

# BIND 9 Cache Improvements

**Michael Graff**

**Internet Systems Consortium**

**[michael\\_graff@isc.org](mailto:michael_graff@isc.org)**

# Reported Symptoms

- Servers which stop answering
- Oddly high CPU usage
- max-cache-size is ignored
- Restarting helps
- Busy recursive server
- Mostly from support customers

# The Test

- 3,000,000 A records (TTL 2 hours)
- Subset 1,000,000 A records
- Repeated query for single name
  
- Unlimited cache size (default)
- 10 minute cache cleaning interval

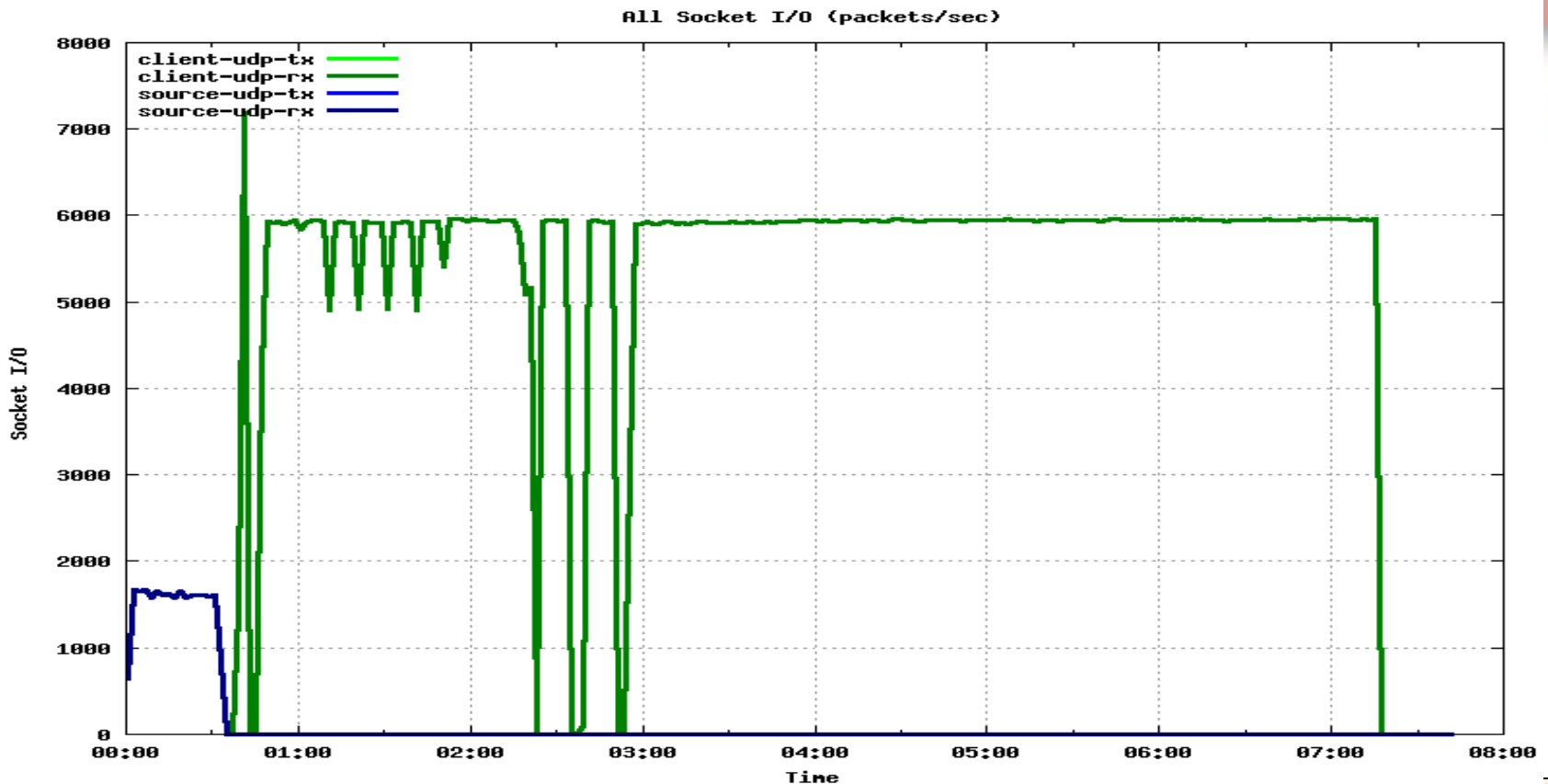
# The Numbers

- 1,500 queries/sec when filling cache
- 7,200 queries/sec when using cache
- 6,000 queries/sec while repeating the same question

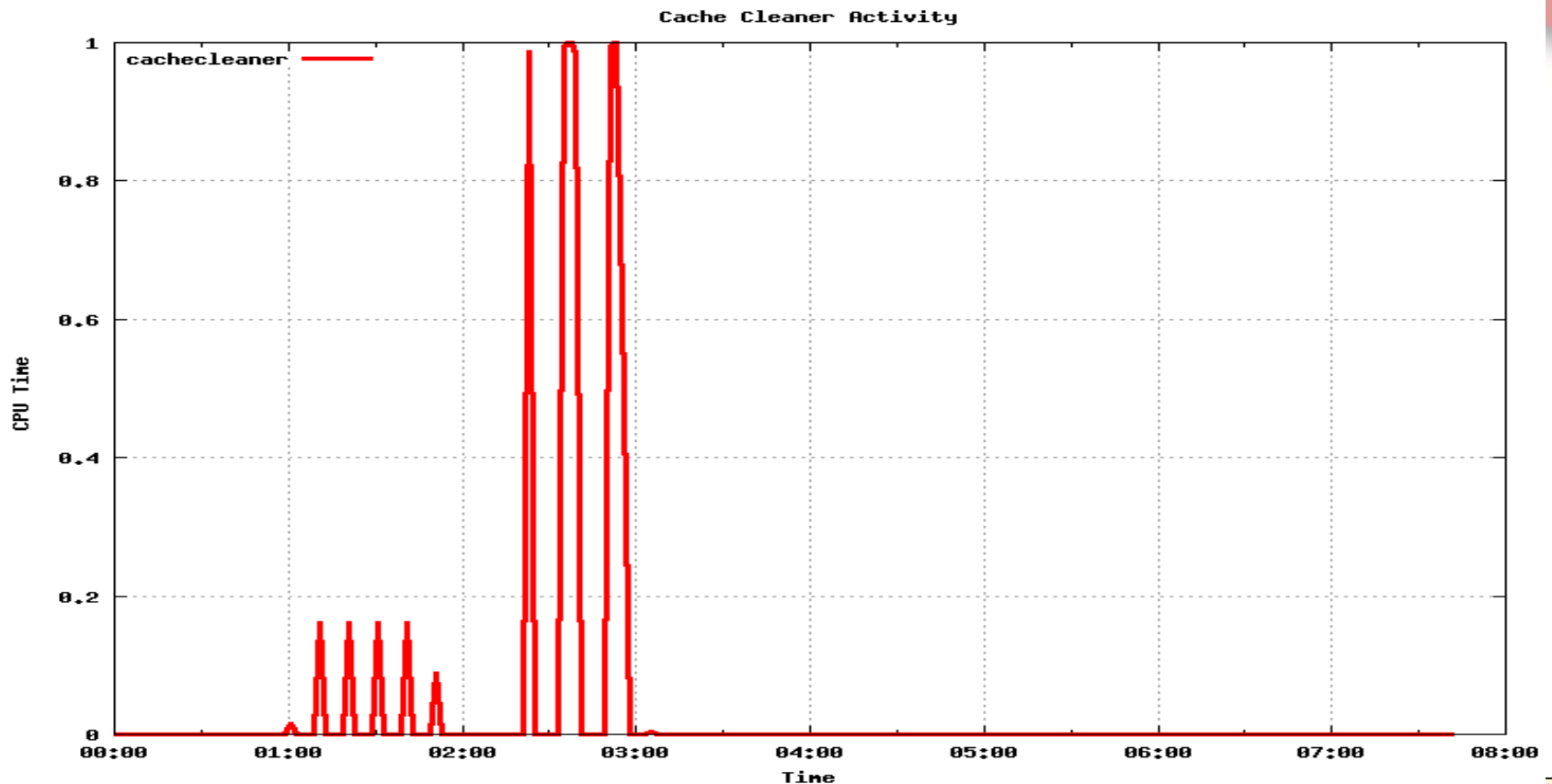
# The Numbers

- 1,500 queries/sec when filling cache
- 7,200 queries/sec when using cache
- 6,000 queries/sec while repeating the same question
- ... except when it was 24 queries/sec

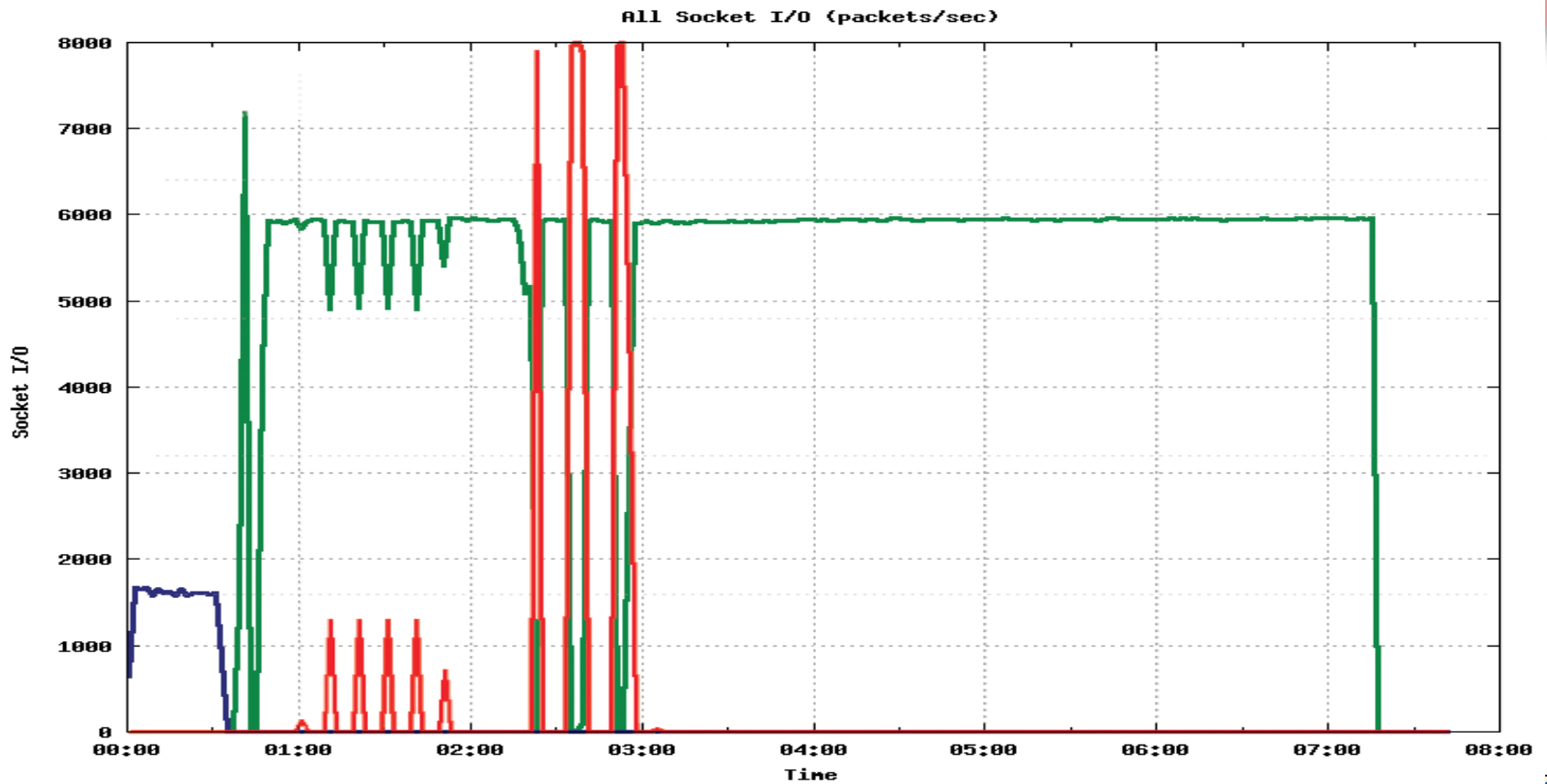
# Queries per Second (Unlimited, 395 MB peak)



# Cache Cleaner CPU Usage (Unlimited, 395 MB peak)



# Ah ha...





# Why the huge drop?

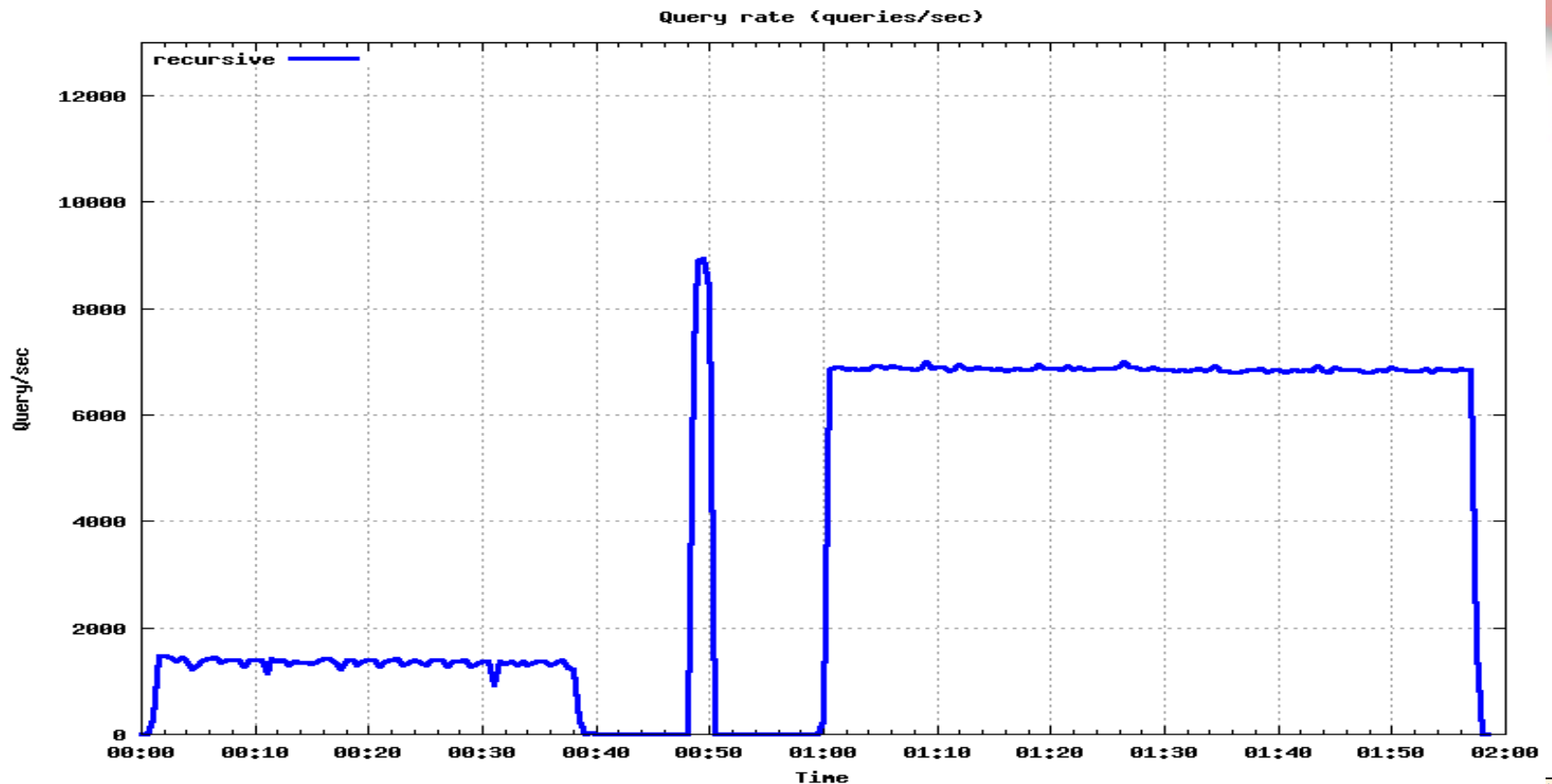
- When BIND 9 was designed, a "huge cache" was 512 M.
- Modern caches can exceed 4 GB
- Cache data structures probably good
- Cache locking probably good
- Tree walking to clean is bad

# Simple LRU Cleaning Design

- Initial implementation was from Jinmei Tatsuya
- Scalable to any size of cache
- Maintain or exceed current performance
- Predictable memory footprint
- No tree walking permitted

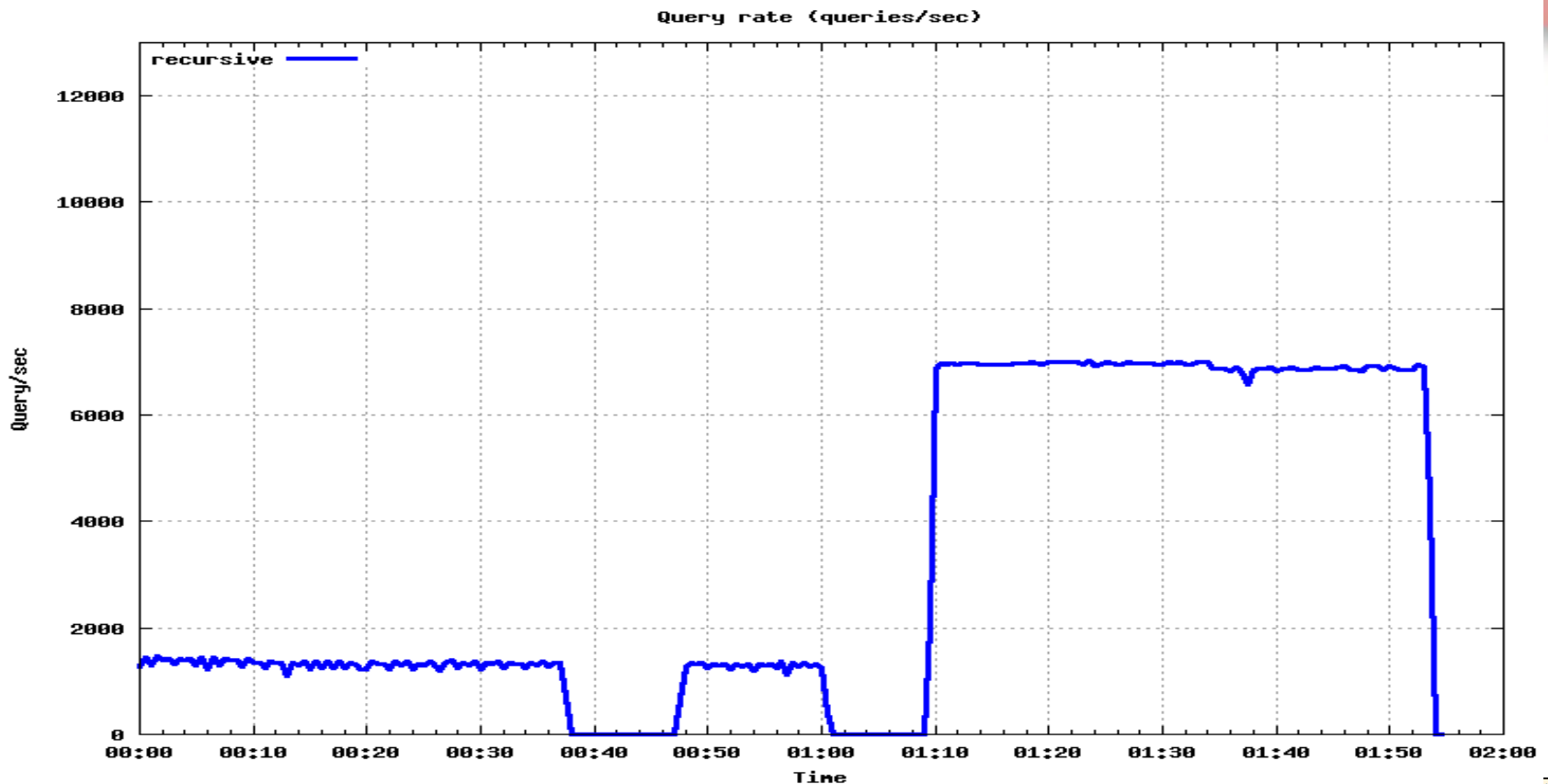
# LRU Cache Performance

Unlimited cache, 440 MB peak, 1 CPU



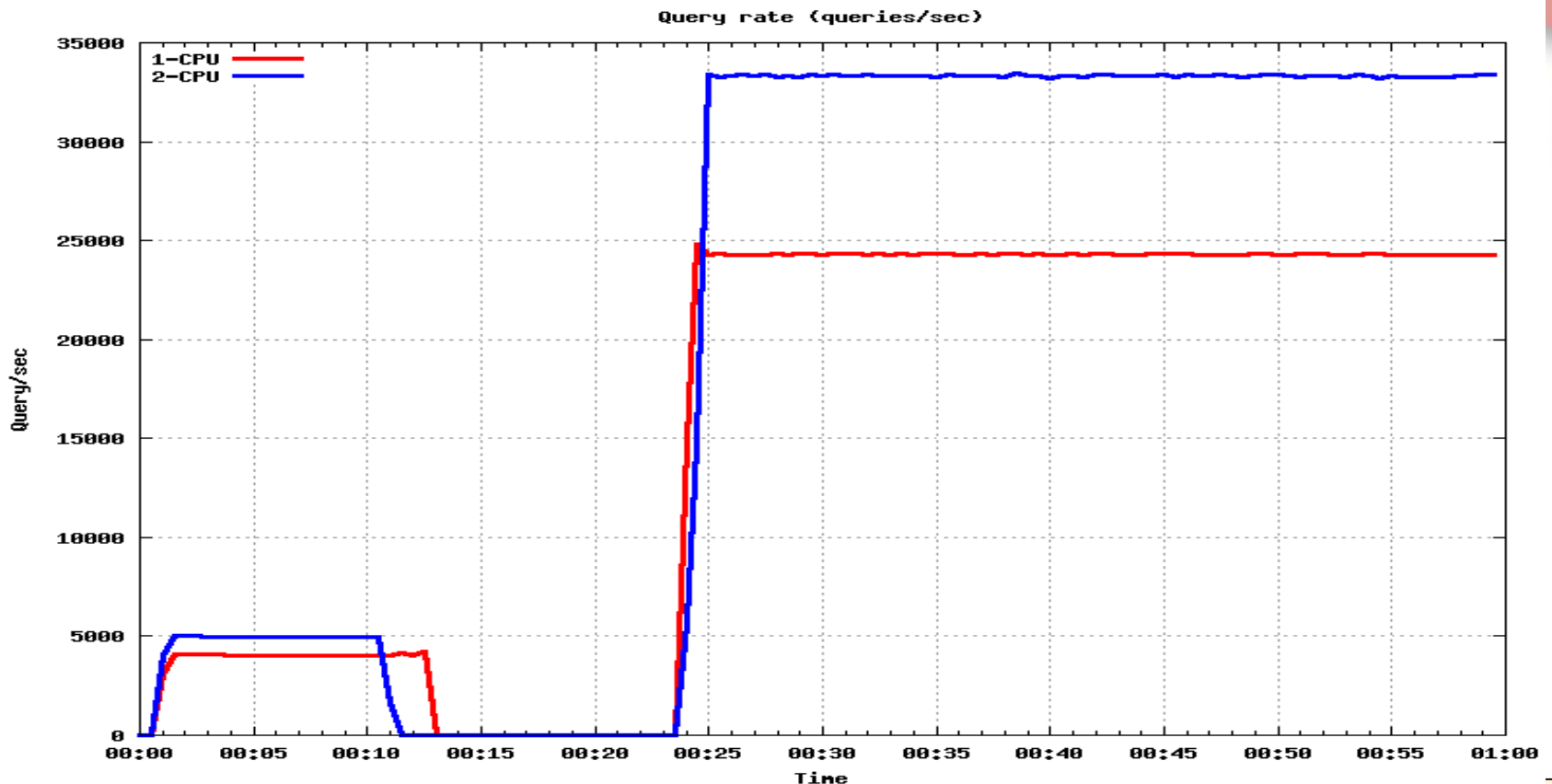
# LRU Cache Performance

140 MB limit, 147 MB peak, 1 CPU



# LRU Cache Performance

Unlimited, 764 MB peak, 1-2 CPUs  
Using NFS-funded test lab



# The Results

- Query rate is comparable ( $\sim 1400$ )
- Query rate drops only a little when aggressively cleaning ( $\sim 100$ )
- Even while aggressively cleaning, query rate is flat
- **Memory restriction is respected**
- **CPU load makes sense**

# Behavioral Changes

- Predictable behavior
  - Capacity planning
- Memory limits are respected
- Far less CPU time is wasted
- TTL is only partially used for expiry
- Default max-cache-size
- Approximately 10% more memory

# Future Work

- TTL-based expiry (linear time)
- Cache statistics (rrtype, queries)
- Controls for behavior
- Testing, analysis, documentation



# When can you get it?

- BIND Forum members can use CVS to get it today
- It will be in the next 9.5.0-alpha
- ... which means 9.5.0 will have it
- For support customers, 9.4 is an option we are evaluating

# What do you think?

- **Feedback** is critical
  - Defaults
  - Usefulness
  - Reliability