August 28

KEYSPACE

Amsterdam

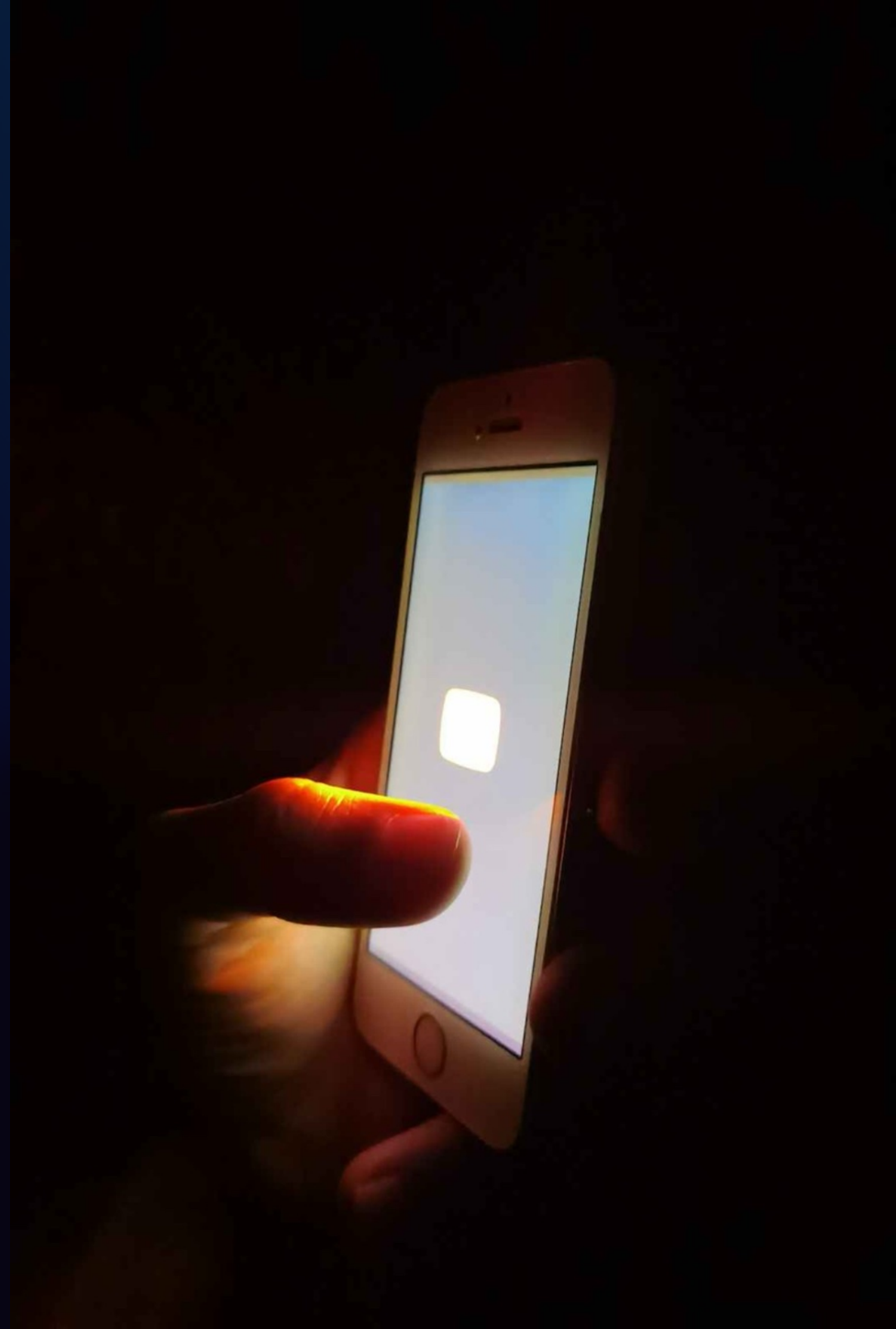
Connection Storms:
Detecting and Defusing Outages Before Valkey Crashes

Vadym Khoptynets
Senior Software Development Engineer at AWS

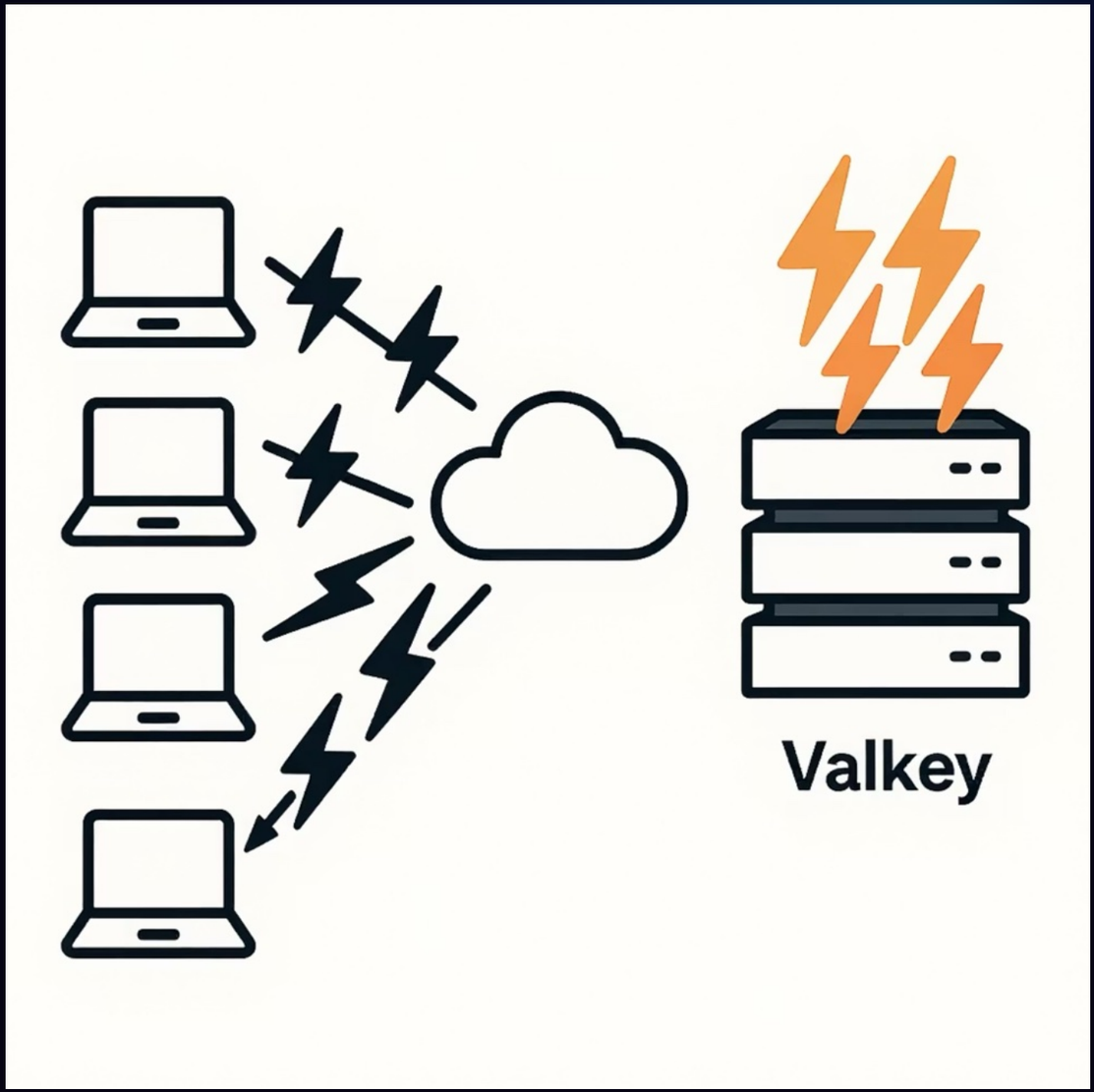
Valkey

The 2 AM Wakeup Call

- Pager goes off - Valkey cluster alert
- Grafana shows **gaps metrics**
- App team: "We can't connect to Valkey!"
- Root cause: connection storm



What is a Connection Storm?



Sudden surge of TCP connections overloads the networking stack

What is a Connection Storm?

Clients fail to connect

Network is overloaded

Accept queue is full

SYN queue is full

Conntrack table is full

Client Connection Management

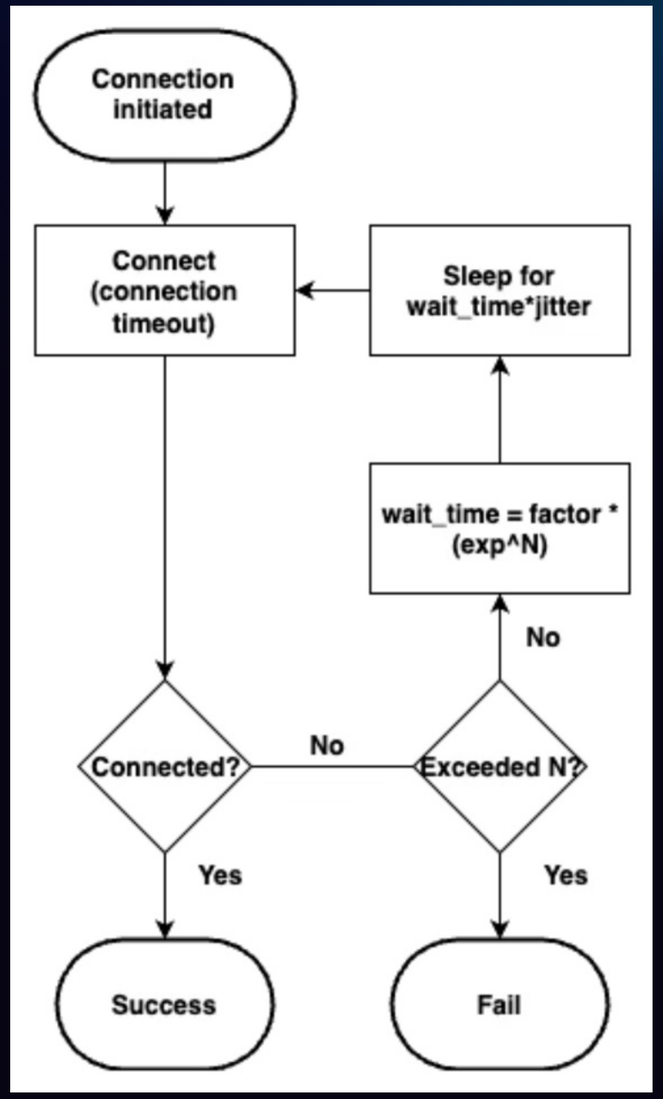
Connection Pool

- valkey-py
- Multiple sockets
- Reuses connections
- Risk of storms if pool is unbounded

Multiplexing

- valkey-glide
- Commands multiplexed on one TCP channel
- Lower overhead
- Can be a bottleneck

Client Reconnect Strategies



Reconnect Strategy

Number of retries
Number of retry attempts that the client should perform

Factor
The multiplier that will be applied to the waiting time between each retry.

Exponent base
The exponent base configured for the strategy

Jitter percent
The Jitter percent on the calculated duration

GLIDE example

```
backoff = BackoffStrategy(
    num_of_retries=5,
    factor=1000,
    exponent_base=2,
    jitter_percent=20,
)
```

```
config = GlideClientConfiguration(
        ...
        reconnect_strategy=backoff,
        ...
    )
client = await GlideClient.create(config)
```

Observability Gaps

Conntrack Table

What it does

Tracks all active network connections

Overflow policy

Drop

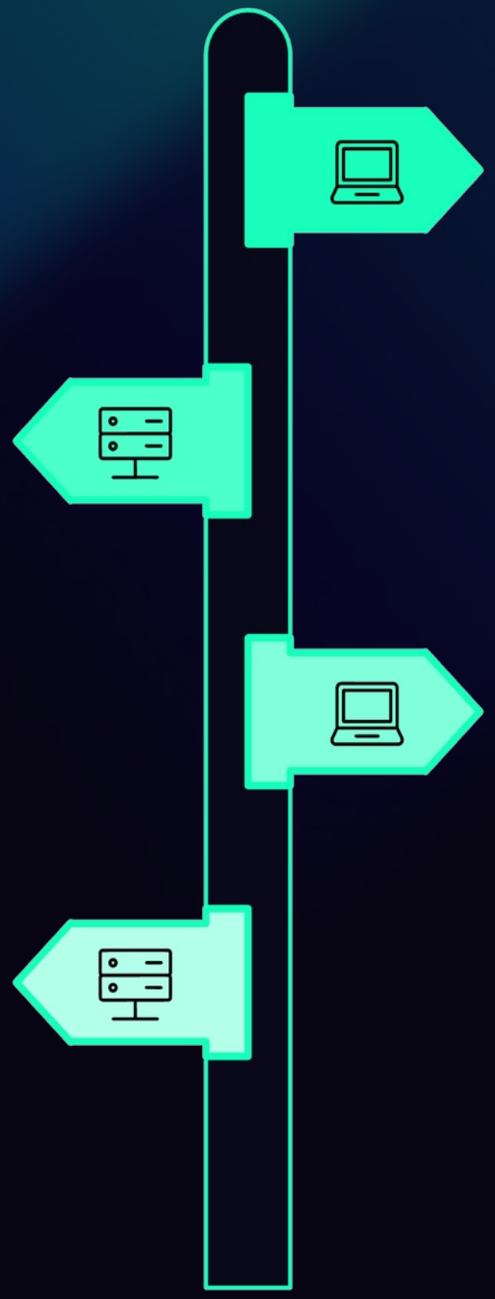
How to observe?

`net.netfilter.nf_conntrack_count`

How to configure?

`net.netfilter.nf_conntrack_max`

SYN Queue



Client sends SYN

Server appends to SYN queue

Responds with SYN, ACK

Client sends ACK

Server appends to accept queue

Waits for the process to accept

SYN Queue

What it does

Holds half-open TCP handshakes

Overflow policy

Sends SYN cookies (unless off)

How to observe?

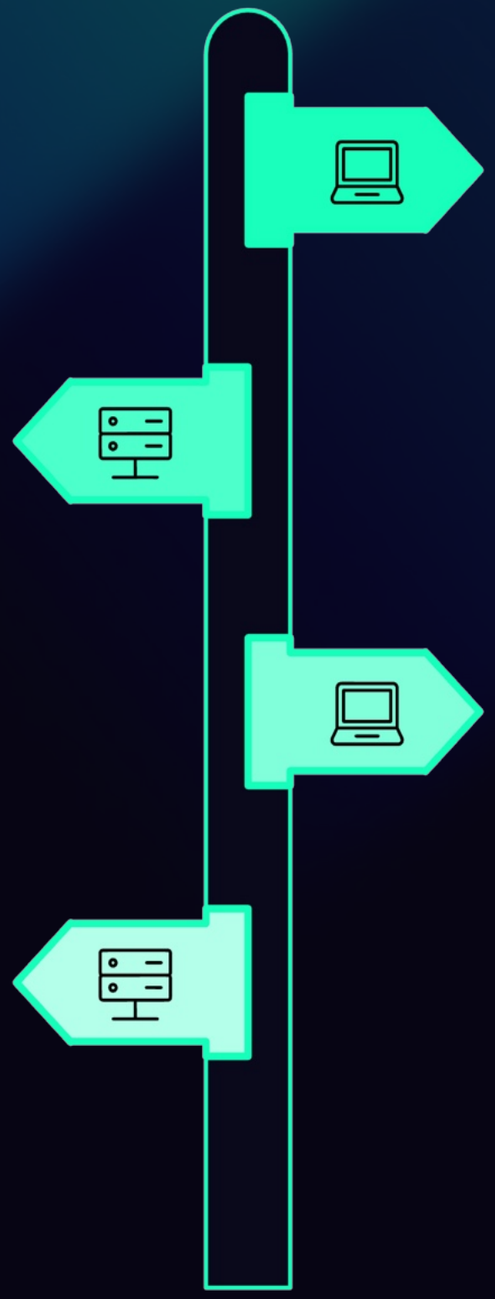
```
ss -anH state syn-recv | wc -l
```

How to configure?

`tcp_max_syn_backlog`

`tcp_syncookies`

SYN Queue



Client sends SYN

Server appends to SYN queue

Responds with SYN, ACK

Client sends ACK

Server appends to accept queue

Waits for the process to accept

Accept Queue

What it does

Tracks connections ready for Valkey to accept

Overflow policy

Sends RST when full

How to observe?

```
ss -Hln sport = <port> src <host>
       | awk '{print $4}'
```

How to configure?

`net.core.somaxconn`

`tcp-backlog`

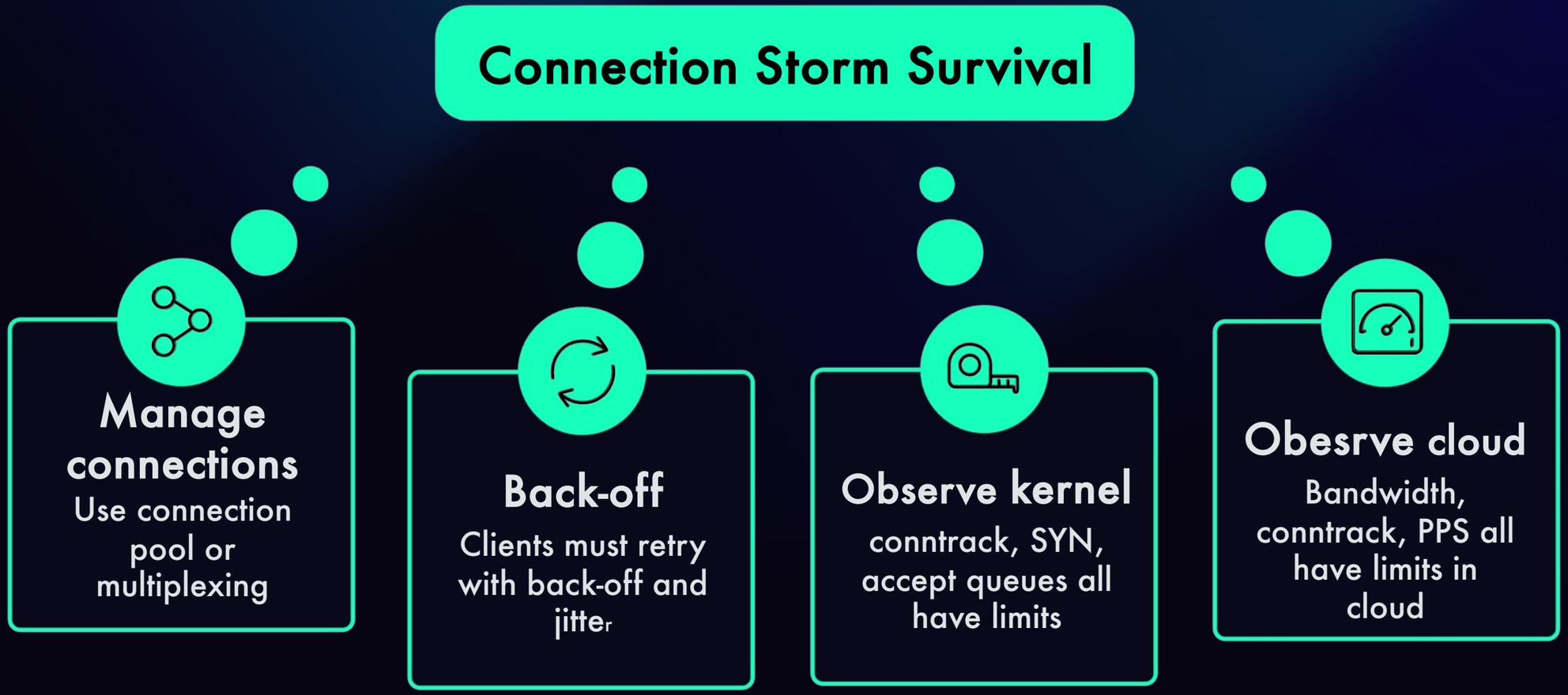
Cloud

Bandwidth

Conntrack

PPS

Connection Storm Checklist



Thank You!