Fault Tolerance

What is fault tolerance

Fault tolerance refers to the ability of a system (computer, network, cloud cluster, etc.) to continue operating without interruption when one or more of its components fail.

The objective of creating a fault-tolerant system is to prevent disruptions arising from a single point of failure, ensuring the high availability and business continuity of mission-critical applications or systems.

Fault-tolerant systems use backup components that automatically take the place of failed components, ensuring no loss of service. These include:

- **Hardware systems** that are backed up by identical or equivalent systems. For example, a server can be made fault tolerant by using an identical server running in parallel, with all operations mirrored to the backup server.
- **Software systems** that are backed up by other software instances. For example, a database with customer information can be continuously replicated to another machine. If the primary database goes down, operations can be automatically redirected to the second database.
- **Power sources** that are made fault tolerant using alternative sources. For example, many organizations have power generators that can take over in case main line electricity fails.

In similar fashion, any system or component which is a single point of failure can be made fault tolerant using redundancy.

Fault tolerance can play a role in a disaster recovery strategy. For example, fault-tolerant systems with backup components in the cloud can restore mission-critical systems quickly, even if a natural or human-induced disaster destroys on-premise IT infrastructure.

Fault tolerance vs. high availability

High availability refers to a system's ability to avoid loss of service by minimizing downtime. It's expressed in terms of a system's uptime, as a percentage of total running time. Five nines, or 99.999% uptime, is considered the "holy grail" of availability.

In most cases, a business continuity strategy will include both high availability and fault tolerance to ensure your organization maintains essential functions during minor failures, and in the event of a disaster.

While both fault tolerance and high availability refer to a system's functionality over time, there are differences that highlight their individual importance in your business continuity planning.

Consider the following analogy to better understand the difference between fault tolerance and high availability. A twin-engine airplane is a fault tolerant system – if one engine fails, the other one kicks in, allowing the plane to continue flying. Conversely, a car with a spare tire is highly available. A flat tire will cause the car to stop, but downtime is minimal because the tire can be easily replaced.

Some important considerations when creating fault tolerant and high availability systems in an organizational setting include:

- **Downtime** – A highly available system has a minimal allowed level of service interruption. For example, a system with "five nines" availability is down for approximately 5 minutes per year. A fault-tolerant system is expected to work continuously with no acceptable service interruption.
- **Scope** – High availability builds on a shared set of resources that are used jointly to manage failures and minimize downtime. Fault tolerance relies on power supply backups, as well as hardware or software that can detect failures

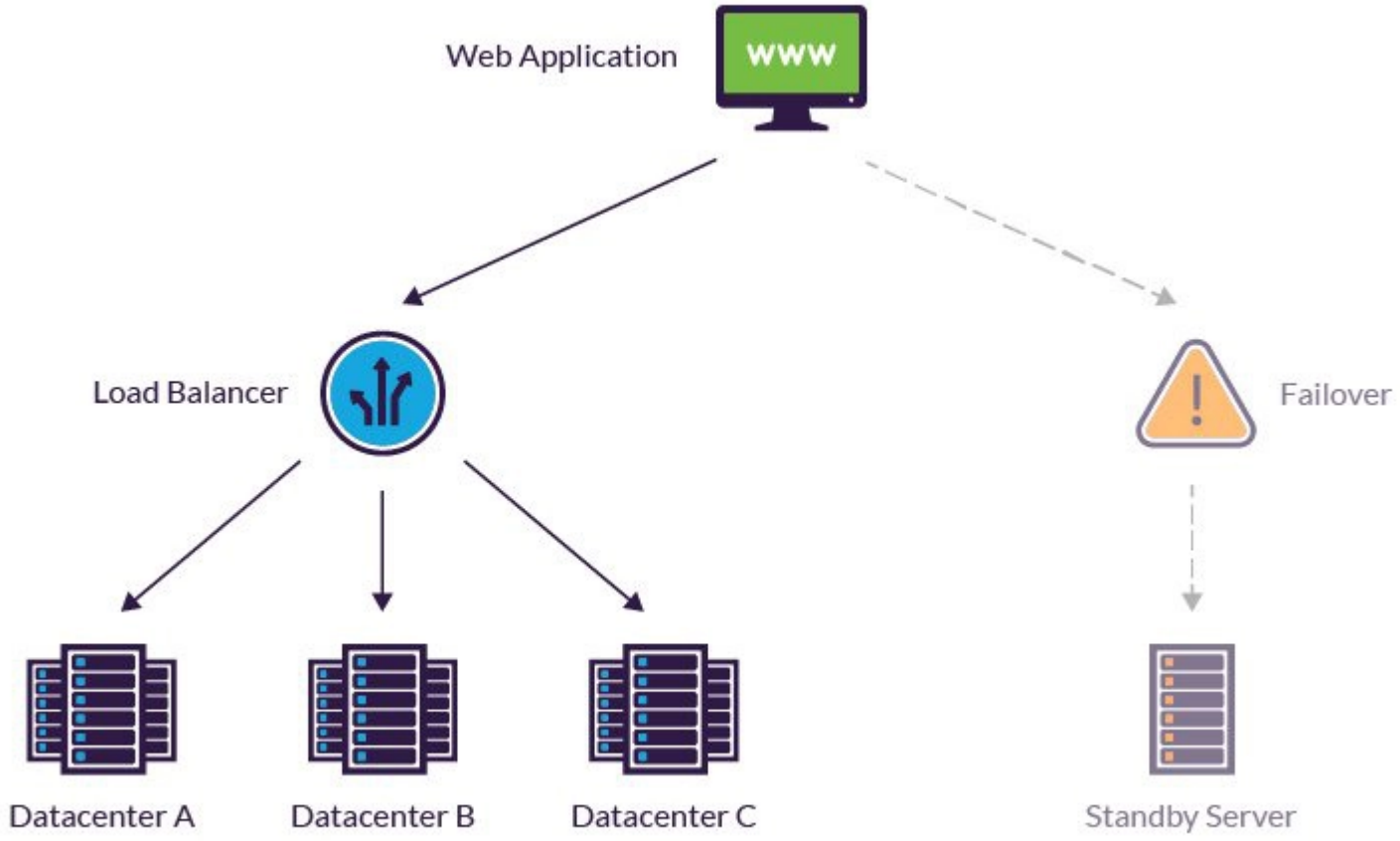
and instantly switch to redundant components.

- **Cost** – A fault tolerant system can be costly, as it requires the continuous operation and maintenance of additional, redundant components. High availability typically comes as part of an overall package through a service provider (e.g., load balancer provider).

Some of your systems may require a fault-tolerant design, while high availability might suffice for others. You should weigh each system's tolerance to service interruptions, the cost of such interruptions, existing SLA agreements with service providers and customers, as well as the cost and complexity of implementing full fault tolerance.

Load balancing and failover: fault tolerance for web applications

In the context of web application delivery, fault tolerance relates to the use of load balancing and failover solutions to ensure availability via redundancy and rapid disaster recovery.



Load balancing and failover are both integral aspects of fault tolerance.

Load balancing solutions allow an application to run on multiple network nodes, removing the concern about a single point of failure. Most load balancers also optimize workload distribution across multiple computing resources, making them individually more resilient to activity spikes that would otherwise cause slowdowns and other disruptions.

In addition, load balancing helps cope with partial network failures. For example, a system containing two production servers can use a load balancer to automatically shift workloads in the event of an individual server failure.

Failover solutions, on the other hand, are used during the most extreme scenarios that result in a complete network failure. When these occur, a failover system is charged with auto-activating a secondary (standby) platform to keep a web application running while the IT team brings the primary network back online.

For true fault tolerance with zero downtime, you need to implement “hot” failover, which transfers workloads instantly to a working backup system. If maintaining a constantly active standby system is not an option, you can use “warm” or “cold” failover, in which a backup system takes time to load and start running workloads.

Imperva load balancing and failover solutions

Imperva offers a complete suite of web application fault tolerance solutions. The first among these is our cloud-based application layer load balancer that can be used for both in-datacenter (local) and cross-datacenter (global) traffic distribution.

The solution is provided via a load balancing as a service (LBaaS) model and is delivered from a globally-distributed network of data centers for rapid response and added redundancy.

Intelligent data-driven algorithms (e.g., least pending requests) are used to track server loads in real-time for optimized traffic distribution.

The other side of the coin is our failover solution that uses automated health checks from multiple geolocations to monitor the responsiveness of your servers.

In the event of a server failure, site traffic is instantly rerouted to a backup site within seconds, ensuring uninterrupted availability. The service is delivered from the cloud. As a result, even the execution of a remote failover doesn’t suffer from any TTL-related delays commonly found in other DNS-based solutions.

For peace of mind, all Imperva Incapsula enterprise customer are also offered a 99.999% uptime SLA that reflects our confidence in the resiliency of our solution and the quality of our services.