Cloud Computing Module-19

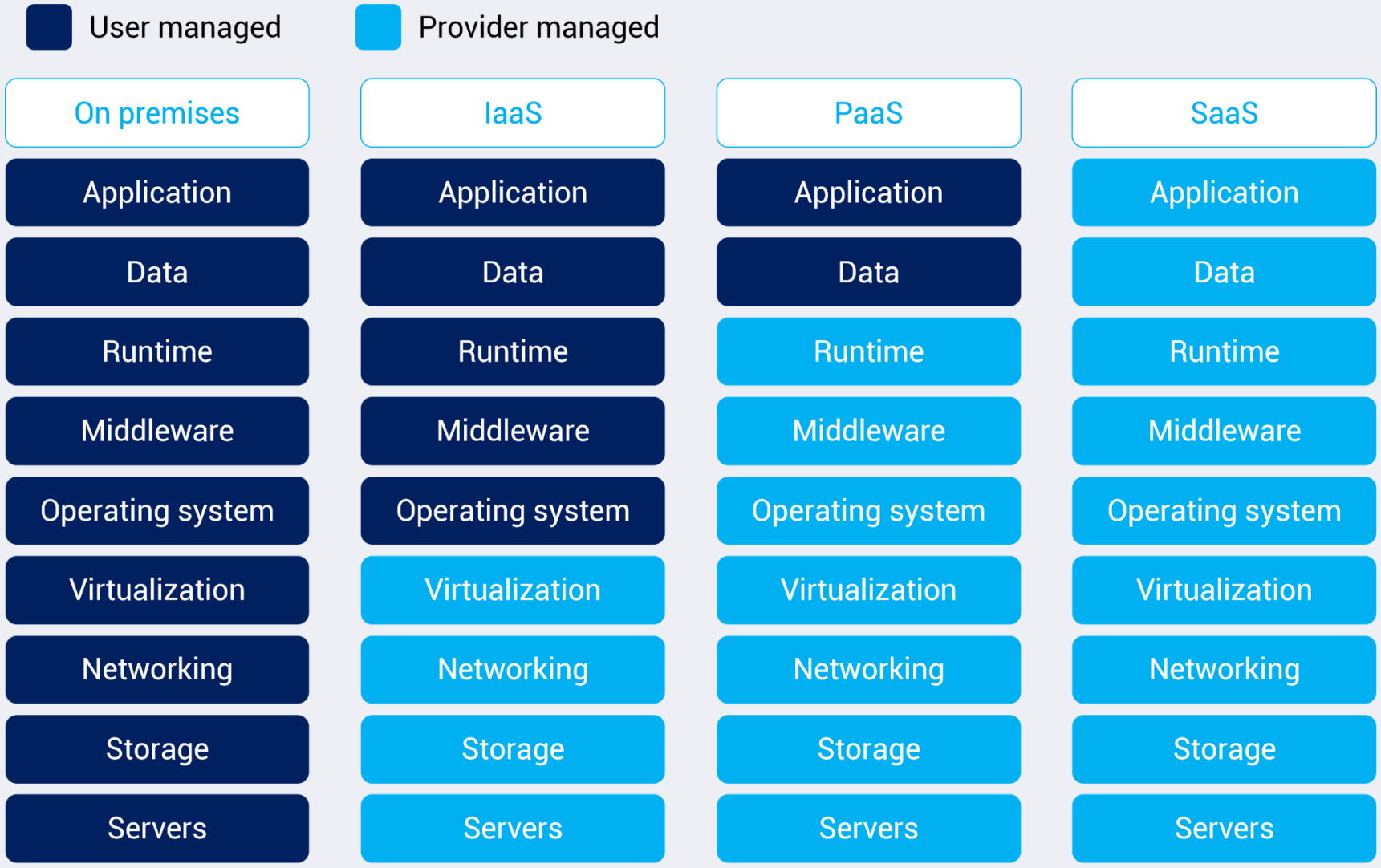
Cloud Computing

Cloud computing refers to the delivery of computing services, including storage, processing power, and software applications, over the internet. Instead of relying on local servers or personal devices to handle applications, users can access these services remotely through the internet. Cloud computing eliminates the need for organizations and individuals to invest in and maintain physical infrastructure, allowing them to leverage resources on a pay-as-you-go basis.

Three Types of Service Models:

1. **Infrastructure as a Service (IaaS)** Provides virtualized computing resources. Third-party hosts servers with hypervisors running virtual machines (VMs). Subscribers usually pay on a per-use basis. Examples: AWS, Microsoft Azure, Digital Ocean, Google Cloud.
2. **Platform as a Service (PaaS)** Geared towards software development. Hardware and software hosted by the provider. Enables development without worrying about hardware or software. Examples: Heroku, SalesForce.
3. **Software as a Service (SaaS)** Providers supply on-demand applications to subscribers. Offloads the need for patch management, compatibility, and version control. Examples: Microsoft Office 365, Dropbox storage, Google Docs.

Tech stack	Type
Software	SaaS
Apps	PaaS
OS	IaaS
Virtualization	managed by provider
Storage/Networking	managed by provider



Cloud Deployment Models

- **Private Cloud** - Cloud solely for use by one tenant; usually done in larger organizations.
- **Community Cloud** - Is make up of infrastructure from several different entitites wich may be cloud providers, business partners, and so on. (members only type of thing)
- **Public Cloud** - Services provided over a network that is open for public to use; Amazon S3, Microsoft Azure - Open for business.
- **Hybrid Cloud** - A composition of two or more cloud deployment models.

NIST Cloud Architecture

The NIST cloud computing reference architecture (NIST SP 500-292) define five major actors; Each actor is an entity (a person or an organization) that participates in a transaction or process and/or perform tasks in cloud computing.

- **Cloud Consumer** - A person or org. that maintains a business relationship with, and use servies from Cloud Providers; aquires and uses cloud products and services.
- **Cloud Provider** - A person, org. or entity responsible for making a service available; Purveyor of products and services.
- **Cloud Auditor** - Independent assor of cloud service an security controls.
- **Cloud Broker** - Manages use, performance and delivery of services as well as relationships between Cloud Providers to Cloud consumers.
- **Cloud Carrier** - Organization with responsibility of transferring data; Intermediary that provides connectivity and transport of Cloud services from Cloud providers to Cloud consumers. (e.g: Telecom's)

characteristics of cloud computing

The National Institute of Standards and Technology (NIST) *defines cloud computing as it is known today through five particular characteristics.*

1. **On-demand self-service**
2. **Broad network access**
3. **Multi-tenancy and resource pooling**
4. **Rapid elasticity and scalability**
5. **Measured service**

Threats:

- **Data Breach or Loss** - Biggest threat; includes malicious theft, erasure or modification
- **Shadow IT** - IT systems or solutions that are developed to handle an issue but aren't taken through proper approval chain
- **Abuse of Cloud Resources** - Another high threat (usually applies to Iaas and PaaS)
- **Insecure Interfaces and APIs** - Cloud services can't function without them, but need to make sure they are secure
- **Service Oriented Architecture** - API that makes it easier for application components to cooperate and exchange information
- **Insufficient due diligence** - Moving an application without knowing the security differences
- **Shared technology issues** - Multitenant environments that don't provide proper isolation
- **Unknown risk profiles** - Subscribers simply don't know what security provisions are made int he background
- **Wrapping Attack** - SOAP message intercepted and data in envelope is changed and sent/replayed
- **Session riding** - CSRF under a different name; deals with cloud services instead of traditional data centers
- **Others include malicious insiders, inadequate design and DDoS**
- Other threats:
  - Loss/compromise of encryption keys
  - Isolation failure
  - Compliance risk
  - VM vulnerabilities
  - Vendor lock-on
  - Jurisdictional issues based on chaning geographic boundaries
  - E-discovery/subpoena
  - Cloud service termination/failure
  - Improper/incomplete data handling & disposal
  - Management network failure/interface compromise

Attacks:

1. Service hijacking via Social engineering & network sniffing
2. Session hijacking using XSS
3. DNS attacks
4. Side channel attacks - (e.g.: Using an existing VM on the same physical host to attack another)
5. Cross VM attacks
6. SQL injection
7. Cryptanalysis attacks
8. Wrapping attacks - performed during the translation of SOAP messages in the TLS layer; attackers duplicate the body of the message and send it to the targeted server impersonating the legitimate user.
9. DoS/DDoS attack
10. Main-in-the-Cloud attacks - abuse of cloud file synchronization services br tracking the user into installing malicious software that places the attacker's synchronization token for the service ton their machine, allowing the attacker to steal the user's token and gain access to their files.

OWASP Top 10 Application Security Risks

1. **Injection** - Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.
    - Input validation
    - Limit account privileges
2. **Broken Authentication** - Application functions related to authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities temporarily or permanently.
3. **Sensitive Data Exposure** - Many web applications and APIs do not properly protect sensitive data, such as financial, healthcare, and PII. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes. Sensitive data may be compromised without extra protection, such as encryption at rest or in transit, and requires special precautions when exchanged with the browser.
4. **XML External Entities (XXE)** - Many older or poorly configured XML processors evaluate external entity references within XML documents. External entities can be used to disclose internal files using the file URI handler, internal file shares, internal port scanning, remote code execution, and denial of service attacks.
    - If your application uses SAML for identify processing with federated security or Single Sing on (SSO). SAML uses XML.
    - If applications accepts XML directly or XML uploads from unstrusted sources, or inserts untrusted data into XML documents.
    - Any of XML processors in the application or SOAP based web services that have (DTDs) enabled.
5. **Broken Access Control** - Restrictions on what authenticated users are allowed to do are often not properly enforced. Attackers can exploit these flaws to access unauthorized functionality and/or data, such as access other users' accounts, view sensitive files, modify other users' data, change access rights, etc.
6. **Security Misconfiguration** - is the most commonly seen issue. This is commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information. Not only must all operating systems, frameworks, libraries, and applications be securely configured, but they must be patched/upgraded in a timely fashion.
7. **Cross-Site Scripting XSS** - occur whenever an application includes untrusted data in a new web page without proper validation or escaping, or updates an existing web page with user-supplied data using a browser API that can create HTML or JavaScript. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.
    - Reflected XSS
    - Stored XSS
    - DOM XSS
8. **Insecure Deserialization** - often leads to remote code execution. Even if deserialization flaws do not result in remote code execution, they can be used to perform attacks, including replay attacks, injection attacks, and privilege escalation attacks.
9. **Using Components with Known Vulnerabilities** - Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.
10. **Insufficient Logging & Monitoring** - Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data. Most breach studies show time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring.

Additional Attacks

1. **Directory Traversal** (../) - An attacker can get sensitive information like the contents of the /etc/passwd file that contains a list of users on the server; Log files, source code, access.log and so on
2. **Cross-site Request Forgery (CSRF)** - Forces an end user to execute unwanted actions on an app they're already authenticated on
    - Inherits identity and privileges of victim to perform an undesired function on victim's behalf
    - Captures the session and sends a request based off the logged in user's credentials
    - Can be mitigated by sending **random challenge tokens**

Cloud Security Control Layers

Problem with cloud security is what you are allowed to test and what should you test; Another concern is with a hypervisor, if the hypervisor is compromised, all hosts on that hypervisor are as well.

1. **Applications** - SDCL (Software development cycle), WAF (web application firewall)
2. **Information** - DLP, encryption
3. **Management** - GRC, IAM , Patch & Configuration
4. **Network** - NIDS/NIPS, DNSSEC, QoS
5. **Trusted Computing Model** - attempts to resolve computer security problems through hardware enhancements

- **Roots of Trust (RoT)** - set of functions within TCM that are always trusted by the OS

1. **Computer & Network Storage** - Encryption, Host-based firewall, HIDS/HIPS
2. **Physical** - Guards, Gates, Fences etc.

Tools

- **CloudInspect** - pen-testing application for AWS EC2 users
- **CloudPassage Halo** - instant visibility and continuous protection for servers in any cloud
- **Dell Cloud Manager**
- **Qualys Cloud Suite**
- **Trend Micro's Instant-On Cloud Security**
- **Panda Cloud Office Protection**