9. Безсърверно масово обслужване (p2p)

ВАСИЛ ГЕОРГИЕВ

v.georgiev@fmi.uni-sofia.bg

Безсърверно масово обслужване

- Приложения и модели
- p2p мрежи върху IP
  - Маршрутизация, откриване, отказоустойчивост
  - Надеждност, репутация, защита
- p2p данни: разпределени хеш-таблици
- Случаи

Особености

- Безсърверното масово обслужване (*тук* “БсМО”, ≡ p2p):
  - разгръща се като наслоена мрежа (overlay) върху съществуващата клиентска част от IT-инфраструктурата и поддържа нейерархичен модел на обмен върху йерархичния IP; без участие и използване на трафик в опорните мрежи (backbone) и техните възли (приложни и комуникационни сървери)
  - децентрализирана топология – равнопоставени процеси с универсална специализация – едновременно клиентски функции и частични сървърни функции (напр. [малка] част от данните за единичен пряк или групов и дори множествен достъп; маршрутизиране на съобщения между външни процеси и т.н.)
  - самоорганизация – без единен процес на управление, маршрутизация, регистрация на ресурси и данни; с динамични и вариращи параметри на ресурсите, на адресите на обекти; възможно с частична сървърна поддръжка за наблюдение и управление
  - инцидентна (ad hoc) и динамична свързаност и наличност на възлите и процесите в тях, но с изисквания за скалируемост
  - скалируемост (разширимост) – присъща поради отсъствието на централизиран топологичен елемент като сървера (тясно място)
  - отказоустойчивост (fault tolerance) – присъща поради множеството обслужващи ресурси, но с необходимост от репликиране на критичните данни и услуги
  - анонимност на споделянето – възлите като ресурс и свързаните процеси в тях се използват за транспорт и други системни функции (например репликиране на критични данни – „кеширане“) без конкретно оторизиране от администратора на възела или на процеса или на данните, но със свободно доброволно участие на потребителите
  - специална юридическа регулация

Типове

- Наслоени мрежи за БсМО:
  - неструктурирани – произволен граф на свързаност и производителност на комуникационните канали, спонтанно добавяне и отпадане на възли (типично за нововъзникнали БсМО)
  - структурирани – проектирани (а не спонтанни) характеристики:
    - топология – типично кръг, хорда (с „пръсти“ за вс. възел), дърво, верига;
    - комуникационна производителност;
    - ограничения в броя възли и тяхната дистанция (по време)
  - хибридни и йерархични структурирани – неструктурирана супермрежа от супервъзли, участващи в по 1+ структурирана наслоена мрежа
- Разпределени хеш-таблици (DHT) – за безсърверни данни

Случаи на приложение

Междумрежово свързване на групи процеси през защитни стени със собствен транспортен слой и собствена идентификация на процесите, ресурсите и обектите

1) Споделяне на файлове 1:1: заявките се разпространяват в инцидентна верига от процеси, а при установено наличност на файла, обменът се извършва *пряко* между двата възела – Napster, Gnutella, eDonkey.
2) Разпространяване на съдържание (1+):*: процесите пре-предават полученото от тях съдържание докато то се разпространи до всички заявители – BitTorrent.
3) p2p чат ("instant messaging"): текстови съобщения и поточни данни между клиентските процеси без сървърна поддръжка – Skype, MSN.
4) Разпределена обработка: декомпозиция на данните на сложна задача и независимата им обработка от група еднакви процеси (SPMD) – SETI@home.
5) Съвместна работна среда: форматни "събития" и други съобщения се обменят групово между процесите – Microsoft SharePoint Workspace (преди Groove), (защитèн чат и обмен в различни административни области на инцидентно формирана група от клиентски процеси с пре-предаване).
6) Развойни платформи за ИТ-услуги – JXTA, MS .Net: с поддръжка *и* на p2p-проекти за стандартни приложно-ориентиани услуги:
   a) Защита на достъпа и обмена на регистрираните [еднотипни, интерфейсни] процеси
   b) Директория (таблица с двойките <мрежов адрес, регистрирано име>) на процесите
   c) Директория на споделените обекти вкл. споделените файлове или имена на чат-профили – в същата таблица
   - Изисква устойчиво резидентно изпълнение на системните услуги-директории, в които преходните процеси да се регистрират (например поне една устойчива инстанция на услуга-директория за всеки интранет)
   - т.е. оторизацията, търсенето и откриването е в централизирана мрежа от резидентни услуги, а обменът е пряк между регистрираните преходни процеси

Структурирани мрежи за p2p – фиг. 9.6

- Сървърни (централизирани) p2p – Napster за споделяне на файлове: поцесите се регистрират и публикуват данни за споделяните файлове на сървърно базирана директория на съдържанието, от която могат да зареждат адреса на даден обект, а обменът му се извършва безсървърно (пряко)
- Безсървърни p2p – Gnutella за споделяне на файлове: процесите поддържат само директория на локалните споделени файлове, а заявките за търсене се препредават от процеса-заявител към съседите му наводняващо (flooding, към всички съседи на съседи)
- Йерархични p2p – Skype за p2p чат: 2 нива – процеси и суперпроцеси; суперпроцесите формират неструктрирана инцидентна мрежи и наводняването на заявките за откриване [на текущия адрес] на търсен чат-процес се извършва само между суперпроцесите, с което значително се увеличава скалируемостта, тъй като разпространението на заявки се ускорява, а по-точно времето за разпространение T нараства само логаритмично с броя на процесите p

$$T = \log_n p,$$

n = средна топологична мощност на супервъзлите; p=общ брой процеси

p2p протоколи за файлово споделяне

протокол	приложения	описание
BitTorrent	споделяне на файлове[1], BitTorrent и др., вграден	предимно Windows клиенти, ползва сървер директория – Tracker, допуска криптиране и проследяване на IP-адрес
eDonkey	споделяне[2], eMule, FlashGet и др., вграден	център. сърверна директория ed2k и p2p съдържание, осн. Win клиенти, контрол на съотношение на споделяне
Gnutella	споделяне[3], за Mac, Win, JVM, Symbian	пълен p2p – не подлежи на закриване като Napster, статистика на възлите и на скоростта на обмен, набор протоколи: ping/pong, query/hit, push (за стени)
Kad Network	споделяне[4], eMule, eDonkey и др., вграден	разширение на ed2k – поддържа децентр. хеширан списък ключове на файловите имена, преодоляване на стени

BitTorrent протокол

- Протокол, технология и среда за споделяне на файлове в инцидентни (ad hoc) p2p мрежи
- 1/3 от Интернет трафика, но само 1/5 от високоскоростния трафик (понеже се поддържа основно от потребителски мрежи и устройства)
- базира се на споделяне на дискови, компютърни, комуникационни ресурси с минимална сървърна поддръжка
- BT-клиента:
  - поддържа списък на локални работни копия на споделените файлове и прозрачно обслужва заявки (seeder)
  - генерира последователност от заявки за зареждане на файл
    - заявките са TCP-формат – към множество възли поддържащи отделни части от файла (при web-браузърите единична HTTP GET заявка към един сървер – дори и при реплики на файла) – по-сложно управление (signaling), но потенциално уплътнен трафик
    - негарантирана скорост и ред на зареждане – неподходящ за изпреварващо зареждане на поточни MM данни (progressive streaming) – текущи разработки
    - планът на заявките е rarest-first
  - зарежда метаданни за заявен файл – torrent – списък с текущите клиенти, поддържащи копия на части от файла и адрес на сървера, координиращ процеса (tracker)
- BT-сървер – tracker: директория с мета-данни (торенти); контролира коефициента на споделяне на клиентите (за избягване на leeching „пиявици“ – егоистично използване)

Skype протокол

- тук MU = 1500 млн. регистрирани и 240 млн. дневно
- прилага хибриден модел топология, като различава 3 категории процеси („възли“) – възел, супервъзел и сървер на профилите – фиг. 9.9
- Супервъзлите се подбират да имат физическо IP, подходяща локална и мрежова производителност и по режим на работа прозрачно за потребителя/администратора
- за мултимедиен обмен (звук и/ли видео) възелът се свързва със супервъзел
- всеки възел (и супер-) има около 200 съседи с няколко супер („host cache”: IP-тата и скайп-портовете), като доставчикът на услугата е разгърнал и подходящ брой супервъзли в националната мрежа
- при начално инсталиране/свързване започва попълването на хост-кеша; наличността на профилите и заявките за откриване се съхраняват в супер-, а запазените профили – локално
- без защитна стена около един от комуникиращите профили (2+) съобщенията и сигналинга са TCP, а потоците – през UDP (ако маршрутизират през защитна среда – и потоците са върху TCP)

Хетерогенност и скалируемост на p2p мрежите

payload

signaling **- (комуникационен свръхтовар)**

- хетерогенността се изразява в разликите в производителност на равнопоставените възли, платформа (ОС) и мрежова скорост – решават се със средствата на йерархичното разслояване – напр. като при скайп-протокола
- изисква се независимост от броя свързани процеси: повече възли увеличават сигналинга логаритмично, но също добавят нови ресурси
- ако времето за разпространяване не е логаритмична а линейна функция на *p* – обслужването се счита за нескалируемо
- маршрутизацията, основана на наводняване, не може да работи с логаритмичен сигналинг, ако не се йерархизира
- йерархизираното наводняване работи само ако
  - възлите са директно свързани със супервъзли и
  - наводняването се разпространява само между супервъзлите

Ефективност на търсенето и откриването

- обектите данни и техните списъци са разпределени между възлите и търсенето/откриването им предхожда достъпа до тях
- търсенето е сляпо или информирано
- при сляпо търсене възелът или потвърждава наличието на локален обект, или препраща заявката наводняващо до всички свои съседи, или я препраща до произволна част от съседите си без критерий за селекцията им (👎 голям сигналинг)
- при информираното търсене възлите поддържат локално и списъците на съседите си през *r* стъпки (hops или „пръсти“) в топологичния граф, така че препращането не е наводняващо а фокусирано (👎 голям свръхтовар от синхронизацията на външните списъци)

Локалност на данните
Мрежово съседство (proximity)
Ефективно търсене/откриване

- локалност на данните е разполагане на данни с еднакви или близки стойности на даден атрибут на съседни или близки възли – редуцира сигналинга при търсене чрез наводняване на съседите
- мрежово съседство е изискването съседите в наслоената p2p мрежа да са „близки“ и в базовата клиентска (не опорна) IP-мрежа – също редуцира сигналинга – фиг. 9.12
  - близост в намаляващ ред е алтернативно принадлежността на
    1) един интранет
    2) една административна област
    3) общ защитен протокол и/ли маршрутизатор
- ефективно търсене в p2p мрежите и също отказоустойчивост се постига типично с репликиране на данните: само отказоустойчивост – 2 копия, а ефективност – множество съседни или несъседни копия
  - обаче синхронизация на копията!

Анонимност

- анонимността в p2p се разглежда като скриване на
  - инициативния процес/потребител;
  - реактивния процес/потребител;
  - двата;
  - всички (групов).

едно-посочна

дву-посочна

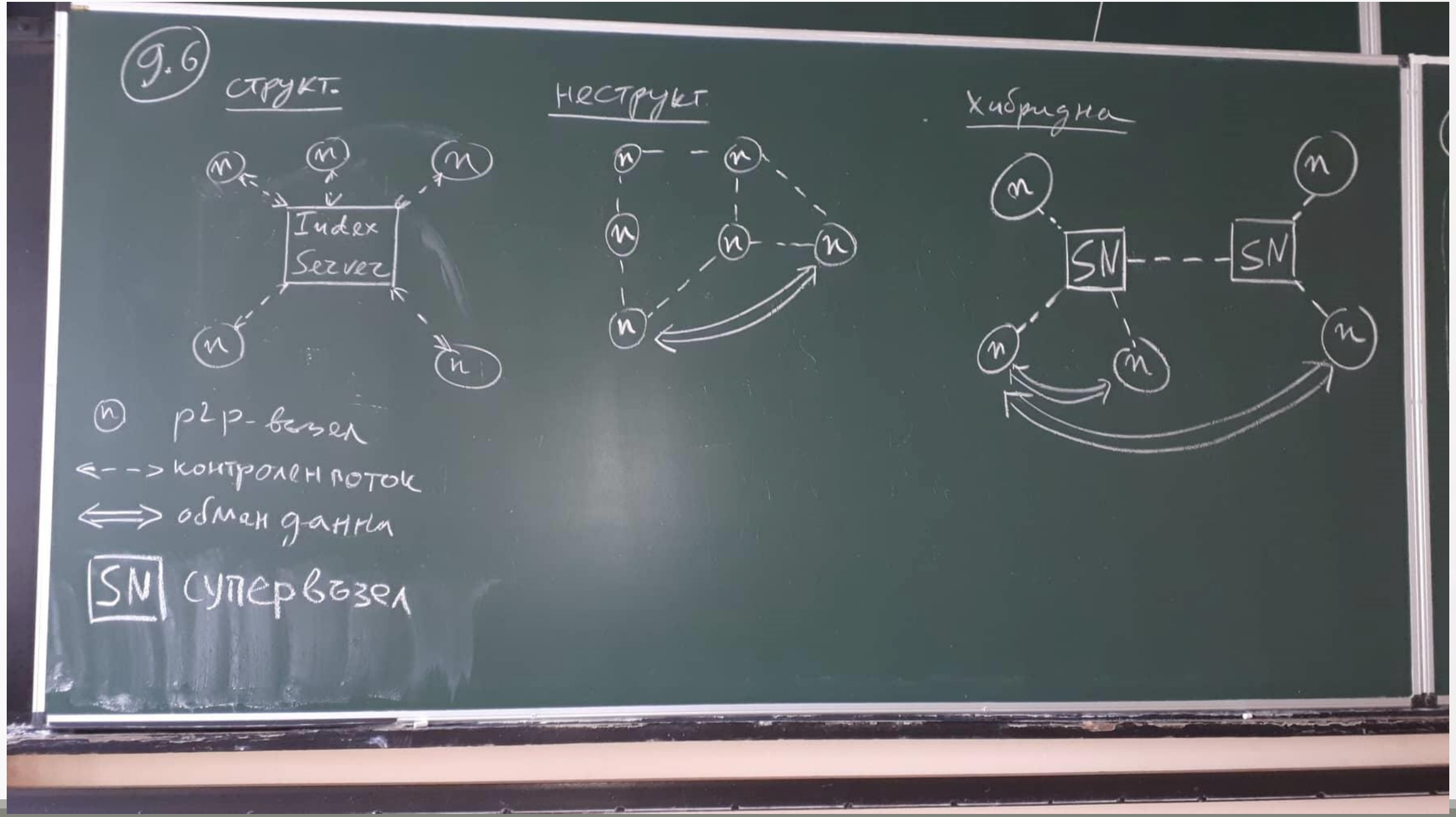
- защитена („лукова“, *onion*) маршрутизация за еднопосочна анонимност – фиг. 9.13:
  - инициативният процес има достъп до съдържанието и адресите на междинните процеси по маршрута вкл. крайния процес, като съдържанието се кодира обикновено с прилагане на публичен или публичен и частен ключ – т.е. симетрично или асиметрично кодиране
  - междинните процеси („лукови маршрутизатори“), както и крайният процес знаят само адресите на своя предшественик и своя наследник
- двупосочна анонимност може да се постигне само с използване на допълнителни доверени процеси-агенти, които поддържат адресната таблица на цялата група

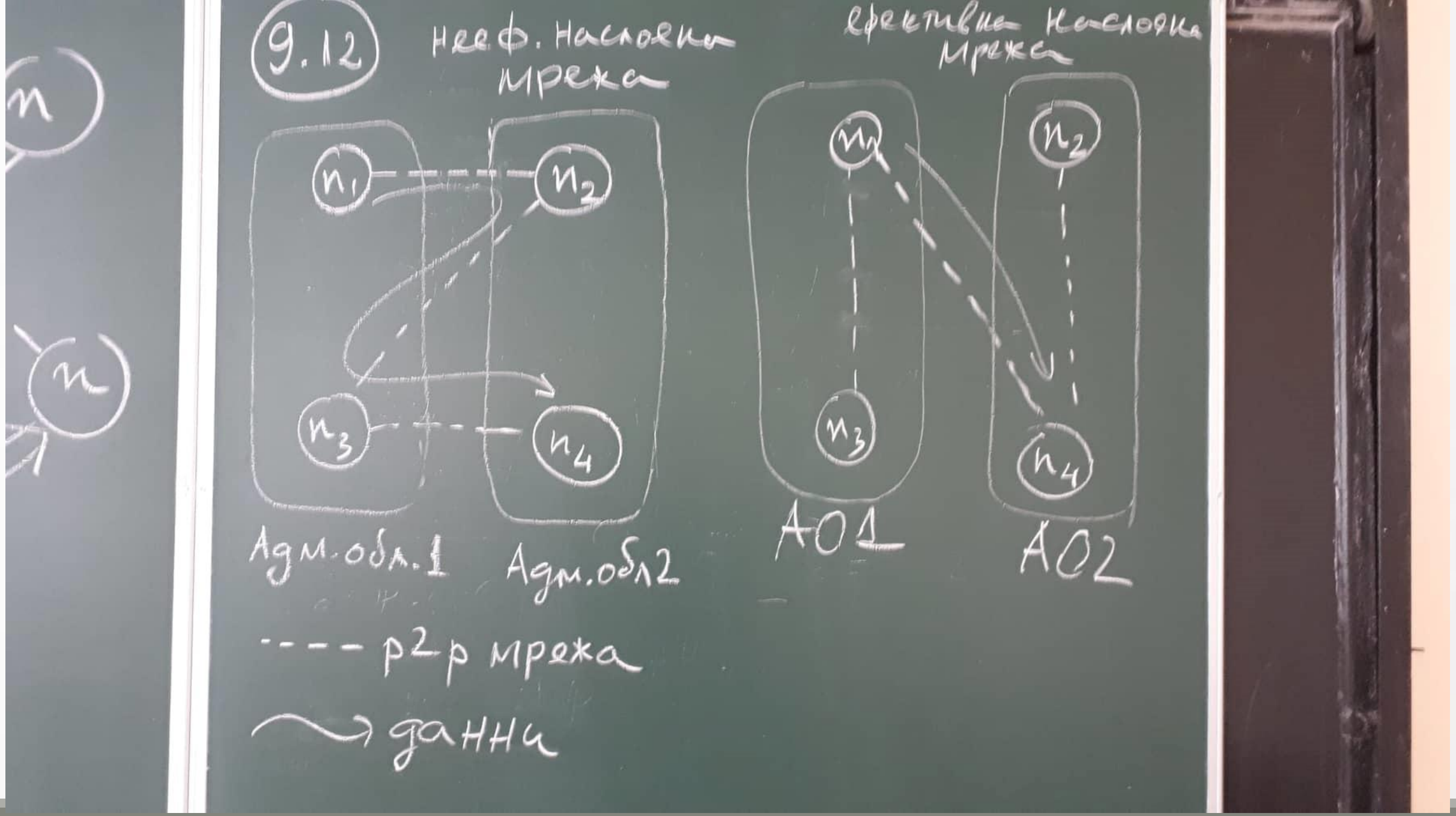
Доверителност, репутация, атаки, егоистично използване

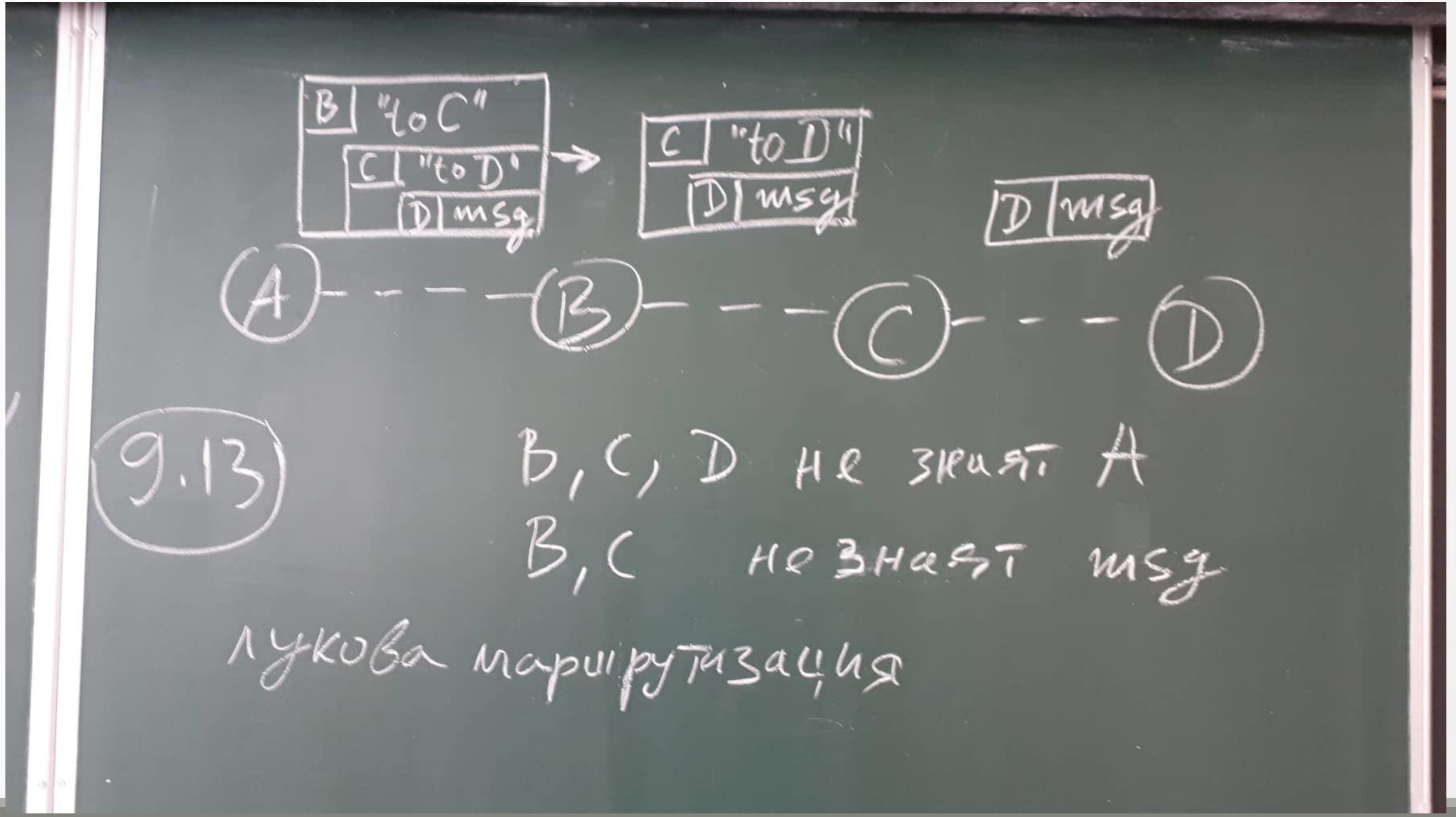
- Анонимността поражда проблеми с доверителността и уязвимостта към наводняващи атаки (разпределен отказ от обслужване, DDoS – фиг. 9.14), които деградират QoS
- по принцип средство срещу наводняваща атака от възли/процеси е регистриране на тяхната репутация като добра или застрашителна (malicious)
- ако регистрирането на репутацията е разпределено между възлите-съседи, то е уязвимо именно от застрашаващите възли, затова също се използват специализирани процеси-регистри
- егоистичното използване (leeching, free-riding) подобно на репутацията на възлите се измерва и коефициент на използване/обслужване

p2p данни: разпределени хеш-таблици

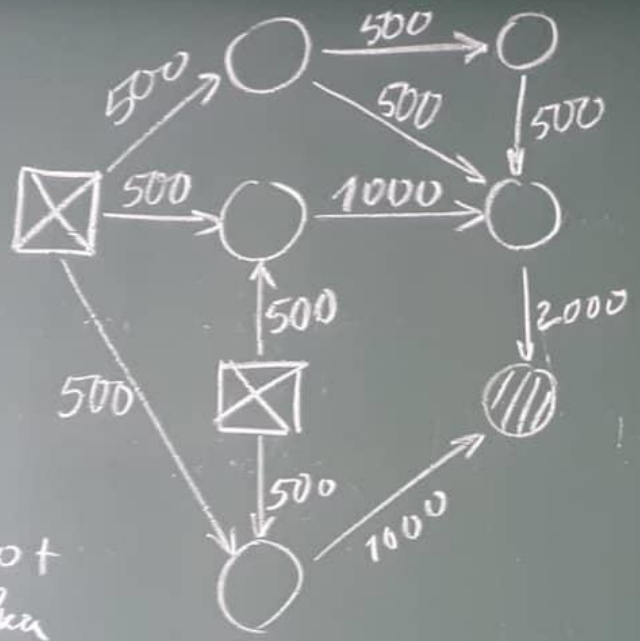
- DHT (фиг. 9.15) е не-p2p услуга на междинния слой (платформата), която разпределя двойката <ключ, стойност> в адресно пространство с мощност 2^{Lkey}, напр. за 64-битов ключ пространството е с 2^{64} позиции за обекти (до 2^{256})
- DHT
  - разпределя адресното пространство между възлите и генерира адресът на съответния възел по зададен ключ
  - наблюдава наличността на възлите и репликира таблицата от проблемните възли
- за да се скалира с тези функции, DHT е структурен (т.е. планиран и проектиран) елемент на p2p мрежата
- участващите възли изпълняват примитивите put(key, data) и get(key)
- търсенето се осъществява чрез наводняване и хорда-топология между съседите (а не през процеса DHT, който само асоциира съответния ключ с даден новоприсъединен възел, и ги наблюдава)
- DHT е основната алтернатива за p2p данни и е централен компонент на BitTorrent Bitcoin


9.6
структ.
Index Server
неструкт.
хибридна
SN
SN
p2p-възел
контролен поток
обмен данни
SN супервъзел


9.12
Нееф. Наслояна мрежа
ефективна наслояна мрежа
n1
n2
n3
n4
Адм.обл.1
Адм.обл2
АО1
АО2
---- p2p мрежа
данни


B "to C"
C "to D"
D msg
C "to D"
D msg
D msg
A
B
C
D
9.13
B, C, D не знаят A
B, C не знаят msg
лукова маршрутизация

9.14



Входен възел

възел

1000 → следствот + бр. заявки в минута

потопен възел

9.15