



[Schema della lezione](#)

[Hacker](#)

[Free software](#)

[Nascita di Linux](#)

Lezione OS2

Il software libero

Sistemi operativi open-source, embedded e real-time

12 ottobre 2017

Marco Cesati

Dipartimento di Ingegneria Civile e Ingegneria Informatica
Università degli Studi di Roma Tor Vergata

Di cosa parliamo in questa lezione?



Schema della lezione

Hacker

Free software

Nascita di Linux

Il movimento del software libero

- 1 La quarta generazione di calcolatori
- 2 La cultura degli hacker
- 3 Il movimento del software libero
- 4 La nascita del kernel Linux



Schema della lezione

Hacker

Free software

Nascita di Linux

- Generalmente l'origine del movimento degli **hacker** è attribuita agli studenti dei primi anni '50 del **MIT** (**M**assachussets **I**nstitute of **T**echnology)
- Il termine **hack** aveva il significato di *scherzo goliardico* o altra attività studentesca non legata allo studio
- Verso la fine degli anni '50 il termine **hack** assume una connotazione più ribelle, per reazione ad un ambiente autoritario e competitivo
 - **Tunnel hacking**: esplorazione delle innumerevoli gallerie sotterranee del MIT
 - **Phone hacking**: scherzi basati sul'(ab)uso del sistema telefonico (→ **phreaking**)
- Era bandita ogni attività che potesse essere malevola, dolorosa o dannosa

Il movimento degli hacker (2)

- Negli anni '50 e '60 gli studenti continuarono a chiamare **hacking** le attività che erano sia divertenti che “esplorative”
- Gli studenti del MIT appassionati di modellismo ferroviario avevano fondato il **Tech Model Railroad Club**
- All'interno del club, una ristretta élite di studenti formava il comitato **Signals and Power**:
 - Si occupava del circuito elettrico della ferrovia in miniatura
 - Aveva collegato il circuito elettrico al sistema telefonico, quindi era possibile comandare i modellini tramite i telefoni
 - Per poter riutilizzare i componenti elettrici cercava di realizzare i circuiti con il minimo di componenti ed il massimo dell'efficienza → nuovo significato di **hacking**
- Verso la fine degli anni '50 i membri del comitato **Signals and Power** cominciano ad interessarsi di informatica:
 - Vorrebbero lavorare con un mainframe IBM 704 in possesso dell'MIT, ma non riescono ad avvicinarsi
 - Gli viene invece concesso di “giocare” con un prototipo da \$ 3 000 000: un calcolatore a transistor chiamato **TX-0**



Nascita dei computer hacker



Source: Matthew Hutchinson, www.flickr.com

- Nel 1959 la Digital Equipment Corp. (DEC) costruisce il primo **minicomputer**: il **PDP-1** (**P**rogrammed **D**ata **P**rocessor), basato su ricerche condotte al MIT Lincoln Lab (**TX-0**)
- Nel 1962 DEC donò il prototipo del **PDP-1** all'MIT

- Il nuovo mini-calcolatore attira ovviamente l'interesse degli **hacker**, che cominciano a modificare il suo software e a scrivere nuovi programmi
- Un gruppo di loro, tra cui Steve Russell, nel 1962 completa la prima versione del gioco **Spacewar!**



Source: Jai Ito, www.flickr.com



Nascita dei computer hacker (2)

Negli anni '60 e '70 il movimento degli [hacker](#) al MIT coinvolse studenti e professori, soprattutto nel Artificial Intelligence Lab

Rispetto agli [hacker](#) degli anni '50:

- Interesse focalizzato sui computer
- Le attività non sono più segrete e ristrette a piccoli gruppi
- Al contrario vengono incoraggiati:
 - lo spirito di collaborazione
 - la condivisione dei risultati del lavoro
- Produssero:
 - tanto software (tra l'altro il codice per ARPANET)
 - un proprio *slang* (sintetizzato nel [JARGON.TXT](#) file)
 - propri ideali e principi





[Schema della lezione](#)

[Hacker](#)

[Free software](#)

[Nascita di Linux](#)

- Access to computers — and anything which might teach you something about the way the world works — should be unlimited and total. Always yield to the Hands-On Imperative!
- All information should be free.
- Mistrust authority — Promote decentralization.
- Hackers should be judged by their hacking, not criteria such as degree, age, race, sex, or position.
- You can create art and beauty on a computer.
- Computers can change your life for the better.

S. Levy, *Hackers: Heroes of the Computer Revolution*, Penguin 1984.

Quando si acquista un calcolatore elettronico, si assume la proprietà della macchina, ed il costo pagato è proporzionale al valore della tecnologia di cui si viene in possesso



Quando si paga per il software **non** si assume la proprietà del codice

Schema della lezione

Hacker

Free software

Nascita di Linux

- Legalmente il software è generalmente trattato allo stesso modo delle altre opere dell'ingegno (opere letterarie, musicali, ...)
- Alla base c'è la nozione di **diritto d'autore (copyright)**
 - garantisce il riconoscimento di un giusto compenso all'autore o detentore dei diritti dell'opera
- In genere il software non viene acquistato, nel senso che non vengono trasferiti i diritti sull'opera (ossia il **copyright**)
- Ciò che si paga è la **licenza d'uso**, ossia il permesso di utilizzare il codice alle precise condizioni stabilite dal detentore dei diritti

- In molti paesi esiste anche un'altra possibilità per proteggere il software: il **brevetto** (**patent**)
- Il **brevetto** è un meccanismo legale che impedisce l'utilizzo di una invenzione tecnologica non autorizzato da parte dell'inventore o del detentore dei diritti sull'invenzione
- Il **brevetto** dovrebbe essere limitato alla protezione di meccanismi tecnologici “non banali” e realmente “innovativi”
- Nel caso del software, in pratica gli uffici brevetti hanno rilasciato milioni di brevetti banali e/o non innovativi
- Ogni azienda di IT possiede un proprio *portafoglio di brevetti* che usa come arma offensiva o difensiva contro le altre aziende di IT
- A complicare le cose, alcune aziende comprano brevetti unicamente per poter “estorcere” il pagamento di licenze alle grandi aziende di IT



Il movimento del software libero



Schema della lezione

Hacker

Free software

Nascita di Linux

- Agli inizi degli anni '80 l'idea di software diffusa tra gli hacker del MIT viene messa in crisi da
 - le cause nei tribunali su Unix
 - una nuova legge sul copyright del 1976
 - lo sviluppo di una economia basata sul valore del software piuttosto che dell'hardware
- Sul software vengono sempre più spesso poste licenze restrittive che impediscono di avere accesso al codice sorgente
- Molti ricercatori del MIT lasciano il lavoro per andare a lavorare in aziende di software
- Questa situazione colpisce in modo particolare un ricercatore del MIT, Richard Stallman
 - Hacker di seconda generazione, conosciuto come “rms”
 - Carattere intransigente
 - Propenso ad azioni di protesta clamorose
 - Eccellente programmatore

The Last True Hacker: rms



Source: archive.computerhistory.org

Il software libero

Marco Cesati



Schema della lezione

Hacker

Free software

Nascita di Linux

Il movimento del software libero (2)

- Nel 1983 Richard Stallman avvia il **Progetto GNU** (GNU's **N**ot **U**nix)



- Scrivere un intero SO “libero” da diritti d’autore e licenze
- Compatibile con Unix
- Quasi completato nei primi anni '90: manca solo il nucleo del SO
- Il kernel GNU (**Hurd**) ad oggi non è stato ancora completato
- Stallman non poteva utilizzare un kernel Unix BSD:
 - Esisteva la possibilità che qualche azienda in futuro potesse rivendicare diritti sul progetto GNU
 - La licenza BSD permette di includere il codice in prodotti commerciali “closed-source”
- In ogni caso il progetto **GNU** riscuote un grande successo
 - I programmi **GNU** vengono installati sui SO proprietari perché ritenuti più efficienti, robusti e consistenti dei corrispondenti tool proprietari



Schema della lezione

Hacker

Free software

Nascita di Linux

Il movimento del software libero (3)

- Nel 1985 Stallman fonda la **Free Software Foundation (FSF)**
 - Promuove scrittura e diffusione del software libero
- Nel 1989 Stallman scrive la prima versione della licenza **GNU GPL (General Public License)**
 - La seconda versione è del 1991, la terza vers. è del 2007
 - È la più diffusa licenza utilizzata per il **software libero**
- Il numero di progetti basati sul concetto di software libero è enorme



Le motivazioni di Stallman e dei proponenti del **software libero** sono primariamente di ordine “etico” e a salvaguardia dei diritti degli utenti



Schema della lezione

Hacker

Free software

Nascita di Linux

Le libertà fondamentali degli utenti

Secondo Stallman, il **software libero** (**free software**) garantisce quattro **libertà fondamentali** degli utenti dei calcolatori

- 1 Libertà di eseguire il programma per qualsunque scopo
- 2 Libertà di leggere e studiare il codice sorgente
- 3 Libertà di ridistribuire copie del programma originale
- 4 Libertà di migliorare e adattare alle proprie esigenze il programma e distribuire le versioni così modificate

La licenza **GNU GPL** garantisce queste libertà fondamentali

- Chi modifica un programma **GPL** deve rilasciarlo con la stessa licenza **GPL** \implies l'utente del programma modificato ha gli stessi diritti di chi ha fatto le modifiche
- La **GPL** non esclude che il programmatore possa essere retribuito per il suo lavoro
- Se il codice è utilizzato entro un programma diverso (*prodotto derivato*), anche questo ricade sotto la **GPL**
- La licenza **GNU GPL** è un esempio di software "*copyleft*"

Tipologie di software

- **Pubblico dominio (public domain)**: software senza **copyright**, ossia senza detentore di diritti (può essere “rubato” e venduto come software proprietario)
- **Copyleft** (“permesso d’autore”): software il cui detentore dei diritti acconsente alla sua distribuzione secondo i principi del software libero e senza restrizioni aggiuntive (es.: licenza **GNU GPL**)
- **Non copyleft**: software il cui detentore dei diritti acconsente alla modifica e redistribuzione anche con restrizioni aggiuntive (es.: licenza **BSD**)
- **Semi-libero**: software con licenza che permette l’uso, la copia, la modifica e la distribuzione con qualunque scopo eccetto quello di trarne profitto
- **Proprietario**: software non libero né semi-libero; generalmente viene ceduto a pagamento solo il codice eseguibile, che non può essere redistribuito liberamente
- **Freeware**: software in formato eseguibile a titolo gratuito
- **Shareware**: software in formato eseguibile che può essere redistribuito ma per cui è richiesta una licenza d’uso dopo un periodo di prova



Open Source Initiative (OSI)

- 1998: B. Perens e E.S. Raymond fondano la [Open Source Initiative](#)
- Nasce per presentare al mondo industriale e commerciale una visione meno radicale ed idealistica di quella della [Free Software Foundation](#)
- Lo scopo è quello di promuovere software “aperto” che l’utente finale possa studiare ed adattare
- La definizione di [software libero](#) e la definizione di [software open source](#) non coincidono esattamente
 - Esistono licenze [free](#) non [open source](#), e viceversa
 - In pratica la stragrande maggioranza dei progetti di [software libero](#) soddisfa i requisiti dell’[open source](#), e viceversa
- Idea: la disponibilità del codice sorgente del software
 - aiuta a creare un prodotto migliore grazie allo sviluppo distribuito, ai processi trasparenti ed alla “peer review”
 - evita alle aziende i meccanismi di “vendor lock-in”
 - può essere commercialmente vantaggioso





Schema della lezione

Hacker

Free software

Nascita di Linux

- Agosto 1991: appare un nuovo SO per PC Intel 80386

From: torvalds@klaava.Helsinki.FI

(**Linus Benedict Torvalds**)

Newsgroups: comp.os.minix

Subject: What would you like to see most in minix?

Summary: small poll for my new operating system

Date: 25 Aug 91 20:57:08 GMT

Organization: **University of Helsinki**

Hello everybody out there using minix -

I'm doing a (free) operating system (**just a hobby, won't be big and professional like gnu**) for **386(486) AT clones**. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat [...]



Schema della lezione

Hacker

Free software

Nascita di Linux

- **MINIX** è un SO simile a Unix
- Creato dal prof. Andrew Tanenbaum nel 1987 come ausilio didattico per un testo di SO
- Sviluppato per l'IBM PC e per l'IBM PC/AT
- Il codice sorgente era disponibile per scuole e università
- La licenza d'uso prevedeva un costo incluso nel prezzo del libro di SO di Tanenbaum
- In effetti Tanenbaum manteneva uno stretto controllo sul codice sorgente
- Inoltre rifiutava di accettare modifiche che complicassero troppo il SO
- Nel 2000 la licenza di **MINIX** è stata modificata, ed è ora molto permissiva