Интеграция Брест со службой каталога ALD Pro

06/10/2025

Содержание

ПК СВ "Брест" - российская платформа со встроенными средствами защиты информации серверной ОС Astra Linux Special Edition для создания и управления облачными виртуальными ИТ-инфраструктурами.

Интеграция позволяет обеспечить аутентификацию доменных пользователей в веб-интерфейсе системы Брест по безопасному протоколу Kerberos V5.

В настоящей инструкции описана процедура интеграции Брест со службой каталога ALD Pro.

1 Описание стенда

В тестовой среде установлены:

- домен ALD Pro с именем ALD.COMPANY.TEST (версия продукта ALD Pro 2.1.0, версия ОС ALSE 1.7.4);
- домен MS Active Directory с именем WIN.TEST (версия Windows Server 2016);
- Брест 3.2 с ОС 1.7.2.

В таблице 1 приведены полные имена машин и их IP-адреса:

Таблица 1 - Имена машин и их IP-адреса

dc-1.ald.company.test	10.0.61.10/24
dc.win.test	10.0.62.10/24
brest-mgt.ald.company.test (фронтальная машина)	10.0.61.30/24

2 Ввод в домен машины с Брест

В текущих репозиториях для Брест 3.2 (ОС Astra1.7.2) недоступны пакеты для aldpro-client, salt. Родной способ ввода клиентов в домен ALD Pro выполняется с помощью утилиты aldpro-client-installer, входящей в состав пакета aldpro-client. Salt - важный компонент ALD Pro для настройки рабочих станций и распространения групповых политик. Поскольку пакет aldpro-client недоступен в репозиториях, ввод в домен ALD Pro выполняется командой astra-freeipa-client с дополнительными ключами:

```
astra-freeipa-client -d domain -u admin -p password --par “--enable-dns-updates --
mkhomedir --force-join”.
```

3 Установка сервисов фронтальной машины

После установки пакета brestcloud-ipa требуется запустить bash-скрипт brestcloud-configure для настройки фронтальной машины. Скрипт brestcloud-configure создает две группы brestadmins и brestusers в нескольких местах: на фронтальной машине, в домене и во внутренней базе пользователей Брест. Группы brestadmins и brestusers являются являются важными для взаимодействия Брест и ALD Pro, их нельзя удалять. Включение доменных пользователей в одну из групп brestadmins или brestusers позволяет им получить доступ к веб-порталу Брест. Дополнительно создаются служебные группы kvm, libvirt, libvirt-qemu, libvirt-admin на фронтальной машине и в домене. В качестве сервера базы данных на фронтальной машине используется PostgreSQL, в базе хранится информация о пользователях и группах.

Кроме того, скриптом в домене добавляется роль “Brestusers Administrator”, привилегия “Manage users – brestusers” и разрешение “Manage brestusers”. Разрешение “Manage brestusers” дает право управления составом участников группы brestusers. Привилегия “Manage users – brestusers” включает в себя несколько разрешений: “Manage brestusers”, “System: Add Users”, “System: Change User password”, “System: Modify Group Memberships”. Роль “Brestusers Administrator” содержит в себе привилегию “Manage users – brestusers” и назначается группе brestadmins. Таким образом пользователи группы brestadmins обладают расширенными правами в домене - они могут управлять участниками всех групп в домене, кроме admins, менять пароли пользователям, исключая тех, кто находится в группе admins. При этом добавлять новых пользователей в домене они не могут.

Скрипт также предлагает создать доменного пользователя, который будет добавлен в группы admins, brestadmins, brestusers. Пользователь будет обладать правами администратора в домене, но не на серверах Брест, так как его уровень целостности меньше 127 и равен 0. Дополнительно на фронтальной машине создается локальный пользователь brestadmin. Он является локальным администратором, так как входит в группу astra-admin и имеет уровень целостности 127.

```
root@brest-mgt:~# id brestadmin
uid=9870(brestadmin) gid=9869(oneadmin) группы=9869(oneadmin),6(disk),333(astra-
console),1001(astra-admin),240400004(brestadmins)
```

На фронтальной машине и на других серверах Брест уровень целостности пользователя, обладающего повышенными правами, должен быть равен 127. Поэтому администратор admin домена ALD Pro не является полноценным администратором, так как его уровень целостности равен 63. Чтобы дать повышенные права пользователю домена, ему необходимо поднять уровень целостности до 127 и включить его в локальную группу astra-admin. По умолчанию, только brestadmin и локальный пользователь, созданный при установке системы, являются администраторами. Пользователь brestadmin включен в группу brestadmins, но так как он не доменный и его нет во внутренней базе пользователей Брест, он не может войти в веб-интерфейс Брест.

Чтобы проверить какие мандатные метки присвоены пользователям, используется команда pdpl-user. Локальный пользователь user был создан при установке ОС, он является администратором системы. В выводе команд уровень “Высокий” равен 127.

```
root@brest-mgt:~# pdpl-user admin
минимальная метка: Уровень_0:Низкий:Нет:0x0
   0:0:0x0:0x0
максимальная метка: Уровень_0:63:Нет:0x0
   0:63:0x0:0x0

root@brest-mgt:~# pdpl-user brestadmin
минимальная метка: Уровень_0:Низкий:Нет:0x0
   0:0:0x0:0x0
максимальная метка: Уровень_0:Высокий:Нет:0x0
```

```
   0:127:0x0:0x0

root@brest-mgt:~# pdpl-user user
минимальная метка: Уровень_0:Низкий:Нет:0x0
   0:0:0x0:0x0
максимальная метка: Уровень_0:Высокий:Нет:0x0
   0:127:0x0:0x0
root@brest-mgt:~#
```

В отличие от доменных рабочих станций ALD Pro, на серверах Брест пользователям группы astra-admin при запуске команд с помощью sudo вводить пароль не требуется:

```
%astra-admin  ALL=(ALL:ALL) NOPASSWD: ALL
```

4 Проверка аутентификации пользователей домена ALD Pro в веб интерфейсе Брест

Для доступа к веб порталу Брест используется Kerberos-аутентификация. На фронтальной машине установлен веб-сервер apache2, он принимает входящие запросы и после аутентификации перенаправляет пользователя на внутренний веб-сервер Брест (puma). Модуль mod_auth_kerb в apache2 отвечает за аутентификацию пользователей по протоколу Kerberos. Apache2 может использовать Basic метод для проверки пользователя, в этом случае пользователь вводит свой логин и пароль в браузере для входа в веб-интерфейсе Брест. Apache2 использует введенные данные для аутентификации пользователя на KDC (Key Distribution Center) и проверки того, что он имеет доступ к веб-серверу. Если на стороне пользователя браузер настроен в режиме прозрачной аутентификации (SSO), пользователю не нужно вводить логин и пароль, apache2 использует метод Negotiate для проверки пользователя. Браузер и apache2 с помощью механизма SPNEGO проводят аутентификацию пользователя для доступа к веб-серверу. При этом в сессии пользователя должен быть заранее получен Kerberos-билет.

Apache2 проксирует запросы на веб-сервер Брест puma, который запущен на порту 9869. Порт 9869 доступен на всех интерфейсах фронтальной машины, и, если не настроен фаервол, доступ к нему открыт для всех, значит с любой машины, имеющей сетевую связность с фронтальной машиной, можно попасть на страницу внутреннего веб сервера Брест, минуя Kerberos-аутентификацию. Веб сервер puma запускается systemd-файлом сервиса opennebula-sunstone.service. Конфигурационный файл puma находится по пути /usr/lib/one/sunstone/puma.rb.

После того, как пользователь прошел проверку подлинности в apache2, он попадает на страницу внутреннего веб-сервера Брест (puma) (рис. 1).

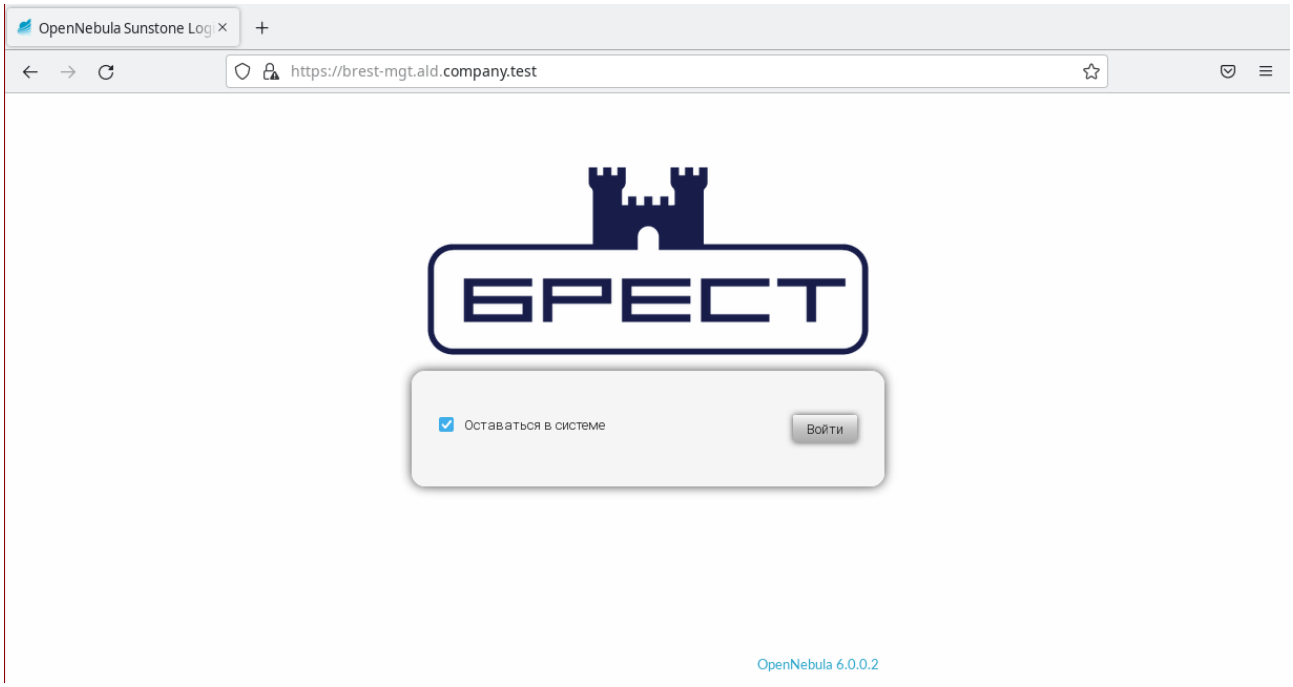


Рисунок 1 – Вход в веб-портал системы Брест

Причем во время загрузки данных с разных URL веб-сервера Брест аутентификация пользователя для доступа к этим URL проходит каждый раз, так как apache2 в данный момент не сохраняет информацию о предыдущем процессе аутентификации.

После нажатия кнопки “Войти” происходит поиск пользователя во внутренней базе Брест. Если он не найден, запускается скрипт /usr/lib/one/sh/check_user.sh, который создает пользователя в Брест. После этого откроется главная страница веб интерфейса пользователя (рис. 2).

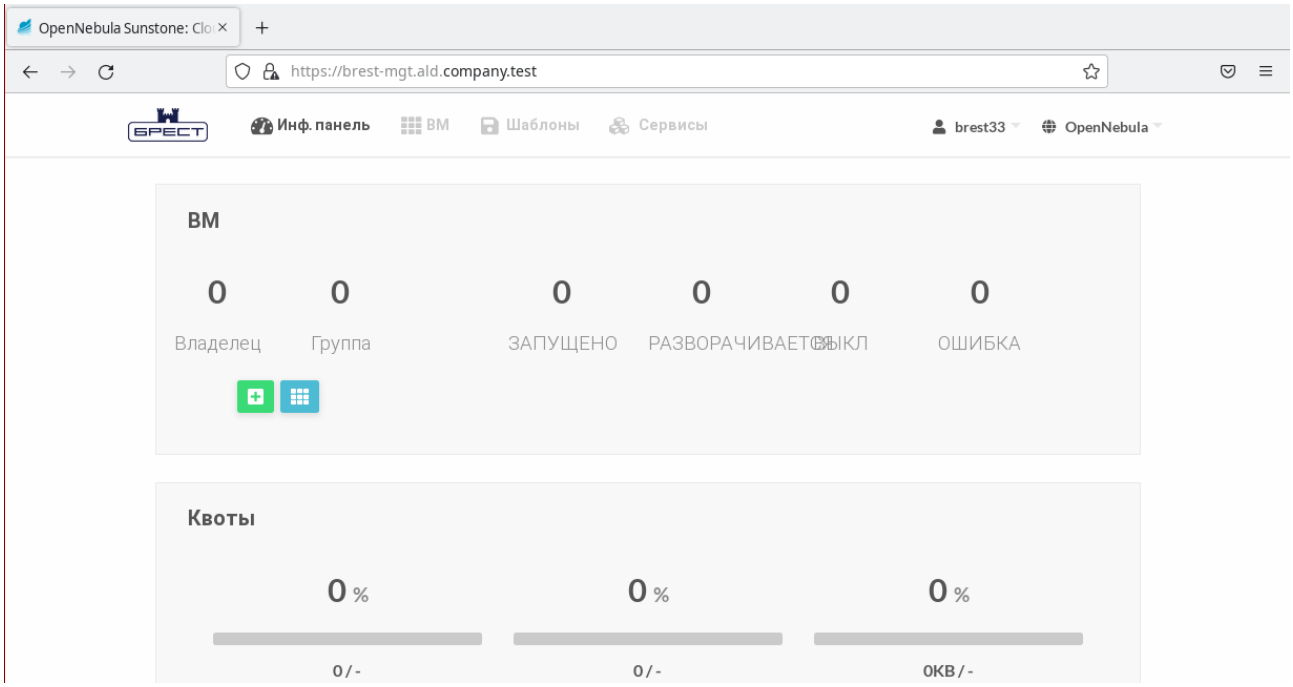


Рисунок 2 - Главная страница веб-интерфейса пользователя

Скрипт /usr/lib/one/sh/check_user.sh предназначен для автоматического создания пользователей в базе пользователей Брест при их входе в веб-интерфейс. Это относится и к пользователям домена Windows, имеющего установленные доверительные отношения с доменом ALD Pro. Пользователь домена ALD Pro должен быть включен в группу brestusers или в группу brestadmins. Часть скрипта, отвечающая за определение того, что пользователь включен в нужные группы, приведена ниже:

```
if [[ "$AD" == "false" ]]; then
    username=${1%@*}
    if getent group brestadmins | grep -q "\b${username}\b"; then
        oneuser create ${username} "$1" --driver public --group brestadmins
        if [[ "$?" == "0" ]]; then
            check_token ${username}
        fi
    else
        if getent group brestusers | grep -q "\b${username}\b"; then
            oneuser create ${username} "$1" --driver public --group brestusers
            if [[ "$?" == "0" ]]; then
                check_token ${username}
            fi
        fi
    fi
fi
```

Из имени пользователя username удаляется доменная часть, затем командой getent group brestadmins или getent group brestusers проверяется, что пользователь является членом группы. Если это верно, в Брест создается пользователь, иначе вход на веб-интерфейс Брест не будет успешным. Но даже если пользователь состоит в группах, команда getent может вернуть отрицательный результат. Это связано с работой службы sssd, она кэширует записи пользователей, групп и считает записи кэша действительными в течение определенного времени (по умолчанию 5400 секунд, параметр entry_cache_timeout). После истечения этого интервала времени кэш автоматически обновляется, и информация о группах и пользователях становится актуальной. Уменьшение entry_cache_timeout - это возможный вариант для решения данной проблемы, но все равно останется какой-то временной промежуток, когда пользователь не сможет войти в Брест, и к тому же частое обновление кэша может увеличить нагрузку на LDAP-сервер ALD Pro. Еще один обходной вариант состоит в том, чтобы в

скрипте удалять кэш перед созданием пользователя в Брест командой sss_cache -g brestusers, sss_cache входит в состав пакета sssd-tools, который не установлен по умолчанию. Другой способ исправления проблемы - это использовать команду id username, даже если пользователь не найден в кэше службы sssd, актуальная информация о пользователе запрашивается на LDAP-сервере. Но у этого решения тоже есть недостаток: если пользователь уже находится в кэше и на момент попадания в кэш он не состоял в группе brestusers или brestadmins, информация о нем обновится спустя интервал entry_cache_timeout. Если его добавить в необходимые группы, он не сразу сможет войти в веб-интерфейс Брест. Есть еще вариант решения - использовать прямые запросы к LDAP-серверу для определения участия пользователя в группе brestusers.

5 Создание доменных пользователей в Брест

Пользователи, созданные в веб-интерфейсе Брест, автоматически создаются в домене ALD Pro. Функционал реализован с помощью cgi-скрипта /usr/lib/one/brestcloud/manage-user.cgi. Если при создании пользователя не снимать галочку “Сменить пароль при первом входе в систему”, то срок жизни пароля нового пользователя домена устанавливается равным 365 дней. Ниже приведен фрагмент кода скрипта:

```
plyear = datetime.now() + timedelta(days=365)
krbpwexp = plyear.strftime('%Y%m%d%H%M%S') + "Z"

user_md = dict(krbpasswordexpiration=krbpwexp)
api.Command.user_mod(str(username), **user_md)
```

Согласно политикам по умолчанию для паролей Kerberos в ALD Pro, срок жизни пароля должен составлять три месяца. В Брест при создании пользователя нет проверки сложности задаваемого пароля, поэтому, если снята галочка “Сменить пароль при первом входе в систему”, то, независимо от настроек политик Kerberos, созданный пользователь будет в течение года иметь небезопасный пароль.

Новый пользователь в домене ALD Pro попадает в корень иерархии подразделений и прикреплять его к конкретному подразделению необходимо будет вручную.

При удалении пользователя из Брест он остается в домене. В скрипте manage-user.cgi определена функция user_del_ipa(), она не удаляет пользователя, а исключает его из всех групп, в которых он был в Брест. После удаления пользователя из Брест он не сможет входить на портал Брест:

```
def user_del_ipa():
...
    if "1" in udict.get('summary'):
        for grp in groups:
            #grp = grp.decode("utf-8")
            igrp = api.Command.group_find(str(grp, 'utf-8'))
            if "1" in igrp.get('summary'):
                api.Command.group_remove_member(cn=str(grp, 'utf-8'), user=username)
...
```

Однако на практике пользователь остается в тех группах, в которых он был изначально. Для примера: пользователь brest1 был удален в Брест и команда ipa была запущена после его удаления. Пользователь brest1 остался в ALD Pro и состав его групп не изменился (выделены ниже в выводе команды):

```
root@dc-1:~# ipa user-show brest1
  Имя учётной записи пользователя: brest1
  Имя: brest1
  Фамилия: brest1
  Домашний каталог: /home/brest1
  Оболочка входа: /bin/bash
  Имя учётной записи: brest1@ALD.COMPANY.TEST
  Псевдоним учётной записи: brest1@ALD.COMPANY.TEST
  Адрес электронной почты: brest1@ald.company.test
  UID: 240400010
  ID группы: 240400003
  Учётная запись отключена: False
```

```
  Link to department:
ou=ald.company.test,cn=orgunits,cn=accounts,dc=ald,dc=company,dc=test
  Link to head department: ald.company.test
  Пароль: True
  Участник групп: brestusers, ipausers
  Indirect Member of group: kvm, libvirt-qemu, libvirt
  Indirect Member of role: Organization units
  Доступные ключи Kerberos: True
```

Как и для пользователей, скрипт /usr/lib/one/brestcloud/manage-group.cgi запускается при создании групп в Брест, он автоматически их добавляет в домене ALD Pro. Созданные группы попадают в корень иерархии подразделений, если ее необходимо прикрепить к конкретному подразделению, это придется сделать вручную. Если удалить группу в Брест, она успешно удаляется в ALD Pro, но при удалении группы в ALD Pro, она останется в Брест. Расхождение в группах может возникнуть, если по какой-либо причине в момент создания группы в Брест не было связи с доменом ALD Pro, так как не происходит синхронизации групп в фоновом режиме.

Изменения участия пользователя в группах в веб-интерфейсе Брест успешно отражаются на составе групп пользователя в ALD Pro, это при условии, что группы в Брест существуют в ALD Pro. Но в обратном направлении это неверно, если произошли изменения списка групп пользователя в ALD Pro, они не отразятся в Брест. Таким образом информация о пользователях с той и с другой стороны может отличаться.

В скриптах manage-user.cgi и manage-group.cgi взаимодействие с API FreeIPA происходит с учетной записью пользователя, созданного при настройке Брест командой brestcloud-configure.

6 Добавление дополнительных групп для пользователей в Брест

По умолчанию, все пользователи ALD Pro, впервые авторизовавшиеся в веб-интерфейсе Брест, добавляются в группу brestusers или brestadmins. Может возникнуть необходимость автоматически включить пользователя в дополнительные группы Брест. Однако пользователям, хотя бы раз заходившим на портал Брест, дополнительные группы назначаются вручную.

Если групп еще нет, их необходимо создать через веб-интерфейс Брест (рис. 3).

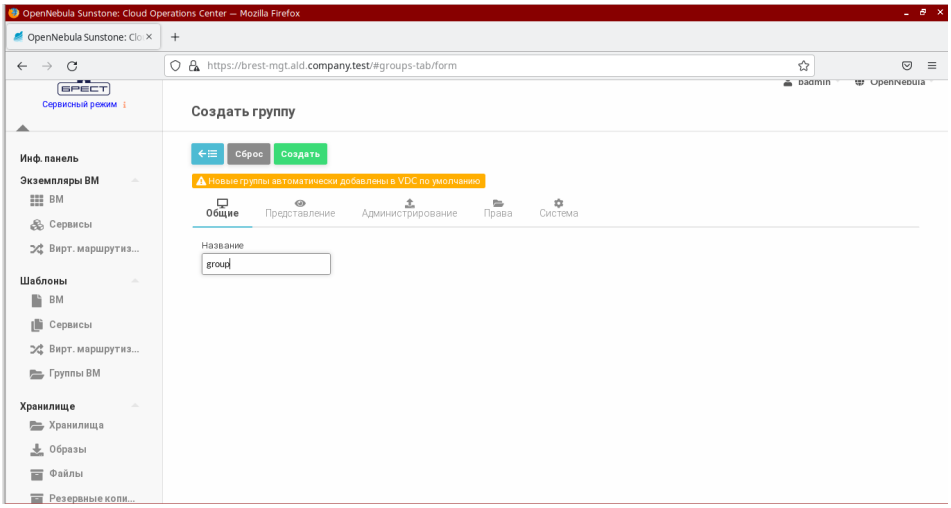


Рисунок 3 – Создание группы через веб-интерфейс Брест

Группа должна автоматически появиться в ALD Pro. На портале ALD Pro пользователь включается в новую группу и выполняет вход на портал Брест.

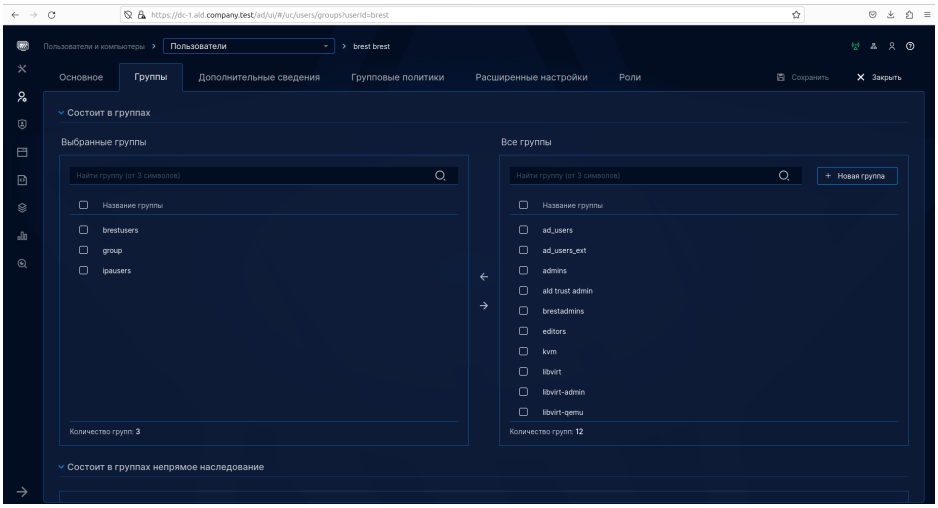


Рисунок 4 – Просмотр на портале управления ALD Pro списка групп, участником которых является пользователь brest

Добавить группу “group” в файл /etc/one/ldap_group.conf и проверить в каких группах состоит пользователь после входа на портал Брест.

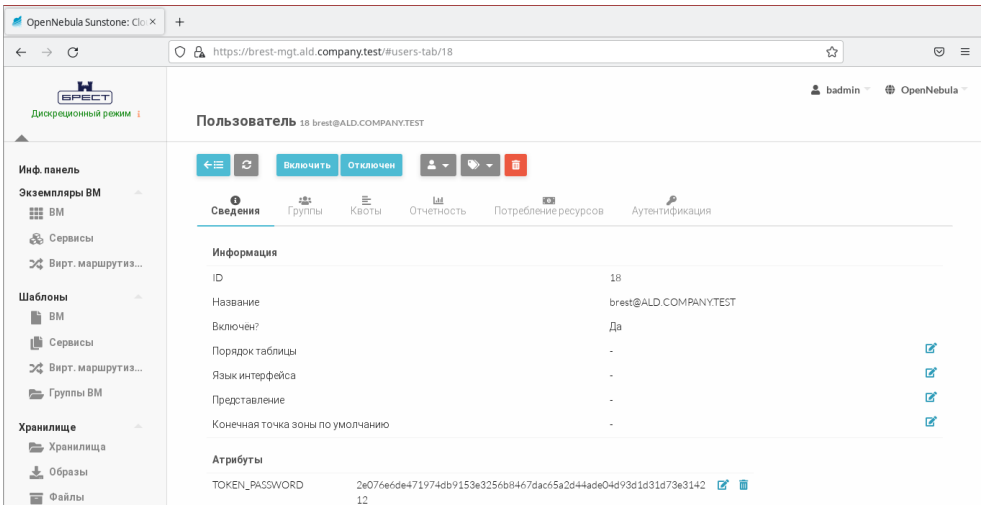


Рисунок 5 - Просмотр сведений о пользователе brest на портале управления Брест

На рисунке 4 видно, что пользователь состоит в двух группах: group и brestusers. На рисунке 5 видно, что основной группой пользователя является brestusers. Но стоит обратить внимание, что имя пользователя не короткое brest, а длинное, т.е. с указанием полного имени домена brest@ALD.COMPANY.TEST.

7 Вход пользователей из доверенного домена Windows

Пользователи доверенного домена Windows имеют возможность входа в веб-интерфейс Брест. Подробная документация по доверительным отношениям доступна в отдельном документе. На тестовом стенде были выполнены следующие настройки, описанные ниже.

Для установки доверительных отношений ALD Pro и Windows необходимо настроить DNS-службы со стороны обоих доменов. На портале ALD Pro указать имя домена Windows и IP-адрес DNS-сервера в зоне WIN.TEST. Если имя домена Windows пересекается с уже существующим именем в сети Интернет, необходимо поставить галочку “Пропустить проверку пересечения” (рис. 6).

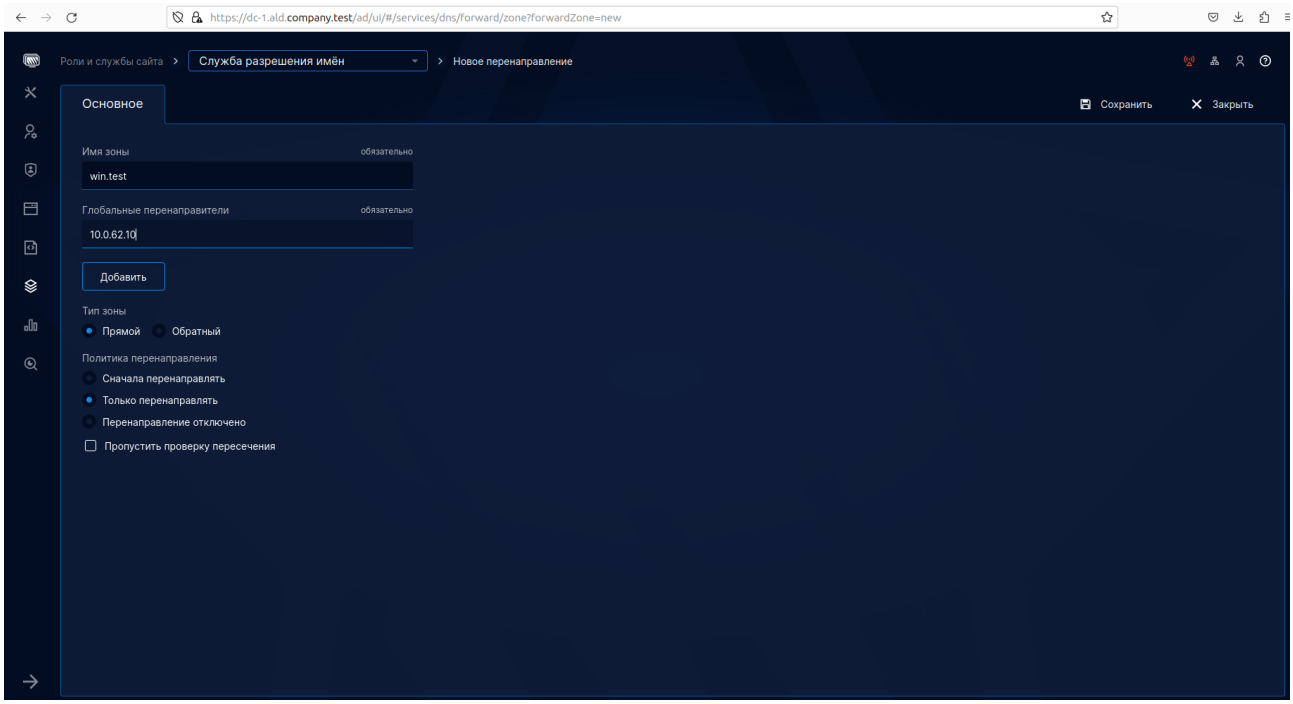


Рисунок 6 – Настройка перенаправления DNS-зоны

Со стороны Windows настройки выполняются на контроллере домена. Необходимо создать сервер условной пересылки. В “Диспетчер серверов” в меню “Средства” выбрать пункт “DNS” (рис. 7), далее “Серверы условной пересылки”, создать сервер. Заполнить поле DNS-домен и указать IP-адрес DNS-сервера домена ALD.COMPANY.TEST (рис. 8,9).

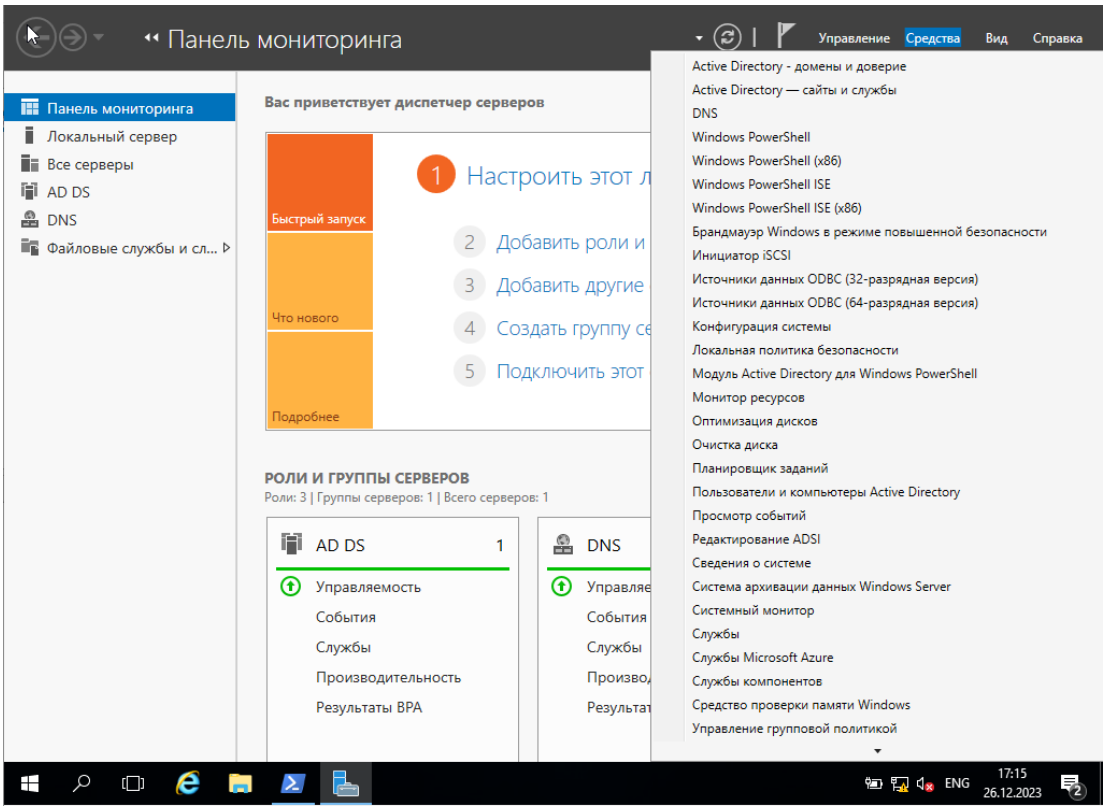


Рисунок 7 – Переход к оснастке DNS из Диспетчера серверов

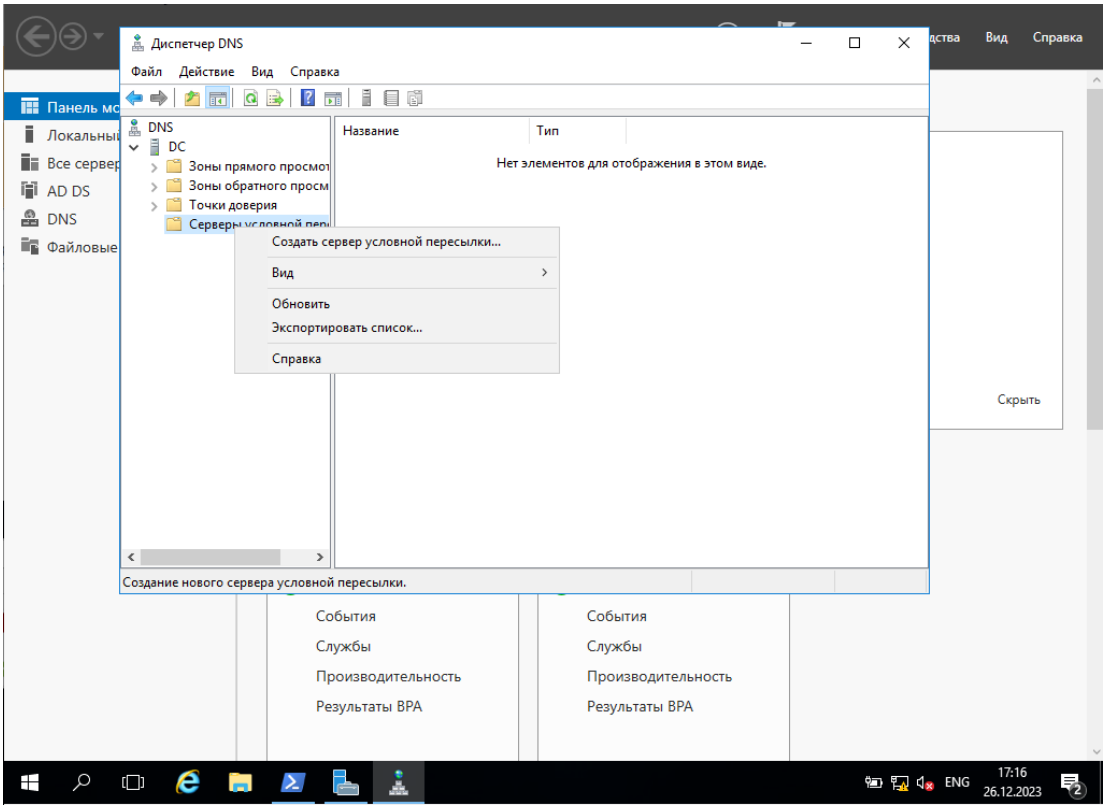


Рисунок 8 – Переход к настройке условной пересылки

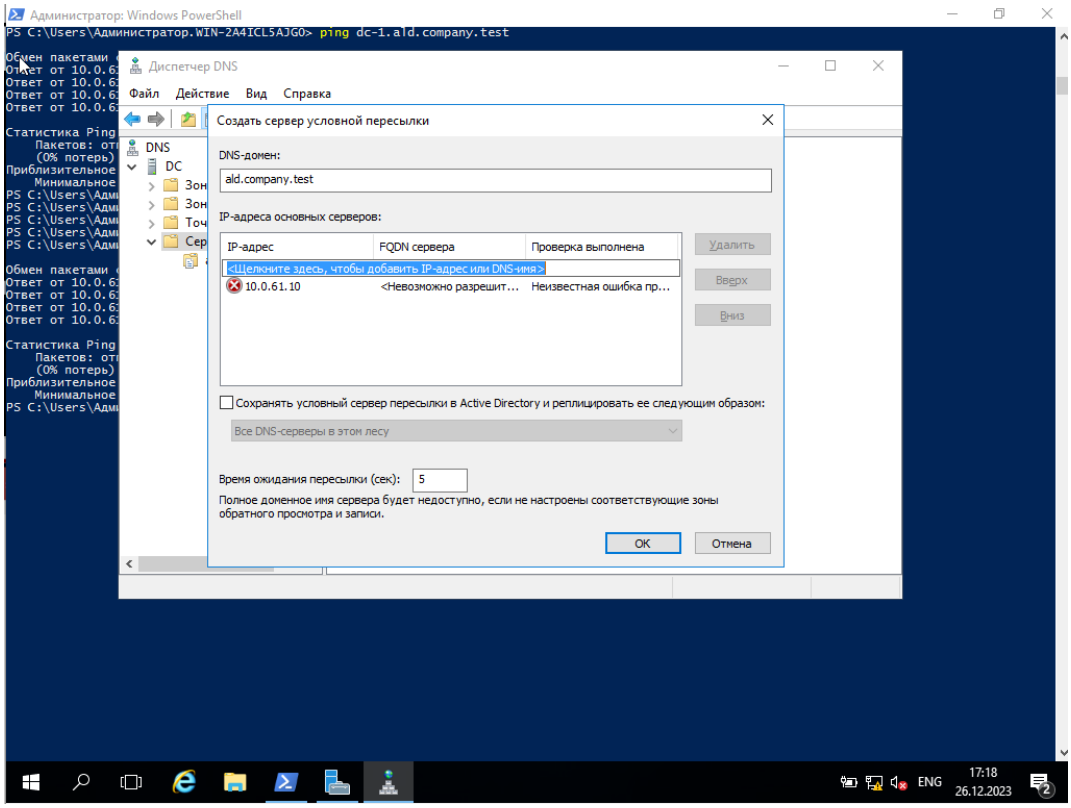


Рисунок 9 – Создание сервера условной пересылки

Настройка DNS-служб завершена. Таким образом, DNS-обращения из одного домена к другому домену будут пересылаться на соответствующие DNS-серверы.

Доверительные отношения устанавливаются командой ipa на контроллере ALD Pro, необходимо убедиться, что в сессии пользователя admin присутствует Kerberos-билет:

```
root@dc-1:~# klist
Ticket cache: KEYRING:persistent:240400000:krb_ccache_LkZ5X6l
Default principal: admin@ALD.COMPANY.TEST

Valid starting       Expires              Service principal
26.12.2023 07:15:03  27.12.2023 07:15:03  krbtgt/ALD.COMPANY.TEST@ALD.COMPANY.TEST
```

Если его нет, выполнить kinit:

```
root@dc-1:~# kinit admin
Password for admin@ALD.COMPANY.TEST:
```

Выполнить установку доверительных отношений (вывод команды сокращен для краткости):

```
root@dc-1:~# ipa -d -v trust-add --type=ad win.test --admin Администратор --password
--two-way=true

ipa: DEBUG: Loading Index file from '/var/lib/ipa-client/sysrestore/sysrestore.index'
...
Пароль администратора домена Active Directory:
...
-----------------------------------------------------------------------------
Добавлено отношение доверия Active Directory для области (realm) "win.test"
```

```
------------------------------------------------------------------------------
  Имя области (realm): win.test
  Имя домена NetBIOS: WIN
  Идентификатор безопасности домена: S-1-5-21-2512014374-231365287-8402620
  Направление отношения доверия: Двустороннее отношение доверия
  Тип отношения доверия: Домен Active Directory
  Состояние отношения доверия: Установлено и проверено
```

Далее проверить установку доверительных отношений:

```
root@dc-1:~# ipa trustconfig-show
  Домен: ald.company.test
  Идентификатор безопасности: S-1-5-21-1696290783-3000723298-4078756448
  Имя NetBIOS: ALD
  GUID домена: 0a4a7bb4-210d-4bfa-8699-7c30b06f5e97
  Резервная основная группа: Default SMB Group
  Агенты отношений доверия AD IPA: dc-1.ald.company.test
  Контролёры отношений доверия AD IPA: dc-1.ald.company.test

root@dc-1:~# smbclient -k -L dc.win.test
lpcfg_do_global_parameter: WARNING: The "domain logons" option is deprecated

Sharename       Type      Comment
---------       ----      -------
ADMIN$          Disk      Удаленный Admin
C$              Disk      Стандартный общий ресурс
IPC$            IPC       Удаленный IPC
NETLOGON        Disk      Общий сервер входа
SYSVOL          Disk      Общий сервер входа
SMB1 disabled -- no workgroup available
```

В домене Windows нужно создать группу ad_users, добавить в нее пользователей. Далее необходимо добавить внешнюю группу, включить в нее группу пользователей из домена WIN.TEST:

```
root@dc-1:~# ipa group-add --desc='AD users for Brest' ad_users_ext --external
root@dc-1:~# ipa group-add-member ad_users_ext --external "WIN.TEST\ad_users"
```

Создать POSIX группу в ALD Pro и включить в нее ранее созданную внешнюю группу:

```
root@dc-1:~# ipa group-add --desc='AD users' ad_users
root@dc-1:~# ipa group-add-member ad_users --groups ad_users_ext
```

Чтобы пользователи Windows группы ad_users могли управлять виртуальными машинами в Брест, они включаются в служебные группы:

```
root@dc-1:~# ipa group-add-member kvm --groups ad_users
root@dc-1:~# ipa group-add-member libvirt --groups ad_users
root@dc-1:~# ipa group-add-member libvirt-qemu --groups ad_users
```

В файле /etc/krb5.conf нужно убедиться, что параметры имеют следующие значения:

```
dns_lookup_realm = true
```

```
dns_lookup_kdc = true
```

На фронтальной машине brest-mgt.ald.company.test требуется настройка apache2. Для совместимости работы с пользователями Windows отключается режим AstraMode в файле /etc/apache2/sites-enabled/one-apache2.conf. Параметр AstraMode служит для дополнительной авторизации пользователей по мандатным атрибутам, так как пользователи Windows их не содержат, AstraMode устанавливается в off:

```
<VirtualHost _default_:443>
    AstraMode off
    ServerName brest-mgt.ald.company.test
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
```

Даже если AstraMode=off, в файле /etc/apache2/apache2.conf включается параметр IncludeRealm:

```
IncludeRealm on
```

После изменения конфигурационных файлов apache2 требуется рестарт службы:

```
root@dc-1:~# systemctl restart apache2
```

POSIX-группу ad_users необходимо добавить в файл /etc/one/ldap_group.conf и затем создать ее в Брест.

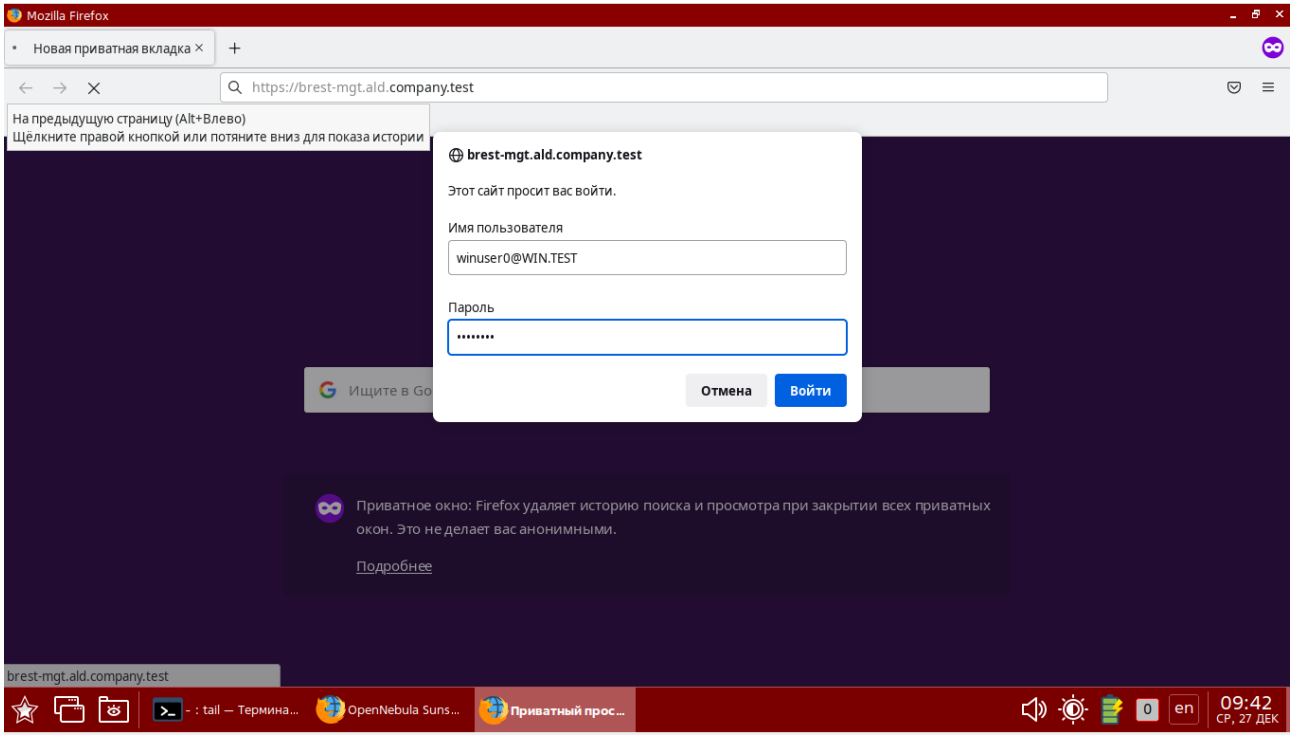
```
root@brest-mgt:~# echo ad_users > /etc/one/ldap_group.conf
root@brest-mgt:~#
root@brest-mgt:~# onegroup create "ad_users"
```

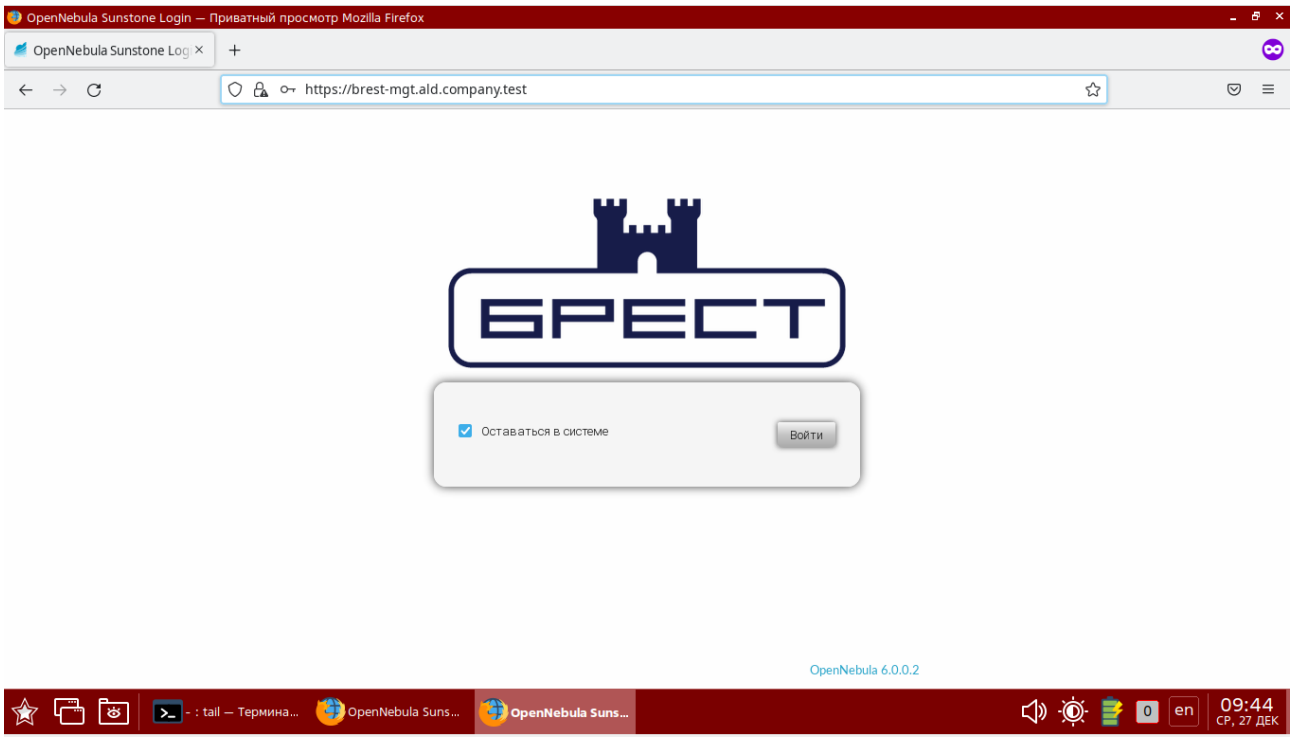
В скрипте /usr/lib/one/sh/check_user.sh проверяется, что пользователь состоит в группе из файла /etc/one/ldap_group.conf, если это так, он добавляется в нее. Переменная cpath содержит путь к файлу /etc/one/ldap_group.conf. Если пользователь входит в первый раз, он создается во внутренней базе Брест.

```
while read -r groupname; do
    idg=$(getent group "$groupname" | cut -d: -f3)
    if [ -n "$idg" ]; then
        AD="true"
        if id "$username" | grep -q "\b${idg}\b"; then
            if oneuser list | grep -q "${username}"; then
                oneuser addgroup "$username" "$groupname"
            else
                oneuser create "${username}" "${username}" --driver public --group
brestusers
                if [[ "$?" == "0" ]]; then
                    check_token ${username}
                fi
                oneuser addgroup "$username" "$groupname"
            fi
        fi
    fi
```

```
done < "$cpath"
```

Необходимо выполнить проверку входа в веб-интерфейс Брест под пользователем winuser0 (рис. 10). Пользователь winuser0 должен быть включен в групп WIN.TEST\ad_users на контроллере домена dc.win.test. Имя пользователя вводится с указанием домена:





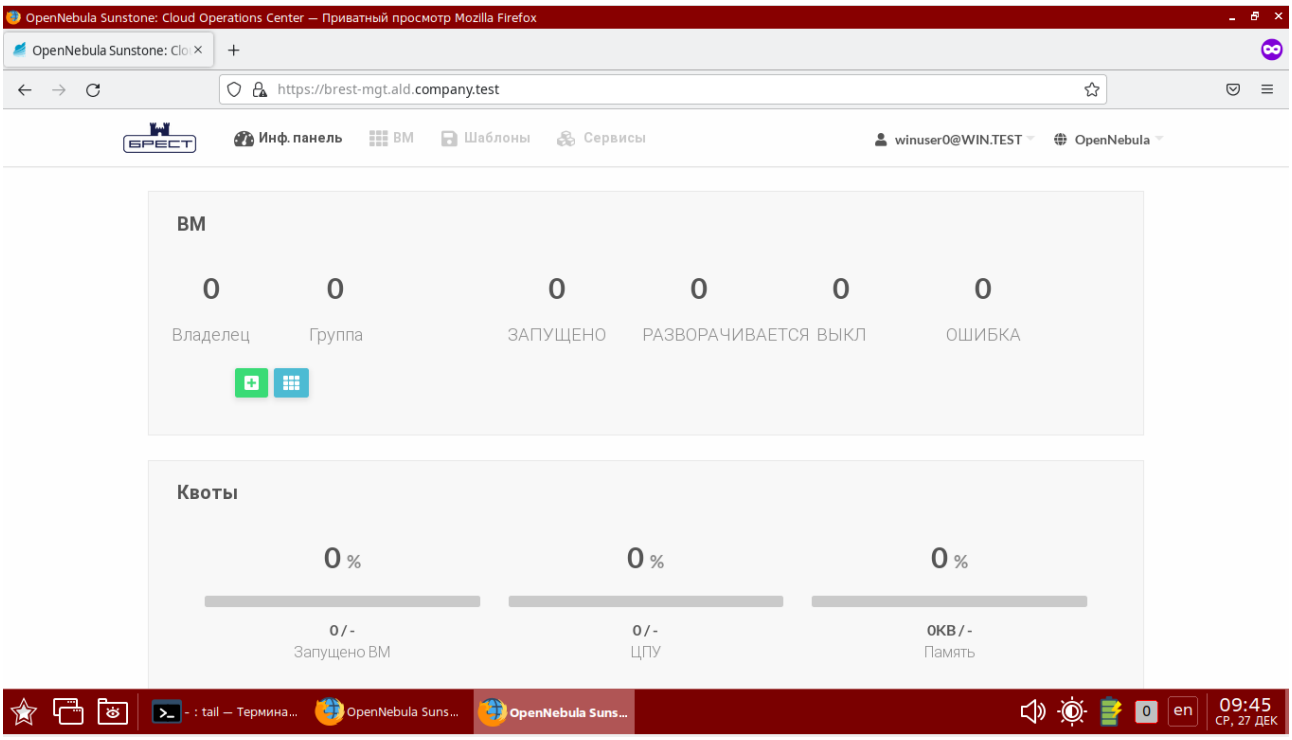


Рисунок 10 - Выполнение проверки входа в веб-интерфейс Брест под пользователем winuser0

Таким образом, в домене Windows достаточно добавлять новых пользователей в группу WIN.TEST\ad_users для разрешения им доступа к веб-интерфейсу Брест.

8 Проблемы создания пользователя после настройки входа пользователям Windows

После настроек, сделанных в разделе 6, появилась проблема с синхронизацией пользователей из Брест в домен ALD Pro. На рисунке 11 в форме для создания пользователя в Брест пропала галочка “Сменить пароль при первом входе в систему”.

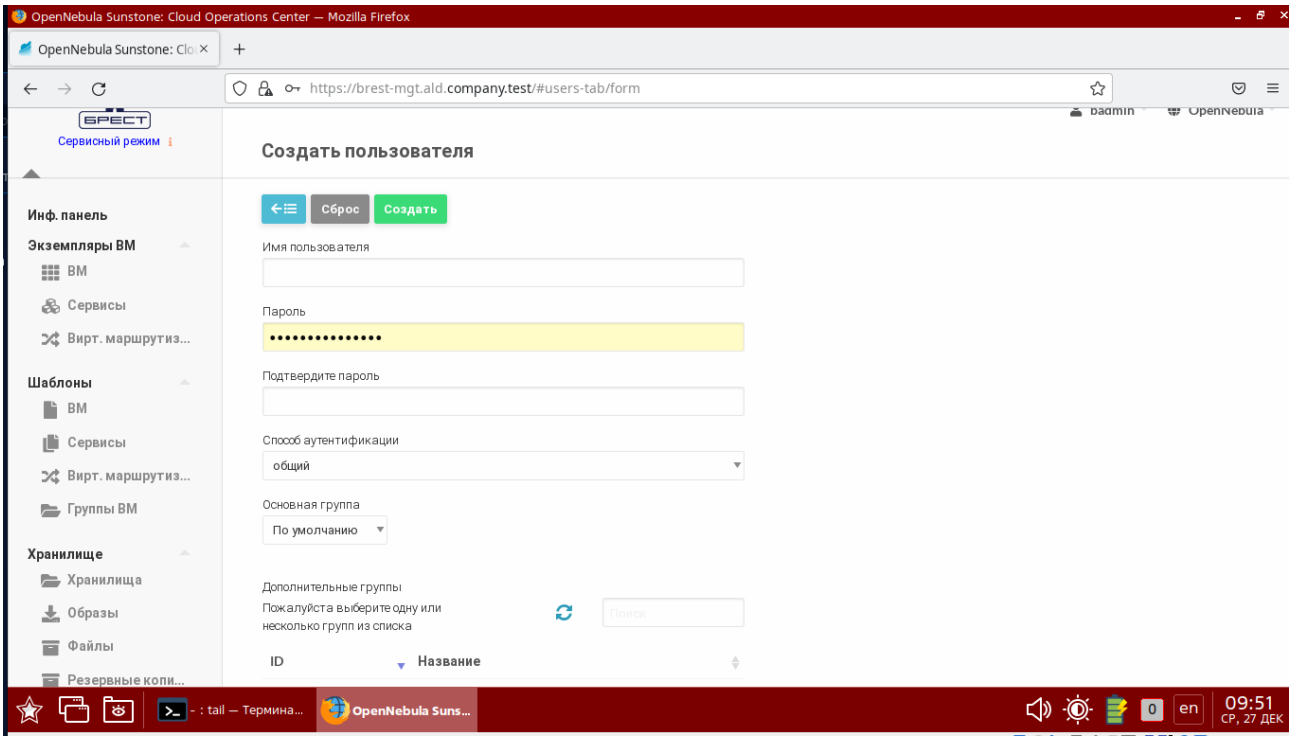


Рисунок 11 – Форма для создания нового пользователя

Если заполнить все поля и нажать кнопку “Создать”. Новый пользователь не появится в ALD Pro. Для групп аналогично, новые группы в Брест не создаются в ALD Pro (рис. 12.1, 12.2).

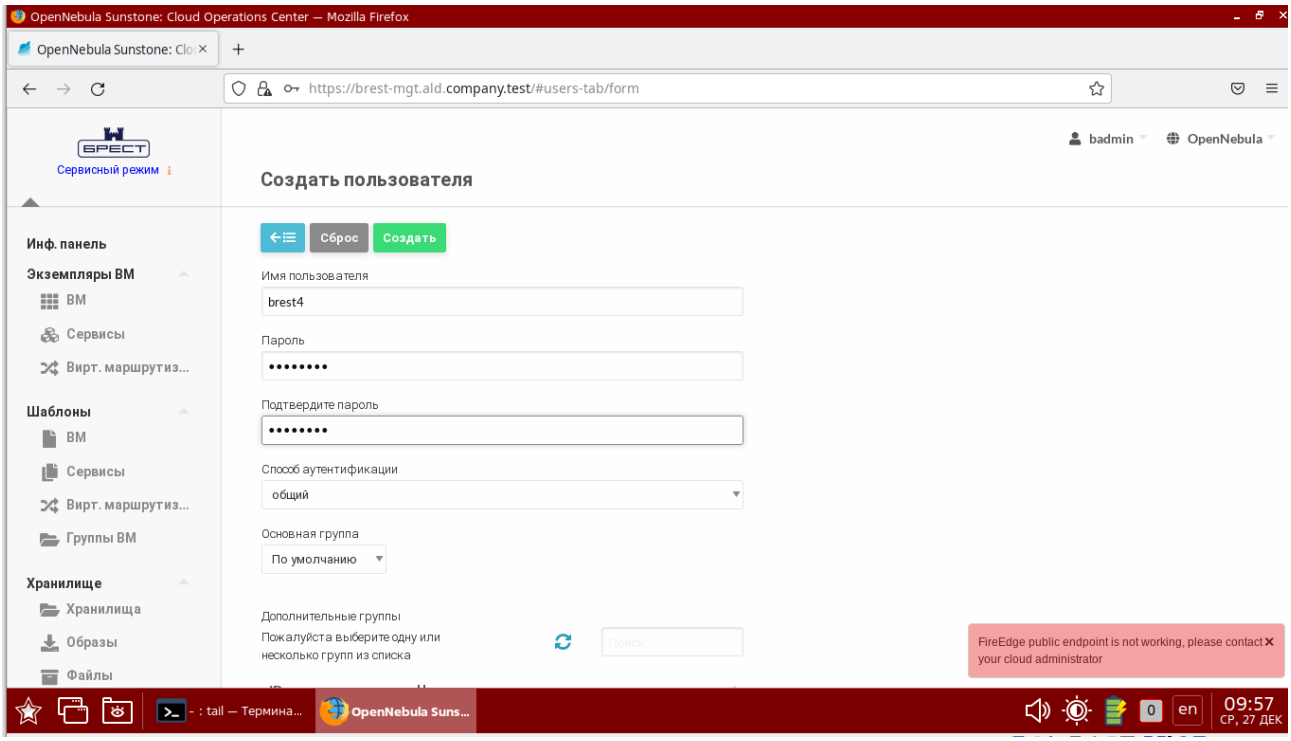


Рисунок 12.1 – Создание пользователя

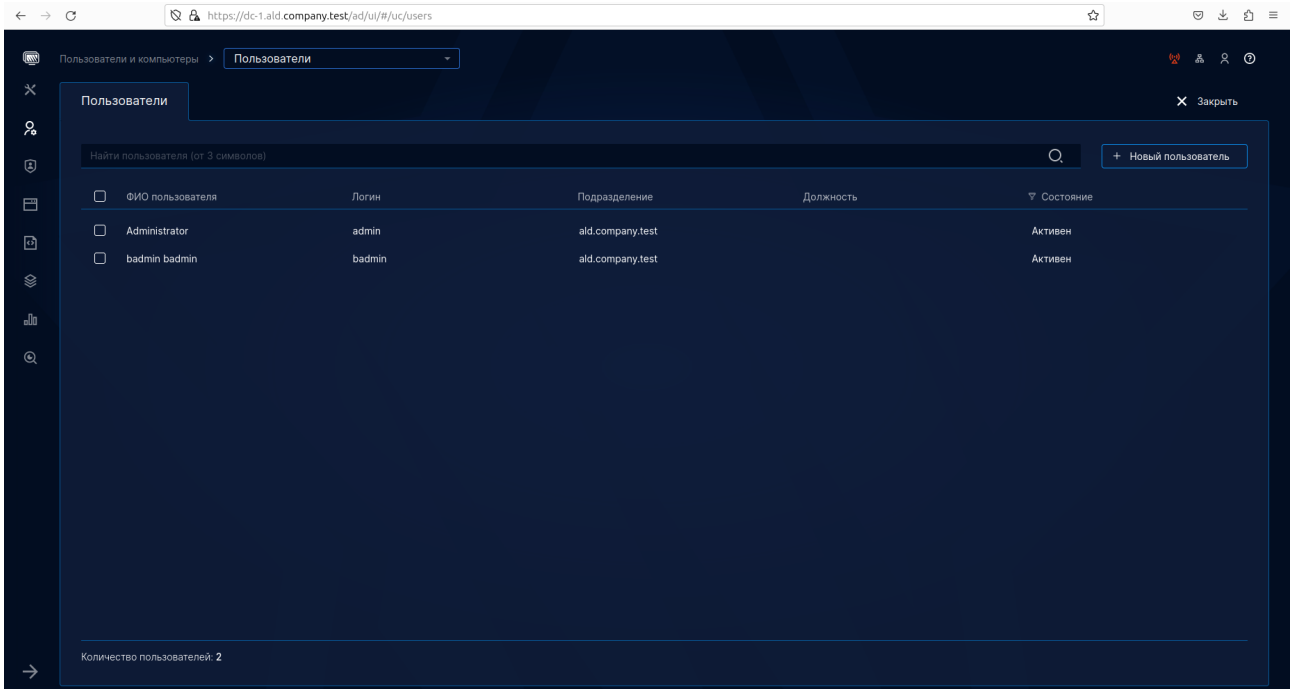


Рисунок 12.2 – Просмотр списка пользователей

При загрузке ОС запускается скрипт /usr/sbin/one-brestmode, который сохраняет в файле /var/run/one/brest.env глобальные переменные:

```
root@brest-mgt:~# cat /var/run/one/brest.env

ONE_ASTRA_MODE=2
ONE_BREST_MODE=0
```

Параметр ONE_ASTRA_MODE говорит о том, что ОС работает с включенным мандатным контролем, а ONE_BREST_MODE указывает на то, в каком режиме должна работать система виртуализации Брест. Если ONE_BREST_MODE равен 0, то Брест работает в дискреционном режиме. Если ONE_BREST_MODE равен 1, то Брест находится в сервисном режиме.

Учитывая, что в скрипте /usr/sbin/one-brestmode значение параметра ONE_BREST_MODE выставляется в соответствии с значением параметра AstraMode из файла /etc/apache2/sites-enabled/one-apache2.conf, после перезагрузки системы параметр ONE_BREST_MODE станет равен 0. Пример кода из скрипта:

```
root@brest-mgt:~# cat /usr/sbin/one-brestmode
#!/bin/bash

echo "ONE_ASTRA_MODE=$(astra-modeswitch get)" > /var/run/one/brest.env
if [ -f /etc/apache2/sites-enabled/one-apache2.conf ]; then
    aa_mode=$(cat /etc/apache2/sites-enabled/one-apache2.conf | sed -e '/
*<VirtualHost *\_default_:443>/,/.*<\/VirtualHost>/!d' | awk '{if($1 == "AstraMode")
print $2}')
    if [ "$aa_mode" == "off" ]; then
        echo 'ONE_BREST_MODE=0' >> /var/run/one/brest.env
    else
        echo 'ONE_BREST_MODE=1' >> /var/run/one/brest.env
    fi
else
    echo 'ONE_BREST_MODE=1' >> /var/run/one/brest.env
fi
```

Таким образом, проблема связана с тем, что изменение AstraMode меняет режим работы Брест. Брест начинает функционировать в дискреционном режиме без домена, поэтому дополнительные функции, доступные в сервисном режиме, как, например, синхронизация пользователей и групп, перестают работать. Согласно документу 2 (см. Материалы), глобальное значение AstraMode находится в файле /etc/apache/apache2 и оно равно off, мы можем переопределять его для каждого сайта. Поэтому для того, чтобы пользователи Windows могли входить на портал Брест, AstraMode был выключен в /etc/apache2/sites-enabled/one-apache2.conf, что привело к проблеме.

9 Материалы

1. https://life.astralinux.ru/pages/viewpage.action?pageId=96684416
2. https://wiki.astralinux.ru/pages/viewpage.action?pageId=238749508
3. https://wiki.astralinux.ru/brest/2.12/avtorizatsiya-pol-zovatelej-ad-v-pk-sv-brest-238447075.html
4. Инструкция по доверительным отношениям (из личного кабинета)