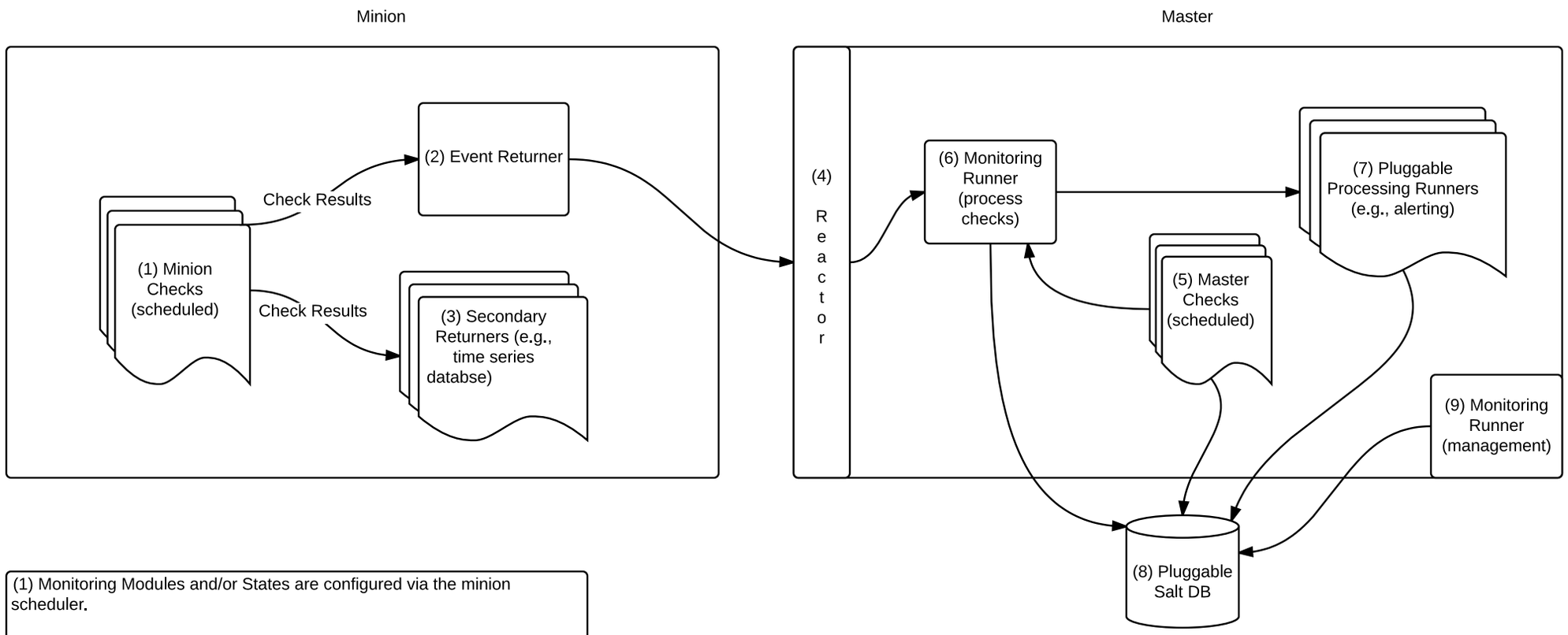

(1) Monitoring Modules and/or States are configured via the minion scheduler.

(2) Check results are returned to the monitoring event returner, which fires a master event with aggregated summary of checks.

(3) Additional (optional) returners may be configured. E.g., for storing check results/metrics to a time series database, such as Graphite or InfluxDB.

(4) The reactor is configured to match monitoring events and pass them to the monitoring runner (5).

(5) Checks that are not specific to a minion can be run by the master. One essentially responsibility of these checks would be to ensure all expected checks are being submitted, by querying the database (8).

(6) The monitoring runner receives checks and route them through a series of configurable processing modules (7). Additionally, state information is checked and maintained in a pluggable database conforming to the SDB interface (8).

(7) Pluggable processing modules can perform additional actions on events, such as aggregation of related events, alerting, etc.

(8) Any database with an sdb adapter can be used as the persistent datastore. E.g., adapters for redis, riak, memcached, etc. could be used.

(9) Maintenance functions, such as notifying the monitoring system of new (expected) checks, placing entire minions or specific checks in maintenance mode, removing checks, etc. are performed by additional functions of the monitoring runner.