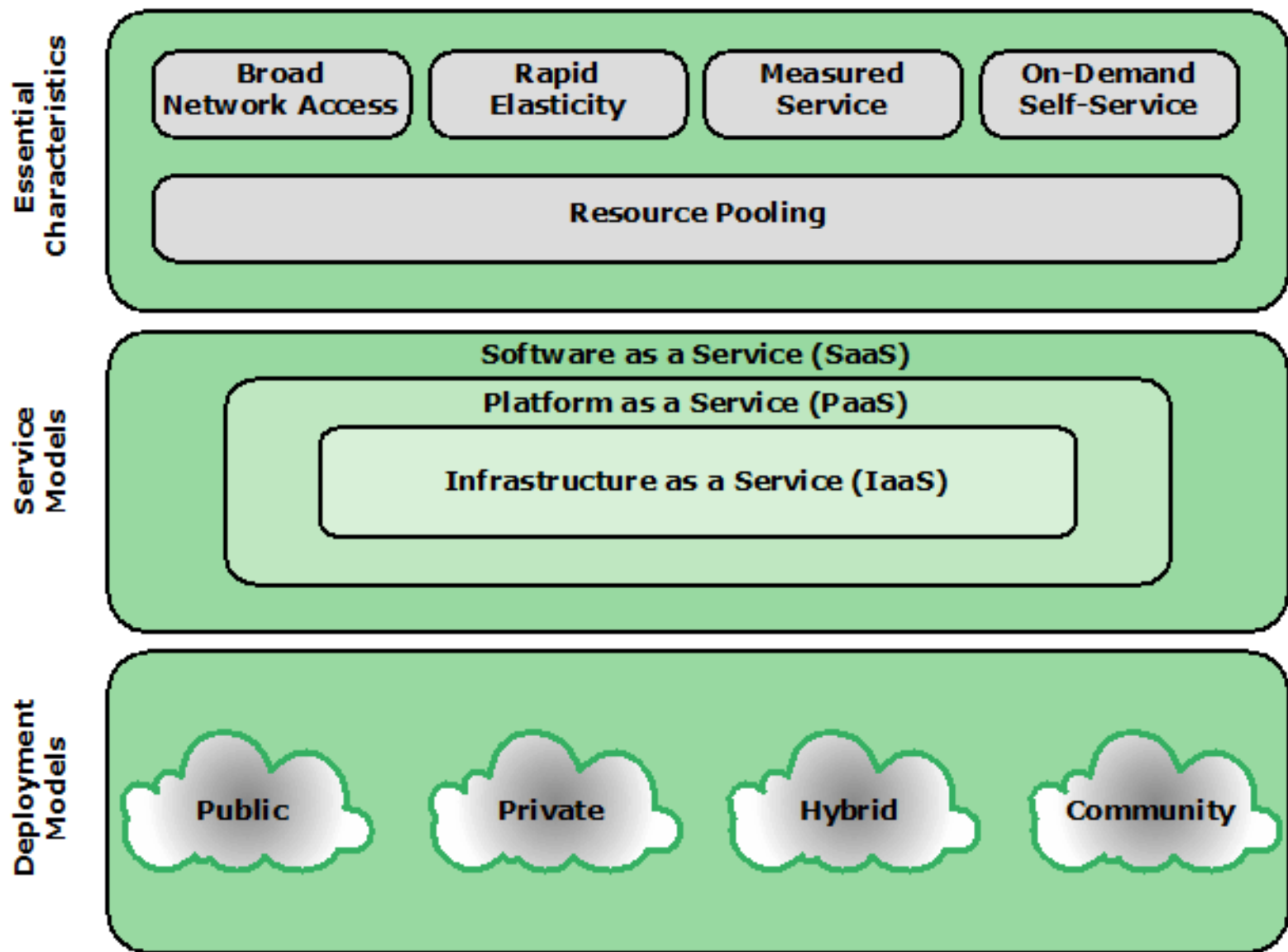


# Cloud and Internet of Things(IoT) Security

# Cloud Computing:

- NIST defines cloud computing, in NIST SP-800-145 (*The NIST Definition of Cloud Computing*, September 2011) as follows:

**“Cloud computing:** A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models.”



**Figure 13.1 Cloud Computing Elements**

# Essential characteristics

The essential characteristics of cloud computing include the following:

- **Broad network access:** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and tablets) as well as other traditional or cloud-based software services.
- **Rapid elasticity:** Cloud computing gives you the ability to expand and reduce resources according to your specific service requirement. For example, you may need a large number of server resources for the duration of a specific task. You can then release these resources upon completion of the task.
- **Measured service:** Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

# Essential characteristics

- **On-demand self-service:** A cloud service consumer (CSC) can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider. Because the service is on demand, the resources are not permanent parts of the consumer's IT infrastructure.
- **Resource pooling:** The provider's computing resources are pooled to serve multiple CSCs using a multitenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a degree of location independence, in that the CSC generally has no control or knowledge of the exact location of the provided resources, but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines (VMs). Even private clouds tend to pool resources between different parts of the same organization.

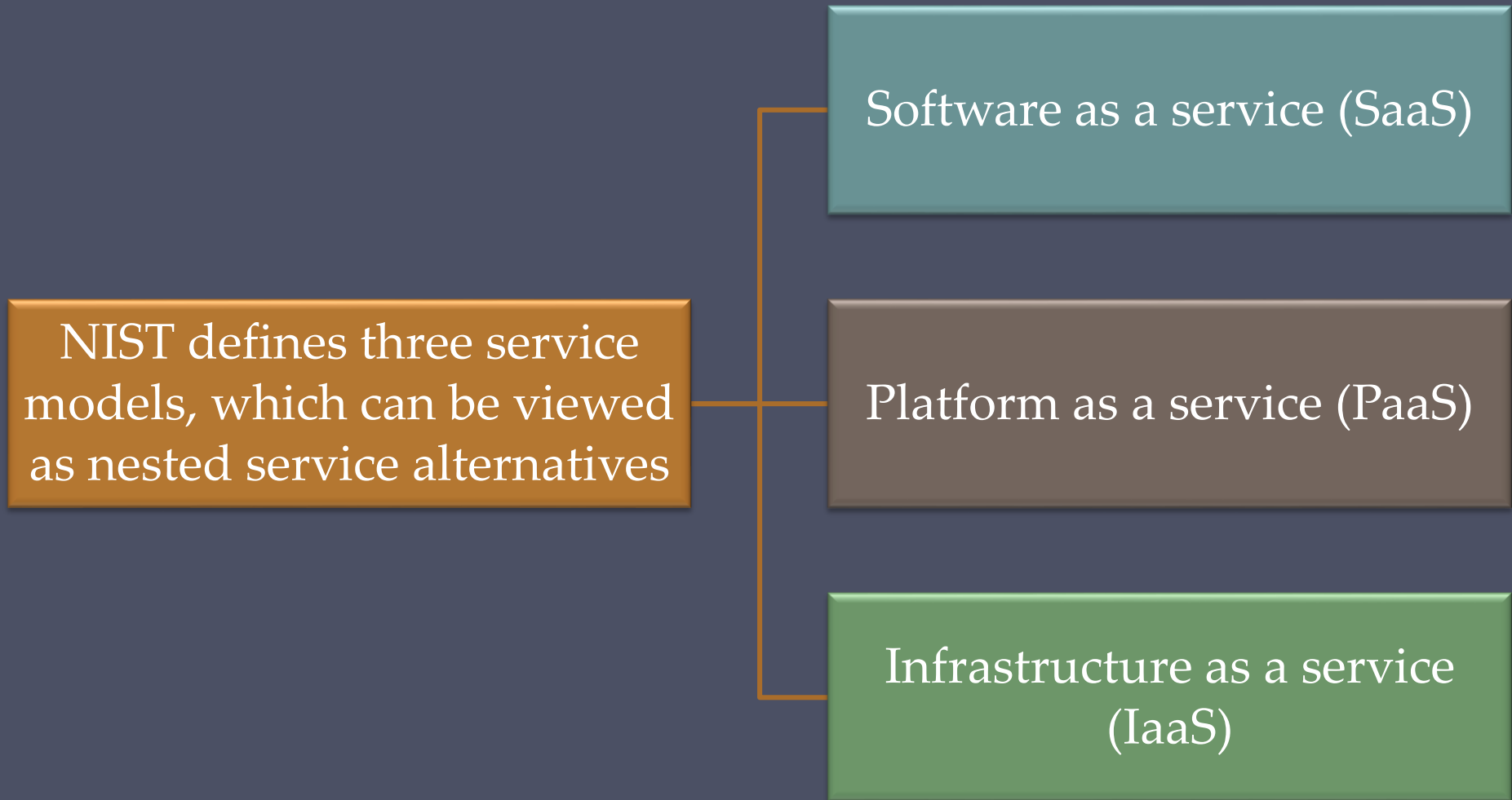
# Cloud Service Models

NIST defines three service models, which can be viewed as nested service alternatives

Software as a service (SaaS)

Platform as a service (PaaS)

Infrastructure as a service (IaaS)



# Software as a Service(SaaS)

SaaS provides service to customers in the form of software, specifically application software, running on and accessible in the cloud



It enables the customer to use the cloud provider's applications running on the provider's cloud infrastructure

- The applications are accessible from various client devices through a simple interface, such as a Web browser
- Instead of obtaining desktop and server licenses for software products it uses, an enterprise obtains the same functions from the cloud service



The use of SaaS avoids the complexity of software installation, maintenance, upgrades, and patches



Examples of this service are Google Gmail, Microsoft 365, Salesforce, Citrix GoToMeeting, and Cisco WebEx

# Software as a Service(SaaS)

SaaS provides service to customers in the form of software, specifically application software, running on and accessible in the cloud. SaaS follows the familiar model of Web services, in this case applied to cloud resources. SaaS enables the customer to use the cloud provider's applications running on the provider's cloud infrastructure. The applications are accessible from various client devices through a simple interface, such as a Web browser. Instead of obtaining desktop and server licenses for software products it uses, an enterprise obtains the same functions from the cloud service. The use of SaaS avoids the complexity of software installation, maintenance, upgrades, and patches. Examples of services at this level are Google Gmail, Microsoft 365, Salesforce, Citrix GoToMeeting, and Cisco WebEx.

Common subscribers to SaaS are organizations that want to provide their employees with access to typical office productivity software, such as document management and email. Individuals also commonly use the SaaS model to acquire cloud resources. Typically, subscribers use specific applications on demand. The cloud provider also usually offers data-related features, such as automatic backup and data sharing between subscribers.



# Platform as a Service(PaaS)

A PaaS cloud provides service to customers in the form of a platform on which the customer's applications can run

PaaS enables the customer to deploy onto the cloud infrastructure customer-created or acquired applications

A PaaS cloud provides useful software building blocks, plus a number of development tools, such as programming language tools, run-time environments, and other tools that assist in deploying new applications

In effect, PaaS is an operating system in the cloud

It is useful for an organization that wants to develop new or tailored applications while paying for the needed computing resources only as needed, and only for as long as needed

Examples of PaaS include AppEngine, Engine Yard, Heroku, Microsoft Azure, Force.com, and Apache Stratos

# Platform as a Service(PaaS)

- A PaaS cloud provides service to customers in the form of a platform on which the customer's applications can run. PaaS enables the customer to deploy onto the cloud infrastructure customer-created or acquired applications.
- A PaaS cloud provides useful software building blocks, plus a number of development tools, such as programming language tools, run-time environments, and other tools that assist in deploying new applications.
- In effect, PaaS is an operating system in the cloud. PaaS is useful for an organization that wants to develop new or tailored applications while paying for the needed computing resources only as needed, and only for as long as needed.
- AppEngine, Engine Yard, Heroku, Microsoft Azure, Force.com, and Apache Stratos are examples of PaaS.

# Infrastructure as a Service(IaaS)

With IaaS, the customer has access to the resources of the underlying cloud infrastructure

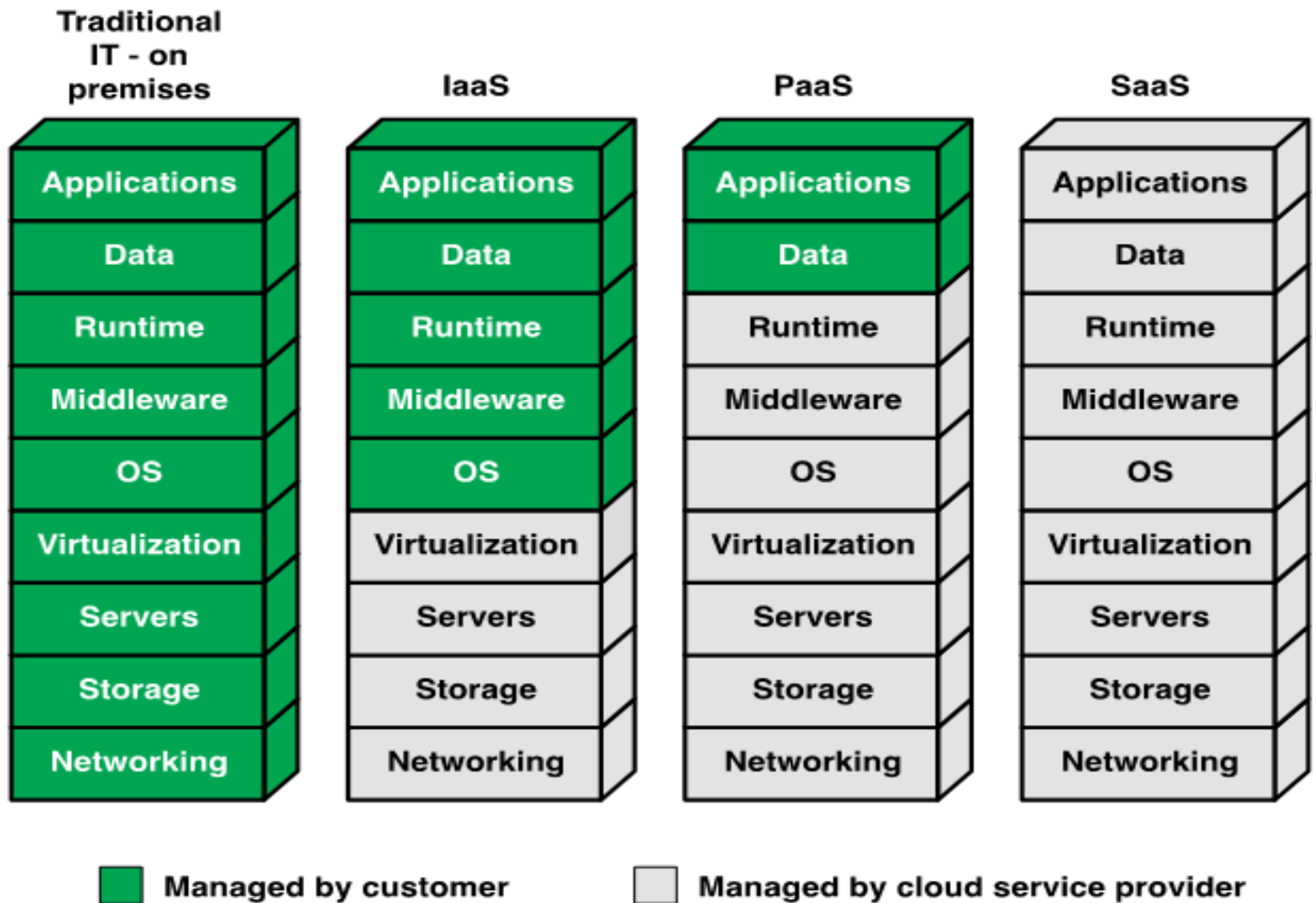
The cloud service user does not manage or control the resources of the underlying cloud infrastructure, but has control over operating systems, deployed applications, and possibly limited control of select networking components

IaaS provides virtual machines and other virtualized hardware and operating systems

IaaS offers the customer processing, storage, networks, and other fundamental computing resources so the customer is able to deploy and run arbitrary software, which can include operating systems and applications

IaaS enables customers to combine basic computing services, such as number crunching and data storage, to build highly adaptable computer systems

Examples of IaaS are Amazon Elastic Compute Cloud, Microsoft Windows Azure, Google Compute Engine, and Rackspace



**Figure 13.2 Separation of Responsibilities in Cloud Service Models**

# Infrastructure as a Service(IaaS)

- With IaaS, the customer has access to the resources of the underlying cloud infrastructure. The cloud service user does not manage or control the resources of the underlying cloud infrastructure, but has control over operating systems, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).
- IaaS provides virtual machines and other virtualized hardware and operating systems. IaaS offers the customer processing, storage, networks, and other fundamental computing resources so the customer is able to deploy and run arbitrary software, which can include operating systems and applications. IaaS enables customers to combine basic computing services, such as number crunching and data storage, to build highly adaptable computer systems.
- Typically, customers are able to self-provision this infrastructure, using a Web based graphical user interface that serves as an IT operations management console for the overall environment. API access to the infrastructure may also be offered as an option.
- Examples of IaaS are Amazon Elastic Compute Cloud (Amazon EC2), Microsoft Windows Azure, Google Compute Engine (GCE), and Rackspace.

# Cloud Deployment Models

Public cloud

Community cloud

The four most prominent deployment models for cloud computing are:


Private cloud

Hybrid cloud

# Public Cloud

- A public cloud infrastructure is made available to the general public or a large industry group, and is owned by an organization selling cloud services
  - The cloud provider is responsible both for the cloud infrastructure and for the control of data and operations within the cloud
- A public cloud may be owned, managed, and operated by a business, academic, or government organization, or some combination of them
  - All major components are outside the enterprise firewall, located in a multitenant infrastructure
  - Applications and storage are made available over the Internet via secured IP, and can be free or offered at a pay-per-usage fee
- The major advantage of the public cloud is cost
- The principal concern is security

# Private Cloud



A private cloud is implemented within the internal IT environment of the organization

The organization may choose to manage the cloud in house or contract the management function to a third party

The cloud servers and storage devices may exist on premise or off premise

Private clouds can deliver IaaS internally to employees or business units through an intranet or the Internet via a virtual private network (VPN), as well as software or storage as services to its branch offices

Examples of services delivered through the private cloud include database on demand, email on demand, and storage on demand

A key motivation for opting for a private cloud is security

Other benefits include easy resource sharing and rapid deployment to organizational entities



# Community Cloud



A community cloud shares characteristics of private and public clouds

- Has restricted access like a private cloud
- The cloud resources are shared among a number of independent organizations like a public cloud

The organizations that share the community cloud have similar requirements and, typically, a need to exchange data with each other

- An example would be the health care industry

The cloud infrastructure may be managed by the participating organizations or a third party, and may exist on premise or off premise

- In this deployment model, the costs are spread over fewer users than a public cloud so only some of the cost savings potential of cloud computing are realized

# Hybrid Cloud

- The hybrid cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability
- With a hybrid cloud solution, sensitive information can be placed in a private area of the cloud, and less sensitive data can take advantage of the benefits of the public cloud
- A hybrid public/private cloud solution can be particularly attractive for smaller business
- Many applications for which security concerns are less can be offloaded at considerable cost savings without committing the organization to moving more sensitive data and applications to the public cloud

	<b>Private</b>	<b>Community</b>	<b>Public</b>	<b>Hybrid</b>
<b>Scalability</b>	Limited	Limited	Very high	Very high
<b>Security</b>	Most secure option	Very secure	Moderately secure	Very secure
<b>Performance</b>	Very good	Very good	Low to medium	Good
<b>Reliability</b>	Very high	Very high	Medium	Medium to high
<b>Cost</b>	High	Medium	Low	Medium

**Table 13.1 Comparison of Cloud Deployment Models**

# Cloud Computing:

- NIST SP-500-292 (*NIST Cloud Computing Reference Architecture*) establishes reference architecture, described as follows:

“The NIST cloud computing reference architecture focuses on the requirements of “what” cloud services provide, not a “how to” design solution and implementation. The reference architecture is intended to facilitate the understanding of the operational intricacies in cloud computing. It does not represent the system architecture of a specific cloud computing system; instead it is a tool for describing, discussing, and developing a system-specific architecture using a common framework of reference.”

# Objectives

NIST developed the reference architecture with the following objectives in mind:

To illustrate and understand the various cloud services in the context of an overall cloud computing conceptual model

To provide a technical reference for CSCs to understand, discuss, categorize, and compare cloud services

To facilitate the analysis of candidate standards for security, interoperability, and portability and reference implementations

# NIST Cloud Computing Reference Architecture

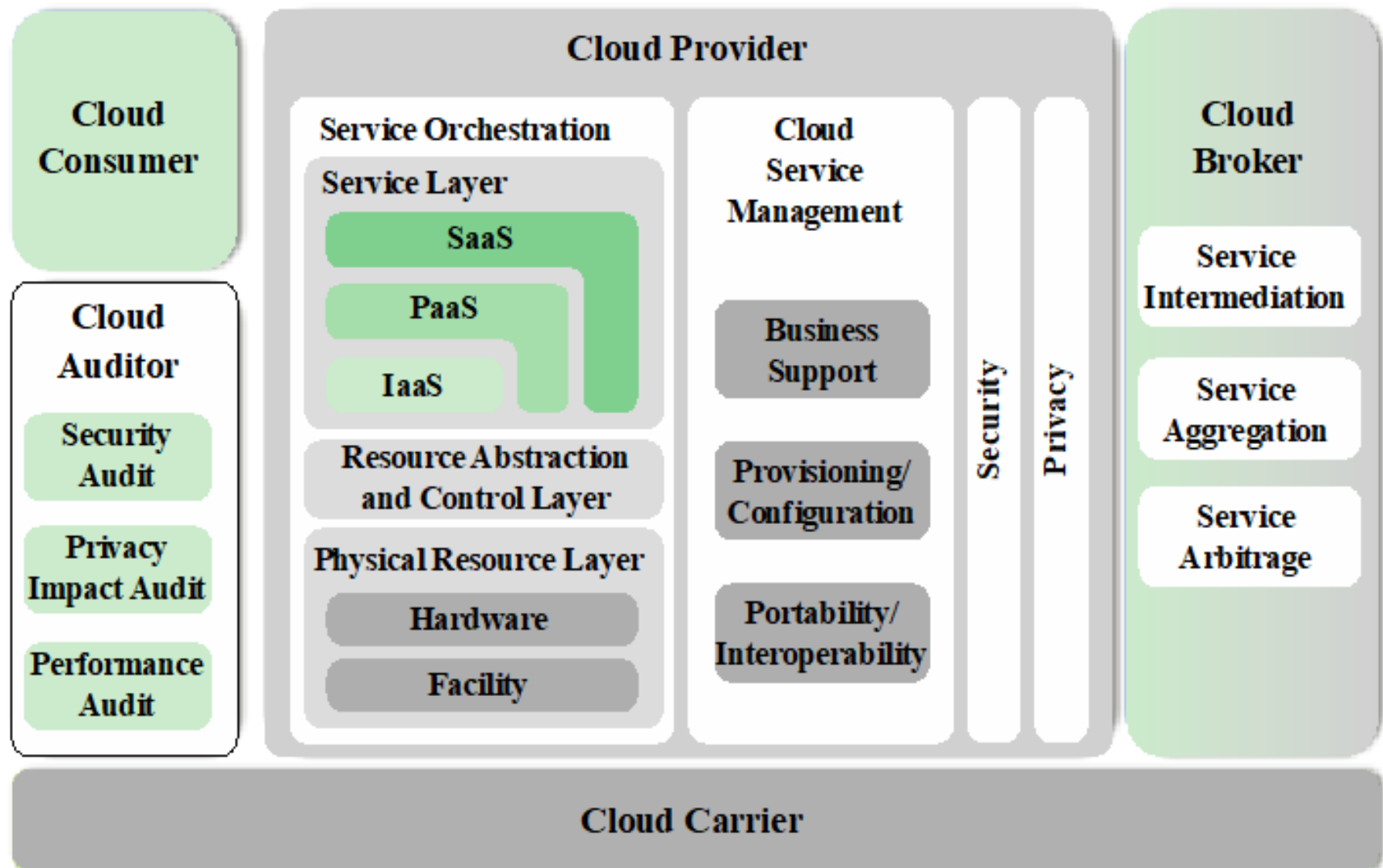


Figure 13.3 NIST Cloud Computing Reference Architecture

# Major Actors in NIST Reference Architecture

The reference architecture, depicted in Figure 13.3, defines five major actors in terms of the roles and responsibilities:

- **Cloud Service Consumer (CSC):** A person or organization that maintains a business relationship with, and uses service from, cloud providers.
- **Cloud Service Provider (CSP):** A person, organization, or entity responsible for making a service available to interested parties.
- **Cloud Auditor:** A party that can conduct independent assessment of cloud services, information system operations, performance, and security of the cloud implementation.
- **Cloud Broker:** An entity that manages the use, performance, and delivery of cloud services, and negotiates relationships between CSPs and cloud consumers.
- **Cloud Carrier:** An intermediary that provides connectivity and transport of cloud services from CSPs to cloud consumers.

# Cloud Service Provider and Consumer (CSP & CSC)

- A **cloud service provider** can provide one or more of the cloud services to meet IT and business requirements of **cloud service consumers** .
- For each of the three service models (SaaS, PaaS, IaaS), the CSP provides the storage and processing facilities needed to support that service model, together with a cloud interface for cloud service consumers.
- For SaaS, the CSP deploys, configures, maintains, and updates the operation of the software applications on a cloud infrastructure so that the services are provisioned at the expected service levels to cloud consumers.
- The consumers of SaaS can be organizations that provide their members with access to software applications, end users who directly use software applications, or software application administrators who configure applications for end users.



# Cloud Service Provider and Consumer (CSP & CSC)

- For PaaS, the CSP manages the computing infrastructure for the platform and runs the cloud software that provides the components of the platform, such as runtime software execution stacks, databases, and other middleware components.
- Cloud consumers of PaaS can employ the tools and execution resources provided by CSPs to develop, test, deploy, and manage the applications hosted in a cloud environment.
- For IaaS, the CSP acquires the physical computing resources underlying the service, including the servers, networks, storage, and hosting infrastructure. The IaaS CSC in turn uses these computing resources, such as a virtual machine, for their fundamental computing needs.

# Cloud Broker

A cloud broker is useful when cloud services are too complex for a cloud consumer to easily manage. A cloud broker can offer three areas of support:

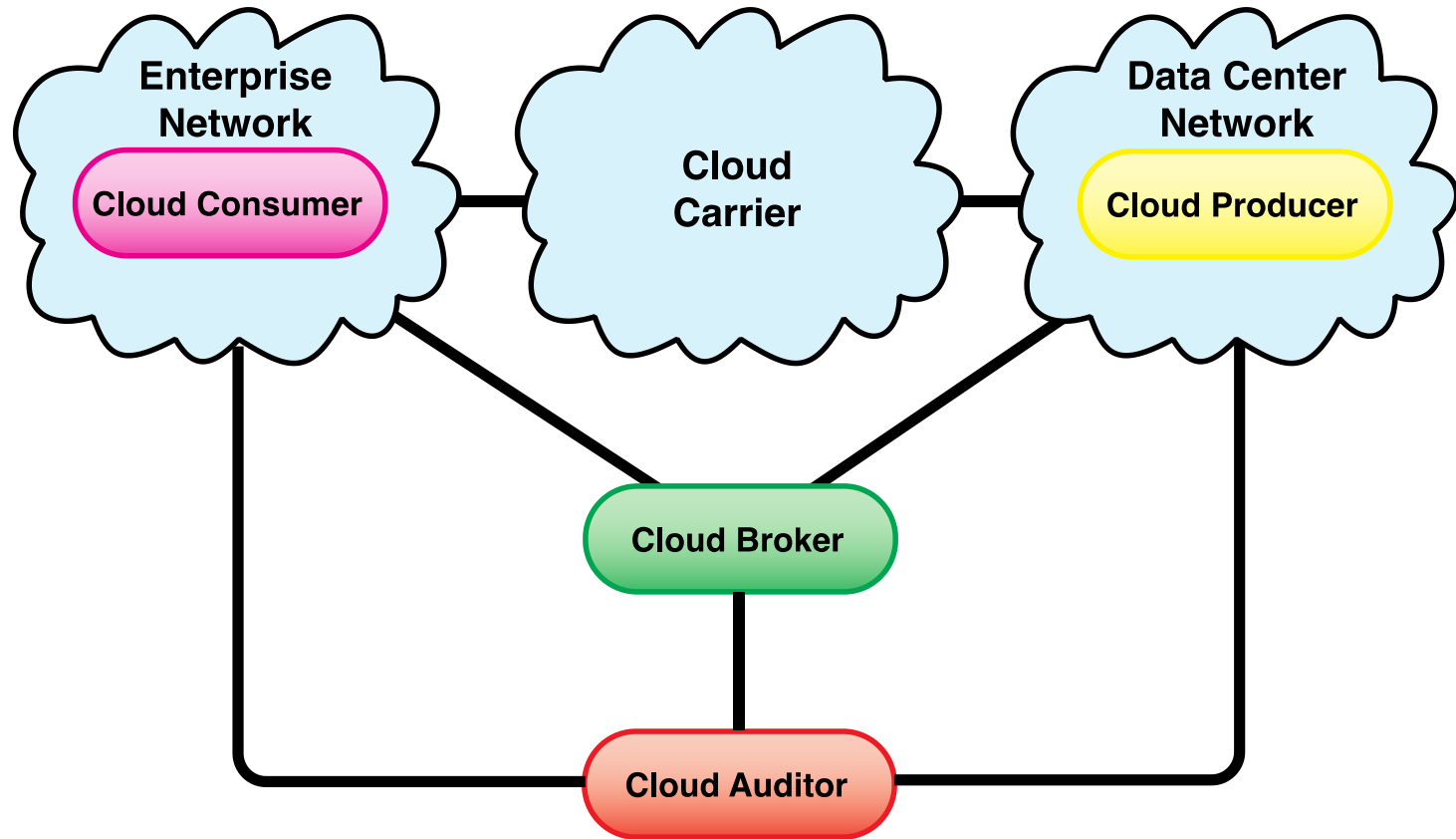
- **Service intermediation:** These are value-added services, such as identity management, performance reporting, and enhanced security.
- **Service aggregation:** The broker combines multiple cloud services to meet consumer needs not specifically addressed by a single CSP, or to optimize performance or minimize cost.
- **Service arbitrage:** This is similar to service aggregation except that the services being aggregated are not fixed. Service arbitrage means a broker has the flexibility to choose services from multiple agencies. The cloud broker, for example, can use a credit-scoring service to measure and select an agency with the best score.

# Cloud Carrier

The **cloud carrier** is a networking facility that provides connectivity and transport of cloud services between cloud consumers and CSPs. Typically, a CSP will set up service level agreements (SLAs) with a cloud carrier to provide services consistent with the level of SLAs offered to cloud consumers, and may require the cloud carrier to provide dedicated and secure connections between cloud consumers and CSPs.

# Cloud Auditor

A **cloud auditor** can evaluate the services provided by a CSP in terms of security controls, privacy impact, performance, and so on. The auditor is an independent entity that can assure that the CSP conforms to a set of standards.



**Figure 13.4 Interactions Between Actors in Cloud Computing**

# Interactions between the Actors

Figure 13.4 illustrates the interactions between the actors.

- A cloud consumer may request cloud services from a cloud provider directly or via a cloud broker. A cloud auditor conducts independent audits and may contact the others to collect necessary information.
- This figure shows that cloud networking issues involve three separate types of networks.
- For a cloud producer, the network architecture is that of a typical large datacenter, which consists of racks of high-performance servers and storage devices, interconnected with high-speed top-of-rack Ethernet switches. The concerns in this context focus on virtual machine placement and movement, load balancing, and availability issues. The enterprise network is likely to have a quite different architecture, typically including a number of LANs, servers, workstations, PCs, and mobile devices, with a broad range of network performance, security, and management issues.
- The concern of both producer and consumer with respect to the cloud carrier, which is shared with many users, is the ability to create virtual networks, with appropriate SLA and security guarantees.

# NIST Guidelines on Cloud Security and Privacy Issues and Recommendations

## **Governance**

Extend organizational practices pertaining to the policies, procedures, and standards used for application development and service provisioning in the cloud, as well as the design, implementation, testing, use, and monitoring of deployed or engaged services.

Put in place audit mechanisms and tools to ensure organizational practices are followed throughout the system lifecycle.

## **Compliance**

Understand the various types of laws and regulations that impose security and privacy obligations on the organization and potentially impact cloud computing initiatives, particularly those involving data location, privacy and security controls, records management, and electronic discovery requirements.

Review and assess the cloud provider's offerings with respect to the organizational requirements to be met and ensure that the contract terms adequately meet the requirements.

Ensure that the cloud provider's electronic discovery capabilities and processes do not compromise the privacy or security of data and applications.

## **Trust**

Ensure that service arrangements have sufficient means to allow visibility into the security and privacy controls and processes employed by the cloud provider, and their performance over time.

Establish clear, exclusive ownership rights over data.

Institute a risk management program that is flexible enough to adapt to the constantly evolving and shifting risk landscape for the lifecycle of the system.

Continuously monitor the security state of the information system to support ongoing risk management decisions.



# NIST Guidelines on Cloud Security and Privacy Issues and Recommendations

## **Architecture**

Understand the underlying technologies that the cloud provider uses to provision services, including the implications that the technical controls involved have on the security and privacy of the system, over the full system lifecycle and across all system components.

## **Identity and access management**

Ensure that adequate safeguards are in place to secure authentication, authorization, and other identity and access management functions, and are suitable for the organization.

## **Software isolation**

Understand virtualization and other logical isolation techniques that the cloud provider employs in its multi-tenant software architecture, and assess the risks involved for the organization.

## **Data protection**

Evaluate the suitability of the cloud provider's data management solutions for the organizational data concerned and the ability to control access to data, to secure data while at rest, in transit, and in use, and to sanitize data.

Take into consideration the risk of collating organizational data with those of other organizations whose threat profiles are high or whose data collectively represent significant concentrated value.

Fully understand and weigh the risks involved in cryptographic key management with the facilities available in the cloud environment and the processes established by the cloud provider.



# NIST Guidelines on Cloud Security and Privacy Issues and Recommendations

## **Availability**

Understand the contract provisions and procedures for availability, data backup and recovery, and disaster recovery, and ensure that they meet the organization's continuity and contingency planning requirements.

Ensure that during an intermediate or prolonged disruption or a serious disaster, critical operations can be immediately resumed, and that all operations can be eventually reinstituted in a timely and organized manner.

## **Incident response**

Understand the contract provisions and procedures for incident response and ensure that they meet the requirements of the organization.

Ensure that the cloud provider has a transparent response process in place and sufficient mechanisms to share information during and after an incident.

Ensure that the organization can respond to incidents in a coordinated fashion with the cloud provider in accordance with their respective roles and responsibilities for the computing environment.

# Security Issues for Cloud Computing

- Security is a major consideration when augmenting or replacing on-premises systems with cloud services
- Allaying security concerns is frequently a prerequisite for further discussions about migrating part or all of an organization's computing architecture to the cloud
- Availability is another major concern
- Auditability of data must be ensured
- Businesses should perform due diligence on security threats both from outside and inside the cloud
  - Cloud users are responsible for application-level security
  - Cloud vendors are responsible for physical security and some software security
  - Security for intermediate layers of the software stack is shared between users and vendors
- Cloud providers must guard against theft or denial-of-service attacks by their users and users need to be protected from one another
- Businesses should consider the extent to which subscribers are protected against the provider, especially in the area of inadvertent data loss

# Risks and Countermeasures

The Cloud Security Alliance lists the following as the top cloud-specific security threats:

- Abuse and nefarious use of cloud computing

For many CSPs, it is relatively easy to register and begin using cloud services, some even offering free limited trial periods. This enables attackers to get inside the cloud to conduct various attacks, such as spamming, malicious code attacks, and denial of service. PaaS providers have traditionally suffered most from this kind of attacks; however, recent evidence shows that hackers have begun to target IaaS vendors as well. The burden is on the CSP to protect against such attacks, but cloud service clients must monitor activity with respect to their data and resources to detect any malicious behavior.

- Countermeasures include:

- Stricter initial registration and validation processes
- Enhanced credit card fraud monitoring and coordination
- Comprehensive inspection of customer network traffic
- Monitoring public blacklists for one's own network blocks

# Risks and Countermeasures

- Insecure interfaces and APIs

CSPs expose a set of software interfaces or APIs that customers use to manage and interact with cloud services. The security and availability of general cloud services is dependent upon the security of these basic APIs. From authentication and access control to encryption and activity monitoring, these interfaces must be designed to protect against both accidental and malicious attempts to circumvent policy.

- Countermeasures include:

- Analyzing the security model of CSP interfaces
- Ensuring that strong authentication and access controls are implemented in concert with encrypted transmission
- Understanding the dependency chain associated with the API

# Risks and Countermeasures

- Malicious insiders:

Under the cloud computing paradigm, an organization relinquishes direct control over many aspects of security and, in doing so, confers an unprecedented level of trust onto the CSP. One grave concern is the risk of malicious insider activity. Cloud architectures necessitate certain roles that are extremely high risk. Examples include CSP system administrators and managed security service providers.

- Countermeasures include:

- Enforce strict supply chain management and conduct a comprehensive supplier assessment
- Specify human resource requirements as part of legal contract
- Require transparency into overall information security and management practices, as well as compliance reporting
- Determine security breach notification processes

# Risks and Countermeasures

- Shared technology issues:

IaaS vendors deliver their services in a scalable way by sharing infrastructure. Often, the underlying components that make up this infrastructure (CPU caches, GPUs, etc.) were not designed to offer strong isolation properties for a multi-tenant architecture. CSPs typically approach this risk by using isolated VMs for individual clients. This approach is still vulnerable to attack, by both insiders and outsiders, and so can only be a part of an overall security strategy.

- Countermeasures include:

- Implement security best practices for installation/configuration
- Monitor environment for unauthorized changes/activity
- Promote strong authentication and access control for administrative access and operations
- Enforce SLAs for patching and vulnerability remediation
- Conduct vulnerability scanning and configuration audits

# Risks and Countermeasures

- Data loss or leakage

There are many ways to compromise data. Deletion or alteration of records without a backup of the original content is an obvious example. Unlinking a record from a larger context may render it unrecoverable, as can storage on unreliable media. Loss of an encoding key may result in effective destruction. Finally, unauthorized parties must be prevented from gaining access to sensitive data.

The threat of data compromise increases in the cloud, due to the number of, and interactions between, risks and challenges that are either unique to the cloud or more dangerous because of the architectural or operational characteristics of the cloud environment.

Data must be secured while at rest, in transit, and in use, and access to the data must be controlled. The client can employ encryption to protect data in transit, though this involves key management responsibilities for the CSP. The client can enforce access control techniques, but, again, the CSP is involved to some extent depending on the service model used.



# Risks and Countermeasures

- Data loss or leakage

For data at rest, the ideal security measure is for the client to encrypt the database and only store encrypted data in the cloud, with the CSP having no access to the encryption key. So long as the key remains secure, the CSP has no ability to decipher the data, although corruption and other denial-of-service attacks remain a risk.

- Countermeasures include:

- Implement strong API access control
- Encrypt and protect integrity of data in transit and at rest
- Analyze data protection at both design and run time
- Implement strong key generation, storage and management, and destruction practices



# Risks and Countermeasures

- Account or service hijacking

Account and service hijacking, usually with stolen credentials, remains a top threat. With stolen credentials, attackers can often access critical areas of deployed cloud computing services, allowing them to compromise the confidentiality, integrity, and availability of those services.

- Countermeasures include:

- Prohibit the sharing of account credentials between users and services
- Leverage strong two-factor authentication techniques where possible
- Employ proactive monitoring to detect unauthorized activity
- Understand CSP security policies and SLAs

# Risks and Countermeasures

- Unknown risk profile:

In using cloud infrastructures, the client necessarily cedes control to the cloud provider on a number of issues that may affect security. Thus the client must pay attention to and clearly define the roles and responsibilities involved for managing risks. For example, employees may deploy applications and data resources at the CSP without observing the normal policies and procedures for privacy, security, and oversight.

- Countermeasures include:

- Disclosure of applicable logs and data
- Partial/full disclosure of infrastructure details
- Monitoring and alerting on necessary information

# Data Protection in the Cloud

The threat of data compromise increases in the cloud, due to the number of, and interactions between, risks and challenges that are either unique to the cloud or more dangerous because of the architectural or operational characteristics of the cloud environment

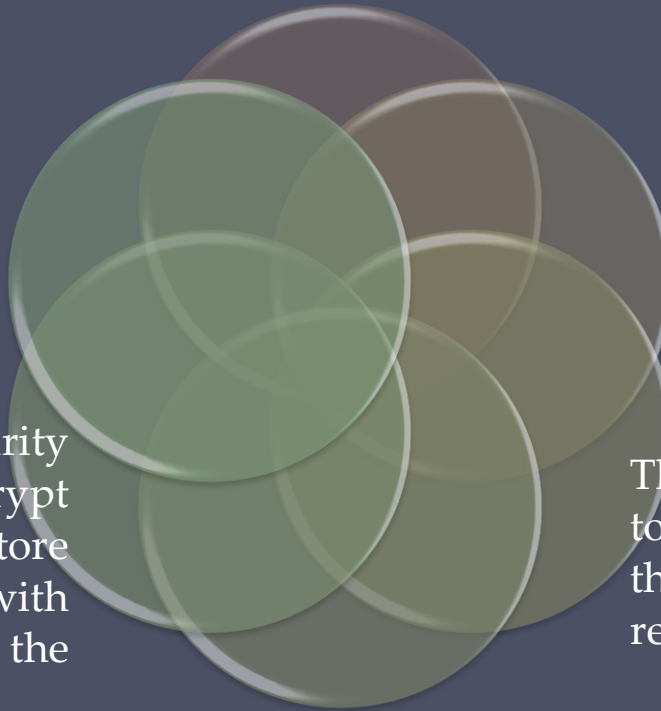
Even with these precautions, corruption and other denial-of-service attacks remain a risk

For data at rest, the ideal security measure is for the client to encrypt the database and only store encrypted data in the cloud, with the CSP having no access to the encryption key

Data must be secured while at rest, in transit, and in use, and access to the data must be controlled

The client can employ encryption to protect data in transit, though this involves key management responsibilities for the CSP

The client can enforce access control techniques, but CSP is involved to some extent depending on the service model used



# Data Protection in the Cloud

## Multi-instance Model

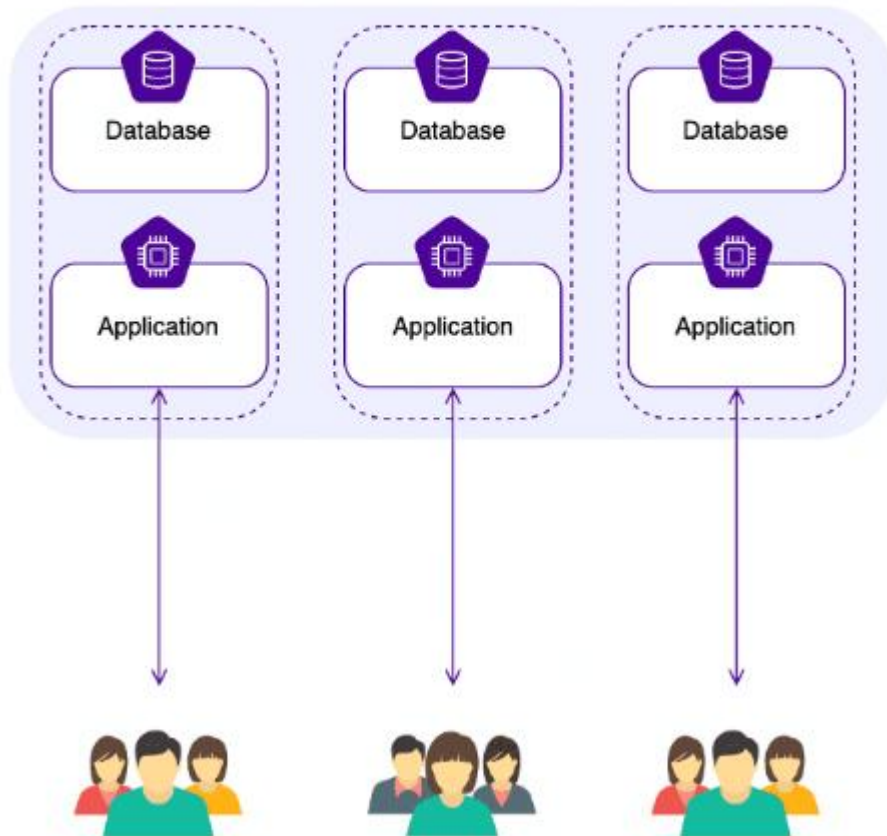
- Provides a unique DBMS running on a VM instance for each cloud subscriber
- This gives the subscriber complete control over role definition, user authorization, and other administrative tasks related to security

## Multi-tenant Model

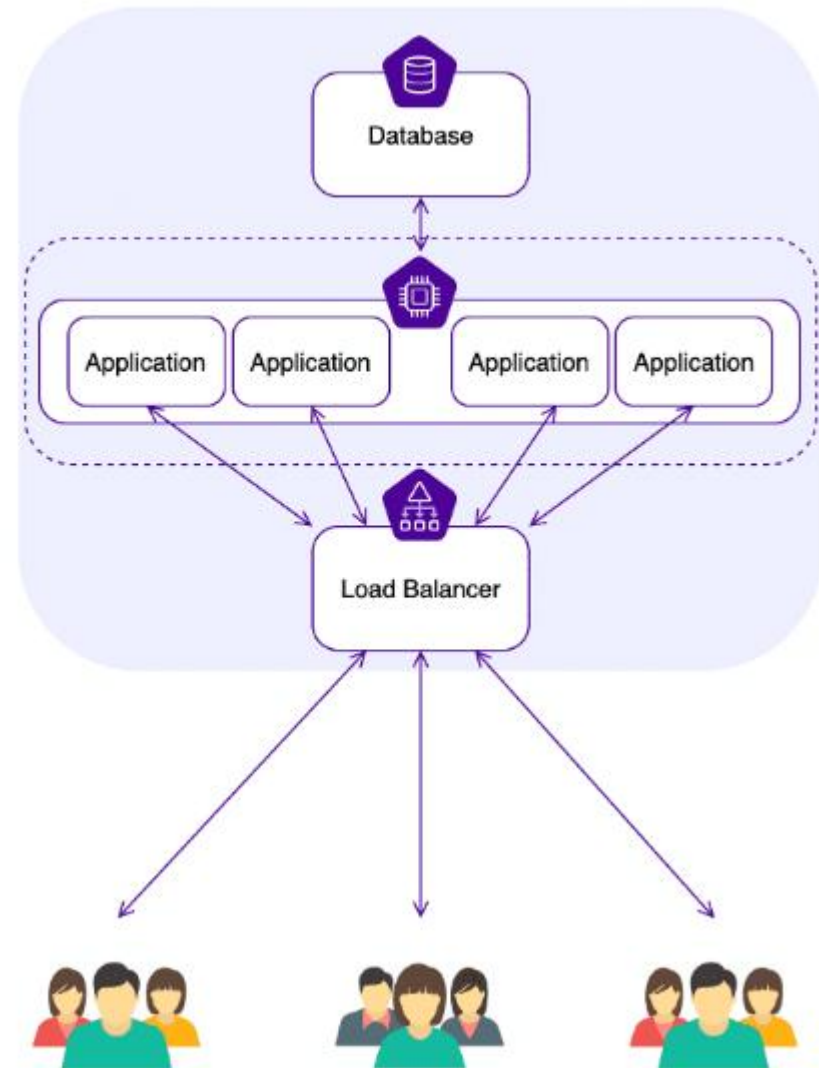
- Provides a predefined environment for the cloud subscriber that is shared with other tenants, typically through tagging data with a subscriber identifier
- Tagging gives the appearance of exclusive use of the instance, but relies on the cloud provider to establish and maintain a sound secure database environment

# Multi-instance vs Multi-tenant Model

Multi-instance

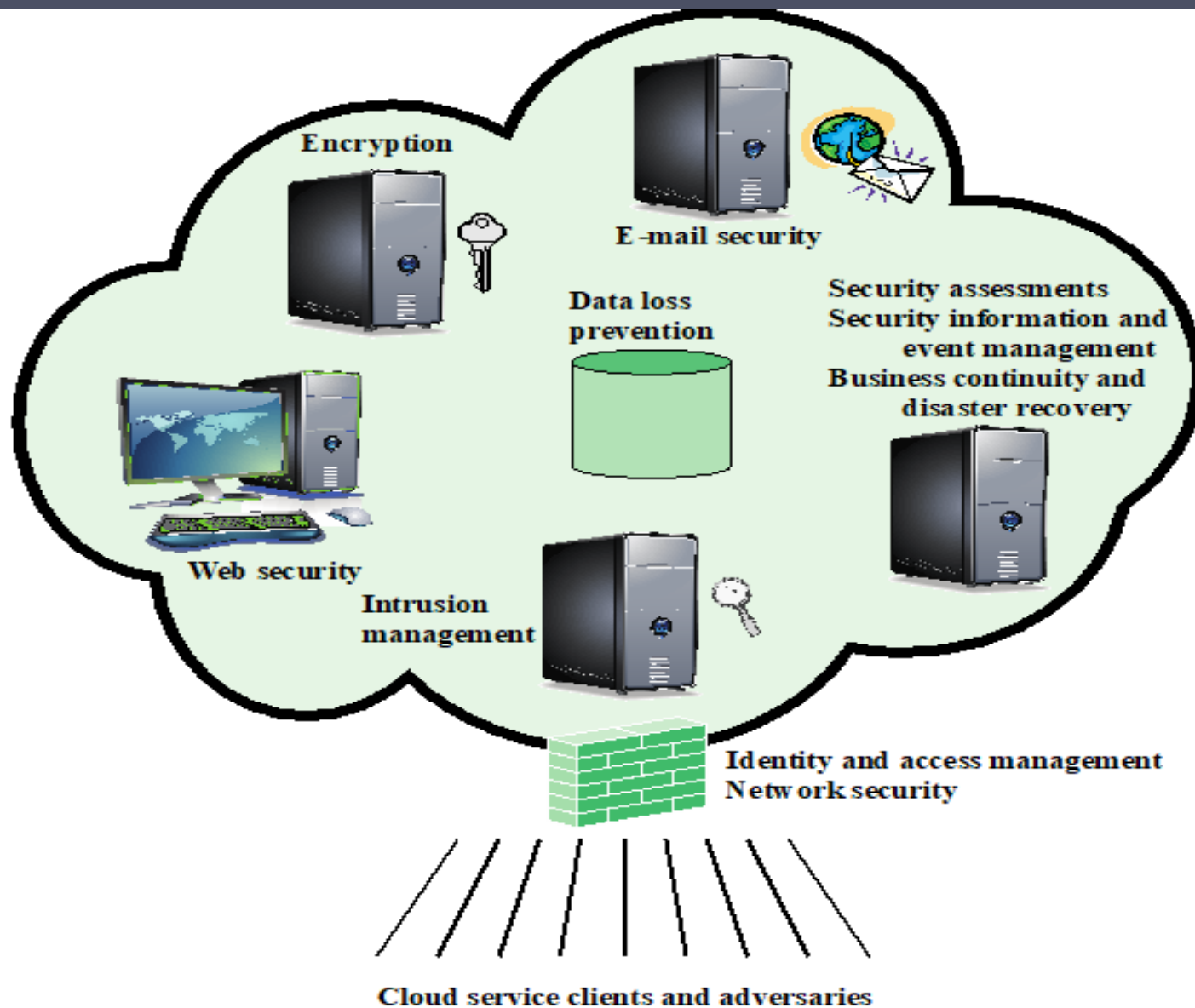


Multi-tenant



# Cloud Security as a Service(SecaaS)

- In the context of cloud computing, cloud security as a service, designated SecaaS, is a segment of the SaaS offering of a CSP
- The CSA defines SecaaS as the provision of security applications and services via the cloud either to cloud-based infrastructure and software, or from the cloud to the customers' on-premise systems
- The CSA has identified the following SecaaS categories of service:
  - Identity and access management
  - Data loss prevention
  - Web security
  - E-mail security
  - Security assessments
  - Intrusion management
  - Security information and event management
  - Encryption
  - Business continuity and disaster recovery
  - Network security



**Figure 13.6 Elements of Cloud Security as a Service**



# Identity and Access Management (IAM)

- IAM includes people, processes, and systems that are used to manage access to enterprise resources by assuring that the identity of an entity is verified, then granting the correct level of access based on this assured identity. One aspect of identity management is identity provisioning, which has to do with providing access to identified users and subsequently deprovisioning, or denying access, to users when the client enterprise designates such users as no longer having access to enterprise resources in the cloud. Among other requirements, the cloud service provider must be able to exchange identity attributes with the enterprise's chosen identity provider.
- The access management portion of IAM involves authentication and access control services. For example, the CSP must be able to authenticate users in a trustworthy manner. The access control requirements in SPI environments include establishing trusted user profile and policy information, using it to control access within the cloud service, and doing this in an auditable way.



# Data loss prevention (DLP)

- **Data loss prevention (DLP)** is the monitoring, protecting, and verifying the security of data at rest, in motion, and in use. Much of DLP can be implemented by the cloud client, such as discussed in previously in this section (Data Protection in the Cloud). The CSP can also provide DLP services, such as implementing rules about what functions can be performed on data in various contexts.

# Web security

- **Web security** is real-time protection offered either on premise through software/ appliance installation or via the cloud by proxying or redirecting Web traffic to the CSP.
- This provides an added layer of protection on top of things like antiviruses to prevent malware from entering the enterprise via activities such as Web browsing.
- In addition to protecting against malware, a cloud-based Web security service might include usage policy enforcement, data backup, traffic control, and Web access control.

# E-mail security

- A CSP may provide a Web-based e-mail service, for which security measures are needed.
- **E-mail security** provides control over inbound and outbound e-mail, protecting the organization from phishing, malicious attachments, enforcing corporate policies such as acceptable use and spam prevention.
- The CSP may also incorporate digital signatures on all e-mail clients and provide optional e-mail encryption.

# Security assessments

- **Security assessments** are third-part audits of cloud services. While this service is outside the province of the CSP, the CSP can provide tools and access points to facilitate various assessment activities.

# Intrusion Management

- **Intrusion management** encompasses intrusion detection, prevention, and response.
- The core of this service is the implementation of intrusion detection systems (IDSs) and intrusion prevention systems (IPSs) at entry points to the cloud and on servers in the cloud.
- An IDS is a set of automated tools designed to detect unauthorized access to a host system.
- An IPS incorporates IDS functionality and in addition includes mechanisms designed to block traffic from intruders.

# Security Information and Event Management (SIEM)

- Security information and event management (SIEM) aggregates (via push or pull mechanisms) log and event data from virtual and real networks, applications, and systems.
- This information is then correlated and analyzed to provide real-time reporting and alerting on information/events that may require intervention or other type of response.
- The CSP typically provides an integrated service that can put together information from a variety of sources both within the cloud and within the client enterprise network.

# Encryption

- **Encryption** is a pervasive service that can be provided for data at rest in the cloud, e-mail traffic, client-specific network management information, and identity information.
- Encryption services provided by the CSP involve a range of complex issues, including key management, how to implement virtual private network (VPN) services in the cloud, application encryption, and data content access.

# Business continuity and disaster recovery

- **Business continuity and disaster recovery** comprise measures and mechanisms to ensure operational resiliency in the event of any service interruptions.
- This is an area where the CSP, because of economies of scale, can offer obvious benefits to a cloud service client.
- The CSP can provide backup at multiple locations, with reliable failover and disaster recovery facilities.
- This service must include a flexible infrastructure, redundancy of functions and hardware, monitored operations, geographically distributed data centers, and network survivability.



# Network security

- **Network security** consists of security services that allocate access, distribute, monitor, and protect the underlying resource services.
- Services include perimeter and server firewalls and denial-of-service protection. Many of the other services listed in this section, including intrusion management, identity and access management, data loss protection, and Web security, also contribute to the network security service.

# OpenStack

Open-source software project of the OpenStack Foundation that aims to produce an open-source cloud operating system

The principal objective is to enable creating and managing huge groups of virtual private servers in a cloud computing environment

OpenStack is embedded, to one degree or another, into data center infrastructure and cloud computing products

It provides multi-tenant IaaS, and aims to meet the needs of public and private clouds, regardless of size, by being simple to implement and massively scalable

# OpenStack

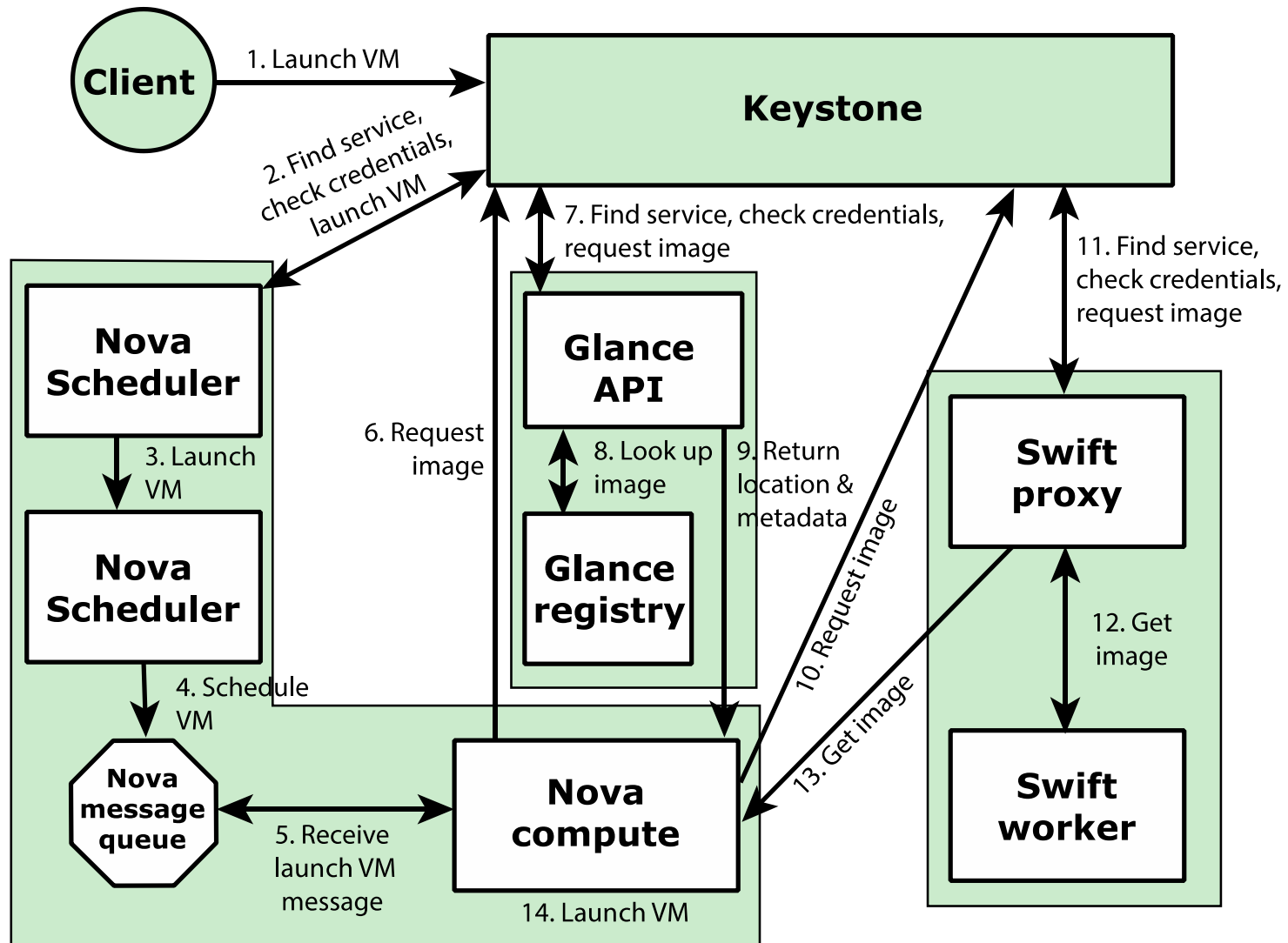
- The OpenStack OS consists of a number of independent modules, each of which has a project name and a functional name. The modular structure is easy to scale out and provides a commonly used set of core services. Typically, the components are configured together to provide a comprehensive IaaS capability. However, the modular design is such that the components are generally capable of being used independently.
- The security module for OpenStack is Keystone.
- Keystone provides the shared security services essential for a functioning cloud computing infrastructure
  - It provides the following main services:
    - Identity
    - Token
    - Service catalog
    - Policies

# OpenStack Services

- **Identity:** This is user information authentication. This information defines a user's role and permissions within a project, and is the basis for a role-based access control (RBAC) mechanism. Keystone supports multiple methods of authentication, including user name and password, Lightweight Directory access Protocol (LDAP), and a means of configuring external authentication methods supplied by the CSC.
- **Token:** After authentication, a token is assigned and used for access control. OpenStack services retain tokens and use them to query Keystone during operations.

# OpenStack Services

- **Service catalog:** OpenStack service endpoints are registered with Keystone to create a service catalog. A client for a service connects to Keystone and determines an endpoint to call based on the returned catalog.
- **Policies:** This service enforces different user access levels. Each OpenStack service defines the access policies for its resources in an associated policy file. A resource, for example, could be API access, the ability to attach to a volume, or to fire up instances. These policies can be modified or updated by the cloud administrator to control the access to the various resources.



**Figure 13.7 Launching a Virtual Machine in OpenStack**

# Launching a VM in OpenStack

- Figure 13.7 illustrates the way in which Keystone interacts with other Open-Stack components to launch a new VM.
- Nova is the management software module that controls VMs within the IaaS cloud computing platform. It manages the lifecycle of compute instances in an OpenStack environment.
- Responsibilities include spawning, scheduling, and decommissioning of machines on demand. Thus, Nova enables enterprises and service providers to offer on-demand computing resource by provisioning and managing large networks of VMs.
- Glance is a lookup and retrieval system for VM disk images. It provides services for discovering, registering, and retrieving virtual images through an API. Swift is a distributed object store that creates a redundant and scalable storage space of up to multiple petabytes of data.
- Object storage does not present a traditional file system, but rather a distributed storage system for static data such as VM images, photo storage, e-mail storage, backups, and archives.

# The Internet of Things(IoT)

- IoT is a term that refers to the expanding interconnection of smart devices, ranging from appliances to tiny sensors
  - A dominant theme is the embedding of short-range mobile transceivers into a wide array of gadgets and everyday items, enabling new forms of communication between people and things, and between things themselves
  - The Internet supports the interconnectivity usually through cloud systems
- The objects deliver sensor information, act on their environment, and in some cases modify themselves, to create overall management of a larger system
- The IoT is primarily driven by deeply embedded devices
  - These devices are low-bandwidth, low-repetition data capture, and low-bandwidth data-usage appliances that communicate with each other and provide data via user interfaces
  - Embedded appliances, such as high-resolution video security cameras, video VoIP phones, and a handful of others, require high-bandwidth streaming capabilities



# Evolution

With reference to the end systems supported, the Internet has gone through roughly four generations of deployment culminating in the IoT:

---

## Information technology (IT)

---

PCs, servers, routers, firewalls, and so on, bought as IT devices by enterprise IT people, primarily using wired connectivity

---

## Operational technology (OT)

---

Machines/appliances with embedded IT built by non-IT companies, such as medical machinery, SCADA, process control, and kiosks, bought as appliances by enterprise OT people, primarily using wired connectivity

---

## Personal technology

---

Smartphones, tablets, and eBook readers bought as IT devices by consumers (employees) exclusively using wireless connectivity and often multiple forms of wireless connectivity

---

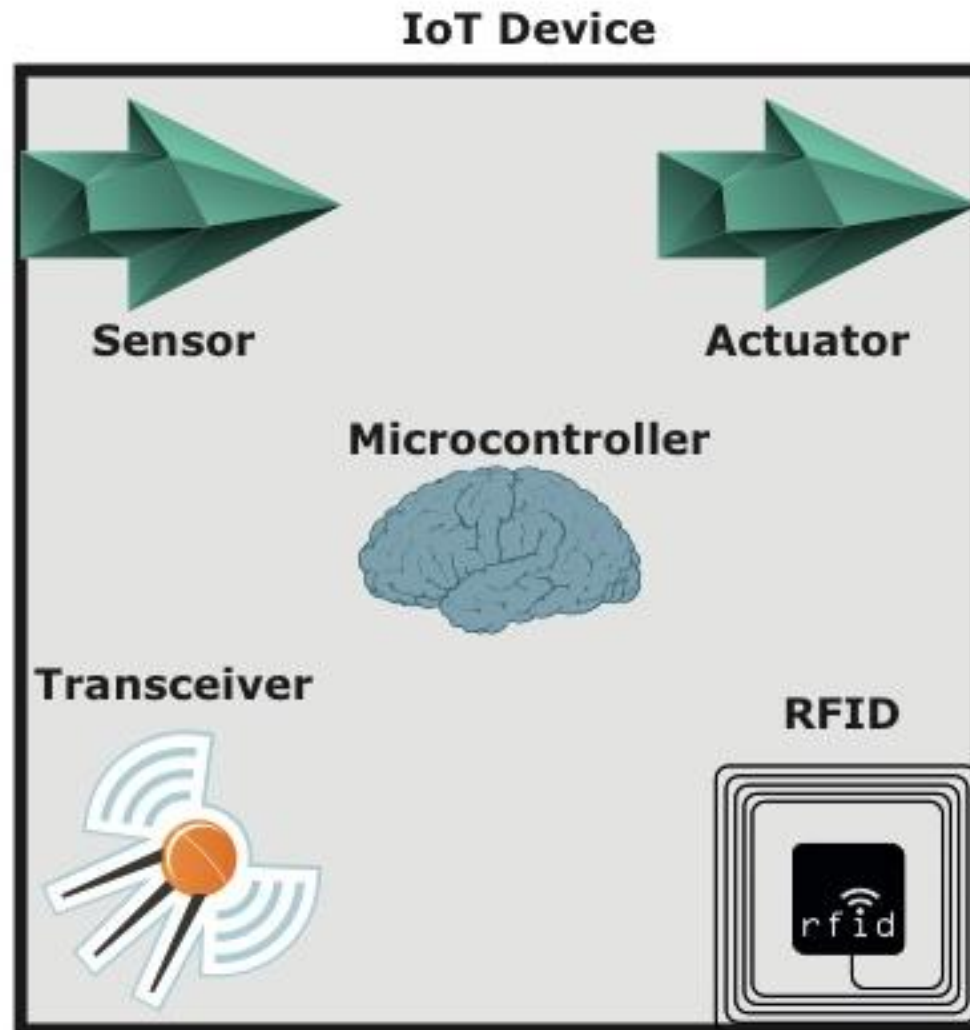
## Sensor/actuator technology

---

Single-purpose devices bought by consumers, IT and OT people exclusively using wireless connectivity, generally of a single form, as part of larger systems

---

It is the fourth generation that is usually thought of as the IoT, and which is marked by the use of billions of embedded devices



**Figure 13.8 IoT Components**

# IoT Components

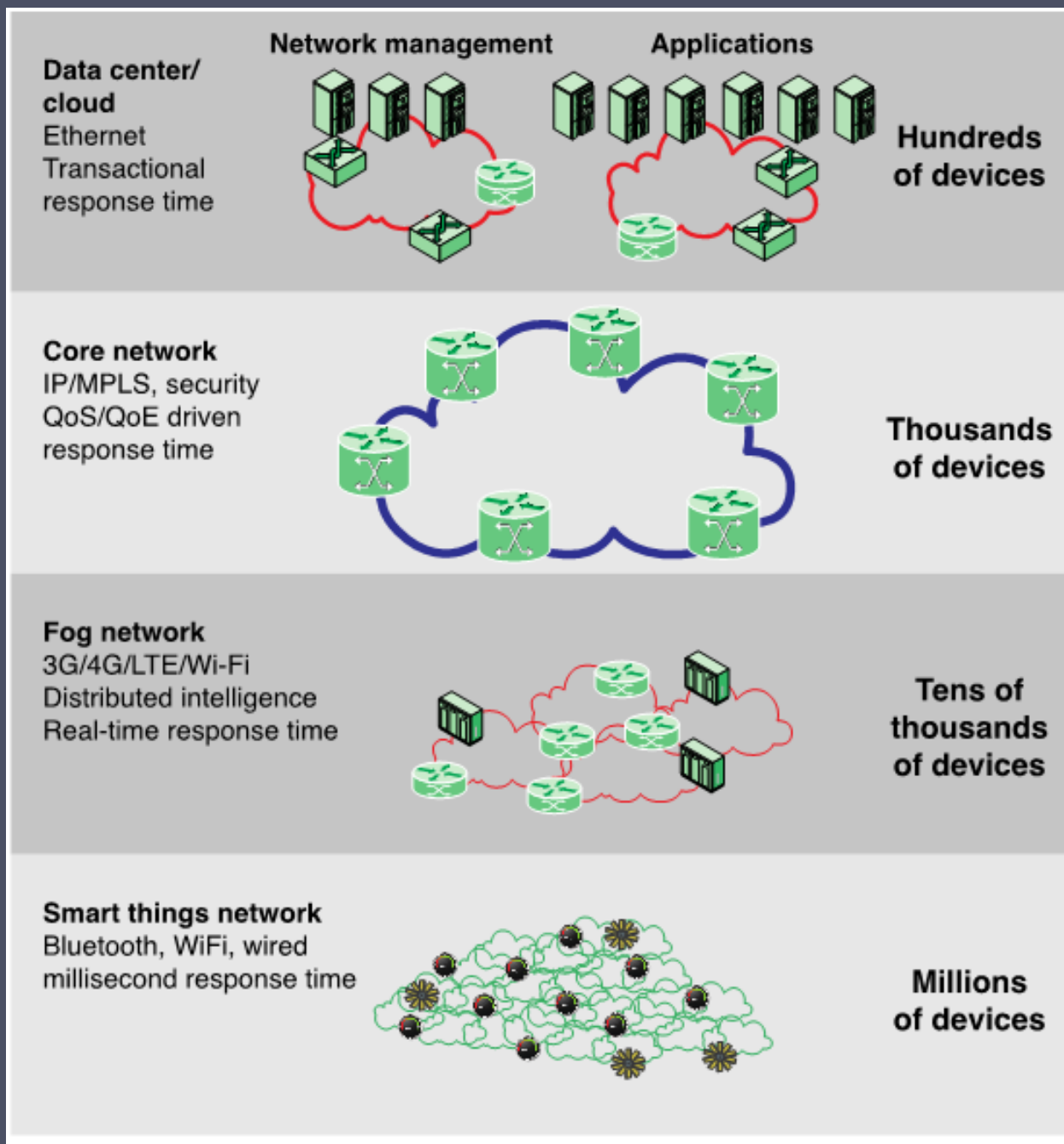
- **Sensor:** A sensor measures some parameter of a physical, chemical, or biological entity and delivers an electronic signal proportional to the observed characteristic, either in the form of an analog voltage level or a digital signal. In both cases, the sensor output is typically input to a microcontroller or other management element.
- **Actuator:** An actuator receives an electronic signal from a controller and responds by interacting with its environment to produce an effect on some parameter of a physical, chemical, or biological entity.

# IoT Components

- **Microcontroller:** The “smart” in a smart device is provided by a deeply embedded microcontroller.
- **Transceiver:** A transceiver contains the electronics needed to transmit and receive data. Most IoT devices contain a wireless transceiver, capable of communication using Wi-Fi, ZigBee, or some other wireless scheme.


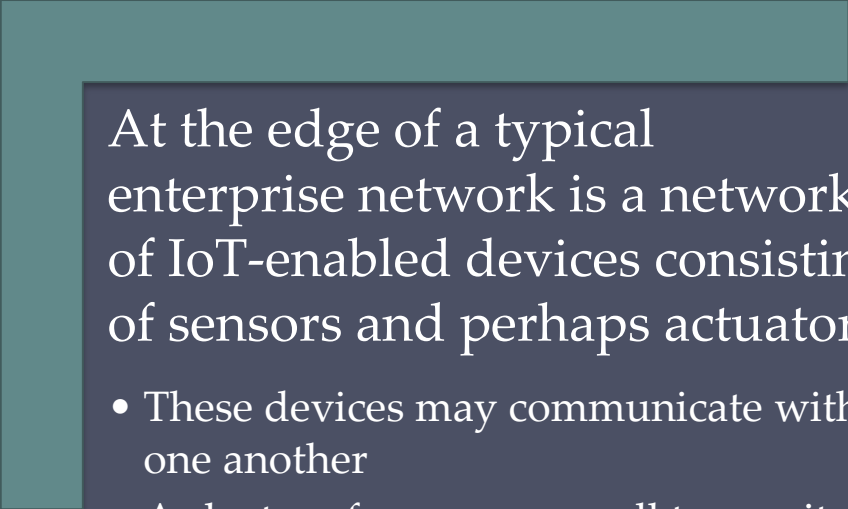
# IoT Components

- **Radio-frequency Identification (RFID):**
- (RFID) technology, which uses radio waves to identify items, is increasingly becoming an enabling technology for IoT.
- The main elements of an RFID system are tags and readers. RFID tags are small programmable devices used for object, animal, and human tracking.
- They come in a variety of shapes, sizes, functionalities, and costs. RFID readers acquire and sometimes rewrite information stored on RFID tags that come within operating range (a few inches up to several feet).
- Readers are usually connected to a computer system that records and formats the acquired information for further uses.



**Figure 13.9 The IoT/Cloud Context**

# Edge



At the edge of a typical enterprise network is a network of IoT-enabled devices consisting of sensors and perhaps actuators

- These devices may communicate with one another
- A cluster of sensors may all transmit their data to one sensor that aggregates the data to be collected by a higher-level entity



A *gateway* interconnects the IoT-enabled devices with the higher-level communication networks

- It performs the necessary translation between the protocols used in the communication networks and those used by devices
- It may also perform a basic data aggregation function

# Fog

- In many IoT deployments, massive amounts of data may be generated by a distributed network of sensors
- Rather than store all of that data permanently (or at least for a long period) in central storage accessible to IoT applications, it is often desirable to do as much data processing close to the sensors as possible
- The purpose of what is sometimes referred to as the edge computing level is to convert network data flows into information that is suitable for storage and higher-level processing
- Processing elements at these levels may deal with high volumes of data and perform data transformation operations, resulting in the storage of much lower volumes of data
- The following are examples of fog computing operations:

Evaluation

Formatting

Expanding/  
decoding

Distillation/  
reduction

Assessment



# Fog computing operations

- **Evaluation** : Evaluating data for criteria as to whether it should be processed at a higher level.
- **Formatting** : Reformatting data for consistent higher-level processing.
- **Expanding/decoding** : Handling cryptic data with additional context (such as the origin).
- **Distillation/reduction** : Reducing and/or summarizing data to minimize the impact of data and traffic on the network and higher-level processing systems.
- **Assessment** : Determining whether data represents a threshold or alert; this could include redirecting data to additional destinations.

# Fog

- Generally fog computing devices are deployed physically near the edge of the IoT network near the sensors and other data-generating devices
- Fog computing and fog services are expected to be a distinguishing characteristic of the IoT
- Fog computing represents an opposite trend in modern networking from cloud computing
  - With cloud computing, massive, centralized storage and processing resources are made available to distributed customers over cloud networking facilities to a relatively small number of users
  - With fog computing, massive numbers of individual smart objects are interconnected with fog networking facilities that provide processing and storage resources close to the edge devices in an IoT
- Fog computing addresses the challenges raised by the activity of thousands or millions of smart devices, including security, privacy, network capacity constraints, and latency requirements
- The term *fog computing* is inspired by the fact that fog tends to hover low to the ground, whereas clouds are high in the sky

# Core

- The *core network*, also referred to as a *backbone network*, connects geographically dispersed fog networks as well as providing access to other networks that are not part of the enterprise network
- Typically the core network will use very high-performance routers, high-capacity transmission lines, and multiple interconnected routers for increased redundancy and capacity
- The core network may also connect to high-performance, high-capacity servers such as large database servers and private cloud facilities
- Some of the core routers may be purely internal, providing redundancy and additional capacity without serving as edge routers

	Cloud	Fog
Location of processing/storage resources	Center	Edge
Latency	High	Low
Access	Fixed or wireless	Mainly wireless
Support for mobility	Not applicable	Yes
Control	Centralized/hierarchical (full control)	Distributed/hierarchical (partial control)
Service access	Through core	At the edge/on handheld device
Availability	99.99%	Highly volatile/highly redundant
Number of users/devices	Tens/hundreds of millions	Tens of billions
Main content generator	Human	Devices/sensors
Content generation	Central location	Anywhere
Content consumption	End device	Anywhere
Software virtual infrastructure	Central enterprise servers	User devices

**Table 13.4 Comparison of Cloud and Fog Features**

# Patching Vulnerability

There is a crisis point with regard to the security of embedded systems, including IoT devices

The embedded devices are riddled with vulnerabilities and there is no good way to patch them

Chip manufacturers have strong incentives to produce their product as quickly and cheaply as possible

The device manufacturers focus is the functionality of the device itself

The end user may have no means of patching the system or, if so, little information about when and how to patch

The result is that the hundreds of millions of Internet-connected devices in the IoT are vulnerable to attack

This is certainly a problem with sensors, allowing attackers to insert false data into the network

It is potentially a graver threat with actuators, where the attacker can affect the operation of machinery and other devices

# IoT Security and Privacy Requirements

- ITU-T Recommendation Y.2066 includes a list of security requirements for the IoT
- The requirements are defined as being the functional requirements during capturing, storing, transferring, aggregating, and processing the data of things, as well as to the provision of services which involve things
- The requirements are:
  - Communication security
  - Data management security
  - Service provision security
  - Integration of security policies and techniques
  - Mutual authentication and authorization
  - Security audit

# IoT Security and Privacy Requirements

- **Communication security:** Secure, trusted, and privacy protected communication capability is required, so unauthorized access to the content of data can be prohibited, integrity of data can be guaranteed and privacy-related content of data can be protected during data transmission or transfer in IoT.
- **Data management security:** Secure, trusted, and privacy protected data management capability is required, so unauthorized access to the content of data can be prohibited, integrity of data can be guaranteed, and privacy-related content of data can be protected when storing or processing data in IoT.

# IoT Security and Privacy Requirements

- **Service provision security:** Secure, trusted, and privacy protected service provision capability is required, so unauthorized access to service and fraudulent service provision can be prohibited and privacy information related to IoT users can be protected.
- **Integration of security policies and techniques:** The ability to integrate different security policies and techniques is required, so as to ensure a consistent security control over the variety of devices and user networks in IoT.



# IoT Security and Privacy Requirements

- **Mutual authentication and authorization:** Before a device (or an IoT user) can access the IoT, mutual authentication and authorization between the device (or the IoT user) and IoT is required to be performed according to predefined security policies.
- **Security audit:** Security audit is required to be supported in IoT. Any data access or attempt to access IoT applications are required to be fully transparent, traceable and reproducible according to appropriate regulation and laws. In particular, IoT is required to support security audit for data transmission, storage, processing, and application access.

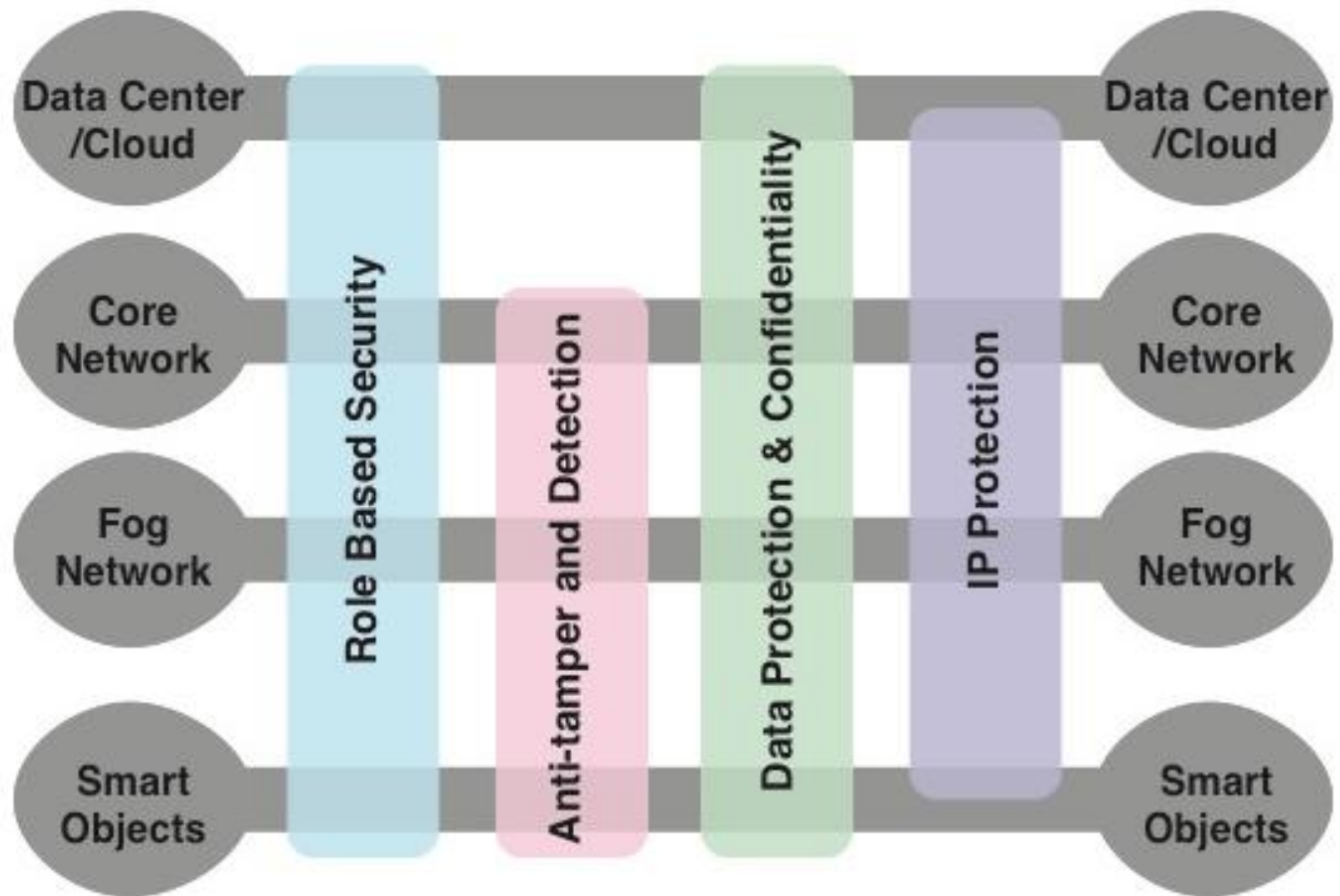


Figure 13.12 IoT Security Environment

# IoT Security Environment

Cisco has developed a framework for IoT security [FRAH15] that serves as a useful guide to the security requirements for IoT. Figure 13.12 illustrates the security environment related to the logical structure of an IoT. The IoT model is a simplified version of the World Forum IoT Reference Model.

It consists of the following levels:

- Smart objects/embedded systems
- Fog/edge network
- Core network
- Data center/cloud

# IoT Security Environment

- **Smart objects/embedded systems:** Consists of sensors, actuators, and other embedded systems at the edge of the network. This is the most vulnerable part of an IoT. The devices may not be in a physically secure environment and may need to function for years. Availability is certainly an issue. Network managers also need to be concerned about the authenticity and integrity of the data generated by sensors and about protecting actuators and other smart devices from unauthorized use. Privacy and protection from eavesdropping may also be requirements.
- **Fog/edge network:** This level is concerned with the wired and wireless interconnection of IoT devices. In addition, a certain amount of data processing and consolidation may be done at this level. A key issue of concern is the wide variety of network technologies and protocols used by the various IoT devices and the need to develop and enforce a uniform security policy.

# IoT Security Environment

- **Core network:** The core network level provides data paths between network center platforms and the IoT devices. The security issues here are those confronted in traditional core networks. However, the vast number of endpoints to interact with and manage creates a substantial security burden.
- **Data center/cloud:** This level contains the application, data storage, and network management platforms. IoT does not introduce any new security issues at this level, other than the necessity of dealing with huge numbers of individual endpoints.

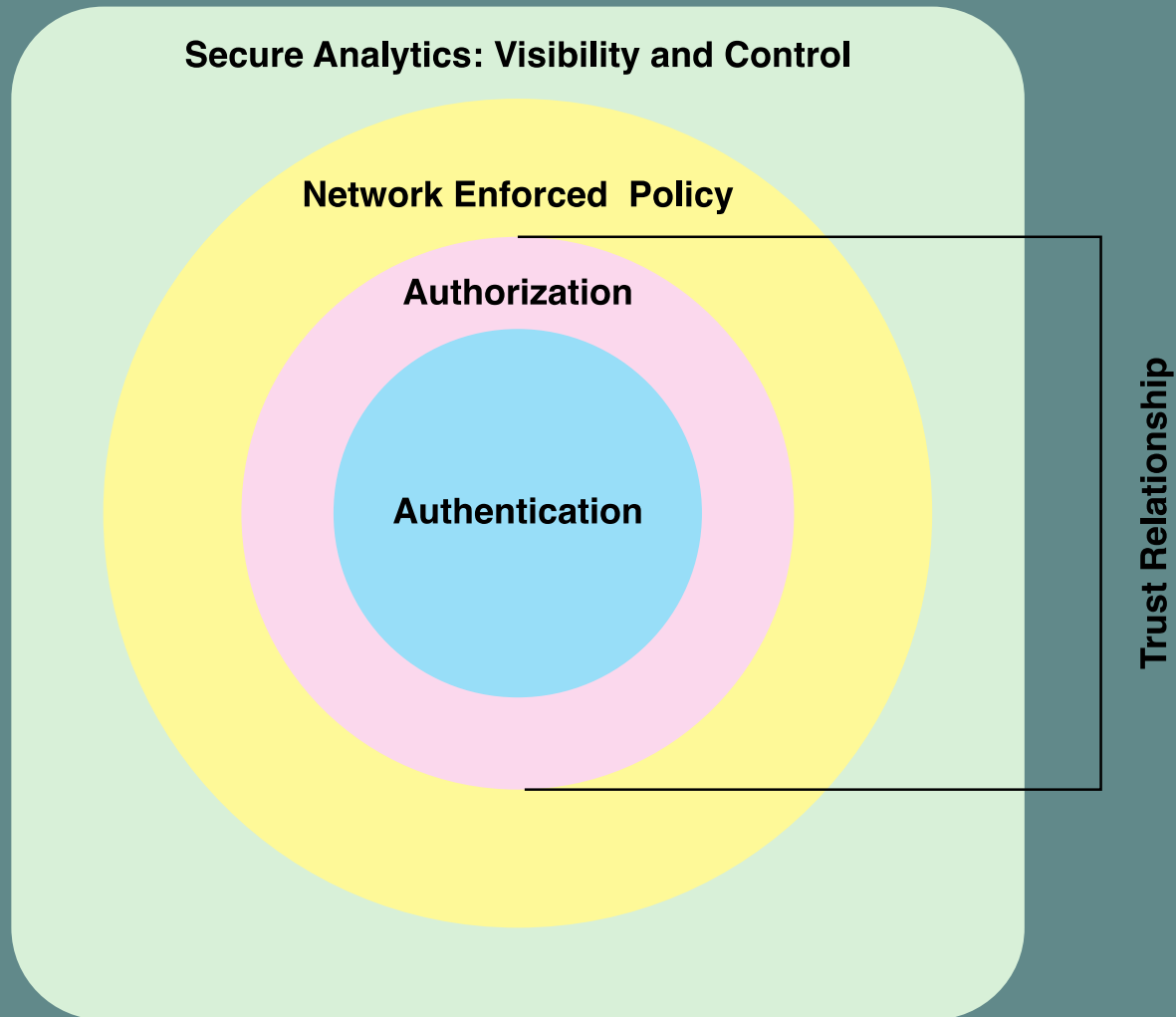
# IoT Security Environment

Within this four-level architecture, the Cisco model defines four general security capabilities that span multiple levels:

- **Role-based security:** RBAC systems assign access rights to roles instead of individual users. In turn, users are assigned to different roles, either statically or dynamically, according to their responsibilities. RBAC enjoys widespread commercial use in cloud and enterprise systems and is a well-understood tool that can be used to manage access to IoT devices and the data they generate.
- **Anti-tamper and detection:** This function is particularly important at the device and fog network levels but also extends to the core network level. All of these levels may involve components that are physically outside the area of the enterprise that is protected by physical security measures.

# IoT Security Environment

- **Data protection and confidentiality:** These functions extend to all level of the architecture.
- **Internet protocol protection:** Protection of data in motion from eavesdropping and snooping is essential between all levels.



**Figure 13.13 Secure IoT Framework**



# Secure IoT Framework

[FRAH15] also proposes a secure IoT framework that defines the components of a security facility for an IoT that encompasses all the levels, as shown in Figure 13.13. The four components are:

- **Authentication:** Encompasses the elements that initiate the determination of access by first identifying the IoT devices. In contrast to typical enterprise network devices, which may be identified by a human credential (e.g., username and password or token), the IoT endpoints must be fingerprinted by means that do not require human interaction. Such identifiers include RFID, x.509 certificates, or the MAC address of the endpoint.

# Secure IoT Framework

- **Authorization:** Controls a device's access throughout the network fabric. This element encompasses access control. Together with the authentication layer, it establishes the necessary parameters to enable the exchange of information between devices and between devices and application platforms and enables IoT-related services to be performed.
- **Network enforced policy:** Encompasses all elements that route and transport endpoint traffic securely over the infrastructure, whether control, management, or actual data traffic.

# Secure IoT Framework

- **Secure analytics, including visibility and control:**

This component includes all the functions required for central management of IoT devices. This involves, firstly, visibility of IoT devices, which simply means that central management services are securely aware of the distributed IoT device collection, including identity and attributes of each device. Building on this visibility is the ability to exert control, including configuration, patch updates, and threat countermeasures.

# Importance of Trust in Secure IoT Framework

An important concept related to this framework is that of trust relationship. In this context, trust relationship refers to the ability of the two partners to an exchange to have confidence in the identity and access rights of the other.

The authentication component of the trust framework provides a basic level of trust, which is expanded with the authorization component.

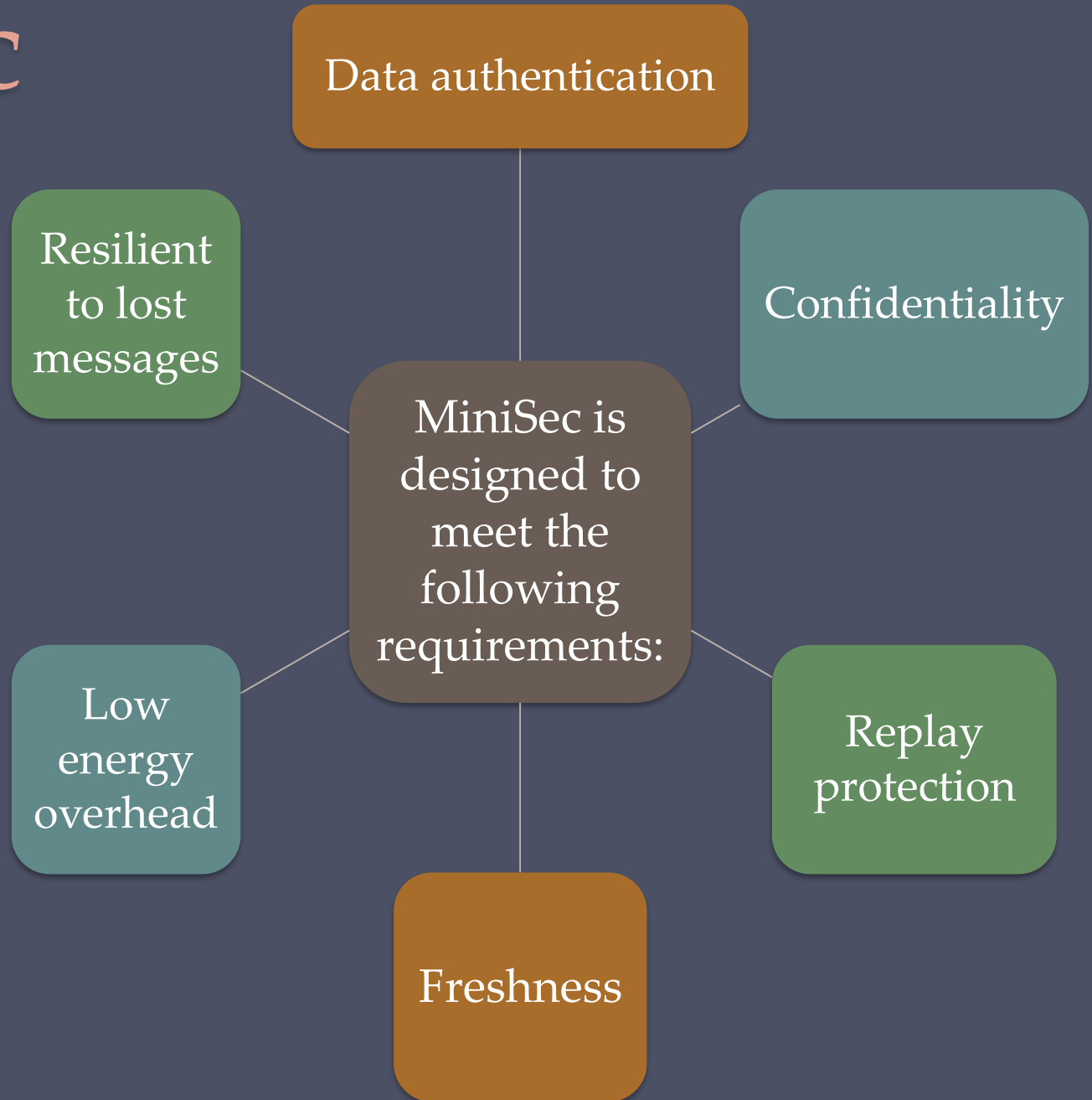
[FRAH15] gives the example that a car may establish a trust relationship with another car from the same vendor. That trust relationship, however, may only allow cars to exchange their safety capabilities.

When a trusted relationship is established between the same car and its dealer's network, the car may be allowed to share additional information such as its odometer reading and last maintenance record.

# MiniSec

- MiniSec is an open-source security module that is part of the TinyOS operating system
- It is designed to be a link-level module that offers a high level of security, while simultaneously keeping energy consumption low and using very little memory
- MiniSec provides confidentiality, authentication, and replay protection
- MiniSec has two operating modes, one tailored for single-source communication, and another tailored for multi-source broadcast communication

# MiniSec



# MiniSec

MiniSec is designed to meet the following requirements:

- **Data authentication:** Enables a legitimate node to verify whether a message originated from another legitimate node (i.e., a node with which it shares a secret key) and was unchanged during transmission.
- **Confidentiality:** A basic requirement for any secure communications system.
- **Replay protection:** Prevents an attacker from successfully recording a packet and replaying it at a later time.

# MiniSec

- **Freshness:** Because sensor nodes often stream time-varying measurements, providing guarantee of message freshness is an important property.

There are two types of freshness: Strong and weak.

MiniSec provides a mechanism to guarantee weak freshness, where a receiver can determine a partial ordering over received messages without a local reference time point.

- **Low energy overhead:** This is achieved by minimizing communication overhead and by using only symmetric encryption.

- **Resilient to lost messages:** The relatively high occurrence of dropped packets in wireless sensor networks requires a design that can tolerate high message loss rates.