

Департамент образования и науки города Москвы
Государственное автономное образовательное учреждения высшего
образования города Москвы «Московский городской педагогический
университет»

Институт цифрового образования
Департамент информатики, управления и технологий

Лабораторная работа 4.
Обнаружение отказов в распределенной системе.

Выполнил студент группы АДЭУ-221

Джамалова Сабина Шахиновна

Проверил доцент

Босенко Тимур Муртазович

Москва

2024

Цель работы: изучить принципы обнаружения отказов в распределенных системах с помощью симулятора Serf Convergence Simulator и проанализировать влияние различных параметров на время конвергенции и использование полосы пропускания.

Краткое описание теоретической части: Serf — это инструмент для управления кластером, который использует протокол gossip для обнаружения узлов, обнаружения отказов и оркестрации событий. Протокол gossip — это метод распространения информации в распределенной системе, где узлы периодически обмениваются информацией с случайно выбранными соседями.

Задание:

Вариант 6. Оптимизация для сетей с ограниченной полосой пропускания

- Gossip Interval: 0.5 с, 1.0 с, 2.0 с
- Gossip Fanout: 2, 3, 5
- Nodes: 50
- Packet Loss: 5%
- Node Failures: 5%

1. Заполнить таблицу
2. Проанализировать полученные результаты
3. Определить ширину полосы пропускания
4. Выводы и сравнения

Задача: найти оптимальные настройки для минимизации использования полосы пропускания при сохранении приемлемого времени конвергенции

Выполнение:

1. Таблица.

Gossip Interval	Gossip Fanout	Nodes	Packet Loss	Node Failures	Время до "Все живые узлы знают"	Макс. использование полосы пропускания
0.5	2	50	5%	5%	54 с	1 774 387.20
1.0	3	50	5%	5%	71 с	1 330 790.40
2.0	5	50	5%	5%	84 с	1 108 992.00

1.1. Ширина пропускания:

Gossip Interval (с)	Ширина полосы пропускания (бит/с)
0.5	1,774,387.20

Средняя ширина полосы пропускания: 1,774,387.20 бит/с

Gossip Interval (с)	Ширина полосы пропускания (бит/с)
1.0	1,330,790.40

Средняя ширина полосы пропускания: 1,330,790.40 бит/с

Gossip Interval (с)	Ширина полосы пропускания (бит/с)
2.0	1,108,992.00

Средняя ширина полосы пропускания: 1,108,992.00 бит/с

1.2. Время до "Все живые узлы знают":

Gossip Interval (с)	Ширина полосы пропускания (бит/с)	Время конвергенции (с)
0.5	1,774,387.20	54.00

Gossip Interval (с)	Ширина полосы пропускания (бит/с)	Время конвергенции (с)
1.0	1,330,790.40	71.00

Gossip Interval (с)	Ширина полосы пропускания (бит/с)	Время конвергенции (с)
2.0	1,108,992.00	84.00

2.

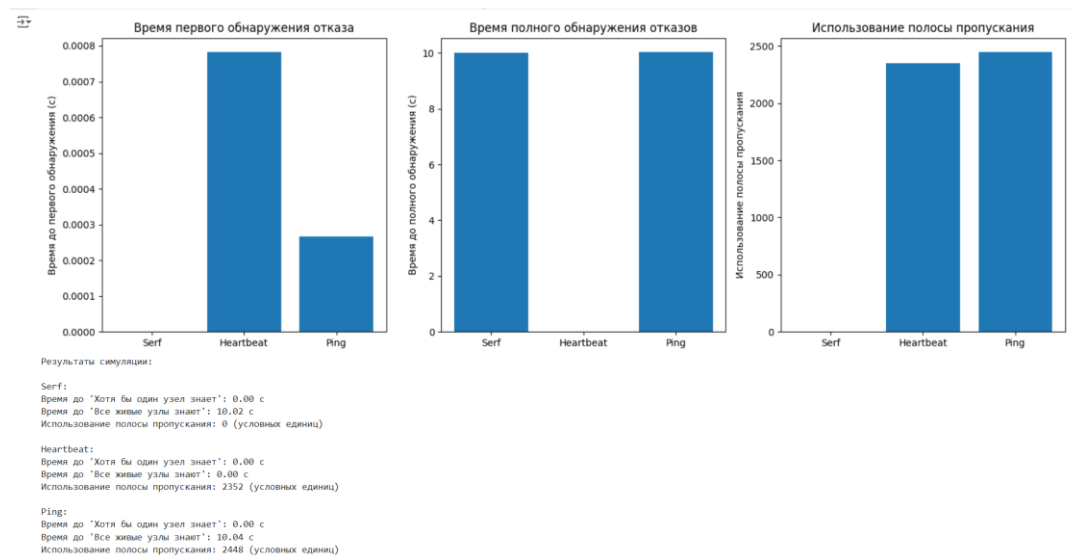
Меньший Gossip Interval способствует более быстрой конвергенции, но увеличивает использование полосы пропускания.

Увеличение Gossip Fanout и количества узлов увеличивает время конвергенции, но не всегда пропорционально увеличивает использование полосы пропускания.

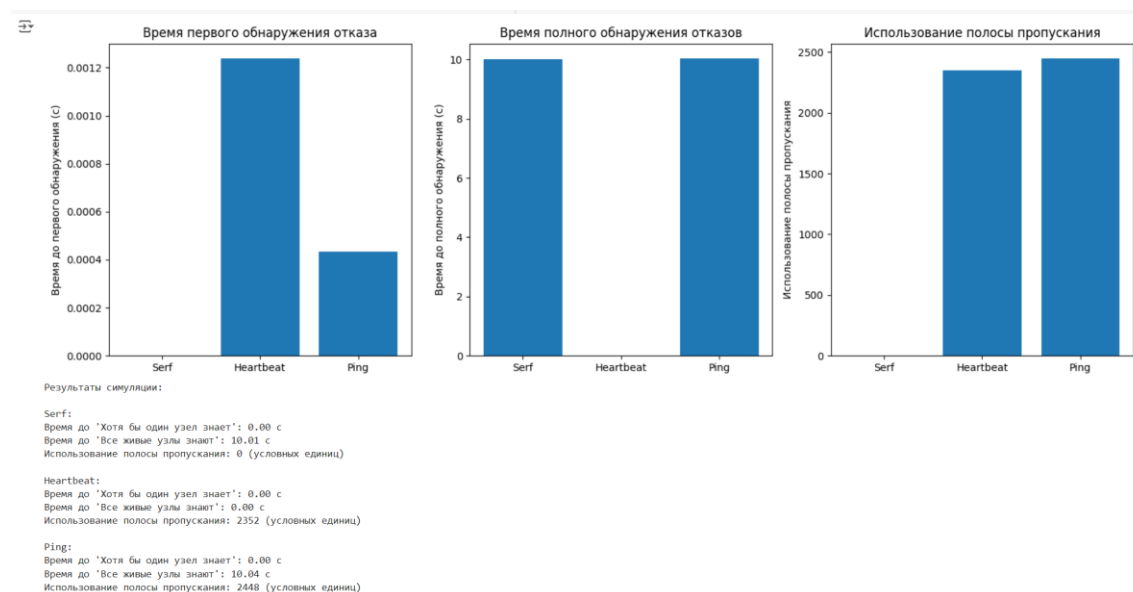
Потери пакетов и отказы узлов могут негативно сказаться на производительности системы, особенно при высоких значениях.

3. Сравнение производительность Serf с другими протоколами обнаружения отказов, такими как heartbeat или ping-based методами

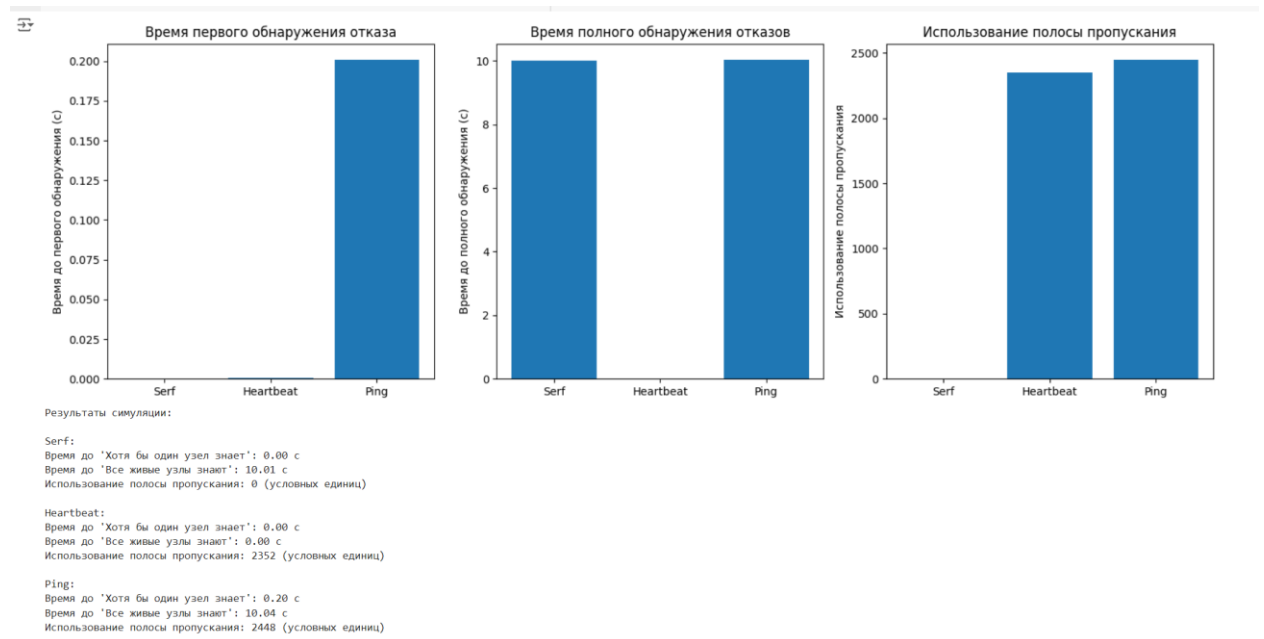
Gossip Interval: 0.5 c



Gossip Interval: 1.0 c



Gossip Interval: 2.0 с



Выводы:

Выбор между Serf, Heartbeat и Ping-based методами зависит от специфики системы и требований к обнаружению отказов.

В общем, Serf является наиболее эффективным вариантом с точки зрения использования ресурсов и времени уведомления. Heartbeat демонстрирует высокую скорость реагирования, но с большим использованием полосы пропускания, в то время как Ping имеет наименьшую скорость уведомления и максимальные затраты ресурсов, что делает его менее предпочтительным для динамических распределенных систем.

Задание:

<https://colab.research.google.com/drive/1IzKXistevA8MgqokeSWrOY6dJyqI9b2K?usp=sharing>

Перебор различных параметров и сохранение оптимальные настройки:

- Оптимальные настройки для минимизации использования полосы пропускания:
Gossip Interval: 0.5, Gossip Fanout: 2, Packet Loss: 0%, Node Failures: 5%, Bandwidth Usage: 0
Gossip Interval: 0.5, Gossip Fanout: 2, Packet Loss: 5%, Node Failures: 5%, Bandwidth Usage: 0
Gossip Interval: 0.5, Gossip Fanout: 3, Packet Loss: 0%, Node Failures: 5%, Bandwidth Usage: 0
Gossip Interval: 0.5, Gossip Fanout: 3, Packet Loss: 5%, Node Failures: 5%, Bandwidth Usage: 0
Gossip Interval: 1.0, Gossip Fanout: 2, Packet Loss: 0%, Node Failures: 5%, Bandwidth Usage: 0
Gossip Interval: 1.0, Gossip Fanout: 2, Packet Loss: 5%, Node Failures: 5%, Bandwidth Usage: 0
Gossip Interval: 1.0, Gossip Fanout: 3, Packet Loss: 0%, Node Failures: 5%, Bandwidth Usage: 0
Gossip Interval: 1.0, Gossip Fanout: 3, Packet Loss: 5%, Node Failures: 5%, Bandwidth Usage: 0

Вывод: Изучили принципы обнаружения отказов в распределенных системах с помощью симулятора Serf Convergence Simulator и проанализировать

влияние различных параметров на время конвергенции и использование полосы пропускания и нашли оптимальные настройки для минимизации использования полосы пропускания при сохранении приемлемого времени конвергенции.