

**Информация, необходимая для эксплуатации экземпляра
программного обеспечения, предоставленного для проведения
экспертной проверки
программы «Программное обеспечение для сбора информации из
разнородных источников, структурированных и неструктурированных
данных для классификации и извлечения заранее определенных
сущностей и критериев. Peraspera NeuroDoc»**

1 Состав инсталляционного пакета

В состав инсталляционного пакета программы «Peraspera NeuroDoc» входит инсталлятор, который устанавливает указанное в нижеприведенной таблице (Таблица 1) свободно распространяемое ПО, необходимое для функционирования программы «Peraspera NeuroDoc».

Таблица 1 – Состав инсталляционного пакета программы «Peraspera NeuroDoc»

№	Подсистема	Свободно распространяемое ПО	Назначение
1	Подсистема управления оркестрацией	Kubernetes Dashboard	Обеспечение оркестрации сервисов
		Kubernetes Metrics Service	
2	Подсистема информационной безопасности	Keycloak	Обеспечение информационной безопасности
3	Подсистема аналитики, включающая модели машинного обучения и сервисы аналитики	NVIDIA Device Plugin	Обеспечение обработки данных
		TensorFlow Serving	
		Peraspera-Neuro	
		HAProxy	
		PostgreSQL	
		Keycloak	
		Elasticsearch	
		Docreader	

**2 Требования к квалификации специалиста, выполняющего
установку**

Специалист должен иметь навыки работы с ОС Linux(Ubuntu), Docker, Kubernetes, протокол безопасности OIDC (OpenID Connection).

3 Требования к оборудованию

Минимальные требования к оборудованию для инсталляции и функционирования программы «Peraspera NeuroDoc»:

- 1) Для корректного функционирования среды исполнения программы необходим сервер, обладающий следующими характеристиками:
 - ЦПУ восьмиядерный (8 Core);
 - Жесткий диск – не менее 250Gb;
 - Оперативной памяти – не менее 16Gb;
 - Видеокарта GeForce GTX 1080, объем памяти - 8Gb.
- 2) Для функционирования приложений, запускаемых в среде программы «Peraspera NeuroDoc», необходимо оборудование, характеристики которого приведены в нижеприведенных таблицах (Таблица 2, Таблица 3).

Таблица 2 – Требования к среде установки

Наименование виртуальной машины	Выделенное пространство	Объем памяти	Кол-во ядер процессора (CPUs)	Кол-во контроллеров (NICs)
kubernetes-master-01	100 GB	8 GB	4	1
kubernetes-master-02	40 GB	8 GB	4	1
kubernetes-master-03	40 GB	8 GB	4	1
perasp-node1	32 GB	16 GB	16	1
perasp-node2	32 GB	16 GB	16	1

Таблица 3 – Требования к физическим серверам

Наименование сервера	Выделенное пространство	Объем памяти	Кол-во ядер процессора (CPUs)	Кол-во контроллеров NICs
perasp-nr-01	10 ГБ	4 GB	2	2
perasp-nr-02	10 ГБ	4 GB	2	2
perasp-gp-03	10 ГБ	16 GB	4	2
perasp-gp-04	10 ГБ	16 GB	4	2
perasp-gp-05	10 TB	16 GB	4	2

4 Требования к системному программному обеспечению

Требования, предъявляемые к системному программному обеспечению для установки и эксплуатации Программы «Peraspera NeuroDoc»:

- ubuntu сервер 18.04 и выше,
- docker 19.03.2 и выше,
- kubernetes 1.18.4 и выше

5 Порядок установки

Для установки программы «Peraspera NeuroDoc» необходимо:

На оборудовании, соответствующем требованиям, приведенным в пункте 3:

- 1) Развернуть ОС.
- 2) Создать кластер машин.
- 3) Установить Kubernetes версии не ниже 1.18.4.
- 4) Загрузить по ссылке <https://download.ec-leasing.ru/Peraspera/NeuroDoc/neuro-doc-images.tar.gz> и распаковать архив с дистрибутивом (набором манифестов Kubernetes) в любую удобную директорию;
- 5) Настроить манифесты Kubernetes, скорректировав при необходимости в них значения по умолчанию (см. Таблица 5).
- 6) Запустить инсталлятор в командной строке операционной системы с помощью следующей команды:

kubectl apply -f <папка с настроенными манифестами>

Таблица 4 – Значения по умолчанию настраиваемых параметров программы «Peraspera NeuroDoc»

№	Имя параметра	Значение параметра	Описание группы параметров
namespace.yml			
1	apiVersion	v1	Настройка пространства имен neuro
2	kind	Namespace	
3	metadata		
4	name	neuro	
5			
6	---		
7			
8	apiVersion	networking.k8s.io/v1	Настройка внешнего доступа к сервисам neuro-analyzer, domain-info, api
9	kind	Ingress	
10	metadata		
11	name	neuro-api	
12	namespace	neuro	
13	annotations		
14	nginx.ingress.kubernetes.io/proxy-body-size	"500m"	
15	nginx.ingress.kubernetes.io/use-regex	"true"	
16	nginx.ingress.kubernetes.io/rewrite-target	/\$3	
17	nginx.ingress.kubernetes.io/configuration-snippet		
18	if (\$request_method = 'OPTIONS') {		
19	add_header 'Access-Control-Allow-Origin' "\$http_origin";		
20	add_header 'Access-Control-Allow-Credentials' "true";		
21	add_header 'Access-Control-Allow-Headers' 'DNT,X-CustomHeader,Keep-Alive,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Content-Type,Authorization';		
22	return 204;		
23	}		
24	if (\$request_method != 'OPTIONS') {		

№	Имя параметра	Значение параметра	Описание группы параметров
25	add_header 'Access-Control-Allow-Origin' "\$http_origin" always;		
26	add_header 'Access-Control-Allow-Credentials' "true" always;		
27	}		
28	spec		
29	ingressClassName	nginx	
30	rules		
31	- host	neuro.ec-leasing.ru	
32	http		
33	paths		
34	- path	/(analyzer)(/ \$)(.*)	
35	pathType	Prefix	
36	backend		
37	service		
38	name	neuro-analyzer	
39	port		
40	number	8080	
41	- path	/(domain-info)(/ \$)(.*)	
42	pathType	Prefix	
43	backend		
44	service		
45	name	domain-info-service	
46	port		
47	number	5000	
48	- path	/(api-server)(/ \$)(.*)	
49	pathType	Prefix	
50	backend		
51	service		
52	name	api-server	
53	port		

№	Имя параметра	Значение параметра	Описание группы параметров
54	number	5000	
55			
56	---		
57			
58	apiVersion	networking.k8s.io/v1	Настройка внешнего доступа к сервису neuro-frontend
59	kind	Ingress	
60	metadata		
61	name	neuro	
62	namespace	neuro	
63	spec		
64	ingressClassName	nginx	
65	rules		
66	- host	neuro.ec-leasing.ru	
67	http		
68	paths		
69	- path	/	
70	pathType	Prefix	
71	backend		
72	service		
73	name	neuro-frontend	
74	port		
75	number	80	
76			
77	---		
78			
79	kind	Secret	
80	apiVersion	v1	
81	metadata		
82	name	nexus-docker-registry	

№	Имя параметра	Значение параметра	Описание группы параметров
83	namespace	neuro	
84	data		
85	.dockerconfigjson	>-	
86	eyJhdXRocyI6eyJkZXZvcHMuZWVhc2luZy5ydSI6eyJlc2VybmFtZSI6InBlcmFzcGVyYS1rOHMiLCJwYXNzd29yZCI6IjV0RlZyOUUyZWoiLCJhdXRoljoiY0dWeVIYTndaWEpoTFdzNGN6bzFkRVpXY2psRk1tMXEifX19		
87	type	kubernetes.io/dockerconfigjson	
88			
89	---		
90			
91	kind	Role	Создание роли в пространстве имен neuro
92	apiVersion	rbac.authorization.k8s.io/v1	
93	metadata		
94	name	neuro-admin-role	
95	namespace	neuro	
96	rules		
97	- apiGroups	["*"]	
98	resources	["*"]	
99	verbs	["*"]	
100			
101	---		
102			
103	apiVersion	rbac.authorization.k8s.io/v1	Привязки кластерной роли
104	kind	ClusterRoleBinding	
105	metadata		
106	name	neuro_admin_cluster_view	
107	roleRef		
108	apiGroup	rbac.authorization.k8s.io	

№	Имя параметра	Значение параметра	Описание группы параметров
109	kind	ClusterRole	
110	name	view	
111	subjects		
112	-apiGroup	rbac.authorization.k8s.io	
113	kind	Group	
114	name	neuro-admin	
115	- apiGroup	rbac.authorization.k8s.io	
116	kind	Group	
117	name	neuro-operator	
118			
119			
120	---		
121			
122	apiVersion	rbac.authorization.k8s.io/v1	Привязки кластерной роли
123	kind	ClusterRoleBinding	
124	metadata		
125	name	neuro_admin_cluster_nodes_view	
126	roleRef		
127	apiGroup	rbac.authorization.k8s.io	
128	kind	ClusterRole	
129	name	nodes-reader	
130	subjects		
131	- apiGroup	rbac.authorization.k8s.io	
132	kind	Group	
133	name	neuro-admin	
134			
135	---		
136			
137	apiVersion	rbac.authorization.k8s.io/v1	Привязки роли

№	Имя параметра	Значение параметра	Описание группы параметров
138	kind	RoleBinding	
139	metadata		
140	name	neuro_admin_role_binding	
141	namespace	neuro	
142	roleRef		
143	apiGroup	rbac.authorization.k8s.io	
144	kind	Role	
145	name	neuro-admin-role	
146	subjects		
147	- apiGroup	rbac.authorization.k8s.io	
148	kind	Group	
149	name	neuro-admin	
150			
151	---		
152			Привязки роли
153	apiVersion	rbac.authorization.k8s.io/v1	
154	kind	RoleBinding	
155	metadata		
156	name	neuro_operator_role_binding	
157	namespace	neuro	
158	roleRef		
159	apiGroup	rbac.authorization.k8s.io	
160	kind	Role	
161	name	view	
162	subjects		
163	- apiGroup	rbac.authorization.k8s.io	
164	kind	Group	
165	name	neuro-operator	

№	Имя параметра	Значение параметра	Описание группы параметров
neuro-postgresql.yml			
1	apiVersion	v1	Настройка внутреннего доступа к сервису PostgreSQL
2	kind	Service	
3	metadata		
4	name	postgresql	
5	namespace	neuro	
6	labels		
7	app	postgresql	
8	spec		
9	ports		
10	- port	5432	
11	name	postgresql	
12	selector		
13	app	postgresql	
14	sessionAffinity	None	
15			
16			
17	---		
18			
19	apiVersion	v1	Создание секрета для доступа к сервису PostgreSQL
20	kind	Secret	
21	metadata		
22	name	postgresql-secret	
23	namespace	neuro	
24	data		
25	pg_password	MXEwcDJ3OW8=	
26	pg_user	cG9zdGdyZXM=	
27			

№	Имя параметра	Значение параметра	Описание группы параметров
28	---		
29			
30	apiVersion	v1	Конфигурация сервиса PostgreSQL
31	kind	ConfigMap	
32	metadata		
33	name	postgresql-config	
34	namespace	neuro	
35	labels		
36	app	postgresql	
37	data		
38	pg_hba.conf		
39	local replication all peer		
40	host replication all 127.0.0.1/32 md5		
41	host replication all	: 1/128 md5	
42			
43	local all postgres ident		
44	local all all md5		
45	host all all 127.0.0.1/32 md5		
46	host all all	: 1/128 md5	
47	host all all 0.0.0.0/0 md5		
48	postgresql.conf		
49	data_directory = '/var/lib/postgresql/data/db'		
50	hba_file = '/etc/postgresql/13/main/pg_hba.conf'		
51	ident_file = '/etc/postgresql/13/main/pg_ident.conf'		
52	external_pid_file = '/var/run/postgresql/13-main.pid'		
53			
54	listen_addresses = '*'		
55	port = 5432		
56			

№	Имя параметра	Значение параметра	Описание группы параметров
57	max_connections = 300		
58	superuser_reserved_connections = 3		
59	unix_socket_directories = '/var/run/postgresql'		
60			
61	# - SSL -		
62	# ssl = off		
63	# ssl_cert_file = '/etc/ssl/certs/ssl-cert-snakeoil.pem'		
64	# ssl_key_file = '/etc/ssl/private/ssl-cert-snakeoil.key'		
65			
66	shared_buffers = 1GB		
67	work_mem = 50MB		
68	maintenance_work_mem = 1GB		
69	autovacuum_work_mem = 1GB		
70	dynamic_shared_memory_type = posix		
71	synchronous_commit = off		
72			
73	checkpoint_timeout = 10min		
74	max_wal_size = 1GB		
75	min_wal_size = 80MB		
76	checkpoint_completion_target = 0.9		
77			
78	effective_cache_size = 1GB		
79	log_line_prefix = '%m [%p] %q%u@%d '		
80	log_timezone = 'Europe/Moscow'		
81			
82	cluster_name = '13/main'		
83	stats_temp_directory = '/var/run/postgresql/13-main.pg_stat_tmp'		
84			
85			

№	Имя параметра	Значение параметра	Описание группы параметров
86	# - Locale and Formatting -		
87	datestyle = 'iso, dmy'		
88	timezone = 'Europe/Moscow'		
89	lc_messages = 'ru_RU.UTF-8'		
90	lc_monetary = 'ru_RU.UTF-8'		
91	lc_numeric = 'ru_RU.UTF-8'		
92	lc_time = 'ru_RU.UTF-8'		
93			
94	default_text_search_config = 'pg_catalog.russian'		
95			
96	include_dir = 'conf.d'		
97			
98	---		
99			
100	apiVersion	apps/v1	Параметры развертывания сервиса PostgreSQL
101	kind	StatefulSet	
102	metadata		
103	name	postgresql	
104	namespace	neuro	
105	spec		
106	serviceName	"neuro-dbus-pg"	
107	replicas	1	
108	selector		
109	matchLabels		
110	app	postgresql	
111	template		
112	metadata		
113	labels		
114	app	postgresql	

№	Имя параметра	Значение параметра	Описание группы параметров
115	spec		
116	imagePullSecrets		
117	- name	nexus-docker-registry	
118	containers		
119	- name	postgresql	
120	image	devops.ec-leasing.ru/postgis/postgis:13-03.1	
121	ports		
122	- containerPort	5432	
123	name	postgresql	
124	env		
125	- name	POSTGRES_USER	
126	valueFrom		
127	secretKeyRef		
128	name	postgresql-secret	
129	key	pg_user	
130	- name	POSTGRES_PASSWORD	
131	valueFrom		
132	secretKeyRef		
133	name	postgresql-secret	
134	key	pg_password	
135	- name	PGDATA	
136	value	/var/lib/postgresql/data/db	
137	volumeMounts		
138	- name	postgresql-data	
139	mountPath	/var/lib/postgresql/data	
140	- name	pg-config	
141	mountPath	/etc/postgresql/13/main/pg_hba.conf	
142	subPath	pg_hba.conf	
143	- name	pg-config	

№	Имя параметра	Значение параметра	Описание группы параметров
144	mountPath	/etc/postgresql/13/main/postgresql.conf	
145	subPath	postgresql.conf	
146	volumes		
147	- name	pg-config	
148	configMap		
149	name	postgresql-config	
150	volumeClaimTemplates		
151	- metadata		
152	name	postgresql-data	
153	spec		
154	accessModes	["ReadWriteOnce"]	
155	storageClassName	peraspera-blade-center-2-ssd	
156	resources		
157	requests		
158	storage	10Gi	
neuro-analyzer.yml			
1	kind	Service	Настройка внутреннего доступа к сервису Neuro-Analyzer
2	apiVersion	v1	
3	metadata		
4	name	neuro-analyzer	
5	namespace	neuro	
6	spec		
7	selector		
8	app	neuro-analyzer	
9	ports		
10	- port	8080	

№	Имя параметра	Значение параметра	Описание группы параметров
11	name	http	Создание секрета для доступа сервиса Neuro-Analyzer к сервису PostgreSQL
12			
13	---		
14			
15	apiVersion	v1	
16	kind	Secret	
17	metadata		
18	name	neuro-analyzer-secret	
19	namespace	neuro	
20	data		
21	db_user	bmV1cm9fdXNlcg==	Параметры развертывания сервиса Neuro-Analyzer
22	db_password	bmV1cm9fdXNlcg==	
23			
24	---		
25			
26	apiVersion	apps/v1	
27	kind	Deployment	
28	metadata		
29	name	neuro-analyzer	
30	namespace	neuro	
31	labels		
32	app	neuro-analyzer	
33	spec		
34	replicas	1	
35	selector		
36	matchLabels		
37	app	neuro-analyzer	
38	template		
39	metadata		

№	Имя параметра	Значение параметра	Описание группы параметров
40	labels		
41	app	neuro-analyzer	
42	spec		
43	imagePullSecrets		
44	- name	nexus-docker-registry	
45	containers		
46	- name	neuro-analyzer	
47	image	devops.ec-leasing.ru/neuro/neuro-analyzer:release	
48	imagePullPolicy	Always	
49	volumeMounts		
50	- name	configs	
51	mountPath	/config	
52	env		
53	- name	PROFILE	
54	value	production	
55	- name	TZ	
56	value	Europe/Moscow	
57	- name	DB_USER	
58	valueFrom		
59	secretKeyRef		
60	name	neuro-analyzer-secret	
61	key	db_user	
62	- name	DB_PASSWORD	
63	valueFrom		
64	secretKeyRef		
65	key	db_password	
66	name	neuro-analyzer-secret	
67	ports		

№	Имя параметра	Значение параметра	Описание группы параметров
68	- containerPort	8080	
69	name	http	
70	protocol	TCP	
71	volumes		
72	- name	configs	
73	configMap		
74	name	neuro-analyzer-config	

6 Информация, необходимая для эксплуатации

Эксплуатация программы «Peraspera NeuroDoc» осуществляется администратором с использованием программных продуктов, представленных в нижеприведенной таблице (Таблица 5).

Таблица 5 – Перечень продуктов для конфигурирования и мониторинга компонент программы «Peraspera NeuroDoc»

№	Продукт	Назначение
4	Kubernetes Dashboard	Конфигурирование и управление Kubernetes: <ul style="list-style-type: none"> - разворачивание контейнеризированных приложений в кластере Kubernetes; - конфигурирование специализированных ресурсов Kubernetes (Deployments, Pods, Replica Sets и др.) - мониторинг выполняющихся приложений; - мониторинг и управление ресурсами кластера
6	Keycloak	Управление подсистемой информационной безопасности <ul style="list-style-type: none"> - создание и конфигурирование ролей; - создание и конфигурирование пользователей; - управление доступом пользователей к приложениям.

Для эксплуатации программы «Peraspera NeuroDoc» с целью решения прикладных задач разрабатываются специализированные контейнеризируемые приложения, решающие конкретную задачу. Такие приложения могут быть написаны на любом языке программирования, поддерживающем контейнеризацию с использованием docker (например: Java, python, nodeJS). Типы приложений, разрабатываемых для исполнения в среде программы «Peraspera NeuroDoc», и используемые ими продукты, входящие в состав программы «Peraspera NeuroDoc», приведены в нижеприведенной таблице (Таблица 6).

Таблица 6 – Типы приложений, разрабатываемых для исполнения в среде программы «Peraspera NeuroDoc»

№	Тип приложений	Используемые продукты	Цель использования
1	Приложения для сбора данных из открытых источников	Keycloak	Аутентификация и авторизация
2	Аналитические сервисы	TensorFlow Serving	Взаимодействие с моделями машинного обучения для получения результатов аналитики
		Keycloak	Аутентификация и авторизация

№	Тип приложений	Используемые продукты	Цель использования
4	Графы Tensorflow для выполнения тензорных вычислений	TensorFlow Serving	Среда исполнения
5	Приложения для взаимодействия с пользователем	Keycloak	Аутентификация и авторизация

[illegible]