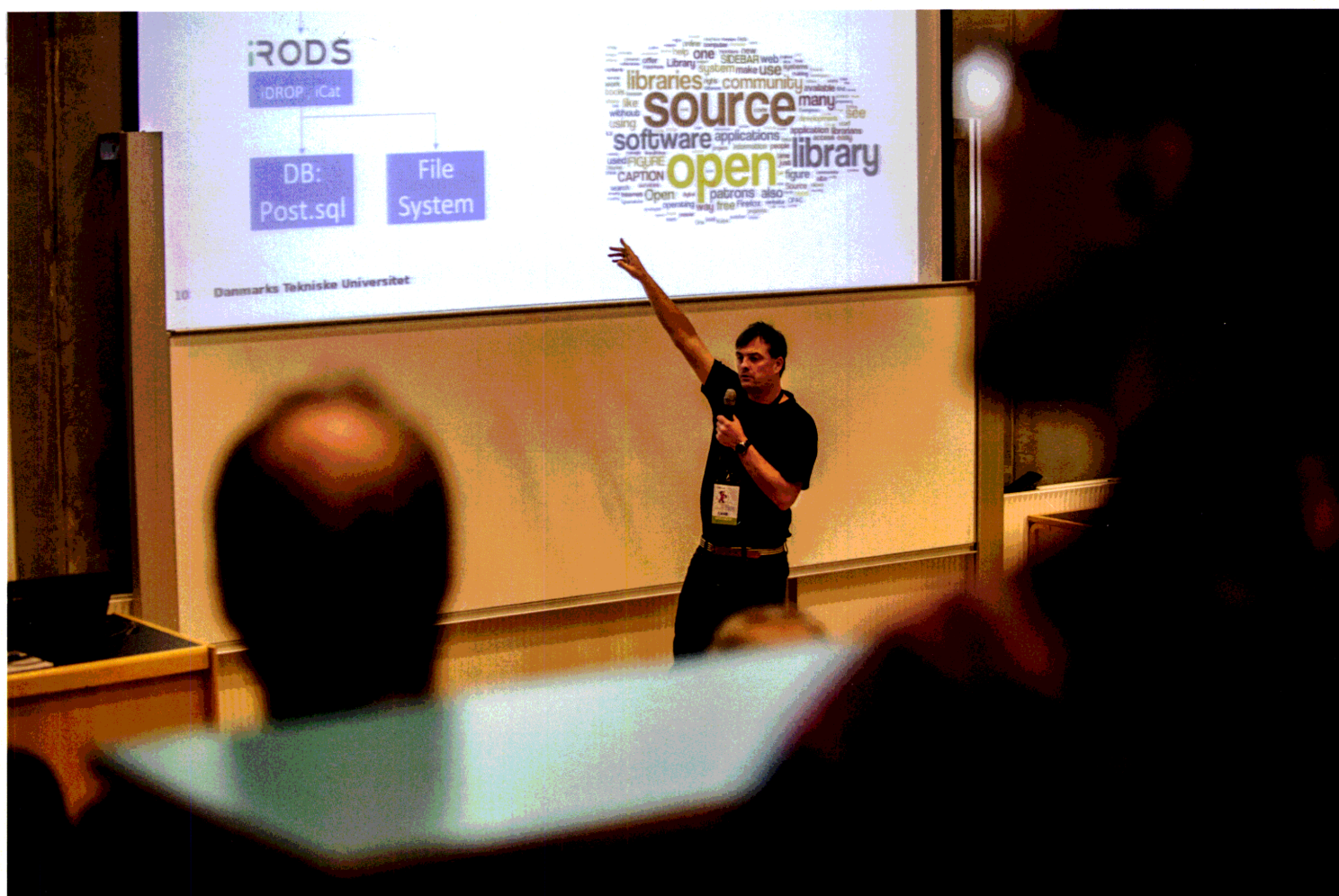


NYT

Open Source Days 2017 er foredrag, udstillinger og socialt samvær

Dansk Forum for Åbne Systemer

DKUUG - Unix Brugere (Linux/BSD) system administratorer

Community for IT-specialister og IT-interesserede.

Talere - speakers - til OSD 2017, Lørdag 18. Marts
i Metropol Professionshøjskole, Sigurdsgade 26
og Business Day Fredag d.17. Marts, Symbion

**DKUUG**

Teknologi-politisk set

Efterhånden som bilerne får flere sikkerhedsforanstaltninger skal vi til at overveje ansvaret for ulykker

Mens redaktionen ledte efter et illustrativt citat til artiklen om Microbitten, hvori der optræder et (simpelt) program, som efterligner HAL9000, den legendariske computer, som styrer en interstellær rum-mission, hvor besætningen er lagt i dvale, fra romanen **2001** af Arthur C. Clarke, tændte jeg for radioen og hørte programmet *Harddisken i P1*. Jeg fandt et citat, og lidt efter afspillede P1 værten lige netop det citat, som jeg havde fundet. Et besynderligt sammentræf, skal jeg lige love for, men en bekræftelse på at emnet Kunstig Intelligens, Artificial Intelligence (AI), er oppe i tiden og tages alvorligt.

Desværre er der mange indlæg i den offentlige debat, som er hårrejsende naive. Skal robotter have rettigheder, blev der således spurgt i en nyhedsudsendelse.

Kunstig intelligens bruges om mange forskellige ting, selvstyrende biler er et eksempel. Selvstyrende/selvkørende biler er skræmmende (for nogle af os): hvad vil der ske, hvis et barn træder ud foran en bil. Hvem har ansvaret?

Fabrikanterne har valgt at introducere dele af selvstyringsprogrammerne, beskyttelse mod kollision med forankørende, afstand til hvide sidestribes, hastighedskontrol osv. Men er det så også AI, hvis bilen "nægter" at køre ind i en mur eller er den betegelse bare et smart salgstrick?

Der er nogle, som mener at AI (når det er bedst) er en simulering af menneskelig adfærd, men ved vi nok om menneskelig adfærd til at den definition giver mening?

Bladets rammer

Det er ikke muligt for et foreningsblad som dette, DKUUG-NYT, at være part i debatten omkring IT-anvendelse, men redaktionen har i min tid lagt mere og mere vægt på at bladet skal have de væsentligste emner omtalt på en sådan måde, at læserne kan sætte det i relation til Open Source som begreb, som "commodity" (vare) og som ledelsesprincip.

Højdepunkter i Open Source anvendelse

Inde i bladet omtales flere emner fra OSD-2017. Foredragene er på engelsk hvis der er internationale gæster. Nogle af emnerne er på grænsen til at være kontroversielle. Fx. Tails, et *live*

system, som kan bootes fra en USB stick og som starter op i samme tilstand hver gang.

Ved Murens fald 1989 blev Internettet (som er baseret på Open Source) brugt til at kommunikere nyheder ud over jernetæppet, og ind igen, så at sige, så borgerne i bl.a. Rumænien kunne se hvordan tingene ændrede sig lynhurtigt. Engang i SSLUG fortalte Nick, som var flygtet fra Rumænien i netop de år, hvordan det var at sidde med forskellige forbindelser. Når den ene glippede eller blev stoppet, måtte man finde en anden.

I krig er medierne en del af krigen. Får vi hele historien at vide fra Syrien, Irak og Libyen? Medie-kontrol er et problem, falske nyheder har spillet en rolle i US valgkampagnerne. Med Internettet (som er Open Source baseret) kan både de gode, de onde og de grusomme kommunikere frit om politiske forhold over landegrænserne; men når der tales om anonymitet på internettet, så kan det skade Open Source og frit Internet, fordi der bliver associeret til kriminalitet i stedet for bedre nyhedsdækning.

De, der deltog i sidste års Open Source Days, vil måske huske Ole Tanges keynote om forebyggende kryptering. Det, der var den bærende tanke, var at man i et diktatur må kunne beskytte sig mod forfølgelse. Hvis ingen bruger krypterede mail, forklarede Ole Tange, så vil en krypteret mail straks skille sig ud. Hvis kun de, der er forfulgt, benytter kryptering, så kan et undertrykkende regime hurtigt finde de forfulgtes forbindelser.

Det lyder lidt skræmmende. Det lyder også lidt af sølvpapirshat: Forebyggende kryptering! Jeg har ikke noget at skjule. Jeg vil gerne have at man kan spore de, der måske planlægger en bombe i Metroen. Men på den anden side, det er vigtigt at vide, hvordan man kan bruge og misbruge Internettet.

DKUUG tager chancen og omtaler **Tails** inde i bladet.

Mindst lige så vigtige emner -

Ingen software uden hardware: Open Hardware, ildsjæl og forsker Alicia Gibb kommer og fortæller.

PostgreSQL er repræsenteret af "vores egen" Svenne Krap, som har valgt at kalde foredraget *Reliable, fast and flexible - but older and unloved* YesSQL

Dette **unloved** om PostgreSQL som modsætning til NoSQL gælder ikke redaktionen her, som har omtalt PostgreSQL ved flere lejligheder: det er et projekt som hører til i eliten af Open Source sammen med Linux, BSD og X11.

Donald Axel

■



DKuug-NYT er medlemsblad for DKuug, foreningen for Åbne Systemer og Internet Nr. 183 - Marts 2017

Udgiver:

DKUUG
Fruebjergvej 3
2100 København Ø
Tlf. 39 17 99 44
email: blad@dkuug.dk

Redaktion:

Donald Axel (ansvarshavende)

Forsidecredits:

Redaktionen

Design og layout:

DKUUG/Donald Axel med LibreOffice

Annoncer:

pr@dkuug.dk

Tryk:

Lasertryk i Aarhus

Oplag:

350 eksemplarer

Artikler og inlæg i DKUUG-Nyt er ikke nødvendigvis i overensstemmelse med redaktionens eller DKUUGs bestyrelses synspunkter.

Eftertryk i uddrag med kildeangivelse er tilladt.

Deadline for nr. 184: Fredag d. 19. maj 2017.

Medlem af Dansk Fagpresse

DKUUG-Nyt
ISSN-1395-1440



Vores møder og foredrag holdes - med mindre andet udtrykkeligt angives - på vores adresse:

**DKUUG
SYMBION
Fruebjergvej 3
2100 København Ø**

Hvis man kommer lidt før, er der tid til en snak på kontoret. DKUUG bor i en virksomhedsfarm, Symbion, hvor der er åbne døre indtil kl.18 eller 19 (afhængig af mødetidspunkt). Efter den tid har vi på foredragsaftener en vagt ved døren.

INDHOLD:

Open Source Days 2017	4
Introduktion til BBC Microbit SBC af David Askirk	10
FOSDEM-2017 set fra NOC	14
FOSDEM-2017 foredrag on-line	18
Kommando(linie)centralen	19

Kalender:

Open Source Days 2017, Business Day: 17. Marts 2017

The doors will open at 8:30 for registration and breakfast, and the day will be packed with talks from 9:00 - 15:00. We are this year hosted by [Symbion](#), Fruebjergvej 3, DK-2100 Copenhagen Ø

Lunch is included and will be served by the Symbion cantina.

Buy your [ticket](#) today for the business day for 950 DDK and come and see what open source can do for You and Your company.
<https://opensource.org/business/tickets>

Open Source Days 2017, Community Day: 18. Marts 2017

Open Source Days 2017 is again targeted at newcomers to open source technology and experienced members of the community alike. With 2 geek tracks and a novice track, we strive to create an atmosphere which inspires to learn and get involved. We will be opening the doors at 10.00 and will have the day packed with talks until 18.00.

We are this year again hosted by [Metropol](#), Sigurdsgade 26, DK-2200 Copenhagen N, a good place with large auditoriums as well as smaller ones. Also cantina. Buy your [ticket](#) today for only 200 DDK and join us to make this an unforgettable event. We can't do it without You.



Metropol Professionshøjskole, Sigurdsgade 26, DK-2200 København N

Onsdage : møde på kontoret fra kl.18 ca. - somme tider 17 eller før. Kontoret er i indgangshallen i Symbion, Fruebjergvej 3, DK-2100 København Ø.

DKUUG udgiver foreningsbladet DKUUG NYT 3 eller 4 gange om året. Bladet fremstilles indtil videre med LibreOffice. Skriv til os - brug fantasien eller kom ud med personlige oplevelser vedrørende Unix, Linux, BSD, AIX, OSX, Posix på MS-kerner, kom med hvad det nu end er. Fortæl om egne oplevelser med Open Source, applikationer såvel som databaser og webservere.

Redaktionen kan hjælpe med brugen af Libre Office, ikke kun for indlæg til bladet, men indlæg kan sendes i andre formater - plain text er altid godt.

Open Source Days Speakers - på dansk, i tilfældig orden

Vi bringer her omtale af foredrag - lectures, og talere - speakers, som kommer til OSD-2017, - og som i skrivende stund kan ses på websitet opensourcedays.org og vi prøver at fiske lidt mere baggrundsinformation frem - på dansk

Jurre Van Bergen: Tails

Tails er et operativsystem, som kan bootes på snart sagt enhver computer fra USB stick eller SD kort, eller fra DVD. Formålet er at bevare privacy - og anonymitet - og at hjælpe med at bruge internettet anonymt og undgå censurering. Det sigter efter at beskytte privatlivet, og gør det muligt at bruge Internettet privat, gennem TOR netværket for mindskelse af reverse IP lokalisering. Anvendelse af live Tail-boot efterlader ikke spor på den anvendte computer. Kryptering af filer, mail og instant messaging forhindrer aflytning.

Fra OSD-web om taleren:

Jurre is all over the place, by day he works at Greenhost, a sustainable webhosting company and in addition contributes to some open source projects like Tails and serves on the board of 'Hart voor Internetvrijheid' a non-profit organisation that runs Tor relays. Previously, he co-founded, Technologia Incognita, an Amsterdam hackerspace.

We'll give a small introduction to what Tails is how we rely on maintainers of various open source projects (likely you in this room!). We'll also give an overview of where we are and where we're going and how you can help.

Baseret på Debian

Tails kommer i nuværende udgave med Debians pakkesystem (Advanced Packaging Tool, APT) konfigureret til Debian's Onion service. Onion er en service som tillader fildeling (og dermed OS opdatering) via TOR (The Onion Router) uden at afsløre sit opholdssted (eller rettere IP-adresse). Inden vi går

videre bør man erindre at uden TOR/Onion havde vi ikke haft Snowdons afsløringer.

Onion routing sker ved kryptering af en hel protokol-stak i applikations-laget (end-user laget, data til dokumenter, billeder, lyd; de andre lag er hhv. transport, Internet (globale-adresser/"telefonnummer"-system), link, det fysisk-elektriske lag, som igen er opdelt i flere abstraktionslag, eftersom impulser på en ledning ikke altid opfører sig pænt.

Dermed kan TOR servere sende data videre gennem nettet uden at afsenderen ses indtil man havner i den anden ende hos en anden TOR server, som pakker ud og sender til den anden ende af kommunikationsforbindelsen.

Kerneprincippet i Tor, "onion routing", blev udviklet i midten af 1990'erne af ansatte i [United States Naval Research Laboratory](http://www.nrl.navy.mil), matematiker [Paul Syverson](http://www.nrl.navy.mil) og computer scientists [Michael G. Reed](http://www.nrl.navy.mil) og [David Goldschlag](http://www.nrl.navy.mil), med det formål at beskytte U.S. [intelligence](http://www.eff.org) online kommunikation. Onion routing blev derefter yderligere udviklet af [DARPA](http://www.darpa.mil) i 1997.

Fortsat udvikling og vedligehold af TOR

Alfa-versionen af Tor, som var lavet af Syverson med flere, fik navnet The Onion Routing project (TOR), og blev startet op 2002. En offentlig tilgængelig version kom året efter.

I 2004 releasede Naval Research Laboratory source koden. Derefter tog Electronic Frontier Foundation (EFF) over og finansierede, at Syverson og Nick Mathewson kunne fortsætte udvikling af projektet.

EFF er en non-profit, digital rights gruppe baseret i San Francisco.

Men derefter grundlagde to af udviklerne (Roger Dingledine og Mathewson) *The Tor Project* i Massachusetts, research og non-profit organisation som stod for vedligehold af TOR. Der er en række velanskrevne firmaer, som på dette tidspunkt støttede TOR finansielt: U.S. [International Broadcasting Bureau](http://www.fcc.gov), [Internews](http://www.fox.com), [Human Rights Watch](http://www.hrw.org), [University of Cambridge](http://www.cam.ac.uk), [Google](http://www.google.com), og [Stichting NLnet](http://www.stichting-nl.net) i Holland.

Med andre ord: TOR var ikke og er ikke et lyssky foretagende oprettet af hackere, som ønsker at dyrke deres laster i hemmelighed, men er skabt af grupper, som forudså at demokratisk politisk arbejde kunne komme i fare, hvis en præsident fik den idé at censurere pressen. Det er som bekendt set før.

Tor anvendes i dag over hele verden, Wikipedia viser et kort over anvendelse, gengivet her nederst på siden.

[Stefano.desabbata grafik](http://stefano.desabbata.com) - [link til Wikipedia](http://en.wikipedia.org): <https://commons.wikimedia.org/w/index.php?curid=34897647>

The anonymous Internet

Daily Tor users per 100,000 Internet users

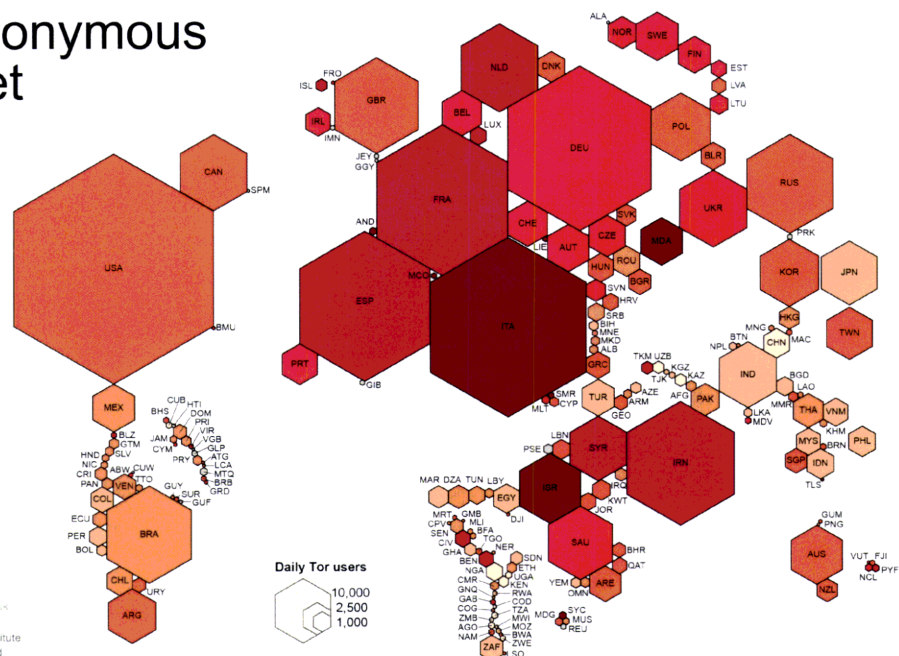
> 200
100 - 200
50 - 100
25 - 50
10 - 25
5 - 10
< 5
no information

Average number of Tor users per day calculated between August 2012 and July 2013

data sources: Portal Metrics, Tor Project, World Bank, data.worldbank.org

by Mark Graham (@geoplaces) and Stefano De Sabbata (@maps4thought) Internet Geographies at the Oxford Internet Institute 2014 • geography.ox.ac.uk

Oxford Internet Institute University of Oxford



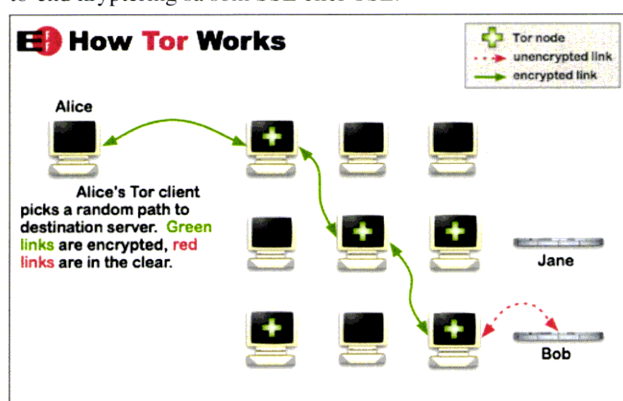
Er online anonymitet godt eller ondt?

I marts 2015 udgav det engelske parlaments styrelse for videnskab og teknologi ([Parliamentary Office of Science and Technology](#)) en rapport, som udtalte at "Der er udbredt enighed om at forbud mod systemer til online anonymitet generelt ikke ses som en acceptabel politik i U.K." Der blev tilføjet: "Selv om de var, ville der være tekniske udfordringer."

Rapporten konkluderede yderligere, at TOR kun spiller en mindre rolle i distribution via nettet af børneporno. Desuden blev Internet Watch Foundation rost for sin måde at bruge TOR til såkaldt skjulte services for whistleblowers og til omgåelse af den Kinesiske Firewall.

Exit node eavesdropping

I September 2007 afslørede Dan Egerstad, en svensk sikkerhedskonsulent, at han havde opsnappet brugernavne og passwords for e-mail konti ved at drifte og monitorere en TOR exit node. Da TOR ikke kan kryptere trafikken mellem en exit node og target serveren vil enhver exit-node kunne opfange trafik, som passerer gennem den, med mindre der benyttes end-to-end kryptering så som SSL eller TLS.



Selv om denne analyse ikke automatisk bryder kildens (brugerens) anonymitet, så kan trafik, som opfanges på denne måde af en tredjepart, afsløre information om kilden, enten ved payload data eller ved protokol-data. Egerstad gjorde opmærksom på at TOR kunne bruges af efterretningstjenester, som på denne og andre måder tog del i tjenesterne.

Illustrationen ovenfor er forsimplet, naturligvis. Hvis man vil aflytte TOR forbindelser, skal man have fat i flere stier gennem nettet, men hvis man har en exit-node, kan man naturligvis se, hvordan denne distribution foregår.

Forskere fra det Franske Computer Science institut (French Institute for Research in Computer Science and Automation, **INRIA**) kombinerede flere metoder, alle sammen baseret på at man kan aflytte en exit-node. Blandt andet opsnappes BitTorrent control messages, og da Bit Torrent bruger forskellige metoder - som alle er kendte - til at lokalisere servere og dermed forbindelser, kan man på denne måde opfange slutbruger IP-adresser.

INRIAs rapport (som kan downloades via link på Wikipedia, den er på engelsk) siger at de var i stand til at afsløre private IP numre ved at køre 6 exit-noder over en periode på 23 dage.

Implementering af TOR

Selve begrebet TOR dækker kun over den software, som kører på en node - mellemstation - og en exit-node. Denne del af TOR netværket (dvs serversoftware) er for det meste programmer skrevet i C, men der er også noget Python kode.

Brugeren kan bruge hvadsomhelst-software, men hvis man bruger en almindelig browser, så skal man jo lige fortælle den, at den skal finde en TOR-indgang.

Andre metoder til tracking af Internet aktivitet

Illustrationen viser hvordan TOR fungerer, altså lagvis indpakning af det, som er kernen i transporten, brugernes data, payload. Det er ikke så svært at se hvor sårbarhederne er. Hvilke angrebsvinkler kan man anvende, hvor kan det betale sig at lægge en "snabel" ind?

En exit-node er det oplagte sted. Men hvad hvis man kunne "lytte" direkte på brugeren - ved at have en lille spion, svarende til malware, i brugerens browser eller netsoftware?

Ja, tænk hvis man kunne få brugeren til at bruge en patched Firefox browser, som fx. sender besked "hjem" til forskeren (eller efterretningstjenesten) - så ville det være klareret. NSA har brugt den metode under kodenavnet "den egoistiske giraf" - engang i 2013, iflg. The Guardian.

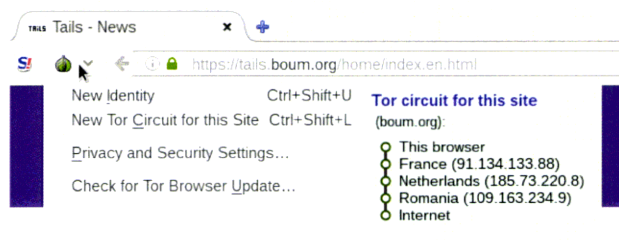
Egotistical Giraffe var javascript i browseren, som kunne aktiveres remote, og som så kunne sende informationer.

Af flere grunde er det vigtigt, at der forskes både i beskyttelse af privatlivet og i beskyttelse mod kriminelle på Internettet.

Hvordan kommer man så i gang med TOR?

For at gøre det lettere har Tails opsat end-user applikationer, sådan at de bruger TOR og det er meget praktisk for både nye og erfarne brugere.

Foredraget af Jurre van Bergen fortæller om download, installation på USB, boot fra USB stick, applikationer, som følger med - og opdatering eller download af programmer fra "onion" service.



Open Source Days 18.Marts



Allan Jacobsen

Containers and virtualisation (cloud)

I would like to give an introduction to what containers and virtualisation is, and why it is widely used in the IT industry.

Containers: Virtualisering på operativsystem niveau: mindre overhead, mindre flexibilitet.

Alicia Gibb

Open Source Hardware

With open Source Hardware advancing in several fields, the Open Source Hardware Association has launched a new Open Source Hardware Certification, to help standardize the definition and grow the community. Learn about why Open Source Hardware is vitally important to innovation and how we can push the movement forward.

Bio

Alicia is the founding president of the Open Source Hardware Association. In her spare time, Alicia tinkers with open source hardware. Prior to becoming an advocate for OSHWA, Alicia was a researcher and prototyper at Bug Labs where she ran the academic research program and the Test Kitchen, an open R&D Lab. She is a member of NYCResistor, co-founder of the Open Hardware Summit, and a member of the advisory board for Linux Journal. She holds a degree in art education, a M.S. in Art History and a M.L.I.S. in Information Science from Pratt Institute.

Alicia Gibb er fortaler for open hardware, forsker og hun er også programmør på hardwareniveau. Hun har grundlagt Open Source Hardware Association, en non-profit organisation, som underviser i fordelene ved at bygge og bruge open source hardware (man bliver ikke så let fanget i udsolgte CPU'er!)

*Open Source hardware community mødes ved **Open Hardware Summit**, som Alicia også fik startet, her diskuteres forretning, produktion, jura, og alle ting, som vedrører Open Source hardware.*

Rasmus Frey

How Danish Municipalities use open source to drive development and cooperation

Insights to how Danish municipalities through "OS2 – Public Digitalisation Network" work together and share results.

OS2 is:

Openness – transparency is key in an open community.

Collaboration – we are better and stronger if we unite.

Sharing – knowledge and good solutions should not be proprietary.

Digital development – we want to be frontrunners where relevant.

Open source licenses – prevents vendor lock-in.

I will explain why open source create value for both the municipalities (local government, public sector) as for the vendors. And I will explain how we take action and do it in practice.

OS2 = OS and the 2 as the exponent. The meaning is that OS is short for both Open Source and Offentligt Samarbejde (Public collaboration).

Freso (Frederik Sandberg)

*MusicBrainz and other *Brainz (WT)*

MusicBrainz is a crowd-sourced open source & open data music metadata database.

Bio

Freso works as Community Manager for the MetaBrainz Foundation (the legal entity behind MusicBrainz and the other *Brainz projects), but has also been a volunteer with them since 2006.

Besides working on and with all the *Brainz, he's also involved with Lutris, LOOT, and random other FLOSS projects.

[MusicBrainz](#)
[MetaBrainz Foundation](#)

Niklas Nisbeth

Open source tools for FPGAs

Field Programmable Gate Arrays (FPGA) er en chip, der indeholder programmerbar logikkomponenter kaldet "logikblokke" og programmerbar/rekonfigurerbare ledninger og databusser. Logikblokkene kan udføre Nand-gate, Exclusive-or-gate eller mere komplekse logikfunktioner som f.eks. dekodere eller simple matematiske funktioner. I de fleste FPGAer indeholder logikblokkene også hukommelselementer eller mere omfattende hukommelsesmængder.

Whether the topic is bitcoin mining, software-defined radio, retrocomputing, or the promise of having infinite Arduini on a stick, FPGAs are moving closer to the front of hacker consciousness.

But what exactly are FPGAs, and how do they relate to CPUs, GPUs, DSPs and ASICs? This talk will clear it all up and show the present status of Yosys and related projects that provide a fully open source toolchain for FPGAs.

We'll show some of the fun things that people from Labitat have been doing with FPGAs and sound a call to action - to use, abuse, and improve the tools so that open source can be a first-class citizen of the reconfigurable future.

Bio

Niklas Nisbeth (aka nn, rvense) is a hacker from Copenhagen interested in all aspects of open hardware and software, especially the kinds that make electronic music.

[The Yosys Open Synthesis Suite](#)

Georg Koppen

Surfing the Web over Tor with Tor Browser

In this talk we will explain how Tor Browser allows users to surf the web privately and anonymously. After giving a short overview of Tor and the Tor network we will examine the Tor Browser components and how our privacy-by-design approach protects users without sacrificing usability.

Marcin

The basics of Snort - a network-based intrusion detection and prevention system

Snort is one of the very few open-source network-based intrusion detection and prevention systems. While the general idea what such a system does seems easy to grasp, the practical configuration and usage is challenging due to the flexibility this tool offers. Writing a traditional documentation describing real-life scenarios is almost unfeasible in the world of the software changing constantly. Let's instead create a virtual test lab and play the attacker vs. security analyst game ...



Jeffrey Goff

The Top 10 Things you need to know about Perl 6

Sit back and learn about the top ten features that Perl 6 brings to the table, like Unicode support, functional programming, reactive and concurrent programming, built-in expression grammars, built-in vector operators and a full metaprogramming system including support for roles. We'll talk about where Perl 6 started, where it is today, and where the language is going in the future.

Bio

Jeffrey Goff has spoken at conventions worldwide and is an active contributor to both Perl 5 and Perl 6. Jeffrey is a senior developer at <http://evozon.com/>, a web design agency in Cluj, Romania, where he architects and builds custom web applications with Perl, HTML, and JavaScript. He is one of the original release managers for Parrot née Perl 6 and has written some core interface modules for Perl 6. At home, when he's not creating Perl 6 libraries or doling out Perl 5 advice online, Jeffrey does origami and travels. You can follow Jeff on <http://twitter.com/drforr> and view his open source contributions on <http://github.com/drforr/>. He also blogs at <http://theperlfisher.blogspot.ro/>.



Ronni Lindsgaard

How to (not) Store Your Password

There is much dispute on how to securely store passwords. Traditionally there are two schools: Online and Offline password management.

Both have strengths and weaknesses when it comes to security as well as accessibility and portability. Both methods have one weakness in common though: they store your password. In this talk I will discuss the various strengths and weaknesses of traditional password managers and present a method that has the main strengths of both online and offline password managers, but allows you to generate secure passwords on-the-fly, never to store them again.

Bio

Ronni Elken Lindsgaard is maintainer and primary developer of RndPhrase. He works as Software Engineer for a Danish security company and is active in various open source projects and local groups in Copenhagen related to personal security and privacy awareness.

Thue Janus Kristensen

(Lack of) Cryptographic Security of NemID, Dankort, and Mobilepay

I will be giving my definition of the concept "cryptographic security", using NemID, Dankort, and Mobilepay as examples of systems which are not cryptographically secure, and how to fix them.

I recently wrote a blog post "NemID is not cryptographically secure", which was well received: <https://www.version2.dk/blog/nemid-er-ikke-kryptologisk-sikker-og-myndighederne-er-lige-glade-513303>.

Cornelius Kölbel

two factor authentication with open source project privacyIDEA

privacyIDEA is a two factor authentication system. It manages many different kind of OTP tokens, yubikeys, SSH keys, X.509 certificates, the nitrokey... It comes with a great WebUI and a simple REST API. Using plugins it can be integrated into simpleSAMLphp, Wordpress, SSH, PAM, ownCloud, Nextcloud, Typo3, FreeRADIUS and many other applications. With a sophisticated policy concept and event handler framework it is easy to adapt the system according to your workflows. It also allows easy and smooth migrations from proprietary 2FA systems to privacyIDEA. This talk will be about migrating to this open source two factor authentication system, integration into ownCloud and Nextcloud and current developments in 2017.

[Privacyidea](#)

[Privacyidea documentation](#)

[Privacyidea youtube channel](#)

Emma Lilliestam

IoT ethics / Cyborgs

IoT brings up entirely sets of ethical problems. The issues of ethics in devices that can identify and monitor us needs to be addressed before they are adopted by the broad public, not, like in the case of mobile phones, after the fact.

In March, I implanted myself with a chip implant. I partly chose to do that as an immersive risk assessment. My initial thoughts can be found here. <https://www.linkedin.com/pulse/usability-versus-security-why-i-decided-become-cyborg-emma-lilliestam?trk=mp-reader-card>

Bio:

Emma Lilliestam is an IT security analyst, head of devops at the IoT startup Ewa Care and a chip implanted cyborg.



Svenne Krap

Reliable, fast and flexible - but older and unloved (YesSQL)

The modern world runs on data, which streams around it increasingly greater amounts.

Managing this data and keeping it safe is an unheroic and slightly boring task, but it is critical - more so now than ever.

To combat this amount of "just keep working" (and often to make a pretty penny), people are always looking for the new way. The last few years, we have seen an uneven matched group of technologies usually called NoSQL.

This talk is a counter point, it's main idea is that we have an incredible tech known as RDBMS ("sql databases") which we know and trust. It has served dutifully for 45+ years. Like sanitation, you should really want a piece of tech that silently and dutifully does it's job.

This talk is aimed at developers of from beginners to experts. It will contain a rationale for the main three concepts of RDBMS (relations, transactions, and declarativeness) and why you really want those things.

It will go through some examples that show the immense power a modern database gives you both in business-as-usually transactional and adhoc data exploration duties.

It will use PostgreSQL (version 9.6) as an example tool, but most content is applicable for other databases.



Bo Simonsen

Kopano - sharing & communication software for business

This talk will introduce Kopano, a fork of Zarafa which aims to provide a fully AGPLv3-licensed collaboration platform. Kopano Core provides the core MAPI-enabled messaging stack with the stability and flexibility of the linux platform. Kopano Core acts as the solid foundation for groupware messaging based on MAPI and enabling rich web clients such as Kopano WebApp or Kopano DeskApp as well as mobile and sync clients (by the Z-push project). Kopano Core also offers the Kopano Gateway that provides access mailboxes and calendars via standard protocols: IMAP, POP3, iCal and CalDAV. With the modular architecture, Kopano Core hereby allows a variety of setup scenarios, scalable from a very low powered system to a multi-datacenter setup providing messaging capabilities to tens of thousands of users.

Bio

I am Senior Backend Developer working for Kopano B.V. on the core of Kopano. I was born in Denmark, earned M.Sc. in Computer Science, and used Linux since 1999 as my preferred operating system.

[Kopano](#)
[Kopano Community](#)

Ximin Luo

Reproducible builds, its uses and the future

In the past few years the Reproducible Builds project has managed to make 94% of ~25k Debian source packages reproducible (77% when varying the build path). Ximin Luo will talk the ways this can be used, not only to gain more confidence about what you're running, but also to improve efficiency when developing your own software. He'll also talk about our plans for the future, including how we'll turn both the first and second figures into 100%.

Hanno

Is Linux Desktop security worse than Windows?

A couple of very scary security vulnerabilities have recently been discovered in the Linux Desktop. Given the design choices browsers, Linux distributions and desktop environments made this isn't that surprising.

Desktop search tools like Tracker (GNOME) and Baloo (KDE) expose downloaded files to a wide variety of parsers of very questionable quality. Automatic file downloads allow exploiting these bugs remotely through the browser.

How bad is the security of the Linux desktop and what can we do about it?

Speaker Info:

Hanno Böck is a freelance journalist and hacker. He regularly writes for the German IT news webpage Golem.de. He also runs the Fuzzing Project and tries to improve the security of Free and Open Source Software. The Fuzzing Project is funded by the Linux Foundation's Core Infrastructure Initiative.

David Fotel

Learn Python - The language of the future

Learn how to code and think in Python.

Learn how to code in python using Python 3. Learn OO programming, learn to use python to solve day-to-day issues and create larger apps in python.

To get the most from the session, please bring your laptop with python pre-installed.

Bio

David has used Linux and open source for more than 15 years. Python have been favorite language for 10 years.



BBC Microbit computer (arbejds titel)

Af David Askirk

Instruktør v. SuperUsers a/s



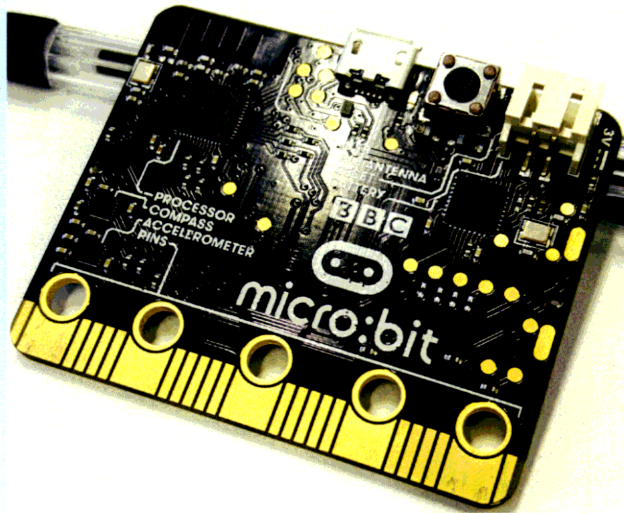
Intro

BBC Micro var en computer lavet i England i 1981 med fokus på at lære folk i almindelighed om computere. Det var et projekt startet af BBC.

Den blev en stor succes og var udbredt. Den kom endda til Danmark.

Nu har BBC lavet BBC Micro:bit som også er lavet til undervisning. Den er så blevet udleveret til alle engelske skolebørn i alderen 11-12 år, for at vække glæden ved programmering. Det var over 1 million enheder, der skulle uddeles. Det blev forsinket, så de først blev uddelt i marts 2016.

En Micro:bit er ca 4 x 5 cm og indeholder 5x5 LED, BlueTooth, accelerometer, kompas og to knapper. Alt dette er til rådighed når man skal programmere den. Der er også en række I/O porte som man kan bruge. Og så er der et USB Micro stik til at forbinde til computeren, samt et batteri stik, hvor man kan tilslutte et eksternt batteri.



England har besluttet at børn behøver et skub for at lære at programmere og skabe ting i stedet for at forbruge. I Furesø kommune har der været Week of Code, hvor BBC Micro:bit også er blevet brugt.

Bit'en er lavet til at passe ind sammen med Raspberry Pi i den engelske dataundervisning.

Udover selve dimsen kan man tilkøbe andre ting. Blandt andet en motor controller hvor man blot sætter bit'en ned i og derefter kan man styre forskellige motorer - dvs man kan bruge den til at styre en robot. Et andet, meget praktisk, udvidelsesmodul er et modul hvor man får en buzzer samt mulighed for at sætte et CR2032 knapcelle batteri i. Så kan den køre uden at være forbundet til computeren, og man kan spille simple melodier.

En BBC Micro:bit kan programmeres på flere måde. Fælles for alle måderne er at det foregår i en browser.

En Micro:bit understøtter forskellige programmeringssprog, herunder Python og noget mere grafisk programmering hvor man kan trække klodser ind og forbinde dem.

Når man forbinder bit'en til computeren dukker den op som et drev på computeren.

Når man har skrevet sit program, downloader man programmet som en hex fil og gemmer denne fil på BBC Micro:bit'en. Man behøver ikke at slette de gamle programmer, da systemet sørger for at der kun kan ligge en fil på bit'en ad gangen. De gamle filer bliver fjernet af styresystemet.

Når systemet kan se at der kommet en ny fil, bliver den indlæst og programmet kører. Mere skal der ikke til.

```
# Add your Python code here. E.g.  
from microbit import *
```

```
while True:  
    display.scroll('Hello, World!')  
    display.show(Image.HEART)  
    sleep(2000)
```

Hvis man vælger at køre dette kode, så skriver den "Hello, World!" som scrolltekst på de 5x5 LEDer samt viser et hjerte.

HAL 9000 eksempel

Nu til et lidt større eksempel. Der skal laves en HAL 9000 inspireret applikation.

HAL 9000 er en rolle, en fiktiv karakter, og hovedskurken i Arthur C. Clarke's Rum-Odyssee, roman og film. Første introduktion er i filmen 2001: A Space Odyssey; her er HAL (Heuristically programmed ALgorithmic computer) en computer med menneskelige reaktioner, som kontrollerer rumskibets systemer og som snakker med astronaut-besætningen. En smule af HAL's hardware vises sidst i filmen, men den forekommer mest som en kameralinse, der indeholder en rød skive med en gul prik i midten, og den slags "øjer" forekommer flere steder på rumskibet.

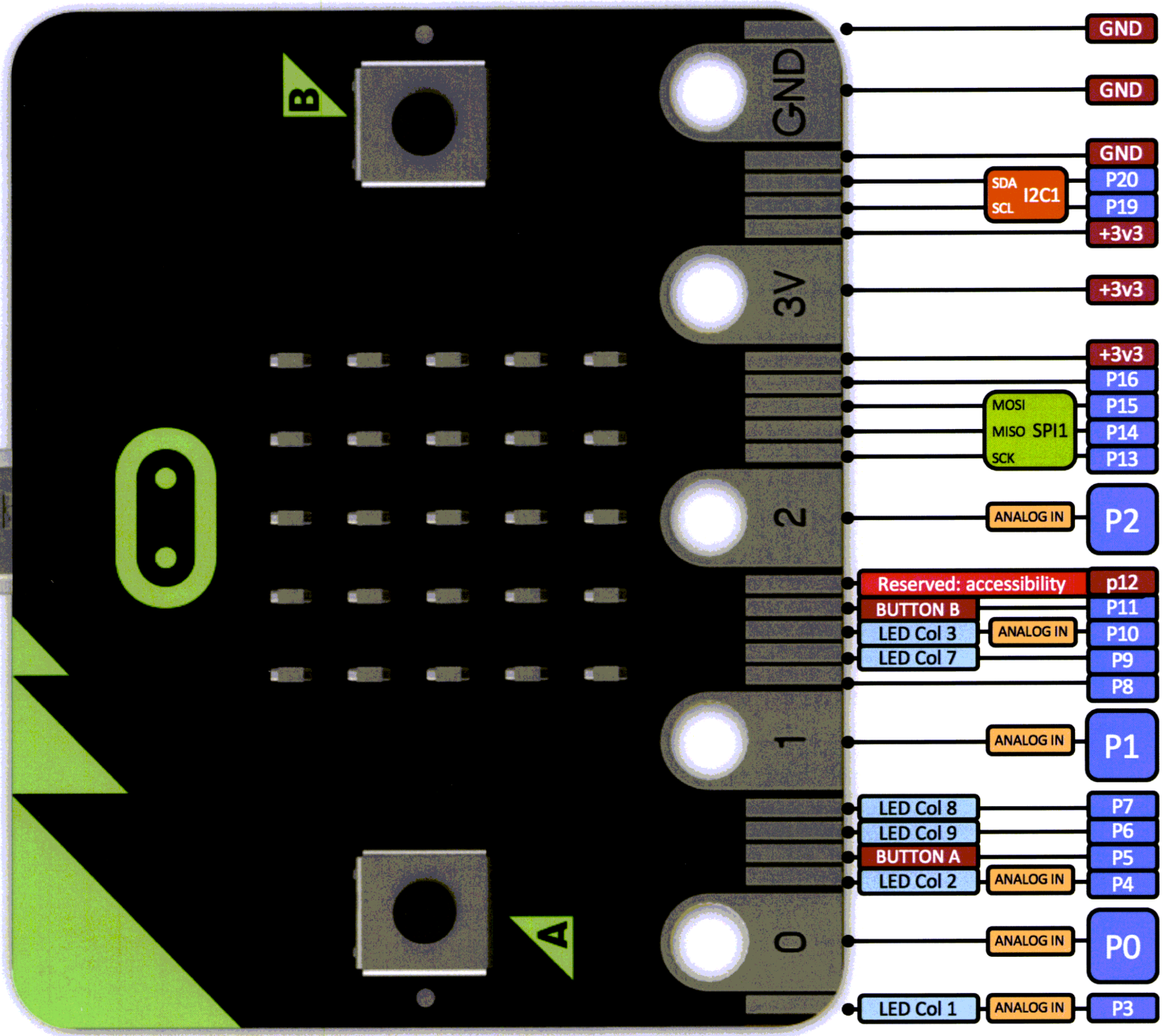
HAL-navnet kan også have forbindelse til navnet IBM - træk 1 fra character-code for hvert bogstav i IBM, så får du HAL.

Først skal der importeres nogle ting så de er til rådighed senere:

```
from microbit import *  
import music
```

Dermed importeres alt, hvad der skal bruges til at snakke med en Micro:bit, samt til at spillet lidt musik.

→ side 12



HAL9000 i romanen 2001 kan ikke lide at man rører ved den! Klart nok, den er sårbar, hvis man tager strømmen eller skiller den ad. Dave ser sin ven blive udelukket fra rumskibet, ilt-forsyningen kappes af HAL; i en berømt scene prøver Dave forgæves at redde sin ven, og da han vender tilbage, vil HAL ikke åbne luftslusen.

Dave Bowman: Hello, HAL. Do you read me, HAL?

HAL: Affirmative, Dave. I read you.

Dave Bowman: Open the pod bay doors, HAL.

HAL: I'm sorry, Dave. I'm afraid I can't do that.

Dave Bowman: What's the problem?

HAL: I think you know what the problem is just as well as I do.

Dave Bowman: What are you talking about, HAL?

HAL: This mission is too important for me to allow you to jeopardize it.

Dave Bowman: I don't know what you're talking about, HAL.

HAL: I know that you and Frank were planning to disconnect

me, and I'm afraid that's something I cannot allow to happen.

Dave Bowman: [feigning ignorance] Where the hell did you get that idea, HAL?

HAL: Dave, although you took very thorough precautions in the pod against my hearing you, I could see your lips move.

Dave Bowman: Alright, HAL. I'll go in through the emergency airlock.

HAL: Without your space helmet, Dave? You're going to find that rather difficult.

Dave Bowman: HAL, I won't argue with you anymore! Open the doors!

HAL: Dave, this conversation can serve no purpose anymore. Goodbye.

Dave ser at HAL er løbet løbsk. Han kommer ind ad en nødudgang og lukker ned for Hal.

Dernæst skal der laves nogle "billeder" som definerer hvilke LEDer der er tændt og slukket samt deres styrke:

```
face1 = Image("00500:"  
              "05950:"  
              "59995:"  
              "05950:"  
              "00500")
```

```
face2 = Image("03730:"  
              "37973:"  
              "79997:"  
              "37973:"  
              "03730")
```

```
face3 = Image("35953:"  
              "59990:"  
              "99999:"  
              "59995:"  
              "35953")
```

For at kunne bruge kompasset skal det kalibreres først. Det gøres således:

```
compass.calibrate()
```

For at gøre klar til at spille en animation med de billeder som er defineret tidligere skal de lægges i nogle lister:

```
all_faces = [face1, face2, face3]  
all_faces_rev = [face3, face2, face1]
```

I denne applikation skal den fortælle at den ikke vil røres. Dette gøres ved at bruge accelerometeret. Når man løfter på bit'en skal den skrive en besked til brugeren:

Leave me alone!

Først skal de nuværende værdier aflæses og gemmes i en variabel:

```
old_vals = accelerometer.get_values()
```

Så skal der laves et loop som starter med animation af ansigt (meget primitivt, men giver dog en fornemmelse) derefter aflæser den knapperne, og skriver beskeder og lignende, se hele programmet side 13. ▶

Sådan skal programmet køre:

Først bliver billederne vist i den ene rækkefølge som en animation med 200 ms mellem hvert billede.

Dernæst bliver der set om der er trykket på den ene eller den anden knap. Hvis der er blevet trykket på knap A skriver den en besked på skærmen og spiller en lille melodi. Ved tryk på knap B viser den retningen til nord.

Så bliver der vist billeder i omvendt rækkefølge for at lave animationen.

Til sidst aflæses accelerometer værdierne igen. Hvis der er for stor ændring gætter bit'en på at der er blevet rørt ved den og der bliver skrevet en besked til brugeren og accelerometer værdierne bliver opdateret til de nyeste værdier.

Links

MicroPython reference:

<https://microbit-micropython.readthedocs.io/en/latest/>



Et bud på hvorfor python er så populært

Python er vokset i popularitet. Specielt i de seneste år er den steget kraftigt.

Det er en selvforstærkende trend - python bliver brugt flere og flere steder. Mange af stederne har sprog som perl tidligere været enehersker.

Python er fra 1991. Python har siden sin begyndelse haft fokus på læsbarhed.

Noget man skal vide om python er at det ikke er et oversat sprog, men et fortolket sprog. Dog er der stadig noget oversætning når et python program bliver afviklet. Et python program bliver oversat til bytecode, som så bliver afviklet på den virtuelle python maskine.

Som en kommentar, der måske ikke helt har hold i virkeligheden, kan man lave en sammenligning mellem perl og python:

Hvis du har en erfaren perl programmør bruger han primært alle andre tegn en a-z. Dette gør koden meget svær at læse. I python bruger man alle tegnene, og dette giver meget meget læsbar og forståelig kode, som man også kan forstå mere end 3 timer efter man har skrevet koden.

Man siger, nok også for at drille, at python er executable pseudo-code hvorimod perl er executable noise.

I matematikkens verden er python også blevet mere og mere udbredt. Med pakker som numpy og sympy har python overtaget en stor del af den numeriske matematik. Med muligheder for mange grene af matematikken kan python klare de fleste krav.

Til stort set alle opgaver har python muligheder. Hvis du ønsker at lave analyse af sprog, er der NLTK - Natural Language Toolkit.

Python er meget nemt at lære. Sproget har få konstruktioner, så det er nemt at overskue. Samtidig er det meget læsbart, og det ligger i sproget selv. Det gør, at begyndere meget hurtigt kan komme i gang og skrive avancerede programmer. Kombiner dette med nogle af de mange forskellige biblioteker der er til rådighed, og man kan lave nogle fantastiske ting uden at skrive mange linjers kode selv.

```

while True:
    display.show(all_faces, delay=200)
    if button_a.was_pressed():
        display.scroll('I cannot let you do that Dave')
        music.play(music.POWER_DOWN)
    if button_b.was_pressed():
        needle = ((15 - compass.heading()) // 30) % 12
        display.show(Image.ALL_CLOCKS[needle])
        sleep(2000)
    display.show(all_faces_rev, delay=200)
    new_vals = accelerometer.get_values()
    if abs(new_vals[0]-old_vals[0]) > 30 or
       abs(new_vals[1]-old_vals[1]) > 30 or
       abs(new_vals[2]-old_vals[2]) > 30:
        display.scroll('Leave me alone')
        old_vals = new_val

```

BBC Microbit er en singleboard computer med 32 bit ARM ®Cortex™M0 CPU 16K statisk ram, 16MHz, Bluetooth Low Energy

Microbitten kommer med Python installeret. Det er ikke samme fulde Python installation som på en workstation, men til gengæld har Microbitten et Python-modul, som kommunikerer med hardwaren og som gør det nemt at programmere den lille sag (43 mm × 52 mm, halv-kreditkort størrelse!)

CPU'en har on board Bluetooth, og med den kan man lave trådløs netforbindelse.

Microbit har en NXP controller, som giver fuld speed USB-2.0, spændingsregulering, - et accelerometer, - et magnetometer, - micro-USB stik, 23-pin kant-connector, - display 5x5 LED'er (altså meget lille!) - to knapper.

Den kan bruge eksterne AAA batterier. Det hele er så småt, at man kan gå med den i lommen.

Hvordan programmerer man Microbitten?

Man bruger en browser, som startes op og som kan "se" Microbitten; men man kan også bruge et lidt mere specialiseret program, en editor kaldet **MU**. Denne editor kan downloades og installeres meget nemt, den er baseret på (skrevet i) Python og kører på Microsoft Windows, OSX, Linux og Raspberry Pi (også Linux) X-Windows Den kan også køre i terminal modus.

Når man har skrevet sin kode i MU, er der en "flash" knap så man let og elegant kan sende koden til Microbittens statiske ram.

Men foruden MU-editoren er der adskillige andre, som kan mere og derfor er mere uoverskuelige - MU anbefales for begyndere.

BBC micro:bit rystes også i danske skoler, skriver en af butikkerne, podconsult.dk -- Den bliver rystet, drejet, kastet og kodet.

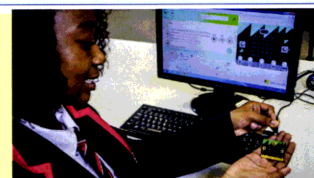
Siden sommeren 2016 har butikken podconsult.dk haft BBC micro:bit på lager. Også appacademy og andre elektronik-specialister har denne mini-maskine. Prisen? med starter kit små 200 kr. Ikke nogen herregård. Her er nogle links:

<http://podconsult.dk>

<http://itogleg.dk>

<http://inklusiononline.dk>

<http://raspberrypod.dk>



BBC.co.uk: Kvinde med Microbit

Debugging - se hvad der sker

Fejlfinding - stepvis kørsel af et program, (tracing et step ad gangen, undersøgelse af hvad status der er på forskellige bits og bytes på hardwaren) er muligt, så man kan få indblik på maskin-niveau.

Demoprogrammer

På Microbit-websitet er der små programmer, som man kan downloade og prøve; men der er også forslag til, hvordan man selv kan gå videre.



FOSDEM 2017: a view from the NOC



Peter Van Eynde - February 8, 2017 -

[FOSDEM 2017](#) was again a great success. We did a bit less analysis compared to 2016, but the numbers we got indicate the number of visitors grew significantly compared to last year: the total number of unique MAC addresses went from 9711 to a stunning 11918, an increase of 22.7%.

The number of mobile devices, a more accurate indication of the number of visitors, also went up. For Android, the number of unique MAC addresses went from 3892 to 4640 (+19.2%) and for iOS from 1060 to 2579 (+143.3%).

As in the past years we had a IPv6-only main network and a dual-stack legacy network for the people who needed it. The SSID of the dual-stack network was changed to encourage visitors to try the IPv6-only network. This seems to have worked as the IPv6-only network was used more to connect to IPv4-only hosts compared to the previous edition: this NAT64 traffic went from 6.1 million sessions in 2016 to 10.1 million in 2017 (+65%).

The traffic towards the internet rose from a mere 2982 million packets and 979.8 GB to 7924 million packets and 9.321 TB of traffic (+65% and +851%). From the internet, we received 2621 million packets and 2.912 TB of traffic in 2016, in 2017 it was 3620 million packets and 2.733 TB (+38% and -6.14%).

Most of this increase in outgoing traffic was due to the amount of traffic the [Video team](#) were pushing. They report: *The video team pushed ~288 GB over the internet to the primary restreamer, the same amount to the backup one, and 7.1 TB (sustained 300 Mbps) to the small monitoring/control host that generated the thumbnails used in the control of the video mixer. This probably makes us the biggest user of the internet connection.*

In fact, they were pushing *too much* traffic. We had not planned for this increase in traffic and the switches we used for the last few years were reaching their limits. We noticed this when we got reports of packets getting dropped. First we checked the load on the switches:

```
video-switch-1#show controllers\
utilization
Port          Receive  Transmit Utilization
Gi0/1         1        1        1
...
Gi0/25        12       22       22
Gi0/26        10       16       16
```

```
Total Ports : 26
Switch Receive Bandwidth Percentage
Utilization : 1
Switch Transmit Bandwidth Percentage
Utilization : 2
```

Switch Fabric Percentage Utilization : 1

This seemed normal, but when checking for drops we noticed the hard truth:

```
video-switch-1#show mls qos interface statistics | i
GigabitEthernet|queue|dropped
...
GigabitEthernet0/26
  output queues enqueued:
  queue:  threshold1  threshold2  threshold3
  queue 0:      0          0          0
  queue 1:      0        150564      119978
  queue 2:      0          0          0
  queue 3:      0          0        1645256287
  output queues dropped:
  queue:  threshold1  threshold2  threshold3
  queue 0:      0          0          0
  queue 1:      0          0          0
  queue 2:      0          0          0
  queue 3:      0          0        7154647
```

Clearly we were dropping a number of packets (0.43% of packets) because we ran out of buffers on some queues. We tried to fix the problem using flow-control, but that was a mistake and it did not help. Trying to change the buffer allocation was not possible as these switches are limited in their QoS features. In the end we were unable to fix this problem without risking interrupting the traffic.

Designing, configuring and testing a proper QoS architecture and replacing the old switches which have served us well for the last 8 years with switches more adapted to these higher amounts of traffic, is an action point for the next year for us.

This year we used a more general http-user-agent analysis, so the client numbers are not directly comparable, but we detected the following client distribution: →

"Hardware: The parts of a computer system that can be kicked."

(Jeff Pesis)

"I've finally learned what 'upward compatible' means. It means we get to keep all our old mistakes."

(Dennie van Tassel)

OS	Clients
Android	4640
iOS	2579
Unknown	1038
Ubuntu	884
Linux	697
Debian	499
Other	447
Mac OS X	366
Fedora	223
Windows 10	105
Windows 7	98
Maemo	94
Windows	85
Windows XP	33
openSUSE	26
Windows 8.1	24
Windows 8	22
Firefox OS	12
Windows Phone	10
Windows 95	9
FreeBSD	8
OpenBSD	5
Solaris	3
Symbian OS	3
Windows Vista	2
NetBSD	1
Windows 98	1
Windows Mobile	1
MeeGo	1
Windows RT 8.1	1
SUSE	1

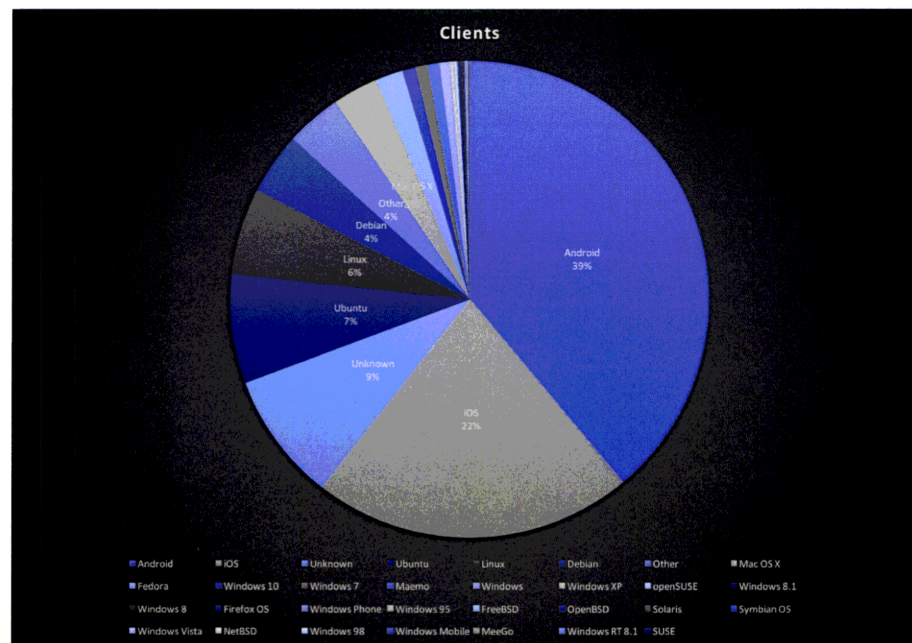
I'm really hoping that these machines running Windows 95 and friends were virtual machines or emulations.

See you all next year where we hope to be able to use telemetry instead of snmp/netflow!



Peter Van Eynde, CCIE 23042, is a Data Scientist at Cisco Systems Technical Assistance Center working on creating smarter, faster, easier ways to fix or prevent problems. He has been involved with networking so long he remembers using gopher. When he is not solving complex technical problems at TAC he is writing decoders in wireshark or utilities in Python, Clojure or Common Lisp.

Most of his time is spend fishing in the sea with Wiresharks.



Den store blå er Android, så kommer den grå iOS

Et brev fra redaktøren

Når man opdager at andre har tænkt samme tanke

Blandt foredragene fra FOSDEM var der én video-optagelse, som jeg meget gerne ville se, en beretning om hvordan man brugte PostgreSQL til en større database; **Bringing the Semantic Web closer to reality** med Jimmy Angelakos fra Edinburgh University. Ikke fordi jeg kendte manden, og heller ikke fordi jeg er familier med det semantiske web, jeg har en ide om hvad det er, men så heller ikke mere.

Semantiske data er i IT-forstand sammenknytning af X med Y.

En ting, en entitet, knyttes til et udsagn om denne ting (med en udsagnstype), og en værdi for udsagnet.

Altså:

Subjekt, prædikat, værdi - en slags grund-grammatik.

Fx. *Bodil er 42*, eller *Rita kender Linda*.

Det ligner også det man i filosofi og logik kalder et udsagn eller en dom.

Det ligner også det, vi kender fra objekt-orienteret Javascript som objekt, attribute og value. Det kaldes en tripel.

Hvert led i en tripel kan være repræsenteret af en URI - en henvisning til en anden ressource (Uniform Resource Identifier).

Det semantiske web er en efter min mening misvisende betegnelse for opdeling af web-indhold i kategorier, som har mange forskellige indbyrdes relationer, men det er meget praktisk. Hvorfor misvisende? jo, fordi computere ikke har nogen mening om noget som helst hvis ikke programmøren har podet eller kodet det.

Men tanken er selvfølgelig nyttig; fx. er hestesport og hestetransport-trailer tættere knyttet sammen end så meget andet, og der kan tegnes en linie mellem dem (deraf navnet *directed graph* for diagrammer, som viser komplekse sammenhæng), forholdet kan registreres og kan lette fx. maskinoversættelse.

Betegnelsen "semantisk web" er kort; alene dét er praktisk. Derfor lagde jeg min irritation over "semantisk" på hylden og lytter til, hvad Jimmy Angelakos har at fortælle:

PostgreSQL as RDF Graph Database

Presentation of an investigation into how Python's RDFLib and SQLAlchemy can be used to leverage PostgreSQL's capabilities to provide a persistent storage back-end for Graphs, and become the elusive practical RDF triple store for the Semantic Web (or simply help you export your data to someone who's expecting RDF)!

RDF står for Resource Description Framework; den består af de nævnte "tripler" med forbindelser, graphs (directed graphs), som viser forbindelserne, sammenhængen.

Det er klart at Angelakos forventer, at folk ved lidt om RDF; men det ville nu alligevel netop i en lecture være en god idé at holde en dør på klem for begyndere, som ikke lige med det samme kan se meningen med "maskinlæsbar information".

Jamen er alle websider da ikke maskinlæsbare?

Først digitaliserer man, altså, man repræsenterer dokumenter med en serie character codes. Det er da i højeste grad machine readable! (**undren!**) Men "semantisk" er abstraktion et trin over "character". Kategorisering i emneområder er god idé. Det bruges jo også allerede i de søgemaskiner, som gør livet lettere.

Machine readable er en "fattig" eller nød-betegnelse for at systemet, når dokumentet er læst, kan katalogisere indholdet og udtrække sammenhænge. Drømmen er at brugerne kan håndtere information mere effektivt. Altså ting som: "Denne webside handler om *heste*" og den slags meta-informationer.

Det, der var mest spændende, var at høre Angelakos fortælle om performance. Efter at have kørt nogle test, samlede hans team ikke bare tusindvis, men milliardvis af sådanne tripler op og puttede dem i en PostgreSQL database. På almindelig hardware! han kalder det "desktop maskineri", vi må forstå at det er x86_64-arkitektur med 32 GB ram og masser af disk - men ikke mere end at det kan købes for penge.

Semantic Web? RDF?

- Resource Description Framework
 - Designed to overcome the limitations of HTML
 - Make the Web machine readable
 - Metadata data model
 - Multigraph (Labelled, Directed)
 - Triples (Subject – Predicate – Object)



Bringing the Semantic Web closer to reality
PostgreSQL as RDF Graph Database

EDINA

Målet er at indhold af websider beskrives med triples i grupper

Særligt vågnede jeg ved Angelakos' bemærkning om at der findes kommercielle løsninger, som ivrige mennesker forsøgte at sælge til ham, men at disse mennesker glemmer at fortælle ham at de kommercielle systemer kører på dyr specialiseret hardware.

Ikke nok med det, de Open Source libraries, som håndterer RDF og sætter det i "lineær form", som så kan hældes i en database, er mindre og performer bedre end de dyrere løsninger.

Wow - det er godt at høre!

Jeg har ofte tænkt på at offentlige registre over biler eller ejendomme mv. ville kunne laves med Open Source tools og blive pænt hurtige med almindelige "desktop" maskiner. Den tanke vågner igen, når jeg hører Angelakos gøre rede for performance. Vil man virkelig opnå bedre performance end med kommercielle systemer? derved kunne man jo spare skatteyderne for millioner, og der ville blive råd til uddannelse og forskning i IT. Det ville være det bedste bidrag til Open Source: Uddannelse og god, fast ansættelse.

Open Source er nemlig ikke gratis. Der betales på en eller anden måde. Linux Foundation er et af eksemplerne, men også PostgreSQL har store sponsorer, oprindeligt fordi det var den eneste RDBM med support af asiatiske tegnsystemer.

Optimering i flere trin

Allerede i første omgang må Angelakos og hans team se på hvordan der genereres *tripler* og hvordan de håndteres af de to libraries, som skal hjælpe med at proppe dem i en PostgreSQL database:

Der bruges to libraries, rdflib og sqlalchemy, og de leverer SQL-statements i enkeltlinier:

```
INSERT INTO ... VALUES (...);
```

```
INSERT INTO ... VALUES (...);
```

```
INSERT INTO ... VALUES (...);
```

og det er hamrende ineffektivt.

Der anvendes to Python libraries, og det fungerer. Men i default opsætning dannes der indexet.

Team må derfor selv transformere SQL-statements til nogle mere effektive INSERT INTO ... VALUES (...),(...), hvor mange mange records indsættes i én omgang inden der åbnes for at andre kan bruge basen.

Det er også en fordel at droppe indices og danne dem igen bagefter.

Når det er sagt, så laver Angelakos ikke flere optimeringer, og det gør ham og hans team optimistiske med hensyn til at systemet vil kunne opnå endnu bedre performance i fremtiden.

Man vil som sagt kunne finde slides og foredrag på FOSDEM websitet, og desuden henviser Angelakos til

<http://labs.edina.ac.uk>

<http://github.com/vyruss/rdflib-sqlalchemy>

Alt i alt værd at høre

Selv om man ikke har tænkt sig at downloade en million web-sider og lave RDF-tripler og directed graphs som puttes i en PostgreSQL database, så er det et spændende foredrag, som berører væsentlige IT-tekniske problemstillinger. Filosofien bag RDF anvendelse står lidt uklart, men det forringer ikke værdien af foredragets indblik i performance og effektiv brug af hardware.



Knowledge and Expertise

Through Research and Innovation

[VIDEO](#) | [Simulating user load using JMeter](#)

03/08/2016

[Jimmy Angelakos](#)



[VIDEO](#) | [\(De\)?mystifying Regular Expressions](#)

13/07/2016

[Ben Soares](#)



[VIDEO](#) | [HTML5 Multimedia Streaming](#)

06/07/2016

[Niall Munro](#)



[Workshop with Jimmy](#)

[Angelakos at EDINA on 03/08/2016](#)

[Read more...](#)

[Presentation by Ben Soares at EDINA on](#)

[13/07/2016.](#)

[Read more...](#)

[Presentation by Niall Munro at EDINA on](#)

[06/07/2016.](#)

[Read more...](#)

Recent posts

[WORKSHOP](#) | [Static Site](#)

[Generators: what they are and when they are useful](#)

29/06/2016 | [Paul Walk](#)

[WORKSHOP](#) | [Data Munging with](#)

[Python](#)

08/06/2016 | [Jimmy Angelakos](#)

[WORKSHOP](#) | [Looking into the past](#)

[- feature extraction from historic maps using Python, OpenCV & PostGIS](#)

25/05/2016 | [James Crone](#)

FOSDEM video katalog

Listen over video af lectures ved FOSDEM-2017 er samtidig en liste over de vigtigste IT-emner

Når man ser bort fra velkomst-tale og afrunding, så berører de fire resterende keynotes kerneområderne i IT-udvikling:

1. Deployment af software på cluster (kort sagt) oprindeligt fra Google, **Kubernetes**,
2. Software arven - projektet *Software Heritage* indsamler Open Source projekter og arkiverer, med det formål at bevare kernen af Free Software (vidensdeling, fællesskab) for eftertiden,
3. Copyrigh (copyleft) forstået som beskyttelse af Open Source rettigheder, (retten til deling)
4. Linux som Air Traffic Control system (HW/SW) - ja det er jo et lidt specialiseret område, men problematikken er generel: Jeg så for snart 15 år siden et IT-system til indsamling og visning af radar-oplysninger, fly-transponder-oplysninger og check-in oplysninger fra gates, systemet kørte på Data General Køleskabsstore computersystemer, som ikke blev produceret mere, og softwaren kunne ikke "bare lige" flyttes til et andet system, så man var nødt til at købe gamle DG-maskiner. Desuden var både skærme og diske udgåede modeller, og man holdt så at sige systemerne i luften ved hjælp af kunstigt åndedræt indtil man kunne få en Unix/Linux baseret system op at køre.

FOSDEM (Free and Open Source Software Developers' European Meeting) er en almennyttig, ikke-kommerciel frivilligt organiseret europæisk event, fokuseret på Free Open Source Software (FOSS) udvikling.

Alle lectures/foredrag kan ses på video som kan downloades fra fosdem.org, søg efter video, full event-liste, på fosdem.org

Højt niveau

FOSDEM programmet har mange indforståede eller uforståelige forkortelser, hvad er fx. GAFAM? En by i Indien? Nej, det er **Google Apple Facebook Amazon Microsoft**, og cozy.io er et projekt som vil fremme decentralisering og give mindre og mellemstore virksomheder mulighed for at følge med ved at danne deres egen profilerede kundebase uden at gå til Google.

Nogle foredrag er mest for selvstændige erhvervsdrivende med en profil, som ikke er typisk for FOSDEM gæsterne, - men spændende er det. Vil jeg kunne leve af at tilbyde virksomheder ejerskab til deres egne data? Deres egen cloud?

For **Cozy** er også en "cloud" løsning for privat brug, kalender, email, backup, frys eller husk hvad du har set på websider - sharing af fotos gennem en android app osv.

Fosdem har speaks om det hele: Config, udrulning, udvikling, Open Java, distributioner, embedded systemer, Guile (et listesprog, LISP baseret på Gnu Scheme (som bruges i emacs)), geospatiale ting som Open Street Map mv. Go (programmeringssprog), big data, Internet of Things, LLVM compiler toolchain, MySQL friends, PostgreSQL i forskellige projekter (se nærmere omtale foregående sider, 16-17), Open Document, Open Game, Valgrind (memory debugging/profiling) og meget mere.

Desværre vil det tage ca. 50 arbejdsdage, 2 en halv måned, at høre alle foredragene.



Building (3)

Bazel <i>How to build at Google scale?</i>	Klaus Aehlig	K.1.105 (La Fontaine)	Saturday	14:00	14:50	webm mp4
Continuous Integration at a Distribution Level <i>Shepherding 30.000 packages to never break</i>	Martin Pitt	K.1.105 (La Fontaine)	Saturday	15:00	15:50	webm mp4
Continuous Localisation using FOSS tools <i>Building a fast responsive localisation process using open source tools</i>	Dwayne Bailey	K.1.105 (La Fontaine)	Saturday	16:00	16:50	webm mp4

Cloud (2)

CloudABI <i>Easily develop sandboxed apps for UNIX</i>	Ed Schouten	Janson	Saturday	12:00	12:50	webm mp4
Resurrecting dinosaurs, what can possibly go wrong? <i>How Containerised Applications could eat our users.</i>	Richard Brown	Janson	Saturday	13:00	13:50	webm mp4

Documentation (3)

Write a Better FM <i>Read The F* Manual? Maybe you need to write a better f* manual</i>	Rich Bowen	K.1.105 (La Fontaine)	Sunday	10:00	10:50	
Bridging the Gap between Legacy Docs and Modular Content	Robert Kratky	K.1.105 (La Fontaine)	Sunday	11:00	11:50	webm mp4
What open source and J.K. Rowling have in common <i>Importance of storytelling in open source projects</i>	Justin W. Flory	K.1.105 (La Fontaine)	Sunday	12:00	12:50	webm mp4

Miscellaneous (11)

The Veripeditus AR Game Framework <i>Enabling everyone to freely create Augmented Reality Games</i>	Dominik George, Eike Jesinghaus	K.1.105 (La Fontaine)	Saturday	11:00	11:50	
LoRaWAN for exploring the Internet of Things	Benedict Gaster (cuberoo)	K.1.105 (La Fontaine)	Saturday	12:00	12:50	webm

Uddrag af FOSDEM programmet

Kommandocentralen

Kør et GUI-program igennem PuTTY

PuTTY er en GUI indpakning af et terminal-program til remote login. (TTY=teletype, de første typer computer-terminaler) Med putty kan man også køre grafiske programmer fra remote på den lokale maskine på en sikker måde.

Installation fra pakkesystem til de fleste Linux-typer vil tilføje et menupunkt, men man kan også starte fra kommandolinien - simpelthen: `$ putty`

Kommandolinien er i sin basis blot en kommando-loop, som læser en linie, (kommando-linien, det, som man taster ind ved promptet i et terminal-vindue) analyserer, opstiller og udfører `exec()` hvis det iøvrigt kan lade sig gøre.

På Unix kaldte man denne funktion *shell*, en "skal" om computerens kerne, navne valgt i en tid, hvor atomkraft var nyt og spændende.

Men **ssh** - secure shell - er faktisk ikke en kommando-fortolker, den læser ikke kommandolinier, men skaber forbindelse til en anden computer (remote host) og starter derefter en login proces på remote, sender output fra den proces til et terminal-vindue, som åbnes på vores maskine, localhost.

Derefter - hvis man logger ind - får man en kommandofortolker, som typisk er **bash**, men kan også være **zsh** eller **ksh** eller **ash**.

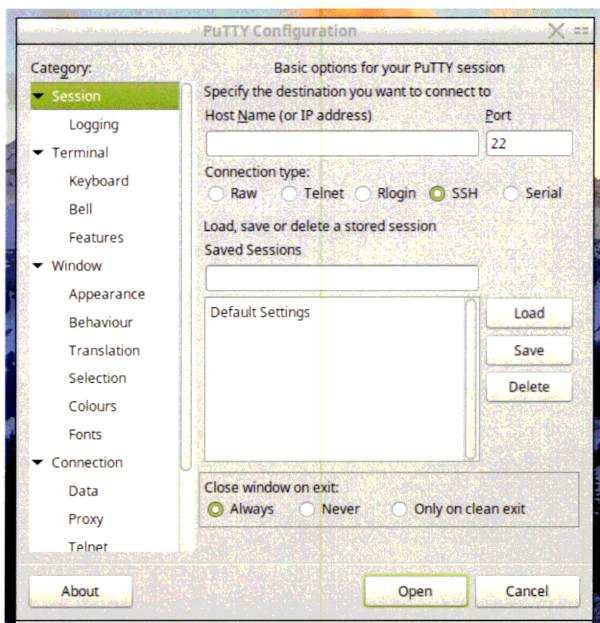
Putty er ("kun") en front-end for **ssh**, secure shell. Den forventer, at der kører en sshd (server) på remote host og starter en login proces som - forhåbentlig - starter en kommandofortolker.

X11 - forwarding

Går det hele bare ud på at få en kommandolinie? Nej ikke kun det - sidegevinsten er at man kan køre grafiske programmer fra remote på sin lokale desktop - et ad gangen, så at sige. Det er med andre ord ikke "remote desktop" men "remote program".

Konfiguration af forbindelsen er lettere med putty end med kommandolinie-options.

X11-Windows er meget modulært og har kunnet dette med at sende grafik-programmers data over nettet i mere end 25 år.



Bemærk, man kan godt lave en opsætning, så man straks får remote-forbindelsen i gang med automatisk login.

Default opsætningen fungerer, men du milde moster, hvor ser det sort ud. De ældste skermterminaler havde lysende forfargrønne bogstaver ("forgrundsfarve"). Med menupunkt Window -> Colours sætter man sine præferencer - fx.:

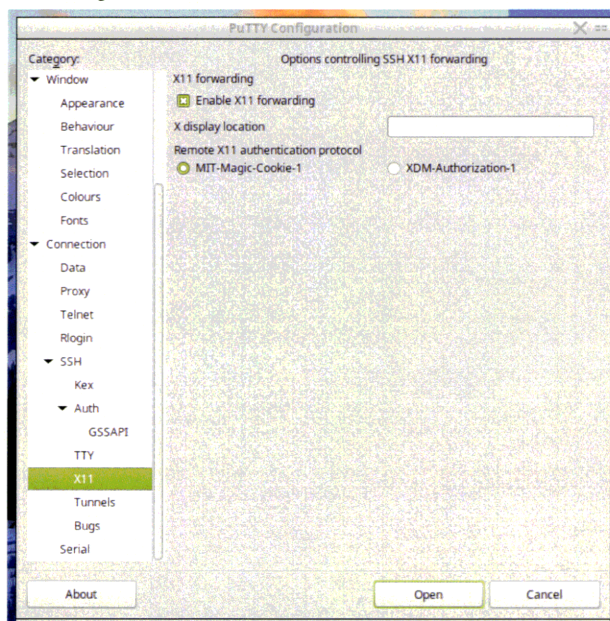
```
Jupiter-putty
login as: root
root@Jupiter's password:
Welcome to Linux Mint 17.2 Rafaela (GNU/Linux 3.13

Welcome to Linux Mint
* Documentation: http://www.linuxmint.com
No mail.
Last login: Thu Mar  2 18:21:57 2017 from sat2
Jup2 ~ # ksh
# ./ff
Term: xterm-color, Display: localhost:11.0
Jup2:/hjem/root #
```

Opsætningen er ændret på flere punkter, baggrund er gjort lys (brækket hvid) og forgrund mørkeblå, bold skal også have farver osv.

Men vigtigst er opsætningen af X11. Næstnederste linie viser, at display er localhost:11.0 - det betyder, at et GUI program, som vil tegne på skærmen skal skrive til skærm nr. 11. De data bliver opfanget af sshd som sender dem til vores putty, som videresender til vores fysiske skærm.

Man kan nemt overse X11 i config: det er en undermenu til **SSH**, som er et underpunkt til **Connection**, der kan krybe helt ned under vinduets underste kant. Find det og kryds af: Enable X11 forwarding.



Hvis det OGSÅ er gjort i remote hostens `/etc/ssh/sshd.conf` så kan man køre X11 programmer fra remote maskinen på localhost.

Det kan være praktisk fx ved monitorering af raid-systemer på en remote server. Det er typisk en softwarepakke, der skal køre grafisk - grafik-menuer kan være meget lettere at overskue.

Man kan prøve om det virker fx. `# xclock -bg red`, så får man et skrigende rødt vindue.

Husk at gå tilbage til "session" menupunktet og gem opsætningen inden etablering af forbindelsen, - på min Linux kunne jeg ikke få "session"-vinduet tilbage efter først at have etableret remote forbindelsen.



IT-sikkerhed

White Hat How To Anti Hack

Gennem de seneste år er presset omkring sikkerhed af IT-systemer vokset markant.

Både den offentlige og private sektor er blevet ramt af flere angreb, som enten har sat systemer ud af drift, stjålet følsomme data, eller "taget data til fange" med Ransomware-angreb.

Derfor er der stigende efterspørgsel på IT-sikkerhedsspecialister og uddannelse inden for området.

SuperUsers tilbyder en stribe af sikkerhedskurser, som øger dine kompetencer med både teori og hands-on erfaring med at hacke, teste og beskytte jeres systemer.

Malware

Scanning af netværk

Ransomware

Kryptering

Huller og Exploits

Virus

Sniffing

Social Engineering

DDoS angreb

IT Sikkerhed - Det store overblik

IT-sikkerhed er et kæmpe område som kan være svært at komme i gang med. Der er rigtig mange begreber og indgangsvinkler, og det hele kan virke meget uoverskueligt.

Det vil dette kursus at lave om på! I løbet af kort tid vil du komme hele vejen rundt om IT-sikkerhed, og få et samlet overblik over hvad man skal forholde sig til.

Kursus SU-400

Varighed: 1 dag

Etisk Hacker - Tænk som en hacker

På dette kursus kan du komme på forkant med hackerne ved selv at lære de værktøjer og huller, som typisk udnyttes. Ofte er hacking forbundet med noget ulovligt, og derfor hedder dette kursus netop "Etisk Hacker" - du lærer at tænke som en hacker, men bruger ikke din viden til ulovligt at udnytte huller i systemer. I stedet bruges denne viden til at gardere sig imod trusler.

Kursus SU-410

Varighed: 3 dage

Flere nye sikkerhedskurser:

Securing Windows Server 2016 // Linux Security // macOS Security // IoT Security

www.superusers.dk/sikkerhed | Altid 300 garanterede datoer | 50.000 kursister siden 1984

SUPERUSERS