Автентикация, оторизация, акаунтинг (AAA)

Основни моменти

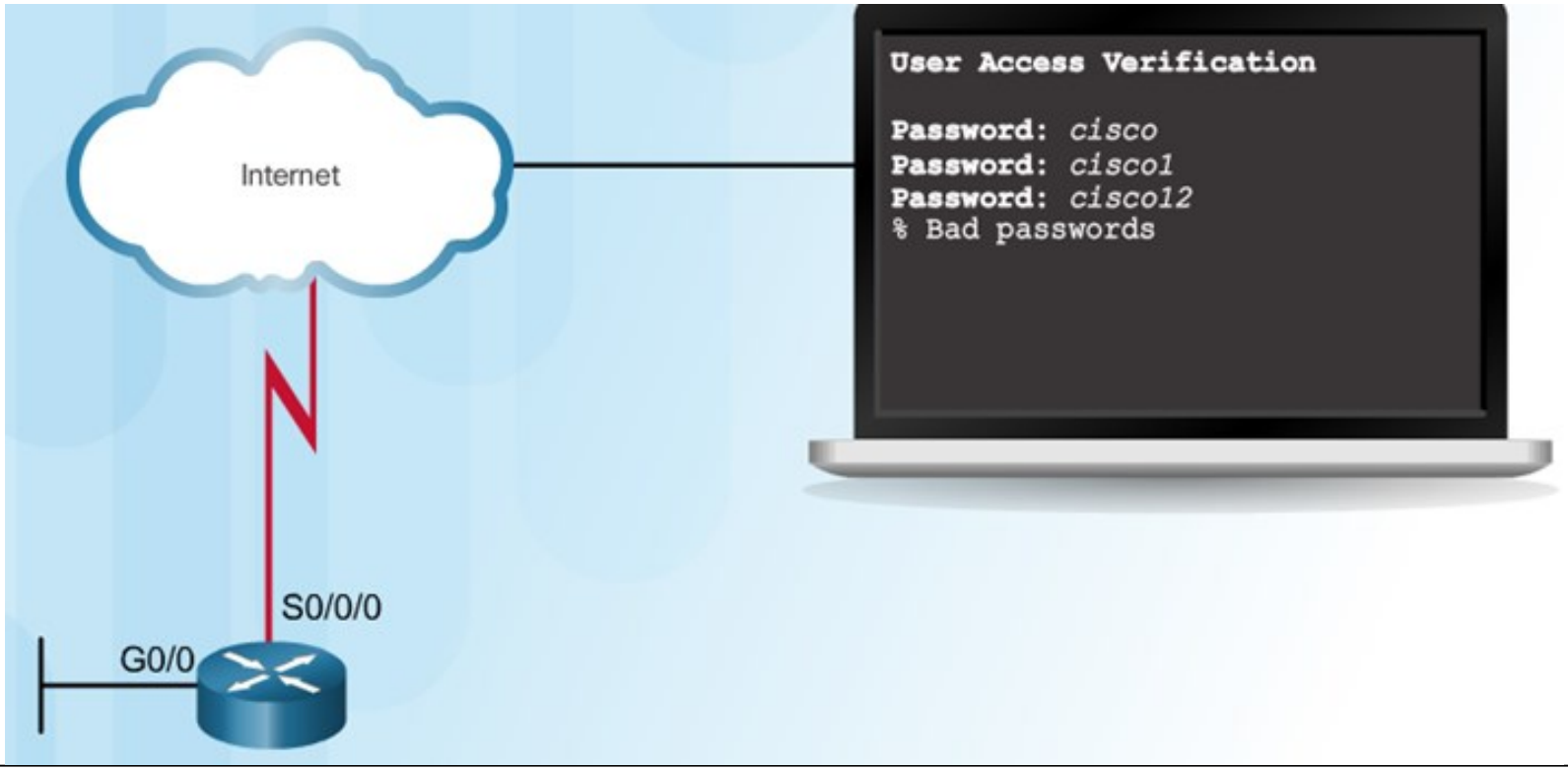
- Автентикация
- Оторизация
- Акаунтинг
- AAA сървър

Аспекти на сигурността

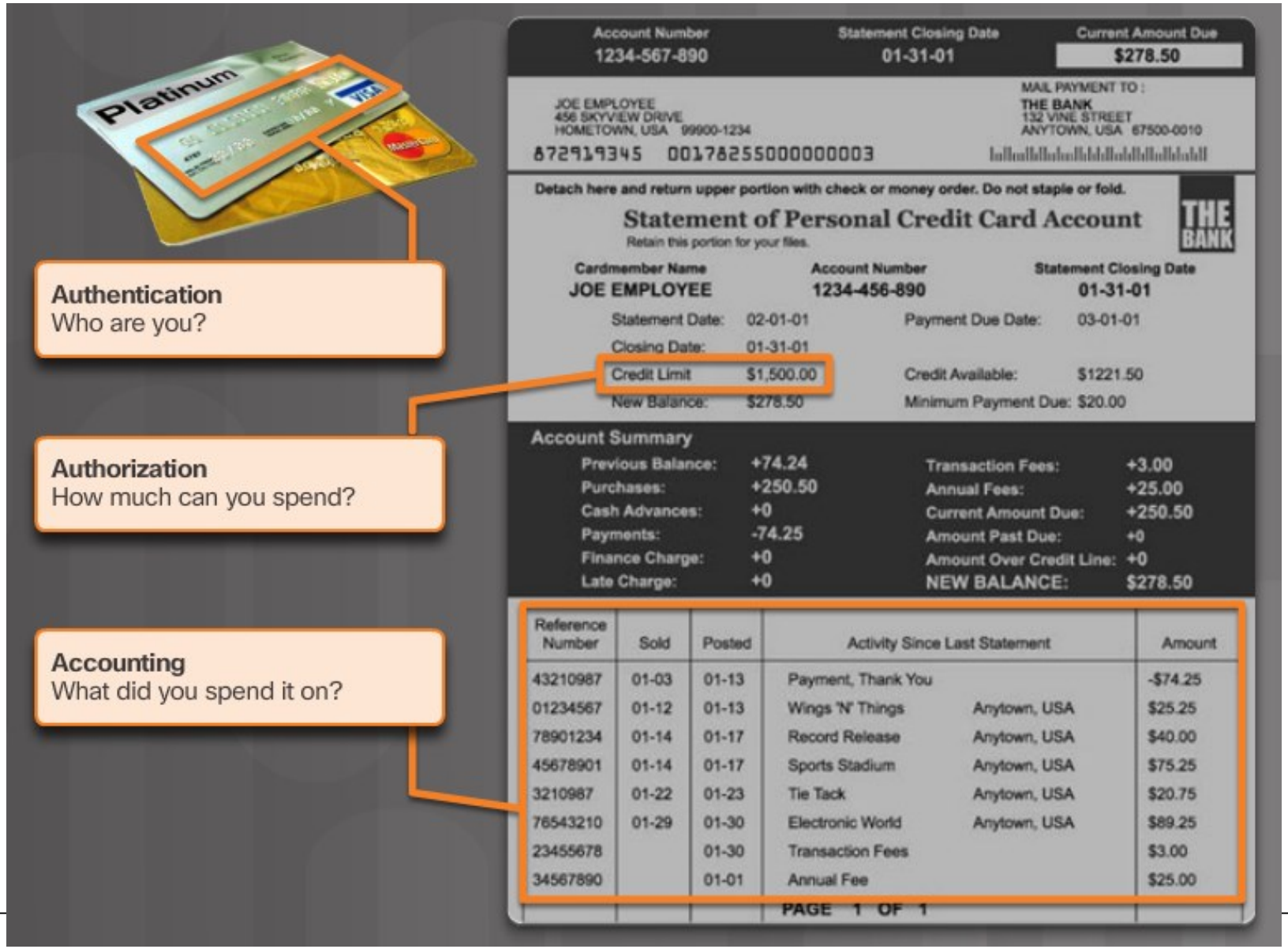
- Регулиране на трафика:
  - Firewalls
- Сигурни връзки:
  - VPN тунели
- За конкретни сайтове в мрежата:
  - Контрол на достъпа
  - Автентикация
  - Оторизация
  - Акаунтинг
  - Мониторинг и контрол

Автентикация без AAA

- Telnet достъпът може да се атакува с Brute-Force атака
- SSH също



Компоненти на AAA

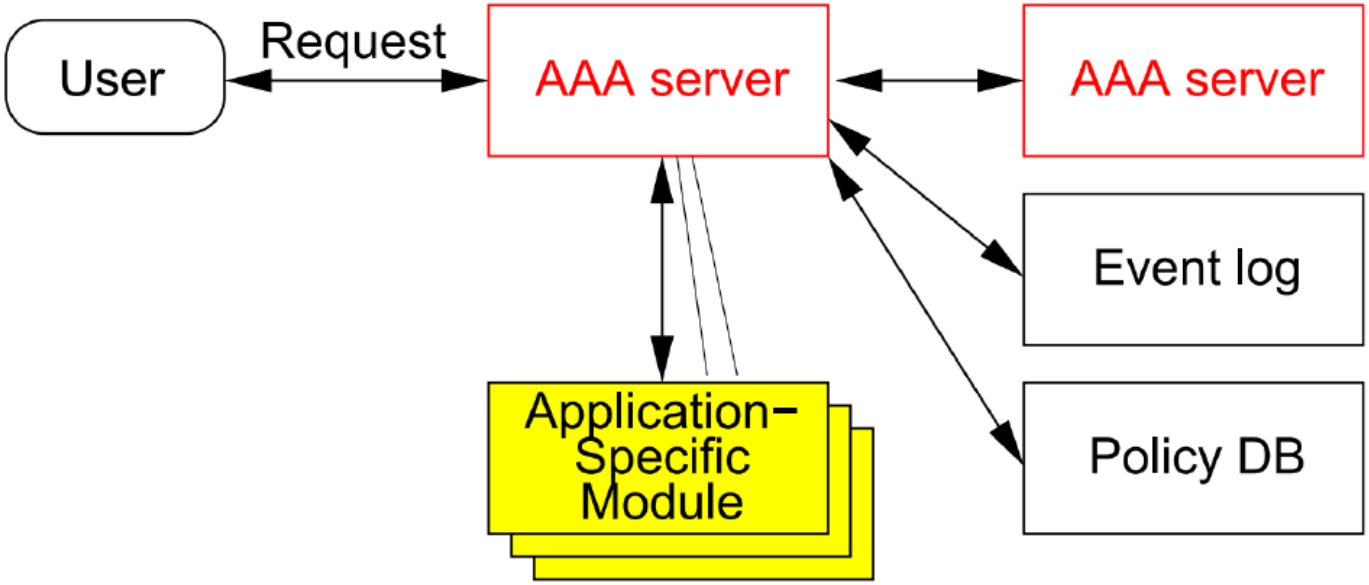


Автентикация/оторизация/акаунтинг

- **Автентикацията** удостоверява устройството или потребителят дали са законни.
- **Оторизацията** позволява или забранява на законните потребителите достъп до определени области и програми в мрежата.
- Автентикацията предхожда оторизацията.
- **Акаунтинг**- събиране на информация за употребата на ресурси

AAA Системи

- Цел: да регулира използването на даден домейн.
  - Достъп до ресурсите на собствения домейн
  - Достъп до ресурси на друг домейн - потребителят трябва да се оторизира за разрешение за искания ресурс.



AAA

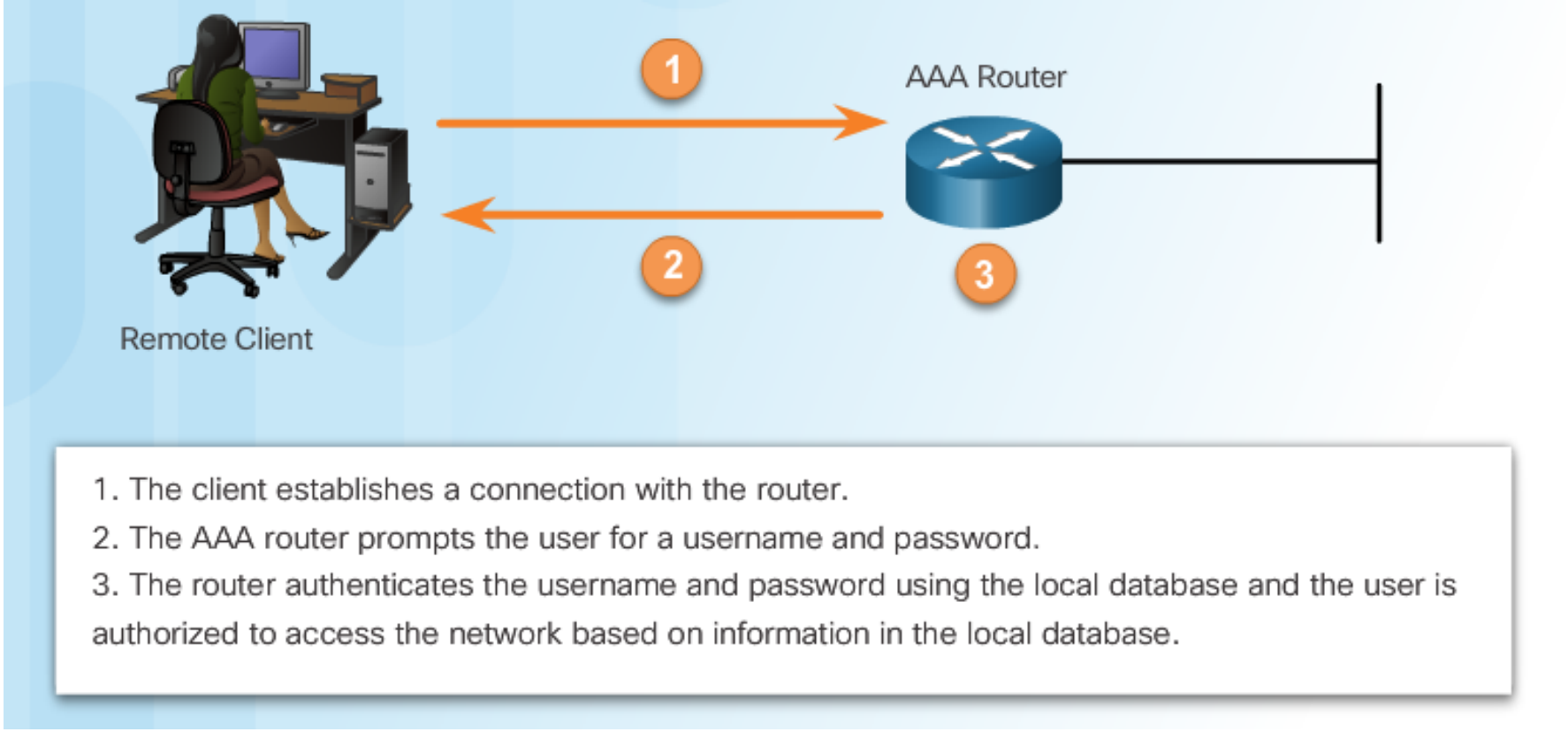
- Автентикация
- Оторизация
- Акаунтинг

Същност на автентикацията

- Проверка на декларираната идентичност с:
  - UserID/password
  - Passphrase
  - Certificate
  - Хардуерни устройства-dongle, smartcard...
  - Биометрична информация
- Тази информация трябва да се предава:
  - Сигурно - трети страни да не я придобият
  - Защитено - от replay атаки

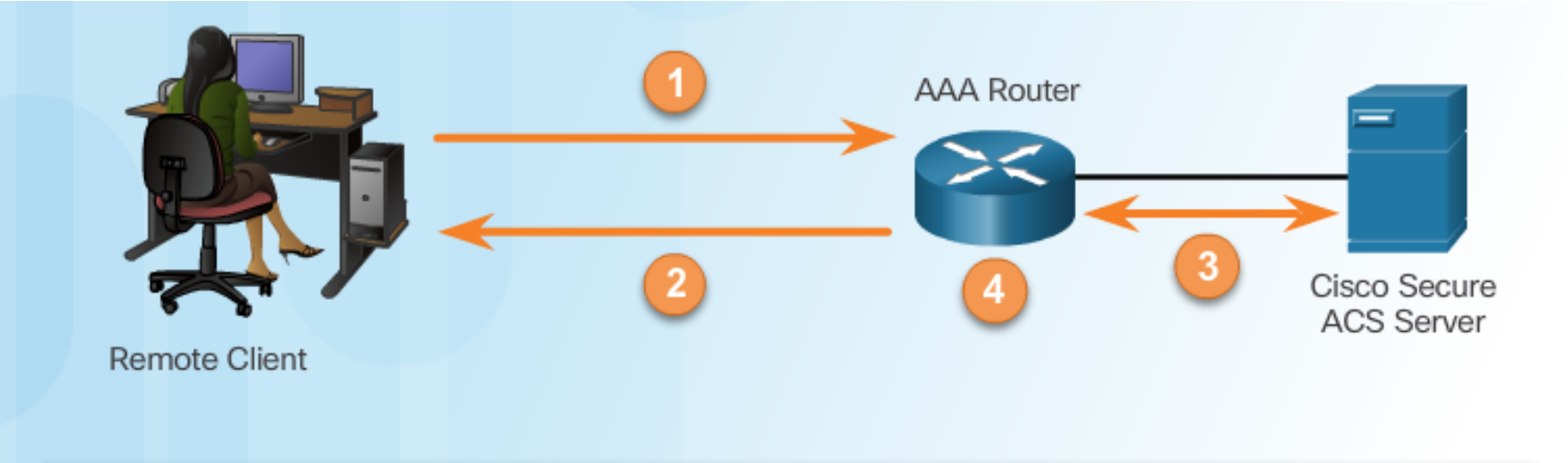
Автентикация

AAA, реализирана с локална БД за автентикация



Автентикация

AAA, реализирана с базирана на сървърна БД за автентикация



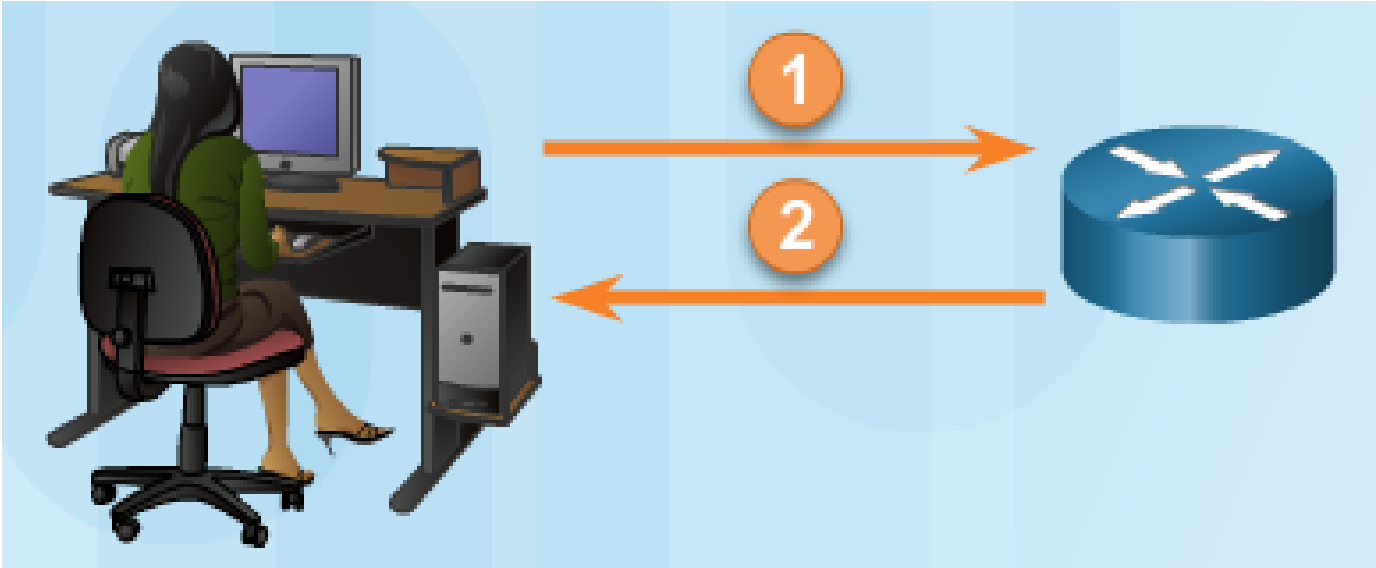
1. The client establishes a connection with the router.
2. The AAA router prompts the user for a username and password.
3. The router authenticates the username and password using a remote AAA server.
4. The user is authorized to access the network based on information on the remote AAA Server.

Автентикационни методи

- С локална парола за логване към устройството
- С парола, записана в БД:
  - Локално - на домейна, в който е логнат.
  - Частично или пълно - на домейна, към който е отправено искане.
  - Чрез включване на трети лица - брокери.

Локална автентикация

- Username & password на самата локална машина
- Криптирано/некриптирано съхранение на пароли в локална БД



Пример: автентикация в Linux

- Паролата се хешира с псевдослучайно число – ***salt***.
- Типично за Linux се използва MD5 хеш алгоритъм и 48 бита salt стойност.

Пример: автентикация в Linux

```
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/:/bin/false
news:x:9:13:news:/usr/lib/news:/bin/false
uucp:x:10:14:uucp:/var/spool/uucppublic:/bin/false
operator:x:11:0:operator:/root:/bin/bash
games:x:12:100:games:/usr/games:/bin/false
ftp:x:14:50::/home/ftp:/bin/false
```

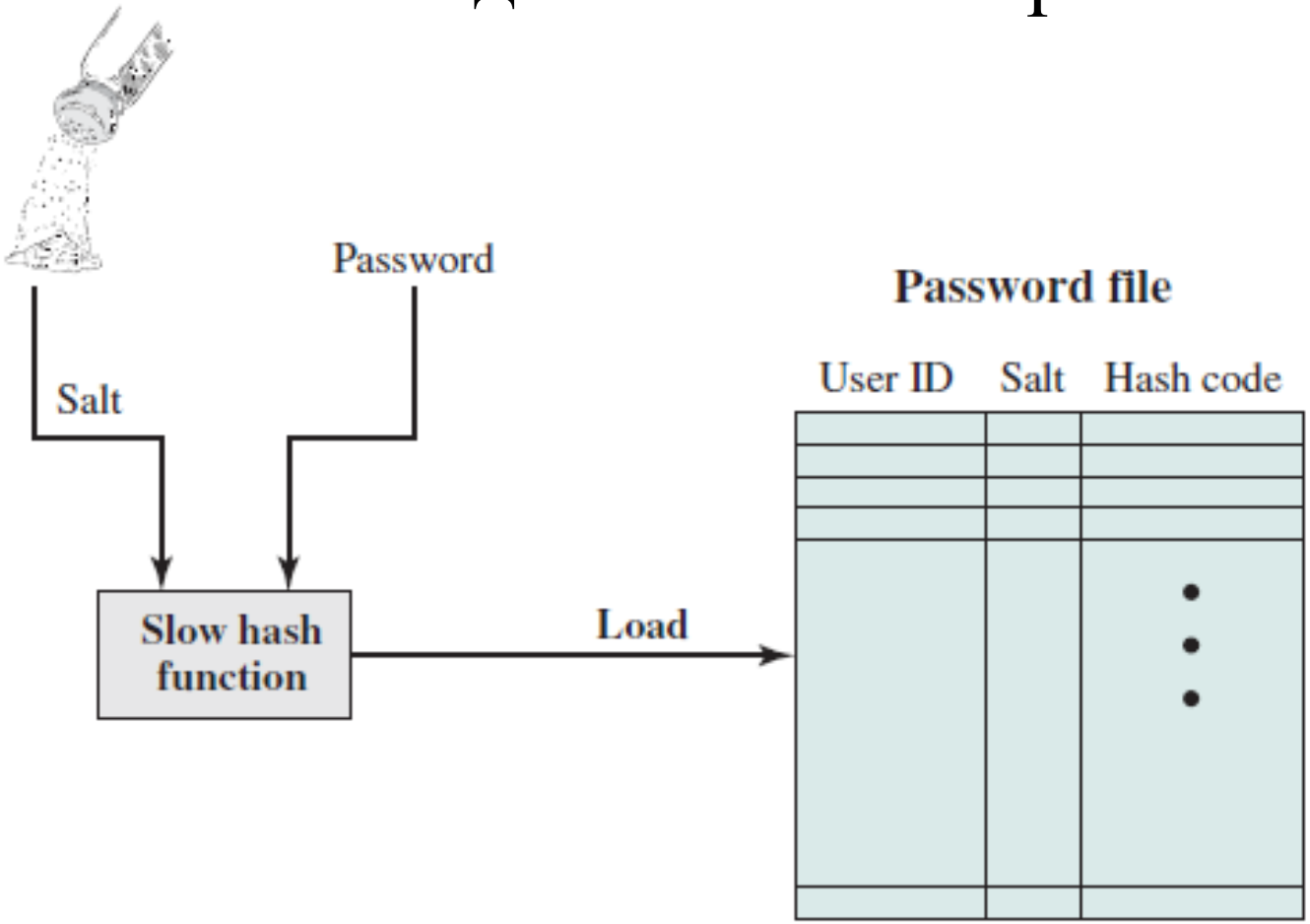
/etc/passwd

```
halt:*:9797:0:::::
mail:*:9797:0:::::
news:*:9797:0:::::
uucp:*:9797:0:::::
operator:*:9797:0:::::
games:*:9797:0:::::
ftp:*:9797:0:::::
smmsp:*:9797:0:::::
mysql:*:9797:0:::::
rpc:*:9797:0:::::
sshd:*:9797:0:::::
```

/etc/shadow

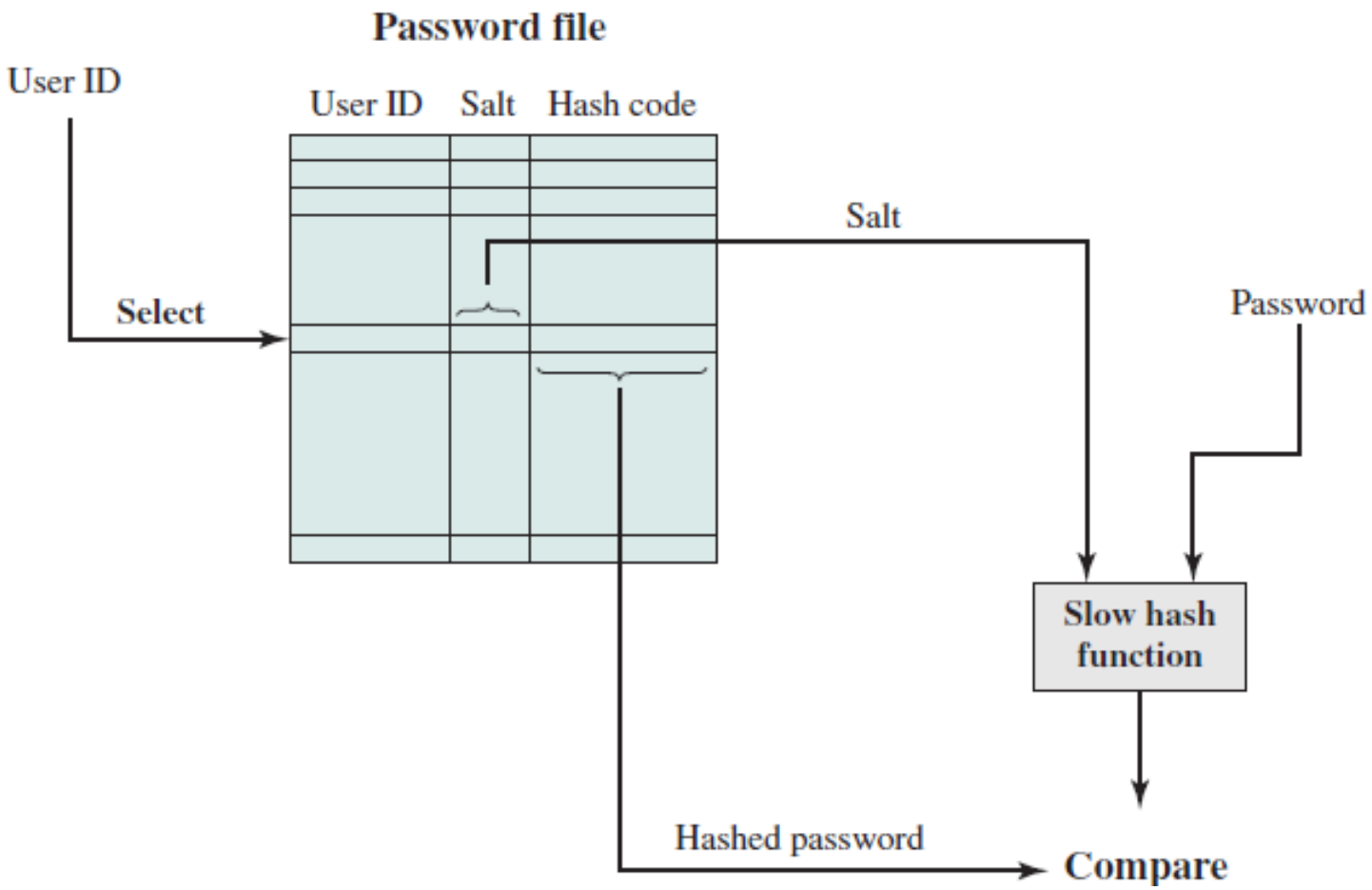
Пример: автентикация в Linux

Въвеждане на нова парола



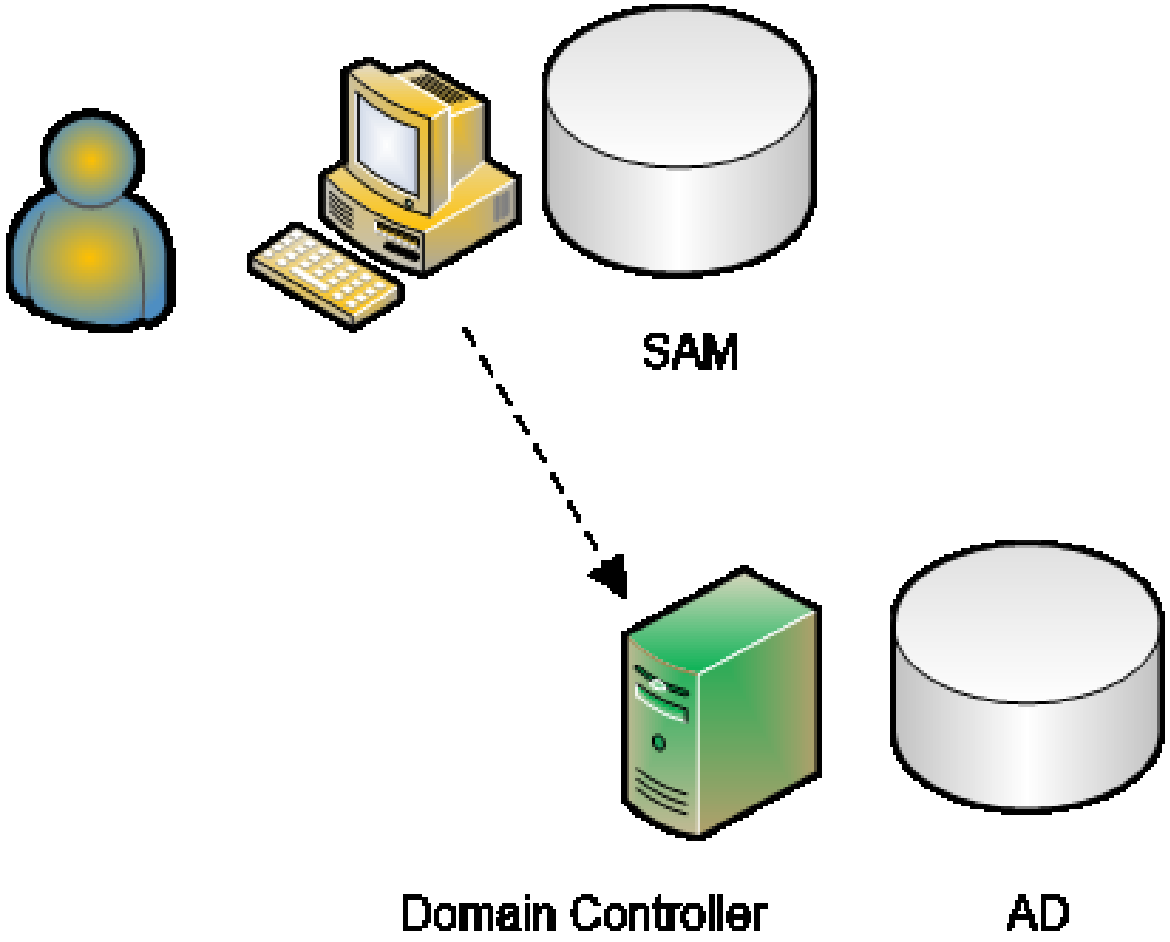
Пример: автентикация в Linux

Проверка на парола



Пример: автентикация в Windows

Локален и домейн logon



Пример: автентикация в Windows

Ресурсни записи за DC в DNS:

_ldap._tcp.pdc.example.com. IN SRV 10 50 389 ds1.example.com.

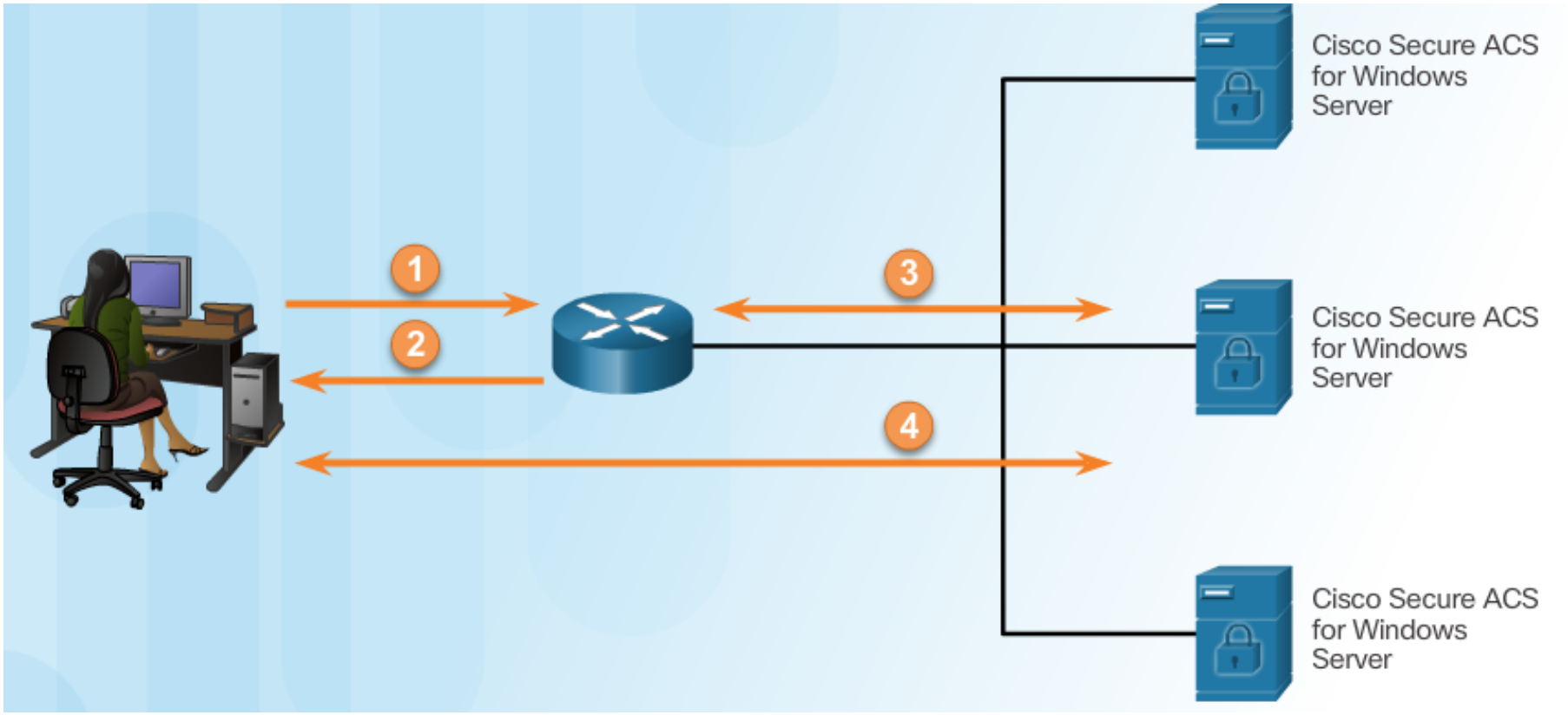
_ldap._tcp.gc.example.com. IN SRV 10 50 389 ds1.example.com.

_ldap._tcp.dc.example.com. IN SRV 10 50 389 ds1.example.com.

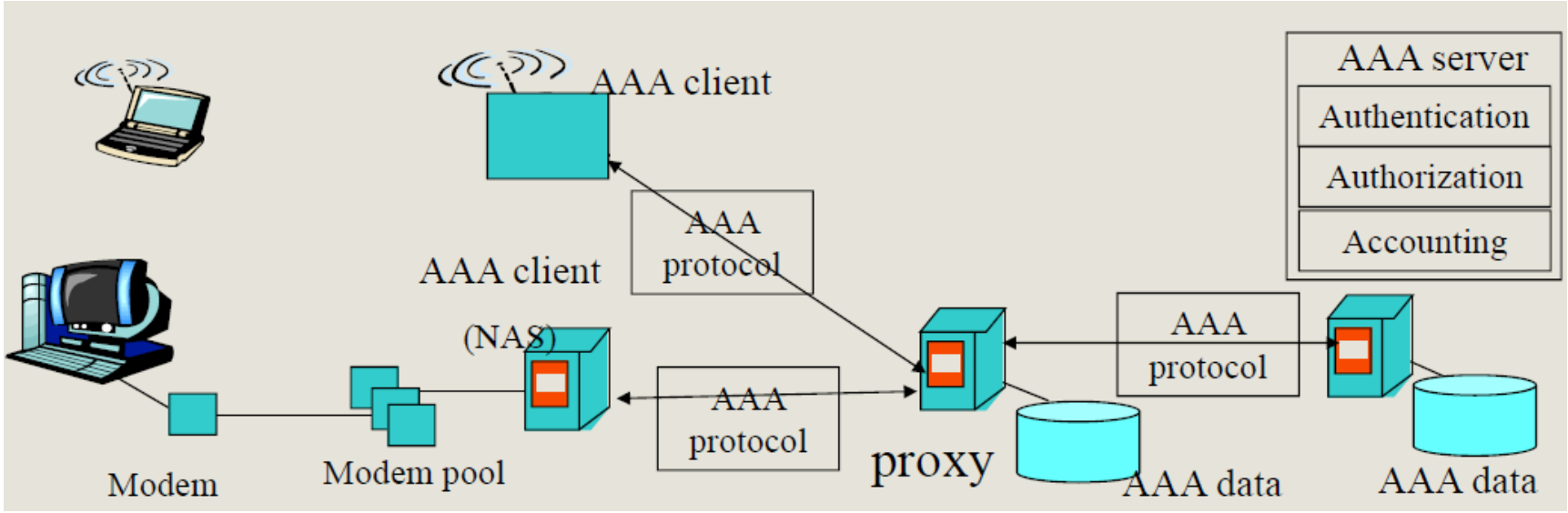
Автентикация с AAA сървър

1. Потребителят се свързва с устройството.
2. Устройството изисква username & password.
3. Устройството ги праща до AAA сървър.
4. AAA сървърът автентицира потребителя.
5. Протоколите за комуникация при автентикацията са TACACS+, RADIUS, DIAMETER

Автентикация с AAA сървър

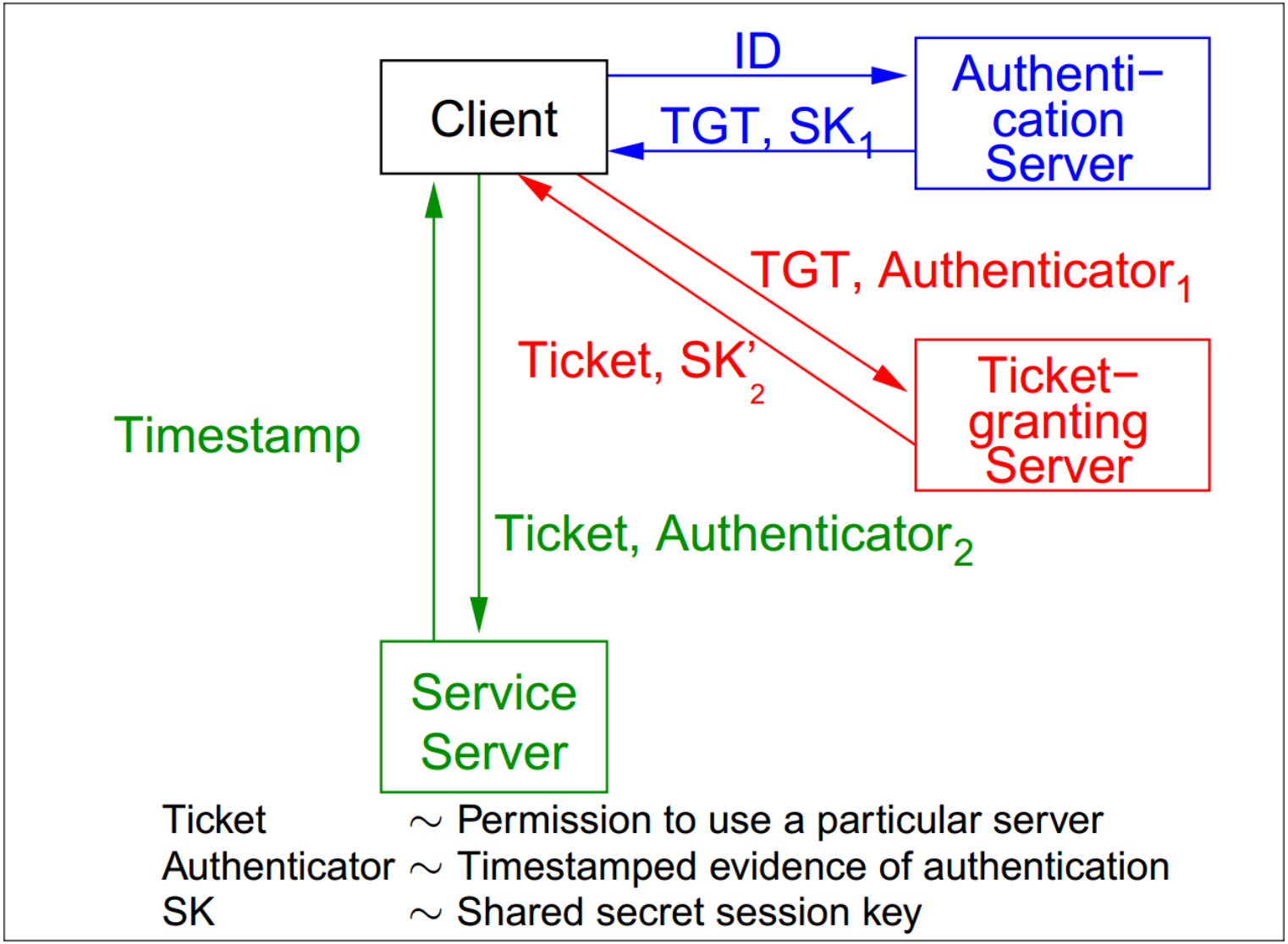


Компоненти на AAA



- AAA клиент - крайно устройство
- AAA протокол - TACACS+, RADIUS, DIAMETER
- AAA proxy сървър
- AAA сървър
- AAA база данни

Автентикация с Kerberos



AAA

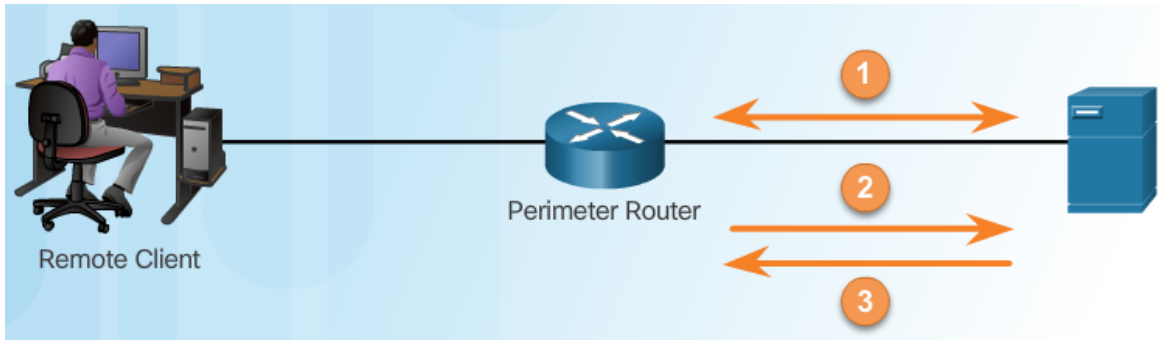
- Автентикация
- Оторизация
- Акаунтинг

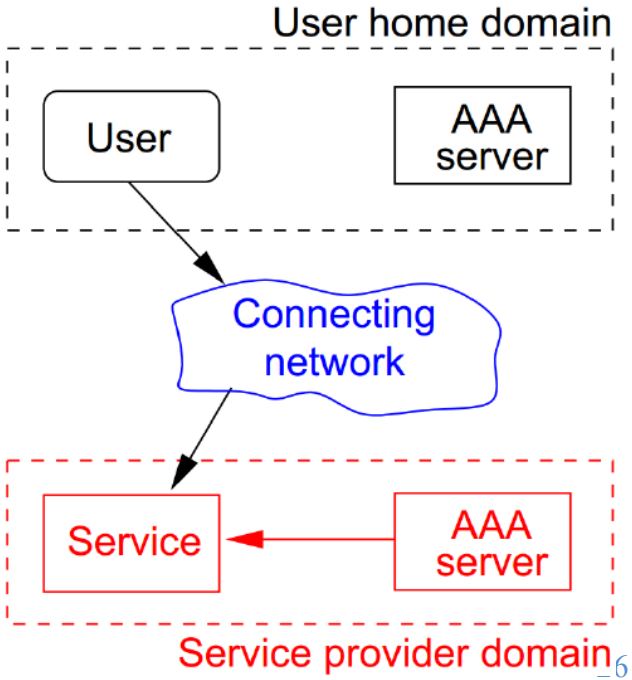
Оторизация

- Определяне на степента, до която е предоставен достъп до конкретен ресурс/услуга на упълномощения субект.
- Резултатът на оторизацията се изисква от системата, когато се достъпва ресурса/ услугата.
- За централизирано оторизиране това става с:
  - Механизми за обработка на контрол на достъпа
  - Ресурсът /услугата трябва да са в състояние да наложат резултата.
- Пример: Kerberos, Active Directory, …

Оторизация

- След автентикацията се преминава към оторизация за заявената услуга от AAA сървъра.
- Той връща отговор за допускане или отхвърляне на оторизацията.





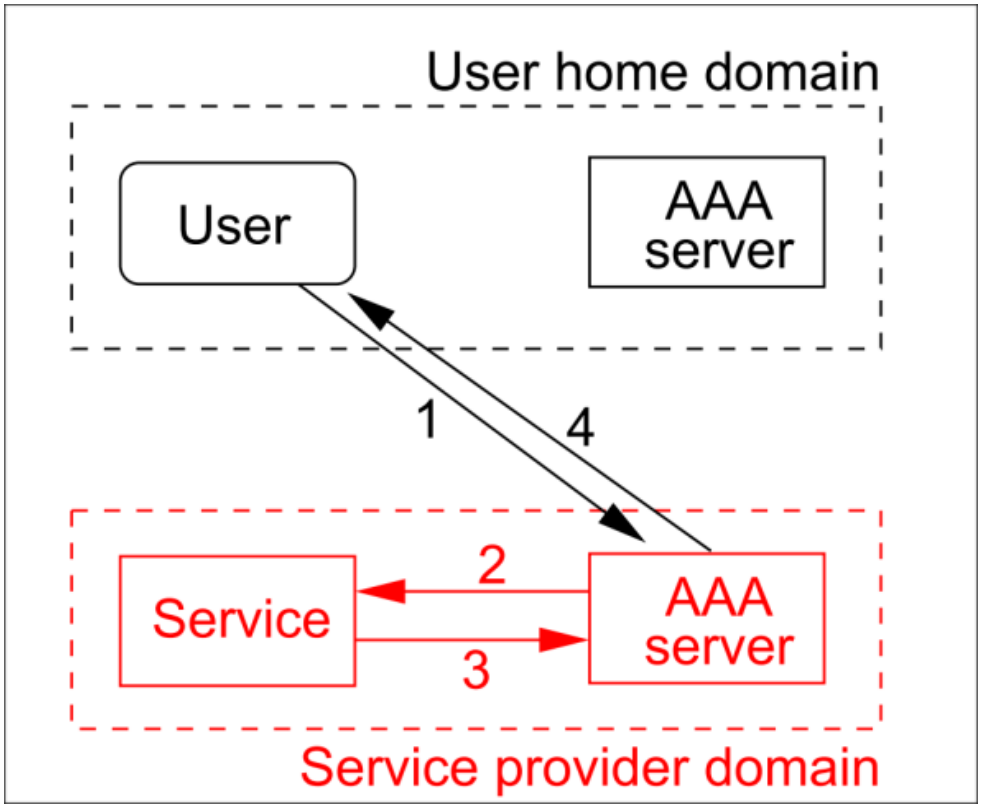
Изисквания към оторизацията

В RFC2906 са дефинирани общи изисквания за оторизация, като:

- AAA протоколът трябва да поддържа изискване за автентикация и оторизация в обща заявка или в отделни.
- Трябва да поддържа препращане към друг AAA server.
- Трябва да позволява на брокерите да добавят собствена информация в съобщенията за заявка/отговор.
- Трябва да позволява сигурност на връзката от край до край когато се използват брокери.
- Брокерите трябва да могат да препратят адреси към заявителя на услугата, който да се използва за достъп до услугата
- Разрешението трябва да се основава на заявител / услуга / среда.
- Трябва да поддържа сигурна комуникация на информацията за разрешението

Начини за оторизация

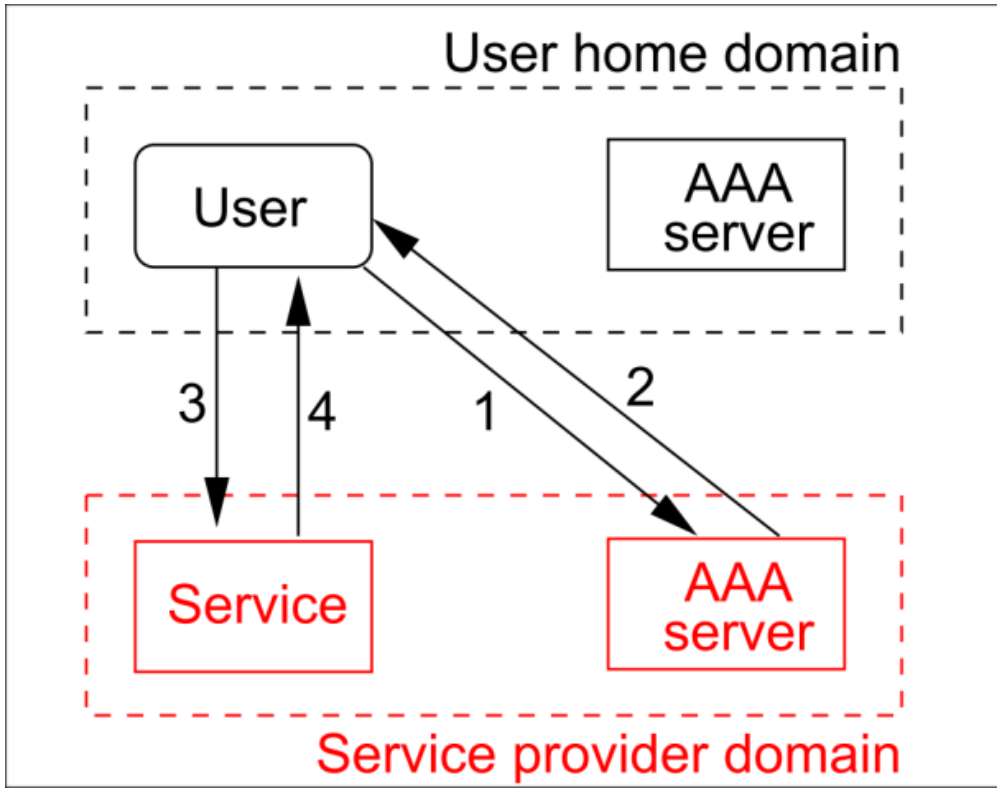
- **Agent:**
  - **Потребителят контактува с AAA сървъра в домейна на ISP.**
  - AAA сървърът работи като агент, пропускайки оторизацията до услугата, която съхранява информация за това “кой какво може”.



-

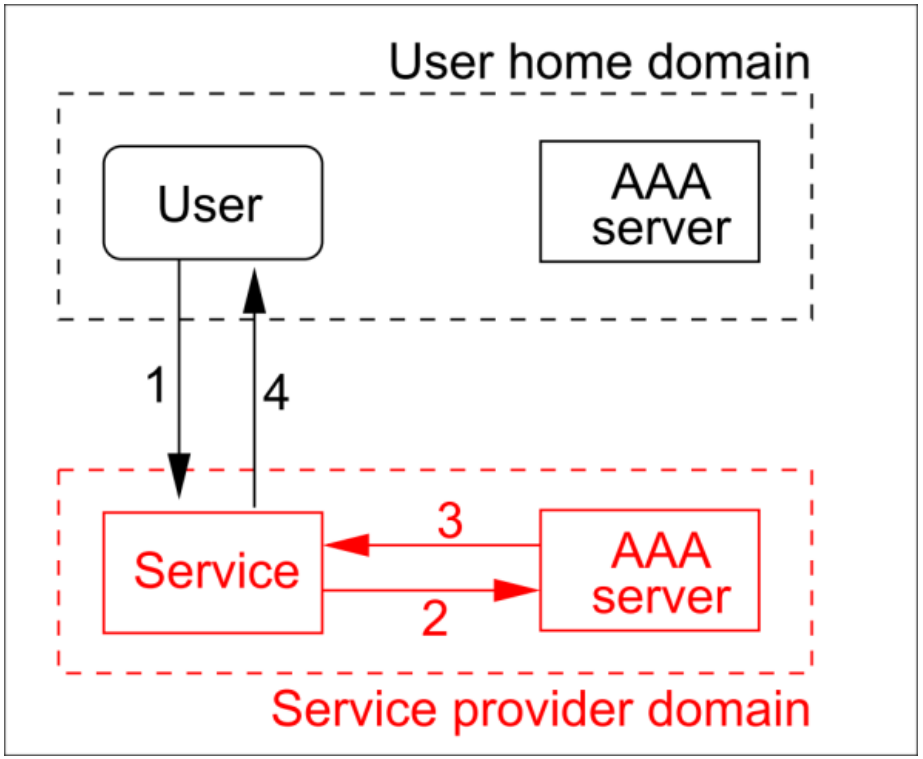
Начини за оторизация

- **Push:**
  - **Потребителят контактува с AAA сървъра в домейна на ISP.**
  - AAA сървърът съхранява информация “кой за коя услуга какви права има” и връща оторизацията на потребителя, който я изпраща към услугата.



Начини за оторизация

- **Pull:**
  - Потребителят контактува с услугата, която взема оторизацията му от AAA сървъра.



AAA

- Автентикация
- Оторизация
- Акаунтинг

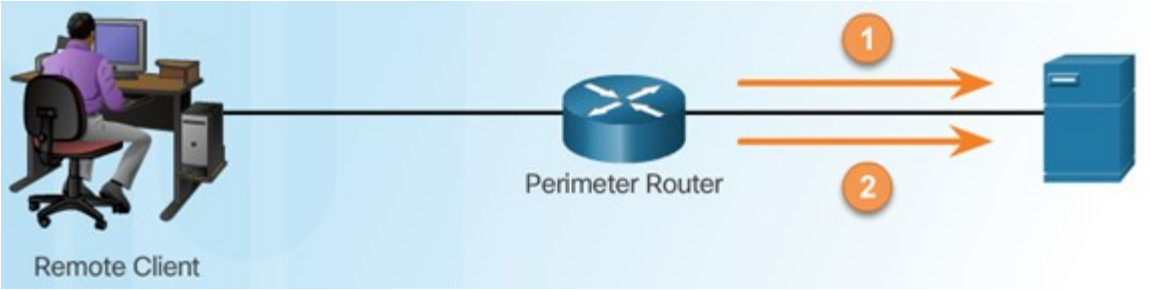
Акаунтинг

- Събира се информация за ползването на продуктите и услугите.
- Използва се за:
  - Разпределяне на разходи
  - Заплащане на услуги
  - Наблюдение на услуги и системи
  - Анализ на трафик, използване на услуги
  - Планиране на промени в мрежата и услугите

Акаунтинг

Информацията, за която се води статистика:

- Мрежа
- Връзки
- Изпълнявани команди
- Системи
- Ресурси



Изисквания към акаунтинга

- Отказоустойчивост
- Потребление на ресурси
- Модел за събиране на данни
  - Предизвикани от събития:
    - Единично събитие
    - Поредица от събития
    - Събрани в устройството резултати от събитията

Техники за събиране на информация

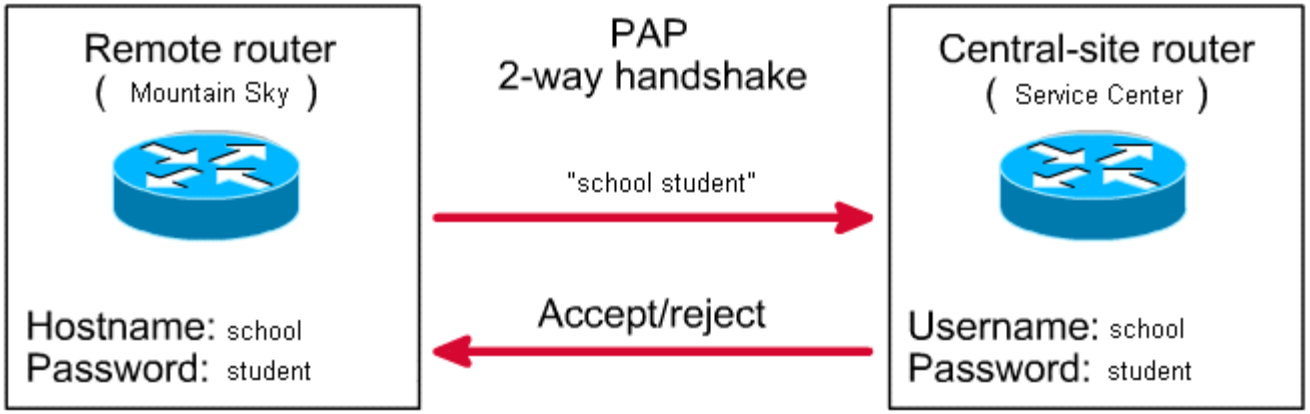
- Polling – непрекъснато запитване
  - Натоварва трафика
  - Използва се, но не се препоръчва
- Управлявано от събития
  - При настъпване на определено събитие се събира информация
  - Може да е за единични събития или пакет от събития
  - Средства – syslog, XpoLog

AAA

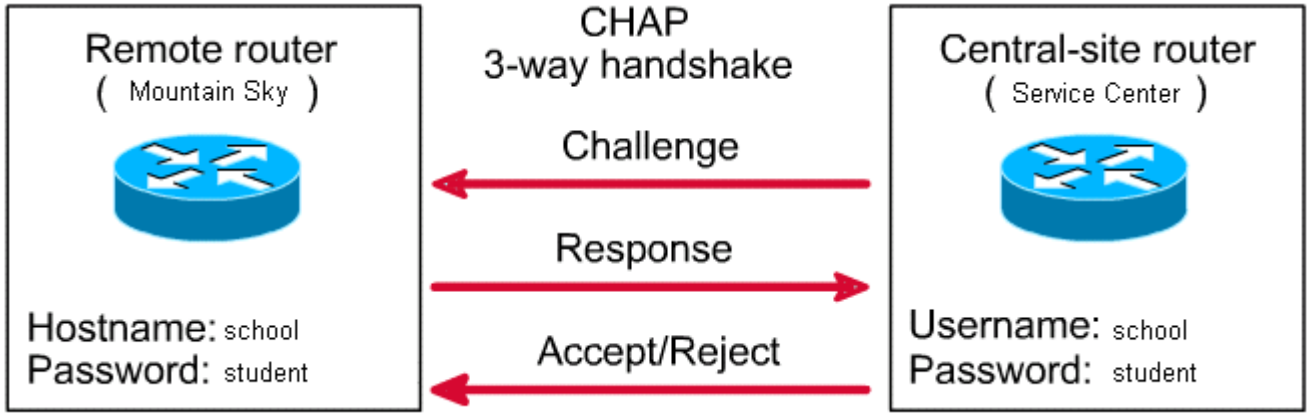
- Автентикация
- Оторизация
- Акаунтинг
- Реализация на AAA

Средства за реализация на AAA

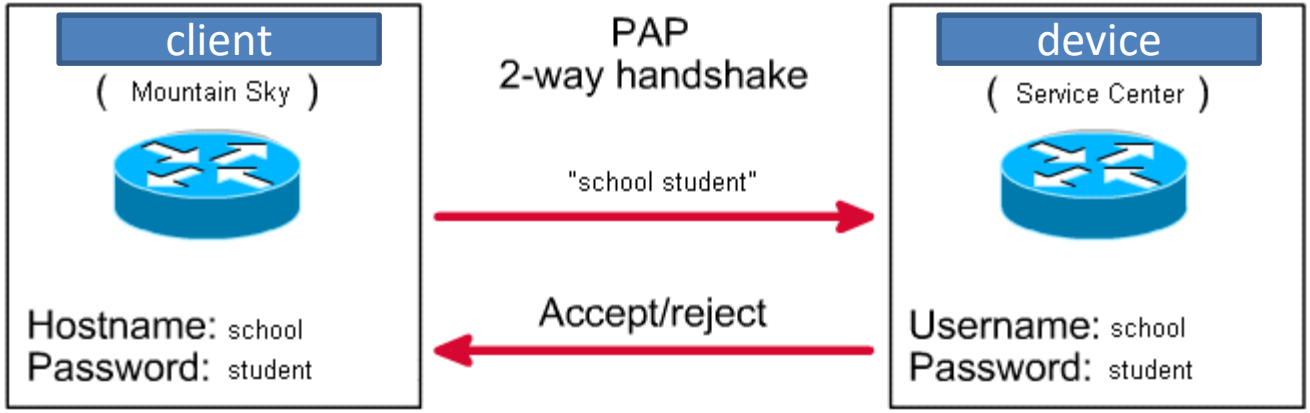
- По отношение на автентикация
  - PAP (Password Authentication Protocol)



  - CHAP (Challenge Authentication Protocol)

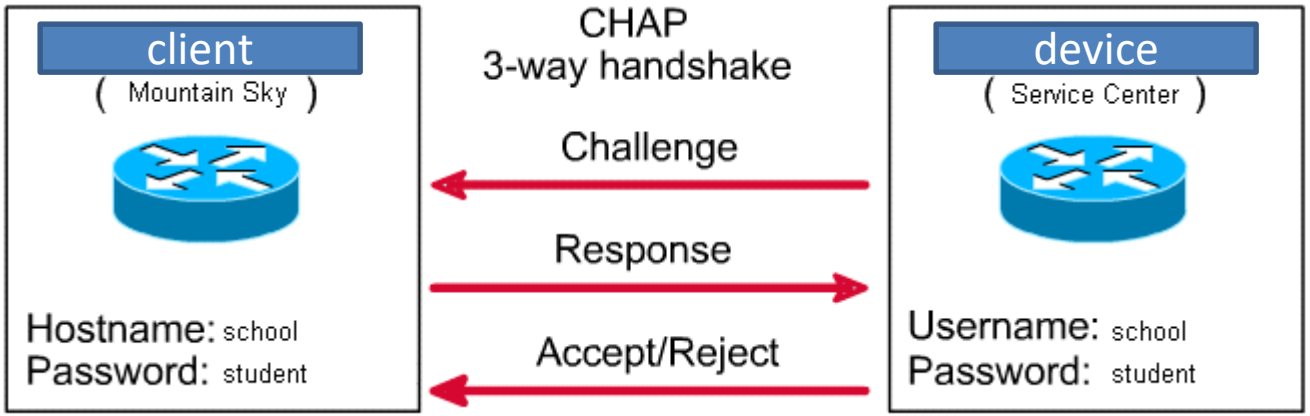


PAP



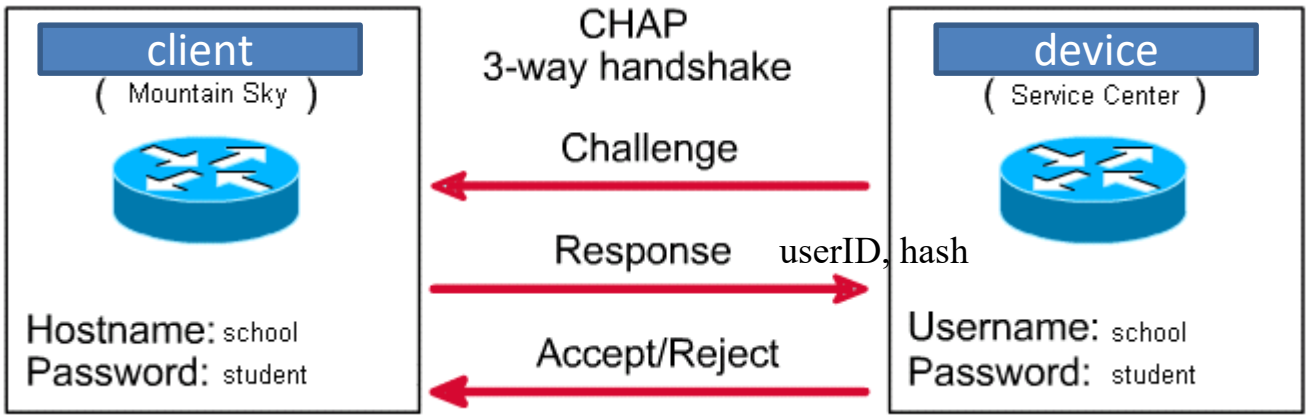
- Креденциалите (username, password) се изпращат непрекъснато, докато автентикацията се потвърди или връзката се прекрати.
- Не е силен, т.к. верификацията е еднократна.
- Паролата се изпраща некриптирана, в прав текст.
- Няма защита от възпроизвеждане или опити за атаки.
- Отдаленият хост контролира броя на опитите за логване, както и времето, в което са допуснати да се правят.

CHAP



- Прави се при началното установяване на връзката и може да се прави по всяко време по-късно.
- Предлага периодична верификация за увеличаване на сигурността (по-ефективен е от PAP)
- Не позволява на извикващия да се опита да се автентикира без покана (challenge).
- Защитава от повторни атаки чрез използване на променливи challenge, които са уникални и непредсказуеми.

CHAP

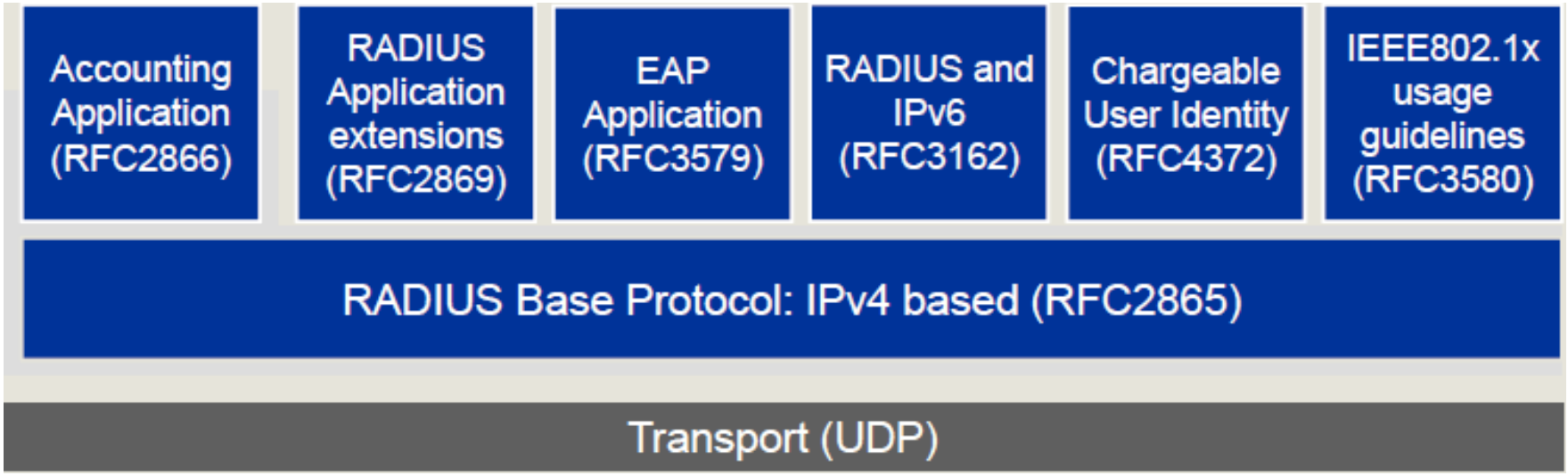


- Устройството изпраща съобщение-покана на клиента (случайно генерирано число и No на сесия за даден потребител).
- Клиентът отговаря с userID (No на сесия за даден потребител) и променлива (Hash), която е получена от неговата парола, криптирана с MD5 с получения ключ (съобщението challenge).
- Устройството я проверява отново и ако съвпада, потвърждава автентикацията. В противен случай връзката се прекъсва.

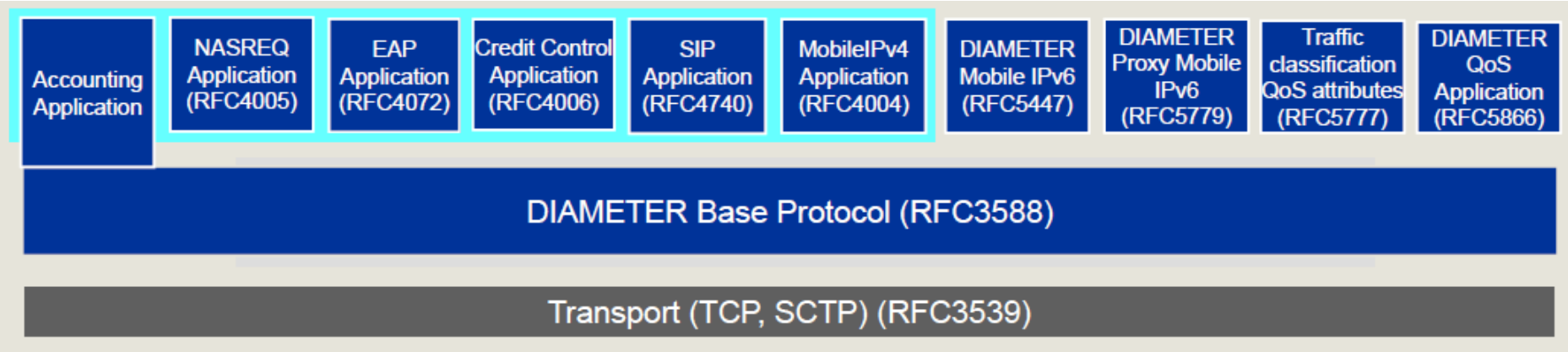
Варианти на CHAP

- MSCHAP (Microsoft CHAP)
  - Изпраща се криптирано и името
- MSCHAP-2
  - Обратно автентициране от страна на сървъра

AAA протоколи – RADIUS (Remote Authentication Dial In User Service protocol)



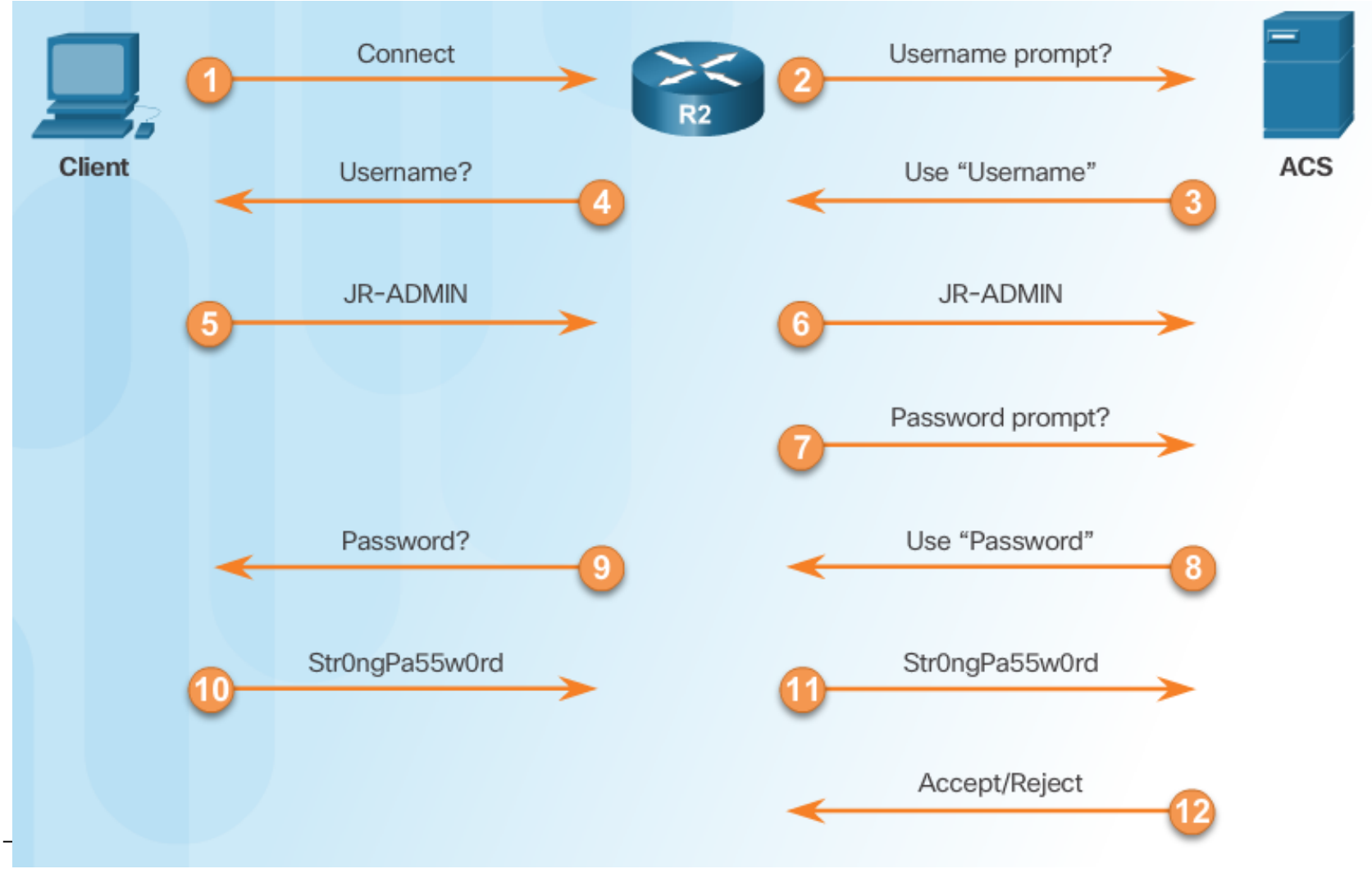
AAA протоколи - DIAMETER



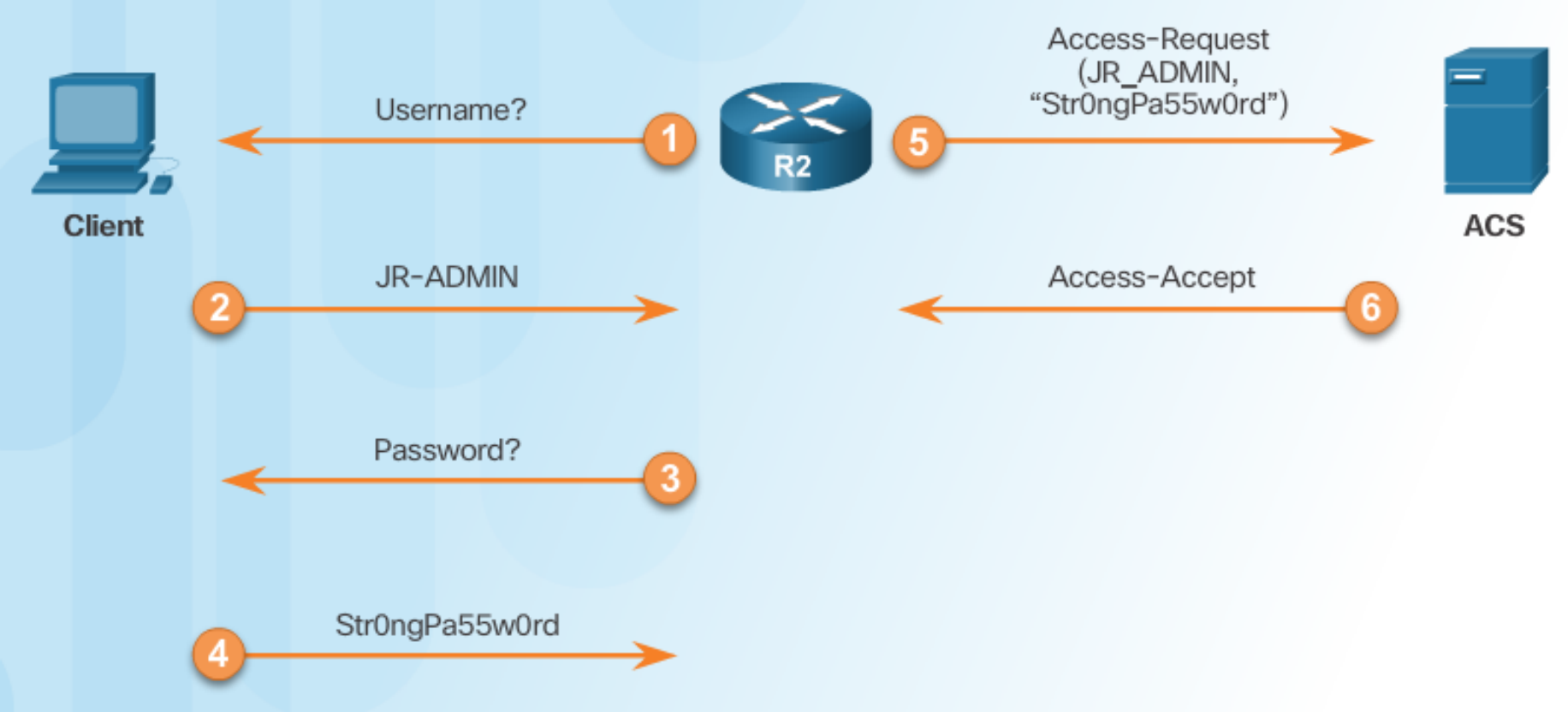
TACACS+ или RADIUS

	TACACS+	RADIUS
Действие	Разделя автентикация от оторизация	Не разделя автентикация от оторизация, но отделя акаунтинга
Стандарт	Най-често от Cisco устройства	Отворен стандарт
Транспортен протокол	TCP	UDP
CHAP	Двупосочна покана/отговор	Еднопосочна покана/отговор от сървъра към клиента
Поддържани протоколи	всички	Без ARA, NetBEUI…
За IP		IPv4(RFC2865) и IPv6(RFC3162)
Конфиденциалност	Пакетът е криптиран	Паролата е криптирана
Персонализиране на настройки	Оторизацията е за потребител или група потребители	Не може да прави оторизация на потребител или група потребители
Акаунтинг	Ограничен	Интензивен

Автентикация с TACACS+



Автентикация с RADIUS

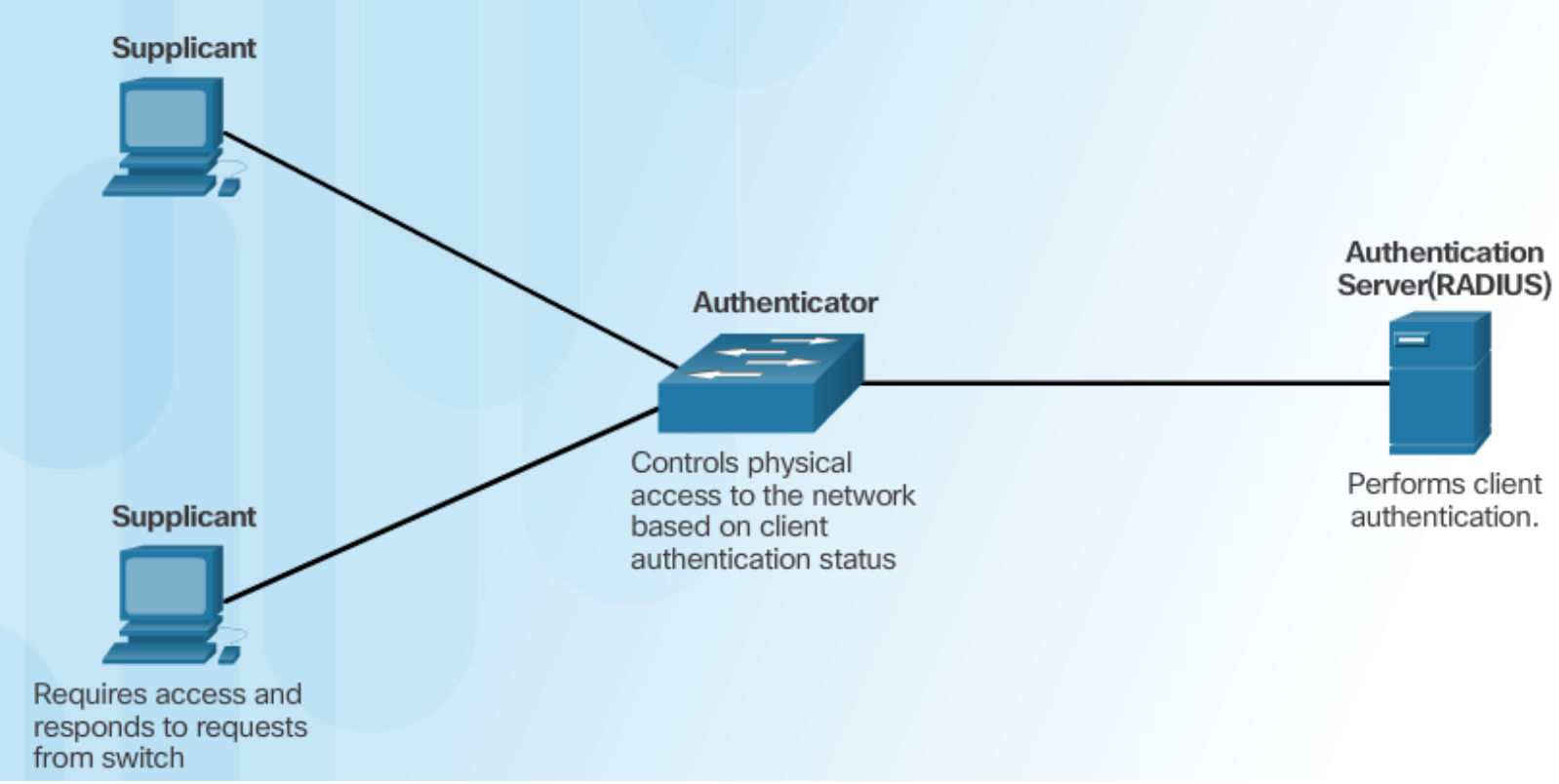


Конфигуриране

1. На устройството се разрешава AAA.
2. Посочва се IP адрес на AAA сървър.
3. Конфигурира се секретният ключ.
4. Избира се протокол за автентикация - RADIUS или TACACS+ според сървъра.

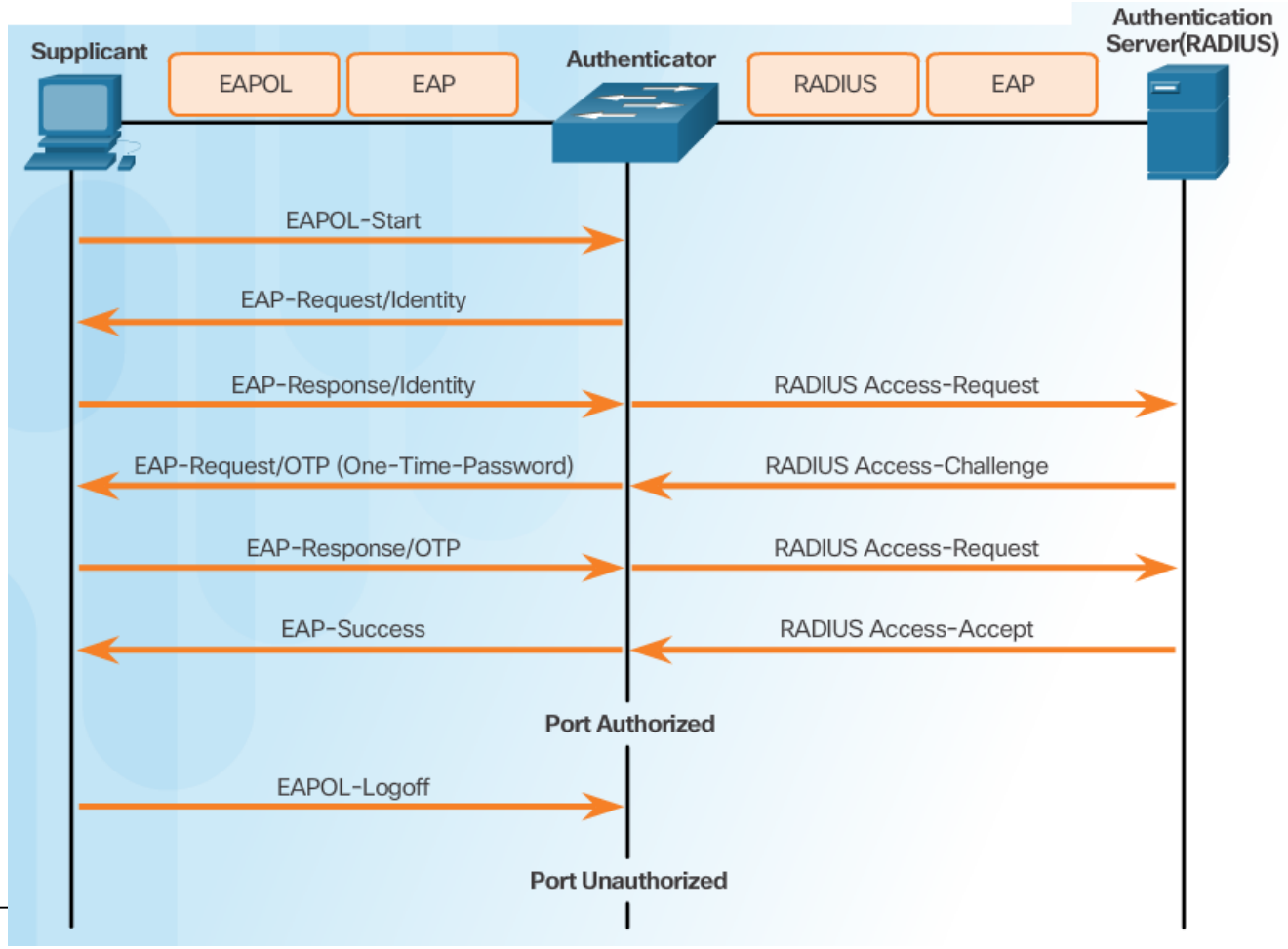
Сигурност с 802.1X порт-базирана автентикация

802.1X роли



Сигурност с 802.1X порт-базирана автентикация

Обмяна на съобщения

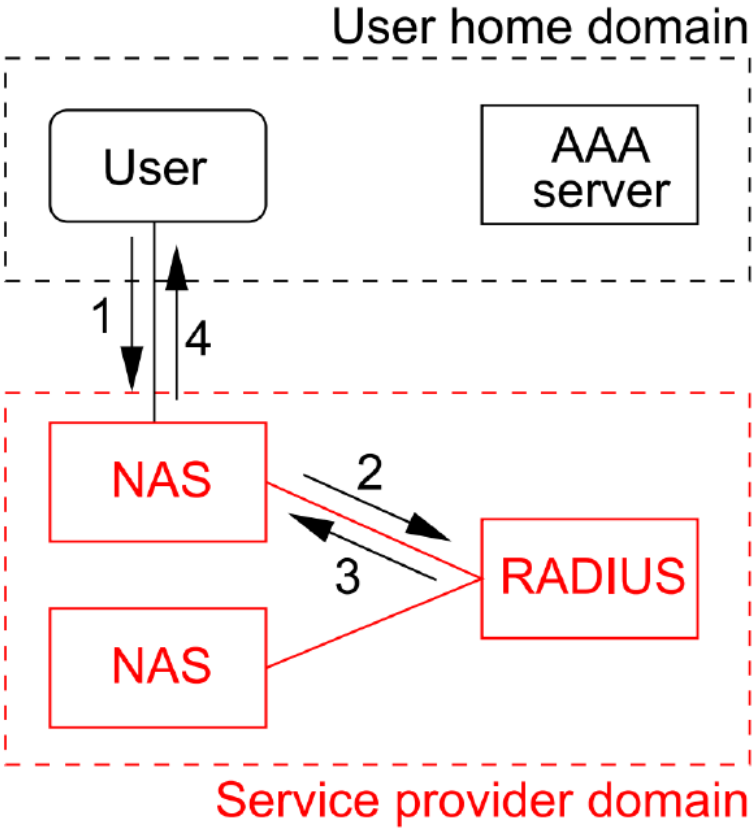


RADIUS

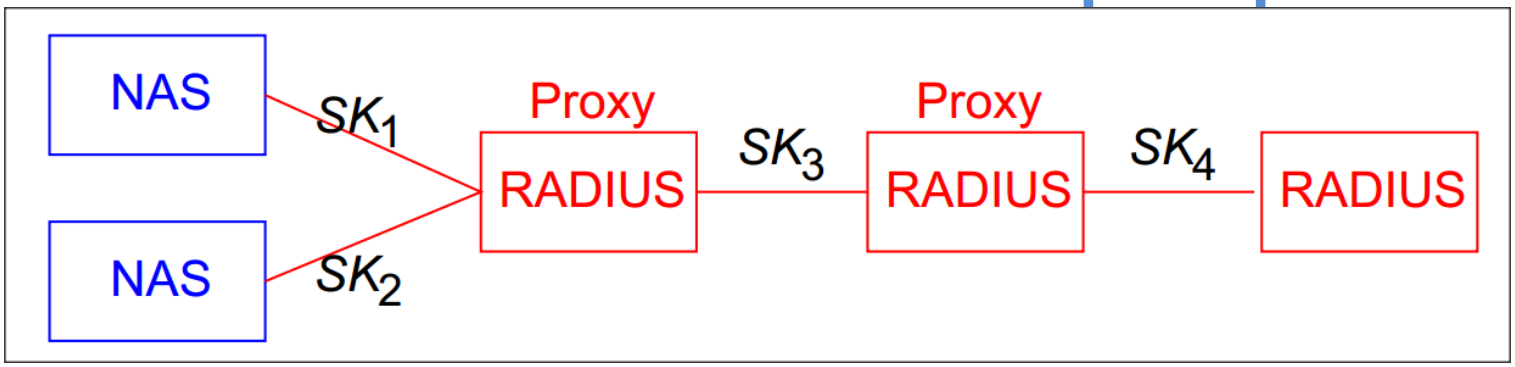
RFC2865: “RADIUS”,

RFC2866: “RADIUS Accounting”

- Използва се за AAA.
- Изисква NAS (Network Access Server), които имат роля на No на сесия за даден потребител RADIUS клиенти.
- Към NAS се свързват потребители, които искат да получат автентикация от RADIUS сървъра.

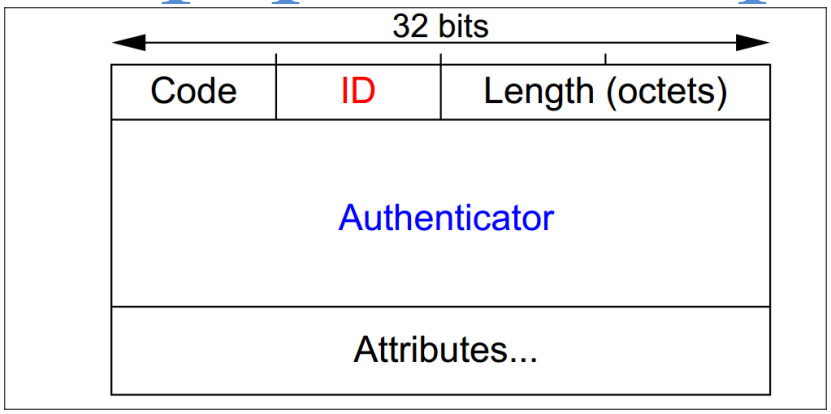


Роля на RADIUS сървъра



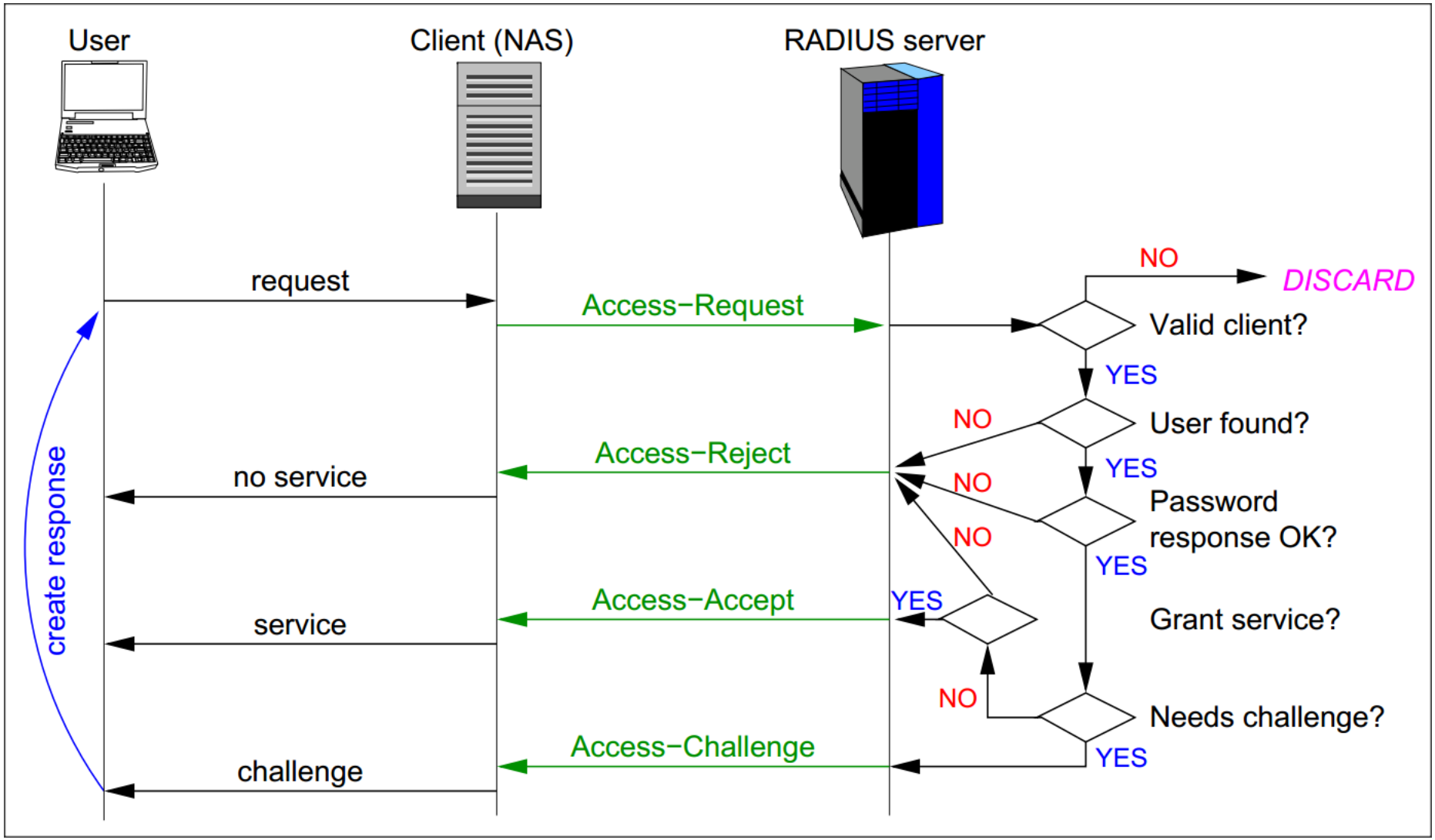
- Съдържа БД с user ID + password и други данни за потребителя
- Сървърът може да работи като proxy, отговаряйки на заявки, насочени към друг сървър
- Може да има повече от един proxy между клиента и сървърът, който съдържа автентикационната информация
- Сигурността се прехвърля от устройство в устройство:
  - Всеки сървър споделя различна парола с всеки клиент (NASs) и останалите сървъри, с които комуникира.

RADIUS формат на протокола

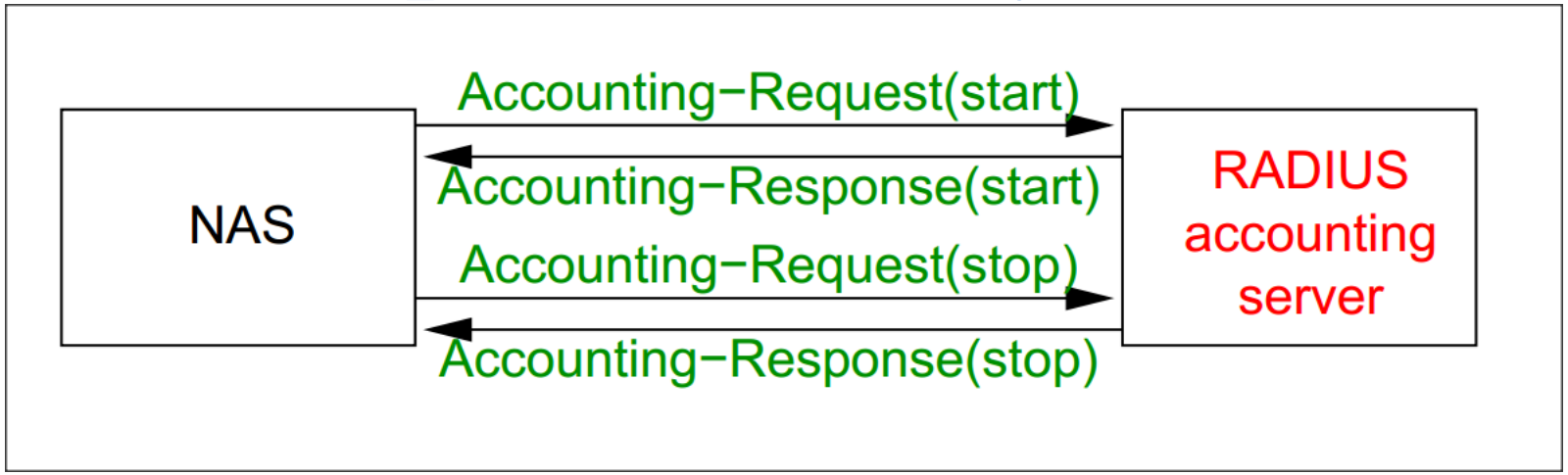


- Stateless протокол, който използва UDP (port 1812, 1645 за автентикация).
- Ако заявката е с times out, клиентът може да препредаде до същия или друг RADIUS сървър.
- ID е номерът на текущата сесия. Сменя се ако source IP, source port или атрибутите се сменят.
- Length е дължината на пакета от 20 до 4096.
- Code е тип на съобщението и е стандартизиран от IANA.
- Authenticator – request/response, криптиран с MD5
- Attributes – един или няколко, определят се от типа на съобщението

Процес на автентикация



Процес на акаунтинг



- Указва се начало и спиране на отчитането при настъпване на определени събития
- Портове 1813, 1646

Ограничения

- Ограничен размер на атрибутни данни (255B)
- Ограничен брой на едновременните връзки (8-bit ID поле)
- Неспособност да се контролира потока към сървърите (UDP)
- Сървърите не откриват типа на проблемите (всички неуспехи са равни)
- Няма съобщение за грешка при отхвърляне на пакети
- Няма възможност за предпазване от Replay атаки
- Няма сигурност от край до край
- Задължително споделена парола (дори ако се използва защитен тунел)

Реализации

- Microsoft RADIUS - портове 1812 и 1813.
- Cisco RADIUS - традиционни портове 1645 и 1646.
- Juniper RADIUS - портове 1645 и 1646.

DIAMETER

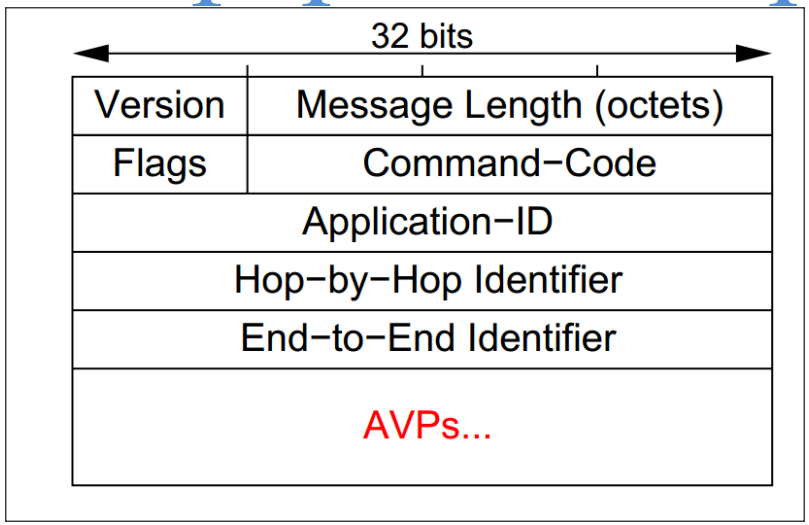
RFC 3588 и RFC 6733

- Connection-oriented протокол, използващ TCP или SCTP (порт 3868)
- При проблем с proxy веригата от Hop-by-Hop се предизвиква препредаване.
- Акаунтингът може да се изпълнява отделно от автентикация/оторизация, като се ползва отделен сървър.
- Използва Peer-to-peer архитектура, като и клиентът и сървърът могат да инициират операцията.
- Сървърът може да инициира прекъсване и повторна автентикация/оторизация.
- Hop-до-Hop сигурност се подобрява с използване на IPsec или TLS.

Diameter устройства

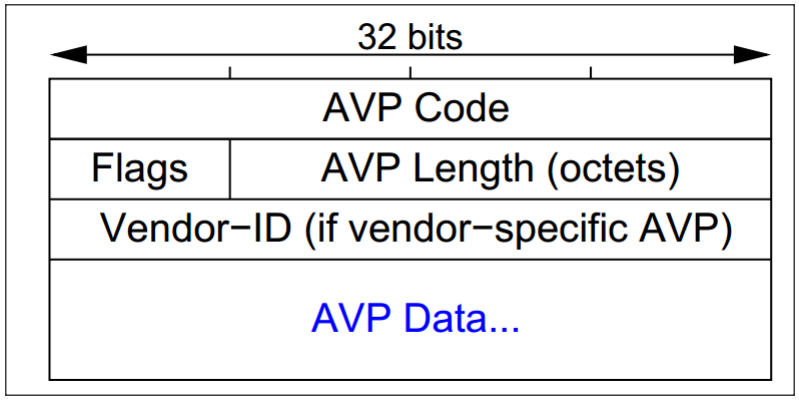
- Клиент: NAS
- Сървър: AAA сървър.
- Relay агент: маршрутизира съобщенията на Diameter прозрачно.
- Proxy агент: маршрутизира съобщенията, но може и да ги променя.
- Redirect агент: Инструктира клиента да опита отново да се автентицира, но да използва друг сървър.
- Translation агент: преобразува съобщения между Diameter и други
- AAA сървъри– като RADIUS.

DIAMETER формат на протокола



- Attribute-Value Pairs (AVPs) пренася информация за атрибутите
- Типове съобщения:
  - Заявки (Requests)
  - Отговори (Answers), вкл. Резултатни кодове от AVP
- И двата типа съобщения съдържат Origin-Host AVP с идентификация за Diameter.

Формат на Diameter AVP



- AVP кодовете са регистрирани в IANA.
  - кодове 1–255 са запазени за съвместимост с RADIUS.
- Флаговете са:
  - M:AVP е задължителен
  - P: Тялото трябва да е криптирано
  - V: специфично за производителя AVP (като Vendor-ID).
- Данните:
  - основни типове (integers, floats, octet strings)
  - композитни типове (“Grouped types”)
  - определени типове (Addresses, Time/date, URIs, . . .)

Въпроси?