

ПРОТОКОЛ № 30781/2025
проведения совместных испытаний программного обеспечения "SI3000 ЕНМ
Приложение оператора Системы-112" версии 2025.09.05 и операционной системы
специального назначения "Astra Linux Special Edition" РУСБ.10015-01
(очередное обновление 1.7)

г. Москва

09.12.2025

1 Предмет испытаний

1.1 В настоящем протоколе зафиксирован факт проведения совместных испытаний в период с 08.12.2025 по 09.12.2025 программного обеспечения "SI3000 ЕНМ Приложение оператора Системы-112" версии 2025.09.05 (далее – ПО), разработанного АО "Искра Технологии", и операционной системы специального назначения "Astra Linux Special Edition" РУСБ.10015-01 (очередное обновление 1.7) с установленным оперативным обновлением безопасности БЮЛЛЕТЕНЬ № 2025-0923SE17 (оперативное обновление 1.7.8) (далее – Astra Linux SE 1.7.8), разработанной ООО "РусБИТех-Астра".

2 Объект испытаний

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

Таблица 1 – Дистрибутив и документация ПО

Наименование файла	Контрольная сумма (MD5)	Ссылка на эксплуатационную документацию
system112-2025.09.05.7z	ebf7d835f97bfdc671f9b9ee5e960a21	-
OnlineHelp.zip	ca4c8150f691c3d806d6914b6fc6f4bc	

3 Ход испытаний

3.1 В ходе проведения настоящих испытаний были выполнены проверки ПО в среде Astra Linux SE 1.7.8 в объеме, указанном в Приложении 1.

3.2 Перечень используемых репозиторий приведен в Приложении 2.

3.3 Для функционирования ПО в среде операционной системы с активным режимом ЗПС внедрение электронной цифровой подписи в дистрибутив не требуется. Зафиксировано создание временных файлов в каталоге "/tmp/jna-3506402", которые влияют на функционирование ПО, но не могут быть подписаны в связи с их автоматическим удалением.

3.4 В рамках испытаний проводилось тестирование компонентов "FiasService", "ЕНМ", "OnlineHelp", "LogService", "ClientCoffee", "ClientReact", "Monitoring", "Symmetric".



3.5 При установке и эксплуатации ПО вносятся изменения в конфигурационный файл параметров ядра, что ограничивает функционирование механизма защиты "Защита ядра". Подробнее в РКСЗ раздел 9.4 и МР п. 2.5.

3.6 Проверка корректности функционирования ПО с уровнем конфиденциальности 1-3 механизма мандатного разграничения доступа не проводилась по причине отсутствия поддержки ПО соответствующей функциональности ОС. Факт отсутствия упомянутой поддержки был определён стороной ООО "РусБИТех-Астра".

4 Вывод

4.1 "SI3000 ЕНМ Приложение оператора Системы-112" версии 2025.09.05 функционирует в среде операционной системы специального назначения "Astra Linux Special Edition" РУСБ.10015-01 (очередное обновление 1.7) уровень защищенности "Базовый" и признано совместимым, принимая во внимание информацию, содержащуюся в разделе 3.

5 Состав рабочей группы и подписи сторон

5.1 Данный протокол составлен участниками рабочей группы:

Проканюк Д. С. – начальник сектора ООО "РусБИТех-Астра";

Дончук А. И. – инженер ООО "РусБИТех-Астра".



Перечень проверок совместимости ПО и Astra Linux SE 1.7.8

Таблица 1.1 - Результаты проверок ПО

Перечень проверок	Версия ядра	Статус механизмов безопасности в процессе выполнения проверки		
	6.1.141-1-generic	ЗПС	МКЦ	МРД
Установка ПО	Успешно	Неактивен	Активен	Активен
Эксплуатация ПО	Успешно	Неактивен		
Удаление ПО	Успешно	Неактивен		
Эксплуатация ПО. Уровень конфиденциальности 1-3	Не проводилась	Неактивен		



Инструкция по установке и удалению ПО

1 Используемые репозитории в Astra Linux SE 1.7.8:

- deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.8/repository-base/ 1.7_x86-64
main contrib non-free non-free-firmware
- deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.8/repository-update/ 1.7_x86-64
main contrib non-free non-free-firmware
- deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.8/repository-extended/ 1.7_x86-64
main contrib non-free non-free-firmware

2 Установка ПО:

Выполнить следующие команды:

```
sudo apt update
sudo apt install -y nginx aspnetcore-runtime-6.0 openjdk-17-jdk postgresql-11
postgresql-contrib postgresql-11-postgis-3
echo "postgres:12345678" | sudo chpasswd
sudo pdpl-user -i 63 postgres
sudo usermod -a -G shadow postgres
sudo setfacl -d -m u:postgres:r /etc/parsec/macdb
sudo setfacl -R -m u:postgres:r /etc/parsec/macdb
sudo setfacl -m u:postgres:rx /etc/parsec/macdb
sudo -u postgres psql -tc "ALTER USER postgres WITH PASSWORD '12345678';"
7z x system112-2025.09.05.7z
```

1) Подготовка базы данных для FiasService.

Загрузить базу данных ФИАС с ресурса "<https://fias.nalog.ru/Frontend>" (выбрать полную версию)

Создать базу данных и пользователя:

```
sudo -u postgres psql -c "CREATE USER fias WITH PASSWORD '12345678';"
sudo -u postgres psql -c "CREATE DATABASE fias_database OWNER fias;"
sudo useradd -r -s /usr/sbin/nologin fias
sudo pdpl-user -l 0:0 fias
cd ./GarParser/FiasGarDatabaseParser
```



В файл "appsettings.json" внести нужные значения параметров "Regioncode", "RegionCodeForLoading"; для параметра "FiasDirectory" указать полный путь до каталога с распакованной базой данной ФИАС; внести строку "ConnectionString":

```
"Host=<Ip-адрес FIAS>;Port=5432;Username=fias;Password=12345678;
Database=fias_database"
```

Запустить установку базы данных:

```
dotnet FiasGarDatabaseParser.dll
```

2) Установка FiasService

```
sudo mkdir -p /opt/fias-service
```

```
sudo cp -r ./FiasService/* /opt/fias-service
```

```
sudo groupadd fias-service
```

```
sudo useradd -s /bin/false -g fias-service -d /opt/fias-service fias-service
```

```
sudo chown -R fias-service:fias-service /opt/fias-service
```

```
sudo mkdir -p /var/log/iut/fias-service
```

```
sudo chown -R fias-service:fias-service /var/log/iut/fias-service
```

В файл "/opt/fias-service/EnvironmentConfigs/appsettings.Production.json" внести следующие строки:

```
"FiasDbType": "Pg",
```

```
"FiasPgConnectionString": "Host=<IP-адрес FIAS>;Port=5432;Username=fias;
Password=12345678;Database=fias_database"
```

Создать файл "/etc/systemd/system/fias-service.service", внести в него следующее содержимое:

```
[Unit]
```

```
Description=.NET Web API FiasService running on Astra
```

```
[Service]
```

```
WorkingDirectory=/opt/fias-service
```

```
ExecStart=/usr/bin/dotnet /opt/fias-service/FiasApiService.dll --urls
http://0.0.0.0:8200
```

```
Restart=always
```

```
RestartSec=10
```

```
KillSignal=SIGINT
```

```
SyslogIdentifier=fias-service
```



User=fias-service

Group=fias-service

Environment=ASPNETCORE_ENVIRONMENT=Production

[Install]

WantedBy=multi-user.target

Выполнить следующие команды:

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable fias-service.service
```

```
sudo systemctl start fias-service.service
```

3) Установка EHM

```
sudo mkdir -p /opt/ehm
```

```
sudo cp -r ./EHM/EHM/* /opt/ehm
```

```
sudo mkdir /opt/ehm/EnvironmentConfigs/License/
```

```
sudo cp ./license.lic /opt/ehm/EnvironmentConfigs/License/
```

```
sudo groupadd ehm
```

```
sudo useradd -s /bin/false -g ehm -d /opt/ehm ehm
```

```
sudo chown -R ehm:ehm /opt/ehm
```

```
sudo mkdir -p /var/log/iut/ehm
```

```
sudo chown -R ehm:ehm /var/log/iut/ehm
```

```
sudo pdpl-user -l 0:0 ehm
```

Создать базу данных и выполнить восстановление:

```
sudo -u postgres psql -c "CREATE DATABASE ehm_database WITH OWNER
postgres;"
```

```
cp ./EhmDbTemplate/PostgreSql/ehm_template_v257_20241127.sql /tmp
```

```
sudo su - postgres -c "psql -d ehm_database -f
/tmp/ehm_template_v257_20241127.sql"
```

В файл "/opt/ehm/EnvironmentConfigs/ehmSettings.Production.config" внести следующие секции:

```
<!-- EHM Database settings -->
```

```
<add key="PostgresConnectionString" value="Host=<IP-адрес
EHM>;Port=5432;Username=postgres;Password=12345678;Database=ehm_database" />
<add key="IsPostgres" value="true" />
```



```
<add key="CommandTimeoutSec" value="30" />
```

```
<!-- FIAS settings -->
```

```
<add key="FiasServiceHost" value="<IP-адрес FIAS>"/>
```

```
<add key="FiasServicePort" value="8200" />
```

```
<add key="FiasServiceRequestTimeoutMs" value="1000" />
```

```
<add key="FiasRegionCode" value="<код региона>" />
```

```
<!-- Дополнительные порты EHM -->
```

```
<add key="WebSocketServerPort" value="3000" />
```

```
<!-- Naumen-specific settings -->
```

```
<add key="NccIp" value="<IP-адрес Naumen Contact Center>"/>
```

```
<add key="NccPort" value="3242"/>
```

```
<add key="NccVersion" value="1000"/>
```

```
<add key="NccLogin" value="ehm"/>
```

```
<add key="NccHa1" value="158fe4334f99a746c9ebea7b0cbcf7f8"/>
```

```
<add key="NccTimeoutSec" value="14" />
```

```
<add key="NccReconnectIntervalSec" value="10" />
```

```
<add key="NccKeepAliveIntervalSec" value="5" />
```

```
<!-- IsActive -->
```

```
<add key="IsEhmActiveCheckType" value="NccConnection" /> <!--
```

```
NccConnection, IskraCrmTest, FloatingIp -->
```

```
<add key="IpCheckIntervalSec" value="2" />
```

```
<add key="FloatingIp" value="" />
```

Значения параметров "NccLogin" и "NccHa1" можно узнать из файла "/opt/naumen/nauphone/cfg/naucore/key.service.ehm.xml" на хосте с установленным "Naumen Contact Center".

Необходимо удалить из файла неиспользуемые секции, если таковые имеются.

В файл "/opt/ehm/EnvironmentConfigs/appsettings.Production.json" внести записи "IP-адрес:порт" для всех хостов, со страниц которых разрешено обращение к данному API; убрать секции для LRS и EIIM, если данные компоненты не используются.

Создать файл "/etc/systemd/system/ehm.service", внести в него следующее содержимое:



[Unit]

Description=.NET Web API Ehm running on Astra

[Service]

WorkingDirectory=/opt/ehm

ExecStart=/usr/bin/dotnet /opt/ehm/Ehm.dll --urls http://0.0.0.0:8080

Restart=always

RestartSec=10

KillSignal=SIGINT

SyslogIdentifier=ehm

User=ehm

Group=ehm

Environment=ASPNETCORE_ENVIRONMENT=Production

Environment=DOTNET_PRINT_TELEMETRY_MESSAGE=false

[Install]

WantedBy=multi-user.target

Выполнить следующие команды:

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable ehm.service
```

```
sudo systemctl start ehm.service
```

4) Установка OnlineHelp

Выполнить следующие команды:

```
sudo mkdir -p /opt/online-help/OnlineHelp
```

```
sudo unzip ./OnlineHelp.zip -d /opt/online-help/OnlineHelp/
```

```
sudo chown -R www-data:www-data /opt/online-help
```

В файл "/etc/nginx/conf.d/default.conf" внести следующее содержимое:

```
server {
    listen 8084;
    server_name localhost;

    access_log /var/log/nginx/OH.log;
    error_log /var/log/nginx/OH.log;
```




```
rewrite_log on;
```

```
root /opt/online-help/OnlineHelp;
```

```
index Index112.html index.html index.htm;
```

```
location / {
```

```
    try_files $uri $uri/ /Index112.html;
```

```
}
```

```
}
```

Выполнить команду:

```
sudo nginx -s reload
```

5) Установка LogService

Выполнить следующие команды:

```
sudo mkdir /opt/clientlog-service
```

```
sudo cp -r ./LogService/* /opt/clientlog-service/
```

```
sudo groupadd clientlog-service
```

```
sudo useradd -s /bin/false -g clientlog-service -d /opt/clientlog-service clientlog-  
service
```

```
sudo chown -R clientlog-service:clientlog-service /opt/clientlog-service/
```

```
sudo mkdir -p /var/log/iut/clientlog-service
```

```
sudo chown -R clientlog-service:clientlog-service /var/log/iut/clientlog-service
```

Создать файл "/etc/systemd/system/clientlog-service.service", внести в него следующее содержимое:

```
[Unit]
```

```
Description=.NET Web API ClientLogService running on Astra
```

```
[Service]
```

```
WorkingDirectory=/opt/clientlog-service/ClientLogService/
```

```
ExecStart=/usr/bin/dotnet
```

```
/opt/clientlog-service/ClientLogService/ClientLogService.dll --urls http://0.0.0.0:8089
```

```
Restart=always
```

```
RestartSec=10
```

```
KillSignal=SIGINT
```



SyslogIdentifier=clientlog-service

User=clientlog-service

Group=clientlog-service

Environment=ASPNETCORE_ENVIRONMENT=Production

[Install]

WantedBy=multi-user.target

В файле "/opt/clientlog-service/ClientLogService/EnvironmentConfigs/appsettings.Production.json" указать значение "<IP-адрес LogService>:8089" для параметра "Hosts".

Выполнить следующие команды:

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable clientlog-service.service
```

```
sudo systemctl start clientlog-service.service
```

6) Установка ClientCoffee

```
sudo mkdir /opt/client-coffee
```

```
sudo cp -r ./ClientCoffee/* /opt/client-coffee/
```

```
sudo chown -R www-data:www-data /opt/client-coffee/
```

В файл "/opt/client-coffee/dist/config/config.js" внести изменения в следующие строки:

```
apiUrl: 'http://<IP-адрес EHM>:8080',
```

```
wsUrl: 'ws://<IP-адрес EHM>:3000',
```

```
logUrl: 'http://<IP-адрес LogService>:8089/log_msg',
```

```
knowledgeUrl: 'http://<IP-адрес OnlineHelp>:8084/OnlineHelp/Index112.html'
```

В файл "/etc/nginx/conf.d/default.conf" внести следующее содержимое:

```
server {
    listen 8082;
    server_name localhost;

    location / {
        root /opt/client-coffee/dist;
        index index.html index.htm;
        try_files $uri $uri/ /index.html;
    }
}
```



```
}
```

Выполнить команду:

```
sudo nginx -s reload
```

7) Установка ClientReact

```
sudo mkdir /usr/share/nginx/client-react
```

```
sudo cp -r ./ClientReact/* /usr/share/nginx/client-react/
```

В файле "/usr/share/nginx/client-react/dist/config/config.json" внести изменения в следующие строки:

```
"api": "<IP-адрес ЕНМ>:8080",
```

```
"socketURL": "<IP-адрес ЕНМ>:3000",
```

```
"knowledgeUrl": "http://<IP-адрес OnlineHelp>:8084/OnlineHelp/Index112.html",
```

```
"CTIPhoneConnectUrl": "ws://<IP-адрес Naumen Contact Center>:8098",
```

```
"CTIApiUrl": "ws://<IP-адрес Naumen Contact Center>:8098",
```

```
"logSettings": {
```

```
  "logUrl": "http://<IP-адрес LogService>:8089/log_msg"
```

В файл "/etc/nginx/conf.d/default.conf" внести следующее содержимое:

```
server {
```

```
    listen 8083;
```

```
    server_name localhost;
```

```
    location / {
```

```
        root /usr/share/nginx/client-react/dist;
```

```
        index index.html index.htm;
```

```
        try_files $uri $uri/ /index.html;
```

```
    }
```

```
}
```

Выполнить команду:

```
sudo nginx -s reload
```

8) Установка Monitoring

```
sudo mkdir /opt/monitoring
```

```
sudo cp -r ./Monitoring/* /opt/monitoring
```

```
sudo chown -R www-data:www-data /opt/monitoring
```



В файле "/opt/monitoring/build/config/config.json" указать IP-адрес и порт ЕНМ.

В файл "/etc/nginx/conf.d/default.conf" внести следующее содержимое:

```
server {
    listen 8085;
    server_name localhost;

    location / {
        root /opt/monitoring/build;
        index index.html index.htm;
        try_files $uri $uri/ /index.html;
    }
}
```

Выполнить команду:

```
sudo nginx -s reload
```

9) Установка Symmetric

```
sudo mkdir /opt/sym
```

```
sudo groupadd sym
```

```
sudo useradd -s /bin/false -g sym -d /opt/sym sym
```

```
sudo chown -R sym:sym /opt/sym/
```

```
sudo cp -r ./symmetric/* /opt/sym/
```

```
sudo /opt/sym/bin/sym_service install
```

В файле "/etc/init.d/SymmetricDS" указать параметру "RUN_AS_USER" значение "root".

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable SymmetricDS
```

```
sudo systemctl start SymmetricDS
```

Создать файл "/etc/systemd/system/sym-gui.service", внести в него следующее содержимое:

```
[Unit]
```

```
Description=sym ds gui
```

```
After=network.target
```

```
[Service]
```



```
Environment=JAVA_HOME=/usr/
Environment=SYM_HOME=/opt/sym/
WorkingDirectory=/opt/sym/bin/
```

```
ExecStart=/usr/bin/java -cp ${SYM_HOME}/patches:${SYM_HOME}/patches/*:${SYM_HOME}/lib/*:${SYM_HOME}/web/WEB-INF/lib/*:${SYM_HOME}/web/gui-lib/*
si.iskratel.service112.dbsync.mn.SymdsGui
```

```
User=sym
Group=sym
UMask=0007
RestartSec=30
Restart=always
```

[Install]

```
WantedBy=multi-user.target
```

В файле "/opt/sym/conf/symds-gui.properties" указать параметру "path.app" значение "/opt/sym/" и параметру "path.webapp" значение "/opt/sym/web/webapp".

Выполнить следующие команды:

```
sudo systemctl daemon-reload
sudo systemctl enable sym-gui
sudo systemctl start sym-gui
```

Дополнительная информация по установке ПО содержится в файле "system112_install_guide_2025.09.05.docx".

3 Удаление ПО:

```
sudo systemctl stop ehm clientlog-service sym-gui fias-service
sudo systemctl disable ehm clientlog-service sym-gui fias-service
sudo rm -r /opt/client-coffee /opt/clientlog-service /opt/ehm /opt/fias-service
/opt/monitoring /opt/online-help /opt/sym /usr/share/nginx/client-react
sudo rm -r /var/log/iut/fias-service /var/log/iut/ehm /var/log/iut/clientlog-service
```



Приложение 3 к Протоколу № 30781/2025

Перечень используемых сокращений

Astra Linux SE 1.7.8 - операционная система специального назначения "Astra Linux Special Edition" РУСБ.10015-01 (очередное обновление 1.7) с установленным оперативным обновлением безопасности БЮЛЛЕТЕНЬ № 2025-0923SE17 (оперативное обновление 1.7.8);

РКСЗ - Документ из состава эксплуатационной документации Astra Linux SE 1.7.8, Руководство по КСЗ. Часть 1;

КСЗ - комплекс средств защиты;

ОС - операционная система;

ЗПС - замкнутая программная среда;

МКЦ - мандатный контроль целостности;

МРД - мандатное управление доступом;

ПО - программное обеспечение "SI3000 ЕНМ Приложение оператора Системы-112" версии 2025.09.05.

Идентификатор документа 675c2ca2-2527-4920-a578-3a9fa435a288

Документ подписан и передан через оператора ЭДО АО «ПФ «СКБ Контур»

Подписи отправителя:		Доверенность: рег. номер, период действия и статус		Сертификат: серийный номер, период действия	Дата и время подписания
ООО "РУСБИТЕХ-АСТРА" Проканюк Дмитрий Сергеевич		Не приложена при подписании		058F6B830091B36D914AE938D BF2830C31 с 10.11.2025 10:48 по 10.11.2026 10:48 GMT+03:00	10.12.2025 15:49 GMT+03:00 Подпись соответствует файлу документа