

Лекция 12

2. Децентрализованные сети.

Сети, устойчивые к разрушению,

Инженеры по заказу военных предпринимают упорные попытки создать новые сетевые протоколы, которые будут гарантировать устойчивую работу сети в самых сложных условиях. Например, когда каждый радиоузел активен в течение всего лишь 20% времени, а конфигурация беспроводной сети постоянно изменяется. Такую сеть практически невозможно уничтожить.

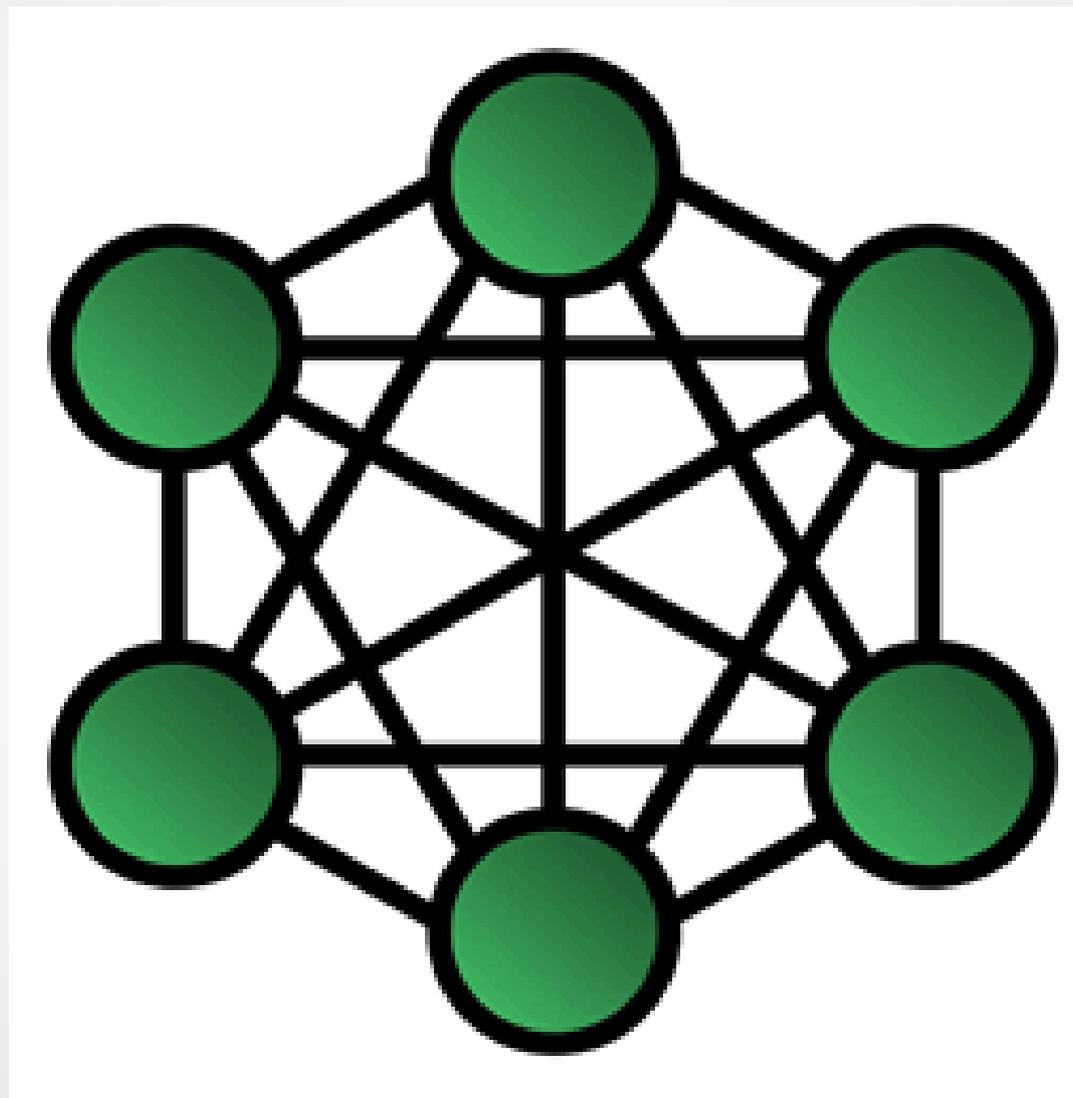
В частности в журнале NetworkWorld была опубликована статья с описанием различных проектов в области **DTN (disruption-tolerant networks или delay-tolerant networks)**, - беспроводных сетей, устойчивых к разрушению и задержкам.

MESH-СЕТЬ

Это **ячеистые сетевые топологии, состоящие из полностью равноправных узлов** и поддерживающие возможность построения произвольных маршрутов. Если одна нода (узел сети) откажется принимать информацию, ее примет другая. Все это дает очень высокую надежность и невозможность заблокировать распространение информации.

Меш-сети **могут как быть изолированными от интернета, так и работать в режиме оверлея.**

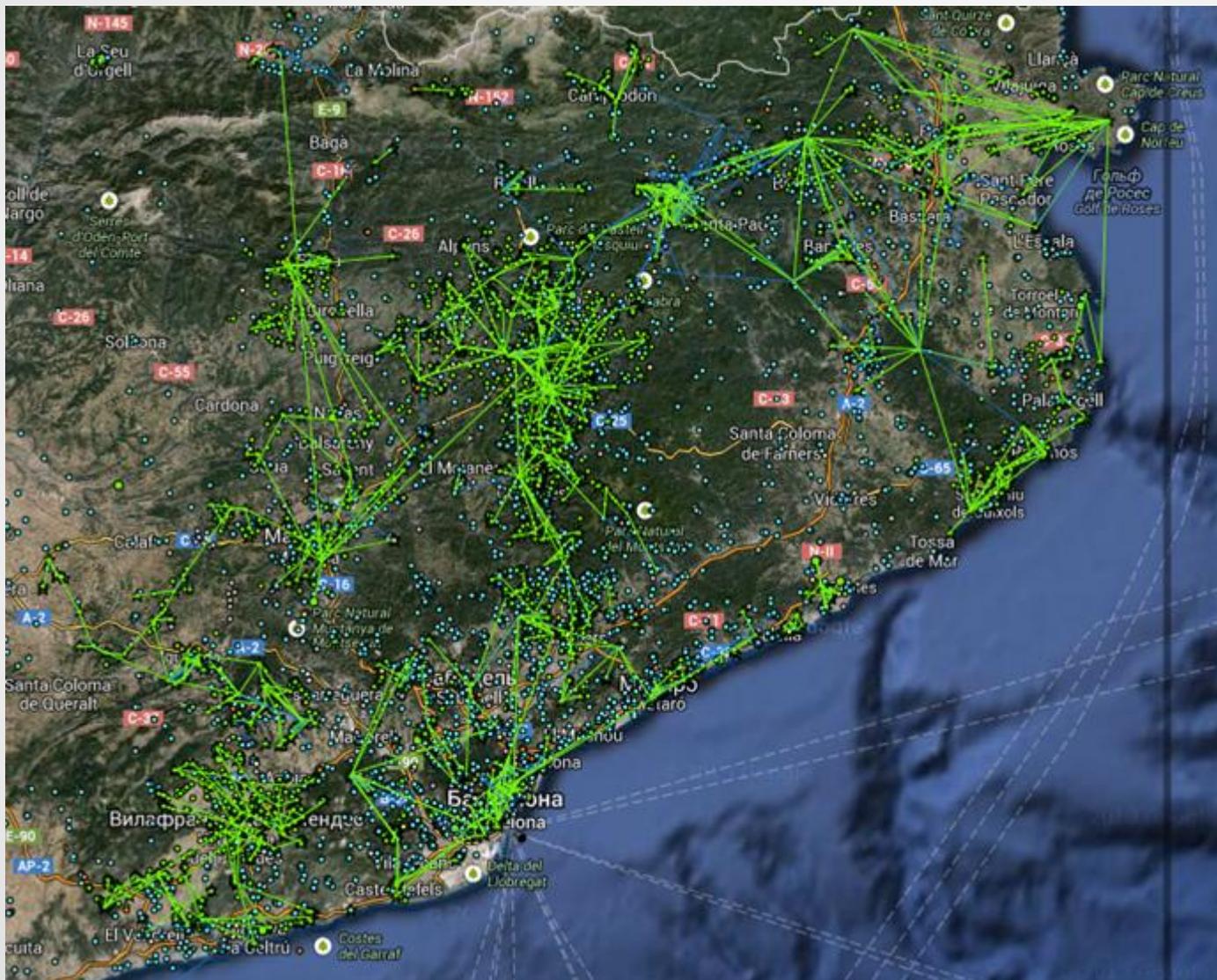
MESH-СЕТЬ



Guifi

- Самая крупная меш-сеть в мире развернута в Каталонии и Валенсии (регионы в Испании) и состоит из 22 тысяч узлов.
- Проект зародился в начале нулевых, когда местные жители устали ждать появления в регионе нормального интернет-провайдера. С тех пор сеть развивается на общественных началах и подключение к Guifi доступно совершенно бесплатно.
- Есть условные единицы — «острова». Каждый остров — это меш-сеть, объединяющая пользователей района, муниципалитета или города. Для подключения используются Wi-Fi-роутеры с видоизмененной прошивкой на базе DD-WRT. Связь между островами обеспечивается при помощи VPN-серверов или прокси-серверов на Squid. Эти же серверы дают пользователям Guifi доступ в интернет.

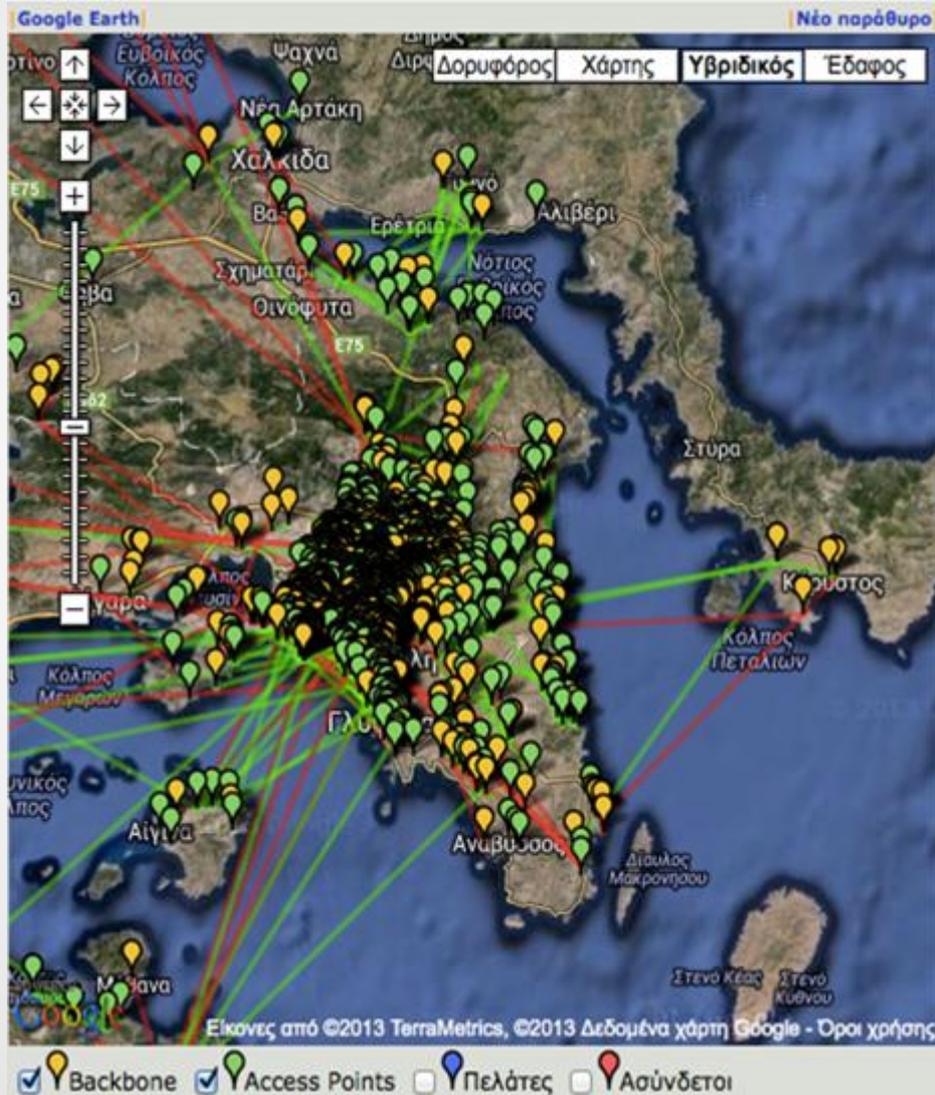
Guifi



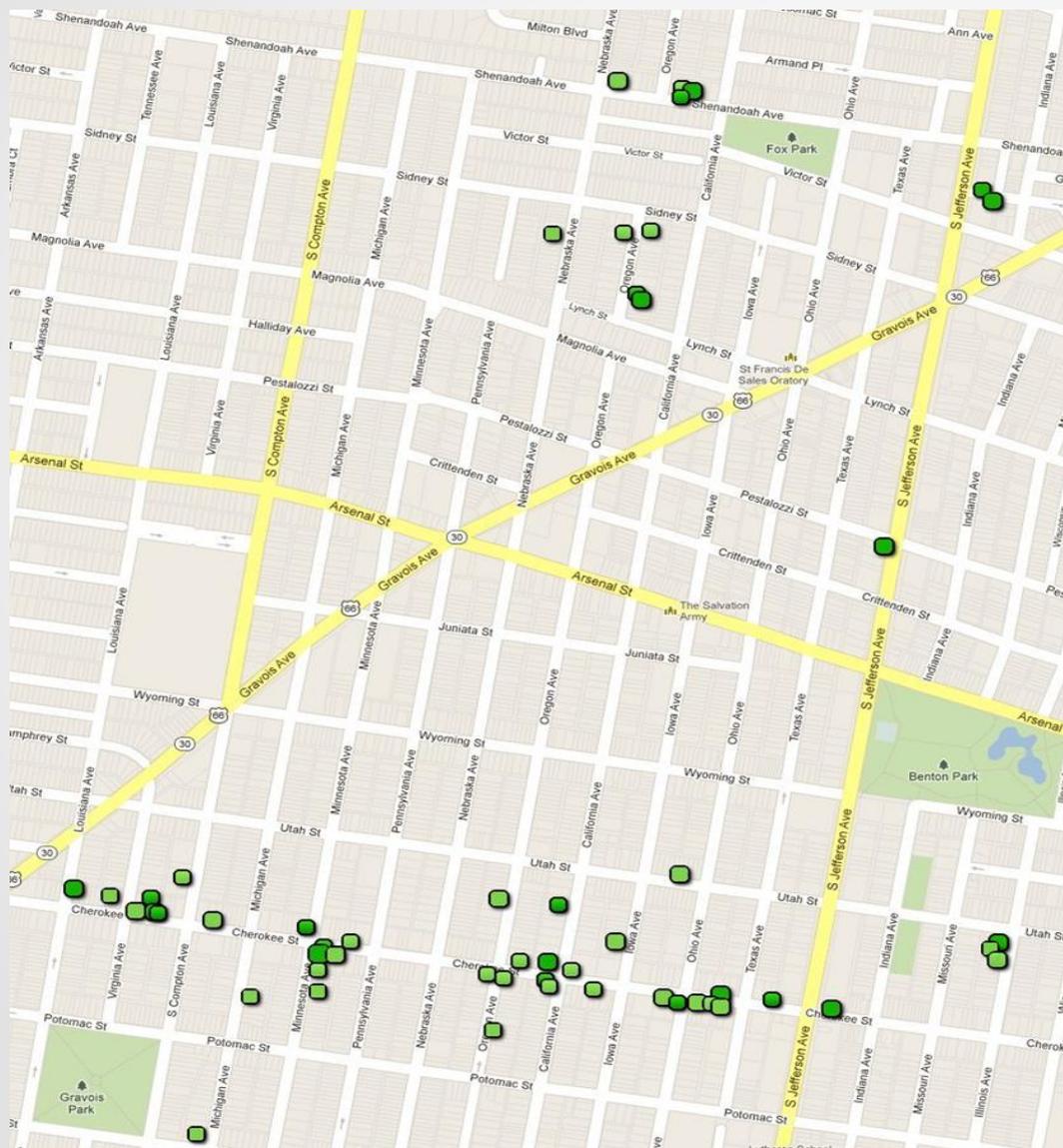
15%
максимальная доля
рынка,
которую меш-сетям удастся
отнять у традиционных
интернет-провайдеров,
по мнению основателя Guifi

Рамона Рока ●

Athens Wireless Metropolitan Network



WasabiNet



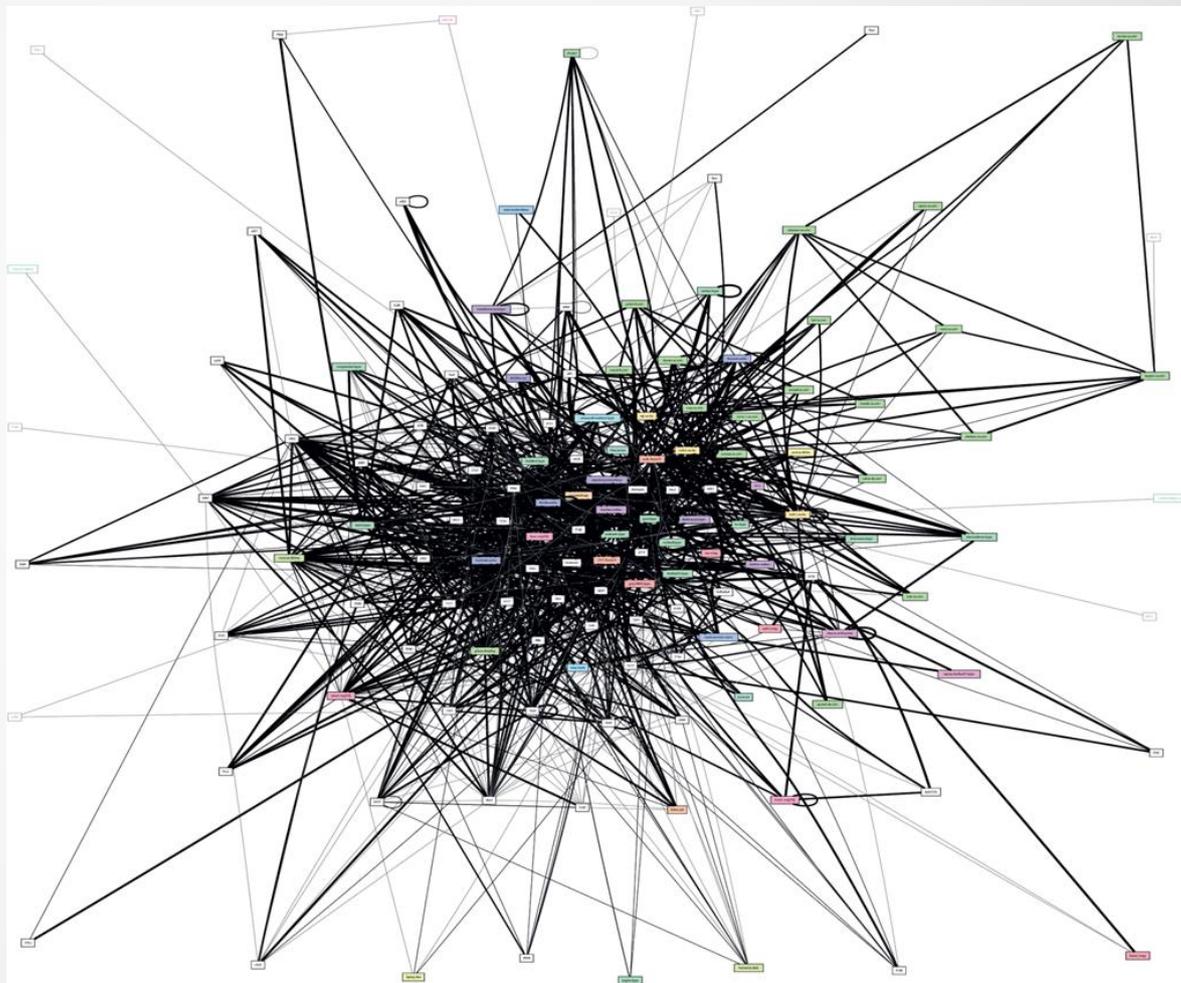
Районная сеть в Сент-Луисе (США). Покрывает территорию вокруг одной улицы (Cherokee Street).

Базовый интернет (до одного мегабита в секунду) предоставляется бесплатно, а более быстрый (3–5 Мбит) — за 10 долларов в месяц, а юридическим лицам услуги предоставляются за 20 долларов в месяц.

Для маршрутизации используется протокол OLSR. Аналогичные сети существуют и во многих других городах и штатах.

Project Meshnet

Проект зародился в недрах Reddit в 2011 году на фоне разговоров о SOPA/PIPA. Участники разработали протокол sjdns, важное отличие которого от других заключается, во-первых, в возможности объединять отдельные сети между собой, а во-вторых, в шифровании трафика, однако в текущем виде sjdns не дает анонимности. Сейчас можно проследить маршрут, по которому идут пакеты, и найти адрес пользователя..



Hyperborea

<https://xakep.ru/2014/09/05/mesh-networks/>

В Hyperborea уже есть несколько популярных сервисов:

HypeOverflow — клон StackOverflow

Uppit — аналог Reddit

Urlcloud — файл-хостинг

neoretro — NTP-сервер

rows.io — Jabber-сервер

Hypediscuss — General Forums

Social Node — микроблоги

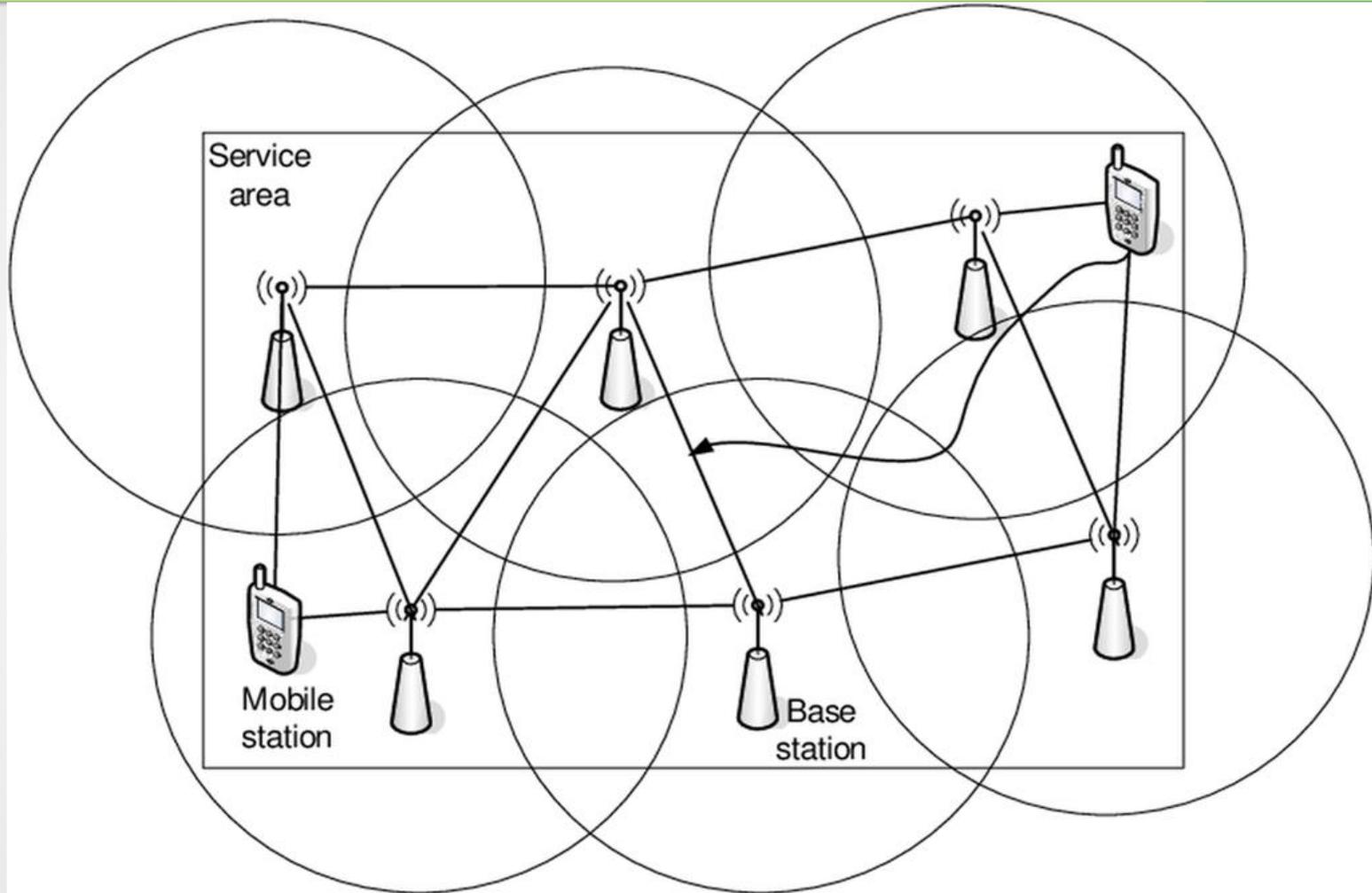


Netsukuku

Это проект реализации отказоустойчивой, распределенной, самоорганизующейся сети, построенной на базе существующих сетевых технологий, таких как TCP/IP и WiFi. Сам софт крайне нетребователен к ресурсам и может выполняться даже на embedded устройствах, вроде точек доступа и SOHO маршрутизаторов.

- Основной особенностью данной технологии является то, что она позволяет строить меш сети с динамической маршрутизацией размером до 2^{128} узлов (!).
 - Основой всей сети Netsukuku является узел.
 - Узел — это сетевое устройство (точка доступа либо ПК пользователя) с выполняющимся на нем софтом, обеспечивающим маршрутизацию.
 - Все узлы равноправны между собой, нет никаких различий между конечным узлом пользователя и маршрутизатором, объединяющим несколько смежных подсетей.
 - Сама сеть подразумевает использование беспроводных технологий, как наиболее удобных в плане масштабирования

Netsukuku



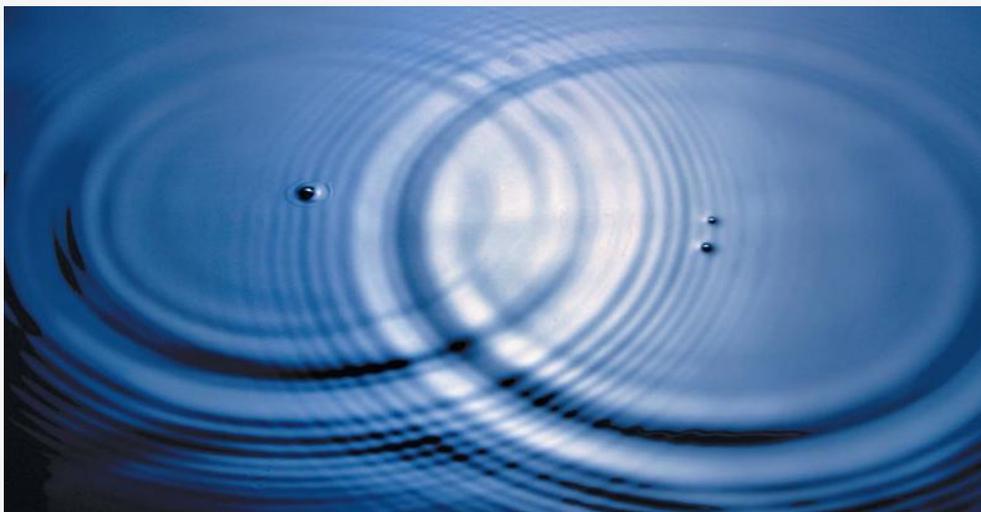
Netsukuku

Сеть Netsukuku строится иерархическим образом: каждые 256 узлов объединяются в т. н. групповой узел (gnode); 256 групповых узлов составляют уже группу более высокого порядка (ggnode) и так далее. Поскольку каждая групповая нода является таким же полноправным узлом сети, протокол **QSPN** может работать одинаково на всех уровнях иерархии. При этом, при поиске маршрута, в каждом случае он оперирует максимум 256 узлами, что делает сам поиск очень легким.

Основой протокола **QSPN** является **трейс пакет (TP)**. Это пакет, который **содержит в себе идентификаторы узлов, через которые он прошел.** Этот пакет не отправляется кому-то конкретно. Вместо этого, устраивается **натуральный трейс флуд.**

Netsukuku

Если бросить камень в бассейн с водой, то можно наблюдать следующее: волны начинают распространяться из начальной точки, причём каждая волна рождает новую волну, которая продолжает распространяться и рождать все новые и новые волны. Когда волна ударяется о края бассейна или о какую-то преграду, она отражается и начинает распространяться в обратную сторону.



В применении к `qspn` камень — это `qspn_starter`, бассейн — `gnode`, а каждая волна — `tracer_pkt`. Каждая новая волна несёт с собой информацию о родившей её волне. Когда `tracer_pkt` (волна) достигает `extreme_node` (препятствия или границы бассейна), рождается `qspn_open` (отражённая волна).

Netsukuku

- Как только пользователь запустил на своем устройстве демон Netsukuku, его узел начинает кидать широковещательные пакеты с целью обнаружить соседей. И как только сосед обнаружен, происходит обмен маршрутной информацией, назначение адреса и т.п.
- За сессию, трейс пакет проходит каждую ноду только единожды. Приняв TR, нода отправляет его всем своим соседям, дописывая себя в него.
- Каждая нода, принявшая TR получает полную информацию о маршруте до узла отправителя, а так же до каждого из узлов-посредников. Поскольку изначально узел А отправляет несколько TR (каждому из своих непосредственных соседей), то в каждый момент времени, в сети существует несколько версий TR, относящихся к одному и тому же флуду, называемых «букетом».
- Произвольный узел X, приняв первый TR от узла А и заглянув внутрь пакета — внезапно получает кратчайший маршрут с минимальным RTT до узла А, а так же всех узлов цепочки. Последующие пришедшие пакеты будут уже альтернативными маршрутами, соответственно, более длинными. Таким образом, маршрутная информация собирается автоматически, причем на основании реальной топологии сети и задержек.

Netsukuku

Адрес узла — величина переменная, следовательно привязывать имя напрямую к IP нельзя.

Все узлы имеют равные права на получение имени. Единственное ограничение связано с количеством имен, ассоциированных с отдельным узлом (256). В остальном — пользователь волен выбирать любое имя длиной до 500 символов.

Служба ANDNA построена по распределенному принципу, не существует единого хранилища данных. Информация «размазана» по всем узлам сети; многократное дублирование обеспечивает надежность и отказоустойчивость ее работы (объем базы на каждом узле даже в худшем случае не должен превышать 300 Кб).

Пусть некоторый узел X хочет зарегистрировать имя «habr.ru».

Для этого, он **вычисляет хеш от имени: $h(\text{«habr.ru»}) = 0x11223344$** . Числовое представление хеша **это адрес узла**, который будет **считаться хранилищем - хеш нода (узел H)**. Далее, хеш нода **вносит ассоциацию имени и адреса в свою базу данных**. После этого, узел X считается владельцем имени и просто так его не потеряет.

Любой узел, желающий узнать адрес узла с именем «habr.ru», точно так же вычисляет хеш и дает запрос к узлу H (а точнее, к случайно выбранному соседу из той же группы).

По умолчанию, **хеш нода будет хранить информацию об имени в течение 30 дней после регистрации.**

Netsukuku

При регистрации нового имени, узел-инициатор генерирует пару криптографических ключей, а затем открытый ключ передает на домен среди прочей регистрационной информации. Этот ключ будет сохраняться хеш нодой все время существования данного имени; любые действия владельца должны быть подтверждены соответствующей цифровой подписью.

Формально, узел может иметь сколько угодно имен. Однако, из практических соображений, вводится ограничение на количество имен, равное 256.

Помимо хеш ноды, в регистрации так же участвует и т. н. counter node — узел счетчик. Этот узел хранит количество имен, ассоциированных с исходным узлом.

При получении запроса на регистрацию, хеш нода N вычисляет хеш от открытого ключа узла X , который в свою очередь уже трактуется как адрес узла счетчика C . Хеш нода дает запрос к счетчику, с тем чтобы выяснить, сколько имен в настоящий момент ассоциировано с регистрирующимся узлом.