Инструкция по настройке и выявлению ошибок push-сервисов

# Настройка

### Добавление доменного имени

Для того, чтобы сервисы могли обращаться к gateway.push.apple.com, необходимо обеспечить сетевую видимость gateway.push.apple.com с серверов в Сигма. В связи с тем, что за именем gateway.push.apple.com скрывается большое количество серверов с разными IP и сделать проброски до всех из них не представляется возможным, было выбрано несколько фиксированных адресов, до которых будет происходить подключение. Для того, чтобы сервисы выбирали только те IP, что были проброшены, необходимо добавить эти адреса в список хостов.

* 1. Зайти на сервер по RDP и отредактировать файл C:\Windows\System32\drivers\etc\hosts
  2. В конец файла дописать IP-адрес Apple сервера и доменное имя (gateway.push.apple.com) через пробел

Пример:

17.110.224.218 gateway.push.apple.com

17.110.224.163 gateway.push.apple.com

17.110.227.91 gateway.push.apple.com

17.110.227.33 gateway.push.apple.com

17.172.232.35 gateway.push.apple.com

17.172.232.30 gateway.push.apple.com

### Настройка push сервиса

Сервис для отправки push уведомлений называется ***pushNotificationService*** и находится в папке «common» на syncserver. Однако для непосредственной передачи уведомлений на сервера Apple он использует сервис ***AppleSender***, находящийся в той же папке. Для соединения с Apple сервером сервис ***AppleSender*** использует сертификаты, расположенные в папке, указанной в параметре *certificatesFolder* (см. четвертую колонку в сервисах), который при запуске заполняется значением из настройки PUSH\_CERTIFICATE\_CONFIG\_FOLDER с добавлением в конец папки «/IOS». В этой папке должны находиться сертификаты для отправки push уведомлений.

Содержимое папки должно иметь следующую структуру папок и файлов:

* 1. *certificatesFolder* по умолчанию заканчивается на IOS (т.е. PUSH\_CERTIFICATE\_CONFIG\_FOLDER/IOS)
  2. в папке IOS должны находиться папки с названиями приложений (идентификаторы приложений). Например, iNavigator2 и iNavigator2Phone
  3. В каждой такой папке должен находиться сертификат с любым названием, но имеющим расширение \*.p12, и файл settings.properties, содержащий 2 строчки:  
     *password=123456  
     productionMode=true*  
     Другие файлы в папке игнорируются.

*При необходимости файл* ***settings.properties*** *можно создать вручную или скопировать из другого приложения, заменив пароль (123456) на пароль от сертификата данного приложения.*

Перезапуск сервиса ***AppleSender*** приводит к перезагрузке сертификатов (перезапуск ***pushNotificationService*** теперь делать не желательно). О найденном сертификате будет свидетельствовать запись в логах сервиса ***AppleSender***. Т.е. после перезапуска сервиса в логах сервиса будет указаны все найденные сертификаты.

|  |  |  |
| --- | --- | --- |
| 20.07.2016 16:41 | ApplePushNotificationSender | Найден сертификат D:\usr\cache\push\IOS\iNavigator |
| 20.07.2016 16:41 | ApplePushNotificationSender | Найден сертификат D:\usr\cache\push\IOS\Competitors |
| 20.07.2016 16:40 | ServiceContainer | Skip stopping service ru.sberbank.syncserver2.service.pushnotifications.senders.ApplePushNotificationSender |

### Настройка сервиса для загрузки уведомления из БД источника.

Для загрузки уведомлений из БД источника в очередь отправки уведомлений используется сервис ***pushNotificationUploader***, в syncserver, имеющий 2 параметра: *connectionService* и *appName*.

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр сервиса | Соответствующая настройка | Описание | Пример |
| *connectionService* | при старте сервиса заполняется значением из настройки PUSH\_NOTIFICATIONS\_SOURCE\_SERVICE | Значение DATAPOWER\_TARGET из проксисервера, который будет использоваться для загрузки уведомлений (каждый TARGET выполняет запросы в определенной БД) | finik2-new (запрос для загрузки уведомлений направится в finik2) |
| *appName* | при страте сервиса заполняется значением из настройки PUSH\_NOTIFICATIONS\_APP\_NAME | Здесь должны быть перечислены через «,» (запятую) без пробелов названия приложений, для которых предназначены эти загруженные уведомления | iNavigator2,iNavigator2Phone |

### Настройка конфигурации мобильного приложения.

Мобильное приложение, которое должно получать push уведомления, должно иметь параметры *PushServer1* и *PushServer2* в настройках, получаемых приложением с confserver-а. Они должны указывать на разные ноды **syncserver-а**, который будет использоваться для регистрации получателей и отправки им push уведомлений. Адреса должны быть доступны из внешней сети и оканчиваться на …/common/request.do. Для проверки ссылки можно вставить ее в адресную строку браузера; должна появиться надпись “Error was ocuried while executing method. (null)”.

# Проверка регистрации устройства

При запуске мобильного приложения, оно должно автоматически пройти регистрацию на push уведомления по ссылкам, указанным в пришедших настройках (*PushServer1* и *PushServer2*).

1. Запустить мобильное приложение.
2. Зайти на форму отправки push уведомлений (или обновить страницу)
3. Убедиться, что данное устройство доступно для выбора в форме.

Последний пункт также означает, что клиент появился (добавился или обновился) в БД syncserver-а в таблице SYNC\_PUSH\_NOTIFICATIONS\_CLIENTS (доступные устройства в форме берутся именно из этой таблицы). Основные колонки в таблице:  
 SYNC\_PNC\_APPLICATION\_CODE – название (идентификатор) мобильного приложения  
 SYNC\_PNC\_EMAIL – email пользователя  
 SYNC\_PNC\_DEVICE\_NAME – имя устройства, соответствует полю «Выберите Устройство:» на странице формы отправки уведомлений.

1. Выбрать данное устройство и отправить тестовое уведомление.

Если при отправке уведомления через форму появилось сообщение «0 пользователей системы получат сообщение в ближайшее время», то выбранные устройства имеют неверный токен (за время заполнения формы произошла инвалидация токена – он стал недействительным; этому устройству отправка уведомления пропущена). Токен должен обновиться при следующей регистрации устройства (запуск/разворачивание мобильного приложения).

# Возможные ошибки

### Ошибки отправки уведомления

Уведомление, добавленное в очередь, через несколько минут должно попасть в архив (в историю). С точки зрения БД, уведомление перейдет из таблицы SYNC\_PUSH\_NOTIFICATIONS в таблицу SYNC\_PUSH\_NOTIFICATIONS\_ARCHIVE.

* 1. Через 2-3 минуты перейти к истории уведомлений и убедиться, что оно там появилось.
  2. Двойным нажатием по уведомлению перейти к списку получателей. Здесь напротив каждого клиента указан ответ, пришедший от сервера Apple, либо описание ошибки сервиса, из-за которой сервис не смог отправить данное уведомление.

Любой статус, отличный от **OK**, является ошибкой. Статус **undefined** указывает на то, что либо произошла ошибка в сервисе (в этом случае следует выгрузить логи сервисов ***AppleSender*** и ***pushNotificationService*** в Excel и отправить разработчикам), либо токен получателя стал недействительным за промежуток времени с момента загрузки уведомления из БД навигатора (из альфы) до попытки отправить уведомление сервисом.

### Ошибки загрузки уведомлений из альфы

В логи сервиса ***pushNotificationUploader*** попадают записи о любых ошибках, возникших при загрузке или обновления статуса уведомления (обновление статуса при этом повторится либо при следующем автоматическом запуске, либо при обновлении результата отправки). Также туда попадает информация о количестве созданных уведомлений в очереди и об общем количестве получателей.

#### Замечание.

В истории уведомлений, если уведомление было загружено из альфы, некоторые получатели могут иметь статус **undefined**. Связано это с тем, что на момент добавления уведомления в очередь, данный клиент имел действительный токен, но в ходе отправки предыдущих уведомлений выяснилось, что этот токен больше не является действительным (invalid token), поэтому отправка уведомления этому клиенту в Apple сервер не произошла.

# Описание основных объектов в БД, используемых push-сервисами

### В БД syncserver

|  |  |
| --- | --- |
| SYNC\_PUSH\_NOTIFICATIONS | очередь уведомлений на отправку, включая уведомления, находящиеся в обработке |
| SYNC\_PUSH\_NOTIFICATIONS\_ARCHIVE | архив уведомлений; в эту таблицу уведомления попадают после отправки и хранятся для отображения истории |
| SYNC\_PUSH\_NOTIFICATIONS\_CLIENTS | таблица зарегистрированных получателей; здесь хранится информация о пользователях и токены устройств |
| SYNC\_PUSH\_NOTIFICATIONS2CLIENT | таблица связей уведомлений с получателями; также в этой таблице хранится статус по каждой паре (уведомление + клиент) |

### В БД Навигатора

|  |  |
| --- | --- |
| Вызовы процедур к БД Навигатора проходят с использованием шаблонов и проходят через прокси процедуры в MIS\_IPADX, где добавляется еще 1 аргумент для обозначения среды | |
| MIS\_NAVIGATOR\_KPI\_20.dbo. SELECT\_NOTIFICATIONS\_FOR\_SEND | Конечная процедура, используемая сервисом для получения списка уведомлений из alpha. Она в свою очередь влияет на внутреннюю структуру таблиц Навигатора (в первую очередь на представления MIS\_NAVIGATOR\_KPI\_20.UI.v\_PushNotification и MIS\_NAVIGATOR\_KPI\_20.UI.v\_PushNotificationStateLog) |
| MIS\_NAVIGATOR\_KPI\_20.dbo. UPDATE\_NOTIFICATION\_STATE\_STATUS | Конечная процедура, используемая сервисом для обновления статуса уведомления в alpha. Она влияет на внутреннюю структуру таблиц Навигатора. |