

## ПРОТОКОЛ № 15563/2023

### проведения совместных испытаний программного обеспечения «Монq» версии 7.11.0 и операционной системы специального назначения «Astra Linux Special Edition» РУСБ.10015-01 (очередное обновление 1.7)

г. Москва

06.12.2023

#### 1 Предмет испытаний

1.1 В настоящем протоколе зафиксирован факт проведения в период с 13.11.2023 по 06.12.2023 совместных испытаний программного обеспечения «Монq» версии 7.11.0 (далее – ПО), разработанного ООО «Монк диджитал лаб», и операционной системы специального назначения «Astra Linux Special Edition» РУСБ.10015-01 (очередное обновление 1.7) (далее – Astra Linux SE 1.7.0), разработанной ООО «РусБИТех-Астра», включая Astra Linux SE 1.7.0 с установленным оперативным обновлением безопасности БЮЛЛЕТЕНЬ № 2023-1023SE17 (оперативное обновление 1.7.5) (далее – Astra Linux SE 1.7.5).

#### 2 Объект испытаний

2.1 Перечень компонентов, эксплуатировавшихся в ходе проведения данных испытаний, относящихся к ПО, представлен в Таблице 1.

Таблица 1 – Перечень компонентов, относящихся к ПО

Описание	Наименование	Версия	Контрольная сумма	Источник
Docker-образ с ПО	pl-monq-agent	1.2.2	sha256:a78a0dfbff84806a5f81fe5f9fda0352123d44119f502cbdf4c4f3947c5418dd	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-identity-management-service	1.36.0	sha256:b0e3375f41a46db1913f9708b2948a1425ce686f3ea0e0cb01b953d661f81c81	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-frontend-service	2.33.1	sha256:c8ed86d7baa78d7ac42efaca635d1e374b2ecff94ec2ac79db e9887629e6c046	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-frontend-a-service	1.15.2	sha256:5a0519427a5de8f40e26f1f30e9cba84b2d3a222f04cb3ddc	Репозиторий разработчика - registry.monq.ru

			a4ac4865ff6d520	
Docker-образ с ПО	sm-rsm-maps-service	1.25.1	sha256:22ce2022ac8d9f25b799175a94550df9dc2178367356a1a53663312a0562d6d5	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-rsm-attributes-service	1.16.0	sha256:eac7de6629f090749c4624ac00e6c21df4d9083834cb774a4adce43aee520d53	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-v2-processor-service-compiler	1.10.2	sha256:9a44fc47d7a763cd40d034a91cee01f14f8d35df269213c7622068d184d4d041	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-v2-processor-service-runner	1.10.2	sha256:18c3ce144c97c41cb424e51c77536baa5c4badab7441ab3a02977d72c815d4a1	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-v2-processor-service	1.10.2	sha256:fbc97399f0fa9c84d5697c7c1552dbbb9008f2b6d7f47628aadaec6e334a019	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-microservice-registry-api-service	3.13.0	sha256:41b39b6f3a7eff45bded8e3e32d70e9c1a53e4ed16067f03540d784fb04b029d	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-frontend-b-service	1.3.1	sha256:a1616661f9aabb837aec3682de96aee68b6b27040fcb46de5b14365a5187cf9	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-signals-service	1.11.0	sha256:7837ea3fa2cfda1834114fc3853f14a2d6e16fab59f840c7621404b986a821cb	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-rsm-service	1.34.0	sha256:426e695e563526ab821dee01892813e1046e407c0c40994bf00896cf4d783723	Репозиторий разработчика - registry.monq.ru

Docker-образ с ПО	sm-gateway-api-service	4.53.0	sha256:ef5367a60ef6e3bfeb6715ab2651e12f889e9e5f4f0bcfb01ca9add430f5c1b9	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-identity-service	2.19.0	sha256:2e5a8b98508f3246d985c668326a61e88f641ffd531c634a9b4d1640f0c50d2a	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-v2-service-scheduler	1.18.0	sha256:76e12ae16d0c854843ee54573d21d4983d833f29cee1642970d449e8d3643800	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-v2-service-housekeeper	1.18.0	sha256:bbd5dba3f34cfccc53f6e9049c4875859e016c29ca46a14ec3b80f653878938b	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-v2-service	1.18.0	sha256:fea1e5d83f8942d88e898819455c335a0c0217b855790e323e593601233fe67a	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	mcs-frontend-service	2.21.0	sha256:7ec0de06abfcad6217316225eecb0987d53d7812290ab1be31a8bf1e2fbef198	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	cl-frontend-a-service	2.24.0	sha256:50605f5403dc5e9bf9947b4fcf9ca029edced9562592cf8c777f83373532d1db	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-rsm-mode-api-service-async	5.27.0	sha256:2d8814a6840121f77cb14bd525c2a8e8e70e0929d7de2f6ce59db8a8762bff02	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-rsm-mode-api-service	5.27.0	sha256:2fe272158b46cc0dccafeb329fa84f5b6373ca848897446a1d4a9bac1e1863c5	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-rsm-health-service-	1.18.2	sha256:6624227708b	Репозиторий разработчика - registry.monq.ru

	generator		5e054075cc8add7859 72e03ac4f55999109e 7d21f7b82236e2f14	ботчика - registry.monq.ru
Docker-образ с ПО	sm-rsm-health-service	1.18.2	sha256:becbeb68cbb9 2ac27a591ce2b98d47 619b973fad47692ccc 8b849e47513d4ea6	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-gateway-api-service	2.36.2	sha256:c19edc77053b 5f05c8404f3ec156ad3 2c98e95129ad9893b3 64365fa37ebd9c4	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	cl-streams-service-buffer	1.24.0	sha256:c425c765557c af4fff77e6f6521332bb 33e21ae970cffca4288 4c6f683a55e4f	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	cl-streams-service- watchdog	1.24.0	sha256:5059faef3689 e5dc5233d017202727 6ff77847e5aa8a05e2c 89549be9bfda244	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	cl-streams-service	1.24.0	sha256:56b61c16755 3d89ed00ef33d544b9 68480951b16956d6a5 a4a6584217436f616	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	plugins-zabbix-service	1.13.0	sha256:17c2a14e6151 84cdfd5674a5e66407 c6dea022e4a8c5a39b 3ffb0d1d05e40fa7	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-userspaces-api-service	1.30.0	sha256:a28b69d67d6 71add23aeec41ebcd1 e482372916b63799c7 b1d11e5abd2b0128d	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-router-service	1.25.0	sha256:43c036e9c96b 8fad519fa17b7407dc6 b1896571b3077ab321 86d35927f35fdc8	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-events-history-api- service-tags-api	2.19.0	sha256:404dc18816f6 2ded87d91bf9100264	Репозиторий разра- ботчика -

			affa337278642656d96 255eb8809d2ae84	registry.monq.ru
Docker-образ с ПО	pl-events-history-api- service-buffer	2.19.0	sha256:2ff5e8284722 ef317c9a02a5b126bee f334e98480eb7cc8a9 6c17d45a4aca92c	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-events-history-api- service	2.19.0	sha256:2f58b8e9fa8e 7800569641eba0a69c e0e4a8adeaab766c25 806a1d065ac5a805	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-content-installer-service	1.2.0	sha256:691c12b22b6 bd94e5e1abb5b6a86f df97bf888f8d66c867f 02e4ad11761ef254	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-agents-service-buffer	1.13.0	sha256:e723b9f6fb4d 351dce6bff418ce4564 2b1e5e6a121652be31 1c98eaad81db15a	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-agents-service- scheduler	1.13.0	sha256:4de9499fd15e 79ae36fda392f2b047c 304f6b9605e0b5ff578 6af079f64f7b4e	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-agents-service	1.13.0	sha256:e2dc9dea79b1 3fa2a8c1ffa2bfd626d 68881a2c298ade8160 79b8bb59a174161	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	frontend-box	2.112.0	sha256:28630e914cca a9de538c315bba5f29 edd0eed1f19923f5ca1 6fc328937ecdbad	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	sm-frontend-service	1.28.0	sha256:257dc9dc18bd 6352c77143139c7e61 3e0fcdfa1a558161fc1 6fce24b9374768b	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	sm-zabbix-connector-api- service-zabbix-api	2.25.0	sha256:6d044002da2 81eaaceb8efcaa1361a 0d5c07012bb6eb6cf6	Репозиторий разра- ботчика - registry.monq.ru

			45165782cde52c90	
Docker-образ с ПО	sm-zabbix-connector-api-service-autodiscovery	2.25.0	sha256:4db0af92b64f912afb8b12a7b9558dd2f915b9689e11cf8a86ddeb47867e756a	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-zabbix-connector-api-service	2.25.0	sha256:28acef36186ebda7e01dd819d047039b1add3b5f7bc8fad27eacaabb98620ee0	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-sla-service-ordered	1.31.0	sha256:a0d6d6d9fc3d2eab7b70250a476de7ea69a29121f4b45a96907eeda1d52d87df	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-sla-service	1.31.0	sha256:8daf5768e8175dee562ccd84c55c5ef5980aebb35ac9db54cb616428f1b07f4	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-action-generator-service-api	4.23.0	sha256:de24e2379b7729129fd9a31ab163ef824ccc8c092230beca07a55334cc80387c	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-action-generator-service	4.23.0	sha256:cdb70d0259a129fc35e33e680ff29787241faee9cf68f240a7e51e66fc655a43	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	mcs-gateway-service	1.17.0	sha256:d3a36dcb0aff8ffa19626dd97bb6cbb73b621d218f7c1c8779a6fc9766a64224	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	mcs-metrics-collector-service	1.4.0	sha256:18a2f6916fae3701e064ab9d5d65ee646941703d8c73d91de4470f16feca4043	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	mcs-thresholds-service-runner	1.5.0	sha256:560e7d1a23e1aee1a38e74c524845f2d93a4a1989f650b767b779a1c2c61c98f	Репозиторий разработчика - registry.monq.ru

Docker-образ с ПО	mcs-thresholds-service	1.5.0	sha256:ca1ef871365fe889c17951046500da4ec32a129b111128f1e39168c6ba0ec5d9	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	fm-frontend-a-service	1.15.0	sha256:af70db0dd0d5646b42f3ebd3bd9b26d6021d4908466b7d34d7562108e34c778f	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	fm-builds-aggregator-api-service	1.10.0	sha256:ba552b284498405a23ccaf0b77722d6f4cecd88d2f3bb89e3b874075fae2bdc5	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	cl-stream-data-service-housekeeper	1.24.0	sha256:c6705e5dcba2822ca486248746eb8bec28400e10368f21ac0b162751803865f3	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	cl-stream-data-service-buffer	1.24.0	sha256:783d240c1794165122545b6a8b19497ac656b02ac8f63eba56da351f2b2e4b70	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	cl-stream-data-service	1.24.0	sha256:c746d29eff84b26a7706ff1fb3e3b3143c81945b033412742b6c0389f9ccb5a	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	cl-gateway-service	1.23.0	sha256:6326c94f838ade2763ec44b8ea862ecc66b5b8d88aa5e328a49f4a05cfc1e989	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	fm-projects-api-service	1.19.0	sha256:5791f6d83ffcc499bec0e78da75ce02eac42fecb1da5c793cea43471631a7969	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	fm-gateway-api-service	1.21.0	sha256:ac2e62fc17911c855b1b5da25a3d76cfea81af04b13a35a3533ca376c8dd2a06	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	fm-data-housekeeper-	3.6.0	sha256:62ca76751c88	Репозиторий разработчика - registry.monq.ru

	service		aeb4c3f13b1fc218a3651da31e88dd1ad98a70cd28e7f458981b	ботчика - registry.monq.ru
Docker-образ с ПО	fm-builds-buffer-service	1.7.0	sha256:249250fc28a90ac46d5992407e24d1eb22c5b94ecf7c6346cf45a590fa5c4686	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-user-grants-api-service	3.24.0	sha256:6b7cbfa161a3e50662b84dd84498de85561f37cf51ed249bbe9d6d99fadbf8b	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-integral-events-configurator-api-service	3.12.0	sha256:efcfd9bce7adb61c0c12d83e21013eb6da4f3457df1e509fc88287f7c6845c05	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-i18n-service	1.5.0	sha256:5ebfc6f3e006e9890684eeca0c4979abf28b570ba9f6b61a5b0241d7487481ef	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-work-groups-api-service	3.21.0	sha256:fa9f51ebad2d875d3870e448b9cbcd b43376da4d3a2f5d2202a7d9847da50faf	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-user-shared-filters-api-service	2.9.0	sha256:181a22c216c9fdf42113d09da228578622ec473409b649829b94e8e4c85abe80	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-tags-api-service	1.12.0	sha256:11a7edde85b5c888284d5ec7b2a2cc72d86396f37a8352f4db8f05e07c3de411	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-delivery-lists-api-service	2.11.1	sha256:6220c977fe764402a4ff1e3e2f31362a29de7fcfa423458f3dedc6788f2441f6	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-plugins-manager-service	1.11.0	sha256:bf973ad290920b9ab7995ed67899b3	Репозиторий разработчика -



			5e34ca4e2f65c6f21abd24ea5af157e382	registry.monq.ru
Docker-образ с ПО	pl-user-groups-service-synchronizer	1.6.0	sha256:b04bb7894be52cd3b9a566e239f86bd69189e2017ebf87d3c14533050cd9abe4	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-user-groups-service	1.6.0	sha256:ba7d31c3cf374f8c75b9e33637e3e686a03fe34371e56176108c3f795acb60aa	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-business-calendar-api-service	1.7.0	sha256:74882e719a02bac8e321fafef4dddcf8a117493646cb376bbc7324208ebbb77	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	sm-rsm-attached-files-service	1.3.0	sha256:693c51cf4dc10f9f68b7826900b0d2827777c476a3ea266728bbc263f37044f3	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	cl-stream-schema-validator-service	1.9.0	sha256:9fa746fd96d6840ead8d095af3f223035759c5c342d89529920ebef7ef08bb01	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	cl-stream-data-collector-service	1.17.0	sha256:3dd454f4e9a151b5092234522e9f911afdc6b9f1f7b3ed4ec3adca78078fe626	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-rules-api-service-buffer	2.12.0	sha256:6e2ee83c1f32c54e6c707345017e043fe108877fbf2c8cbb3cd07253f8f4e6fa	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-rules-api-service	2.12.0	sha256:dd2e0ad11d0b2ad507874c88796086ab724fbac7c156aeb198b0a101ff43c382	Репозиторий разработчика - registry.monq.ru
Docker-образ с ПО	pl-license-service-storage	1.9.1	sha256:02fc7a756dd574aeca6785a13c42f25e7a7a0dd5593178e81	Репозиторий разработчика - registry.monq.ru

			3a2b18aba208141	
Docker-образ с ПО	pl-license-service-buffer	1.9.1	sha256:b22d631256e da85286152e626685e a44da635311a7c5bb4 4021765be72e241c4	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-license-service	1.9.1	sha256:f200412b06fc 8d3ca9264418a36a8c f24fe53e80732663df4 7449c35b2a0d087	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-scheduler- service	1.5.1	sha256:ca35d83fd461 9be570d8db309df53b 2af59ccab10b0c2273 b1721c96738da145	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-message-sender-api- service	2.9.1	sha256:a800ceb8fc41 bd9a851a2f130bc03e c4a30f60aceace195df ad97b2bc097d88f	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-macroses-api-service	1.8.0	sha256:88382e47744 1802331cd0a2051bbd a73a9a65b541af8594 d07cc4fad508ffbd2	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	pl-automaton-runner- service	3.8.0	sha256:b309f61d0c22 9d97d89f46466fb84fa 4de63b2f1793917473 50343cf12144f19	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ с ПО	installer	7.11.0	<none>	Репозиторий разра- ботчика - registry.monq.ru
Docker-образ допол- нительного про- граммного обеспече- ния необходимого для функционирова- ния ПО	registry	2.8.2	sha256:bcece5dd3d4b 6189e13e7ac71b2ccb c2aae649365f0c5898 52d687efeba6b290	hub.docker.com
Docker-образ допол- нительного про- граммного обеспече-	arangodb	3.11.2	sha256:8556a0b65ea1 402eb9e57325c4bdb3 d104712fea7cc5d2f37	hub.docker.com

ния необходимого для функционирования ПО			31e0fec2d1cbd93	
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	postgres	12.15	sha256:2cbb8fd81281d172de9ddaf07de35131a038ab182eaf5709084a2e5ccaae7306	hub.docker.com
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	clickhouse-server	23.3.8	sha256:926b84a4df36213aee9920081b83f8a18a09db64154fca8aebc72f534ecf49e9	yandex
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	redis	7.0.11	sha256:1008c73f08e9f913868e2fa2e843212b62ea5bf3c66435d87bc7a6207bc0f1b4	hub.docker.com
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	victoria-metrics	v1.91.3	sha256:5162ea9f497dab9b416008b11087beff2a8ca3544a3ba637c9a48ba650379a44	hub.docker.com
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	rabbitmq	3.11.18-management	sha256:abe4315e5be7b90060298c5f5cfe8d7e4feb614da3f1c416f453ecf3ac60fe69	hub.docker.com
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	consul	1.8.0	sha256:0e660ca8ae28d864e3eaaed0e273b2f8cd348af207e2b715237e869d7a8b5dcc	hub.docker.com

ния ПО				
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	ingress-nginx/controller	<none>	sha256:744ae2afd433a395eeb13dc03d3313facba92e96ad71d9feafcf85925493fee3	registry.k8s.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	ingress-nginx/kube-webhook-certgen	<none>	sha256:543c40fd093964bc9ab509d3e791f9989963021f1e9e4c9c7b6700b02bfb227b	registry.k8s.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	kube-apiserver	v1.23.17	sha256:46953dc9bc74e9a5dc86a994b3f7717805f9d0c690bafef78ab7af631b757decf	registry.k8s.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	kube-proxy	v1.23.17	sha256:9305d3c1859b77c6e4d7b42ac477f7a4f955883459f2656b3543d551d3245c83	registry.k8s.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	kube-controller-manager	v1.23.17	sha256:fd32cfc25e88bdecd9259b4d207f5ff9389e5b926ac19319af46731cd8469153	registry.k8s.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	kube-scheduler	v1.23.17	sha256:22a823de026e68b2661b0706efcbed4b026efa7e37a0de400251f8576f352ea5	registry.k8s.io
Дocker-образ допол-	etcd	3.5.6-0	sha256:dd75ec974b0a	registry.k8s.io

нительного программного обеспечения необходимого для функционирования ПО			2a6f6bb47001ba0920 7976e625db898d1b16 735528c009cb171c	
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	coredns/coredns	v1.8.6	sha256:5b6ec0d6de9b aaf3e92d0f66cd96a25 b9edbce8716f5f15dcd 1a616b3abd590e	registry.k8s.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	pause	3.6	sha256:3d380ca88645 49e74af4b29c10f9cb0 956236dfb01c40ca07 6fb6c37253234db	registry.k8s.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	cilium/cilium	<none>	sha256:06ce2b0a0a47 2e73334a7504ee5c5d 8b2e2d7b72ef728ad9 4e564740dd505be5	quay.io
Дocker-образ дополнительного программного обеспечения необходимого для функционирования ПО	cilium/operator-generic	<none>	sha256:b296eb7f0f76 56a5cc19724f40a8a7 121b7fd725278b7d61 dc91fe0b7ffd7c0e	quay.io
Официальное руководство по эксплуатации ПО	-	-	-	<a href="https://docs.monq.ru">https://docs.monq.ru</a>
ПО для развертывания кластера kubernetes	kubeadm	1.23.17	-	<a href="https://dl.k8s.io/release/v1.23.17/bin/linux/amd64/">https://dl.k8s.io/release/v1.23.17/bin/linux/amd64/</a>
По для управления кластером	kubectl	1.23.17	-	<a href="https://dl.k8s.io/release/v1.23.17/bin/">https://dl.k8s.io/release/v1.23.17/bin/</a>

kubernetes				linux/amd64/
Агент ноды kubernetes	kubelet	1.23.17	-	<a href="https://dl.k8s.io/release/v1.23.17/bin/linux/amd64/">https://dl.k8s.io/release/v1.23.17/bin/linux/amd64/</a>
Connection tracking system	contrack	1.4.6-2	-	<a href="http://ftp.ru.debian.org/debian/pool/main/c/contrack-tools/contrack_1.4.6-2_amd64.deb">http://ftp.ru.debian.org/debian/pool/main/c/contrack-tools/contrack_1.4.6-2_amd64.deb</a>
Плагины CNI	плагины CNI	v1.3.0	-	<a href="https://github.com/containernetworking/plugins/releases/download/">https://github.com/containernetworking/plugins/releases/download/</a>

### 3 Ход испытаний

3.1 В ходе проведения настоящих испытаний были выполнены проверки корректности функционирования ПО в средах: Astra Linux SE 1.7.0, Astra Linux SE 1.7.5, – в объеме, указанном в Приложении 1.

3.2 Перечень официальных репозиторий ПО, эксплуатировавшихся в упомянутых средах:

- в среде Astra Linux SE 1.7.0: base;
- в среде Astra Linux SE 1.7.5: main, update, base, extended.

3.3 Неофициальные репозитории ПО для указанных сред не эксплуатировались.

3.4 С целью проведения указанных проверок при включённом режиме замкнутой программной среды (далее – ЗПС) упомянутых ОС, использовался соответствующий файл открытого ключа, предоставленный стороной разработчика ПО.

3.5 Проверка корректности функционирования ПО в условиях ненулевого уровня конфиденциальности механизма мандатного разграничения доступа (далее – МРД) указанных сред не проводилась по причине отсутствия поддержки ПО соответствующей функциональности ОС. Информация об отсутствии упомянутой поддержки была заявлена стороной разработчика ПО.

3.6 Проверка ПО в непривилегированном режиме (rootless) для Astra Linux SE 1.7.0 не проводилась.

3.7 Проверка ПО на наличие уязвимости docker-образа и контейнера ПО для Astra Linux SE 1.7.0 не проводилась.

3.8 Для испытания 1.7.0 были скачаны пакеты компонентов не из официального репозитория по причине того, что в 1.7.0 отсутствует frozen репозитории с компонентами kubernetes: kubeadm, kubect1, kubelet, conntrack, плагины CNI.

#### 4 Результаты испытаний

4.1 ПО корректно функционирует в средах: Astra Linux SE 1.7.0, Astra Linux SE 1.7.5.

5 Вывод

5.1 ПО и операционная система специального назначения «Astra Linux Special Edition» РУСБ.10015-01 (очередное обновление 1.7) совместимы, принимая во внимание информацию, содержащуюся в разделах 3, 4 и Приложении 2.

6 Состав рабочей группы и подписи сторон

6.1 Данный протокол составлен участниками рабочей группы:

Герасимов Е.Е. - ведущий системный администратор ООО «Монк диджитал лаб».







## Инструкция по установке и удалению ПО в средах: Astra Linux SE 1.7.0, Astra Linux SE 1.7.5

### 1 Установка ПО:

#### 1.1 выполнить системные команды, действия:

Перечень системных команд, действий

Установка Monq условно разделена на два этапа, установка инфраструктуры и установка самого продукта.

Ввиду того, что для 1.7.0 и 1.7.5 пакеты kubernetes устанавливаются по-разному, здесь приведено две инструкции, для каждой версии соответственно.

Данный пункт имеет следующую структуру:

#### 1.1. Установка инфраструктуры

##### 1.1.1. Astra SE 1.7.0

##### 1.1.2. Astra SE 1.7.5

#### 1.2. Установка Monq.

##### 1.2.1. Astra SE 1.7.0

##### 1.2.2. Astra SE 1.7.5

### 1.1. Установка инфраструктуры

#### 1.1.1. Astra SE 1.7.0

Установить по инструкции, настроить сеть (`/etc/network/interfaces`) и dns (`/etc/resolv.conf`)

Скопировать папку manifests на целевую машину

```
``bash
```

```
server_ip="10.20.0.70"
```

```
scp -r test_plan/6_1/manifests/ u@${server_ip}:/home/u/
```

```
````
```

## Подготовка инфраструктуры

> Выполняется на машине с установленной Astra Linux

```
```bash
sudo apt update; sudo apt install -y ca-certificates jq curl
```

```bash
cat <<EOF | sudo tee /etc/modules-load.d/containerd.conf
overlay
br_netfilter
EOF

sudo modprobe br_netfilter
sudo modprobe overlay
```

```bash
cat <<EOF | sudo tee /etc/sysctl.d/99-kubernetes-cri.conf
net.bridge.bridge-nf-call-iptables = 1
net.ipv4.ip_forward = 1
net.bridge.bridge-nf-call-ip6tables = 1
fs.inotify.max_user_instances = 524288
net.ipv6.conf.all.disable_ipv6 = 1
EOF

sudo sysctl --system
```

```bash
sudo sed -i '/ swap / s/^\#/' /etc/fstab
sudo swapoff -a
```

```bash
cat <<EOF | sudo tee /etc/docker/daemon.json
{
```

```

"exec-opts": ["native.cgroupdriver=systemd"],
"insecure-registries" : ["registry.in.monq.local:5000"],
"log-driver": "json-file",
"log-opts": {
  "max-size": "10m",
  "max-file": "2"
}
}
EOF

```

```
sudo systemctl restart docker
```

```
...
```

Т.к. для 1.7.0 frozen отсутствует extended репозиторий, установка kubernetes будет выполнена согласно официальной документации

```

```bash
#https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-
kubeadm/#installing-kubeadm-kubelet-and-kubectl

CNI_PLUGINS_VERSION="v1.3.0"
ARCH="amd64"
DEST="/opt/cni/bin"
sudo mkdir -p "$DEST"
curl -L "https://github.com/container networking/plugins/releases/download/${CNI_PLUGINS_VERSION}/cni-plugins-linux-${ARCH}-${CNI_PLUGINS_VERSION}.tgz" |
sudo tar -C "$DEST" -xz

```

```
DOWNLOAD_DIR="/usr/local/bin"
```

```

RELEASE="v1.23.17"
cd $DOWNLOAD_DIR
sudo curl -L --remote-name-all https://dl.k8s.io/release/${RELEASE}/bin/linux/
amd64/{kubeadm,kubelet,kubectl}

```

```
sudo chmod +x {kubeadm,kubelet,kubectl}
cd -
```

```
RELEASE_VERSION="v0.16.2"
curl -sSL "https://raw.githubusercontent.com/kubernetes/release/${RELEASE_VERSION}/cmd/krel/templates/latest/kubelet/kubelet.service" | sed "s:/usr/bin:${DOWNLOAD_DIR}:g" | sudo tee /etc/systemd/system/kubelet.service
sudo mkdir -p /etc/systemd/system/kubelet.service.d
curl -sSL "https://raw.githubusercontent.com/kubernetes/release/${RELEASE_VERSION}/cmd/krel/templates/latest/kubeadm/10-kubeadm.conf" | sed "s:/usr/bin:${DOWNLOAD_DIR}:g" | sudo tee /etc/systemd/system/kubelet.service.d/10-kubeadm.conf
```

```
sudo systemctl enable kubelet
...
```

Установка зависимостей для kubelet

```
``bash
sudo apt install ethtool socat
wget http://ftp.ru.debian.org/debian/pool/main/c/conntrack-tools/conntrack_1.4.6-2_amd64.deb
sudo dpkg -i conntrack_1.4.6-2_amd64.deb
rm conntrack_1.4.6-2_amd64.deb
...
```

```
``bash
cat <<EOF | sudo tee -a /etc/hosts
127.0.0.1 k8s.in.monq.local
127.0.0.1 consul.in.monq.local
127.0.0.1 rabbitmq.in.monq.local
127.0.0.1 clickhouse.in.monq.local
127.0.0.1 postgresql.in.monq.local
127.0.0.1 registry.in.monq.local
127.0.0.1 redis.in.monq.local
```

```
127.0.0.1 arangodb.in.monq.local
127.0.0.1 victoriametrics.in.monq.local
EOF
```

```
sudo kubeadm init --control-plane-endpoint "k8s.in.monq.local:6443" \
--upload-certs \
--pod-network-cidr="10.244.0.0/16" \
--service-cidr="10.16.0.0/16" \
--skip-phases=addon/kube-proxy
...

```

```
```bash
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
sudo kubectl completion bash > ./kubectl
sudo mv kubectl /etc/bash_completion.d/kubectl
...

```

```
```bash
wget https://get.helm.sh/helm-v3.12.3-linux-amd64.tar.gz
tar -xvf ./helm-v3.12.3-linux-amd64.tar.gz
sudo mv ./linux-amd64/helm /usr/local/bin/
rm -rf ./helm-v3.12.3-linux-amd64.tar.gz ./linux-amd64
sudo helm completion bash > ./helm
sudo mv helm /etc/bash_completion.d/helm
...

```

```
```bash
helm repo add cilium https://helm.cilium.io/
helm upgrade --install cilium cilium/cilium --version 1.12.5 \
--namespace kube-system \
--set kubeProxyReplacement=strict \
--set k8sServiceHost=k8s.in.monq.local \

```

```

--set k8sServicePort=6443 \
--set operator.replicas=1 \
--set ipam.mode=kubernetes
...

```bash
kubectl label node $(hostname) ingress= function=worker postgresql= rabbitmq=
clickhouse= redis= arangodb= registry= consul= victoriametrics= storage=local
kubectl create ns infra
kubectl create ns production
sudo mkdir -p /storage/monq
kubectl create -f ./manifests/limit-range.yaml
kubectl create -f ./manifests/pvc.yaml
...

```bash
kubectl scale deploy -n kube-system coredns --replicas=1
kubectl create -f ./manifests/coredns-custom.yaml
kubectl get cm -n kube-system coredns -o=jsonpath='{.data.Corefile}' > Corefile
echo "import /etc/coredns/custom/*.server" >> Corefile
kubectl create cm -n kube-system coredns --from-file=./Corefile -o yaml --dry-
run=client | kubectl replace -f -
    kubectl get deploy -n kube-system coredns -o json | jq
'.spec.template.spec.volumes |= . + [{"configMap":{"defaultMode":420,"name":"coredns-
custom"},"optional":true},"name":"custom-config-volume"}] \
    | jq '.spec.template.spec.containers[0].volumeMounts |= . + [{"mountPath":"/etc/
coredns/custom","name":"custom-config-volume","readOnly":true}]' | kubectl replace -f -
...

```bash
kubectl create -f ./manifests/installer-rbac.yaml
kubectl create -f ./manifests/installer-rbac-coredns.yaml
kubectl create -f ./manifests/storage-class.yaml
kubectl taint node also node-role.kubernetes.io/master:NoSchedule-

```

```
...
```

```
```bash
```

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
```

```
helm upgrade --install ingress-nginx ingress-nginx/ingress-nginx --version 4.7.0 \
```

```
--namespace ingress-nginx --create-namespace \
```

```
--set controller.replicaCount=1 \
```

```
--set controller.hostPort.enabled=true \
```

```
--set controller.nodeSelector.ingress="" \
```

```
--set controller.service.type=ClusterIP \
```

```
--set controller.config.body-size=50m \
```

```
--set controller.config.hsts=false \
```

```
--set controller.config.large-client-header-buffers="4 32k" \
```

```
--set controller.config.proxy-body-size=50m \
```

```
--set controller.config.proxy-buffer-size=128k \
```

```
--set controller.config.proxy-buffers="4 256k" \
```

```
--set controller.config.proxy-busy-buffers-size=256k \
```

```
--set controller.config.proxy-connect-timeout="15" \
```

```
--set controller.config.proxy-read-timeout="300" \
```

```
--set controller.config.proxy-send-timeout="300" \
```

```
--set controller.config.server-name-hash-bucket-size="256" \
```

```
--set controller.config.worker-shutdown-timeout=10s
```

```
...
```

```
```bash
```

```
sudo mkdir /storage/registry
```

```
kubectl create -f ./manifests/registry.yaml
```

```
...
```

```
```bash
```

```
wget https://github.com/google/go-containerregistry/releases/download/v0.15.2/go-  
containerregistry_Linux_x86_64.tar.gz
```

```
tar -xvf go-containerregistry_Linux_x86_64.tar.gz
```



```

sudo mv ./crane /usr/local/bin/
rm ./go-containerregistry_Linux_x86_64.tar.gz ./LICENSE ./README.md ./
gcrane ./krane
...

```bash
crane copy postgres:12.15 registry.in.monq.local:5000/postgres:12.15 --insecure
crane copy arangodb:3.11.2 registry.in.monq.local:5000/arangodb:3.11.2 --insecure
crane copy redis:7.0.11 registry.in.monq.local:5000/redis:7.0.11 --insecure
crane copy rabbitmq:3.11.18-management registry.in.monq.local:5000/
rabbitmq:3.11.18-management --insecure
    crane copy clickhouse/clickhouse-server:23.3.8 registry.in.monq.local:5000/
clickhouse-server:23.3.8 --insecure
    crane copy consul:1.8.0 registry.in.monq.local:5000/consul:1.8.0 --insecure
    crane copy victoriametrics/victoria-metrics:v1.91.3 registry.in.monq.local:5000/
victoria-metrics:v1.91.3 --insecure
...

```bash
sudo mkdir /storage/postgresql
kubectl create -f ./manifests/postgresql.yaml
sudo mkdir /storage/arangodb
kubectl create -f ./manifests/arangodb.yaml
sudo mkdir /storage/redis
kubectl create -f ./manifests/redis.yaml
sudo mkdir /storage/rabbitmq
kubectl create -f ./manifests/rabbitmq.yaml
...

Дождаться запуска контейнера и создать пользователя
```bash
kubectl get po -n infra rabbitmq-0 -w

curl -X PUT -H "Content-Type: application/json" -u guest:guest \

```

```
http://rabbitmq.in.monq.local:15672/api/users/root -d \
{'password':"12345678","tags":"administrator"}
```

```
curl -X PUT -H "Content-Type: application/json" -u guest:guest \
http://rabbitmq.in.monq.local:15672/api/permissions/%2F/root -d \
{'configure':".*", "write":".*", "read":".*"}
```

```
curl -X DELETE -u root:12345678 http://rabbitmq.in.monq.local:15672/api/users/
guest
```

```
...
```

```
```bash
sudo mkdir /storage/clickhouse
kubectl create -f ./manifests/clickhouse.yaml
sudo mkdir /storage/consul
kubectl create -f ./manifests/consul.yaml
```

```
...
```

Дождаться запуска контейнера и выполнить инициализацию, передав токен в переменную.

```
```bash
kubectl get po -n infra consul-0 -w
```

```
CONSUL_TOKEN=$(curl -s -X PUT http://consul.in.monq.local:8500/v1/acl/
bootstrap | jq -r '.ID')
```

```
echo ${CONSUL_TOKEN}
```

```
...
```

```
```bash
sudo mkdir /storage/victoriametrics
kubectl create -f ./manifests/victoriametrics.yaml
sudo mkdir -p /opt/monq
kubectl create -f ./manifests/k8s-api-ingress.yaml
```

```
...
```

```

```bash
export CONSUL_TOKEN=${CONSUL_TOKEN}
export K8S_TOKEN=$(kubectl get secrets -n production installer-token -
o=jsonpath='{.data.token}' | base64 -d)
envsubst < ./manifests/system_auth.json.template | sudo tee /opt/monq/
system_auth.json
...

```

```

```bash
rm -rf ./manifests
...

```

### 1.1.2. Astra SE 1.7.5

Установить по инструкции, настроить сеть (`/etc/network/interfaces`) и dns (`/etc/resolv.conf`)

Скопировать папку manifests на целевую машину

```

```bash
server_ip="10.20.0.70"
sshpass -p "12345678" scp -r test_plan/6_1/manifests/ u@${server_ip}:/home/u/
...

```

## Подготовка инфраструктуры

> Выполняется на машине с установленной Astra Linux

```

```bash
sudo apt update; sudo apt install -y ca-certificates jq curl
...

```

```

```bash
cat <<EOF | sudo tee /etc/modules-load.d/containerd.conf
overlay
br_netfilter
EOF

```

```

sudo modprobe br_netfilter
sudo modprobe overlay
...

```bash
cat <<EOF | sudo tee /etc/sysctl.d/99-kubernetes-cri.conf
net.bridge.bridge-nf-call-iptables = 1
net.ipv4.ip_forward = 1
net.bridge.bridge-nf-call-ip6tables = 1
fs.inotify.max_user_instances = 524288
net.ipv6.conf.all.disable_ipv6 = 1
EOF

```

```

sudo sysctl --system
...

```

```

```bash
sudo sed -i '/ swap / s/^\#/' /etc/fstab
sudo swapoff -a
...

```

При обновлении системы в unit docker меняется драйвер по умолчанию на cgroupfs, а в процессе установки используется systemd.

По этому данную настройку следует изменить.

```

```bash
sudo mkdir -p /etc/systemd/system/docker.service.d
cat <<EOF | sudo tee /etc/systemd/system/docker.service.d/override.conf
[Service]
ExecStart=
ExecStart=/usr/sbin/dockerd -H fd:// \
  --containerd=/var/run/containerd/containerd.sock \
  --pidfile /var/run/docker/docker.pid \
  $DOCKER_OPTS

```

```
EOF
```

```
sudo systemctl daemon-reload
```

```
...
```

```
```bash
```

```
cat <<EOF | sudo tee /etc/docker/daemon.json
```

```
{
```

```
  "exec-opts": ["native.cgroupdriver=systemd"],
```

```
  "insecure-registries" : ["registry.in.monq.local:5000"],
```

```
  "log-driver": "json-file",
```

```
  "log-opts": {
```

```
    "max-size": "10m",
```

```
    "max-file": "2"
```

```
  }
```

```
}
```

```
EOF
```

```
sudo systemctl restart docker
```

```
...
```

```
```bash
```

```
sudo apt update; sudo apt install -y kubernetes-admin kubernetes-client
```

```
kubernetes-node
```

```
...
```

```
```bash
```

```
cat <<EOF | sudo tee -a /etc/hosts
```

```
127.0.0.1 k8s.in.monq.local
```

```
127.0.0.1 consul.in.monq.local
```

```
127.0.0.1 rabbitmq.in.monq.local
```

```
127.0.0.1 clickhouse.in.monq.local
```

```
127.0.0.1 postgresql.in.monq.local
```

```
127.0.0.1 registry.in.monq.local
```

```
127.0.0.1 redis.in.monq.local
127.0.0.1 arangodb.in.monq.local
127.0.0.1 victoriametrics.in.monq.local
EOF
```

```
sudo kubeadm init --control-plane-endpoint "k8s.in.monq.local:6443" \
--upload-certs \
--pod-network-cidr="10.244.0.0/16" \
--service-cidr="10.16.0.0/16" \
--skip-phases=addon/kube-proxy
...
```

```
``bash
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
sudo kubectl completion bash > ./kubectl
sudo mv kubectl /etc/bash_completion.d/kubectl
...
```

```
``bash
wget https://get.helm.sh/helm-v3.12.3-linux-amd64.tar.gz
tar -xvf ./helm-v3.12.3-linux-amd64.tar.gz
sudo mv ./linux-amd64/helm /usr/local/bin/
rm -rf ./helm-v3.12.3-linux-amd64.tar.gz ./linux-amd64
sudo helm completion bash > ./helm
sudo mv helm /etc/bash_completion.d/helm
...
```

```
``bash
helm repo add cilium https://helm.cilium.io/
helm upgrade --install cilium cilium/cilium --version 1.12.5 \
--namespace kube-system \
--set kubeProxyReplacement=strict \
```

```

--set k8sServiceHost=k8s.in.monq.local \
--set k8sServicePort=6443 \
--set operator.replicas=1 \
--set ipam.mode=kubernetes
...

```bash
kubectl label node $(hostname) ingress= function=worker postgresql= rabbitmq=
clickhouse= redis= arangodb= registry= consul= victoriametrics= storage=local
kubectl create ns infra
kubectl create ns production
sudo mkdir -p /storage/monq
kubectl create -f ./manifests/limit-range.yaml
kubectl create -f ./manifests/pvc.yaml
...

```bash
kubectl scale deploy -n kube-system coredns --replicas=1
kubectl create -f ./manifests/coredns-custom.yaml
kubectl get cm -n kube-system coredns -o=jsonpath='{.data.Corefile}' > Corefile
echo "import /etc/coredns/custom/*.server" >> Corefile
kubectl create cm -n kube-system coredns --from-file=./Corefile -o yaml --dry-
run=client | kubectl replace -f -
    kubectl get deploy -n kube-system coredns -o json | jq
'.spec.template.spec.volumes |= . + [{"configMap":{"defaultMode":420,"name":"coredns-
custom"},"optional":true},"name":"custom-config-volume"}]' \
    | jq '.spec.template.spec.containers[0].volumeMounts |= . + [{"mountPath":"/etc/
coredns/custom","name":"custom-config-volume","readOnly":true}]' | kubectl replace -f -
...

```bash
kubectl create -f ./manifests/installer-rbac.yaml
kubectl create -f ./manifests/installer-rbac-coredns.yaml
kubectl create -f ./manifests/storage-class.yaml

```

```
kubectl taint node else node-role.kubernetes.io/master:NoSchedule-  
...
```

```
```bash
```

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
```

```
helm upgrade --install ingress-nginx ingress-nginx/ingress-nginx --version 4.7.0 \
```

```
--namespace ingress-nginx --create-namespace \
```

```
--set controller.replicaCount=1 \
```

```
--set controller.hostPort.enabled=true \
```

```
--set controller.nodeSelector.ingress="" \
```

```
--set controller.service.type=ClusterIP \
```

```
--set controller.config.body-size=50m \
```

```
--set controller.config.hsts=false \
```

```
--set controller.config.large-client-header-buffers="4 32k" \
```

```
--set controller.config.proxy-body-size=50m \
```

```
--set controller.config.proxy-buffer-size=128k \
```

```
--set controller.config.proxy-buffers="4 256k" \
```

```
--set controller.config.proxy-busy-buffers-size=256k \
```

```
--set controller.config.proxy-connect-timeout="15" \
```

```
--set controller.config.proxy-read-timeout="300" \
```

```
--set controller.config.proxy-send-timeout="300" \
```

```
--set controller.config.server-name-hash-bucket-size="256" \
```

```
--set controller.config.worker-shutdown-timeout=10s  
...
```

```
```bash
```

```
sudo mkdir /storage/registry
```

```
kubectl create -f ./manifests/registry.yaml  
...
```

```
```bash
```

```
wget https://github.com/google/go-containerregistry/releases/download/v0.15.2/go-  
containerregistry_Linux_x86_64.tar.gz
```



```

tar -xvf go-containerregistry_Linux_x86_64.tar.gz
sudo mv ./crane /usr/local/bin/
rm ./go-containerregistry_Linux_x86_64.tar.gz ./LICENSE ./README.md ./
gcrane ./krane
...

```bash
crane copy postgres:12.15 registry.in.monq.local:5000/postgres:12.15 --insecure
crane copy arangodb:3.11.2 registry.in.monq.local:5000/arangodb:3.11.2 --insecure
crane copy redis:7.0.11 registry.in.monq.local:5000/redis:7.0.11 --insecure
crane copy rabbitmq:3.11.18-management registry.in.monq.local:5000/
rabbitmq:3.11.18-management --insecure
    crane copy clickhouse/clickhouse-server:23.3.8 registry.in.monq.local:5000/
clickhouse-server:23.3.8 --insecure
    crane copy consul:1.8.0 registry.in.monq.local:5000/consul:1.8.0 --insecure
    crane copy victoriametrics/victoria-metrics:v1.91.3 registry.in.monq.local:5000/
victoria-metrics:v1.91.3 --insecure
...

```bash
sudo mkdir /storage/postgresql
kubectl create -f ./manifests/postgresql.yaml
sudo mkdir /storage/arangodb
kubectl create -f ./manifests/arangodb.yaml
sudo mkdir /storage/redis
kubectl create -f ./manifests/redis.yaml
sudo mkdir /storage/rabbitmq
kubectl create -f ./manifests/rabbitmq.yaml
...

Дождаться запуска контейнера и создать пользователя
```bash
kubectl get po -n infra rabbitmq-0 -w

```

```
curl -X PUT -H "Content-Type: application/json" -u guest:guest \
http://rabbitmq.in.monq.local:15672/api/users/root -d \
'{"password":"12345678","tags":"administrator"}
```

```
curl -X PUT -H "Content-Type: application/json" -u guest:guest \
http://rabbitmq.in.monq.local:15672/api/permissions/%2F/root -d \
'{"configure":".*", "write":".*", "read":".*"}
```

```
curl -X DELETE -u root:12345678 http://rabbitmq.in.monq.local:15672/api/users/
guest
```

```
***
```

```
```bash
sudo mkdir /storage/clickhouse
kubectl create -f ./manifests/clickhouse.yaml
sudo mkdir /storage/consul
kubectl create -f ./manifests/consul.yaml
***
```

Дождаться запуска контейнера и выполнить инициализацию, передав токен в переменную.

```
```bash
kubectl get po -n infra consul-0 -w
```

```
CONSUL_TOKEN=$(curl -s -X PUT http://consul.in.monq.local:8500/v1/acl/
bootstrap | jq -r '.ID')
echo ${CONSUL_TOKEN}
***
```

```
```bash
sudo mkdir /storage/victoriametrics
kubectl create -f ./manifests/victoriametrics.yaml
sudo mkdir -p /opt/monq
kubectl create -f ./manifests/k8s-api-ingress.yaml
```

...

```
```bash
export CONSUL_TOKEN=${CONSUL_TOKEN}
export K8S_TOKEN=$(kubectl get secrets -n production installer-token -
o=jsonpath='{.data.token}' | base64 -d)
envsubst < ./manifests/system_auth.json.template | sudo tee /opt/monq/
system_auth.json
```

...

## 1.2. Установка Monq.

### 1.2.1. Astra SE 1.7.0

Выписать на сайте [monq.ru](https://monq.ru) токен обновлений, добавить в переменную

```
```bash
token="<токен с сайта>"
```

...

```
```bash
```

```
wget https://downloads.monq.ru/tools/monqctl/v1.12.0/linux-x64/monqctl.zip
unzip ./monqctl.zip
sudo mv ./monqctl /usr/local/bin/
rm ./monqctl.zip
```

```
monqctl config set instance astra.monq.ru --server=http://registry.api.astra.monq.ru
monqctl config set credential monq-user --registry-token=000
monqctl config set releasehub monq-release-hub --token=${token}
monqctl config set context astra.monq.ru --instance=astra.monq.ru --credential=monq-user
--releasehub=monq-release-hub
monqctl config use-context astra.monq.ru
monqctl release use-version 7.11.0 --product=monq
monqctl release import-images --registryUri http://registry.in.monq.local:5000 --
registryAuth=None
monqctl release use-version 3.13.0 --product=monq-registry
monqctl release import-images --registryUri http://registry.in.monq.local:5000 --
registryAuth=None
```

```

rm -rf /tmp/monqctl
...

```bash
monqctl installer version download 7.11.0 --token=${token}
sudo docker load -i installer*.img
rm -f installer*.img

echo "127.0.0.1 astra.monq.ru api.astra.monq.ru registry.api.astra.monq.ru" | sudo tee -a /
etc/hosts

sudo docker run --rm \
--name=monq-installer-temp \
--network=host \
-v /opt/monq:/opt/monq \
installer:7.11.0 \
ansible-playbook -e "global_domain='astra.monq.ru' dev=true demo_vm=true" monq/
monq.yaml
...

```

> После окончания установки будут выведены команды в консоль по настройке контекста monqctl, их надо выполнить.

Пример команд:

```

```bash
monqctl config set instance astra.monq.ru --server=http://domain
monqctl config set credential monq-user --registry-token=registryToken
monqctl config set releasehub monq-release-hub --token=YourUpdateToken
monqctl config set context astra.monq.ru --instance=astra.monq.ru --credential=monq-user
--releasehub=monq-release-hub

monqctl config use-context astra.monq.ru
...

```

### 1.2.2. Astra SE 1.7.5

Выписать на сайте monq.ru токен обновлений, добавить в переменную

```
```bash
```

```
token="<токен с сайта>"
```

```
```
```

```
```bash
```

```
wget https://downloads.monq.ru/tools/monqctl/v1.12.0/linux-x64/monqctl.zip
```

```
unzip ./monqctl.zip
```

```
sudo mv ./monqctl /usr/local/bin/
```

```
rm ./monqctl.zip
```

```
monqctl config set instance astra.monq.ru --server=http://registry.api.astra.monq.ru
```

```
monqctl config set credential monq-user --registry-token=000
```

```
monqctl config set releasehub monq-release-hub --token=${token}
```

```
monqctl config set context astra.monq.ru --instance=astra.monq.ru --
```

```
credential=monq-user --releasehub=monq-release-hub
```

```
monqctl config use-context astra.monq.ru
```

```
monqctl release use-version 7.11.0 --product=monq
```

```
monqctl release import-images --registryUri http://registry.in.monq.local:5000 --
```

```
registryAuth=None
```

```
monqctl release use-version 3.13.0 --product=monq-registry
```

```
monqctl release import-images --registryUri http://registry.in.monq.local:5000 --
```

```
registryAuth=None
```

```
rm -rf /tmp/monqctl
```

```
```
```

```
```bash
```

```
monqctl installer version download 7.11.0 --token=${token}
```

```
sudo docker load -i installer*.img
```

```
rm -f installer*.img
```

```
echo "127.0.0.1 astra.monq.ru api.astra.monq.ru registry.api.astra.monq.ru" | sudo
```

```
tee -a /etc/hosts
```

```
```
```

Запустить контейнер с установщиком в режиме ожидания:

```
```bash
sudo docker run -d \
--name=monq-installer-temp \
--network=host \
-v /opt/monq:/opt/monq \
installer:7.11.0 \
sleep 7200
```
```

При запуске контейнера выполняется проверка на уязвимости, это существенно увеличивает время запуска контейнеров.

Будет увеличен таймаут проверок запуска, в будущем будет добавлено как опция для установщика:

```
```bash
sudo docker exec -ti monq-installer-temp bash
scenarios_with_prechecks=$(grep -lr 'retries: 60')
for scenario in $scenarios_with_prechecks; do
  sed -i 's/retries\: 60/retries: 300/g' $scenario
done
exit
```
```

Запустить установку:

```
```bash
sudo docker exec -ti monq-installer-temp ansible-playbook \
-e "global_domain='astra.monq.ru' dev=true demo_vm=true" monq/monq.yaml
```
```

> После окончания установки будут выведены команды в консоль по настройке контекста monqctl, их надо выполнить.

Пример команд:

```

```bash
monqctl config set instance astra.monq.ru --server=http://domain
monqctl config set credential monq-user --registry-token=registryToken
monqctl config set releasehub monq-release-hub --token=YourUpdateToken
monqctl config set context astra.monq.ru --instance=astra.monq.ru --
credential=monq-user --releasehub=monq-release-hub
monqctl config use-context astra.monq.ru
...

```

## 2 Удаление ПО:

### 2.1 выполнить системные команды, действия:

Перечень системных команд, действий

#### 1. Остановить ПО:

```

```bash
sudo kubeadm reset -f
sudo systemctl disable kubelet --now
...

```

#### 2. Удалить монтируемый каталог с СПО и файловым хранилищем monq

```

```bash
sudo rm -rf /storage /opt/monq
...

```

#### 3. Удалить программное обеспечение и настройки:

##### 1. Если версия OS 1.7.0:

```

```bash
sudo rm -f $HOME/.kube/config /usr/local/bin/monqctl /usr/local/bin/crane /usr/
local/bin/helm
sudo docker images -q | xargs -l {} sudo docker rmi {}

sudo dpkg --purge socat conntrack ethtool jq

sudo rm -f /usr/local/bin/{kubeadm,kubelet,kubectl}
sudo rm -rf /home/u/.kube /etc/kubernetes /home/u/.cache/helm /home/u/.net \

```

```

/etc/bash_completion.d/helm \
/etc/bash_completion.d/kubectl \
/etc/cni/net.d/05-cilium.conf \
/etc/docker/daemon.json \
/etc/modules-load.d/containerd.conf \
/etc/sysctl.d/99-kubernetes-cri.conf \
/etc/sysctl.d/99-zzz-override_cilium.conf \
/etc/systemd/system/.service.d \
/etc/systemd/system/kubelet.service \
/etc/systemd/system/kubelet.service.d \
/opt/cni \
/root/.gnupg \
/var/lib/containerd/io.containerd.runtime.v2.task/moby \
/var/lib/docker/image/overlay2/distribution/diffid-by-digest \
/var/lib/docker/image/overlay2/distribution/v2metadata-by-diffid \
/var/log/pods \
/var/lib/docker/image/overlay2/layerdb/mounts \
/var/lib/docker/image/overlay2/layerdb/sha256 \
/var/lib/docker/image/overlay2/layerdb/tmp \
/var/lib/etcd \
/var/lib/kubelet \
/var/lib/cni \
/var/lib/containerd/io.containerd.content.v1.content/blobs \
/usr/libexec/kubernetes \
/var/lib/apt/lists/*

```

```
sudo exipick -i | xargs sudo exim -Mrm
```

```
...
```

6. Если версия OS 1.7.5:

```
```bash
```

```
sudo rm -f $HOME/.kube/config /usr/local/bin/monqctl /usr/local/bin/crane /usr/
```

```
local/bin/helm
```

```
sudo docker images -q | xargs -l {} sudo docker rmi {}
```

```
sudo dpkg --purge kubernetes-admin kubernetes-client kubernetes-node
```



```
sudo dpkg --purge socat contrack ethtool jq containernetworking-plugins
ebtables
```

```
sudo rm -f /usr/bin/kubelet /lib/systemd/system/kubelet.service
sudo rm -rf /home/u/.kube /etc/kubernetes /home/u/.cache/helm /home/u/.net \
/var/lib/containerd/io.containerd.runtime.v2.task/moby \
/var/lib/docker/image/overlay2/distribution/diffid-by-digest \
/var/lib/docker/image/overlay2/distribution/v2metadata-by-diffid \
/var/log/pods \
/var/lib/docker/image/overlay2/layerdb/mounts \
/var/lib/docker/image/overlay2/layerdb/sha256 \
/var/lib/docker/image/overlay2/layerdb/tmp \
/var/lib/etcd \
/var/lib/kubelet \
/var/lib/cni \
/var/lib/containerd/io.containerd.content.v1.content/blobs \
/opt/cni \
/usr/libexec/kubernetes \
/var/lib/apt/lists/* \
/etc/systemd/system/docker.service.d \
/etc/bash_completion.d/helm \
/etc/bash_completion.d/kubectl \
/etc/cni/net.d/05-cilium.conf \
/etc/docker/daemon.json \
/etc/modules-load.d/containerd.conf \
/etc/sysctl.d/99-kubernetes-cri.conf \
/etc/sysctl.d/99-zzz-override_cilium.conf \
/etc/systemd/system/.service.d \
/root/.gnupg
```

```
sudo exipick -i | xargs sudo exim -Mrm
```

```
^^^
```

**Перечень используемых сокращений и определений**

«Руководство по КСЗ Ч. 1» – документ «Операционная система специального назначения «Astra Linux Special Edition». Руководство по КСЗ. Часть 1» РУСБ.10015-01 97 01-1;

Astra Linux SE 1.7.0 – операционная система специального назначения «Astra Linux Special Edition» РУСБ.10015-01 (очередное обновление 1.7);

Astra Linux SE 1.7.5 – операционная система специального назначения «Astra Linux Special Edition» РУСБ.10015-01 (очередное обновление 1.7) с установленным оперативным обновлением безопасности БЮЛЛЕТЕНЬ № 2023-1023SE17 (оперативное обновление 1.7.5);

ДВиС – дирекция внедрения и сопровождения;

ЗПС – замкнутая программная среда ОС;

КСЗ – комплекс средств защиты;

МКЦ – мандатный контроль целостности ОС;

МРД – мандатное управление доступом ОС;

ОС – операционная система;

ПО – программное обеспечение «Монр» версии 7.11.0;

Docker - программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации;

Docker-образ – неизменяемый образ по шаблону которого создается docker-контейнер;

Docker-контейнер – контейнер созданный на основе docker-образа;

Контейнер – изолированная среда с упакованным кодом и зависимостями.