**Egzaminas**

Gali laikyti tik atsiskaitę už semestro darbus studentai.

Apimtis: du teoriniai klausimai ir du uždaviniai iš pratybų medžiagos.

Trukmė 1 val. Raštu.

Atsakymai į teorinius klausimus turi būti išsamūs ir kvalifikuoti, atskleista paskirtis, sandara, veikimo principai, apibūdintos pagrindinės savybės. Remtis paskaitų medžiaga, papildoma literatūra ar kitų universitetų rekomenduotais vadovėliais.

Sprendžiant uždavinius parodyti sprendimo eigą.

Teoriniai klausimai:

1. Autorizacija, prieigų nustatymo mechanizmai. Autentifikacijos metodai

**Autorizacija**

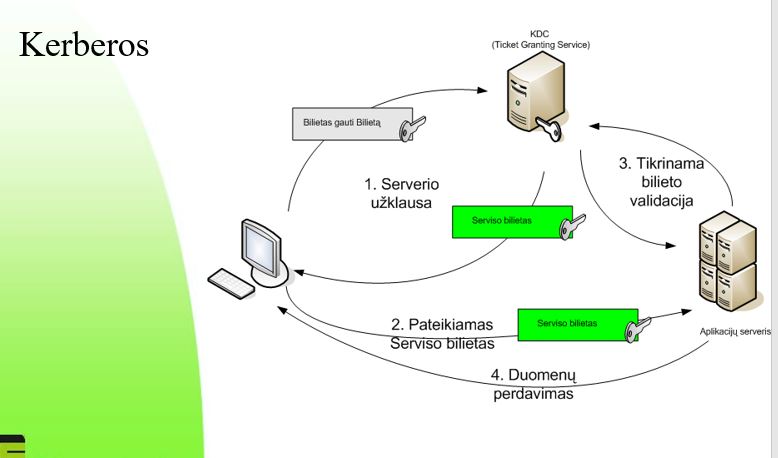
Prieigų nustatymo mechanizmai:

* DAC - Kiekvienas objektas turi sąrašą, aprašantį, kokie subjektai turi konkrečias teises (skaityti, rašyti, vykdyti).
* RBAC - Prieigos kontrolė priklauso nuo rolės. Subjektai priklauso konkrečiai rolei. Subjektas gali priklausyti tik vienai rolei. Prieigos teisės aprašomos rolėmis
* MAC - Privaloma teisių valdymo strategija. Failo savininkas neturi galimybės suteikti sukurtam failui teisių.

**Autentifikacija**

Autentifikacijos metodai:

* Kerberos -



* CHAP - CHAP buvo plėtojamas kaip dalis TCP/IP point-to-point protokolo (PPP), naudojamo perduoti TCP/IP duomenis per dial-up sujungimus. Jis buvo apibūdintas RFC dokumente 1994 metais.
* EAP - EAP suteikia struktūrą skirtingoms autentifikacijos technologijoms. Jis plačiai taikomas nuotoliniams ryšiams ir wireless autentifikacijai. EAP naudojamas kartu su smart card’ais ir biometrika ar paprastesniais duomenimis (vartotojo vardai ir slaptažodžiai).

**SSH (Secure shell)** – tai tinklo protokolas, aprašantis apsaugotą kliento prisijungimą prie serverio aplinkos (shell) ir komandų vykdymą. Standartinis ssh prievadas (portas) yra 22 (TCP).

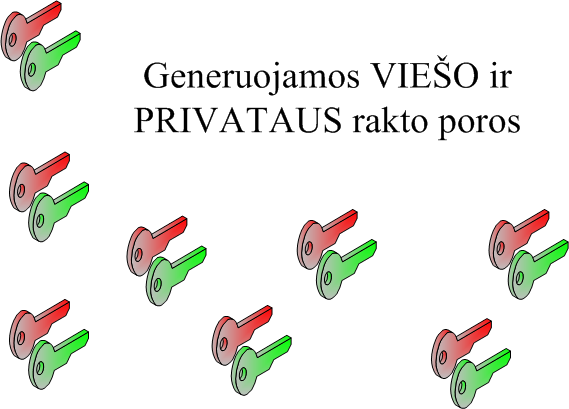
1. Viešųjų raktų infrastruktūra. Sertifikatai

**Viešųjų Raktų Infrastruktūra** (VRI, angl. Public Key Infrastructure – PKI) yra techninės, programinės įrangos, žmonių ir procedurų visuma, kuri naudojama saugoti, kurti, valdyti, suteikti, atnaujinti sertifikatus viešojo rakto kriptografijos metodais.

**Viešųjų raktų infrastruktūros** (PKI) taikymas:

* Elektroniniams parašams;
* El. paštui šifruoti;
* Dokumentams šifruoti ir (ar) autentifikuoti;
* Vartotojams autentifikuoti informacinėse sistemose;
* GRID/Cloud tinkluose.

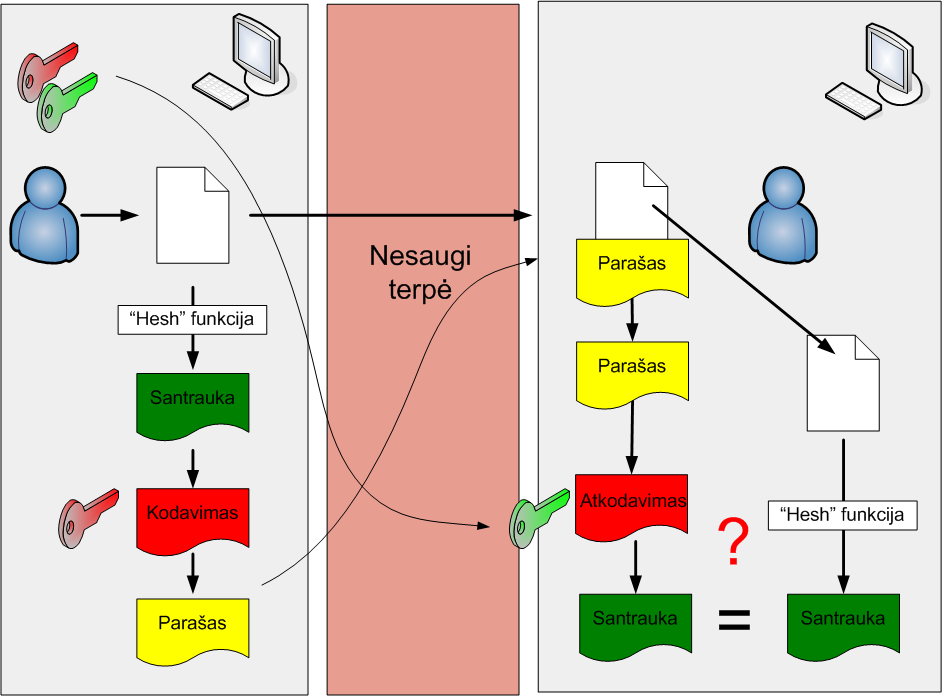
PKI sertifikatai



Šifravimo būdai:

* Simetrinis
* Asimetrinis
* Hibridinis

**Elektroninis parašas** – tai duomenys, kurie įterpiami, prijungiami ar logiškai susiejami su kitais duomenimis pastarųjų autentiškumui patvirtinti ir (ar) pasirašančiam asmeniui identifikuoti.



1. WEB1, WEB2, WEB3, HTML

Web1:



WEB2: Esmė –interneto turinį bendradarbiaudami gali kurti visi (Wisdom of Crouds)

“Read-Write” Web –žiniatinklis ir skaitymui, ir rašymui.

Žiniatinklis –tai tarsi didelė kompiuterinė sistema (platforma), kuriai kurioje kuriami ir vykdomi įvairūs uždaviniai.

WEB3:

* Mobilumas–sprendimai, pritaikyti mobiliems  
   prietaisams
* Statistika –~50% vartotojų internetu naudojasi per  
   mobilųjį telefoną
* Internetas reikiamoje vietoje ir reikiamu laiku
* Galimybės išnaudoti vartotojo kontekstą (pvz. nuo buvimo vietos priklausomos paslaugos)
* Vartotojas gali dar patogiau kurti įvairialypį interneto turinį:
* Tobulėjančios multimedia valdymo galimybės telefone
* Informacija sukuriama ir įkeliama norimoje vietoje ir norimu laiku



HTML:

HTML (Hyper text Markup Language „Hiperteksto žymėjimo kalba“) – tai kompiuterinė žymėjimo kalba, naudojama pateikti turinį internete. Kalbą standartizuoja W3 konsorciumas.

Pagrindinis HTML kalbos vienetas yra elementas. Kaip ir [XML](https://lt.wikipedia.org/wiki/XML), HTML elementas turi vardą ir gali turėti bet kokį skaičių atributų. Elemento viduje gali būti tekstas bei kiti elementai. Tiek tekstas, tiek ir dukteriniai elementai paprastai gali kartotis ir sekti bet kuria tvarka.

HTML versija žymima skaičiumi, nurodant versiją ir subversiją, pavyzdžiui, HTML 3.2.

Paskutinė [W3C](https://lt.wikipedia.org/wiki/W3C) rekomenduojama HTML versija 4.01. Po jos HTML pagrindu buvo plėtojamas [XHTML](https://lt.wikipedia.org/wiki/XHTML) standartas, kuris dar labiau atskyrė struktūros logiką nuo vaizdavimo, bei remiasi ne [SGML](https://lt.wikipedia.org/wiki/SGML), o [XML](https://lt.wikipedia.org/wiki/XML) žymėjimo kalba.

2009 m. [vasario 12](https://lt.wikipedia.org/wiki/Vasario_12) d. [WWW architektų taryba](https://lt.wikipedia.org/wiki/W3C) patvirtino **HTML 5** juodraštį[[1]](https://lt.wikipedia.org/wiki/HTML#cite_note-1). HTML 5 keičia ir HTML 4, ir XHTML 1, bet išlieka suderinamas su jais. Naujasis standartas smulkiai ir iki galo tiksliai aprašo žymėjimo gaires, nepalikdamas vietos interpretacijai – naršyklės turės vienodai atvaizduoti tinklalapius, vienodai elgtis net ir su HTML rašto klaidomis.

Skirtingos HTML kalbos versijos dar vadinamos HTML specifikacijomis.

1. CSS, XML, XSD, XSL

* CSS – kalba, naudojama naršyklėje, kuri nusako kaip atvaizduoti tam tikrą elementą;
* Anksčiau HTML kalboje buvo naudojamos tam tikros žymos, kurios buvo skirtos nurodyti stilius, bet laikui bėgant jas pakeitė CSS stilius.
* CSS stiliaus aprašymai gali būti saugomi atskiruose .css failuose arba talpinami į .html kodą.
* Kuriant puslapius ir juose naudojant CSS kalbą patartina juos patikrinti keliose naršyklės. Nes atsiradus CSS standarto pakeitimams ne visos naršyklės spėja atsinaujint ir tuomet iškyla problemų su CSS palaikymu.
* CSS sukūrimas buvo didelis šuolis internetinių dizainų kūrime;
* CSS dėka galime nurodyti visiškai visą puslapio dizainą, teksto šriftus, išdėliojimą, spalvas ir atstumus tarp įvairių elementų.
* CSS leidžia apibrėžti stilių, bet kokiam konkrečiam HTML elementui ir priskirti jį keliems puslapiams, tad norint pakeisti puslapio teksto stilių užtenka paredaguoti CSS failą.
* Kuriant dizainą rekomenduojama CSS stilių talpinti ne pačiame .html faile, o atskirame .css faile.
* Kas yra XML?
  + XML - EXtensible Markup Language - gairių kalba, sukurta duomenų aprašymui
  + XML gairės nėra fiksuotos. Gaires reikia pačiam susikurti
  + XML su DTD (Document Type Definition) ar su XML schema yra “save aprašanti” kalba
  + XML yra W3C standartas nuo 1998 m.
* XML ir HTML skirtumai:
  + XML sukurta duomenų aprašymui, HTML – duomenų vaizdavimui
  + XML nieko neatlieka; XML skirta sudaryti duomenų struktūras, saugoti ir siųsti duomenis
  + XML yra išplečiama kalba
  + XML yra HTML papildymas
  + XML Web kūrime taip pat svarbi kaip HTML
* XML
  + Naudojant XML duomenys saugomi atskirai nuo HTML
  + Naudojant XML galima keistis duomenimis tarp nesuderinamų sistemų
  + Naudojant XML tekstinius failus galima paprastai saugoti ir dalintis duomenimis
  + XML galima naudoti naujų kalbų kūrimui (pav. WML)
  + Visos ateities taikomosios programos duomenų apsikeitimui naudos XML?
* Pagrindinės su XML susijusios technologijos
  + XML DOM – XML manipuliavimo standartas
  + XSL – XML stilių kalba
  + DTD – dokumentų tipų aprašymas – aprašo XML elementus
  + XSD (XML schema) – DTD alternatyva
  + SOAP – XML pagrindu veikiantis protokolas
  + WSDL – XML pagrindu sudaryta WEB paslaugų aprašymo kalba

RSS – naujienų peržiūros ir platinimo protokolas

XSL

* XSL - EXtensible Stylesheet Language:
  + kalba, aprašanti XML dokumento transformaciją į HTML dokumentą arba į kitokios struktūros XML dokumentą
* Visos dabartinės Interneto naršymo programos supranta XML ir XSLT

XML atvaizdavimas naudojant XSL

* XSL – modernesnė stilių kalba:
* XSL turi, pavyzdžiui, ciklus, kuriuos naudojant galima nagrinėti XML dokumentą ir jį atitinkamai pavaizduoti
* XSL naudojama versti XML į HTML
* XSL – XML stilių kalba

**XSD - XML schema**

XSD

* + XML schema yra XML pagrindu sudaryta DTD (dokumentų tipo aprašymas) alternatyva
  + XML schema aprašo XML dokumento struktūrą
  + XML schemos kalba yra XSD (XML Schema Definition)
  + XSD (XML schema) – DTD(Dokumentų tipų aprašymas) alternatyva

1. OSI modelio sluoksniai. Komunikavimo procesas

**OSI modelio sluoksniai**

* Taikymo – skirtas **vartotojui**, pvz.HTTP aprašo sąveiką “naršyklė - WEB serveris”
* Pateikimo - duomenų **formatai**, šifravimas
* Sesijos - autentifikacija, **ryšio paruošimas, eiga ir nutraukimas**
* Transporto - apsikeitimas **tarp taikomųjų procesų**
* Tinklo - **transportavimas tinklu**, adresacija, maršrutų parinkimas
* Kanalo - kadrai, antraštės, perdavimas **tarp gretimų mazgų**
* Fizinis - **signalai**, jungtys, dažniai ir pan.

**Komunikavimo procesas:**

**1. *Taikymo sluoksnis: Vartotojas naudojasi naršykle***

***ir per ją kreipiasi į tinklo paslaugą***

**2. *Taikymo sluoksnis: Naršyklė prie vartotojo***

***duomenų prideda antraštę***

3. ***Taikymo sluoksnis: Naršyklė prie vartotojo***

***duomenų prideda antraštę ir perduoda paketą***

***transporto sluoksniui - TCP***

4. ***Transporto sluoksnis: TCP formuoja sujungimo***

***su paslauga prašymo paketą (SYN), Antraštėje nurodo***

***paslaugos rūšį (portą 80) ir atidaro savo portą 1212***

***duomenų priėmimui***

5. ***Transporto sluoksnis: TCP formuoja sujungimo***

***su paslauga prašymo paketą (SYN), Antraštėje nurodo***

***paslaugos rūšį (portą 80) ir atidaro savo portą 1212***

***duomenų priėmimui.***

***Perduoda paketą tinklo sluoksniui – IP***

***6. Tinklo sluoksnis: Prieš formuojant antraštę reikia gauti***

***IP adresą.***

***7. Į IP antraštę įrašo gavėjo ir siuntėjo IP adresus ir***

***perduoda į Ethernet sąsają***

8. ***Kanalo sluoksnis: Jis perduos paketą tik lokalaus tinklo***

***ribose, Taigi ne galiniam gavėjui 94.16.59.10 o tarpiniam***

***mazgui 193.12.0.254. Reikia sužinoti šio mazgo MAC adresą***

1. Kanalo sluoksnis L2. Komutavimo algoritmas

* **OSI kanalo sluoksnis užtikrina duomenų paketų (kadrų) formavimą ir perdavimą tarp gretimų tinklo mazgų.**
* **Kanalo sluoksnyje siuntėjas suformuoja duomenų paketą (kadrą), kurio antraštė turi gavėjo ir siuntėjo fizinių sąsajų adresus (Media Access Control - MAC adresus)**
* **Tarpinis kanalo sluoksnio įrenginys nekeičia perduodamo paketo**
* **Kanalo sluoksnio įrenginys turi turėti atmintinę priimtam duomenų paketui įsiminti**
* **L2 perdavimas (Ethernet paketais) vyksta tik vieno LAN ribose**

Komutavimo algoritmas:

* **Priimti į Port1 ateinantį paketą. Žiūrėti jo antraštę: siuntėjo ir gavėjo adresus.**
* **1. Apsimokymas:**
* **Jei siuntėjo adreso dar nėra MAC adresų lentelėje, įrašyti į MAC adresų lentelę (naujas\_siuntėjo adresas, Port1)**
* **Jei siuntėjo adresas yra MAC adresų lentelėje, tačiau ten nurodyta kita jungtis, pakeisti įrašą MAC adresų lentelėje (su išlygom)**
* **2. Persiuntimas:**
* **Nustatyti išėjimo jungtį (Port2) iš MAC adresų lentelės pagal gavėjo adresą.**
* **Jei Port2=Port1 > paketą sunaikinti (filter)**
* **Jei Port2≠Port1 > paketą perduoti (forward) į Port2**
* **Jei gavėjo adreso nėra lentelėje > paketą paskleisti (flood) per visas jungtis, išskyrus Port1**
* **Išmesti įrašus, kuriems baigėsi galiojimo laikas iš MAC adresų lentelės**
* **Flood – viso tinklo užtvindymas tuo pačiu paketu. Panašus “broadcast” efektas,**

**tačiau prigimtis kita: broadcast paketas specialiai skiriamas visiems tinklo mazgams**

1. IEEE 802 standartai. Komutatoriai.

* **10 Mbps IEEE802.3**
* **100 Mbps IEEE802.3u**
* **1000 Mbps IEEE802.3z**
* **10 Gbps IEEE802.3ae**
* **40 ir 100 Gbps IEEE802.3bm-2015**

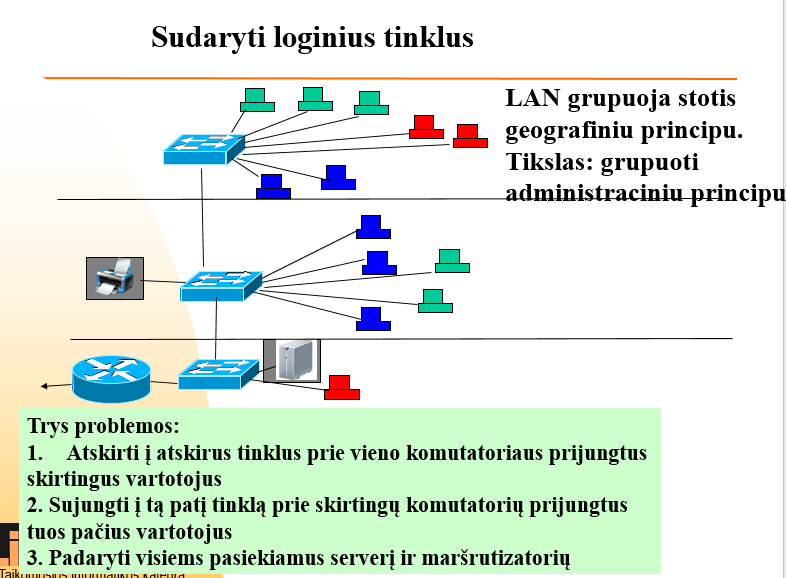
**Komutatoriai:**

* Nekeičia persiunčiamo paketo turinio, nei formato
* Niekaip papildomai neįpakuoja: perduodama tiksli kopija
* Yra skaidrus (nematomas) stotims: visos jos lieka viename LAN’e
* Neturi jokių adresų
* **Tik medžio topologija – kilpos negalimos**

**Našumo ch-kos:**

* **Filtravimo (filtering)/ Perdavimo (forwarding) sparta.**
* **Maksimalus pralaidumas (throughput) –visomis kryptimis vienu metu**
* **Pilnas pralaidumas (Switching Capacity - iki Full wire speed)**
* **Komutatoriais kompiuteriai jungiami į vieną tinklą, tinklai tarpusavyje jungiami maršrutizatoriais**
* **Visi tarpusavyje sujungti komutatoriai ~ vienas didelis**
* **Komutatoriai jungiami be kilpų**
* **Komutatorius iš esmės yra nematomas tinkle automatiškai veikiantis įrenginys, neturi jokių adresų**

1. Virtualūs lokalūs tinklai



* Paprasčiausias būdas sukurti virtualius tinklus, užtikrina didžiausią saugumą
* Įrenginiai priskiriami vienam ar kitam VLAN pagal tai, prie kurios jungties jie prijungti
* Komutatorius virtualiai padalintas į kelis komutatorius, kurie
* tarpusavyje paketų neperduoda
* Jungiant VLANus per kelis komutatorius,

reikia išvengti.

1. Optinių kabelių parametrai

**Optinio signalo stiprumo pakitimo parametrai:**

* + Signalo silpimas, dB/km
  + Netiesiniai iškraipymai (harmonikos- λ1, λ2,...)
  + Triukšmų įtaka (Signalas-trukšmas santykis –SNR)

**Optinio signalo trukmės parametrai:**

* + Dispersija (impulso trukmės pakitimas ). Priežastis - signalo dedamųjų skirtingas vėlinimas

a) modinė dispersija (modų skirtingas vėlinimas), MHz∙km

b) chromatinė dispersija (Spektro dedamųjų skirtingas vėlinimas), ps/nm/km

c) modų poliarizacijos dispersija (Optinių gijų nesimetriškumo sąlygotas dedamųjų skirtingas vėlinimas ), ps/√km

1. CWDM ir DWDM sistemos

-------------------------------------------

1. Debesų kompiuterijos ir paslaugų tipai, virtualizacijos būdai

Paslaugos užsakymo procesas automatizuotas

Platus paslaugų pasiekiamumas įprastais tinklais

Abstraktūs resursų telkiniai, palaikantys daugelį įzoliuotų

klientų

Resursai pagal poreikį nesudėtingai plečiami ar

sumažinami

Panaudotų paslaugų ir resursų apskaitomumas

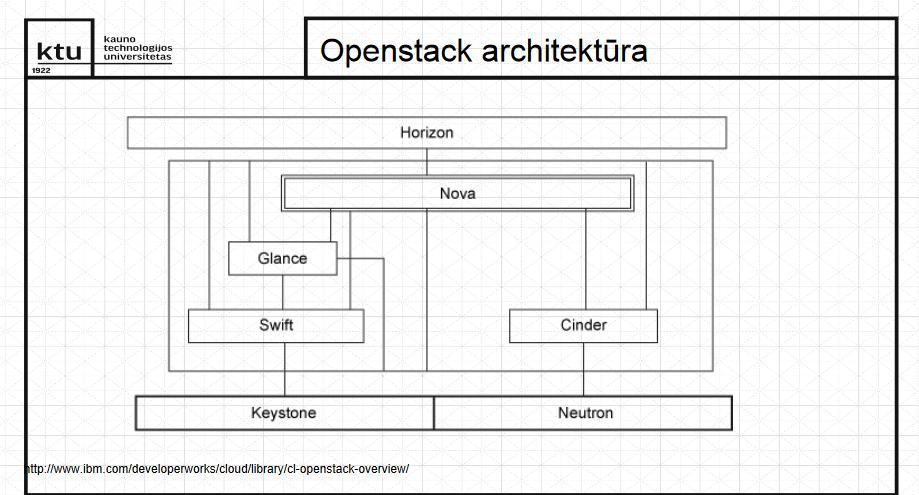
Serviso(Paslaugų) modeliai

* Software as a service (SaaS) - Vartotojas interneto pagalba gali naudotis konkrečiomis programomis (pvz., elektroniniu paštu, CRM, ERP ir kitomis).
* Platform as a service (PaaS) - vartotojui suteikia ne tik infrastruktūrinius išteklius, bet ir operacinę sistemą kartu su programomis, programavimo kalbomis, bibliotekomis ir kitais įrankiais bei paslaugomis.
* Infrastructure as a service (IaaS) - leidžia vartotojams naudotis serverių, duomenų saugyklų ištekliais bei tinklo įranga pagal poreikį.

Virtualizacijos būdai

* Skaičiavimo resursų vitrualizacija
* Duomenų saugyklos virtualizacija
* Tinklo lygio virtualizacija

1. Openstack architektūra



Keystone

* Autentikacijos ir teisių valdymo servisas. Visos API užklausos tarp komponentų yra vykdomo per Keystone, taip užtikrinant prieigos teisių valdymą

Swift(obejtku saugykla)

* Decentralizuota
* Automatiškai atsistatanti trikio atveju
* Plečiama horizontaliai iki daugelio petabaitų
* Aukšto našumo, gaunamo iš paskirstytos architektūros
* Gali būti sukurta naudojant nebrangią aparatūrinę
* įrangą

Glance

Virtualių mašinų atvaizdų saugykla. Atvaizdai gali būti

saugomi tiesiogiai jame, tačiau dažniau būna realizuota

kaip nuorodos į objektinę saugyklą.

Cinder

* Blokinė saugykla. Teikia tokias paslaugas:
* Virtualių diskų valdymas;
* Momentinių atvaizdų (snapshot) valdymas;
* Virtualių diskų prijungimas ar atjungimas nuo virtualių serverių;
* Virtualių diskų klonavimas;
* Virtualių diskų sukūrimas iš momentinių atvaizdų;
* Virtualių diskų kopijavimas į atvaizdus (images) ir atvaizdų
* kopijavimas į virtualius diskus
* Gali būti realizuota kaip interfeisas į blokinę saugyklą

Nova

* Nova-api: skaičiavimo resursų api interfeisas
* Nova-compute: valdo atskirą hipervizorių
* Nova-volume (deprecated)
* Nova-network (deprecated)
* Nova-scheduler: skirsto užduotis tarp skirtingų hipervisorių

Neutron

Tinklo servisas, palaikantis daugelį įskiepių, galinčių dirbti

su skirtinga tinklo įranga

Horizon - valdymo aplinka realizuota web

technologijomis

Užduočių eilė

Duomenų bazė

1. IP paketų perdavimas. IP paketo formatas.

**IP paketų perdavimas tinklu** apibūdinamas tokiomis charakteristikomis:

nesusijęs perdavimas – kiekvienas paketas perduodamas tinklu nepriklausomai nuo kitų;

negarantuotas perdavimas – paketas gali būti naikinamas nesiunčiant jokio pranešimo siuntėjo įrenginiui;

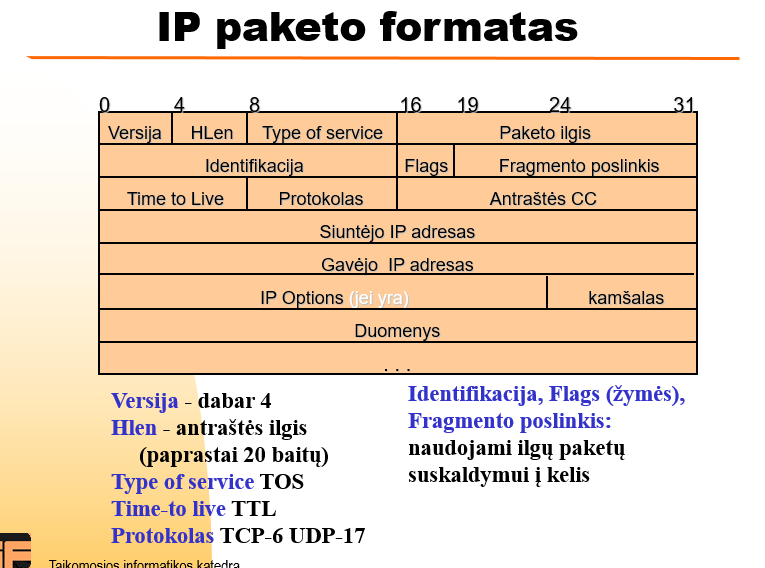
nepatikimas perdavimas – paketai gali dingti, būti sugadinti, pakartoti ar sumaišyti.

IP paketas perduodant gali būti suskaidytas į dalis (fragmentus). Fragmentai atsiranda, kai IP paketas netelpa į vieną kanalo sluoksnio kadrą. Vieną kartą suskaidyti paketai toliau keliauja kaip atskiri IP paketai. Fragmentus į pradinį paketą surenka gavėjo įranga. Tam reikalingi antroje antraštės eilutėje parodyti laukai.

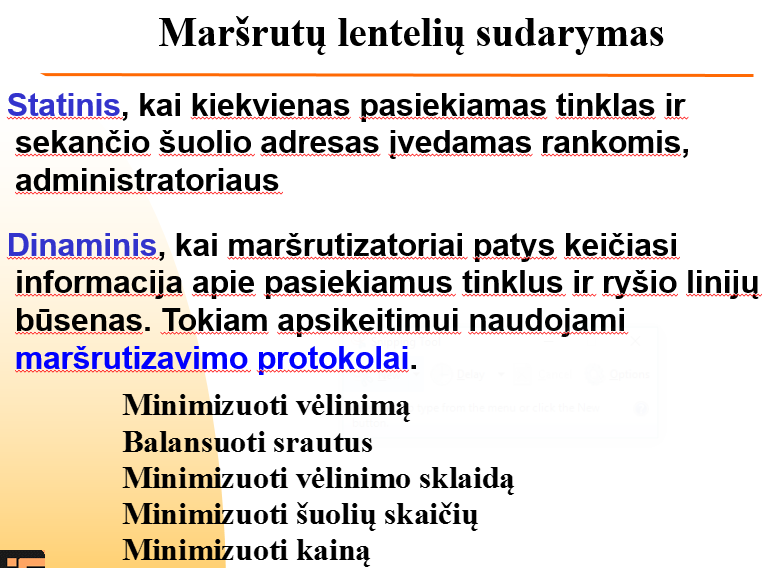
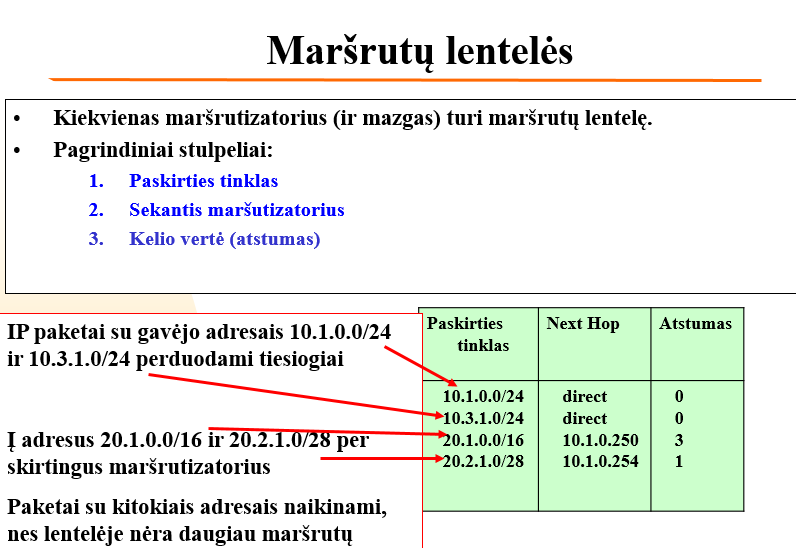
**IP paketo antraštė turi visą informaciją, kuri reikalinga pristatyti paketą į gavėjo kompiuterį:**

1. **Gavėjo IP adresą,**
2. **TTL,**
3. **TOS,**
4. **Siuntėjo adresą,**
5. **Fragmento identifikaciją**

Maršrutizatorius analizuoja kiekvieno paketo IP antraštę ir jį nukreipia gavėjo kryptimi

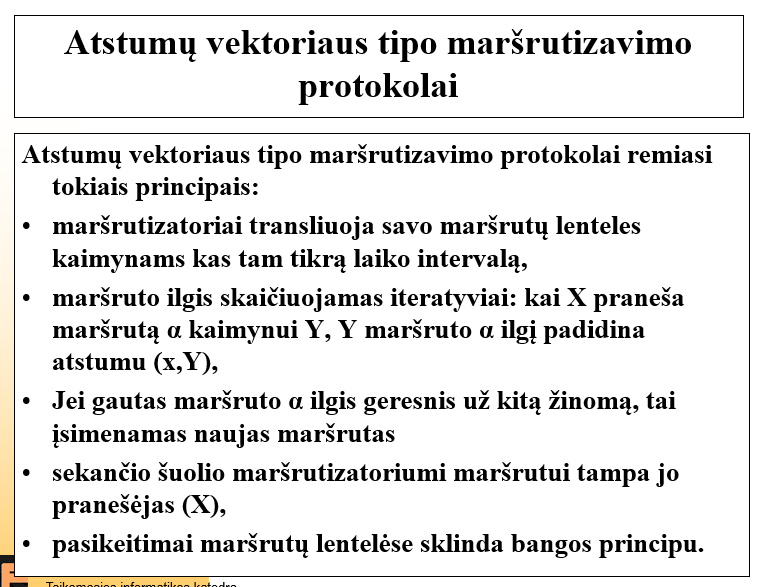
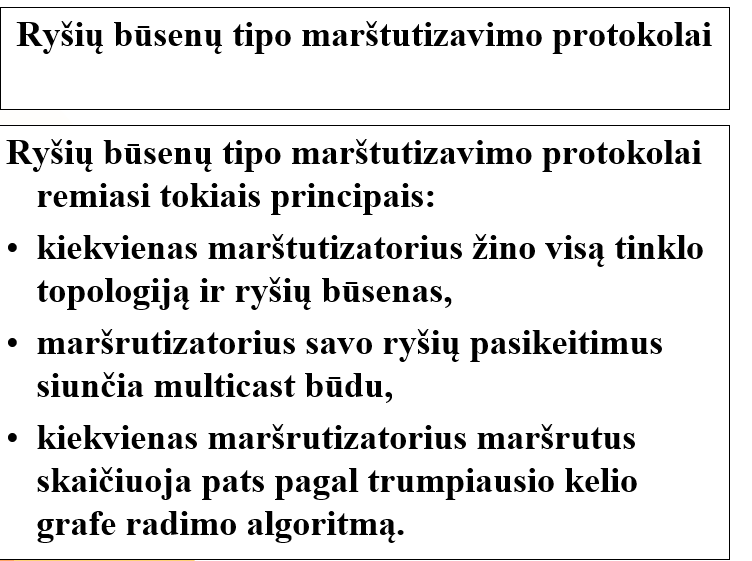


1. Maršrutų lentelės. Maršrutizavimo protokolai



**Maršrutizavimo protokolas aprašo:**

* - kaip pasiųsti maršrutų pasikeitimus,
* - marštutų pasikeitimus apibūdinačią informaciją ir jos formatus,
* - kada siųsti maršrutų pasikeitimus,
* - kaip surasti, kam turi būti siunčiami maršrutų pasikeitimai.



1. Interneto struktūra: autonominės sistemos

**Autonominė sistema (AS) tai centralizuotai ir**

**nepriklausomai nuo kitų administruojama**

**interneto tinklų dalis, turinti bendras**

**maršrutizavimo taisykles**

**\* IP numerių skirstymo sistema**

**\* Maršrutizavimo taisyklės viduje AS**

**\* Duomenų srautų valdymas**

**\* Tinklų skelbimas į kaimynines AS**

**\* Maršrutų į kitas AS (ir globalų internetą)**

**parinkimas**

**Tai stambiausias registruojamas Interneto darinys.**

**Paslaugų teikėjas paprastai turi vieną AS visiems savo ir savo klientų tinklams**

Kiekvienos AS valdymo savarankiškumas.

AS1 tinklai galės naudotis keliais per AS2 tik tokiu atveju jei:

- AS2 kraštinis maršrutizatizatorius skelbs kelius per AS2 į AS1

- AS1 kraštinis maršrutizatorius tuos skelbimus priims

Taip sumažinama apsikeitimų maršrutų lentelėmis apimtis: tik kraštinis vienos AS marštutizatorius keičiasi sutartais duomenimis su kraštiniu kitos AS marštutizatoriumi. Vidiniams maršrutizatoriams perduodama minimali reikalinga gautos informacijos dalis

**Autonominė sistema aprašoma parodant jos vietą kaimyninių AS aplinkoje:**

**- iš kokių AS ir kokius skelbimus priima (from xxx accept)**

**- kokioms AS perduoda savo skelbimus (to xxx announce)**

1. DHCP ir DNS funkcijos ir veikimo principai

DHCP - **Dynamic Host Configuration Protocol (DHCP), RFC 2131**

* + Paskiria laisvą IP adresą iš turimo segmento [laikinam naudojimui]
  + Išplėtimas: RFC **2132**

DHCP klientas gali gauti visus konfigūravimo parametrus ir užkrauti OS iš tinklo

Veikimo principas - **Du etapai:**

-surasti DHCP serverį [-ius], gauti pasiūlymus

-pasirinkti, pranešti kas pasirinkta, gauti papildomus

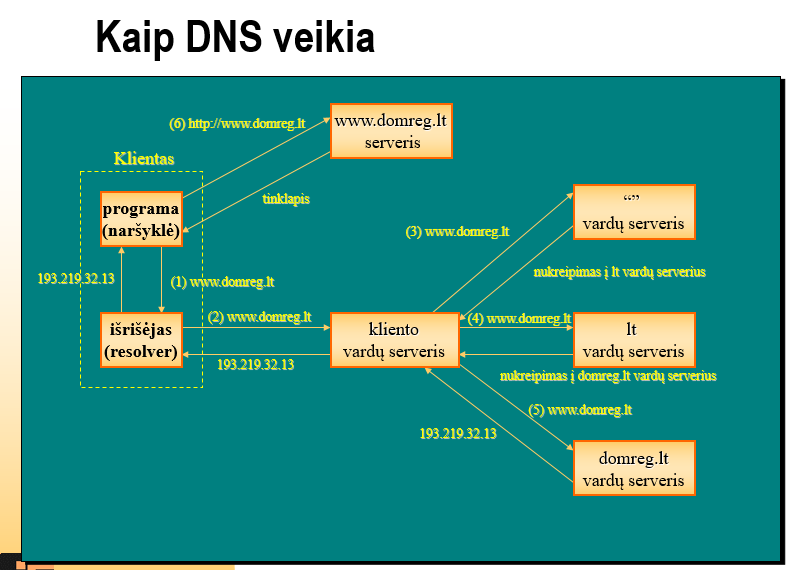
konfigūravimo parametrus, jei reikia.

DNS - DNS “verčia” interneto vardą į IP adresą

DNS - Interneto vardų sistema. Duomenų perdavime tinklu interneto vardai nenaudojami: jie skirti žmonėms, kad būtų lengviau įsiminti.

Transliacija “interneto vardas “ -> “IP adresas” nebūtinai turi būti vienareikšmė. Pasinaudojant tuo DNS gali atlikti papildomas funkcijas

Veikimo principas DNS –



1. Transporto sluoksnis. Klaidų taisymas ir spartos reguliavimas

**Transporto sluoksnis:**

* Aprašo duomenų mainus tarp tinklinių taikomųjų procesų
* Tinklo sluoksnis pristato duomenis į nurodytą tinklo mazgą. Išpakuotų duomenų srautą konkrečiam taikomajam procesui atiduoda transporto sluoksnis
* Tai paskutinis OSI modelio sluoksnis, kuriame numatytas duomenų perdavimo klaidų taisymas

**Transporto sluoksnio funkcijos-> priemonės**

* Keistis duomenimis tarp taikomųjų procesų -> prievadai (port)
* Taisyti perdavimo klaidas

-> patvirtinimai (ACK)

* Valdyti duomenų siuntimo spartą

-> siuntimo langas

**Klaidų valdymas –**

* Siuntėjas numeruoja siunčiamų duomenų porcijas ir kiekvienai iš jų per nustatytą laiką Δt turi gauti patvirtinimą ACK (Acknowlegment) iš gavėjo.
* Nesulaukus ACK per nustatytą laiką Δt, duomenų porcijos siuntimas kartojamas
* Kada siuntėjas siunčia paketą pakartotinai?
  + Jei per užduotą laiko intervalą negaunamas ACK, laikoma, kad paketas nepasiekė gavėjo arba paketas pasiekė gavėją sugadintas. Reikia kartoti siuntimą

**Spartos reguliavimo principas –**

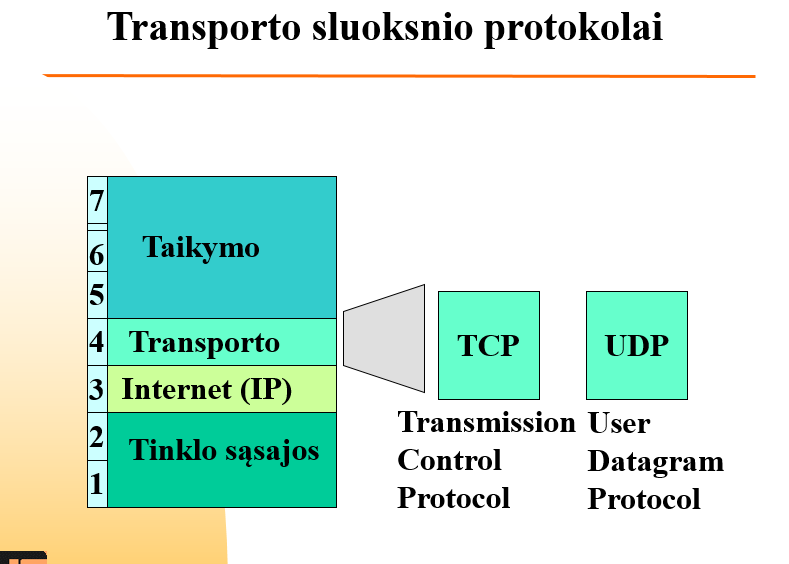
Nesulaukus per nustatytą laiką patvirtinimo apie 2 porcijos gavimą, siuntėjas kartoja siuntimą iš naujo nuo 2 porcijos, nors kelyje jau buvo 3,4 ir 5.

Bet dabar lango dydį sumažina per pusę

Kaitaliojant lango dydį priklausomai nuo tinklo būklės, tiksliau nuo prarandamų paketų kiekio, siuntimo sparta adaptuojama prie esamų sąlygų. Tas padeda tinkle išvengti lavininio perkrovų didėjimo, kuris būtų, jei siuntėjai pradėjus jų paketus naikinti ir toliau nemažintų siuntimo spartos.

Jei nesulaukiama išsiustų duomenų gavimo patvirtinimo, pakartotinai siunčiama mažiau duomenų -sparta mažėja, perkrova ar grūstis kažkuriame tinklo mazge išnyksta. Jei kelias iš eilės porcijas pasiuntėme sėkmingai, galime didinti siuntimo langą.

1. TCP ir UDP protokolai, savybės ir elementai



**TCP(Transmission Control Protocol) ir UDP(User Datagram Protocol) palyginimas**:



**Transporto lygio protokolai:**



UDP – Duomenų perdavimas tarp taikomųjų procesų be pristatymo garantijų. Taikomas kai

- taikomasis procesas negali laukti, kol kelyje prarasti duomenys bus perduoti pakartotinai

- arba taikomasis procesas pats rūpinasi duomenų siuntimo pakartojimu

- arba duomenų perdavimas vyksta rezervuotu kanalu, kuriame paketų praradimo praktiškai nėra

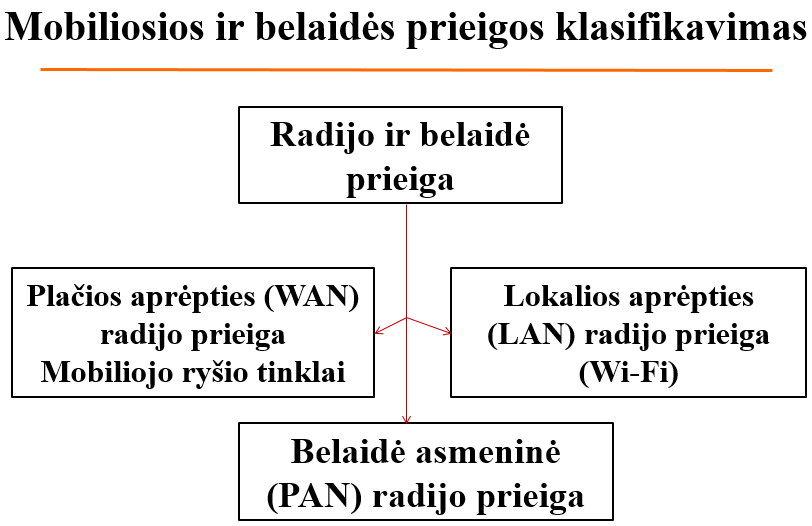
UDP paprastas, spartus, nereikia didelių resursų

Gali būti naudojamas multicast režime

* **TCP savybės –**
* Gali aptarnauti kelis sujungimus tuo pačiu portu
  + Reikia aiškių sujungimo įkūrimo ir užbaigimo procedūrų
* Potencialiai skirtingi RTT
  + Reikia adaptyvaus laukimo laiko nustatymo mechanizmo
* Potencialiai didelis vėlinimas ir didelė vėlinimo sklaida
  + Reikia sugebėti atpažinti vėluojančius paketus
* Potencialiai skirtingi gavėjo talpumai
  + Reikia reguliuoti siuntimą pagal gavėjo galimybes priimti duomenis
* Potencialiai skirtingi tinklo pralaidumai pagal gavėjus/pagal laiką
  + Reikia reaguoti į perkrovas tinkle

ELEMENTAI ???

1. Belaidės prieigos tinklai (WLAN)



**Prieigos taškas (access point)** -

Belaidei prieigai sudaryti būtinas prieigos taškas

Dažnai zonos persidengia

Zonai atskirti būtinas zonos SSID (Service Set Identifier) -

* unikalus 32 simbolių identifikatorius siunčiamas kiekvieno paketo antraštėje

1. Belaidžių tinklų sauga

**Wi-Fi saugos mechanizmų klasifikacija:**

* Atvira autentifikacija - Klientas siunčia paketą su įrenginio (AP) SSID

AP išsiunčia autentifikavimo atsakymo (patvirtinimo) paketą, jei gautasis SSID atitinka AP SSID.



* Bendrojo rakto autentifikacija - Tiek AP, tiek klientas turi naudoti tą patį iš anksto nustatytą (bendrąjį) raktą



* WEP –

Pagrindiniai principai:

* + Konfidencialumas
  + Prieinamumas
  + Vientisumas

Naudojami saugumo mechanizmai

* + RC4 šifravimo algoritmas ir XOR operacija
  + Prieigos taškas gali turėti 4 bendro naudojimo raktus (40 arba 104 bitų ilgio)
  + Naudojamas iniciacijos vektorius (IV)
    - Prie rakto pridedamas 24 bitų iniciacijos vektorius
    - Iniciacijos vektoriaus pagalba sukuriamas kintantis šifravimo raktas
* WPA - Du autentifikavimo metodai:

EAP (Extensible Authentication Protocol) – autentifikavimas 802.1x standartu (dažniausiai naudojamas įmonėse)

PSK (Pre-Shared Key) – iš anksto numatytas raktas: 256 bitų ilgio skaičius arba slaptažodis nuo 8 iki 63 simbolių (dažniausiai naudojamas namuose ir mažose įmonėse)

TKIP (Temporal Key Integrity Protocol) realizacija ir buvo pavadinta WPA.

* 802.1x – IEEE 802.1x sukurtas autentifikavimui pagal prievadus visiems IEEE 802 tinklams

Siūlo ne šifravimą, o skirtingus metodus:

EAP naudojamas autentifikavimui ir šifravimui belaidžiuose tinkluose

Autentifikavime dalyvauja trys objektai:

autentifikavimo serveris (RADIUS serveris)

autentifikatorius (AP ar maršrutizatorius)

prašytojas (klientas)

Belaidžio tinklo saugumo padidinimo taisyklės - Nepalikti atvirų prieigos taškų – naudoti šifravimo protokolus

* WEP protokolą pakeisti WPA protokolu
* Išjungti SSID transliavimą
* Išjungti DHCP paslaugą
* Jei belaidis tinklas nėra būtinas, jo atsisakyti ir naudotis laidiniu tinklu
* Naudoti papildomus aparatūrinius įrenginius (VPN maršrutizatoriai) belaidžio tinklo saugumui užtikrinti
* Naudoti papildomas programines priemonės saugumui sustiprinti: pvz. SSH, VPN tuneliai ir pan.

1. Judriojo ryšio technologijos