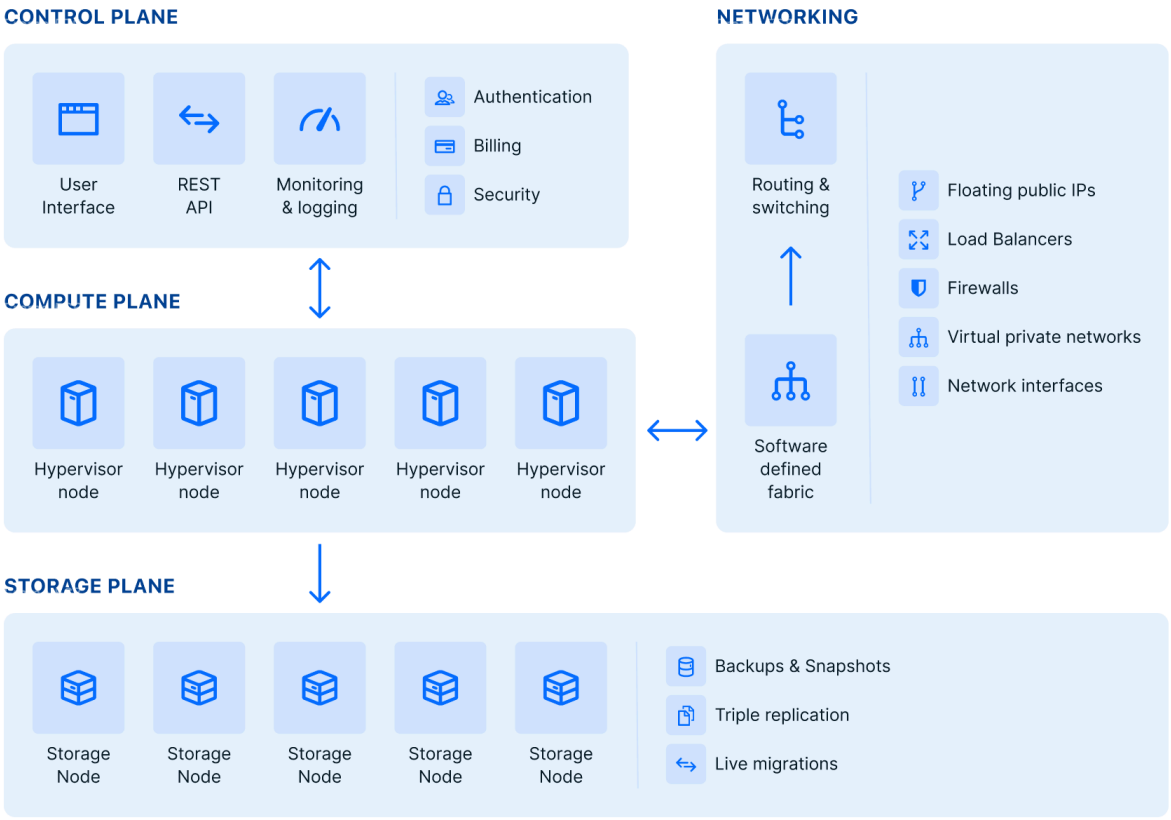
Warren Technical Stack

Warren is built from the ground up following contemporary architectural principles with no dragging legacy restrictions while making sure the platform is highly available, scalable, reliable, and easily maintainable.

Our general architectural approach to cope with the rapidly evolving technology development and market demand is the product's modular and decoupled design that allows us to maintain flexibility. To achieve service providers' reliability requirements, we select only open-source components that are proven to be enterprise-ready and do not define our roadmap or lock us into a specific paradigm. Furthermore, all architectural decisions are made, keeping in mind the scalability and hardware-agnostic approach.

Key components that pull the system together are developed in-house, including the resource manager, virtualisation operators, and the billing system. Many of the open-source components are heavily extended to meet our needs. The control plane consists of at least three nodes (for high availability) and is containerised. Components communicate with REST calls either in the same cluster or between clusters, which makes different setup types and heterogeneous clusters possible. More details can be found in the documentation.

General Architecture



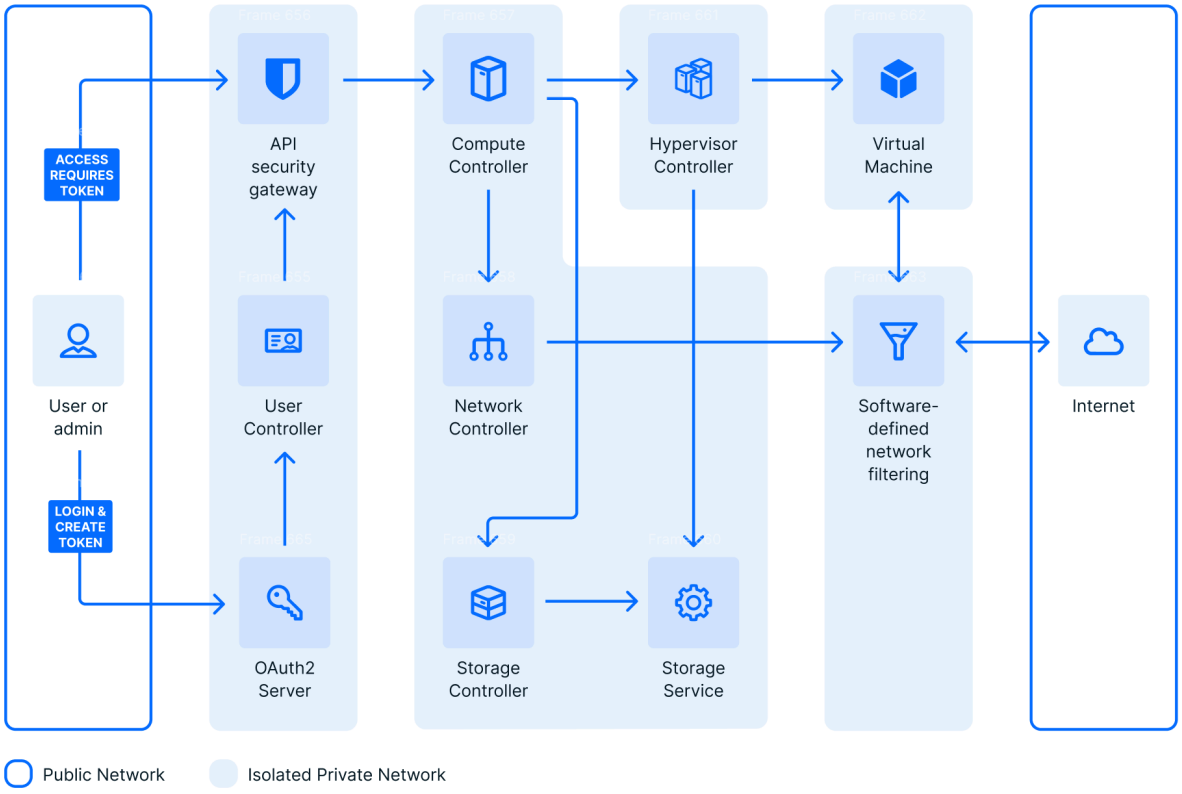
The Technology Stack Components

Technology	Developer	Function
NixOS/NixOps	Open source (heavily extended by Warren)	Configuration management system, Operating system
Terraform	HashiCorp (Open source)	Infrastructure-as-Code
Nomad	HashiCorp (Open source)	Service orchestrator
Teleport	Open source	Zero trust access control
KVM/Qemu	Open source	Hypervisor / virtual machine manager
Ceph	Open source	Distributed redundant block storage, object storage
PostgreSQL	Open source	Main database
Docker	Open source	Container build system
OpenSDN (Previously known as Tungsten Fabric)	Linux Foundation (Open source)	Software-defined networking (floating IPs, virtual private cloud, load balancing, network address translation)
Kafka	Open source	OpenSDN metrics storage
RabbitMQ	Open source	OpenSDN message broker
Cassandra	Open source	OpenSDN configuration database
Consul	Hashicorp (Open source)	Fault-tolerant configuration storage and service discovery
InfluxDB	Open source	Metrics time series storage
Telegraf	Open source	System metrics collector
Kong	Open source	API gateway
Riemann	Open source	Streaming time series aggregator
Traefik	Open source	User-facing reverse proxy for services
HAProxy	Open source	High-availability load balancing
OAuth 2	Open source	Authentication protocol
User Interface	Warren (Proprietary)	Administrative console for all cloud resources
Billing service	Warren (Proprietary)	Manages user payments and integrates with various payment gateways
Charging service	Warren (Proprietary)	Calculates invoices & reports based on resource consumption
Resource manager	Warren (Proprietary)	Controls virtual machine operations and stores state & inventory
Network service	Warren (Proprietary)	Controls network operations and configures SDN
Virtualization agents	Warren (Proprietary)	Run on hypervisors and monitor the virtual machine state

Languages, Libraries and Frameworks

Primary: Java, Python, TypeScript, C++
Secondary: Go, JavaScript, SQL, Clojure, InfluxQL, Lua, Nix Expression Language
Libraries and frameworks: CNI, Libvirt, React Aria, PhosphorJS, Spring, jOOQ, SQLAlchemy, Hug, Gunicorn

Network Isolation Boundaries



A physical network diagram is available on the Warren website.

Release Cycles

We follow an iterative development approach that enables us to release updates frequently. As our service is fully managed, we deploy these updates to all data centre partners immediately upon release, ensuring that they benefit from the latest improvements without delay. Detailed release notes are made publicly available here:

https://warrenio.atlassian.net/wiki/spaces/WARP/pages/42237961/Release+Notes