ETU "LETI"
SAINT PETERSBURG ELECTROTECHNICAL UNIVERSITY

ОСНОВЫ ПОСТРОЕНИЯ ЗАЩИЩЕННЫХ КОМПЬЮТЕРНЫХ СЕТЕЙ

Протоколы идентификации, аутентификации и авторизации

Основные атаки

Маскарад (Impersonation)- выдача себя за другого с целью получения полномочий и возможности действий от лица другого пользователя;

Подмена стороны аутентификационного обмена (interleaving attack)- участие в процессе аутентификационного обмена между двумя сторонами с целью модификации проходящего через него трафика;

Повторная передача (Replay attack) - аутентификационных данных каким-либо пользователем;

Принудительная задержка (Forced delay)- перехват некоторую информацию и передача ее спустя некоторое время;

Атака с выборкой текста (Chosen-text attack)- перехват аутентификационного трафика и с попыткой получения информации о долговременных криптографических ключах.

Методы аутентификации

Способ	Основное применение	Протоколы	Характеристики
По многоразовым паролям	Аутентификация пользователей (web, ОС, сетевые службы)	HTTP, Forms	Мощность множества, длина
По одноразовым паролям	Аутентификация пользователей (Web-сайты, OTP-токены)	Forms	Мощность множества, длина Длина счетчика
По сертификатам	Аутентификация пользователей в безопасных приложениях; аутентификация сервисов	SSL/TLS	Мощность множества, длина Длина счетчика Вид криптографического протокола
По ключам доступа	Аутентификация сервисов и приложений	-	Характеристики ключа
По токенам	Делегированная аутентификация пользователей; делегированная авторизация приложений	SAML, WS-Federation, OAuth, OpenID Connec	Характеристики ключа

Одноразовые пароли

Одноразовый пароль – это динамическая аутентификационная информация, генерируемая различными способами для однократного использования.

Подходы к построению протоколов аутентификации с одноразовыми паролями

1. **Разделяемые списки одноразовых паролей** (пользователь и система имеют согласованный список паролей, который проходит проверку при очередном сеансе).
2. **Последовательно обновляемые одноразовые пароли** (пользователь и система имеют согласованный список паролей, генерируемый на зашифрованном на ключе, полученном из предыдущего пароля)
3. **Последовательности одноразовых паролей, основанные на однонаправленных функциях** (пользователь и система имеют согласованную однонаправленную функцию, на основании которой формируется пароль).

Система S/KEY (Алгоритм Лампорта)

Система S/KEY имеет статус Internet-стандарта (RFC 1938).

1. Пусть односторонняя функция f функция известна и пользователю, и серверу аутентификации. Пользователь знает секретный ключ S, а также число одноразовых паролей– n.
2. Функция f применяется к ключу S n раз, после чего результат сохраняется на сервере.
3. При первичной регистрации сервер присылает пользователю число n-1
4. Пользователь применяет функцию f к секретному ключу S (n-1) раз и отправляет результат по сети на сервер аутентификации.
5. Сервер применяет функцию f к полученному от пользователя значению и сравнивает результат с ранее сохраненной величиной. В случае совпадения подлинность пользователя считается установленной, сервер запоминает новое значение (присланное пользователем) и уменьшает на единицу счетчик (n).
6. При обнулении счетчика n сервер и пользователь изменяют секретный ключ S.

Классификация и характеристики протоколов аутентификации

- Полная анонимность
- Односторонняя проверка подлинности
- Двусторонняя аутентификация
- Многофакторная аутентификация

Вычислительная эффективность – количество операций, необходимых для выполнения протокола

Коммуникационная эффективность – данное свойство отражает количество сообщений и их длину, необходимую для осуществления аутентификации

Наличие третьей стороны – примером третьей стороны может служить доверенный сервер распределения симметричных ключей или сервер, реализующий дерево сертификатов открытых ключей

Хранения секрета – имеется в виду способ хранения критичной ключевой информации

Стандарты

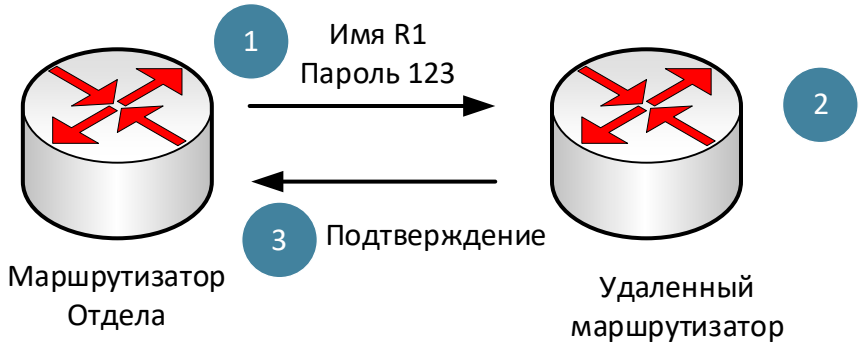
- Стандарт Международной организации по стандартизации и Международной электротехнической комиссии ISO / IEC 9798 - Information technology - Security techniques - Entity authentication mechanisms, состоящий из пяти частей:
- ISO / IEC 9798-1 - «General Model»;
- ISO / EEC 9798-2 - «Mechanisms using symmetric encipherment algorithms»;
- ISO / EEC 9798-3 - «Entity authentication using a public-key algorithm»;
  ISO / IEC 9798-4 - «Mechanisms using a cryptographic check function»;
  ISO / IEC 9798-5 - «Mechanisms using zero knowledge techniques».

Протокол PAP

PPP (англ. Point-to-Point Protocol) — двухточечный протокол канального уровня.

- Управление качеством линии (PPP отключает линк, если количество ошибок превысит заданное значение).
- **Аутентификация:**

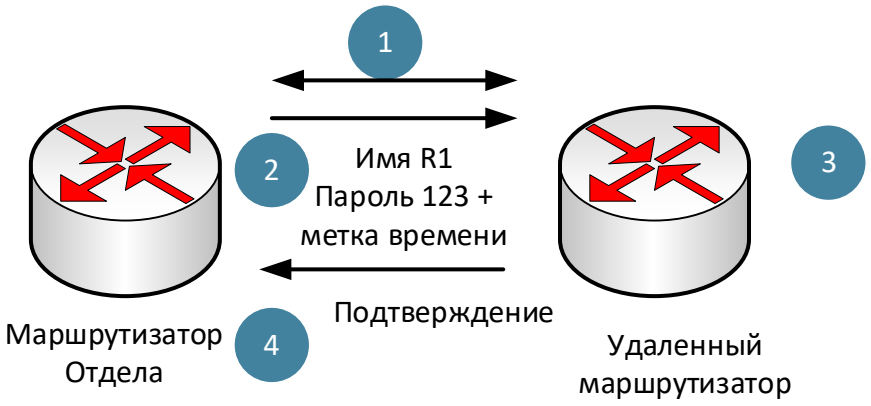
1. Формирование аутентификационных данных и передача их по сети
2. Сравнение аутентификационных данных
3. Передача подтверждения
4. Обмен информации



Стандарт ISO / IEC 9798-2

Односторонняя аутентификация, основанная на метке времени, при этом необходимо, чтобы разница между метками времени отправителя и получателя укладывалась в определенный интервал времени – окно принятия (acceptance window).

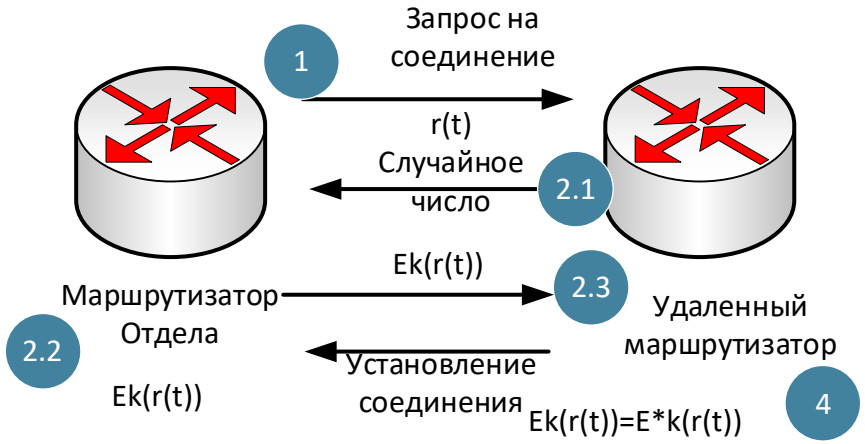
1. Клиент устанавливает соединение
2. Клиент получает метку времени с системного таймера и криптографически складывает с аутентификационными параметрами.
3. Сервер получает текущее время со своих локальных часов и сравнивает с величиной метки времени, полученной от партнера.



Стандарт ISO / IEC 9798-2

Односторонняя аутентификация с использованием случайных чисел.

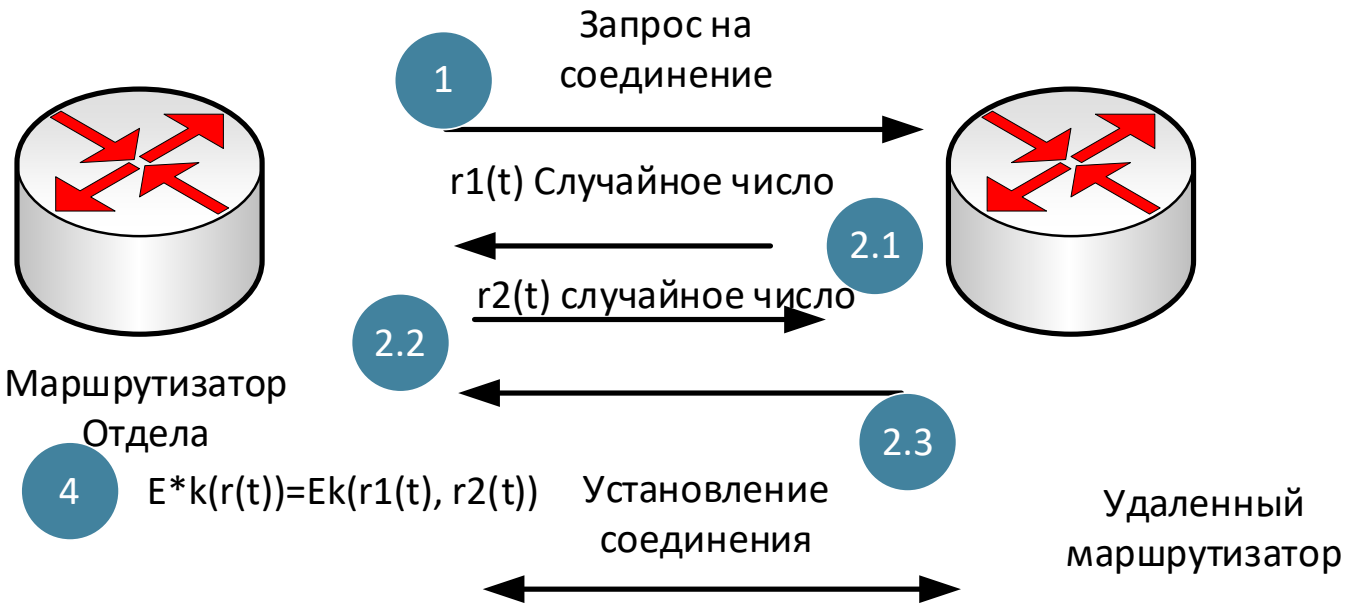
1. Клиент отправляет запрос соединение
2. Клиент получает случайное число, сгенерированное с помощью генератора псевдослучайных чисел. Вычисляет ответ на него – зашифровывает с помощью алгоритма симметричного шифрования полученное случайное число и (если необходимо) идентификатор.
3. Сервер расшифровывает полученный шифртекст и проверяет структуру запроса. Если она верна, соединение устанавливается.



Стандарт ISO / IEC 9798-2

Взаимная аутентификация с использованием случайных чисел. Оба устройства проверяют друг друга.

Протокол допускает замену шифра на хеш-функцию с ключом (стандарте ISO / IEC 9798-4). Для повышения стойкости протокола к передаваемым сообщениям можно добавить метки времени.

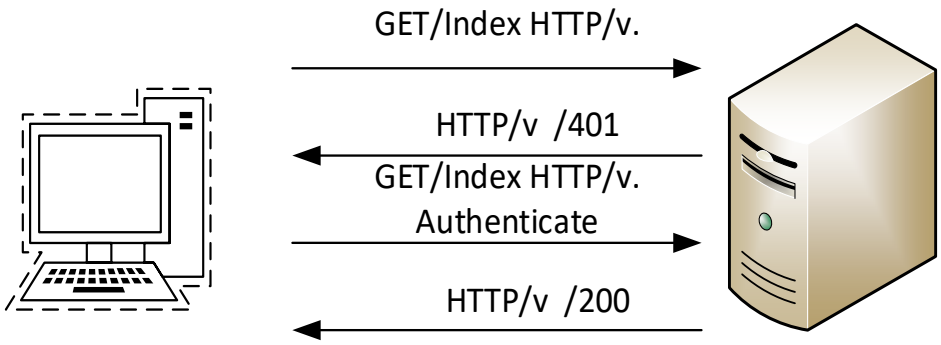


Протокол CHAP

- Основан на шифровании с помощью односторонней хэш-функции и по существу является односторонним, т. е. не сопровождается обратным преобразованием — расшифровыванием на приемной стороне. Обе стороны (отправитель и получатель) используют одну и ту же процедуру одностороннего шифрования.

HTTP

1. Сервер, при обращении клиента к защищенному ресурсу, отсылает HTTP статус “401 Unauthorized” и добавляет заголовок “WWW-Authenticate” с указанием схемы и параметров аутентификации.
2. Браузер, при получении такого ответа, автоматически показывает диалог ввода username и password. Пользователь вводит детали своей учетной записи.
3. Во всех последующих запросах к этому веб-сайту браузер автоматически добавляет HTTP заголовок “Authorization”, в котором передаются данные пользователя для аутентификации сервером.
4. Сервер аутентифицирует пользователя по данным из этого заголовка. Решение о предоставлении доступа (авторизация) производится отдельно на основании роли пользователя, ACL или других данных учетной записи.

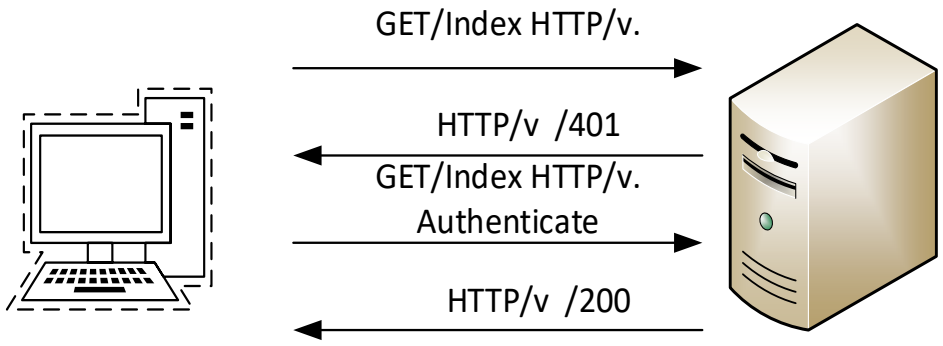


HTTP

Схемы аутентификации:

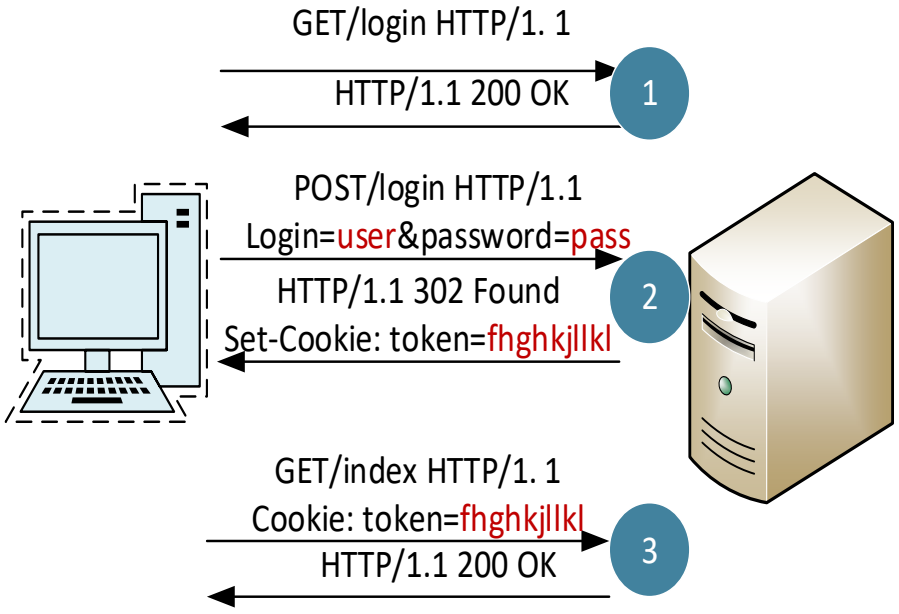
Basic —username и password пользователя передаются в заголовке Authorization в незашифрованном виде (base64-encoded). Требует использование HTTPS.

Digest — challenge-response-схема, при которой сервер посылает уникальное значение nonce, а браузер передает MD5 хэш пароля пользователя, вычисленный с использованием указанного nonce. Более безопасная альтернатива Basic, но подвержена man-in-the-middle attacks (с заменой схемы на basic).

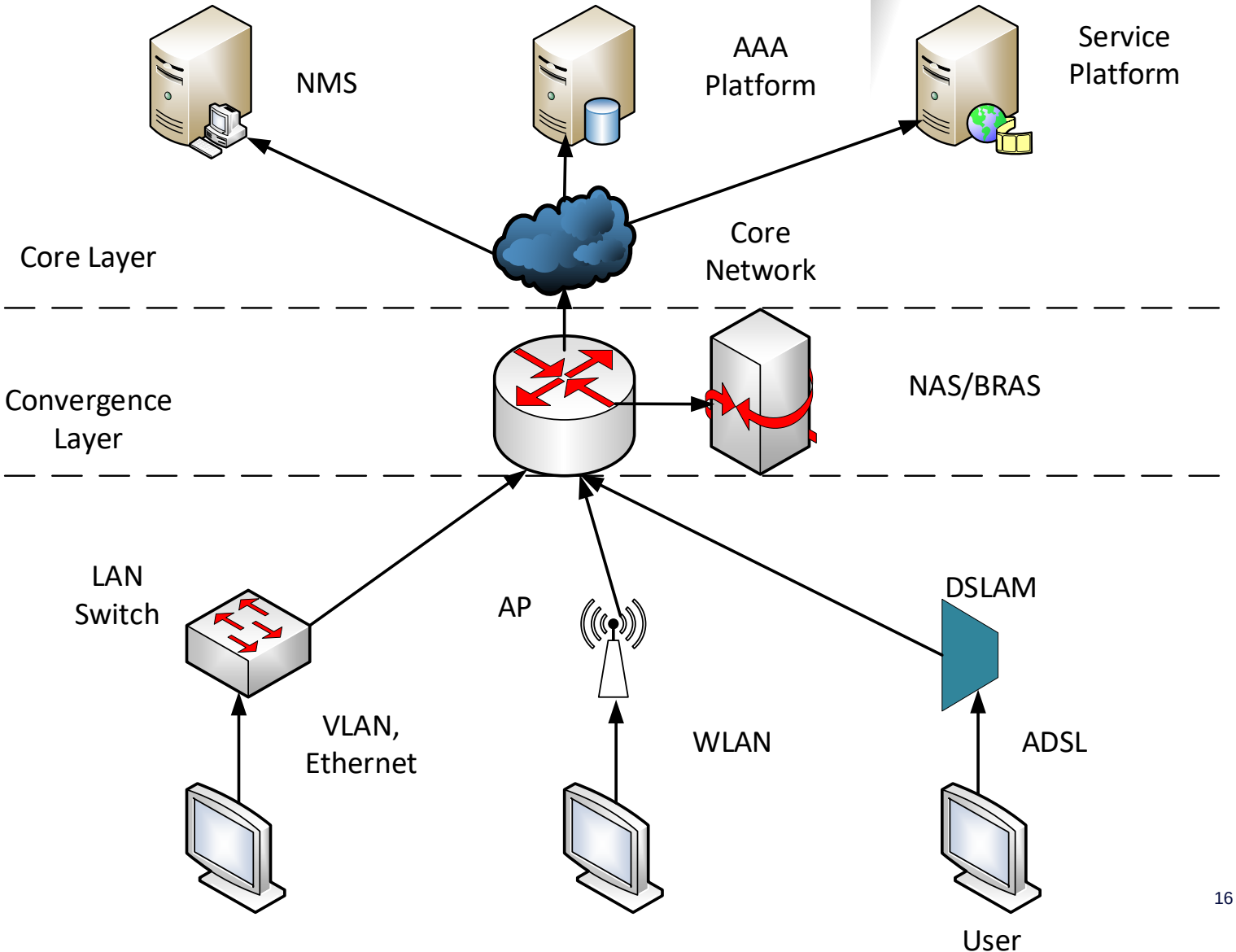


HTTP

1. Запрос на получение формы.
2. В HTML-форму, вводиться username/password и отправить их на сервер через HTTP POST для аутентификации. В случае успеха веб-приложение создает session token, который обычно помещается в browser cookies.
3. При последующих веб-запросах session token автоматически передается на сервер и позволяет приложению получить информацию о текущем пользователе для авторизации запроса.



NAS


NMS
AAA Platform
Service Platform
Core Layer
Core Network
Convergence Layer
NAS/BRAS
LAN Switch
AP
DSLAM
VLAN, Ethernet
WLAN
ADSL
User

BRAS

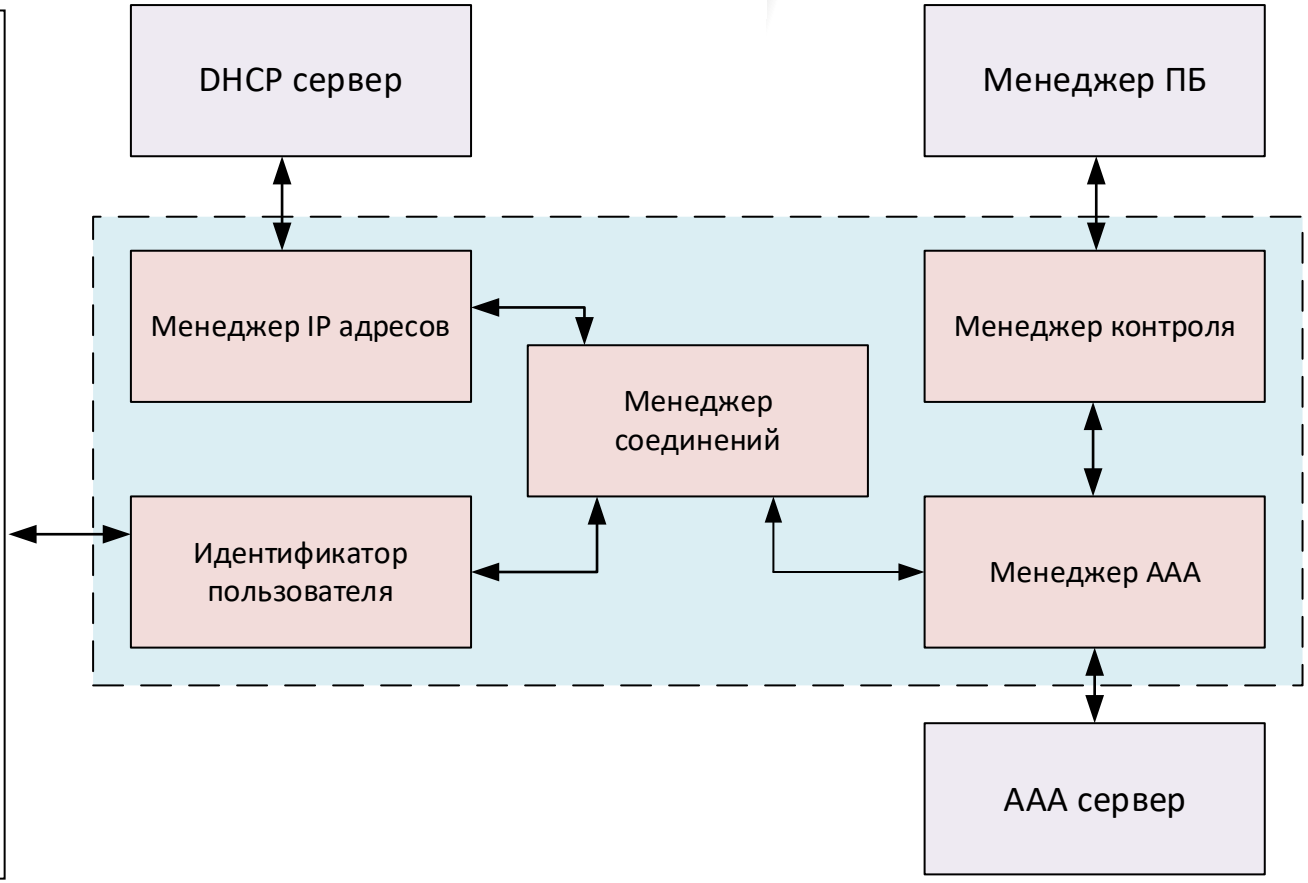
Задачи:

Аутентификация

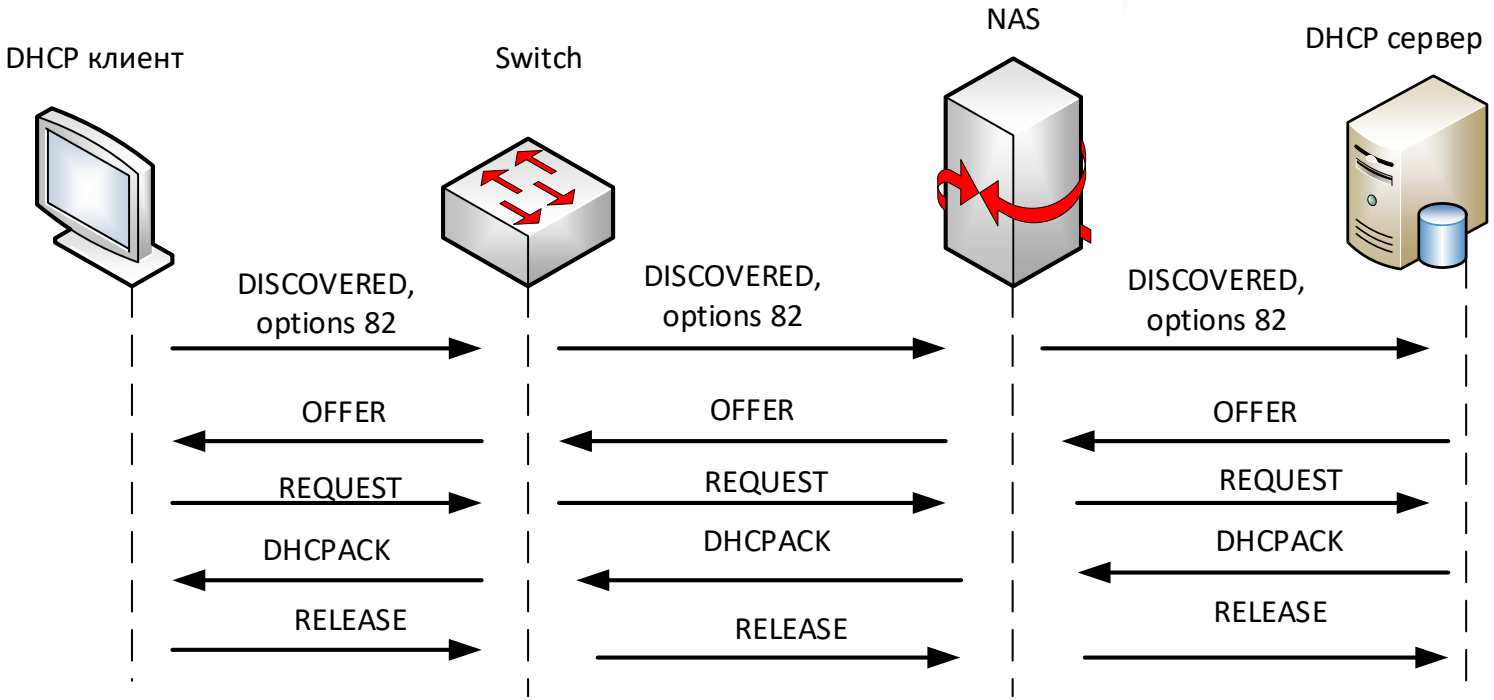
Выделение IP адреса

Управление ресурсами

Обеспечение безопасного соединения



DHCP



- Автоматическое выделение IP (выделяется один и тот же IP)
- Динамическое выделение из пула адресов
- Ручное выделение (но настройки пересылаются по сети)
- Особенность: DHCP не может отозвать IP адрес сам

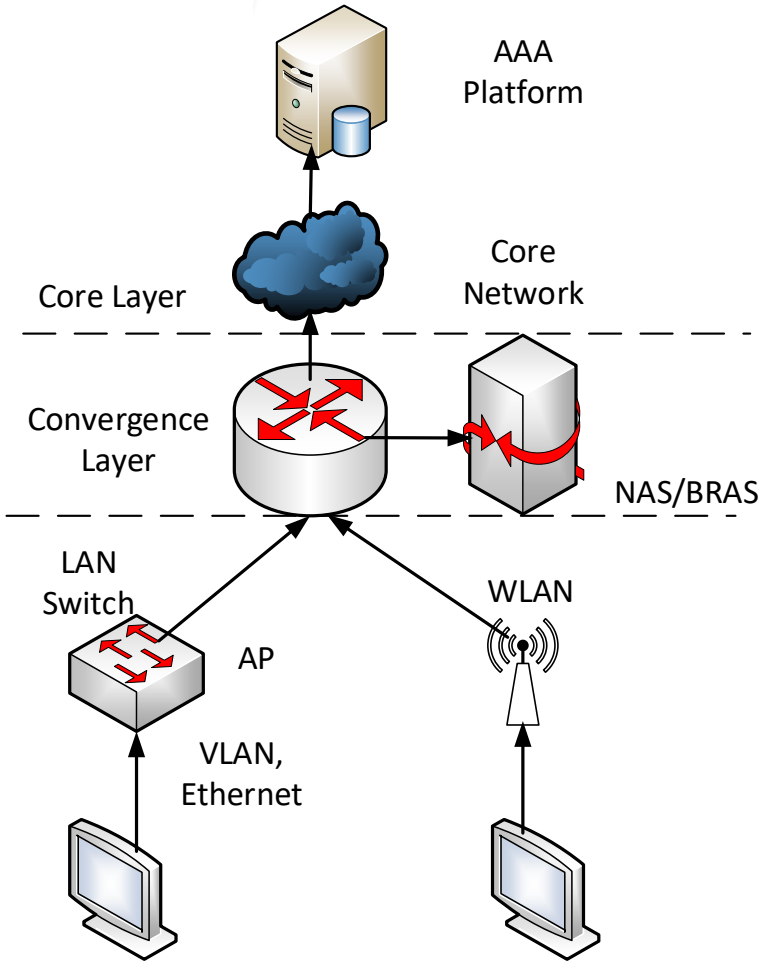
RADIUS

Назначение

- Создание и хранение учётных записей пользователей
- Ручная и автоматическая блокировка учётной записи абонента по достижении заданного критерия или лимита
- Сбор и анализ статистической информации о сессиях пользователя и всей обслуживаемой
- Мониторинг
- Создание, печать и отправка счетов к оплате

UDP

- Проблемы доставки решаются дублированием UDP пакетов
- Использование протокола TCP увеличивает длительность аутентификации при значительном количестве пользователей



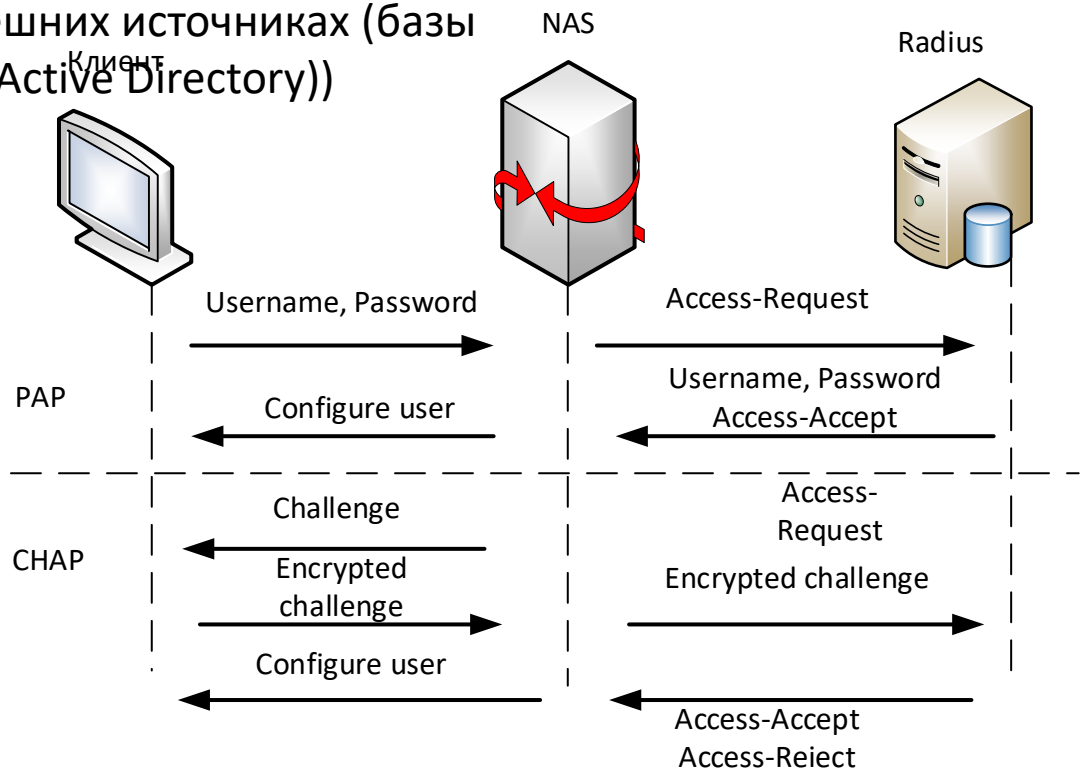
RADIUS. Аутентификация

Процесс:

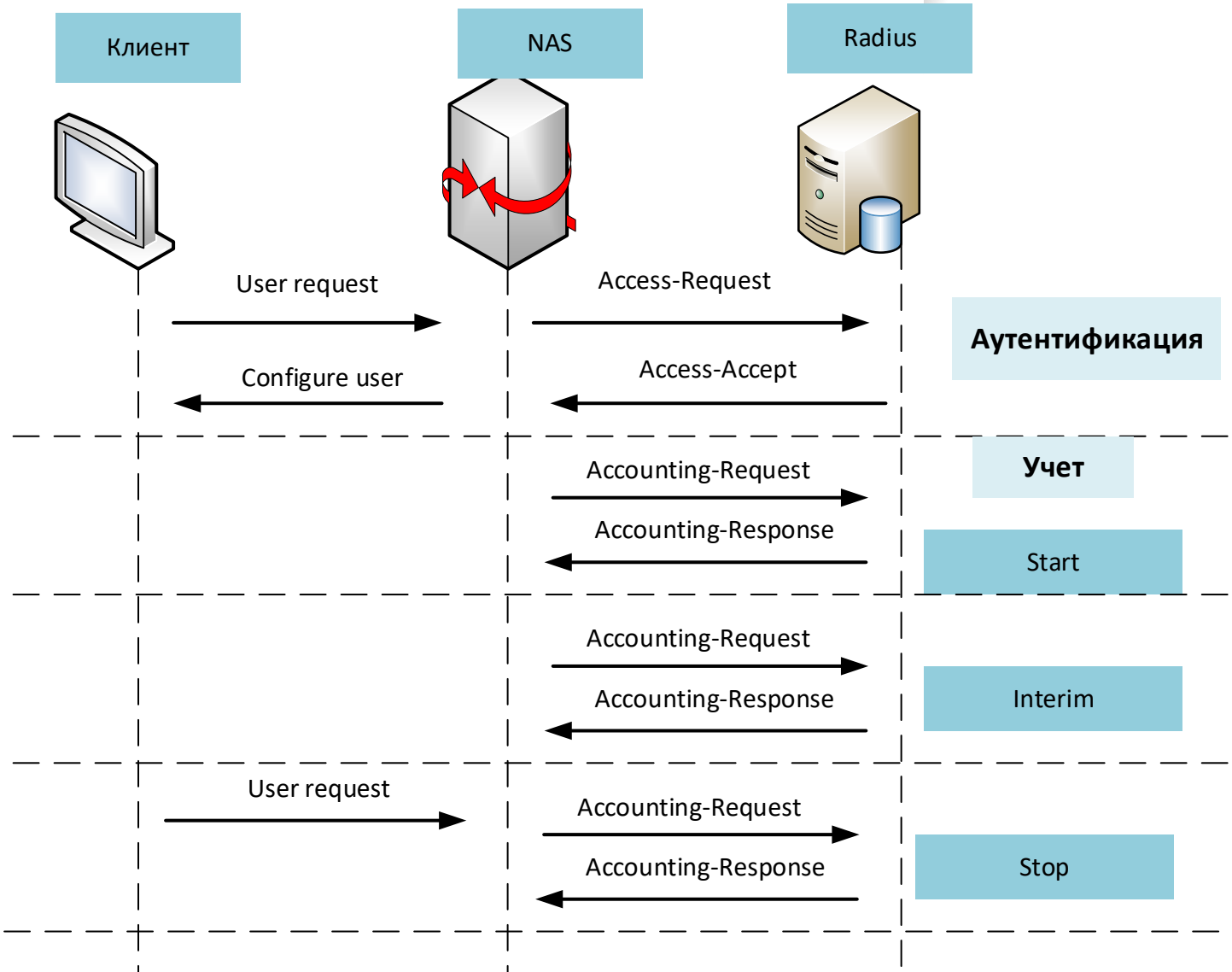
Аутентификация: CHAP, PAP, MD5, EAP

Авторизация

Учет (учетные данные могут хранится локально или во внешних источниках (базы SQL, Kerberos, LDAP, Active Directory))



RADIUS. Авторизация

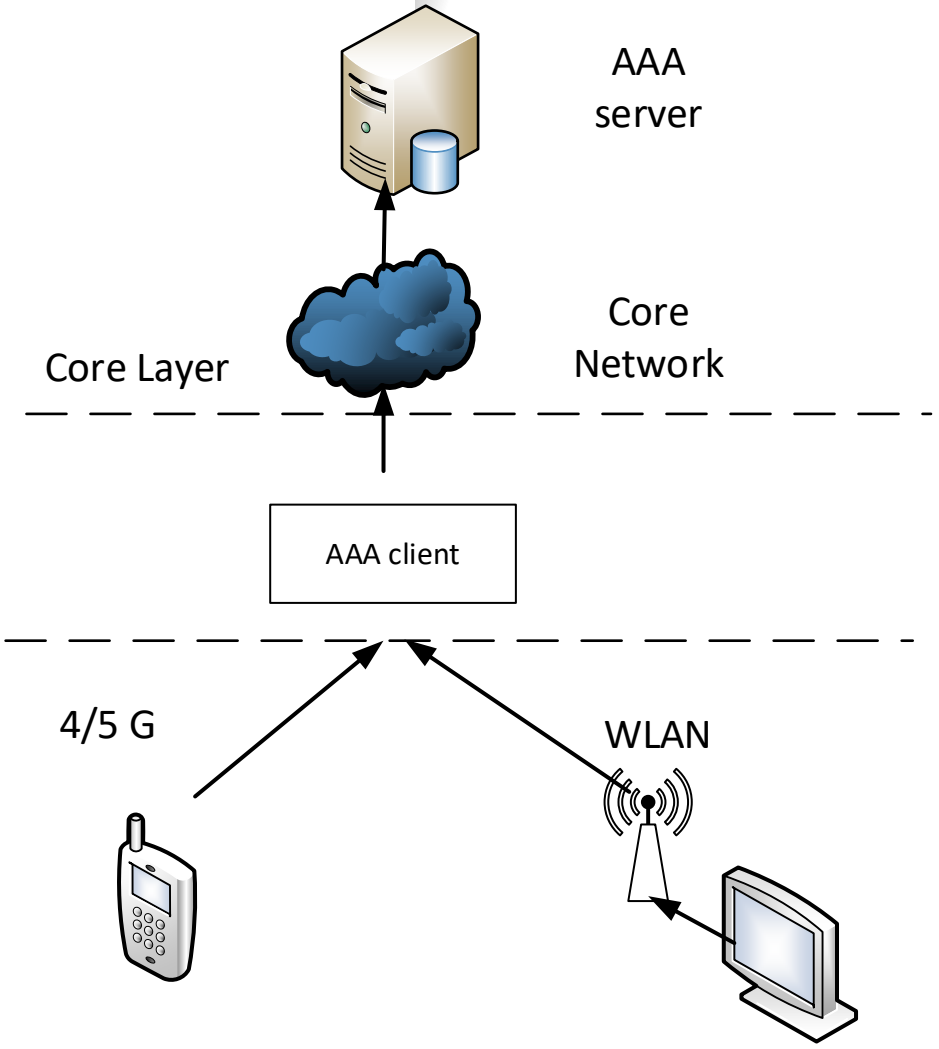


Diameter (RFC 3588)

Характеристики:

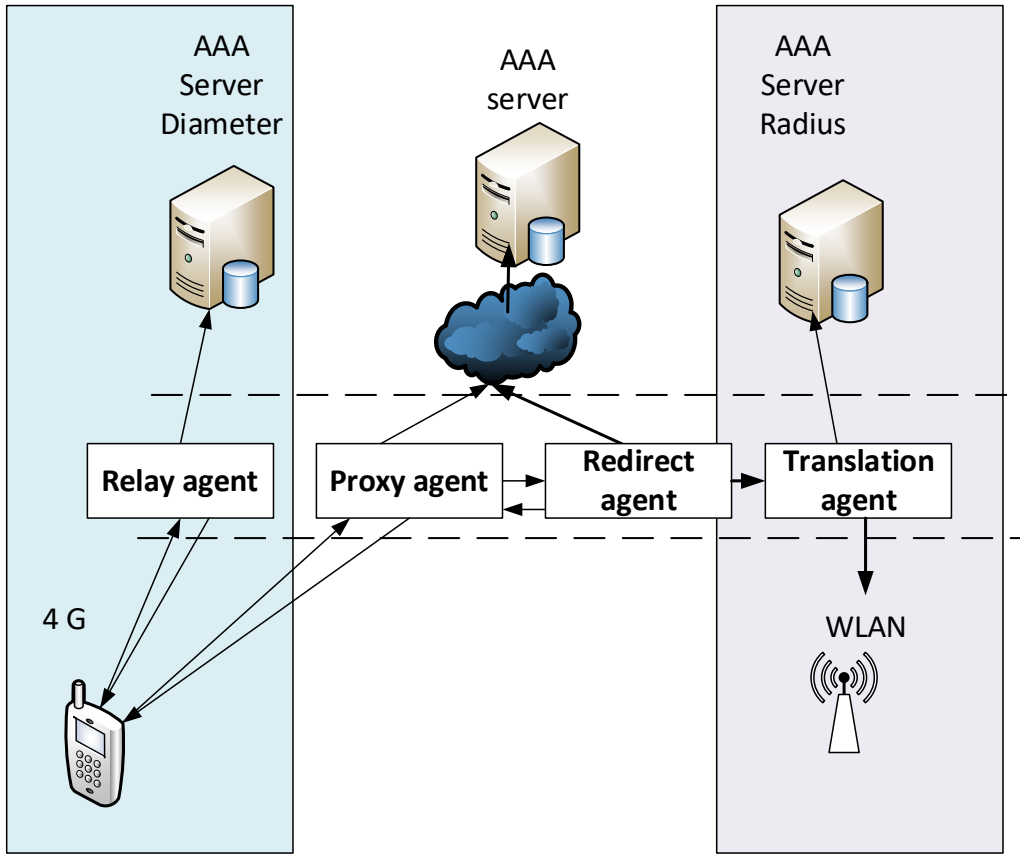
1. Отказоустойчивость
2. Использование протоколов IPSec
3. Сети позволяют работать с агентами, которые перемещаются из одной сети в другую и при этом продолжать использовать один и тот же IP-адрес.

- **Аутентификация**
  - PAP, CHAP, EAP
- **Авторизация**
  - Перенаправление, безопасные прокси, ретрансляция (relay), посредничество (broker)
  - Согласование состояния
  - Отключение по собственной инициативе
- **Учет**
  - Отчетность, мониторинг событий.



Diameter. Виды агентов

- **Relay agent** для перенаправления сообщения соответствующему адресату в зависимости от информации, содержащейся в сообщении. Объединяет запросы от различных областей (или регионов и устраняет обременительную настройку серверов при каждом изменении Diameter-сервера.
- **Proxy** может изменить содержимое сообщения и предоставляет дополнительные службы (контроль доступ к определенным сервисам).
- **Redirect agent** выступает в роли централизованного репозитория конфигураций
- **Translation Agents** преобразование сообщения из одного AAA-протокола в другой (преобразование из RADIUS в Diameter).



Kerberos

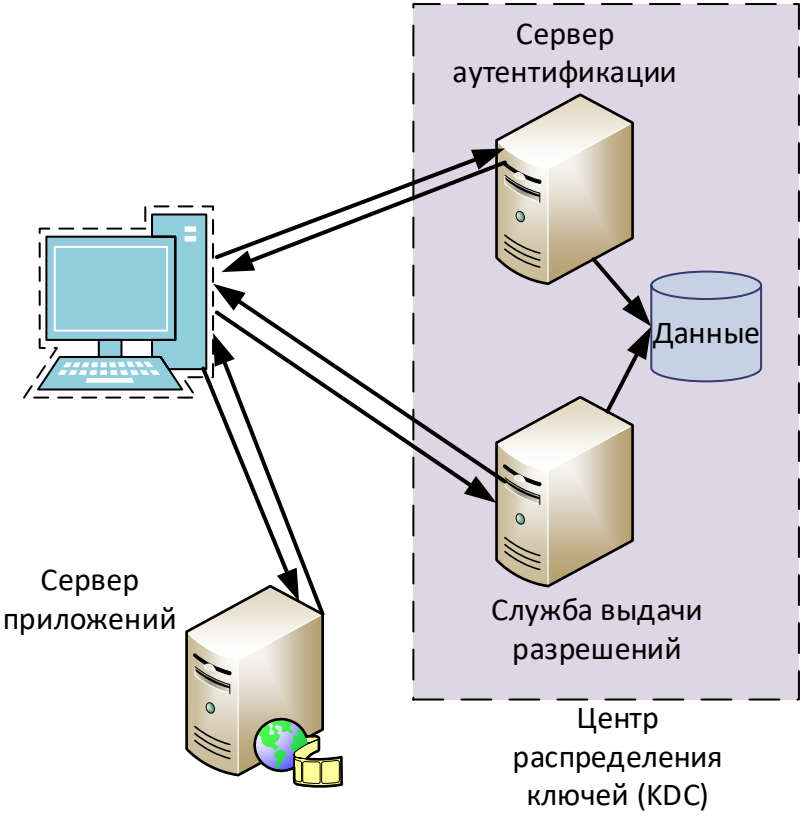
Задачи:

1. Взаимная аутентификация
2. Обмен в незащищенных каналах связи с использованием ассиметричной криптографии

Содержит два логических компонента:

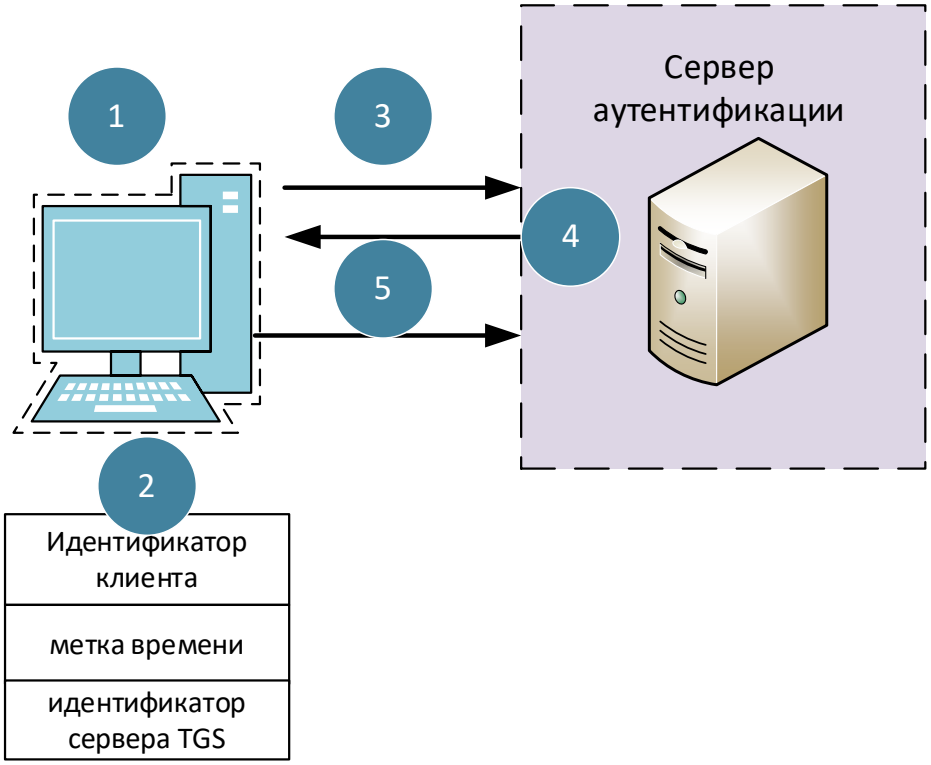
1. Сервер аутентификации (CA)
2. Сервер выдачи билетов (TGS — Ticket Granting Server).

Эти компоненты могут быть единой программой, которая запускается на центре распределения ключей (ЦРК — содержит базу данных логинов/паролей для пользователей и сервисов использующих Kerberos).



Kerberos. Аутентификация клиента

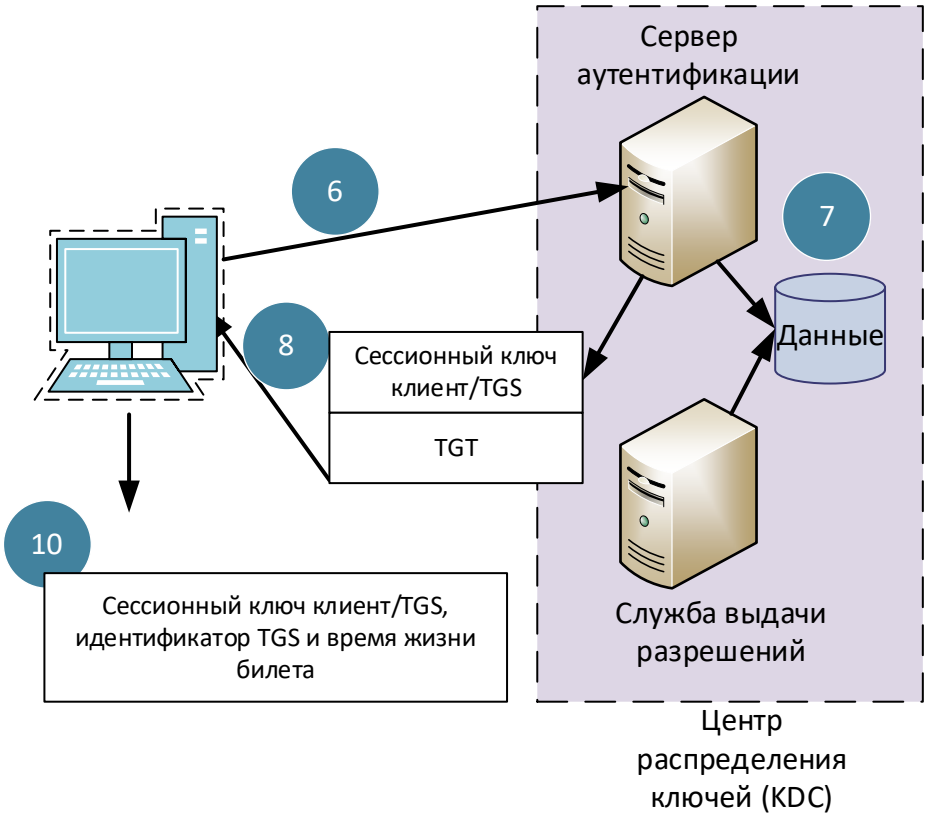
1. Пользователь с использованием имени и пароля выполняет вход в систему на клиентском хосте.
2. Клиентская машина выполняет над паролем одностороннюю функцию (обычно хэш), и результат становится секретным ключом клиента/пользователя.
3. Клиент отсылает запрос (AS_REQ) на CA для получения аутентификационных верительных данных и последующего их предоставления TGS серверу (впоследствии он будет их использовать для получения билетов без дополнительных запросов на применение секретного ключа пользователя)
4. Если политика ЦРК требует предварительной аутентификации, то пользователь получает сообщение KRB_ERROR, в ответ на которое посылает повторный запрос, но с уже данными для установления подлинности (5).



Kerberos

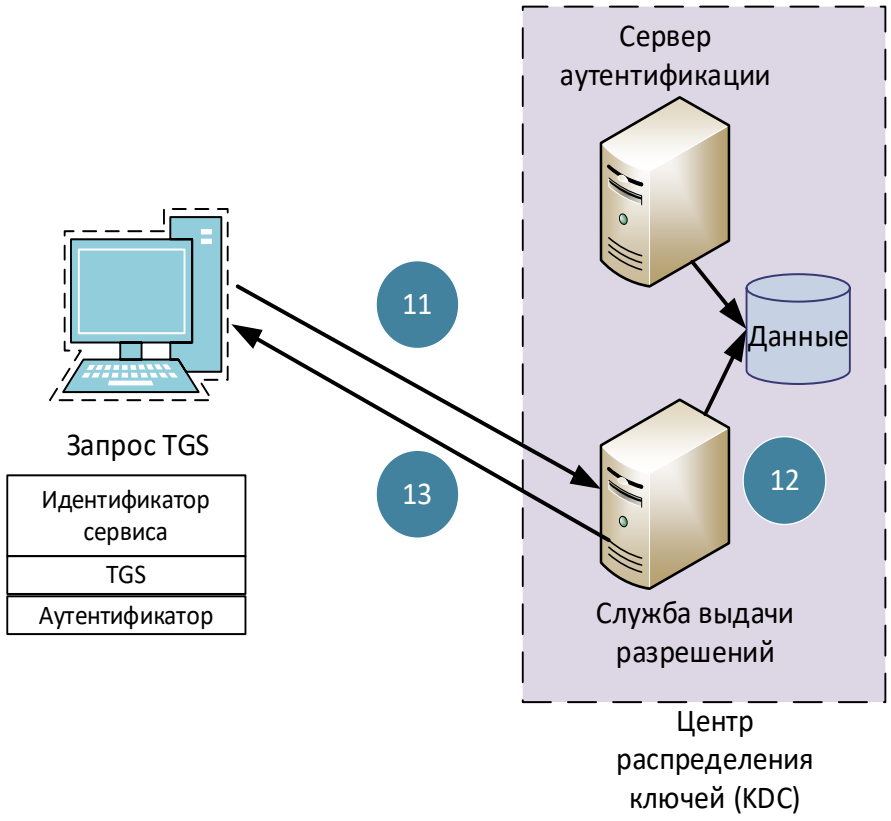
7. Если клиент аутентифицирован CA проверяет, есть ли такой клиент в базе.
8. CA отправляет сообщение (AS_REP).
9. Если же нет, то клиент получает новое сообщение, говорящее о произошедшей ошибке.

10 Получив сообщение, клиент расшифровывает свою часть для получения Сессионного Ключа Клиент/TGS. Этот сессионный ключ используется для дальнейшего обмена с сервером TGS. (Важно: Клиент не может расшифровать TGT, так как оно зашифровано секретным ключом TGS) В этот момент у пользователя достаточно данных, чтобы авторизоваться на TGS.



Kerberos

11. Для запроса сервиса клиент формирует запрос на TGS (TGS_REQ).
12. TGS извлекает из него TGT и расшифровывает его используя секретный ключ TGS. Это дает ему Сессионный Ключ Клиент/TGS. Им он расшифровывает аутентификатор.
13. Затем он генерирует сессионный ключ клиент/сервис и посылает ответ (TGS_REP) включающий:
    - Билет сервиса (который содержит ID клиента, сетевой адрес клиента, метку времени ЦРК, время действия билета и Сессионный Ключ клиент/сервис) зашифрованный секретным ключом сервиса.
    - Сессионный ключ клиент/сервис, идентификатор сервиса и время жизни билета, зашифрованные на Сессионном Ключе Client/TGS.



Kerberos

14. После получения TGS_REP посылает сообщение содержащее:
    - Зашифрованный билет сервиса полученный ранее.
    - Новый аутентификатор, зашифрованный на сессионном ключе клиент/сервис, и включающий ID клиента и метку времени.
15. Сервер расшифровывает билет используя свой секретный ключ и получает сессионный ключ клиента. Используя новый ключ, он расшифровывает аутентификатор и посылает клиенту :
    - Метку времени, указанную клиентом + 1, зашифрованную на сессионном ключе клиент/сервис.
16. Клиент проверяет, действительно ли метка времени корректно обновлена. Если это так, то клиент может доверять серверу и может начать посылать запросы на сервер.
17. Сервер предоставляет клиенту требуемый сервис.