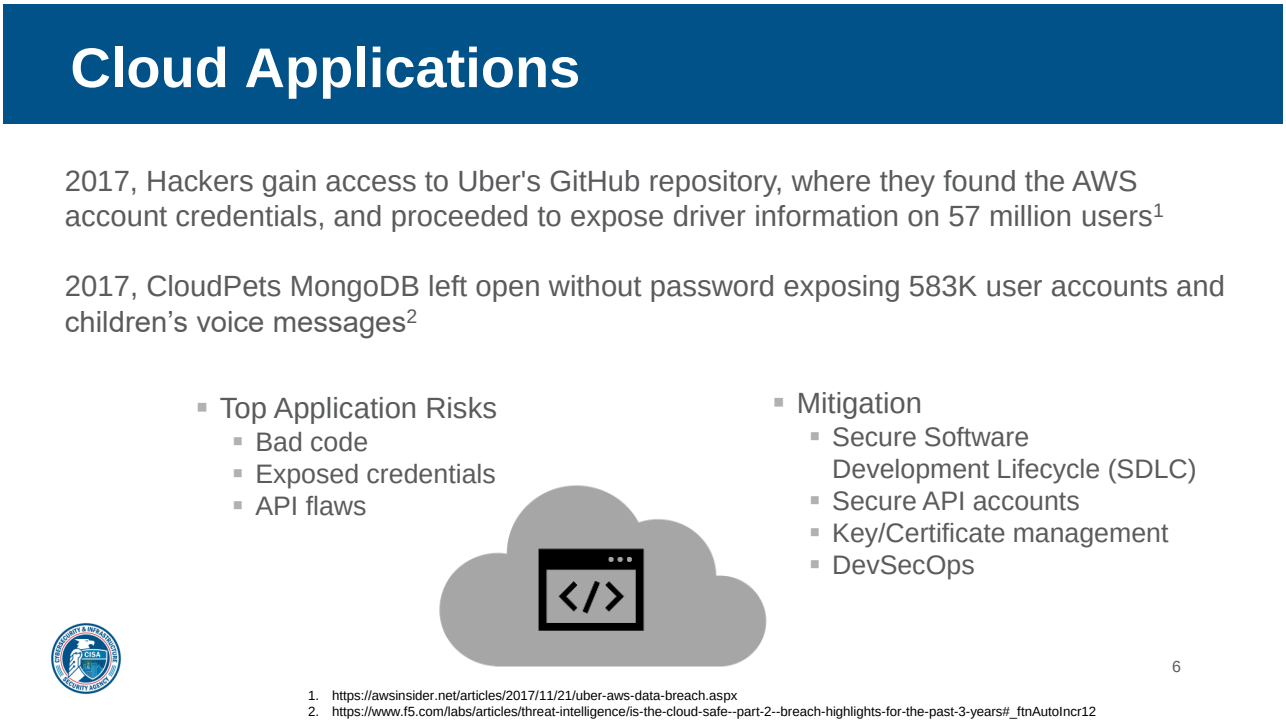
There are even freely available websites that search and index and secure. Amazon Simple Storage Service has three buckets, making it easy for attackers to find that data, so be aware that security through obscurity simply doesn't exist anymore.

Authentication can be tied to those access controls, but it also pertains to the cloud account protection. When possible, multifactor authentication should be enabled. This mitigates brute force password guessing attacks and further validates authorized users.

And of course, cloud logging, monitoring, and alerting services should be finely tuned. Perhaps you don't need to know every time a public file is accessed, but you should certainly evaluate traffic patterns and trends to see if they correlate to a business event or a new configuration. Perhaps you don't need to know ever login attempt, but instead you want to know when a successful login happens from a country that you do not have any cloud admins residing in.

Those are the types of things you need to take a step back and evaluate. What do you need to know, how often, and what actions can be taken? Document that, and then look to implement those controls as per your own policies and guidelines.

Cloud Applications



**006 Like basic web requests, application programming interface calls incorporate URLs, methods, headers, and other application parameters. These are often exposed to developers and to partner applications with some form of authentication in place to validate the request. That request may be as simple as an unauthenticated status query, or it may be a protected function to update a database. In the latter's case, the use of service

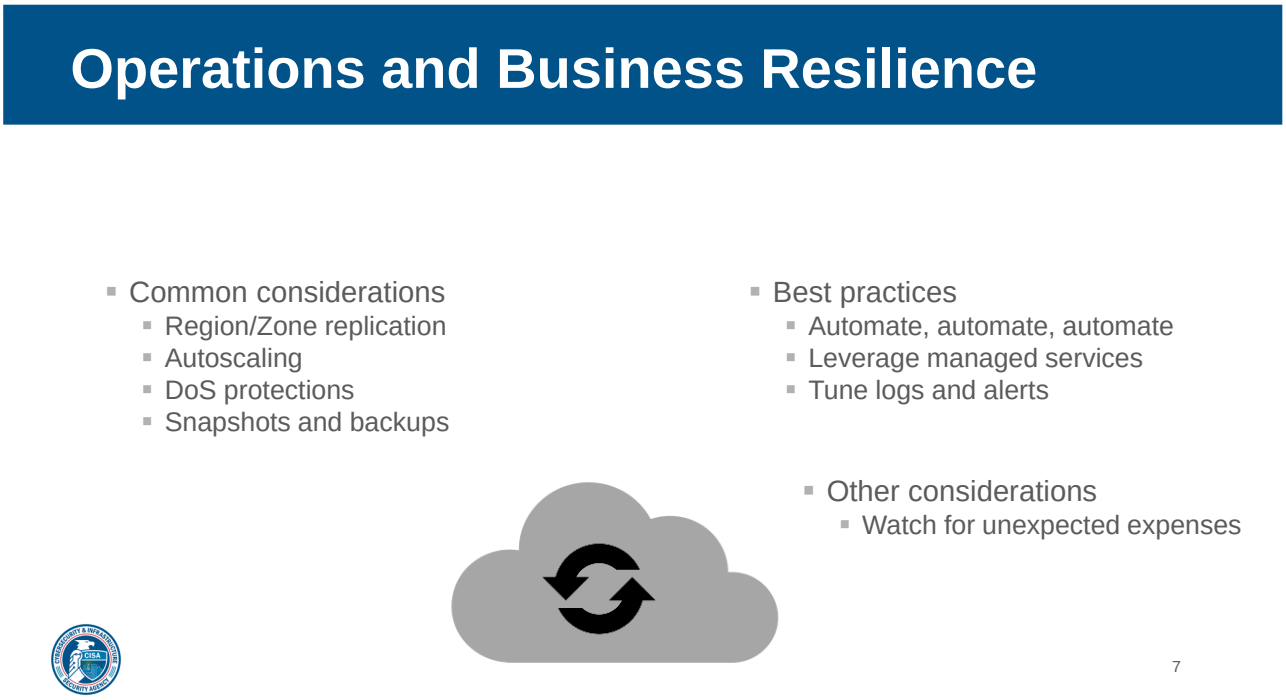
permissions used to perform that update must be carefully crafted not to allow access beyond the intended function. If so, this could lead to unauthorized changes or unintended access to information. There are numerous compromise examples, but these two will suffice.

The first is Uber's compromise in 2017, when attackers gained access to their private GitHub repo. Within the files there, a developer had exposed Amazon Web Server's credentials, which the attackers then leveraged to log into AWS and exfiltrate additional data. Also in 2017, CloudPets had a compromise, but this one came from a very common reason. Simply failing to configure their database correctly, attackers were able to access their entire system without even using a password. The Microsoft SQL Slammer worm took advantage of a similar flaw in 2003 to propagate, and Microsoft now enforces the setting of the system administrator, or SA, account during the installation.

But bringing this back to the cloud context, it is important to understand what services and API endpoints are being exposed to the internet. Is this intentional, and if so, how are these services and accounts protected? Are some of them automatically created by the cloud provider? Many of the advanced services offered by cloud service providers will automatically create or enable API accounts if you choose to take advantage of those add-ons.

So how do you protect yourself? First and foremost, master your own software development looked for. Have developers take secure coding training, implement quality control and testing mechanisms, and leverage proven API frameworks like REST. Don't check API keys and passwords into code repositories, and configure those accounts on systems with minimum amount of access to accomplish the task at hand; and investigate exactly what happens when you enable a service within your cloud environment. Does this create a new exposure for your organization?

DevSecOps bridges the gap between high-velocity software development, DevOps, and traditional security, which often doesn't have the same visibility into development and deployment architectures as it does with traditional IT. DevSecOps introduces cybersecurity into the culture and incorporates checks at all stages of the software development lifecycle.



**007 Cloud providers and platforms have many built-in tools to improve system resilience. Some basic ones include the use of regions and zones. These can be used to replicate data and enable load balancing to minimize outages for region-wide natural disasters. It is important to consider where your resources reside, because there may be regulatory restrictions based on your industry, customers, or compliance requirements.

Auto-scaling is a core tent of cloud computing, but how it is enabled depends greatly on your application and budget. Distributed denial of service, DDOS, protections are built in to some degree, but additional options are available with most cloud service providers. For example, AWS

Shield Standard provides DDOS protection for the most common network and transport layer attacks but target websites and applications at no additional charge. But Amazon also has an advanced protection that protects DNS, load balancing, Elastic Compute Cloud, EC2, and other services.

You also want to leverage the built-in capabilities to back up virtual machines and their deployment configurations, but you do need to be aware of what zones that backups are in. Does your data traverse international boundaries? Is it encrypted by the service provider when doing so? Are there different compliance and protection requirements for these different locations? These are just a few considerations.

One more important one is money. When you enable services like backups, how does that change your bill? You are increasing storage and potential ingress and egress bandwidth for data transfers, and in general, you need to have someone keeping an eye on those monthly invoices. Do you know how many resources you have deployed? Have there been any new instances, and why? What is the utilization of production applications and how will those trends impact expenses? Was there a forgotten resource that you are still getting charged for? Or, worse yet, one that has been compromised and is now being used

to mine Bitcoin, like what happened at Tesla in 2019.

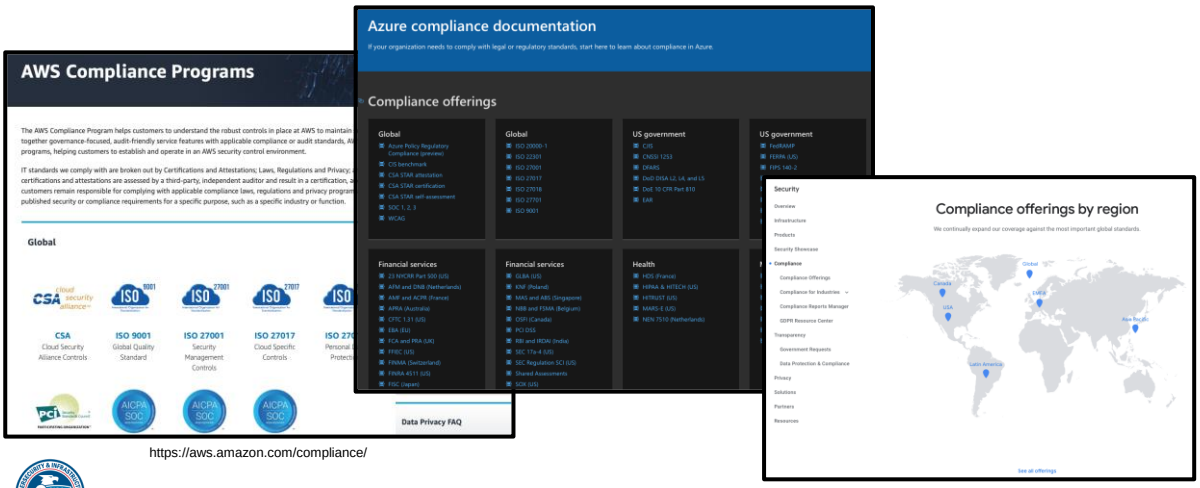
How about some best practices, starting by integrating security testing into image creation, including patch management and regression testing for key applications--regression testing meaning that you want to ensure all functions of the application are working as expected after any updates are made, even if those updates were not targeting that specific capability. If there is a way to update the gold image offline and then, when the resource is automatically deployed, this new, approved and verified image is automatically used instead of the old one.

Similarly, is there a way to minimize the life of a running server or application? There certainly are ways to do this, and it could also be an effective way of dealing with an advanced persistent threat. If the image does not persist beyond a couple of hours, in theory neither would the adversary's foothold. Of course this depends on your ability to leverage orchestration technologies like Terraform, Ansible, Chef, or Puppet to automate deployments and tear-down in conjunction with cloud APIs to query resources, status, and other events. There are, indeed, a number of extras that cloud providers have.

I mentioned AWS Shield, but there are things like Microsoft's Azure Security Center, which also requires

an upgraded subscription. The Security Center Standard service helps you find and fix security vulnerabilities, apply access and application controls to block malicious activity, detect threats using analytics and intelligence, and respond quickly when under attack. Some of these services in Azure or other cloud providers may require a host agent to be installed in order to get them to work effectively. This is particularly true when working with vulnerability assessment solutions. Other services, like Google's Web Security Scan, are pretty easy to use and can help meet compliance requirements. It doesn't install an agent, but it will automatically enable an API on a deployed app engine to make it work.

Compliance Resources



https://docs.microsoft.com/en-us/azure/compliance/

https://aws.amazon.com/compliance/

https://cloud.google.com/security/compliance

8

**008 This slide is just a reminder that Amazon, Microsoft and Google all have in-depth compliance resources that list their certifications, programs, and provide resources for you like checklists and guidance based on countries, industries, and more.

**009 One of the most important justifications for a cloud monitoring program is simply that it's good business. You are dealing with monthly expenses and you want those payments to be in accordance with your business needs. Spend the money necessary to enable services, meet compliance requirements, and ensure your business can recover from an outage. But be careful, because implementation of these services will cost money. Misconfigurations could have you paying way more than you actually need to. Conversely, insufficient scaling could degrade performance and lead to lost revenue.

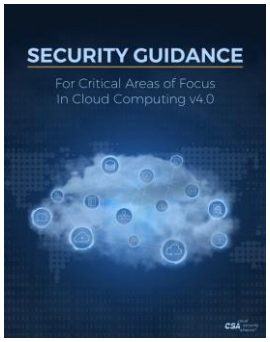
So what to monitor? System performance is just as important in the cloud as traditional IT. High CPU

and RAM utilization may indicate that you need a large virtual machine instance, or that you need to investigate scaling options. If this performance is not directly related to an application being served, perhaps it is due to a compromised host performing unauthorized actions, like crypto-mining.

Monitoring network activity is always valuable. Cloud service providers can provide aggregated network flow data for incoming and outgoing bytes, IP addresses, ports, protocols, and other metadata from traffic traversing your virtual private cloud. Full packet capture is also possible on major cloud providers, usually with a network mirroring service. However, the configuration options and steps are very different between them. But if you need to perform deep packet inspection, there are indeed options available, and it is just as important to monitor key resource actions like the creation of new instances, users, applications, etcetera. Unauthorized actions may be a result of compromise, poor access management, an insider threat, or simply incompetence.

The Cloud Security Alliance (CSA)

- Cloud Security Guidance
- Security, Trust, and Assurance Registry (STAR) Registry
- Cloud Controls Matrix (CCM)
- Consensus Assessment Initiative Questionnaire (CAIQ)
  - *Do you use an automated source code analysis tool to detect security defects in code prior to production?"*
  - *Do you encrypt tenant data at rest (on disk/storage) within your environment?*



https://cloudsecurityalliance.org

10

**010 The Cloud Security Alliance is a not-for-profit organization that promotes cloud computing best practices and also provides security guidance and educational resources. They have a certificate in cloud security knowledge and there is a freely available companion security guidance PDF that can be downloaded from their website.

The CSA also promotes the Security, Trust and Assurance Registry, or STAR, which documents the security and privacy controls provided by popular cloud computing offerings. This publicly accessible resource allows customers to assess their security providers in order to make the best procurement decisions. It goes hand in hand with the Cloud Controls Matrix, which is a

cybersecurity control framework for cloud computing composed of 133 controls structured in 16 domains, covering all key aspects of cloud technology. The controls framework is aligned to that security guidance document and maps to standards, regulations and frameworks like ISO 27001, NIST SP 800-53, PCI-DSS, which provides requirements for securing payment card information, and many others.

The Consensus Assessment Initiative Questionnaire takes a deeper dive into the controls matrix and provides specific questions you should consider when evaluating your security posture. For example, application and interface security includes: Do you use an automated source code analysis tool to detect security defects in code prior to production? Within the encryption and key management controls, do you encrypt tenant data at rest within your environment?

So some of these questions may be more appropriate for the cloud service provider and some may be for you to take action on. This stresses the importance of understanding the Cloud Shared Responsibility Model

**011 and how it applies to your specific architecture and services.