Failure Modes

We will be looking at potential Failure Modes, their potential effects, causes and the appropriate design controls. Credits to https://github.com/adrianco for providing helpful resources.

Item / Function	Potential Failure Mode(s)	Potential Effect(s) of Failure	Potential Cause(s)/ Mechanism(s) of Failure	Design Controls	Recommended Action(s)
Authentication	Client can't authenticate	Can't connect application	Certificate timeout, version mismatch, account not setup, credential changed	Log and alert on authentication failures	
	Slow or unreliable authentication	Slow start for application	Auth service overloaded, high error and retry rate	Log and alert on high authentication latency and errors	
Client Request to API Endpoint	Service unknown, address un-resolvable	Delay while discovery or DNS times out, slow fallback response	DNS configuration error, denial of service attack, or provider failure	Customer eventually complains via call center	Dual redundant DNS, fallback to local cache, hardcoded IP addresses. Endpoint monitoring and alerts
	Service unreachable, request undeliverable	Fast fail, no response	Network route down or no service instances running	Autoscaler maintains a number of healthy instances	Endpoint monitoring and alerts
	Service reachable, request undeliverable	Connect timeout, slow fail, no response	Service frozen/not accepting connection	Retry request on different instance. Healthcheck failure instances removed. Log and alert.	
	Request delivered, no response - stall	Application request timeout, slow fail, no response	Broken service code, overloaded CPU or slow dependencies	Retry request on different instance. Healthcheck failure instances removed. Log and alert.	
	Response undeliverable	Application request timeout, slow fail, no response	Network return route failure, dropped packets	Retry request on different instance. Healthcheck failure instances removed. Log and alert.	
	Response received in time but empty or unintelligible	Fast fail, no response	Version mismatch or exception in service code	Retry request on different instance. Healthcheck failure instances removed. Log and alert.	

	Request delivered, response delayed beyond spec	Degraded response arrives too late, slow fallback response	Service overloaded or GC hit, dependent services responding slowly	Retry request on different instance. Healthcheck failure instances removed. Log and alert.	
	Request delivered, degraded response delivered in time	Degraded timely response	Service overloaded or GC hit, dependent services responding slowly	Log and alert on high service latency and errors	
			*GC – Garbage Collector		
Time Bombs	Internal application counter wraparound				Test long running operations of code base
	Memory leak				Monitor process sizes and garbage collection intervals over time
Content Bombs	Incoming data that crashes the app				Fuzz the input with generated random and structured data to show it doesn't crash.
Configuration Errors	Configuration file syntax errors or incorrect values				Canary test deployments incrementally. Chaos testing.
Versioning Errors	Incompatible interface versions				Canary test deployments incrementally
Retry Storms	Too many retries, too large timeout values				Chaos testing applications under stress
Excessive Logging	Cascading overload				Chaos testing applications under stress