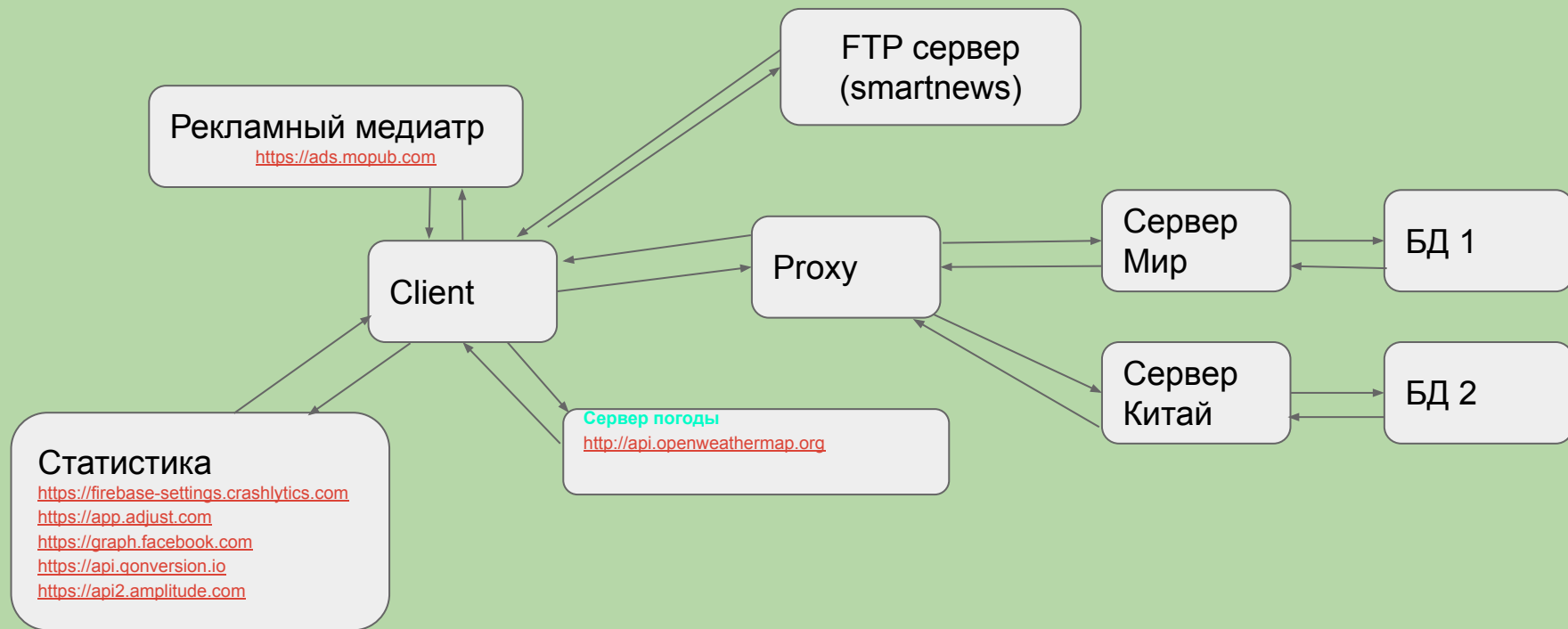


THEMIFY

Yuntsevich

СХЕМА ВЗАИМОДЕЙСТВИЯ КЛИЕНТА И СЕРВЕРА



ПРИМЕР:

ОБЪЕКТ - ВИДЖЕТ, РЕГИОН - НЕ КИТАЙ :

1. Пользователь открыв приложение (клиент) посылает **GET запрос** на сервер за виджетом используя защищенное соединение . Набор шифроф TLSv1.2 на 50305 порт определен протоколом HTTP/2.0 и вшитым SSL сертификатом (используют ранее выбранный комплект шифров для согласования безопасного соединения)

Пример: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

- TLS - протокол
- ECDHE - во время обмена ключи будут заменены эфемерной кривой Diffie Hellman (используется защищенный от прослушки канал).
- RSA - алгоритм аутентификации (клиент отправляет открытый ключ - у сервера для расшифровки есть свой закрытый ключ)
- AES_128_GCM - алгоритм массового шифрования
- SHA-256 - алгоритм хеширования

1. Прокси сервер определяет из какого региона исходит запрос, Китай или Не Китай
2. Определив , что запрос исходит не из Китая – прокси передает запрос в базу данных сервера Мир
3. Сервер Мир из Базы Данных 1 отдает виджет клиенту.

КОМПЬЮТЕРНЫЕ СЕТИ

Участники :

клиент – мобильное приложение **Themify**, порт процессу приложения на девайсе назначается ОС, например **57597**
сервер – **https://aiby.mobi:50305**, где **https://aiby.mobi** – доменное имя хоста, **50305**– порт.

Пояснение работы приложения на хосте 1 (клиент) по уровням:

Прикладной

- **DNS** – клиент отправляет запрос к DNS серверу, для того чтобы по доменному имени **https://aiby.mobi** получить IP адрес хоста
- **HTTP** – клиент формирует и отправляет на сервер POST запрос (за json файлом, где прописаны параметры виджета)
- **URL** – /widget/1.27/widgets/widgets.json
- **body** –

```
{
  "bgColor": "#ffffff",
  "defaultImage": "starWarsDarkSideSix",
  "textColor": "#ffffff",
  "fontName": "Lena",
  "type": "calendarRight",
  "premiumType": "rewarded",
  "category": ["Calendar"],
}
```

Транспортный

- **TCP** (протокол с гарантированной доставкой) – формируется TCP сегмент = данные + заголовок транспортного уровня (имя хоста+порт+порядковый номер сегмента)

Сетевой

- **IP** – формируется IP пакет = TCP сегмент + заголовок сетевого уровня (IP адреса отправителя/получателя)

Канальный

- **Wi-Fi** – формируется кадр = IP пакет + заголовок канального уровня (MAC-адреса отправителя/получателя) + концевик (хеш-сумма (уникальный идентификатор файла, который при помощи специального программного обеспечения высчитывается компьютером путем определенных математических преобразований информации, содержащейся в нем))

Механизм работы на хосте 2 (сервер):

Канальный уровень

Wi-Fi - читается заголовок канального уровня, из кадра извлекается IP пакет

Сетевой уровень

IP - читается IP адрес, из IP пакета извлекается TCP сегмент

Транспортный уровень

TCP - читается порт, из TCP пакета извлекаются данные прикладного уровня

Прикладной уровень

HTTP - данные обрабатываются кодом сервера



ДЕЙСТВИЯ С ВИДЖЕТОМ

1. Пользователь создает кастомный виджет используя шаблон
2. Пользователь может выбрать/просмотреть свой ранее созданный виджет
3. Пользователь обновляет свой виджет видоизменяя его
4. Пользователь удаляет виджет

Что делает запрос?	http метод	url запроса	headers запроса	body запроса	код ответа	headers ответа	body ответа
Создание кастомного виджета в приложении Themify	Post	/widget/1.27/widgets/widgets.json/id/1	-H 'accept: application/json' \n поддерживаемый формат -H 'Content-Type: application/json' / формат передаваемых данных	{"id": "1", "bgColor": "#ffffff", "defaultImage": "starWarsDarkSideSix", "textColor": "#ffffff", "fontName": "Lena", "type": "calendarRight", "premiumType": "rewarded", "category": ["Calendar"], }	200OK	access-control-allow-headers: Content-Type,api_key,Authorization / поддерживаемый формат хедеров ответа access-control-allow-methods: GET,POST,DELETE,PUT / поддерживаемые методы запроса access-control-allow-origin: * content-type: application/json / формат передаваемых данных date: Tue,13 May 2021 12:05:06 GMT / дата запроса server: Jetty(9.2.9.v20150224) / инфо о сервере status: 200 / статус код ответа	{ "code": 200, "type": "unknown", "message": "ok" }
Получить виджет по id	Get	/widget/1.27/widgets/widgets.json/id/1	-H 'accept: application/json' \n поддерживаемый формат		200OK	access-control-allow-headers: Content-Type,api_key,Authorization access-control-allow-methods: GET,POST,DELETE,PUT access-control-allow-origin: * content-type: application/json date: Tue,13 May 2021 12:05:06 GMT server: Jetty(9.2.9.v20150224) status: 200	{"id": "1", "bgColor": "#ffffff", "defaultImage": "starWarsDarkSideSix", "textColor": "#ffffff", "fontName": "Lena", "type": "calendarRight", "premiumType": "rewarded", "category": ["Calendar"], }
Изменение шрифта	Put	widget/1.27/widgets/widgets.json/id/1	-H 'accept: application/json' \n поддерживаемый формат -H 'Content-Type: application/json' / формат передаваемых данных	{"id": "1", "bgColor": "#ffffff", "defaultImage": "starWarsDarkSideSix", "textColor": "#ffffff", "fontName": "RousseauDeco", "type": "calendarRight", "premiumType": "rewarded", "category": ["Calendar"], }	200 ok	access-control-allow-headers: Content-Type,api_key,Authorization access-control-allow-methods: GET,POST,DELETE,PUT access-control-allow-origin: * content-type: application/json date: Tue,13 May 2021 12:05:06 GMT server: Jetty(9.2.9.v20150224) status: 200	{ "code": 200, "type": "unknown", "message": "ok" }
Удаление виджета	Delete	widget/1.27/widgets/widgets.json/id/1	-H 'accept: application/json' \n поддерживаемый формат	{"id": "1", "bgColor": "#ffffff", "defaultImage": "starWarsDarkSideSix", "textColor": "#ffffff", "fontName": "Lena", "type": "calendarRight", "premiumType": "rewarded", "category": ["Calendar"], }	200 ok	access-control-allow-headers: Content-Type,api_key,Authorization access-control-allow-methods: GET,POST,DELETE,PUT access-control-allow-origin: * content-type: application/json date: Tue,13 May 2021 12:05:06 GMT server: Jetty(9.2.9.v20150224) status: 200	{ "code": 200, "type": "unknown", "message": "ok" }
Запрос за виджетом, который отсутствует на сервере	Get	widget/1.27/widgets/widgets.json/id/1	-H 'accept: application/json' \n поддерживаемый формат	{"id": "1", "bgColor": "#ffffff", "defaultImage": "starWarsDarkSideSix", "textColor": "#ffffff", "fontName": "Lena", "type": "calendarRight", "premiumType": "rewarded", "category": ["Calendar"], }	404 not found	access-control-allow-headers: Content-Type,api_key,Authorization access-control-allow-methods: GET,POST,DELETE,PUT access-control-allow-origin: * content-type: application/json date: Tue,13 May 2021 12:05:06 GMT server: Jetty(9.2.9.v20150224) status: 404	{ "code": 404 , "type": "error", "message": "Widget not found" }

```
widgets.json
widgets.json > No Selection
1 [
2
3 {
4   "bgColor": "#ffffff",
5   "defaultImage": "blackheart_0",
6   "textColor": "#ffffff",
7   "fontName": "Dunnu",
8   "type": "time",
9   "premiumType": "rewarded",
10  "category": ["Time"],
11 },
12 {
13   "bgColor": "#ffffff",
14   "defaultImage": "time4",
15   "textColor": "#ffffff",
16   "fontName": "hero",
17   "type": "time",
18   "premiumType": "free",
19   "category": ["Time"],
20 },
21 {
22   "bgColor": "#ffffff",
23   "defaultImage": "orangeAndBeigePaintStrokes",
24   "textColor": "#ffE4C3ff",
25   "fontName": "DUNNU",
26   "type": "time",
27   "premiumType": "free",
28   "category": ["Time"],
29 },
30 {
31   "bgColor": "#ffffff",
32   "defaultImage": "paintOfPinkSplash",
33   "textColor": "#ffC5EDff",
34   "fontName": "RousseauDeco",
35   "type": "time",
36   "premiumType": "rewarded",
37   "category": ["Time"],
38 },
39
```

API

В приложении используется **REST архитектура**, которая обеспечивает общение между клиентом и сервером с помощью https-запросов, передавая информацию от клиента в параметрах самих запросов, информацию от сервера – в теле ответа.

REST является архитектурным стилем, а не стандартом. Для обмена данными используется текстовый формат обмена данными **JSON**.

В json содержится **массив данных** (заключается в квадратные скобки ([и]), который включает в себя **объекты** (заключается в фигурные скобки ({ и })). Объекты содержат разделенный запятой список пар **имя/значение**.

Пара имя/значение состоит из имени поля, заключенного в двойные кавычки, за которым следует двоеточие (:) и значение поля, такие как:

<code>"bgColor": "#ffffff",</code>	➤	цвет фона,
<code>"defaultImage": "blackheart_0",</code>	➤	дефолтное изображение,
<code>"textColor": "#ffffff",</code>	➤	цвет текста,
<code>"fontName": "Dunnu",</code>	➤	стиль шрифта,
<code>"type": "time",</code>	➤	тип виджета,
<code>"premiumType": "rewarded",</code>	➤	премиальность виджета,
<code>"category": ["Time"],</code>	➤	категория виджета.

API

GET /widget/1.27/	Что делает метод.	GET /widget/1.27/widgets/{id}	{id} = айдишка виджета в базе		боди нет в GET запросе	200	в json название виджета
GET /widget/1.27	Идея теста 1. Например, получение информации о виджете	GET http://widget/1.27/widgets/id/1	id = 1		пустое	200	{ "id": "1", "bgColor": "#ffffff", "defaultImage": "starWarsDarkSideSix", "textColor": "#ffffff", "fontName": "Lena", "type": "calendarRight", "premiumType": "rewarded", "category": ["Calendar"], }
	Создать виджет						