

Cloud Architecture

WELL-ARCHITECTED FRAMEWORK · SIX Pillars · DESIGN PRINCIPLES · RELIABILITY · HIGH AVAILABILITY · STUDY & REVIEW
TRUSTED ADVISOR

01 MUST KNOW

if you remember five things, remember these

- 01 Well-Architected Framework:** a guide for designing infrastructures that are **secure, high-performing, resilient, and efficient** — foundational questions + best practices, distilled from reviewing **thousands of customer architectures**. It gives a *consistent approach* to evaluating and implementing cloud architectures.
- 02 Six pillars: operational excellence · security · reliability · performance efficiency · cost optimization · sustainability.** The first five date to the framework's 2015 introduction; sustainability was added in **2021**. Each pillar = design principles + best practice areas, and each area aligns to reviewer questions.
- 03 Reliability ≠ availability:** **reliability** is the *probability* the entire system (hardware, firmware, software) functions as intended for a specified period — measured by **MTBF = total time in service ÷ failures**. **Availability** is *normal operation time ÷ total time* — a percentage of uptime, counted in 9s (five 9s = 99.999%).
- 04 “Everything fails, all the time”** (Werner Vogels, Amazon CTO) — so architect to withstand failure. Three factors determine availability: **fault tolerance** (built-in redundancy), **scalability** (grow without redesign), **recoverability** (restore service after catastrophe). More availability usually means more **cost**.
- 05 AWS Trusted Advisor:** online tool giving **real-time** guidance on your entire AWS environment against best practices, in **five categories**: cost optimization, performance, security, fault tolerance, service limits.

02 KEY TERMS

TERM	DEFINITION
Well-Architected Framework	Consistent approach + best practices for evaluating and implementing cloud architectures; organized into six pillars
Pillar	One of six lenses on an architecture; each has its own design principles and best practice areas
Best practice area	Subdivision of a pillar; aligns to the questions a reviewer asks (e.g. SEC 1...), collected in the framework Appendix
AWS Well-Architected Tool	Console tool: define a workload, answer the framework's questions, receive a step-by-step action plan
Reliability	Probability the entire system functions as intended for a specified period; think statistically
MTBF	Mean time between failures = total time in service ÷ number of failures = MTTF + MTTR
Availability	Normal operation time ÷ total time, as a % of uptime (commonly over 1 year); reduced by scheduled and unscheduled interruptions
High availability (HA)	System withstands some degradation while remaining available; minimal downtime, minimal human intervention
Fault tolerance	Built-in redundancy of an application's components — it stays operational when components fail
Scalability	Ability to accommodate increased capacity needs without changing design; contributes to (doesn't guarantee) availability
Recoverability	Process, policies, and procedures to restore service after a catastrophic event, without lost data
AWS Trusted Advisor	Online tool: real-time recommendations on your whole environment in five best-practice categories

03 THE SIX PILLARS

know each pillar's one-line focus

PILLAR	FOCUS	KEY TOPICS
Operational excellence	Run and monitor systems to deliver business value; continually improve processes and procedures	Automating changes · responding to events · standards for daily operations
Security	Protect information, systems, and assets while delivering value through risk assessment and mitigation	Data confidentiality/integrity · privilege management · protecting systems · detecting security events
Reliability	Workload performs its intended function correctly and consistently when expected	Distributed system design · recovery planning · handling change
Performance efficiency	Use IT and computing resources efficiently as demand changes and technologies evolve	Right resource types and sizes · monitoring performance · informed decisions
Cost optimization	Avoid unnecessary costs	Where money is spent · right type and number of resources · analyzing spend · scaling without overspending
Sustainability	Minimize the environmental impacts of running cloud workloads	Added 2021 as the sixth pillar (the other five date to 2015)

04 DESIGN PRINCIPLES BY PILLAR

count them: 5 · 7 · 5 · 5 · 5

PILLAR	#	DESIGN PRINCIPLES
Operational excellence	5	Perform operations as code · make frequent, small, reversible changes · refine operations procedures frequently · anticipate failure · learn from all operational failures
Security	7	Implement a strong identity foundation · enable traceability · apply security at all layers · automate security best practices · protect data in transit and at rest · keep people away from data · prepare for security events
Reliability	5	Automatically recover from failure · test recovery procedures · scale horizontally to increase aggregate availability · stop guessing capacity · manage change in automation
Performance efficiency	5	Democratize advanced technologies · go global in minutes · use serverless architectures · experiment more often · consider mechanical sympathy
Cost optimization	5	Implement Cloud Financial Management · adopt a consumption model · measure overall efficiency · stop spending on undifferentiated heavy lifting · analyze and attribute expenditure

05 RELIABILITY VS. AVAILABILITY

RELIABILITY <i>does it function as intended?</i> — Probability the entire system works as intended for a specified period — System = hardware + firmware + software; a failed component makes the system unavailable — Measured statistically: MTBF = total time in service ÷ failures — “Everything fails, all the time” — plan for it	AVAILABILITY / HA <i>what share of the time is it up?</i> — Normal operation time ÷ total time — % uptime over e.g. 1 year — Counted in 9s: five 9s = 99.999% — Reduced by scheduled <i>and</i> unscheduled interruptions — HA: withstands degradation, downtime minimized, minimal human intervention, restore often < 1 min
---	---

06 RELIABILITY MATH

app up Mon noon → fails Fri noon → repaired by Mon noon, weekly

METRIC	MEANING	EXAMPLE
MTTF	Mean time to failure — average time running before a failure	96 h (Mon noon → Fri noon)

METRIC	MEANING	EXAMPLE
MTTR	Mean time to repair — average time to diagnose and fix	72 h (Fri noon → Mon noon)
MTBF	Mean time between failures = $MTTF + MTTR$ = total time in service ÷ failures	$96 + 72 = 168$ h (1 week)
Availability	Normal operation time ÷ total time	$96 \div 168 \approx 57\%$

07 TRUSTED ADVISOR'S FIVE CATEGORIES

real-time checks on your whole environment

CATEGORY	WHAT IT CHECKS
Cost Optimization	Unused and idle resources to eliminate; opportunities to commit to reserved capacity
Performance	Service limits, use of provisioned throughput, overutilized instances
Security	Security gaps, AWS security features left off, permissions
Fault Tolerance	Automatic scaling, health checks, Multi-AZ deployments, backup capabilities
Service Limits	Usage above 80% of a service limit — snapshot-based; data can lag up to 24 h

08 EXAM TRAPS

where the knowledge check gets you

- ✗ “The Well-Architected Framework has five pillars.” — **Six** since 2021: sustainability joined the original five from 2015.
- ✗ “Reliability and availability are the same thing.” — Reliability is the **probability** of functioning as intended for a specified period (MTBF); availability is the **share of time** the system operates normally (the 9s).
- ✗ “MTBF is the average running time before a failure.” — That's **MTTF**. $MTBF = MTTF + MTTR$ (total time in service ÷ number of failures).
- ✗ “Fault tolerance protects against software failures.” — The fault-tolerant model uses redundant **hardware**; it does **not** address software failures — the most common cause of downtime.
- ✗ “Scalability guarantees availability.” — It only **contributes** to availability; it doesn't guarantee it.
- ✗ “Maximize availability, always.” — Improving availability usually **raises cost**; balance the improvement against the benefit to your users.
- ✗ “Trusted Advisor and the Well-Architected Tool do the same review.” — The **WA Tool** reviews one *workload* against the framework's questions and delivers an action plan; **Trusted Advisor** watches your live *environment* in real time across five categories.
- ✗ “Trusted Advisor warns when you reach 100% of a service limit.” — It flags usage over **80%** of the limit, based on a snapshot that can be up to 24 hours old.

- Q1** Name the six pillars of the AWS Well-Architected Framework.
- Q2** Which pillar was added to the framework in 2021?
- Q3** “Stop guessing capacity” is a design principle of which pillar? And “go global in minutes”?
- Q4** Who said “Everything fails, all the time”?
- Q5** Give the formula for MTBF.
- Q6** What availability percentage does “five 9s” denote?
- Q7** Name the three factors that determine an application's overall availability.
- Q8** List AWS Trusted Advisor's five recommendation categories.
- Q9** Trusted Advisor flags a security group rule with source /0 on port 3306. What's the recommended fix?
- Q10** Sample CCP exam: a SysOps engineer wants to protect data in transit and at rest. Which could they use — Elastic Load Balancing, Amazon EBS, Amazon S3, or all of the above?

ANSWER KEY (rotate the page)

Q1 operational excellence, security, reliability, performance efficiency, cost optimization, sustainability, **Q2** sustainability, **Q3** reliability, performance efficiency **Q4** Werner Vogels, CTO of Amazon.com **Q5** total time in service ÷ number of failures (= MTTF + MTTR) **Q6** 99.999% performance efficiency **Q7** fault tolerance, scalability, recoverability **Q8** cost optimization, performance, security, fault tolerance, service limits **Q9** restrict access to only the IPs that need it (e.g. /32), then delete the permissive rule **Q10** all of the above