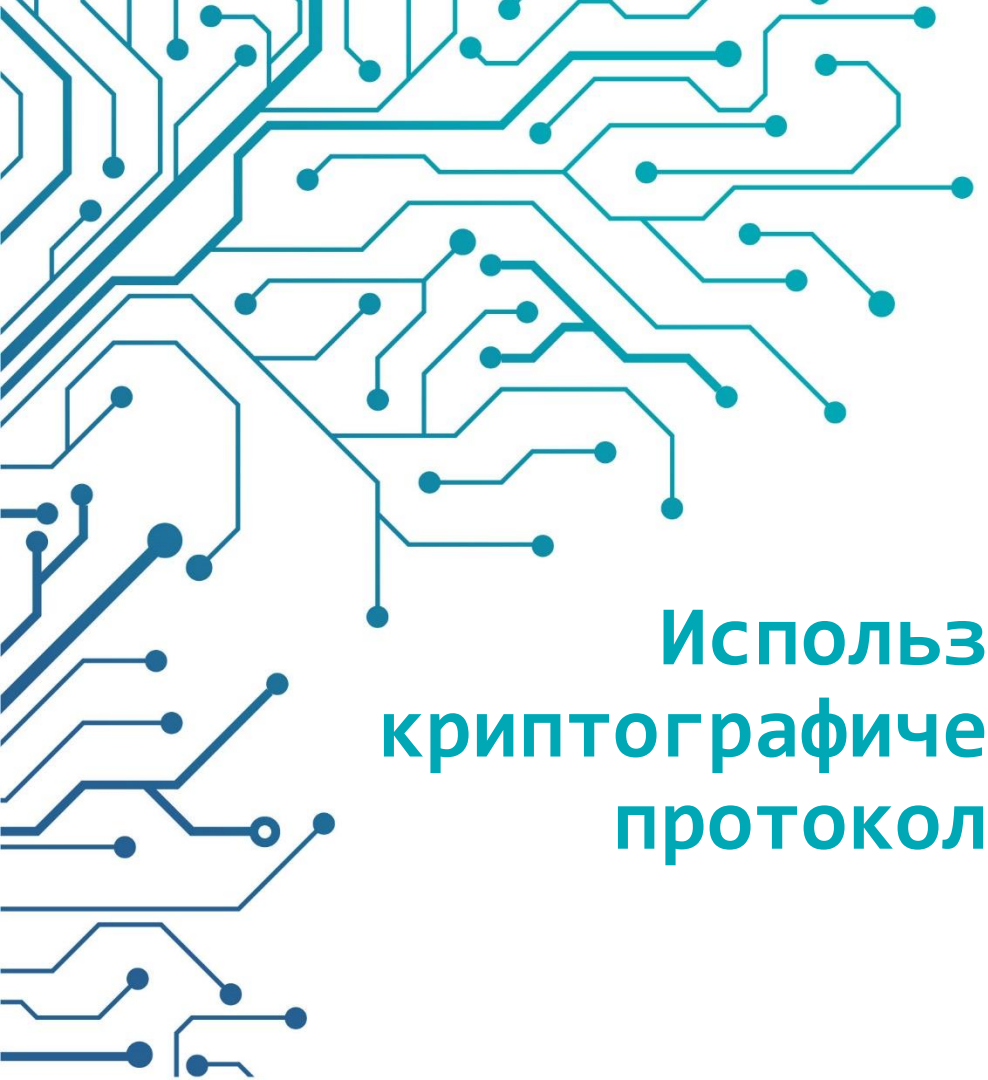


# Использование российских криптографических алгоритмов в протоколах OpenID Connect

Грунтович М.М.
Mikhail.Gruntovich@infotecs.ru

# OpenID Connect

- Единая система идентификации и аутентификации (ЕСИА)
- Единая биометрическая система (ЕБС)
- СТО БР ФАПИ.СЕК-1.6-2020 Безопасность финансовых (банковских) операций (ФАПИ.СЕК). Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID. Требования
- СТО БР ФАПИ.ПАОК-1.0-2021 Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы. Обеспечение безопасности финансовых сервисов при инициации OPENID CONNECT клиентом потока аутентификации по отдельному каналу. Требования

# OpenID Connect

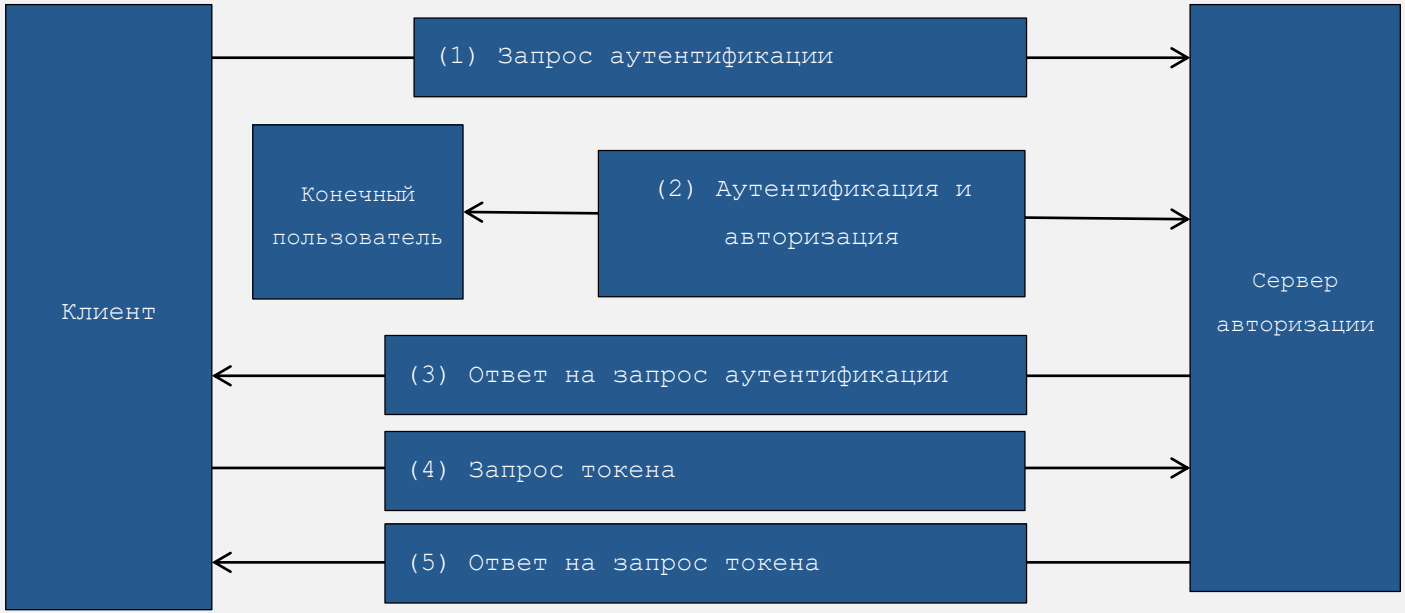
СТО БР ФАПИ.СЕК-1.6-2020:

> «Положения настоящего стандарта применяются совместно с документом Технического комитета ТК26 «Использование российских криптографических алгоритмов в протоколах OpenID Connect»

# OpenID Connect

Технические спецификации ТК26 «Информационная технология. Криптографическая защита информации. Использование российских криптографических алгоритмов в протоколах OpenID Connect»

# Протокол OpenID Connect с кодом авторизации



# Протокол OpenID Connect с кодом авторизации

# Протокол OpenID Connect с кодом авторизации

# Протокол OpenID Connect с кодом авторизации

# Протокол OpenID Connect с кодом авторизации

# Протокол OpenID Connect с кодом авторизации

# Структуры данных

**Код авторизации, токен доступа** – символьные строки

**ID токен** – структура JWT

**JWT** – JSON веб-токен (RFC 7523) – JWS или JWE(JWS)

**JWS** – цифровая подпись в формате JSON (RFC 7515)

**JWE** - шифрование в формате JSON (RFC 7516)

# Ключи

**Ключи TLS** сервера авторизации и клиента

**client_secret** – симметричный ключ клиента

**Ключи ЭЦП** сервера авторизации и клиента

**Ключи вычисления ключей** шифрования сервера авторизации и клиента

# Алгоритмы ГОСТ в JWT

- **“GOST341112_256”** - КА по алгоритму HMAC_GOST3411_2012_256 (Р 50.1.113-2016)
- **“GOST341012”** - цифровая подпись по алгоритму ГОСТ Р 34.10-2012

- **“GKDF256”** – диверсификация ключа по алгоритму KDF_GOSTR3411_2012_256 (Р 50.1.113-2016) на основе хэш-функции ГОСТ Р 34.11-2012
- **“VKO256”** - согласование ключа VKO_GOSTR3410_2012_256 (Р 50.1.113-2016) с длиной 256 бит на основе хэш-функции ГОСТ Р 34.11-2012 с длиной выхода 256 бит

- **“MKE256”** - шифрования данных шифром «Кузнечик» в режиме MGM (Р 1323565.1.026-2019)
- **“MME256”** - шифрования данных шифром «Магма» в режиме MGM (Р 1323565.1.026-2019)

# Алгоритмы ГОСТ в JWT

## Цифровая подпись

- **“GOST341112_256”:**

  $JWS\ Signature = HMAC_{256}(KDF_{256}(client\_secret,“JWS\_GKDF”,seed),M)$

  $HMAC_{256}$, $KDF_{256}$ - по Р 50.1.113-2016
  client_secret - симметричный ключ клиента

- **“GOST341012”:**

  $JWS\ Signature = (r \| s) = Sign(SKey,M)$

  Sign - ЭЦП по ГОСТ Р 34.10-2012
  SKey – ключ подписи отправителя

# Алгоритмы ГОСТ в JWT

## Вычисление ключа шифрования данных CEK

- **“GKDF256”:**

  $CEK = KDF_{256}(client\_secret,“JWE\_GKDF”,seed)$

  $KDF_{256}$ - диверсификация ключа по Р 50.1.113-2016
  client_secret - симметричный ключ клиента

- **“VKO256”:**

  $CEK = KEK_{vko}(d_s, Q_r, UKM)$

  $KEK_{vko}$ - вычисление ключа по Р 50.1.113-2016
  $d_s$, $Q_r$ = ключи отправителя и получателя

# Алгоритмы ГОСТ в JWT

## Шифрование данных

- **“MKE256” и “MME256”:**

  (C ‖ T) = AEAD-Encrypt (CEK, nonce, A, M)

  AEAD-Encrypt – шифрование в режиме MGM по Р 1323565.1.030-2020