

Breve historia de la «cultura hacker»*

Eric S. Raymond

Prólogo: Los Programadores Auténticos

En el principio había Programadores Auténticos.

No se hacían llamar así. Tampoco “hackers”, ni de otra manera en particular: el apodo de “Programadores Auténticos” no se acuñó hasta 1980. Pero de 1945 en adelante la tecnología de la computación atrajo a muchas de las mentes más brillantes y creativas del mundo. Desde la ENIAC⁽¹⁾ de Eckert y Mauchly existió una cultura técnica de programadores entusiastas ininterrumpida y consciente de sí misma, gente que creaba y jugaba con el software para divertirse.

Los “Programadores Auténticos” venían en general de la física y la ingeniería. Vestían medias blancas y chombas de poliéster, y corbatas y anteojos gruesos, y escribían programas en lenguaje de máquina y assembler y FORTRAN y en media docena de lenguajes hoy olvidados. Estos fueron los precursores de la cultura hacker, los protagonistas de su prehistoria a los que las tradiciones ya no recuerdan.

Desde el fin de la Segunda Guerra Mundial hasta comienzos de los ‘70, en los grandes días de la computación batch⁽²⁾ y de las grandes centrales “de hierro”, los Programadores Auténticos fueron la cultura técnica dominante de la computación. Algunos fragmentos de folklore hacker de culto provienen de esta época, incluyendo la conocida histo-

* “Brief History of Hackerdom”. Traducción de Carlos Gradin.

1. La primera computadora fabricada en 1945 en la Universidad de Pennsylvania, Estados Unidos. Ocupaba una habitación y pesaba treinta toneladas. (N. del T.)

2. La época de las computadoras que sólo podían procesar información y no interactuaban con los usuarios en tiempo real. (N. del T.)

ria de Mel (incluida en la Jargon File)⁽³⁾, muchas Leyes de Murphy y el póster de “Blinkenlights” de parodia nazi⁽⁴⁾ que aún hoy decora muchos salones de computadoras.

Algunos de los que se iniciaron en la cultura de los “Programadores Auténticos” siguieron activos en los ‘90. Se dice que Seymour Cray, el diseñador de la línea de súper computadoras Cray, una vez puso en marcha un sistema operativo escrito por él mismo en una computadora armada por él mismo. En octal. Sin errores. Y funcionó. Programador Auténtico de Elite.

A otro nivel, Stan Kelly-Bootle, autor de *The Devil’s DP Dictionary* (New York, McGraw-Hill, 1981) y gran folklorista, programó en la Manchester Mark I, la primera computadora digital totalmente operativa y con un programa almacenado en la memoria, en 1948. Hoy escribe columnas humorísticas sobre tecnología para revistas de computación que suelen adoptar la forma de conversaciones intensas y sabias con la cultura hacker.

Otros, como David E. Lundstrom, han escrito la historia de esos primeros años y sus anécdotas (*A Few Good Men From UNIVAC*. Cambridge, MIT Press, 1987).

Lo que introdujo la cultura de los “Programadores Auténticos” fue el despertar de la computación interactiva, las universidades y las redes. Estas dieron nacimiento a una ininterrumpida tradición de ingeniería y

programas que, eventualmente, daría lugar a la actual cultura hacker de código abierto.

3. La Jargon File (Archivo de Argot, o Jerga) es una recopilación armada colectivamente por hackers de términos que utilizan para hablar de su trabajo, y que también incluye historias y anécdotas de hackers famosos. La cantidad de nuevas expresiones, metáforas y juegos de palabras que contiene, descritas e historizadas con verdadero amor filológico, podrían llenar las páginas de un diccionario mediano. Su origen se remonta a principios de los '70, cumpliendo el rol de herencia común del ámbito hacker. Su editor actual es Eric S. Raymond (<http://www.catb.org/~esr/jargon/>). (N. del T.)

4. Un póster colgado en muchos laboratorios de computación, que parodiaba a los avisos de advertencia nazis (ver Jargon File). (N. del T.)

BREVE HISTORIA DE LA "CULTURA HACKER"

Los Primeros Hackers

Los inicios de la cultura hacker como la conocemos hoy pueden fijarse en 1961, el año en que el MIT [Instituto de Tecnología de Massachusetts] adquirió la primera PDP-1⁽⁵⁾ El Club de Tecnología de Trenes a Escala del MIT adoptó la máquina como su juguete favorito e inventó herramientas de programación, argot y toda una cultura a su alrededor que todavía sigue entre nosotros. Estos primeros años se narran en la primera parte del libro Hackers de Steven Levy (New York, Bantam Books, 1984).

La cultura informática del MIT parece ser la primera en adoptar el término "hacker". Los hackers del Club de Modelos de Trenes se convirtieron en el núcleo del Laboratorio de Inteligencia Artificial (IA) del MIT, el centro líder mundial en investigaciones sobre IA a principios de los '80. Su influencia se extendió sobre todo a partir de 1969, el primer año de la red ARPA.

La red ARPA fue la primera red de computadoras transcontinental de alta velocidad. Fue creada por el Departamento de Defensa como un experimento sobre comunicaciones digitales, pero fue creciendo hasta conectar a cientos de universidades, laboratorios de investigación e industrias armamentísticas. Permitió a los investigadores de todas partes intercambiar información a una velocidad y flexibilidad sin precedentes, dándole un impulso enorme a los trabajos en colaboración e incrementando tremendamente el ritmo y la intensidad del avance tecnológico.

Pero la red ARPA también hizo otra cosa. Sus autopistas electrónicas pusieron en contacto a hackers de todo EE.UU. creando una masa crítica; en lugar de permanecer aislados en pequeños grupos, cada uno desarrollando sus propias culturas locales y efímeras, se descubrieron (o se reinventaron) como tribu unida en red.

Los primeros artefactos deliberados de la cultura hacker –las primeras antologías de argot, las primeras sátiras, las primeras discusiones conscientes sobre ética hacker– se propagaron por la red ARPA en los

5. Minicomputadoras que eran versiones reducidas de las viejas centrales, pero aún de tamaño considerable, similar al de una heladera, (N. del T.)

primeros años (la primera versión de la Jargon File, por ejemplo, data de 1973). La cultura hacker fue creciendo en las universidades conectadas a la Red, en especial (aunque no exclusivamente) en sus departamentos

de ciencia.

En cuanto a técnica, el laboratorio de Inteligencia Artificial del MIT estaba por encima de los demás a fines de los '60. Pero el Laboratorio de Inteligencia Artificial de la Universidad de Stanford (SAIL) y (luego) la Universidad de Carnegie-Mellon (CMU) devinieron casi igual de importantes. Todos eran centros florecientes de ciencia informática e investigación de IA. Todos atrajeron a brillantes personas que le aportaron a la cultura hacker grandes cosas, en ambos niveles, técnico y folklórico.

Pero para comprender lo que vino después tenemos que echarles otra mirada a las propias computadoras, porque el auge de los laboratorios y su eventual caída se debieron ambos a olas de cambios en la tecnología de las máquinas.

Desde los días de la PDP-1 la suerte de la cultura hacker se vio unida a la serie de minicomputadoras PDP de DEC (Corporación de Equipos Digitales). DEC era pionera en computación comercial interactiva y sistemas operativos de recursos compartidos. Debido a que sus máquinas eran flexibles, poderosas y relativamente baratas para la época, muchas universidades las compraban.

Los recursos compartidos fueron el medio en el que floreció la cultura hacker y, durante la mayor parte de su existencia, la red ARPA fue básicamente una red de máquinas DEC. La más importante de éstas fue la PDP-10, de 1967. Las 10 fueron las máquinas favoritas de la cultura hacker por casi quince años; TOPS-10 (el sistema operativo de DEC) y MACRO-10 (su lenguaje de programación) se recuerdan todavía con nostalgia; los dos produjeron abundante argot y folklore.

El MIT, pese a usar las mismas PDP-10 que los demás, tomó un camino ligeramente distinto; descartaron todo el software de DEC para la PDP-10 y se dedicaron a diseñar su sistema operativo propio, el mítico ITS.

ITS quería decir: “Sistema de Recursos Compartidos No Compatible”,⁽⁶⁾ lo que da una idea aproximada de su actitud. Lo querían

6. O sea, de Recursos Compartidos como otros sistemas operativos, pero No-Compatible, por ser mucho mejor que los demás. (N. del T.)

hacer a su manera. Por suerte para todos, la gente del MIT era inteligente además de arrogante. ITS, a pesar de ser excéntrico y complicado, y de estar, a veces, cargado de errores, aportaba una serie de innovaciones técnicas brillantes y seguramente mantiene todavía el récord para el sistema de recursos compartidos vigente por mayor tiempo.

ITS estaba escrito en assembler,⁽⁷⁾ pero muchos proyectos de ITS se escribían en un lenguaje con elementos de Inteligencia Artificial llamado LISP. LISP era más poderoso y flexible que los demás lenguajes de la época; de hecho, posee un diseño superior a la mayoría de los lenguajes de la actualidad, después de veinticinco años. LISP les dio libertad a los hackers de ITS para pensar de maneras creativas e inusuales. Fue un factor importante en sus éxitos, y sigue siendo un lenguaje favorito de la cultura hacker.

Muchas creaciones técnicas de la cultura de ITS hoy siguen vivas; el editor de programas Emacs⁽⁸⁾ quizás sea la más conocida. Y mucho del folklore de ITS sigue “vivo” para los hackers, como se puede ver en la Jargon File.

SAIL (Stanford) y CMU (Carnegie-Mellon) no se quedaron quietas,

tampoco. Muchos cuadros hackers que se formaron en torno a las PDP-10 de SAIL se convirtieron después en figuras clave del desarrollo de las computadoras personales y las interfaces actuales tipo ventana/ícono/mouse. Y los hackers de CMU hacían experimentos que conducirían a las primeras aplicaciones prácticas a gran escala de sistemas expertos y robótica industrial.

Otra zona importante de la cultura era Xerox PARC, el famoso Centro de Investigaciones de Palo Alto. Por más de una década, desde principios de los '70 hasta mediados de los '80, PARC entregó un volumen sorprendente de innovaciones revolucionarias de hardware y software. El estilo moderno de las interfaces con mouse, ventanas e íconos se inventó allí. También las impresoras láser y las redes locales; la serie de computadoras D de PARC se anticipó una década a las poderosas computadoras personales de los '80. Lamentablemente, estos profetas no eran escuchados en su propia compañía; tanto es así que se vol-

7. Lenguaje de bajo nivel, muy técnico y difícil de leer y escribir. (N. del T.)

8. El creador de este programa fue Richard Stallman. (N. del T.)

vió un chiste repetido describir a PARC como un lugar dedicado a desarrollar ideas brillantes para que las usaran otros. Su influencia en la cultura hacker fue amplia.

Las culturas de la red ARPA y PDP-10 aumentaron su vigor y variedad a lo largo de los '70. Los sistemas de listas de correo electrónico que se habían usado para nutrir la cooperación entre grupos de interés dispersos por los continentes, se usaban cada vez más con propósitos sociales y recreativos. En ARPA se hizo la “vista gorda” a toda actividad técnica “no autorizada” –la sobrecarga de datos era un precio menor a cambio de atraer al campo de la informática a una generación de jóvenes brillantes.

De las listas de mails “sociales” de la red ARPA, la más popular debía ser la SF-LOVERS (Amantes de la Ciencia Ficción); de hecho, sigue funcionando hoy dentro de la red más extensa llamada “Internet”, la cual absorbió a la red ARPA. Pero había muchas otras, que inauguraban un estilo de comunicación que más tarde sería explotado comercialmente por servicios de recursos compartidos como CompuServe, GENIE y Prodigy.

El surgimiento de UNIX

Mientras tanto, sin embargo, en un salvaje lugar de Nueva Jersey otra cosa había estado sucediendo desde 1969 que eventualmente opacaría la tradición de la PDP-10. El año del nacimiento de la red ARPA fue también el año en que un hacker de los laboratorios Labs llamado Ken Thompson inventó el sistema operativo UNIX.

Thompson había estado involucrado en el desarrollo de un sistema operativo de recursos compartidos llamado Multics, que tenía orígenes comunes con ITS. Multics fue un espécimen de prueba para algunas ideas sobre cómo la complejidad de un sistema operativo se podría ocultar en su interior, imperceptible para el usuario e incluso para la mayoría de los programadores. La idea era lograr que usar Multics desde afuera (¡y programar para él!) se tornara más sencillo, para que pudiera realizarse más trabajo concreto.

Bell Labs abandonó el proyecto cuando Multics mostró señales de estar mutando en un inservible elefante blanco (el sistema fue puesto a la venta luego por la empresa Honeywell pero nunca tuvo éxito). Ken Thompson se quedó sin el entorno Multics, y empezó a hacer pruebas con una mezcla de sus ideas y otras propias en una DEC PDP-7 rescatada de la basura.

Otro hacker llamado Denis Ritchie inventó un nuevo lenguaje llamado "C" para usar bajo el UNIX embrionario de Thompson. Al igual que UNIX, C fue diseñado para ser amable, ligero y flexible. En Bell Labs se despertó el interés por estas herramientas, y recibieron un impulso en 1971 cuando Thompson y Ritchie ganaron una convocatoria para producir lo que hoy llamaríamos un sistema de automatización de oficina para uso interno. Pero Thompson y Ritchie tenían en mente algo más grande.

Tradicionalmente, los sistemas operativos se escribían en lenguaje assembler ajustado para extraer la eficiencia más alta de las computadoras que los albergaban. Thompson y Ritchie fueron de los primeros en darse cuenta de que la tecnología de hardware y de compiladores se había vuelto lo suficientemente buena para poder escribir un sistema operativo enteramente en C, y para 1974 el sistema completo había sido transportado con éxito a diversas computadoras de diferentes tipos. Esto no se había hecho nunca, y las implicancias fueron enormes. Si Unix podía mostrar el mismo aspecto, las mismas características, en computadoras diferentes, podía servir como entorno de software común para todas ellas. Los usuarios no tendrían que pagar nunca más por el diseño del software cada vez que cambiaban de computadora. Los hackers podrían llevar consigo sus herramientas entre diferentes máquinas, en lugar de tener que reinventar el fuego y la rueda en cada ocasión.

Además de la portabilidad, Unix y C tenían otros aspectos ventajosos. Ambos estaban diseñados según una filosofía de "¡Hazlo simple, estúpido!". Un programador podía manejar en la cabeza toda la estructura lógica del C (a diferencia de la mayoría de los lenguajes anteriores) en vez de tener que consultar los manuales todo el tiempo; y UNIX se estructuraba como un kit de programas simples diseñados para combinarse entre sí de maneras productivas.

Esta combinación mostró ser adaptable a una gama muy amplia de tareas informáticas, incluyendo muchas totalmente imprevistas por sus diseñadores. Se difundió con gran rapidez dentro de AT&T, pese a no contar con ningún apoyo formal de la empresa. Para 1980 se había difundido por un gran número de universidades y sitios de investigación informática, y miles de hackers lo consideraban su hogar.

Las locomotoras de la cultura UNIX inicial fueron las PDP-11 y sus descendientes, las VAX. Pero debido a la portabilidad de UNIX, éste corría sin modificaciones esenciales en una diversidad de computadoras mayor de las que se podía encontrar en toda la red ARPA. Y nadie usaba assembler; los programas en C se traspasaban al instante entre esas computadoras.

Unix tenía incluso su propia red, de tipo UUCP (Protocolo de Transferencia Unix a Unix): lenta e inestable, pero barata. Dos máquinas Unix podían intercambiar correo electrónico punto-a-punto por la línea de teléfono común; esta opción era parte del sistema, no un agregado

especial. Los sitios Unix comenzaron a formar en sí una nación en red, y una cultura hacker la acompañó. En 1980, aparece el primer nodo Usenet que pronto se extendería hasta superar a ARPA.

En la red ARPA había pocos sitios Unix. Las culturas de las PDP-10 de ARPA y la de Unix comenzaron a encontrarse y confluir en los bordes, pero no combinaron bien al principio. Los hackers de la PDP-10 tendían a considerar a los de Unix como una banda de aficionados, que usaba herramientas de aspecto tosco y primitivo si se las comparaba con las barrocas y adorables complejidades de LISP e ITS. “¡Cavernícolas!” –les decían.

Y además había una tercera corriente avanzando. La primera computadora personal había salido al mercado en 1975. Apple se fundó en 1977, y en los años siguientes se produjeron avances a una velocidad casi increíble. La potencialidad de las microcomputadoras⁹ era evidente, y atrajo a una nueva generación de jóvenes y brillantes hackers. Su lenguaje era el BASIC, tan primitivo que tanto los partisanos de la PDP-10 como los aficionados de Unix lo despreciaban.

9. Las nuevas computadoras personales (Apple, PC, Spectrum, Commodore, etc.) (N. del T.)

El Fin de los Viejos Tiempos

Así estaban las cosas en 1980: tres culturas, que se superponían en los bordes pero que estaban organizadas en torno a tecnologías muy diferentes: a) la cultura de la red ARPA/PDP-10, fiel a LISP y MACRO y TOPS-10 e ITS. b) La gente de Unix y C con sus PDP-11's y sus VAX y sus lentas conexiones telefónicas. Y c) una horda anárquica de aficionados a las computadoras chicas y baratas dedicados a darle a la gente el poder de la computación.

Entre éstas, la cultura de ITS todavía podía reivindicarse como “la original”. Pero se cernían nubes de tormenta sobre el MIT. La tecnología de las PDP-10 de la que ITS dependía estaba envejeciendo, y el Laboratorio mismo se dividió en departamentos para los primeros intentos de comercializar tecnologías de Inteligencia Artificial. Algunos de los mejores investigadores del MIT (y del SAIL y la CMU) partieron tras empleos y altos sueldos en empresas de innovación tecnológica. La estocada final se produjo en 1983, cuando DEC canceló sus planes a futuro para la PDP-10 para concentrarse en la PDP-11 y la línea VAX. ITS perdió su hábitat. Ya que no era portable, llevar el sistema ITS a las nuevas computadoras requería un esfuerzo que nadie estaba dispuesto a hacer. La versión de Berkeley de Unix corriendo en máquinas VAX se convirtió en el sistema por excelencia de los hackers, y cualquiera con cierta visión de futuro podía ver que las computadoras personales aumentaban su poder tan velozmente que barrerían con todo a su paso.

En esta época Levy escribió Hackers. Uno de sus mejores informantes fue Richard Stallman (inventor de Emacs), una figura líder en el Laboratorio del MIT y el vocero más fanático contra la venta comercial de su tecnología.

Stallman (a quien se conoce usualmente por sus iniciales y nombre de login, RMS) creó la Free Software Foundation [Fundación Software Libre] y se dedicó a escribir software libre de alta calidad. Levy lo des-

cribió como “el último hacker auténtico”, una descripción que por suerte se probó equivocada.

El gran estilo de trabajo de Stallman es un ejemplo claro de los cambios que atravesaba la cultura hacker a principios de los ‘80; en 1982 comenzó la construcción de un clon completo de Unix, escrito en C y disponible gratuitamente. Así, el espíritu y la tradición de ITS se preservó como parte importante de la renovada cultura hacker de Unix y las VAX.

Fue en esta época también que las tecnologías del microchip y las redes locales comenzaron a tener un impacto serio en la cultura hacker. Ethernet y los microchips 68000 de Motorola fueron una combinación muy potente, y varias empresas se habían creado para diseñar la primera generación de lo que ahora llamamos estaciones de trabajo. En 1982, un grupo de hackers del Unix de Berkeley fundaron Sun Microsystems con la idea de que Unix corriendo en computadoras con el chip 68000 –relativamente barato– sería una dupla ganadora para una amplia gama de aplicaciones. Tenían razón, y su visión estableció el standard para toda la industria. Aunque todavía estuvieran fuera del alcance de las personas comunes, las estaciones de trabajo eran baratas para las corporaciones y las universidades; las redes conectadas a ellas (una estación por usuario) remplazaron rápidamente a las viejas VAX y otros sistemas de recursos compartidos.

La Era del Unix Propietario

En 1984, cuando Unix se convirtió en un producto comercial por primera vez, la separación más importante en la cultura hacker lo constituían una “nación en red” relativamente unida, que había crecido en torno a Internet y Usenet (en su mayoría usando minicomputadoras o terminales que usaban Unix), y un archipiélago en gran medida desconectado de fanáticos de las computadoras personales desde sus casas. Con las máquinas tipo estación de trabajo⁽¹⁰⁾ de Sun y otras empresas se abrieron nuevos mundos para los hackers. Se habían diseñado para generar gráficos de alto rendimiento y compartir datos en redes. En los ‘80 la cultura hacker se concentró en sacar el mayor provecho de estas características por medio de software y el diseño de herramientas. La versión del Unix de Berkeley incorporó soporte para emplear la red ARPA, lo cual solucionó el problema de la conexión a la red y dio impulso al crecimiento de Internet.

Hubo varios intentos de domesticar los gráficos de las estaciones de trabajo. El que prevaleció fue el Sistema X Window. Un factor crucial de

10. Computadoras de alto rendimiento orientadas a funciones específicas (gráficos, cálculos complejos, etc.). (N. del T.)

su éxito fue que los creadores de X deseaban entregar gratuitamente las fuentes⁽¹¹⁾ del programa de acuerdo con la ética hacker, y estuvieron en condiciones de distribuirlas en Internet. Este triunfo de X Window sobre los sistemas de gráficos propietarios (incluido uno ofrecido por la propia Sun) sentó un precedente importante de cambios que, pocos años después, afectarían profundamente al mismo Unix.

Quedaba en el ambiente un poco de melancolía que se sacaba a relucir cada tanto en la rivalidad entre ITS y Unix (sobre todo del lado

de los de ex-ITS). Pero la última máquina ITS dejó de funcionar para bien en 1990; a los fanáticos no les quedó otra que aceptar la derrota y asimilarse a la cultura Unix, aunque sea a regañadientes.

En la cultura hacker conectada en red, la gran rivalidad de los '80 la protagonizaron los fans de las versiones Berkeley y AT&T de Unix. A veces todavía se encuentran copias de un póster de esa época, que muestra una nave X-wing de Star Wars huyendo de una Estrella de la Muerte con el logo de AT&T que vuela en pedazos. Los hackers de Berkeley se veían a sí mismos como rebeldes contra los mezquinos imperios corporativos. El Unix de AT&T nunca igualó al de Berkeley/Sun en ventas, pero ganó la guerra por los standards. Para 1990, se hacía difícil distinguir las versiones de Berkeley y AT&T, pues habían adoptado muchas innovaciones una de la otra.

Al empezar los '90 la tecnología de estaciones de trabajo de la década anterior empezaba a verse severamente asediada por nuevas computadoras personales (PC's), baratas y de alto rendimiento basadas en el chip Intel 386 y sus sucesores. Por primera vez, los hackers podían acceder en su casa a una computadora comparable en poder y almacenamiento a las minicomputadoras de diez años atrás –sistemas Unix capaces de sostener un entorno de desarrollo completo y comunicarse con Internet.

El mundo del MS-DOS ni se enteró de esto. Aunque aquellos primeros entusiastas de las computadoras personales se expandieron rápidamente a una población de hackers de DOS y Mac en una magnitud mayor que los de la cultura de la “nación en red”, nunca se convirtieron

11. El texto original de un programa escrito por sus programadores. Es esencial para saber cómo funciona, y aprender de él (N. del T.)

en una cultura consciente de sí misma. El ritmo de los adelantos era tan veloz que surgieron cincuenta culturas tecnológicas distintas y se extinguieron como moscas, sin alcanzar la estabilidad necesaria para desarrollar una tradición, argot, folklore e historia mítica comunes. La ausencia de una red eficaz comparable a la UUCP o la Internet impidió que se convirtieran en una “nación en red” por sí mismas. El creciente acceso telefónico a servicios comerciales tipo CompuServe y Genie empezaba a imponerse, pero el hecho de que los sistemas operativos distintos a Unix no vinieran con herramientas de desarrollo incorporadas hizo que circulara muy poco código fuente. Así, no se generó ninguna tradición cooperativa de “hacking”.

La cultura hacker, (des)organizada en torno a Internet y a esta altura identificada con la cultura tecnológica de Unix, era indiferente a los servicios comerciales. Ellos querían mejores herramientas y más Internet, y las PC de 32 bits baratas prometían poner ambas cosas al alcance de todos.

¿Pero qué pasaba con el software? Los sistemas Unix seguían siendo caros, varios miles de dólares. A principios de los '90 varias compañías vendían versiones para PC del UNIX de AT&T o Berkeley (BSD). Su éxito fue escaso, los precios no bajaron y (sobre todo) junto al sistema operativo no se adquirían las fuentes modificables. El modelo de negocios tradicional del software no les daba a los hackers lo que querían. Tampoco lo hacía la Fundación de Software Libre. El desarrollo de HURD, el código de Unix abierto y libre para los hackers que Richard

Stallman venía prometiendo desde hace tiempo, quedó trabado durante años y no logró producir nada parecido a un sistema sólido hasta 1996 (aunque para 1990 la FSF ya brindaba casi todas las demás partes complejas de un sistema operativo estilo Unix).

Para peor, se hacía claro a comienzos de los '90 que los diez años de esfuerzos por hacer redituable la venta de Unix terminaban en fracaso. La prometida portabilidad entre plataformas de Unix se extravió en medio de batallas legales entre la más de media docena de versiones propietarias de Unix. Los que vendían Unix fueron tan torpes, ciegos e ineptos para el marketing que Microsoft pudo robarles gran parte del mercado con su tecnología Windows ridículamente inferior.

A principios de 1993, un observador hostil contaba con motivos para decir que la historia de Unix se extinguía, y con ella la suerte de la tribu hacker. No escaseaban testigos así en la prensa informática; la inminente muerte de Unix se predecía cíclicamente cada seis meses desde fines de los '70.

En aquellos días todos creían que la era del tecno-heroísmo se había acabado, que la industria del software y la naciente Internet serían dominadas por colosos como Microsoft. La primera generación de hackers de Unix parecía agotada y envejecida (el grupo de Investigación de Ciencia Informática de Berkeley se quedó sin nafta y perdió el apoyo financiero en el '94). Fueron días deprimentes.

Pero por suerte habían estado pasando cosas fuera de la vista de la prensa comercial, y fuera incluso de la vista de la mayoría de los hackers, que terminarían generando cambios muy motivadores a fines de 1993 y 1994. Eventualmente, éstos llevarían a la cultura en una dirección completamente nueva y hacia un éxito jamás soñado.

Los Primeros Unix Libres

En el bache que dejó el intento fallido de Stallman de crear un Unix libre,⁽¹²⁾ apareció un estudiante de la Universidad de Helsinki llamado Linus Torvalds. En 1991 comenzó a desarrollar un kernel⁽¹³⁾ libre de Unix para máquinas 386 usando las herramientas de la Fundación Software Libre. Su veloz éxito inicial atrajo a muchos hackers de Internet que cooperaron con él para desarrollar Linux, un Unix totalmente equipado con fuentes gratuitas y de distribución libre.

A Linux no le faltaban competidores. En 1991, al mismo tiempo que los primeros experimentos de Linus Torvalds, William y Lyne Jolitz estaban transportando a modo de prueba el Unix de Berkeley (BSD) a la PC 386. La mayoría de los expertos que compararon la tecnología de la versión BSD con los primeros intentos de Linus dijeron que esa adaptación de BSD se convertiría en el Unix libre más importante para la PC.

12. O sea, exento de derechos de autor y con su código fuente disponible para todos, por lo que se permite hacer copias y modificarlo. Ver la segunda sección de este volumen. (N. del T.)

13. El núcleo de instrucciones más importantes de un sistema operativo. (N. del T.)

Pero el aspecto más importante de Linux no era técnico, sino sociológico. Hasta el desarrollo de Linux, todos creían que cualquier software tan complejo como un sistema operativo debía desarrollarse con cuidado, de manera coordinada, por un grupo muy unido y no muy grande de personas. Este modelo es y era típico tanto del software

comercial como de las grandes catedrales de software libre de la Fundación Software Libre de los '80; también de los proyectos BSDlibre/BSDred/BSDabierto que derivaron de las implementaciones de UNIX BSD 386 de Jolitz.

Linux evolucionó de un modo radicalmente distinto. Casi desde el principio, se abalanzaron con entusiasmo sobre él un enorme número de hackers coordinados sólo a través de Internet. La calidad se conseguía no por medio de standards rígidos o de verticalismo, sino por la estrategia tan sencilla de hacer públicas las nuevas versiones todas las semanas y a los pocos días recoger las respuestas de cientos de usuarios, creando una suerte de selección darwiniana acelerada sobre las mutaciones introducidas por programadores. Para sorpresa de la mayoría, esto funcionó bien.

Para fines de 1993, Linux podía competir en estabilidad y confiabilidad con muchos Unix comerciales, y contaba con mayor cantidad de software. Incluso empezaba a interesar a empresas comerciales. Un efecto indirecto de este proceso fue la desaparición de la mayoría de los pequeños distribuidores de Unix comerciales –que sin programadores y hackers a los que venderles, quebraron. Uno de los pocos sobrevivientes, BSDI (Berkeley Systems Design, Incorporated) se plegó a la nueva tendencia, ofreciendo las fuentes completas de su Unix-BSD y cultivando relaciones amistosas con la comunidad hacker.

Estos cambios pasaron desapercibidos en su momento, incluso dentro de la cultura hacker. La cultura hacker, desafiando todos los pronósticos de su desaparición, estaba empezando a rediseñar el mundo del software comercial a su semejanza. Pasarían cinco años más, de cualquier modo, antes de que esta tendencia se hiciera evidente.

La Gran Explosión de la Web

El crecimiento inicial de Linux se potenció con otro fenómeno: el descubrimiento masivo de la Internet. El inicio de los '90 también vio el inicio de una floreciente industria de proveedores de Internet, que vendían el acceso por unos pocos dólares al mes. Con la invención de la World Wide Web,⁽¹⁴⁾ el ya rápido crecimiento de Internet se hizo vertiginoso.

Para 1994, el año en que el grupo de desarrollo del Unix de Berkeley anunció formalmente que se retiraba, varias versiones libres diferentes de Unix (Linux y las descendientes de BSD 386) fueron los focos de atención de la actividad hacker. Linux se estaba distribuyendo comercialmente en CD-ROM y vendiéndose como pan caliente. Para fines de 1995, las grandes compañías de informática comenzaban a publicar anuncios cargados de argot celebrando lo fácil que era entrar a Internet con sus computadoras y software.

A fines de los '90 las actividades principales de la cultura hacker fueron el desarrollo de Linux y la popularización de Internet. La World Wide Web transformó la Internet en un medio masivo, y muchos de los hackers de los '80 y principios de los '90 lanzaron Servicios de Proveedores de Internet vendiendo o facilitando el acceso a las masas. La popularización de Internet trajo además para la cultura hacker el inicio de su respetabilidad pública y de la intervención política. En 1994 y 1995 el activismo de los hackers abortó el programa Clipper, que

hubiera puesto la encriptación avanzada de datos bajo control del gobierno. En 1996 los hackers movilizaron una amplia coalición para combatir la mal llamada “Acta de Decencia de las Comunicaciones” y prevenir la censura en Internet.

14. Internet existía desde fines de los ‘60, pero los protocolos para páginas web aparecen a principios de los ‘90. (N. del T.)

Con esta victoria pasamos de la historia a la realidad actual. También pasamos a un período en el que vuestro historiador se convierte en actor, no sólo observador. Esta narración continuará en “The Revenge of the Hackers” [La Venganza de los Hackers].⁽¹⁵⁾

“Todos los gobiernos son en mayor o menor medida alianzas contra el pueblo ... y en tanto los gobernantes no poseen más virtudes que los gobernados ... el poder de un gobierno sólo puede mantenerse dentro de los límites constitucionales a través de la exhibición de un poder equiparable a él, el sentimiento colectivo del pueblo.”

Benjamin Franklin, en un editorial de Philadelphia Aurora, 1794

15. Este texto de Raymond se encuentra en AA.VV., Open Sources: Voices from the open source revolution, Sebastopol (EE.UU.), O'Reilly, 1999 (versión en internet: <http://www.oreilly.com/catalog/opensources/book/raymond2.html>), así como también en la edición ampliada de su The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. Sebastopol, O'Reilly, 2001 (versión en internet: <http://www.catb.org/~esr/writings/cathedral-bazaar/>). [hay trad. cast. en elaboración de la compilación Open Sources en TLDP-ES/LuCAS (Hispalinux); ver <http://es.tldp.org/htmls/proy-opensources.html>] (N. del E.)