Security threat	Impact	Description	Prevention
Cluster issues / node failures	high	Reliability: If the cluster fails all of our services are unavailable	- proper **metrics and alerts** - using managed service (e.g. Google Kubernetes Engine), to **reduce configuration complexity**

single service in cluster is compromised (e.g. due to human error, like wrong implementation)		If a service was compromised because authentication/authorization failed, the attacker could gain access to all dependencies this service can communicate with.	- always apply **principle of least privilige**:
			  - internal authentication (client_credentials)
			  - db access restriction (one user per service)
			- **code review**, to mitigate implementation mistakes that can compromise application security

database access compromised, or database dropped due to human error		For some reason all protection layers were compromised and the attacker has access to the database.	- database **auditing trails** -> identify user that was used by attacker - database snapshots each day for disaster recovery - reingestion for customer facing serving database - Domain Driven Development -> database separation
DDOS attack on tracking-computation-svc		If the tracking ingestion service is overloaded, ingestion for all routers will fail.	- **IP whitelisting / VPN** - not exposing any tracking-ingestion-service endpoint to the public
customer service admin role compromised		If the attacker has unrestricted access to the customer-service he is able to remove customers or assign them to different routers.	- **IP whitelisting** for all endpoints - no access to tracking data (service only accesses customer Datastore-index) - **input validation (protect against XSS)**
SE employee could loose laptop and k8s authentication keys could get stolen		People with access to the kubernetes cluster have full access to all k8s secrets (database passwords, service-account client credentials, private network access through port-forwarding)	- restrict cluster access to specific IP addresses - restrict cluster access to authenticated users (.kube/config) - storing secrets in a external Vault (track access to this vault)
dashboard-service authentication and authorization compromised (admin account leaked)		As the dashboard-service has direct access to the serving-db, the attacker can leak this db.	- dashboard-service has read-only access to on the database - SQL injection prevention -> prepared statements - kubernetes ingress firewall -> port whitelisting
customer access token / refresh token compromised (XSS, CSRF)		Customer tokens will grant full access to the specific customers analyzing data. Sensible data could be dumped, but not modified.	- access/refresh token revokation endpoint - **Authorization Grant with PKSE**, to improve authentication for untrusted clients - **state** to mitigate CSRF - **disallow cross origin requests** - user input validation on auth-service (**protect against XSS**) - logging of requests for specific customers to identify unusual behavior
router ingestion and router manipulation	medium	As routers are placed in public, they can easily be stolen and unauthorized people could gain access to the hardware. This could enable attackers to ingest invalid data.	- authentication -> each router has a specific key that authorized for just one single store - IP whitelisting (if store with static IP) otherwise VPN - logging, metrics -> review ingested data points, alert on router downtime (reason for modification)
processing stream fails	low	For some reason the processing pipeline is down and we cannot process newly ingested data.	- data queued in Cloud Pub/Sub - ingested data accumulated in ingestion-db -> can be used for reingestion if necessary
DDOS attack on dashboard-service		If the dashboard service is unavailable nobody can access the analyzed data.	- requiring authentication for any request - ratelimiting - caching of repeating queries
user data compromised (email, password)		If user data gets leaked from our auth database, an attacker might be able to access all email addresses of our customers and maybe passwords as well	- proper password hashing - HTTPS during login flow
internal company network breached		In case an attacker gets access to the internal company network, he would breach the cluster whitelisting. He still needs authentication for cluster and it's services.	- company guest and internal network separated - DMZ for developers with dedicated public IP (cluster whitelisting for this IP) - proper firewall for incoming connections - Antivirus on all employee machines - disable Microsoft Office macros -> only use Google Docs
Any application can fail due to unexpected errors or unsufficient ressources		If an application fails it could produce a downtime.	- k8s deployment controllers will try to restart - running replicas will just redirect traffic if one pod fails - error tracking (Sentry)