STUDENT HANDOUT · WEEK 7

Security Implications of Architecture Models

SY0-701 objectives 3.1. Notes, exercises, review questions, and the lab.

Name	
Date	

THIS WEEK YOU WILL

1. Assign security duties in IaaS, PaaS, and SaaS using a responsibility matrix.
2. Explain the security implications of hybrid cloud and third-party vendors.
3. Compare IaC, serverless, microservices, containers, and virtualization for security.
4. Distinguish air-gapped, logically segmented, and software-defined networks.
5. Describe the risks of IoT, ICS and SCADA, RTOS, and embedded systems.
6. Weigh an architecture against availability, cost, patching, power, and other factors.
7. Calculate yearly downtime from an availability percentage.

KEY IDEAS

The responsibility matrix moves with the model

Layer	IaaS	PaaS	SaaS
Data, users, and access	Customer	Customer	Customer
Application	Customer	Customer	Provider (customer configures)
OS, runtime, middleware	Customer	Provider	Provider
Virtualization, hardware	Provider	Provider	Provider
Physical data center	Provider	Provider	Provider

Hybrid setups and vendors add seams

Hybrid considerations	Third-party vendors
Two sets of tools, logs, and identities to align	Their security becomes part of yours
Data moving between sites must be encrypted	Contracts set duties: SLA, breach notice, audits
One policy applied on premises and in the cloud	Ask for assurance such as a SOC 2 Type II report
Network links and latency between environments	Plan for the vendor failing or leaving

Code, functions, services, and containers

Infrastructure as code	Servers and networks defined in versioned files. (Terraform, CloudFormation)
Serverless	The provider runs your functions; no servers to patch. (AWS Lambda, Azure Functions)
Microservices	Small services talk over APIs; each scales alone. (Separate billing service)
Containers	Apps packaged in isolation, sharing the host kernel. (Docker, Kubernetes)
Virtualization	Each VM runs its own OS on a shared hypervisor. (Hyper-V, VMware ESXi)

Separate networks physically or logically

Approach	What it is	Tradeoff
Air-gapped	No network path to other networks at all	Strongest, but updates move by hand
Physical isolation	Separate switches and cabling per zone	Costly, but no shared gear to misconfigure
Logical segmentation	VLANs, subnets, and ACLs on shared gear	Cheap and flexible; one bad rule opens it
Software-defined (SDN)	Control plane split from the data plane	Central, automated control; guard the controller

Centralized and decentralized control

Centralized	Decentralized
One place for policy, logs, and identity	Control spread across sites or teams
Easier to audit and keep consistent	No single point of failure
A single point of failure and a rich target	Harder to keep policy and patching consistent

Specialized systems are hard to patch

IoT	Connected devices with weak defaults and few updates. (Smart thermostats, IP cameras)
ICS and SCADA	Control physical processes; safety and uptime come first. (Water treatment, power grid)
RTOS	An OS that must respond within strict time limits. (Infusion pumps, vehicle brakes)
Embedded systems	A computer built into a device for one job, often unpatchable. (Badge readers, medical carts)

Every architecture is a set of tradeoffs

Consideration	Question to ask
Availability and resilience	Does it stay up, and recover when parts fail?
Cost and risk transference	What does it cost, and who carries the loss?
Responsiveness, scalability	Is it fast enough, and can it grow with demand?
Deployment and recovery	How easily can we build it, and rebuild it?
Patch availability	Will patches come, and can we install them?
Power and compute	Does the device have power and CPU for security?

Each extra nine cuts downtime tenfold

8.76 h	Yearly downtime at 99.9%
52.6 min	Yearly downtime at 99.99%
5.26 min	Yearly downtime at 99.999%
	A year has 8,760 hours. Allowed downtime = 8,760 hours x (1 minus the availability).

EXERCISE 1: PICK THE ARCHITECTURE THAT FITS

8 MINUTES · PAIRS

For each need, choose the architecture model from 3.1 that fits best and name one security concern it brings.

#	Prompt	Your answer
1	A pharmacy app whose traffic spikes only on flu shot days	
2	A water utility control network that must never reach the internet	
3	A school that wants every lab server built identically from a file	
4	A bank that wants to update each payment feature on its own	
5	A clinic that wants email without running any mail servers	
6	A startup that wants apps to run the same on laptops and cloud	
7	A university that wants one rule pushed to 200 switches at once	

EXERCISE 2: SECURE A HOSPITAL'S SPECIALIZED SYSTEMS

7 MINUTES · SMALL GROUPS

For each system, name its type, the consideration that makes it hard to secure, and one compensating control.

#	Prompt	Your answer
1	Infusion pumps on Wi-Fi whose vendor ships updates once a year	
2	HVAC controllers that anyone on the staff network can reach	
3	Security cameras still using the factory admin password	
4	A pharmacy robot on an RTOS that cannot run an EDR agent	
5	Battery-powered door sensors too weak to run strong encryption	
6	A SCADA system that runs the hospital's backup generators	

REVIEW QUESTIONS

1. In a SaaS service, which security responsibilities always stay with the customer?

2. Why do containers provide weaker isolation than virtual machines?

3. How does an air-gapped network differ from logical segmentation?

4. What does inability to patch mean, and how should you respond to it?

5. About how much downtime per year does 99.9% availability allow?

KEY TAKEAWAYS

- **Data and access stay yours.** In IaaS, PaaS, and SaaS the customer owns data, accounts, and settings.
- **New models move the surface.** IaC, serverless, microservices, and containers each shift the risks.
- **Guard what you cannot patch.** Isolate IoT, ICS, RTOS, and embedded devices and watch their traffic.
- **Every model is a tradeoff.** Weigh availability, cost, recovery, patching, power, and compute.

LAB 7: CASE STUDY ON SHARED RESPONSIBILITY IN A CLOUD MIGRATION

60 MINUTES

This is a written exercise; no tools or network access are needed. Work alone or in pairs as your instructor directs, using the responsibility matrix and considerations from this week's lesson.

Steps

1. Read the scenario: Lakeside Family Clinic will move patient records to a SaaS electronic health record (EHR) system, move email to a SaaS suite, and replace the closet file server with two IaaS virtual machines at a public cloud provider. The front desk PCs, the Wi-Fi, and the ultrasound cart stay on premises.
2. Build a responsibility matrix: rows for physical security, hardware, hypervisor, operating system, application, network controls, identities and access, data, and backups; columns for the EHR, email, and the IaaS file server. Mark each cell Provider, Clinic, or Shared.
3. List every cell marked Clinic or Shared that the clinic does not handle well today, such as patching the IaaS VMs or enabling MFA in the SaaS tools. These are the responsibility gaps.
4. Write the vendor requirements the clinic must have in writing before it moves patient data: a HIPAA business associate agreement, breach notification terms, an uptime SLA, an assurance report such as SOC 2 Type II, and how the data is returned if the contract ends.
5. Describe the hybrid considerations: how staff will sign in to on-premises and cloud systems with one identity and MFA, how traffic between the clinic and the IaaS VMs will be protected, and where logs from both sides will be collected.
6. Rate the move against five considerations from objective 3.1, such as availability, cost, ease of recovery, patch availability, and risk transference, as better, worse, or unchanged, with one sentence of reasoning each.
7. Identify what the move does not fix, such as the ultrasound cart that cannot be patched and the front desk PCs, and say how each should be handled.
8. Write a one-page memo to the clinic's office manager summarizing the three most important things the clinic must still do itself after the move, in plain language.

Hand in

One document with your responsibility matrix, gap list, vendor requirements, hybrid notes, considerations rating, and the one-page memo. Keep it: it informs the control recommendations in project milestone 2.

Questions to answer in your report

1. Which responsibilities stay with the clinic in all three services, regardless of the cloud model?
2. Which of the clinic's current risks does the move actually transfer to providers, and which does it only appear to transfer?
3. What is the biggest new risk the move creates, and how would you mitigate it?

4. If the EHR provider has a day-long outage, what should the contract say, and what must the clinic plan for on its own?