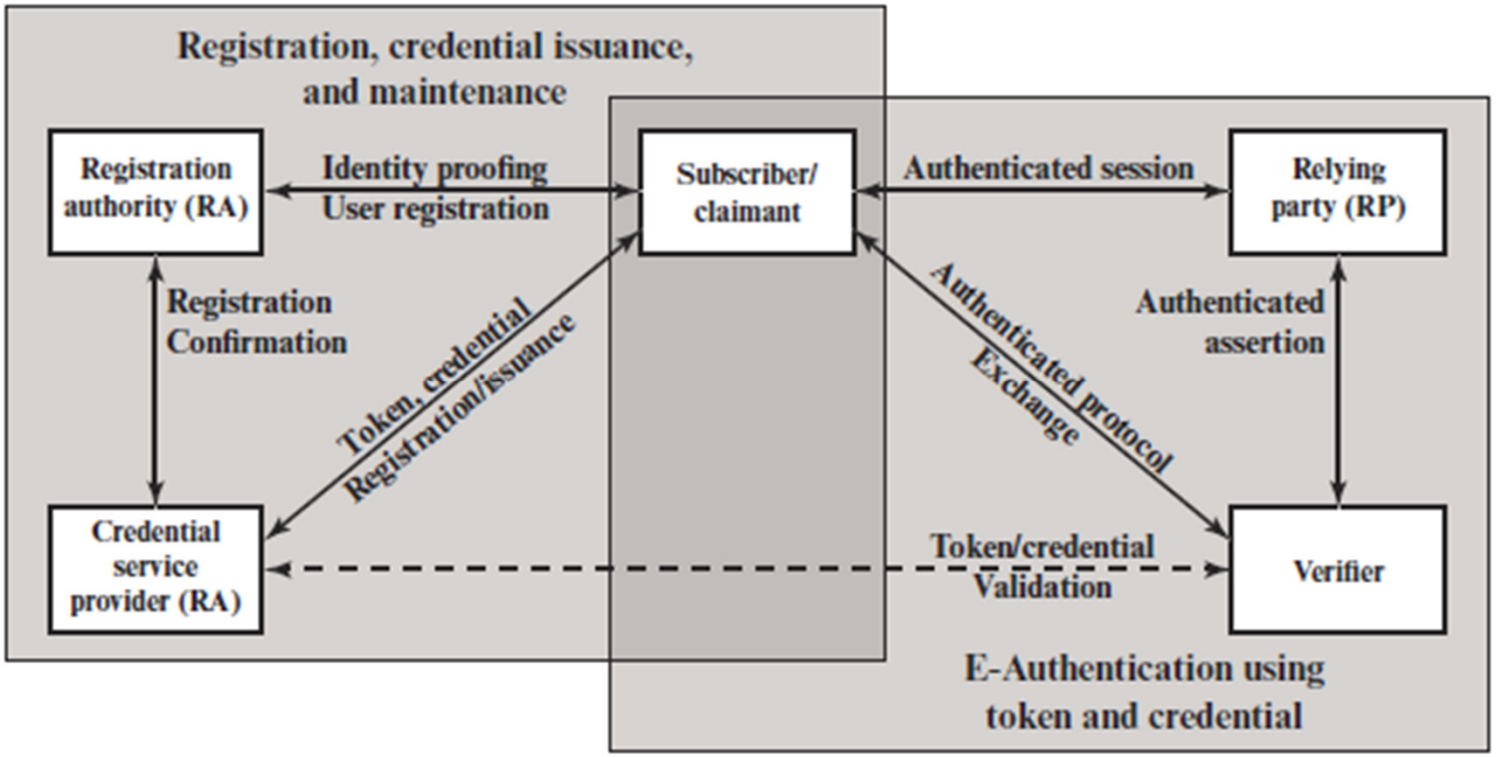
Autentikacija korisnika

Autentikacija korisnika

- Identifikacija
- Verifikacija



Kredencijali

- Kredencijal je struktura podataka koji povezuje identitet nekog korisnika i njegove atribute i koji se šalje verifikatoru radi provere identiteta i prava pristupa
- Sredstva autentikacije
  - Nešto što korisnik zna: Lozinka, PIN,...
  - Nešto što korisnik ima: Ključ, Sertifikat, kartica
  - Biometrijska karakteristika korisnika (otisak prsta, slika lica,...)
  - Dinamički biometrijski podatak (glas, potpis rukom, ritam kucanja,..)

Autentikacioni protokoli

- Koriste se da se uvere učesnici u komunikaciji o identitetima i da se razmene sesijski ključevi
- Mogu biti jednostrani (one-way – npr. za mejl) ili uzajamni (mutual – npr. za enkripciju)
- Ključne teme su
  - Tajnost – da se zaštite ključevi sesije
  - Vremenska komponenta (pravovremenost) – da se spreče replay napadi

Replay napadi

- Gde je validna potpisana poruka kopirana i kasnije ponovno poslata
  - Jednostavan replay – poruka se ponavlja bilo kada
  - Ponavljanje koje se može zabeležiti - vremenski obeležene poruke unutar dozvoljenog vremenskog okvira
  - Ponavljanje koje se ne može detektovati – originalna poruka ne stiže na odredište
  - replay unazad pošiljaocu bez modifikacija
- Protivmere uključuju
  - Upotrebu broja sekvence (generalno nepraktično)
  - Vremenski marker (zahteva sinhronizovane časovnike)
  - prozivanje/odgovor (korišćenjem jedinstvene vrednosti - nonce)

AUTENTIKACIJA KORISNIKA KORIŠĆENJEM SIMETRIČNIH KRIPTOGRAFSKIH ALGORITAMA

Upotreba simetričnog šifrovanja

- Kako je prethodno diskutovano, može koristiti dvo-nivosku hijerarhiju ključeva
- Obično se koristi centar za distribuciju ključeva (KDC)
  - Svaki učesnik deli sopstveni osnovni (master) ključ sa KDC
  - KDC generiše sesijske ključeve koji se koriste za komunikaciju između učesnika
  - master ključevi se koriste za distribuciju sesijskih

Needham-Schroeder Protokol

- Originalan protokol distribucije ključeva od strane third-party
- Za sesiju između A i B uz posredovanje KDC protokol je:

1. A→KDC: $ID_A \| ID_B \| N_1$

2. KDC→A: $E_{Ka}[Ks \| ID_B \| N_1 \| E_{Kb}[Ks \| ID_A]]$

3. A→B: $E_{Kb}[Ks \| ID_A]$

4. B→A: $E_{Ks}[N_2]$

5. A→B: $E_{Ks}[f(N_2)]$

Needham-Schroeder Protokol

- Koristi se za sigurnu distribuciju novih ključeva sesije između A i B
- Osetljiv je na napad ako je stari ključ sesije provaljen jer
  - Poruka 3 se može ponovo poslati da se ubedi B da komunicira sa A
- Promene da se ovo prevaziđe:
  - Vremenski markeri (Denning 81)
  - Upotreba dodatnog nonce-a (Neuman 93)

Denning poboljšanje

- Za sesiju između A i B uz posredovanje KDC protokol je:
  1. A→KDC: $ID_A \,||\, ID_B$
  2. KDC→A: $E_{Ka}[Ks \,||\, ID_B \,||\, T \,||\, E_{Kb}[Ks||ID_A||T]]$
  3. A→B: $E_{Kb}[Ks||ID_A||T]$
  4. B→A: $E_{Ks}[N_1]$
  5. A→B: $E_{Ks}[f(N_1)]$
- Vremenski markeri sprečavaju replay ali zahtevaju sinhronizovane časovnike $|Clock - T| < \Delta t1 + \Delta t2$
- Problem vremenska sinhronizacija

Neumann poboljšanje

- Za sesiju između A i B uz posredovanje KDC protokol je:
  1. $A \rightarrow B: ID_A||N_a$
  2. $B \rightarrow KDC: ID_B||N_b||E(K_b, [ID_A||N_a||T_b])$
  3. $KDC \rightarrow A: E(K_a,[ID_B||N_a||K_s||T_b])||E(K_b,[ID_A||K_s||T_b])||N_b$
  4. $A \rightarrow B: E(K_b, [ID_A||K_s||T_b])||E(K_s, N_b)$

Jednostrana autentikacija

- Zahteva se kada pošiljalac i primalac nisu u komunikaciji u isto vreme (npr. email)
- Zaglavlje treba da bude neizmenjeno da bi se isporučilo od strane email sistema
- Može se zahtevati da sadržaj tela poruke bude zaštićen i pošiljalac autentifikovan

Upotrebom simetričnih algoritama

- Opet upotrebom KDC ali bez finalne razmene nonce:
  1. A→KDC: $ID_A \,||\, ID_B \,||\, N_1$
  2. KDC→A: $E_{Ka}[Ks \,||\, ID_B \,||\, N_1 \,||\, E_{Kb}[Ks||ID_A]\,]$
  3. A→B: $E_{Kb}[Ks||ID_A] \,||\, E_{Ks}[M]$
- ne štiti od replay
  – Može se oslanjati na vremenski marker, ali kašnjenja ga čine problematičnim

Autentikacione aplikacije

- Razmatraju se autentikacione funkcije u klijent server okruženju
- Razvijene da podrže autentikaciju i digitalne potpise na aplikacionom nivou
- Kerberos – autentikacioni servis zasnovan na tajnom ključu
- Radius
- X.509 autentikacioni servis

Kerberos

- Treba rešiti sledeći problem: Kako obezbediti samo autorizovanim korisnicima pristup ka većem broju različitih servisa korišćenjem jedne lozinke
- Pristup sa IP adresama neprihvatljiv
- Server ključeva od poverenja - MIT
- Od Windows 2000 koristi se kao default metod za autentikaciju u Windows okruženju
- Obezbeđuje centralizovanu autentikaciju u distribuiranoj mreži pomoću tajnog ključa
  - Dozvoljava korisnicima pristup servisima distribuiranim na mreži
  - Nema potrebe da se veruje svim radnim stanicama
  - Oslonac na poverenje u centralni autentikacioni server

Kerberos zahtevi

- Prvobitni zahtevi:
  - Sigurnost (od prisluškivanja)
  - Pouzdanost (od otkaza – distribuirana arhitektura)
  - Transparentnost (korisnik je svestan samo unošenja lozinke)
  - Skalabilnost (dodavanje klijenata i servera)
- Koristi autentikacioni protokol zasnovan na Needham-Schroeder

Koncepti - autentikacija

1. $C \rightarrow AS: ID_C||P_C||ID_V$
2. $AS \rightarrow C$: Ticket
3. $C \rightarrow V: ID_C||Ticket$

$Ticket = E(K_v, [ID_C||AD_C||ID_V])$

C = client
AS = authentication server – zna lozinke svih klijenata
V = server
ID_C = identifier of user on C
ID_V = identifier of V
P_C = password of user on C
AD_C = network address of C
K_v = secret encryption key shared by AS and V

Problemi: poruka (1), replay (3), šta ako ima više servisa?

Koncepti - TGS

- Jednom po logon sesiji:
  - (1) C $\rightarrow$ AS: $ID_C||ID_{tgs}$
  - (2) AS $\rightarrow$ C: $E(K_c, Ticket_{tgs})$
- Jednom po tipu servisa:
  - (3) C $\rightarrow$ TGS: $ID_C||ID_V||Ticket_{tgs}$
  - (4) TGS $\rightarrow$ C: $Ticket_v$
- Jednom po sesiji:
  - (5) C $\rightarrow$ V: $ID_C||Ticket_v$

K_c=f(Password)

$Ticket_{tgs} = E(K_{tgs}, [ID_C||AD_C||ID_{tgs}||TS_1||Lifetime_1])$

$Ticket_v = E(K_v, [ID_C||AD_C||ID_v||TS_2||Lifetime_2])$

Problemi: Poruke (3),(5) – replay, vreme trajanja, autentikacija servera

Kerberos 4 osnovno

- bazična third-party autentikacija
- Authentication Server (AS)
  - Korisnici inicijalno pregovaraju sa AS da se identifikuju
  - AS pruža neizmenjivi autentikacioni atestni dokument engl. credential (ticket granting ticket TGT)
- Poseduje Ticket Granting server (TGS)
  - Korisnici naknadno zahtevaju pristup do drugih servisa od TGS na bazi korisnikovog TGT

Kerberos v4

(1) $\mathbf{C} \rightarrow \mathbf{AS}$ $\quad ID_c \| ID_{tgs} \| TS_1$

(2) $\mathbf{AS} \rightarrow \mathbf{C}$ $\quad \mathrm{E}(K_c, [K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}])$

$$Ticket_{tgs} = \mathrm{E}(\mathrm{K}_{tgs}, [\mathrm{K}_{c,tgs} \| \mathrm{ID}_C \| \mathrm{AD}_C \| \mathrm{ID}_{tgs} \| \mathrm{TS}_2 \| \mathrm{Lifetime}_2])$$

(3) $\mathbf{C} \rightarrow \mathbf{TGS}$ $\quad ID_v \| Ticket_{tgs} \| Authenticator_c$

(4) $\mathbf{TGS} \rightarrow \mathbf{C}$ $\quad \mathrm{E}(K_{c,tgs}, [K_{c,v} \| ID_v \| TS_4 \| Ticket_v])$

$$Ticket_{tgs} = \mathrm{E}(\mathrm{K}_{tgs}, [\mathrm{K}_{c,tgs} \| \mathrm{ID}_C \| \mathrm{AD}_C \| \mathrm{ID}_{tgs} \| \mathrm{TS}_2 \| \mathrm{Lifetime}_2])$$

$$Ticket_v = \mathrm{E}(\mathrm{K}_v, [\mathrm{K}_{c,v} \| \mathrm{ID}_C \| \mathrm{AD}_C \| \mathrm{ID}_v \| \mathrm{TS}_4 \| \mathrm{Lifetime}_4])$$

$$Authenticator_c = \mathrm{E}(\mathrm{K}_{c,tgs}, [\mathrm{ID}_C \| \mathrm{AD}_C \| \mathrm{TS}_3])$$

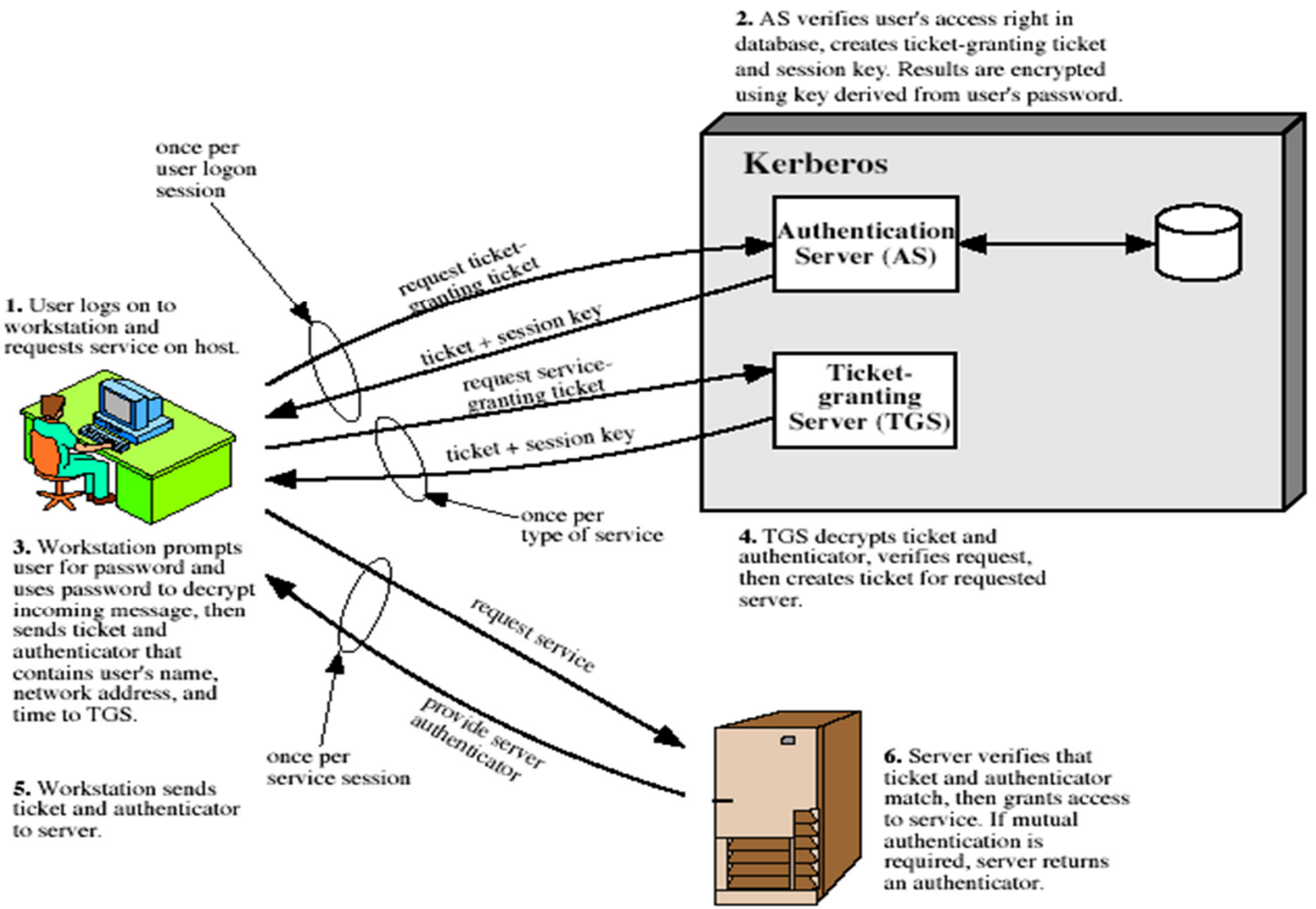
(5) $\mathbf{C} \rightarrow \mathbf{V}$ $\quad Ticket_v \| Authenticator_c$

(6) $\mathbf{V} \rightarrow \mathbf{C}$ $\quad \mathrm{E}(K_{c,v}, [TS_5 + 1])$ (for mutual authentication)

$$Ticket_v = \mathrm{E}(\mathrm{K}_v, [\mathrm{K}_{c,v} \| \mathrm{ID}_C \| \mathrm{AD}_C \| \mathrm{ID}_v \| \mathrm{TS}_4 \| \mathrm{Lifetime}_4])$$

$$Authenticator_c = \mathrm{E}(\mathrm{K}_{c,v}, [\mathrm{ID}_C \| \mathrm{AD}_C \| \mathrm{TS}_5])$$

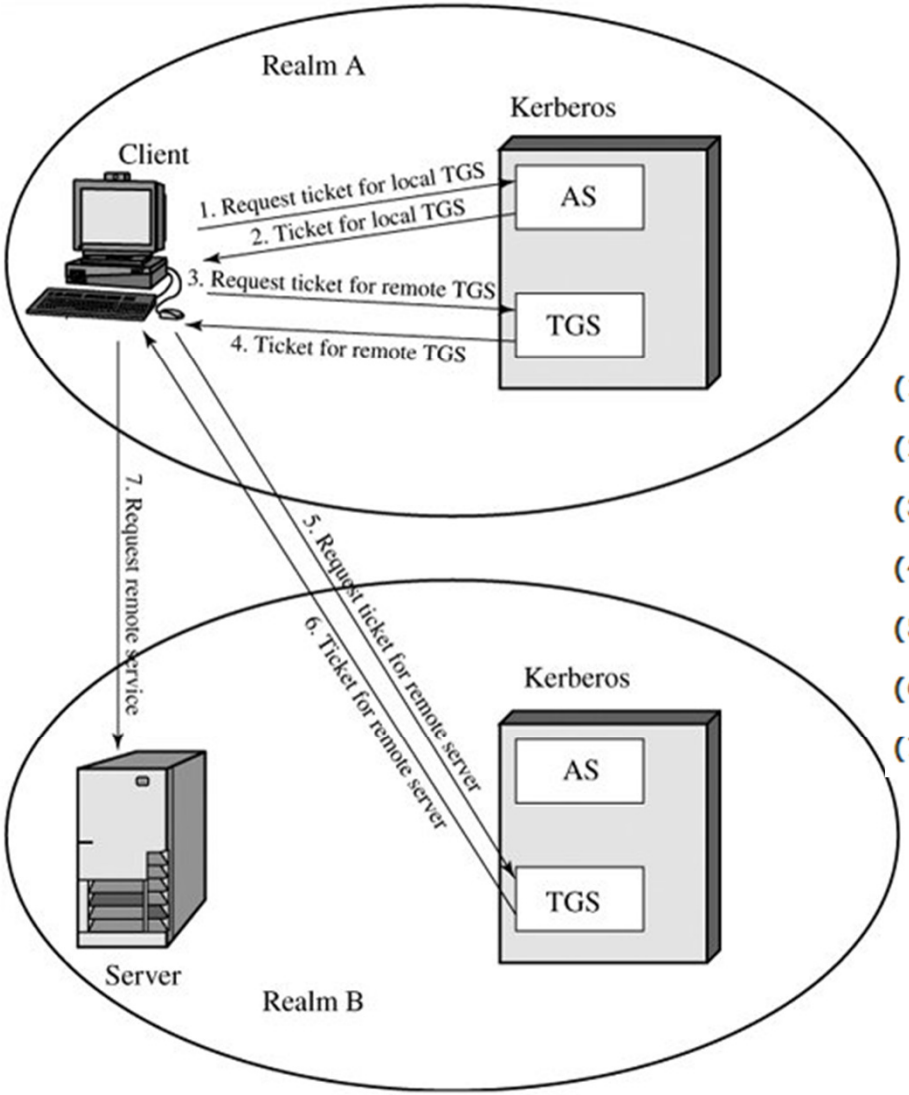
Kerberos 4 pregled



Kerberos okruženje

- Kerberos realm se sastoji od:
  - Kerberos servera
  - Više klijenata, svi registrovani kod servera
  - Applikacionih servera, koji dele ključeve sa Kerberos serverom
- Tipično jedan administrativni domen
- Može i više Kerberos servera koji dele ključeve i poverenje

Komunikacija između realm-ova



- Kerberos serveri u različitim realm-ovima dele međusobno tajne ključeve

(1) C ⟶ AS: $ID_c||ID_{tgs}||TS_1$

(2) AS ⟶ C: $E(K_c, [K_{c,tgs}||ID_{tgs}||TS_2||Lifetime_2||Ticket_{tgs}])$

(3) C ⟶ TGS: $ID_{tgsrem}||Ticket_{tgs}||Authenticator_c$

(4) TGS ⟶ C: $E(K_{c,tgs}, [K_{c,tgsrem}||ID_{tgsrem}||TS_4||Ticket_{tgsrem}])$

(5) C ⟶ TGS_{rem}: $ID_{vrem}||Ticket_{tgsrem}||Authenticator_c$

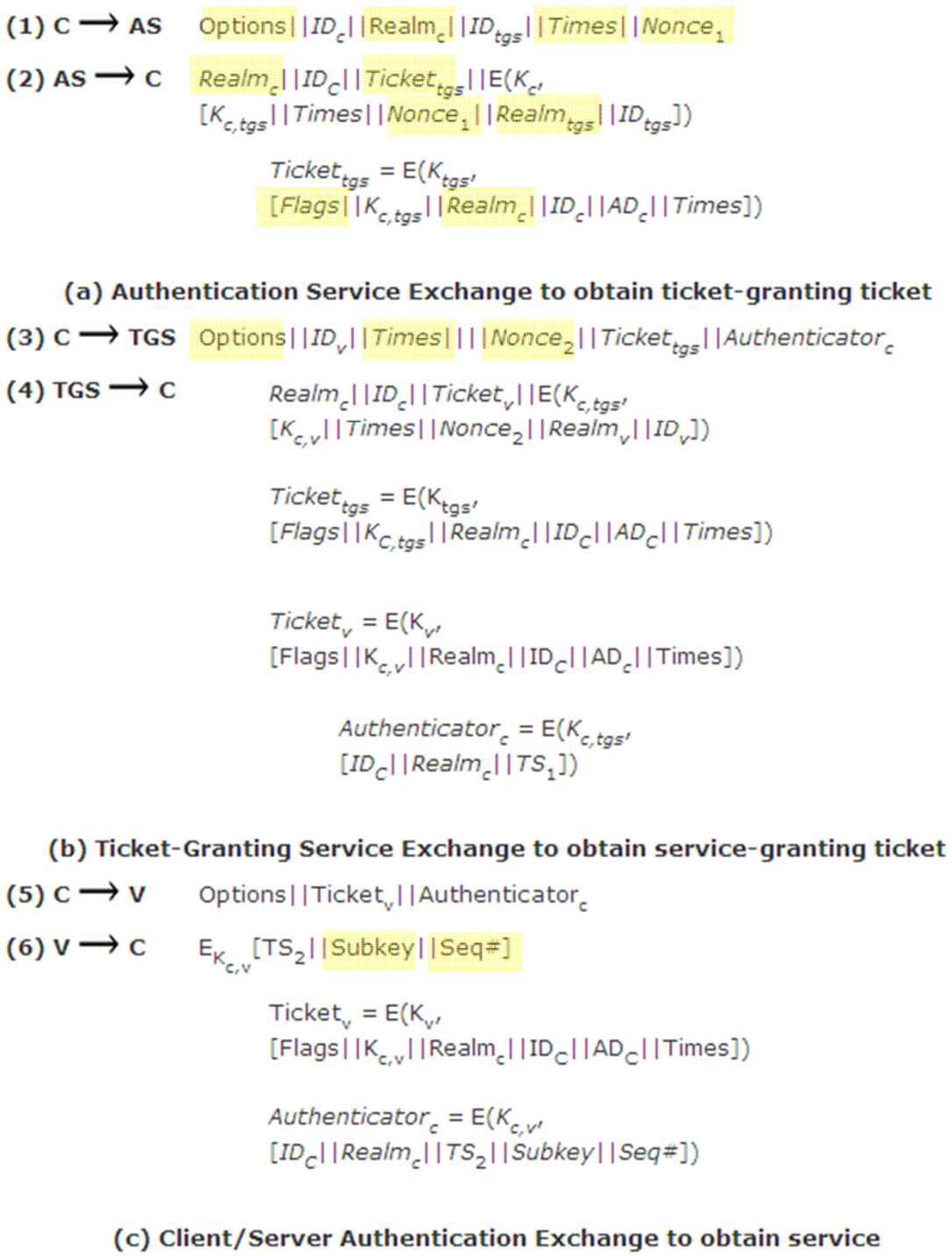
(6) TGS_{rem} ⟶ C: $E(K_{c,tgsrem}, [K_{c,vrem}||ID_{vrem}||TS_6||Ticket_{vrem}])$

(7) C ⟶ V_{rem}: $Ticket_{vrem}||Authenticator_c$

Kerberos 5

- Nije vezan za DES (PCBC)
- Radi i sa ISO adresama
- Poruke kodovane u ASN.1 i BER
- Drugačija definicija lifetime-a
- Authentication forwarding (štampanje fajla koji je na serveru)
- Olakšana interrealm komunikacija (skalabilnija)
- Dvostruko kriptovanje poruka (2) i (4)

- **Realm**: Indicates realm of user
- **Options**:Used to request that certain flags be set in the returned ticket
- **Times**: Used by the client to request the following time settings in the ticket:
  - from: the desired start time
  - till: the requested expiration
  - rtime: requested renew-till time
- **Nonce**: A random value to be repeated in message (2) to assure that the response is fresh and has not been replayed by an opponent
- **Subkey**: The client's choice for an encryption key to be used to protect this specific application session. If this field is omitted, the session key from the ticket ($K_{c,v}$) is used.
- **Sequence number**: An optional field that specifies the starting sequence number to be used by the server for messages sent to the client during this session.



AUTENTIKACIJA KORISNIKA KORIŠĆENJEM ASIMETRIČNIH KRIPTOGRAFSKIH ALGORITAMA

Upotreba javnih ključeva

- Čitav niz pristupa zasnovanih na korišćenju šifrovanja upotrebom javnih ključeva
- Neophodno je da se osigura upotreba korektnih javnih ključeva za ostale učesnike
- Koristi se centralni Autentikacioni server (AS) koji nema uvid u ključeve
- Razni postojeći protokoli koriste vremenske markere ili nonce

Denning AS Protokol

- Pretpostavlja se da učesnici u komunikaciji nemaju razmenjene javne ključeve
- Denning je predložio sledeći protokol:
  1. A→AS: $ID_A \,||\, ID_B$
  2. AS→A: $E(PR_{as}, [ID_A||PU_a||T])||E(PR_{as}, [ID_B||PU_b||T])$
  3. A→B: $E(PR_{as}, [ID_A||PU_a||T])||E(PR_{as}, [ID_B||PU_b||T])$ $||E(PU_b, E(PR_a, [K_s||T]))$

Woo/Lam Protocol

- Nema timestamp, već samo nonce:
  1. $A \rightarrow KDC: ID_A||ID_B$
  2. $KDC \rightarrow A: E(PR_{auth}, [ID_B||PU_b])$
  3. $A \rightarrow B: E(PU_b, [N_a||ID_A])$
  4. $B \rightarrow KDC : ID_A||ID_B||E(PU_{auth}, N_a)$
  5. $KDC \rightarrow B: E(PR_{auth}, [ID_A||PU_a])||E(PU_b, E(PR_{auth}, [N_a||K_s||ID_A||ID_B]))$
  6. $B \rightarrow A: E(PU_a, E(PR_{auth}, [(N_a||K_s||ID_A||ID_B)||N_b]))$
  7. $A \rightarrow B: E(K_s, N_b)$

Jednostrana autentikacija - Pristupi sa javnim ključevima

- Ako je tajnost osnovni cilj:

  $A \rightarrow B: E_{PUb}[Ks] \,||\, E_{Ks}[M]$

  – Šifrovan ključ sesije i šifrovana poruka
- Ako je potrebna autentikacija, potreban je digitalni potpis sa digitalnim sertifikatom:

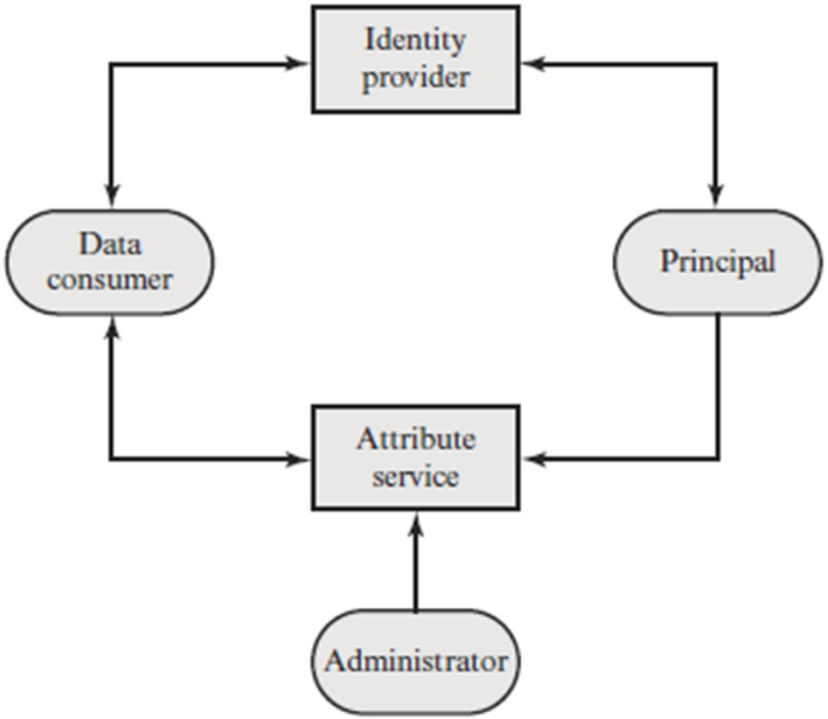
  $A \rightarrow B: M \,||\, E_{PRa}[H(M)] \,||\, E_{PRas}[T||ID_A||PU_a]$

  – Uključuje poruku, potpis i sertifikat

FEDERATIVNO UPRAVLJANJE IDENTITETIMA

Arhitektura

- Provera identiteta i pristup servisima u više domena
- Izbegavanje višestrukih identiteta
- Single sign-on
- Principal - korisnik
- Identity provider - IdP
- Data consumer - Service/resource provider - SP
- Attribute provider



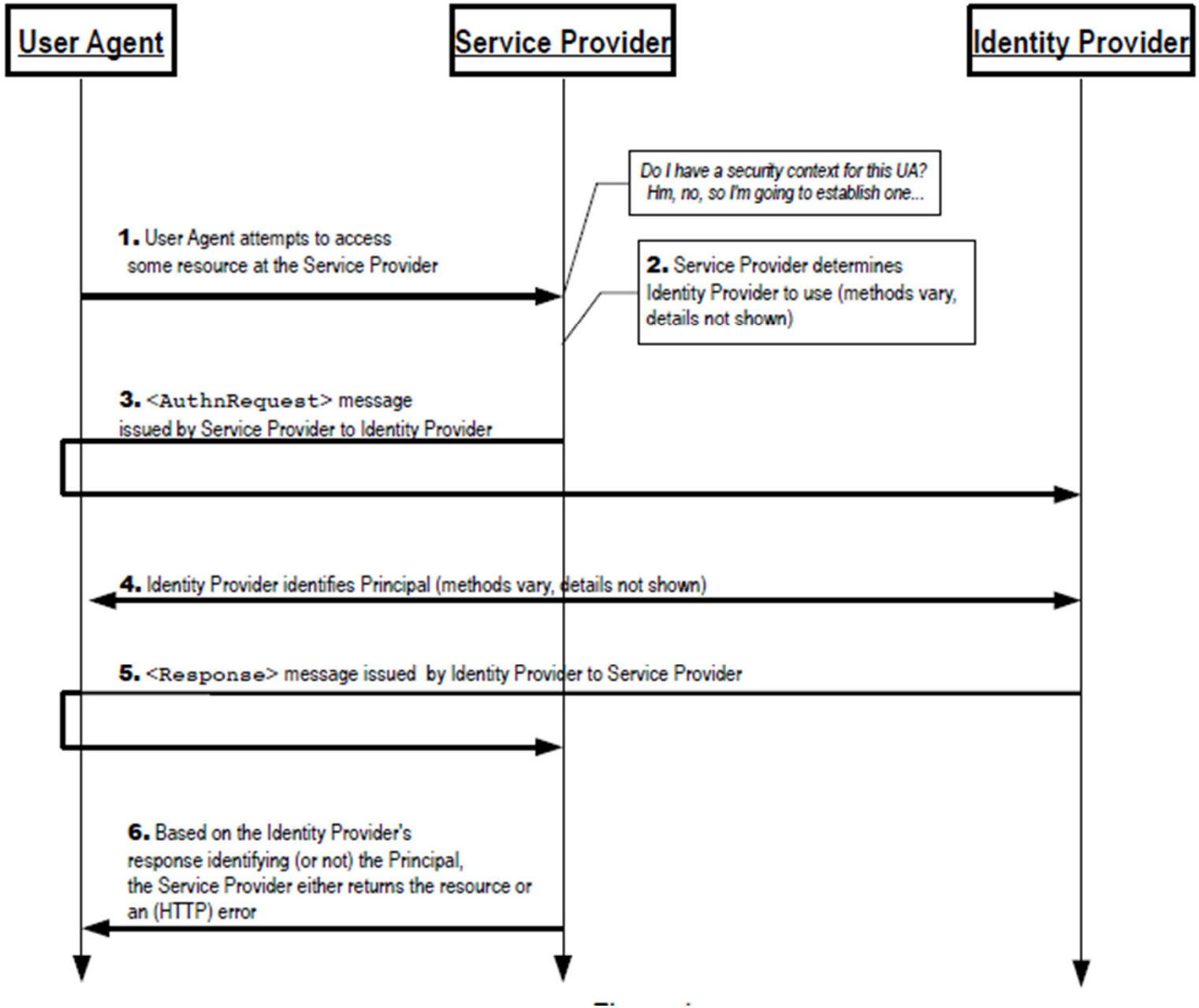
Federacije identiteta

- Davanje pristupa korisnicima iz jednog domena servisu iz drugog domena
- Skup svih provajdera identiteta i provajdera resursa čini federaciju identiteta
- Tehnologije:
  - simplesamlphp
  - Shiboleth (isto zasnovano na SAML)
  - OAuth

Shiboleth mehanizam rada

1. Korisnik browserom pristupa servisu (SP)
2. Servis zahteva autentikaciju koji se redirektuje na IdP
3. Korisnik šalje svoje kredencijale IdP (ako ne postoji već aktivna sesija)
4. Ako je sve u redu odgovor se šalje korisniku i prosleđuje SP
5. SP proverava odgovor
6. Ako je sve u redu korisnik pristupa resursima

https://www.shibboleth.net/index/basic/



https://www.oasis-open.org/committees/download.php/56782/sstc-saml-profiles-errata-2.0-wd-07.pdf

Shiboleth detalji

- Kako SP zna kom IdP da pošalje zahtev?
  - Bira korisnik
  - Na osnovu identifikatora/username-a (@IdP)
- Filtriranje atributa zbog očuvanja privatnosti
- Metapodaci kojim SP i IdP znaju kako da međusobno komuniciraju
  - Imena entiteta
  - URL
  - Kriptografski podaci

https://www.shibboleth.net/index/intermediate/

Komunikacija između IdP i SP

- SAML – protokol zasnovan na XML, SAML 2.0
- SAML authority – IdP
- SAML consumer – SP
- SAML assertion – skup podataka koji se šalje između entiteta:
  - Autentikacioni
  - Atributi
  - Odluka o autorizaciji

Select your identity provider

English | Bokmål | Nynorsk | Sámegiella | Dansk | Deutsch | Español | Svenska | Suomeksi | Français | Italiano | Nederlands | Lëtzebuergesch | Čeština | Slovenščina | Hrvatski | Magyar | Język polski | Português | Português brasileiro | Türkçe | 日本語 | 繁體中文 | ελληνικά | Lietuvių kalba | Åarjelh-saemien giele | русский язык

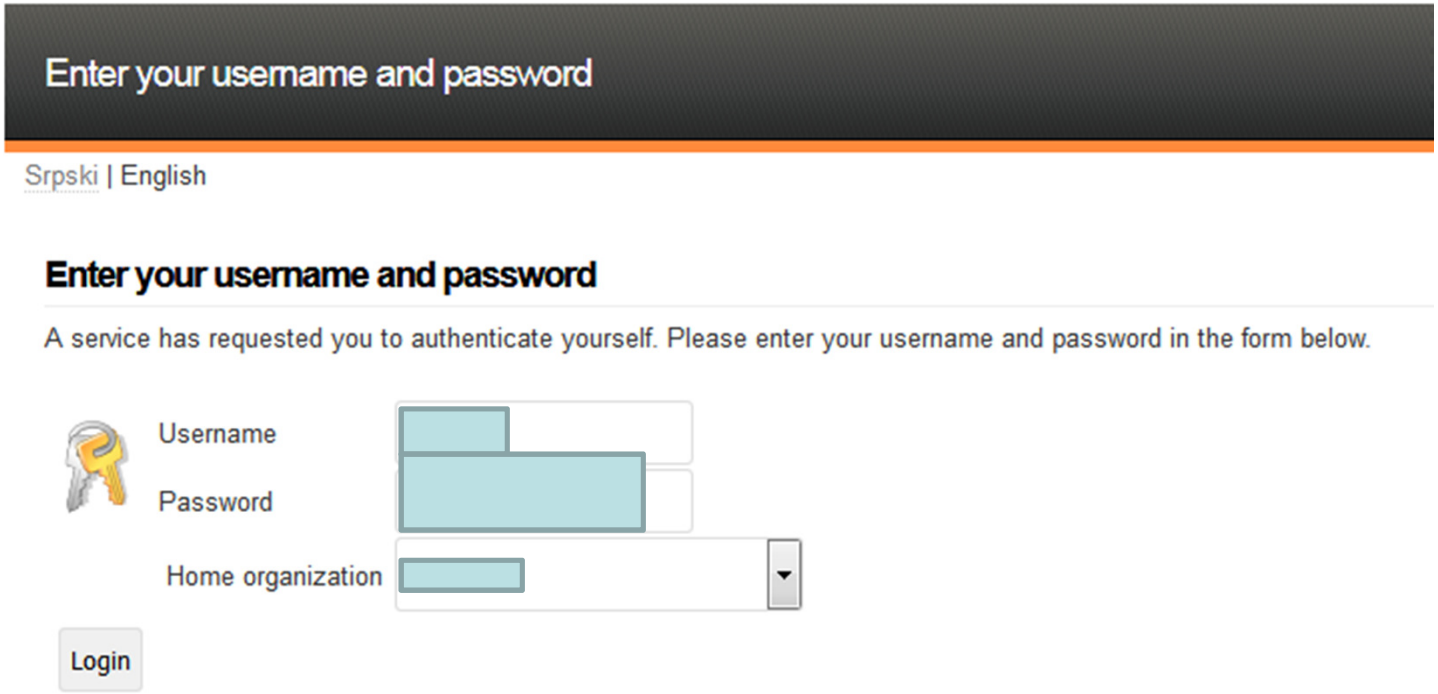
You have previously chosen to authenticate at **iAMRES Web Single Sign-on Portal**

Login at iAMRES Web Single Sign-on Portal

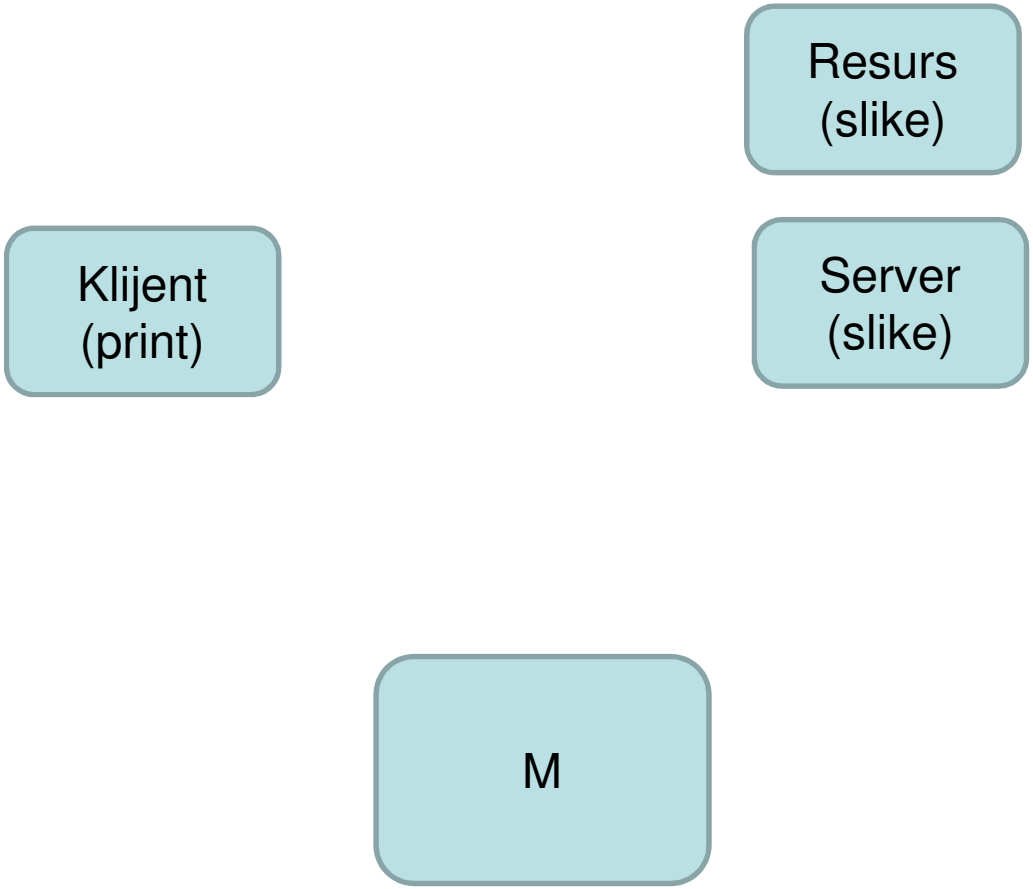
All | eduGAIN | ϕEDUrus | Chile | Spain | UKfederation | US | Italy | NZ | AU | NL | Social networks | Guest providers | Miscellaneous

Incremental search...

iAMRES Web Single Sign-on Portal


Enter your username and password
Srpski | English
Enter your username and password
A service has requested you to authenticate yourself. Please enter your username and password in the form below.
Username
Password
Home organization
Login

OAuth v1.0

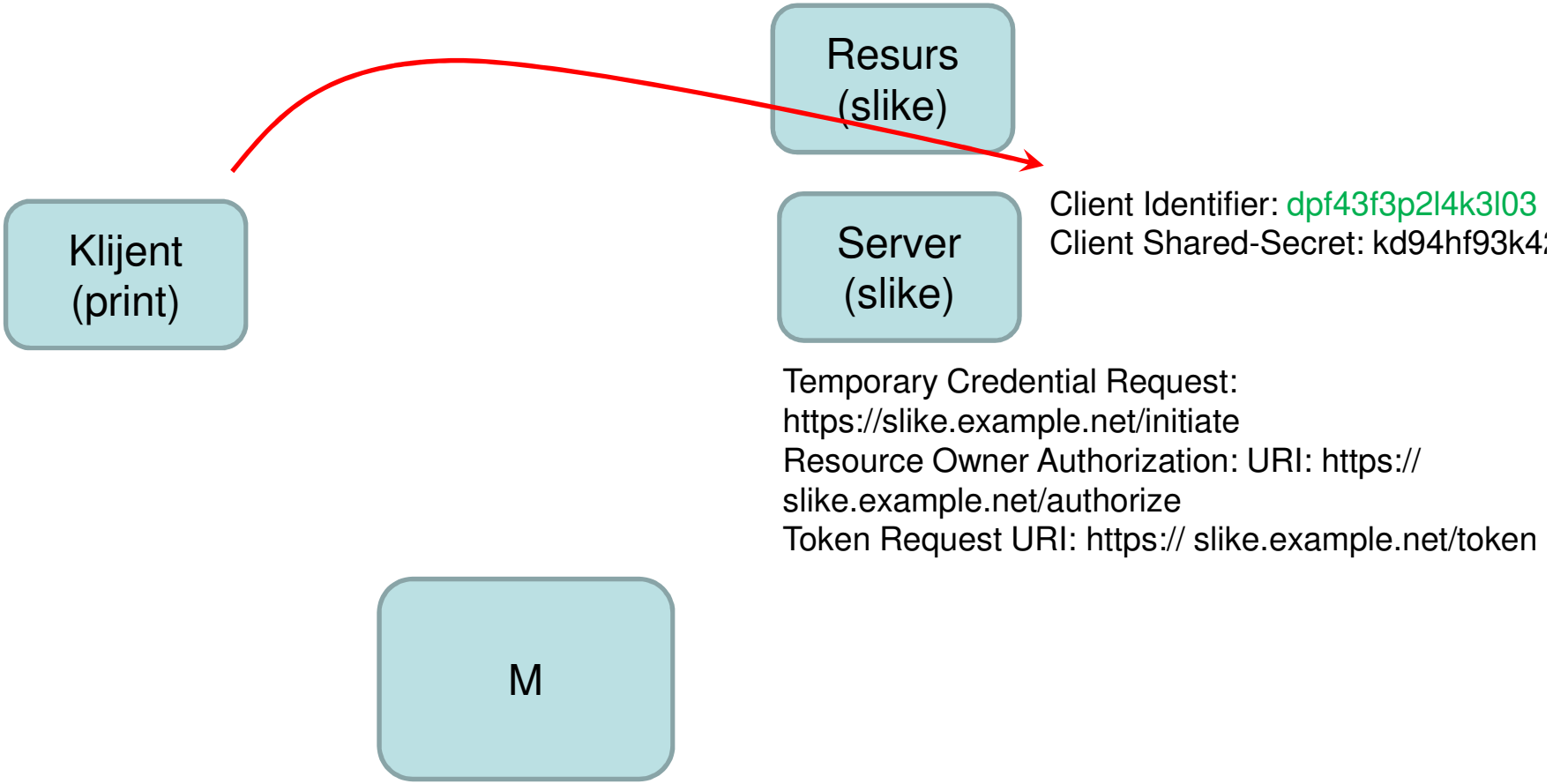


OAuth v1.0 (1)

- Zajedničko korišćenje resursa od strane različitih velikih servis provajdera (google, twitter, fb,...)
- M ima slike na slike.example.com
- M hoće da odštampa slike sa slike.example.com na printer.example.com
- Da bi omogućio pristup slikama, printer.example.com je prijavljen na slike.example.com i ima svoje podatke (pre bilo kakve komunikacije i zahteva M):

Client Identifier: dpf43f3p2l4k3l03
Client Shared-Secret: kd94hf93k423kf44

OAuth v1.0


Resurs
(slike)
Klijent
(print)
Server
(slike)
Client Identifier: dpf43f3p2l4k3l03
Client Shared-Secret: kd94hf93k4
Temporary Credential Request:
https://slike.example.net/initiate
Resource Owner Authorization: URI: https://
slike.example.net/authorize
Token Request URI: https:// slike.example.net/token
M

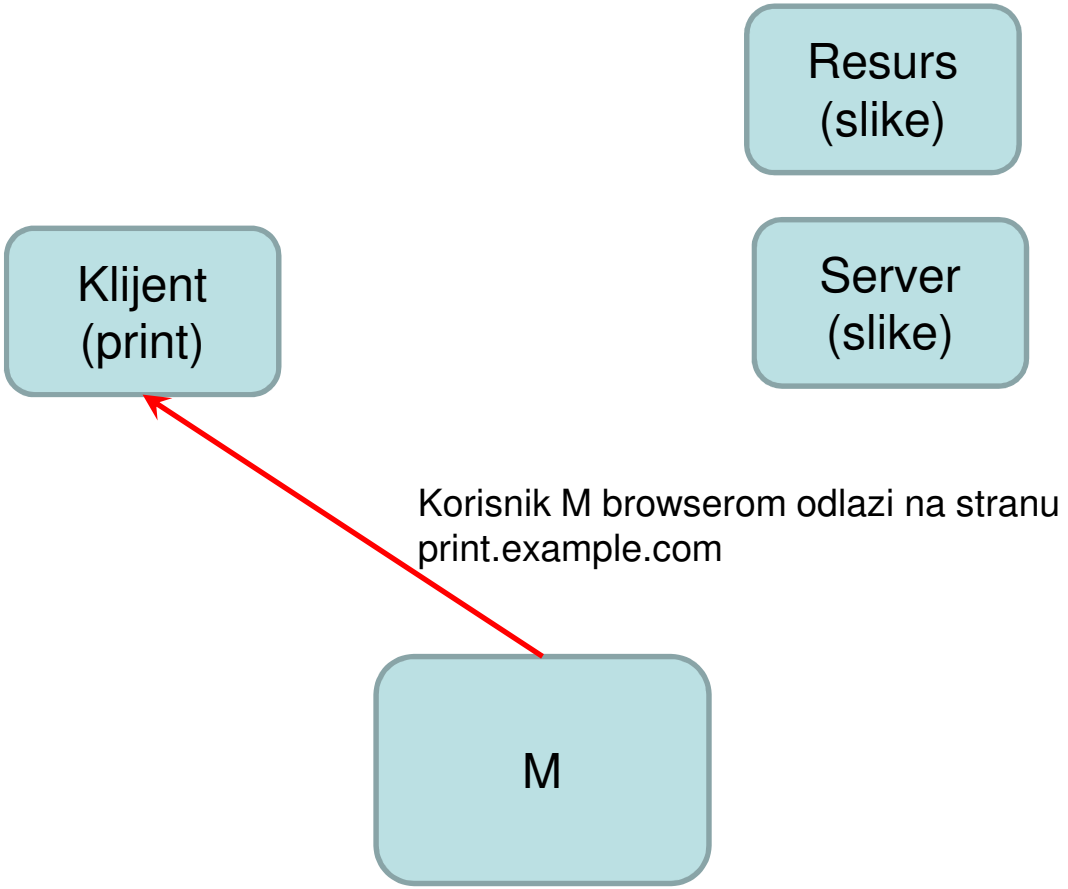
OAuth v1.0 (2)

- printer.example.com je konfigurisan tako da koristi pristupne tačke za slike.example.com:

Temporary Credential Request: https://slike.example.net/initiate
Resource Owner Authorization: URI: https:// slike.example.net/authorize
Token Request URI: https:// slike.example.net/token

- M odlazi na printer.example.com želeći da odštampa svoje slike.
- Na sajtu printer.example.com mora da postoji mogućnost pristupa slikama na slike.example.com (dugme, link,...)

OAuth v1.0


Resurs
(slike)
Klijent
(print)
Server
(slike)
Korisnik M browserom odlazi na stranu
print.example.com
M

OAuth v1.0 (3)

- printer.example.com se obraća slike.example.com sa zahtevom za privremenim kredencijalma za servis slike.example.com

```
POST /initiate HTTP/1.1
Host: slike.example.net
     Authorization: OAuth realm=„Slike",
        oauth_consumer_key="dpf43f3p2l4k3l03",
        oauth_signature_method="HMAC-SHA1",
        oauth_timestamp="137131200",
        oauth_nonce="wIjqoS",
        oauth_callback="http%3A%2F%2Fprinter.example.com%2Fready",
        oauth_signature="74KNZJeDHnMBp0EMJ9ZHt%2FXKycU%3D"
```

OAuth v1.0


```
POST /initiate HTTP/1.1
Host: slike.example.net
     Authorization: OAuth realm=„Slike",
        oauth_consumer_key="dpf43f3p2l4k3l03",
        oauth_signature_method="HMAC-SHA1",
        oauth_timestamp="137131200",
        oauth_nonce="wIjqoS",
        oauth_callback="http%3A%2F%2Fprinter.example.com%2Fready",
        oauth_signature="74KNZJeDHnMBp0EMJ9ZHt%2FXKycU%3D"
```

M

OAuth v1.0 (4)

- slike.example.com odgovara skupom privremenih token kredencijala:

```
HTTP/1.1 200 OK
     Content-Type: application/x-www-form-urlencoded

oauth_token=hh5s93j4hdidpola&oauth_token_secret=hdhd0244k9j7a
o03&oauth_callback_confirmed=true
```

- Zahtev za slikama koje se štampaju se od printer.example.com preusmerava na (vidljivo u URL browsera):

https://slike.example.net/authorize?oauth_token=hh5s93j4hdidpola

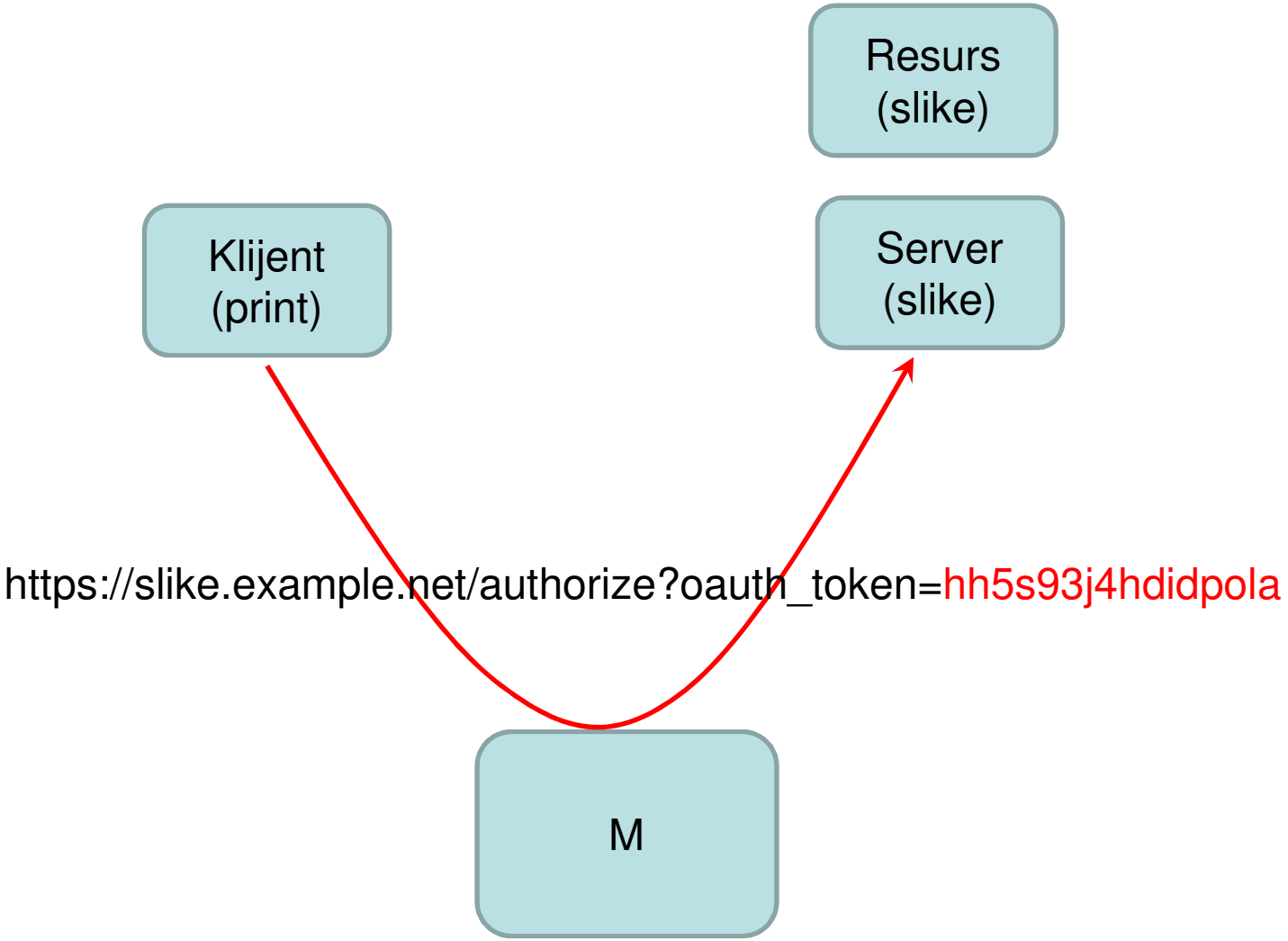
OAuth v1.0


```
HTTP/1.1 200 OK
     Content-Type: application/x-www-form-urlencoded

oauth_token=hh5s93j4hdidpola&oauth_token_secret=hdhd0244k9j7a
o03&oauth_callback_confirmed=true
```

M

OAuth v1.0


Resurs
(slike)
Klijent
(print)
Server
(slike)
https://slike.example.net/authorize?oauth_token=hh5s93j4hdidpola
M

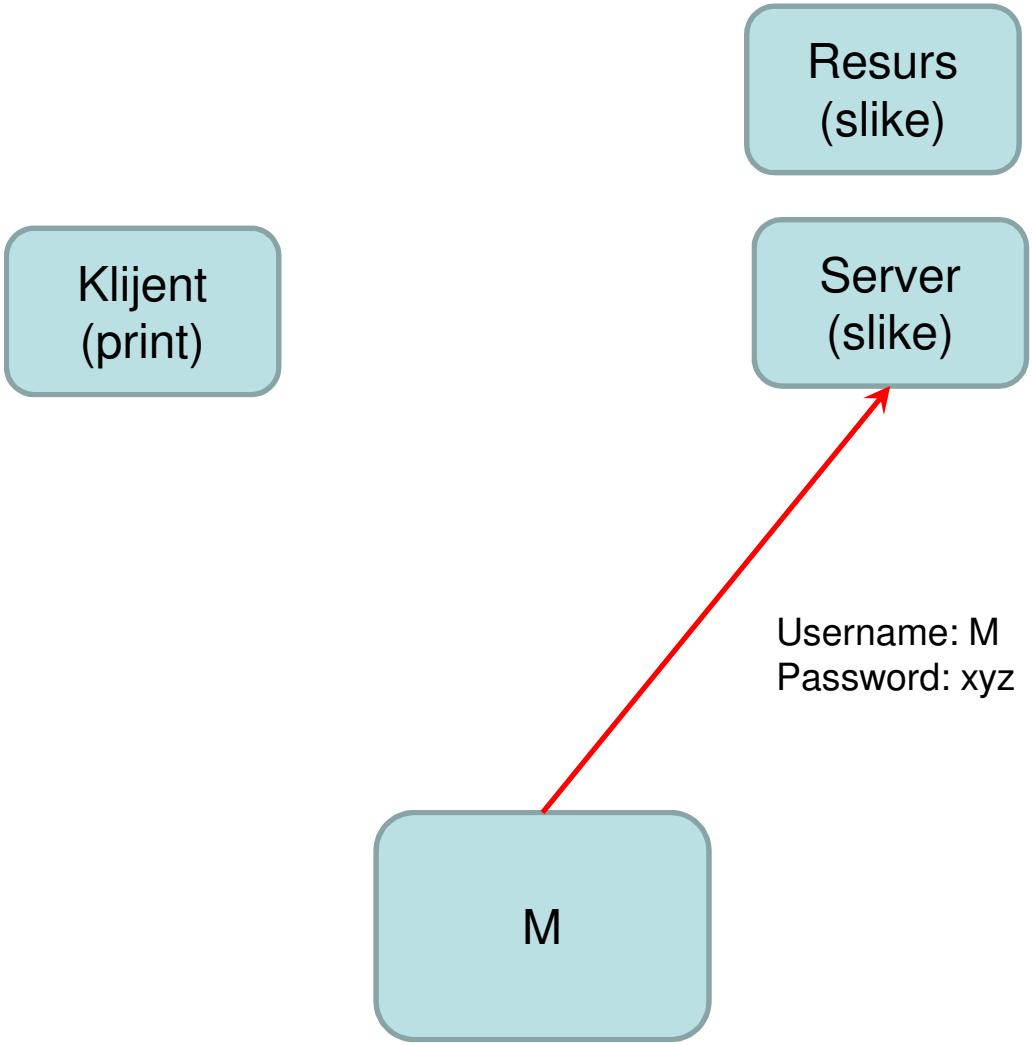
OAuth v1.0 (5)

- M unosi kredencijale za slike.example.com
- Nakon uspešnog unosa, server zahteva dozvolu da pristupi slikama i zahtev se preusmerava na oauth_callback:

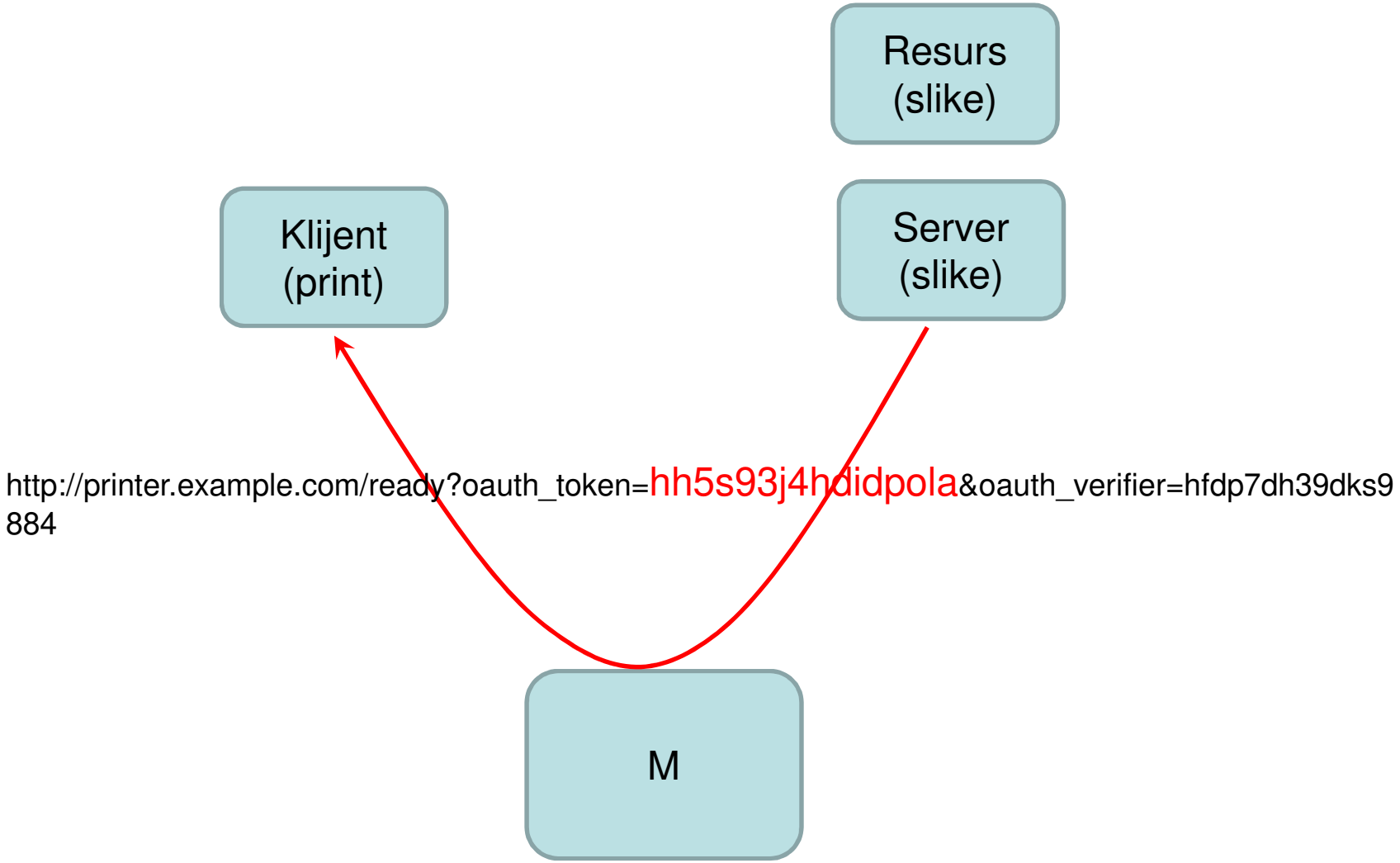
https://printer.example.com/ready?oauth_token=hh5s93j4hdidpola&oauth_verifier=hfdp7dh39dks9884

- Printer.example.com traži set tokena za akreditaciju, a u zahtevu koristi prethodno dobijene privremene tokene

OAuth v1.0


Resurs
(slike)
Klijent
(print)
Server
(slike)
Username: M
Password: xyz
M

OAuth v1.0



http://printer.example.com/ready?oauth_token=hh5s93j4hdidpola&oauth_verifier=hfdp7dh39dks9884

OAuth v1.0 (6)

```
POST /token HTTP/1.1
     Host: slike.example.net
     Authorization: OAuth realm=„Slike",
        oauth_consumer_key="dpf43f3p2l4k3l03",
        oauth_token="hh5s93j4hdidpola",
        oauth_signature_method="HMAC-SHA1",
        oauth_timestamp="137131201",
        oauth_nonce="walatlh",
        oauth_verifier="hfdp7dh39dks9884",

     oauth_signature="gKgrFCywp7rO0OXSjdot%2FIHF7IU%3D"
```

- Ako je sve u redu slike.example.com odgovara preko TLS sa trajnim tokenima:

```
HTTP/1.1 200 OK
Content-Type: application/x-www-form-urlencoded

oauth_token=nnch734d00sl2jdk&oauth_token_secret=pfkkdhi9sl
3r4s00
```

OAuth v1.0


```
POST /token HTTP/1.1
    Host: photos.example.net
    Authorization: OAuth realm="Photos",
      oauth_consumer_key="dpf43f3p2l4k3l03",
      oauth_token="hh5s93j4hdidpola",
      oauth_signature_method="HMAC-SHA1",
      oauth_timestamp="137131201",
      oauth_nonce="walatlh",
      oauth_verifier="hfdp7dh39dks9884",
      oauth_signature="gKgrFCywp7rO0OXSjdot%2FIHF7IU%3D"
```

OAuth v1.0

HTTP/1.1 200 OK
Content-Type: application/x-www-form-urlencoded

oauth_token=nnch734d00sl2jdk&oauth_token_secret=pfkkdhi9sl3r4s00

M

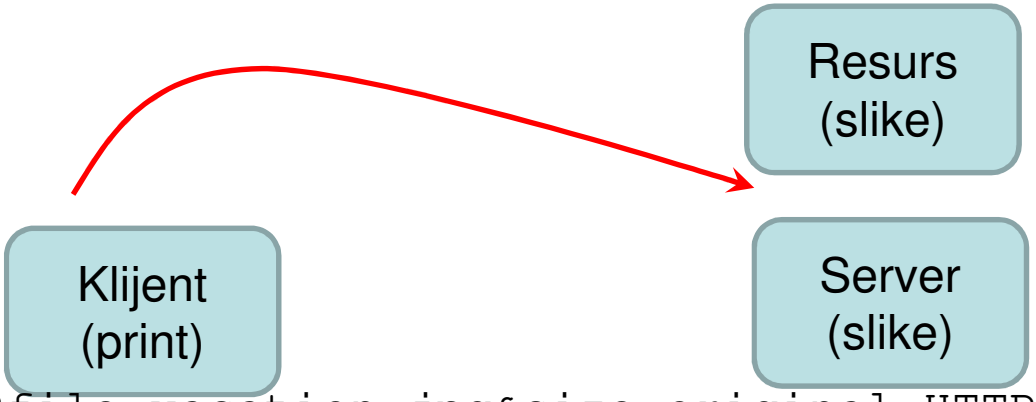
OAuth v1.0 (6)

- Printer.example.com sada traži pristup resursima na slike.example.com

```
GET /photos?file=vacation.jpg&size=original HTTP/1.1
     Host: slike.example.net
     Authorization: OAuth realm=„Slike",
        oauth_consumer_key="dpf43f3p2l4k3l03",
        oauth_token="nnch734d00sl2jdk",
        oauth_signature_method="HMAC-SHA1",
        oauth_timestamp="137131202",
        oauth_nonce="chapoH",
        oauth_signature="MdpQcU8iPSUjWoN%2FUDMsK2sui9I%3D"
```

- printer.example.com ima pristup slikama sa slike.example.com dok M ne ukine to pravo

OAuth v1.0




```
GET /photos?file=vacation.jpg&size=original HTTP/1.1
     Host: slike.example.net
     Authorization: OAuth realm=„Slike",
        oauth_consumer_key="dpf43f3p2l4k3l03",
        oauth_token="nnch734d00sl2jdk",
        oauth_signature_method="HMAC-SHA1",
        oauth_timestamp="137131202",
        oauth_nonce="chapoH",
        oauth_signature="MdpQcU8iPSUjWoN%2FUDMsK2sui9I%3D"
```

M

OAuth sigurnost

- Digitalni potpis:
  - HMAC-SHA1
  - RSA-SHA1
  - Plaintext (preko TLS)

OAuth sigurnost

- HMAC-SHA
  - *izlaz = HMAC-SHA1 (ključ, tekst)*
  - Tekst: vrednost *base string*-a koji se konstruiše od parametara koji se šalju u okviru HTTP zahteva
  - Ključ: konkatenacija sledećeg:
    - klijentova deljena lozinka
    - karakter “&” (ASCII znak 38), koji mora biti uključen loznika tokena koji se trenutno koristi
  - Izlaz: *oauth_signature*

OAuth sigurnost

- RSA-SHA1
  - *S = RSASSA-PKCS1-V1_5-SIGN (K, M)*
  - RFC3447
  - K – RSA privatni ključ klijenta
  - M – base string

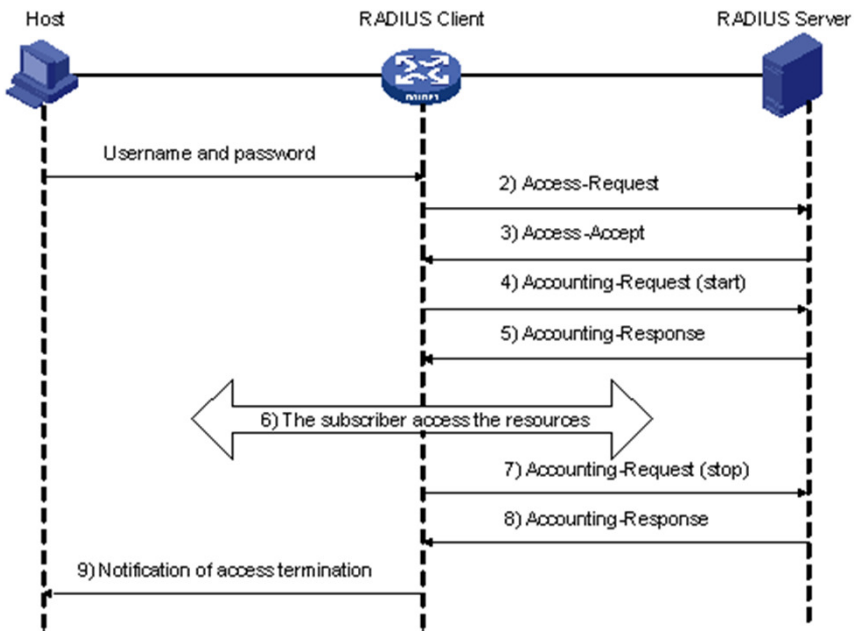
OAuth sigurnost

- Base string:
  - GET, POST, ... parametri se uvek pišu velikim slovom.
  - dodavanje „&” karaktera (ASCII znak 38)
  - URL zahtevanog resursa sa svim svojim parametrima u URL-u
  - dodavanje „&” karaktera (ASCII znak 38)
  - sve parametre redom sem *oauth_signature* parametra

OAuth v2.0

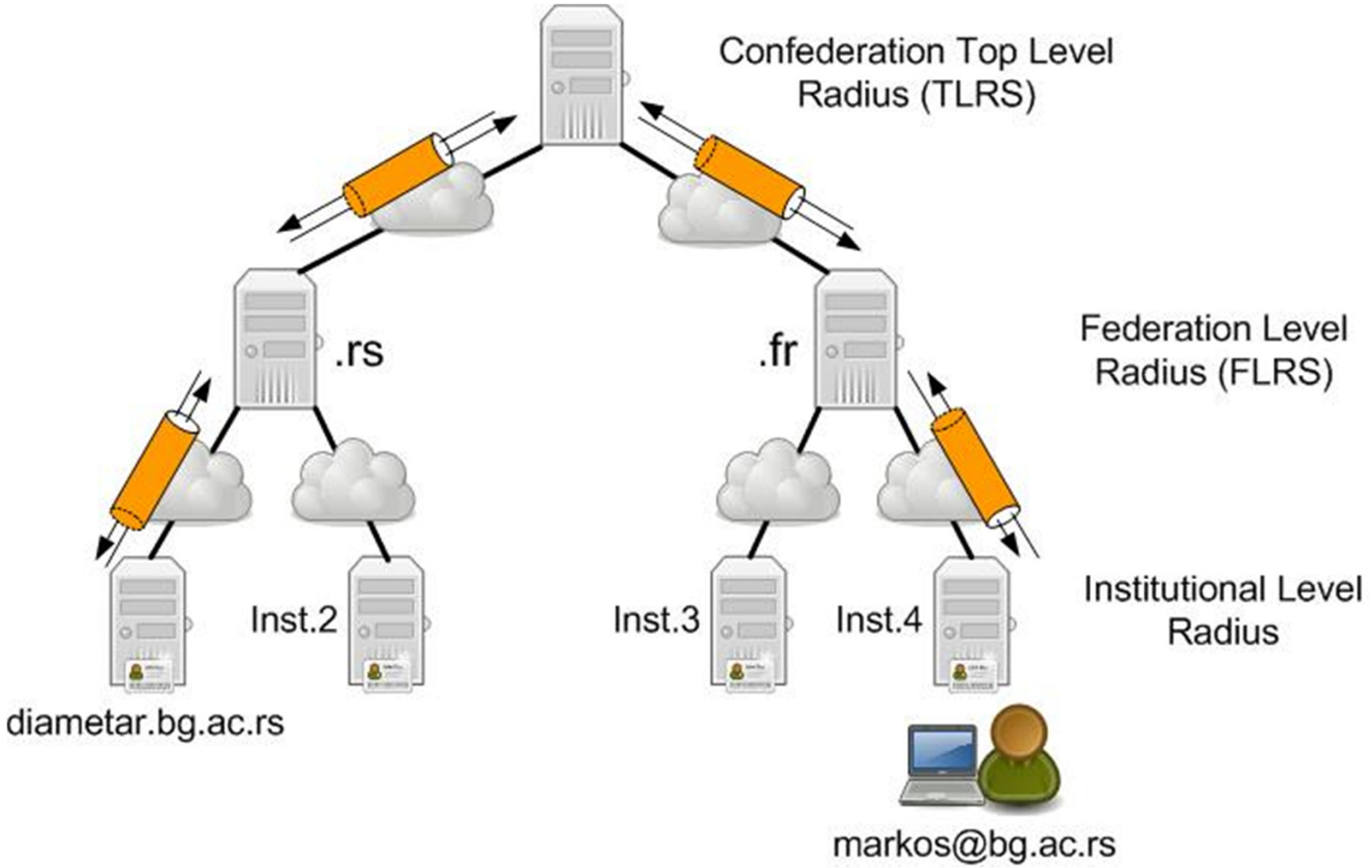
- Nije kompatibilan sa OAuth v1.0
- Pojednostavljen način rada (v1.0 je predstavljao problem za velike pružaoce usluga)
- Manje opterećenje servera

RADIUS protokol



- RFC 2865, 2866
- AAA protokol
- Proširiv velikim brojem AV parova
- RADIUS client – RADIUS server – preshared secret + MD5
- Preporučuje se IPsec

Hijerarhijska struktura Radius servera



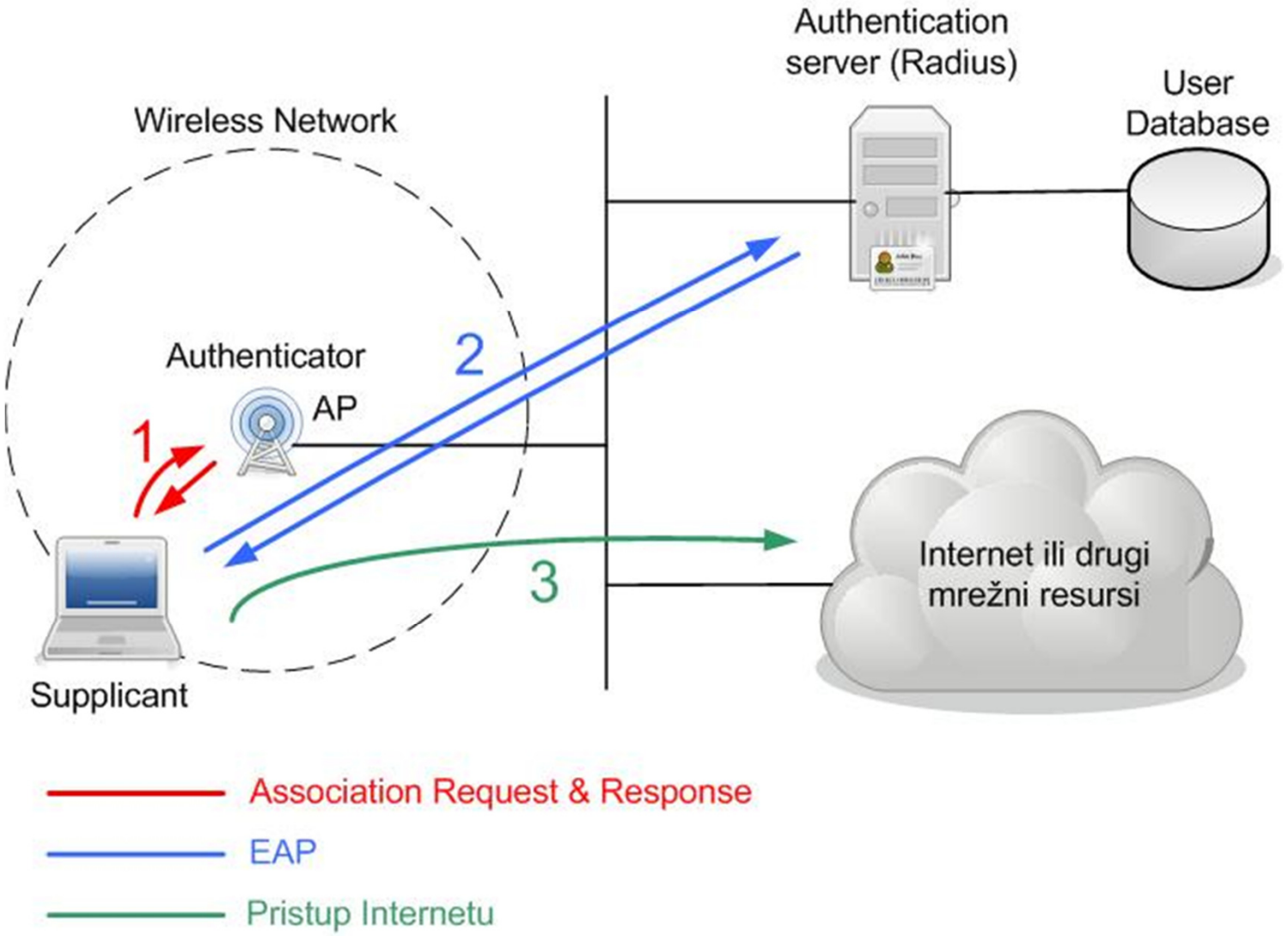
Hijerarhijska struktura RADIUS servera

- Osnovna ideja je da se kroz strukturu RADIUS servera proslede korisničke informacije do korisnikove matične organizacije

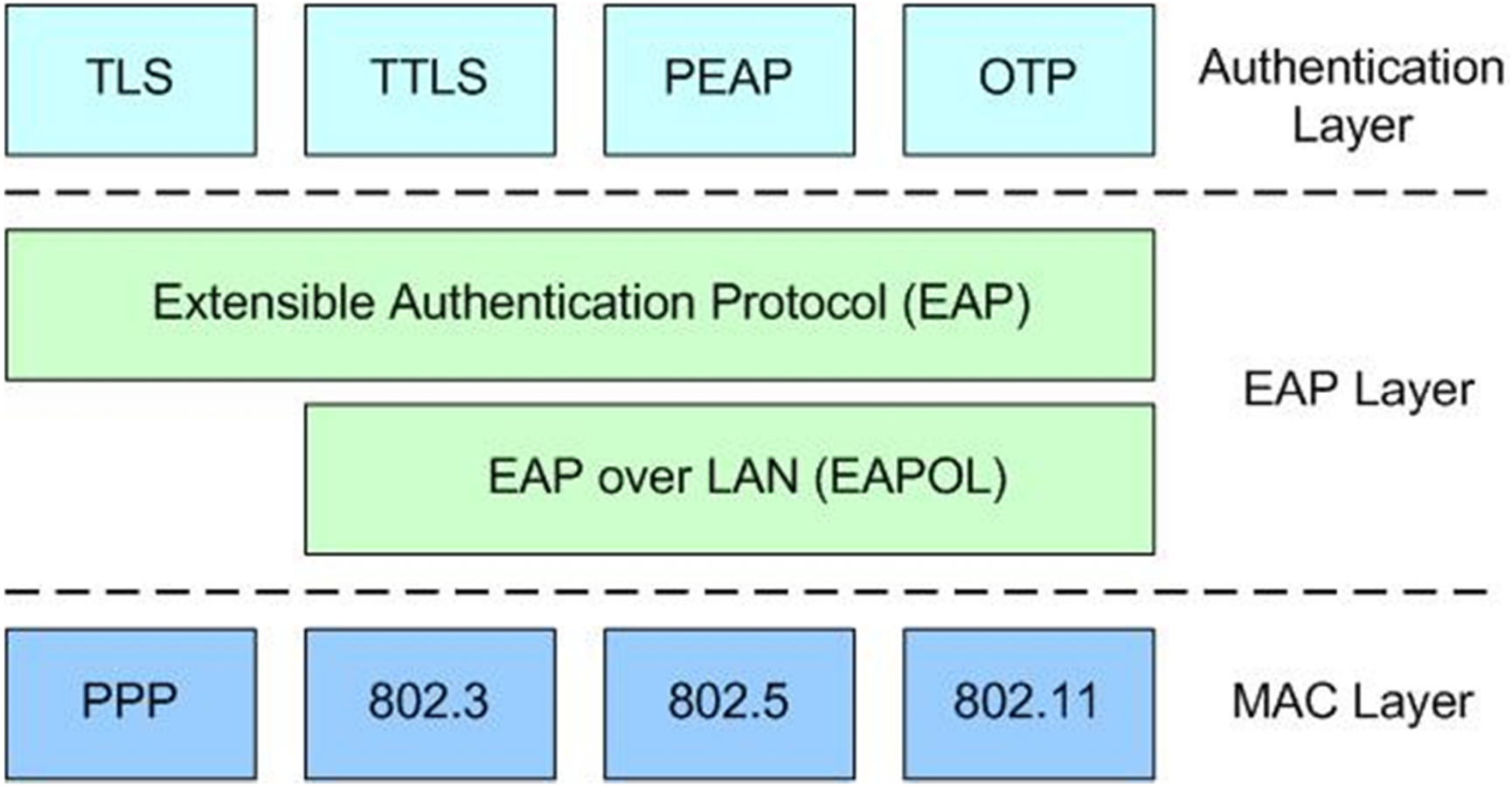
- Rutiranje zahteva se vrši na osnovu domena (*realm*-a):

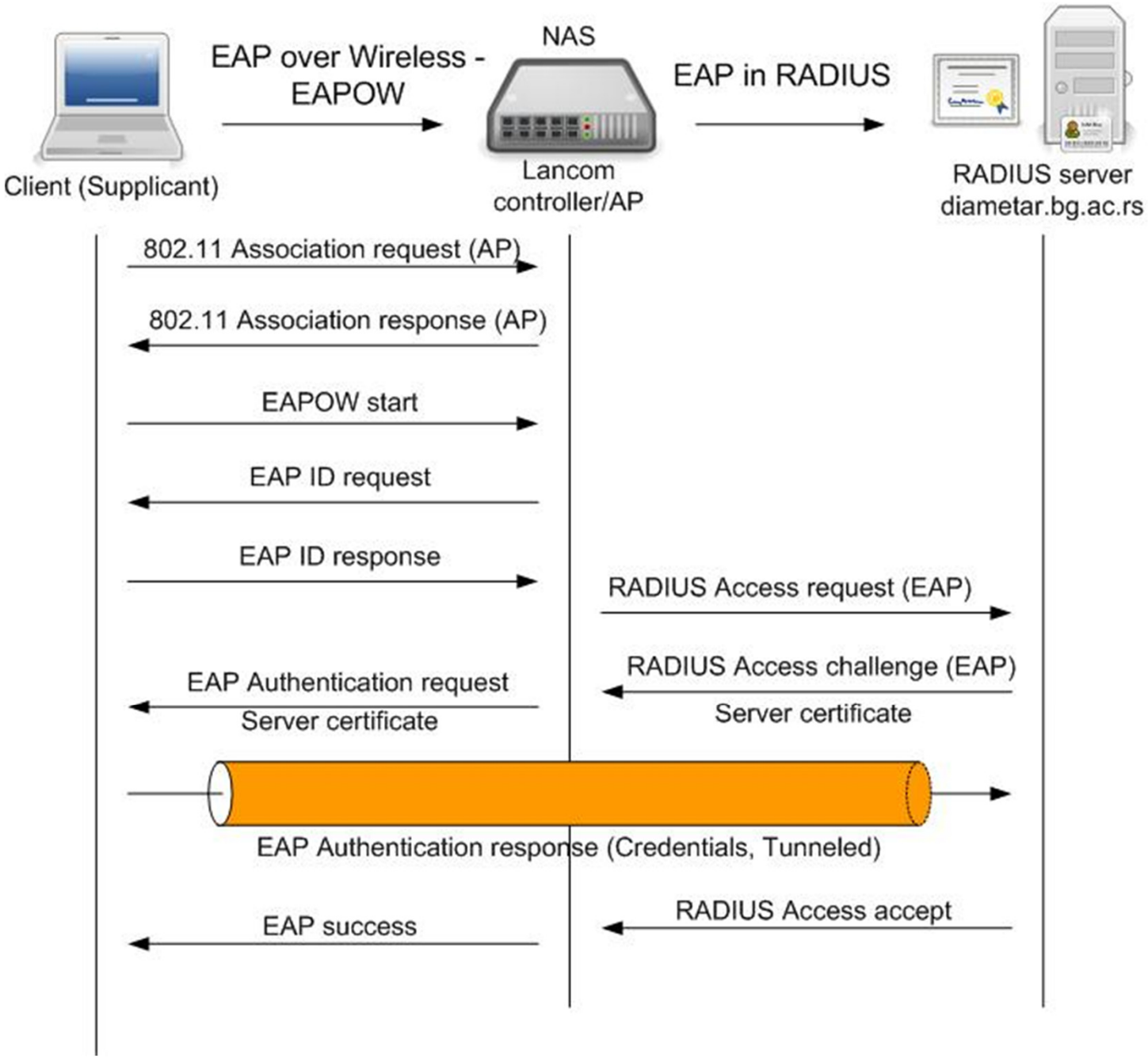
  user-name@home_institution_domain

802.1x

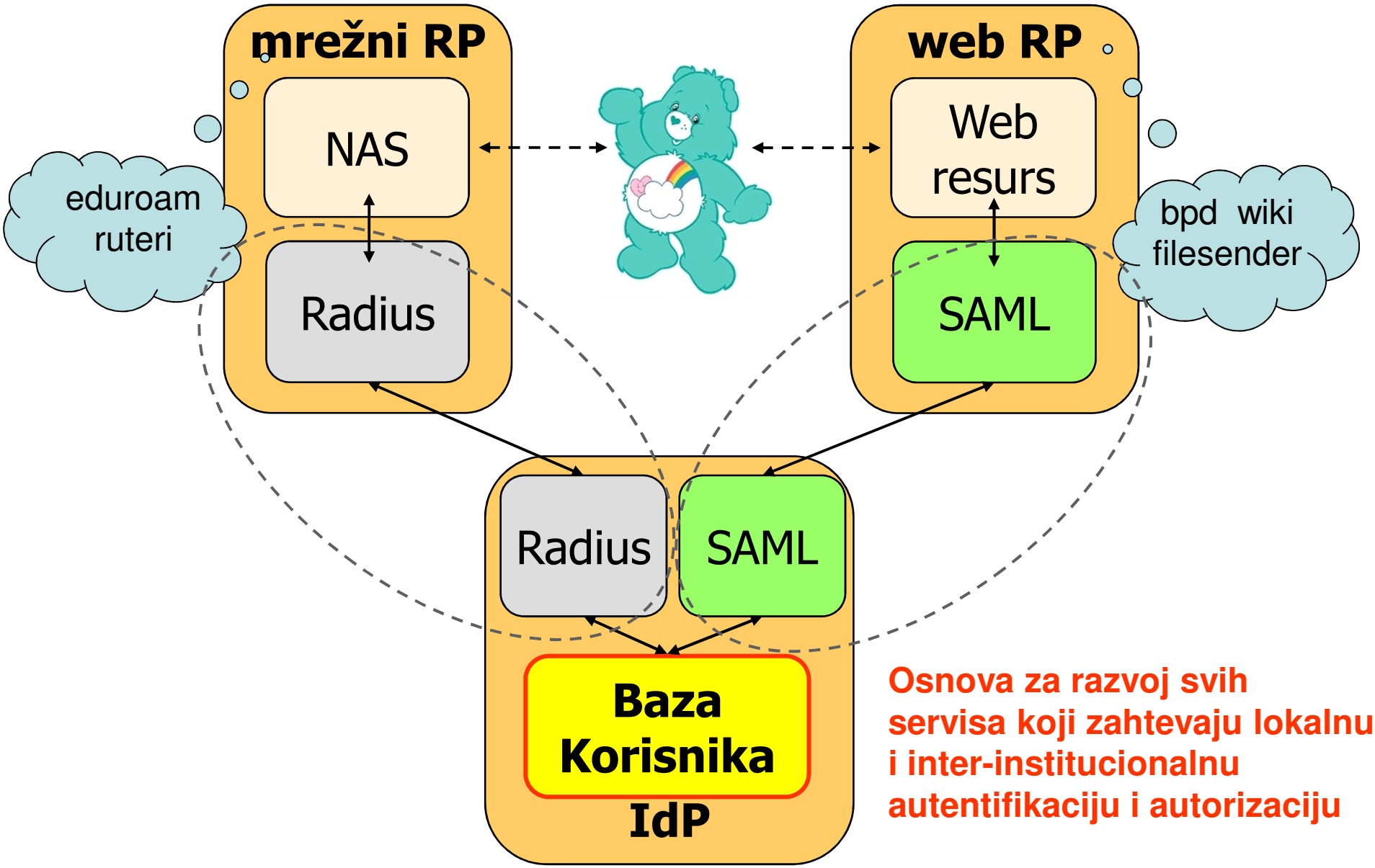

Authentication server (Radius)
User Database
Wireless Network
Authenticator
AP
1
2
3
Internet ili drugi mrežni resursi
Supplicant
Association Request & Response
EAP
Pristup Internetu

EAP Framework




NAS
EAP over Wireless - EAPOW
EAP in RADIUS
Client (Supplicant)
Lancom controller/AP
RADIUS server diametar.bg.ac.rs
802.11 Association request (AP)
802.11 Association response (AP)
EAPOW start
EAP ID request
EAP ID response
RADIUS Access request (EAP)
RADIUS Access challenge (EAP)
EAP Authentication request
Server certificate
Server certificate
EAP Authentication response (Credentials, Tunneled)
RADIUS Access accept
EAP success

Dijagram AAI na visokom nivou



Osnova za razvoj svih servisa koji zahtevaju lokalnu i inter-institucionalnu autentifikaciju i autorizaciju