

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

УДК 004.77, 004.62

СРАВНИТЕЛЬНОЕ ИССЛЕДОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ СИСТЕМ SERVICE MESH ДЛЯ ОРКЕСТРАТОРА KUBERNETESА.В. ШУЛЯК¹, А.Г. САВЕНКО²¹*Divido Financial Services Ltd**Camden Lock Market, Chalk Farm Rd, Лондон, NW1 8AB, Великобритания*²*Институт информационных технологий учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», ул. Козлова 28, Минск, 220037, Беларусь**Поступила в редакцию 15 ноября 2021*

В статье рассматривается проблема выбора системы Service Mesh с точки зрения производительности в высоконагруженных системах. Представлены результаты экспериментальных исследований трех основных систем защиты коммуникации между сервисами одной платформы (Istio, Linkerd, Traefik Mesh) с различными степенями нагрузки на каждую и перечень основных метрик для определения наилучшей пиковой производительности.

Ключевые слова: высокопроизводительные системы, service mesh, сети обработки данных, kubernetes, контейнеризация, веб-сервера, прокси-сервера.

Введение. В современном мире подавляющее число веб-приложений работает внутри контейнерной системы виртуализации. Основной системой оркестрации является система Kubernetes. Kubernetes – это программное обеспечение с открытым исходным кодом, платформа оркестрации и менеджмента контейнеров, созданное компанией Google и занимающее существенную долю рынка. Несмотря на развитость системы, в базовой конфигурации оркестратора существует существенный недостаток – все коммуникации между контейнерами внутри Kubernetes производятся без использования шифрования, т. е. в открытом виде через протокол http [1].

Для решения данной проблемы существуют так называемые системы Service Mesh. Service Mesh представляет собой слой инфраструктуры, который облегчает, упрощает и защищает коммуникации между сервисами внутри одной платформы [2]. Основными представителями систем Service Mesh для оркестратора Kubernetes являются Istio, Linkerd, Traefik Mesh [3].

В данном исследовании предлагается сравнить и проанализировать производительность данных систем относительно друг друга. За эталон для сравнения взяты данные производительности без использования систем Service Mesh, т. е. с коммуникациями в открытом виде.

Методика проведения исследования и конфигурация вычислительного кластера. Исходя из опыта проведения нагрузочных тестов, для создания тестовой нагрузки использовался пакет автоматизации тестирования Kinvolk Service Mesh Benchmark. В качестве целей тестовой нагрузки было выбрано демо-приложение Emojivoto. Для проведения исследования производительности создается 10 экземпляров демо-приложения, а нагрузочное тестирование идет в 96 потоков, в три этапа с различными степенями нагрузки:

- первый этап: 50 запросов в секунду;
- второй этап: 450 запросов в секунду;
- третий этап: 3500 запросов в секунду.

Каждый из этапов тестирования проводился 5 раз для исключения погрешностей измерений и выравнивания результатов. Контрольный замер производительности производился без использования системы Service Mesh (эталонные данные для сравнения).

Ключевой метрикой для оценки производительности систем Service Mesh являются задержки при получении ответов. Задержки оцениваются со стороны клиента, а распределение задержек получено, используя перцентили.

Дополнительными критериями оценки производительности будут являться затраты ресурсов процессора и оперативной памяти на контролируемых узлах и узлах пересылки данных в компонентах систем Service Mesh.

Тестирование производилось на базе вычислительных мощностей облачного провайдера Linode. Использовалась следующая конфигурация вычислительного кластера:

- 3 управляющих вычислительных узла с 4 вычислительными ядрами и 8 Гб оперативной памяти;
- 5 вычислительных узлов с 50 вычислительными ядрами и 128 Гб оперативной памяти;
- все вычислительные узлы связаны сетью с пропускной способностью 10 Гбит/с;
- кластер был развернут при помощи системы удаленной конфигурации Kubespray, базирующейся на инструменте Ansible.

Для исследования был выбран оркестратор Kubernetes версии 1.20 и следующие версии систем Service Mesh: Linkerd версии 2.11.1; Istio версии 1.11.4; Traefik Mesh версии 1.4.4.

Полученные результаты. В результате исследования на первом этапе (50 запросов в секунду) были получены результаты, отраженные в табл. 1–6.

Таблица 1
Задержка перцентилей 50%

Номер измерения	Задержка перцентилей 50%, мс			
	Baseline	Istio	Traefik Mesh	Linkerd
1	7.2	29.33	23.12	19.22
2	7.3	29.66	22.34	19.84
3	7.2	31.12	22.17	19.12
4	7.4	29.9	22.46	19.99
5	7.39	30.7	23.28	19.26

Таблица 2
Задержка перцентилей 90%

Номер измерения	Задержка перцентилей 90%, мс			
	Baseline	Istio	Traefik Mesh	Linkerd
1	10.11	72.12	58.21	47.51
2	10.12	72.66	56.12	47.4
3	10.12	73.09	56.78	48.1
4	10.16	74.0	57.44	46.9
5	10.11	72.52	57.88	48.16

Таблица 3
Задержка перцентилей 99%

Номер измерения	Задержка перцентилей 99%, мс			
	Baseline	Istio	Traefik Mesh	Linkerd
1	14.1	230.22	122.43	89.19
2	14.32	238.12	122.58	88.46
3	14.22	244.57	127.33	74.55
4	14.11	229.6	125.6	74.26
5	14.46	232.44	124.2	70.22

Таблица 4
Задержка перцентилей 100%

Номер измерения	Задержка перцентилей 100%, мс			
	Baseline	Istio	Traefik Mesh	Linkerd
1	18.13	231.67	155.57	107.44
2	18.34	235.82	158.63	108.66
3	18.74	244.55	164.1	106.27
4	18.91	229.37	155.11	106.27
5	18.21	232.75	154.6	107.22

Таблица 5
Загрузка процессора

Загрузка процессора, %			
Слой приложения			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	0.0933	0.0743	0.00342
2	0.12	0.0784	0.00347
3	0.0942	0.0782	0.00355
4	0.0940	0.0755	0.00377
5	0.0966	0.0744	0.00399
Управляющий слой			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	5.12	0.098	0.0822
2	4.88	0.096	0.0826
3	6.22	0.164	0.0833
4	4.36	0.212	0.0828
5	5.55	0.185	0.0870

Таблица 6
Использование оперативной памяти

Использование оперативной памяти, Мб			
Слой приложения			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	215	43	14.2
2	217	42.5	13.6
3	214	44	13.2
4	216	43	13.8
5	216	43.6	14.1
Управляющий слой			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	960	374	310
2	944	382	316
3	970	388	298
4	968	386	305
5	983	379	307

В результате исследования на втором этапе (450 запросов в секунду) были получены результаты, отраженные в табл. 7–12.

Таблица 7
Задержка перцентилей 50%

Задержка перцентилей 50%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	7.11	31.04	23.55	19.76
2	7.21	31.05	24.44	19.43
3	7.23	30.11	22.98	19.54
4	7.24	29.4	24.22	19.34
5	7.23	30.62	22.44	19.76

Таблица 8
Задержка перцентилей 90%

Задержка перцентилей 90%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	10.17	72.36	58.54	47.45
2	10.16	72.99	56.22	47.65
3	10.16	73.09	56.17	48.43
4	10.15	73.20	57.65	47.11
5	10.17	73.1	57.42	47.11

Таблица 9
Задержка перцентилей 99%

Задержка перцентилей 99%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	15.77	247.43	130.43	92.45
2	15.33	245.21	131.58	93.51
3	15.21	254.54	131.33	93.71
4	15.41	245.36	131.46	92.72
5	15.36	243.11	137.22	91.54

Таблица 10
Задержка перцентилей 100%

Задержка перцентилей 100%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	20.46	281.12	185.43	121.48
2	20.81	285.43	188.54	123.76
3	20.23	284.46	184.51	125.47
4	20.17	301.27	185.11	124.27
5	20.45	292.11	184.17	123.21

Таблица 11

Загрузка процессора

Загрузка процессора, %			
Слой приложения			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	0.0921	0.0765	0.00421
2	0.0964	0.0733	0.00388
3	0.0983	0.0721	0.00426
4	0.0955	0.0793	0.00476
5	0.0932	0.0712	0.00396
Управляющий слой			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	6.65	0.0951	0.0865
2	4.41	0.0957	0.0812
3	3.34	0.0988	0.0811
4	4.34	0.1763	0.0977
5	4.15	0.1889	0.0889

Таблица 12

Использование оперативной памяти

Использование оперативной памяти, Мб			
Слой приложения			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	214	44.5	15.1
2	212	48	17.
3	217	51	21.3
4	214	51	19.4
5	214	49	18.4
Управляющий слой			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	954	365	335
2	936	387	342
3	988	401	311
4	954	409	328
5	997	407	331

В результате исследования на третьем этапе (3500 запросов в секунду) были получены результаты, отраженные в табл. 13–18.

Таблица 13

Задержка перцентилей 50%

Задержка перцентилей 50%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	7.45	28.45	22.11	17.43
2	7.51	28.53	22.17	17.12
3	7.65	29.11	22.43	17.43
4	7.41	27.44	21.17	17.43
5	7.11	28.12	23.10	17.65

Таблица 14

Задержка перцентилей 90%

Задержка перцентилей 90%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	9.46	67.34	54.21	43.11
2	9.22	66.82	54.22	44.82
3	9.34	67.25	54.32	44.17
4	9.45	67.82	54.38	43.54
5	9.12	67.99	54.76	43.22

Таблица 15

Задержка перцентилей 99%

Задержка перцентилей 99%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	11.43	210.85	107.45	84.22
2	11.72	210.64	107.84	83.16
3	11.16	212.72	105.98	84.92
4	11.64	210.47	106.54	85.16
5	11.82	210.94	109.17	82.11

Таблица 16

Задержка перцентилей 100%

Задержка перцентилей 100%, мс				
Номер измерения	Baseline	Istio	Traefik Mesh	Linkerd
1	24.93	293.44	199.88	163.77
2	24.83	297.52	199.65	163.55
3	25.33	297.14	201.43	163.43
4	22.74	293.57	195.65	177.41
5	21.45	299.58	192.59	168.54

Таблица 17

Загрузка процессора

Загрузка процессора, %			
Слой приложения			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	0.1045	0.1043	0.00976
2	0.1034	0.1022	0.00985
3	0.1143	0.0994	0.00994
4	0.1541	0.0997	0.00977
5	0.1004	0.0981	0.00985
Управляющий слой			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	3.44	0.0951	0.0844
2	5.22	0.0957	0.0824
3	4.61	0.0988	0.0822
4	4.45	0.1763	0.0877
5	4.65	0.1889	0.0896

Таблица 18

Использование оперативной памяти

Использование оперативной памяти, Мб			
Слой приложения			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	239	54	27
2	232	55	25
3	237	54	24
4	238	61	28
5	239	59	26
Управляющий слой			
Номер измерения	Istio	Traefik Mesh	Linkerd
1	1015	470	380
2	1073	465	392
3	998	482	391
4	1034	475	382
5	1022	477	399

Обсуждение полученных результатов. Рассмотрим полученные задержки при различной нагрузке (50, 450 и 3500 запросов в секунду). Первый этап тестирования – 50 запросов в секунду, что в современном мире соответствует низкой нагрузке на веб-сервер. Как видно из графика (рис. 1а), системы Linkerd и Traefik лучше справляются с нагрузкой. Медианное время запросов не превышает 11 миллисекунд для Linkerd и 20 миллисекунд для Traefik, однако видно ощутимое отставание системы Istio, как по средним задержкам, так и по максимальным, которые превышают задержки Linkerd более чем в два раза.

Замер задержек при средней нагрузке 450 запросов в секунду (второй этап тестирования) имеет схожие показатели (рис. 1б). Система Linkerd также имеет минимальные задержки – не более 125 миллисекунд. И также вторым по показателям является система Traefik с максимальным показателем в 170 миллисекунд. Максимальные задержки по-прежнему имеет система Istio – 288 миллисекунд. При этом медианное значения задержек держатся на уровне 30, 23 и 19 миллисекунд для Istio, Traefik и Linkerd соответственно.

Далее рассмотрим поведение Service Mesh систем под высокой нагрузкой – 3500 запросов в секунду. Медианное время запросов по всем системам все еще остается на уровне 17-28 миллисекунд, однако максимальные задержки существенно отличаются. Задержки через систему Istio достигают практически 300 миллисекунд на каждый запрос, в то время как Linkerd удерживает максимальное время задержки на уровне 167 миллисекунд и Traefik – 197 миллисекунд (рис. 1в).

Рассмотрим показатели производительности при различном количестве запросов. Как можно видеть из графиков загрузки ресурсов (рис. 2 и рис. 3) наименьшую нагрузку дает система Linkerd. На слое приложений различия по загрузке ресурсов процессора системами Istio и Traefik незначительны. В свою очередь потребление оперативной памяти примерно эквивалентно у систем Traefik и Linkerd и существенно больше у системы Istio.

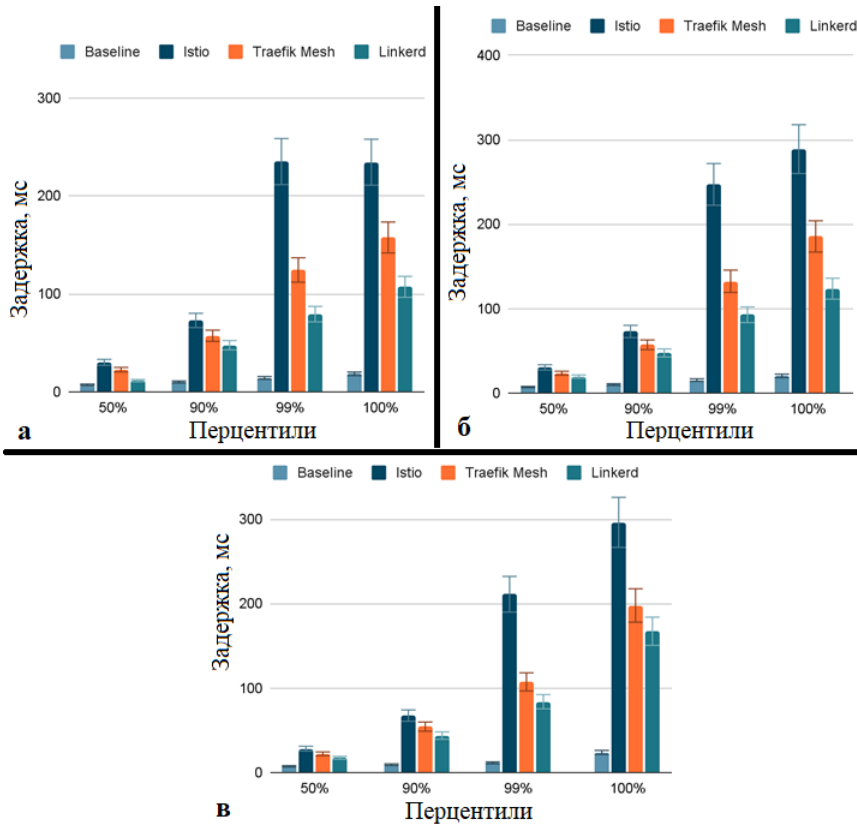


Рис. 1. Диаграммы распределения задержек

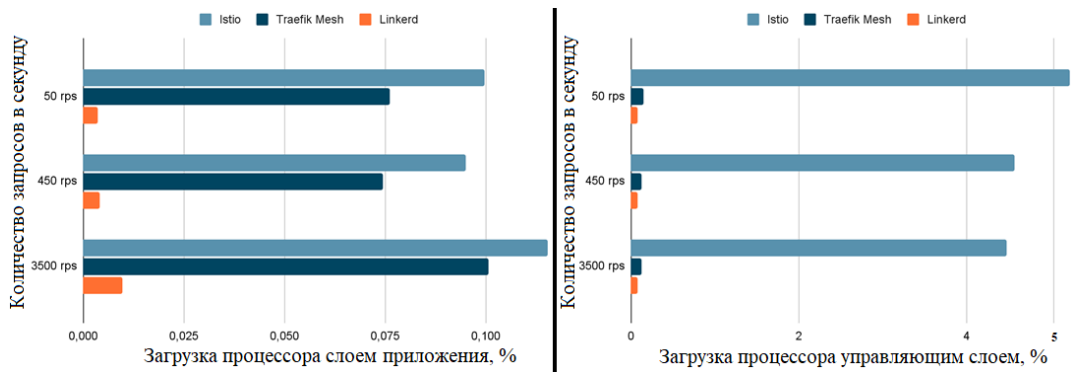


Рис. 2. Диаграммы загрузки процессора при различном количестве запросов в секунду

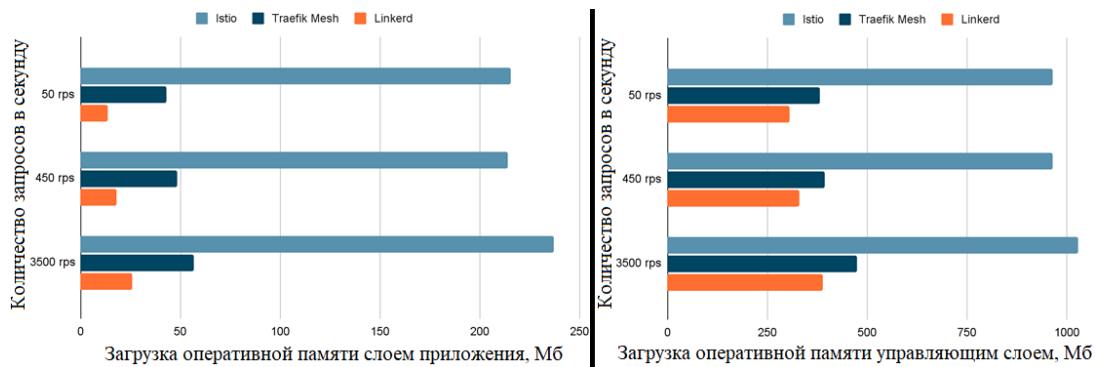


Рис. 3. Диаграммы загрузки оперативной памяти при различном количестве запросов в секунду

Заключение. В результате проведенного исследования было установлено, что наилучшую производительность демонстрирует система Linkerd, она имеет наименьшие задержки в сочетании с наименьшим потреблением ресурсов, что также согласуется со схожими исследованиями [4]. В свою очередь система Traefik не сильно отстает по показателям от системы Linkerd и имеет схожую производительность при немного большем потреблении ресурсов. Существенным аутсайдером по производительности выступает система Istio, она имеет наибольшие средние и максимальные задержки, и при этом расходует ощутимо большие ресурсы. Отчасти данный факт можно объяснить возможной гибкостью настройки данной системы, негативно сказывающейся, в свою очередь, на производительности. Таким образом, при отсутствии необходимости максимальной гибкой настройки системы Service Mesh, Linkerd и Traefik являются оптимальным выбором.

COMPARATIVE RESEARCH OF PERFORMANCE OF SERVICE MESH SYSTEMS FOR ORCHESTRATOR KUBERNETES

A.V. SHULYAK, A.G. SAVENKO

Abstract

The article discusses the problem of choosing a Service Mesh system from the point of view of performance in highly loaded systems. The results of experimental studies of three main systems for protecting communication between services of the same platform (Istio, Linkerd, Traefik Mesh) with different degrees of load on each and a list of basic metrics to determine the best peak performance are presented.

Список литературы

1. Repettoa, M. An architecture to manage security operations for digital service chains / M. Repettoa [et al.] // Future Generation Computer Systems – 2021. – Т. 115. – P. 251–266.
2. Kang, M. Protected Coordination of service mesh for container-based 3-tier service traffic / M. Kang, J. S. Shin, J. Kim // 33rd International Conference on Information Networking. – 2019. – Т. 275. – P. 427–429.
3. Егоров, Н. С. Сравнительная характеристика Service Mesh и Middleware подходов в организации взаимодействия в распределенных информационных системах / Н. С. Егоров // Студенческий Форум. – 2021. – № 142. – С. 46–49.
4. Li, W. Service Mesh: challenges, state of the art, and future research opportunities / W. Li [et al.] // Proceedings - 13th IEEE international conference on service-oriented system engineering, sose. – 2019. – Т. 310 – P. 122–127.