

해커가 되려면...

by Eric S. Raymond.

하필이면 이런 문서를 만들다니...

“해커 영어(은어) 사전(Jargon File)”의 편집자로서, 나는 열광적인 네트워크 신참자들에게서 “마법사 같은 해커가 되려면 어떠한 학습을 해야 하지요?”라는 질문을 종종 전자 우편으로 받는다. 그런데 이러한 생기 있는 질문에 답하는 어떠한 FAQ 나 웹문서를 인터넷에서 찾아 볼 수 없으니 이상한 일이다.

여기에 본인의 개인적인 의견을 진술한다.

해커란 도대체 뭐가?

“해커 은어 사전”에는 “해커”에 대해서 많은 정의를 해놓았다. 그 대부분은 기술적으로 정통한 사람과 문제를 해결하고, 한계를 극복하면서 즐거움을 구하는 사람들과 깊은 연관을 맺고 있는 정의들이다. 만약 여러분이 해커가 되는 방법을 알고 싶다면, 이러한 정의 중 단 두 가지만이 관련된 것이다.

수십 년이라는 역사의 수레를 함께해온 전문 프로그램 언어와 네트워킹 마법사들의 공유된 문화공동체가 있다. 이들은 마이크로컴퓨터와 실험적인 ARPANET 도 함께 공유하고 있다. 이들은 문화의 구성원들은 “해커”라는 말을 만들어 냈고, 이들 해커들은 인터넷을 만들었다. 그리고 해커들은 유닉스 운영체제를 오늘날의 모습으로 가꾸었다. 해커들은 유즈넷(Usenet)을 운영하고 있고, 월드 와이드 웹(World Wide Web)이 제대로 돌아가게 만들었다. 만약 여러분이 이러한 문화의 일부이고 또한 기여하고 있다면, 이 문화의 사람들은 여러분을 해커라고 부른다. “당신을 해커입니다”라고 말이다.

해커들의 사고 방식은 이러한 소프트웨어적인 해커 문화에 국한하지 않는다. 해커들은 자신들의 사고 방식과 심적 경향을 전자 공학이나 음악과 같은 다른 사상에 적용하려는 사람들이다 - 실제로 여러분은 고학 및 예술의 최상의 수준에서도 그와 같은 것을 발견할 수 있다. 소프트웨어 해커들은 그들에게서 동류 의식을 느껴, 그들도 해커라고 부를지도 모른다. 그리고 어떤 사람들은 해커들의 천성은 해커들이 종사하고 있는 특정 매체에 구애되지 않는다고 한다. 그러나 이 글에서 우리는 소프트웨어 해커의 기법과 자세, ‘해커’라는 말을 창시한 공유된 문화의 전통에 초점을 맞춰 얘기를 전개 시킬 것이다.

자신들을 해커라고 소리 높여 부르는 또 다른 집단이 있다. 과연 그럴까? 대답은 ‘아니

오'이다. 이들은 주로 10 대의 청소년들인데, 스틸을 느끼려고 컴퓨터 시스템과 전화 시스템을 프리킹(Phreaking)하는 자들이다. 진정한 해커들은 이들을 '크래커'라고 부르고 있으며, 자신들과는 어떤 상관 관계도 없기를 바라고 있다. 진정한 해커들은 대체로 크래커들을 게으르고 무책임하며, 그다지 현명하지 못한 자들이라고 여기고 있다. 자동차 시동을 걸었다고 해서 당신이 자동차 엔지니어가 될 수 없는 것처럼 컴퓨터 보안 시스템을 깬다고 해서 당신이 해커가 될 수 있는 것은 아니다. 불행히도 거의 대다수의 저널리스트들과 작가들은 크래커들을 묘사하는 말로 '해커'라는 말을 사용하여 스스로를 웃음거리로 만들고 있다. 이들의 이러한 행위는 진정한 해커들을 정말 화나게 만드는 꼴이다.

해커와 크래커의 근본적인 차이는 무엇인가? 해커들은 사상(things)을 건설하지만, 크래커들은 사상을 파괴하는 자들이다.

아직도 해커가 되고 싶다면 계속 읽어주기 바란다. 그런데 크래커가 되고 싶다면 alt.2600 뉴스 그룹으로 가라. 그리고 자신이 생각했던 것만큼 똑똑하지 않음을 깨달았다면, 이 뉴스 그룹에 올라온 게시물을 그대로 따라 하라. 이것이 크래커들에 관해서 말할 수 있는 것의 전부이다.

해커의 마음가짐

해커들은 문제를 해결하고 사상을 건설한다. 그리고 자유와 자발적인 상호 부조의 신념을 믿는다. 해커로서 인정을 받으려면, 여러분은 이러한 유형의 마음가짐으로 자신을 무장해야 할 것이다. 단순히 마음가짐만으로는 안 되고 반드시 실천이 따라야 한다. 그렇다고 결과 속이 달라서는 안 될 것이다.

해커 문화에서 인정을 받기 위한 방편으로 해커식 마음가짐을 배양하려 한다면, 여러분은 주체의 초점을 잃은 것이나 진배없다. 이러한 것을 믿는 유형의 사람이 되려면 여러분은 스스로 끊임없이 동기를 부여하고 스스로 학습해야 될 것이다. 이것이 중요한 점이다. 모든 창조적인 예술과 마찬가지로, 명인이 되기 위한 가장 효과적인 방법은 거장들의 사고 방식과 심적경향을 모방하는 것이다. 지적인 면뿐만 아니라 정서적으로도 말이다.

그래도 해커가 되고 싶다면, 다음의 문구들을 깨달을 때까지 암송하라.

1. 이 세계에는 해결을 기다리는 매력적인 문제들이 산적해 있다.

해커로 가는 길은 상당한 즐거움을 안겨 준다. 그러나 엄청난 노력이 동반되는 즐거움이

며, 노력을 통해서 동기가 유발된다. 성공적인 운동 선수는 자신들의 신체를 단련하고 육체적 한계를 극복하는 과정에서 일종의 육체적 즐거움을 얻음으로써 동기가 유발된다. 이와 같이 해커가 되어 간다는 것은 문제를 해결하고 자신의 기술을 예리하게 가다듬으며, 자신의 지능을 함양하는 과정에서 원초적인 전율을 느끼는 것이다.

이러한 방식에 체질적으로 거부감을 느끼는 그런 부류의 사람이 아니라면, 해커로서 자리매김하기 위해서는 이러한 방식이 하나의 삶이 될 필요가 있다. 그렇지 못한다면 결국에는 여러분의 해킹 기력은 섹스, 돈, 사회적 인정과 같은 일탈적인 것에 침탈 당하게 될 것이다.

그리고 자신만의 학습 능력에 따라 일종의 신념을 개발해야만 한다. 다시 말하면 비록 문제를 해결하는 데 필요한 온갖 것들을 알고 있지 못한다 할지라도 과감히 도전함으로써 배우고, 문제를 완전히 해결할 때까지 도전과 학습을 계속 해나가는 것과 같은 신조와 성실성 등을 자기 철학으로 세워야 할 것이다.

2. 그 누구도 한 문제를 두 번씩이나 해결하려 들면 안 된다.

창조적인 두뇌 활동은 가치 있는 것이 동시에 유한의 자원이다. 해결되어야 할 새로운 문제적이 산적해 있는데, 바퀴를 재발명하는 것과 같은 일에 지력과 시간을 낭비해서는 안 된다.

해커처럼 행동하려면, 여러분은 다른 해커들의 사고 시간이 매우 귀중한 것이라는 믿음이 있어야 한다 - 즉, 여러분은 정보를 공유하고 문제를 해결한 후, 그 해법을 널리 알려서 다른 해커들이 예전의 문제를 한도 끝도 없이 재차 다루기보다는 새로운 문제를 해결할 수 있도록 해야 한다. 이것은 일종의 해커의 도덕률과도 같은 것이다.

그렇다고 여러분이 행한 창조적인 작품 모두를 나누어 주라는 얘기는 아니다. 비록 그렇게 철저한 공유의 원칙을 지키는 해커들이 다른 해커들에게서 최대의 찬사와 존경을 받고 있기는 하지만 말이다. 최소한의 생활비와 컴퓨터 임대 비용을 지불할 만큼으로 자신의 창조적인 작품을 파는 것은 “해커 가치”와 전혀 모순된 것은 아니다. 그리고 가족을 부양하거나 부자가 되는 데 자신의 해킹 기술을 이용하는 것도 모순은 아니다. 단, 여러분이 해킹을 하는 동안에 자신이 해커임을 잊지 않는다면 말이다.

3. 권태와 단조로운 일은 악이다.

해커들(일반적으로 창조적인 사람들)은 절대로 지루함을 느껴서는 안 되며, 어리석고 반복적인 작업에 매달려서도 안 된다. 이러한 일의 발생은 할 수 있는 일을 하지 못한다는 뜻

이기 때문이다. 즉, 새로운 문제를 해결하는 데 참여야 할 시간과 노력을 이런 따분한 일에 빼앗기기 때문이다. 이러한 낭비는 모든 사람들에게 상처를 입히는 결과를 빚는다. 따라서 권태와 단조롭게 반복되는 일은 불쾌한 일이기도 하지만 엄밀히 말해서 악인 셈이다.

해커처럼 행동하기 위해서, 여러분은 따분한 정보를 가능한 많이 자동적으로 날려 보내는 것을 신조로 삼아야 한다. 당신 자신을 위해서 뿐만 아니라 모든 사람(특히 다른 해커들)을 위해서도 말이다.

한 가지 예외가 있기는 하다. 해커들은 관조자의 눈에는 반복적이고 따분하게 보이는 것을 아무 생각 없이 실습하기도 한다. 이것은 특정 기술의 습득 또는 여러분에게는 없는 특정의 경험을 하려는 목적이다. 그러나 생각이 있는 사람이라면 그 누구라도 이러한 권태의 타성에 굴복해서는 안 된다.

4. 자유는 좋은 것

해커들은 타고난 반권위주의자들이다. 여러분에게 지시나 명령을 내릴 수 있는 사람이라면 여러분이 때려뭇쳐 온갖 문제의 해결을 저지시킬 수 있다. 그리고 권위주의적인 마음가짐의 소유자들이 하는 일의 방식은 황당할 정도로 어리석기까지 한다. 그러므로 권위주의적인 자세와 직면하게 될 때마다 투쟁해야 한다. 적어도 당신과 다른 해커들이 질식사하지 않도록 말이다.

이는 모든 권위와 싸우라는 얘기는 아니다. 아이들은 지도를 받을 필요가 있고, 범법자들은 어느 정도 구속이 필요할 수 있다. 해커들은 명령을 좇아 허비하는 시간 이상으로 그 가 원하는 것을 얻기 위해서는 일부 권위를 받아들일지도 모른다. 그러나 이러한 수용은 제한적이고 의식적인 거래이다. 권위주의자들이 원하는 일종의 개인적 굴복 같은 것은 매물이 아니다.

권위주의는 검열과 비밀주의로 변성한다. 그리고 이들은 자발적인 협력과 정보 공유를 믿지 않는다. 즉, 이들은 자신들이 “통제할 수 있는 협력”을 좋아한다. 해커처럼 행동하려면 여러분은 이러한 검열 및 그 제도, 비밀주의 그리고 책임감 있는 성인들을 구축하는 권력과 기만 수단에 본능적인 적의를 노정시킬 수 있도록 자신을 개발해야 한다. 그리고 이 신념에 따라 기꺼이 행동할 수 있어야만 한다.

5. 마음가짐이 되었다고 능력이 없어도 된다는 얘기는 아니다.

해커가 되려면, 여러분은 이러한 자세를 개발해야 한다. 자세만 가지고 뛰어들면 해커가

될 수 없을 것이다. 그 이상의 어떤 것이 있어야 챔피언 또는 록스타가 될 수 있다. 해커가 된다는 것은 지력과 실천, 헌신 그리고 상당한 고역이 동반되는 길이다.

그러므로 여러분은 마음가짐에만 안주하지 말고, 온갖 종류의 능력을 존중해야 한다. 해커들은 시험관들이 자신들의 시간을 어영부영 보내도록 내버려 두지 않을 것이다. 해커들은 능력을 귀하게 여긴다. 특히 해킹 능력을 말이다. 어떤 것에 대한 능력은 좋은 것이다. 그 누구도 거장이 될 수 없는 기술에 관한 능력은 훌륭한 것이라 말할 수 있다. 정신적 예리함과 교묘한 재주, 고도의 집중력이 포함된 기술에 대한 능력은 더 말할 나위도 없다.

만약 여러분이 능력을 숭배한다면 능력 개발 자체를 즐길 것이다. 처절한 실천적 노력과 헌신은 단조로운 일보다는 일종의 철저한 유의가 될 것이다. 해커가 되기 위해서는 이것이 가장 중요한 요소이다.

기본적인 해킹 기술

해커의 자세는 매우 중요하다. 그러나 그 기술은 절대적으로 필요한 것이며, 자세만으로 무능을 보상할 수는 없다. 그리고 자신이 해커로 불려지기를 꿈꾸기 전에 갖추어야 할 어떤 기본적인 기술이라는 수단(toolkit)이 필요하다.

이러한 수단은 테크놀로지가 새로운 기술을 창출하고 옛 것을 쓸모없는 것으로 만들면서 서서히 변하고 있다. 기계어 프로그래밍과 최근의 HTML 을 놓고 볼 때 격세지감을 느끼지 않을 수 없다. 현시점에서는 다음에 기술하는 내용을 분명히 포함할 것이다.

1. 프로그래밍하는 방법을 배워라.

물론 이것은 가장 기본이 되는 해킹 기술이다. 1997 년까지 여러분이 반드시 배워야 할 프로그래밍 언어는 C 언어였다(비록 지금은 가장 먼저 배워야 할 유일한 언어는 아니지만). 여러분이 단 한 가지의 컴퓨터 언어도 모르고 있다면 해커가 아니다. 여러분은 어떤 한 가지 언어에 편향되지 않는 보편적 방식으로 프로그래밍 문제에 관해 사고하는 방법을 배워야 한다. 진정한 해커가 되려면, 매뉴얼에 기술된 내용과 당신이 알고 있는 것을 연관시켜 며칠 이내에 새로운 언어를 배울 수 있는 수준까지 오를 필요가 있다. 이는 여러분이 최소한 매우 다른 몇 가지 언어를 익혀야 됨을 뜻하는 말이다.

C 언어 이외에 LISP 나 PERL 도 익어야 한다(JAVA 를 목록에 올리기는 아직까지는 어렵지 않나 생각된다). 가장 중요한 해킹 언어이지만, 이것들 각각은 프로그래밍에 대한 접근

방법이 서로 다르다. 이들 모두가 매우 귀한 접근법이다.

여기서 어떻게 프로그래밍 언어를 익혀야 하는가를 구체적으로 기술할 수는 없다. 매우 복잡한 기술이기 때문이다. 그렇지만 시중의 서적이나 강좌들을 그렇지 못하다는 것을 밝히지 않을 수 없다(거의 대다수의 해커들이 독학한다). 굳이 말하라면, (1) 코드를 읽고 (2) 코드를 작성하는 것이다.

프로그램을 배운다는 것은 일상의 언어로 매우 좋은 글을 쓰는 것과 같다. 이렇게 하기 위한 최선의 방법은 명인들의 작성한 자료를 읽고 스스로 습작을 해보는 것이다. 또 지속적으로 더 많은 글을 읽고 좀더 많은 글을 작성해 본다. 한마디로 당신이 작성한 글들이 여러분들만의 틀로 일종의 힘과 경제성을 개발하기 시작할 때까지 반복에 반복을 거듭하는 길뿐이다.

읽기 좋은 코드를 찾기로 매우 어렵다. 그 이유는 소스 상태로 애송이 해커들이 읽고 어설픔게 다룰 수 있는 대형 프로그램을 구할 수 없기 때문이다. 이러한 상황은 극적으로 변화되고 있다. 지금은 소스 공개(Open Source) 소프트웨어, 프로그래밍 툴, 운영체제(이 모든 것들은 해커들이 만든 것이다)들을 어디서든 구할 수 있다.

2. 소스 공개 유닉스들 중 하나를 구해서 사용법과 운용법을 배워라.

여러분이 개인용 컴퓨터를 갖고 있거나 이에 접근할 수 있다는 전제하에 이 글을 쓴다. 어떤 신참(newbie) 해커들이 해커 기술들을 습득할 수 있는 가장 중요한 첫 단계는 리눅스 배포본이나 BSD 유닉스 계열의 운영체제를 구하여, 설치하고 이를 돌리는 일이다.

이 세상에는 유닉스 이외에도 무수히 많은 운영체제들이 있다. 그러나 이것들은 바이너리 형태로 배포되기 때문에 소스 코드를 읽고 분석하여 수정할 수 없다. 도스나 윈도우 또는 맥 운영체제에서 해킹하는 법을 배우려는 것은 몸 전체에 깃스를 하고 춤을 추는 법을 배우는 것과 같다.

게다가 유닉스는 인터넷의 운영체제이다. 유닉스를 모르고 인터넷을 배울 수는 있지만, 인터넷을 이해하지 못하고 인터넷 해커가 될 수는 없다. 이러한 연유로 오늘날의 해커 문화는 매우 강한 유닉스 중심적인 문화이다(이 얘기는 언제나 진실은 아니다. 그리고 일부 원로 해커들은 이러한 얘기에 불편한 심기를 노정한다). 그러나 유닉스와 인터넷의 공생 관계는 마이크로소프트의 힘으로도 심대한 타격을 줄 수 없을 만큼 매우 강력한 것이 되었다.

유닉스를 양육하시오 - 나 자신은 리눅스를 좋아하지만 다른 방법도 있다. 유닉스를 돌려

라. 어설프게나마 다뤄라. 유닉스를 통해서 인터넷과 대화하라. 코드를 읽어라. 코드를 수정하라. 여러분은 마이크로소프트의 운영체제가 꿈도 꿀 수 없는 것 이상의 더 말할 나위 없는 프로그래밍 툴들을 얻게 될 것이다(C를 포함하여 LISP와 PERL도). 그리고 즐거움을 누릴 것이며, 거장 해커가 되어 회고하면서 끝없이 배워야 한다는 사실을 깨닫는 것보다도 더욱 풍부한 지식에 흠뻑 빠지게 될 것이다.

유닉스 학습에 대해서 좀더 많은 것을 알려면, “The Loginataka(<http://www.ccil.org/~esr/faqs/longinataka.html>)”를 참조하시오. 리눅스를 입수할 수 있는 곳... <http://www.linuxresources.com>.

3. 월드 와이드 웹의 사용법과 HTML 작성법을 배워라.

해커문화가 건설한 대부분의 것들은 정말로 훌륭한 업적으로 가득 차 있다. 비해커들을 삶의 방식에 어떤 현시적인 충격을 주지 않고, 공장, 사무실 그리고 대학이 돌아갈 수 있게 하였다. 웹은 유일하게 예외이다. 정치인들마저도 인정했던 거대한 해커 장난감이 이제는 세상을 변화 시키고 있다. 이러한 이유하나만으로도 여러분은 웹이 어떻게 작동하는지 배워야 한다.

이 말은 단지 누구나 할 수 있는 브라우저 사용법을 익히는 것 뿐만 아니라, HTML을 작성하는 법도 배우라는 말이다. 만약 여러분이 프로그램하는 법을 모른다면, HTML 작성에 프로그램 작성에 도움이 되는 정신적인 기질을 여러분에게 가르쳐 줄 것이다. 자! 우리 모두 홈페이지를 구축하자.

그러나 홈페이지를 구축하는 것이 여러분을 해커로 만들어 줄 수 있는 충분한 조건은 아니다. 웹에는 홈페이지로 가득 차 있다. 이들 대부분은 핵심과 중심과제가 없는 진흙탕과 같은 것이다. 그것도 여러분의 마음을 빼앗을 수 있는 매우 호화로운 진창들이다. 그렇다고 파리똥이 똥이 아니라고 말할 수 있겠는가.

여러분의 웹 페이지가 가치 있는 것이 되려면 알찬 내용으로 가득 차야 한다. 재미있을 뿐만 아니라 다른 사람(해커)들에게도 유용한 것이 되어야 한다.

해커 문화에서의 신분

화폐 경제가 아니었던 대부분의 문화처럼, 해커 왕국은 명성으로 영위되고 있다. 여러분이 흥미로운 문제를 풀려고 한다. 그런데 그 문제들이 얼마나 재미있는 것인지, 여러

분의 해법이 진정 훌륭한 것인지는 오로지 여러분의 동료 또는 선배들의 판단 여하에 달려 있다. 따라서 여러분이 해커 게임을 할 때, 우선 점수를 유지하는 법을 배워야 한다. 이것으로 다른 해커들의 여러분의 기술이 어떠한지를 판단한다(이것이 다른 해커들이 계속해서 여러분을 해커라고 부를 때까지는 여러분이 해커가 될 수 없는 이유이다). 이러한 사실로 미루어 볼 때, 해킹이 고독한 작업이라는 이미지는 정확한 사실을 반영한 것이라고 볼 수 없다. 또한 자아와 외부의 확인이 각자의 동기부여에 포함되어 있다는 사실을 인정하지 않는 해커의 문화적 금기(지금은 점점 퇴색되어 가고 있지만, 여전히 잠재적인 영향을 끼치고 있다)라는 것도 현실의 반영은 아니다.

특히, 인류학자들은 해커 왕국을 “선물 문화”라고 부르고 있다. 여러분은 해커 왕국에서 신분과 명성을 얻는다. 이는 타인을 지배함으로써, 또한 아름다워지거나 타인이 원하는 것을 소유함으로써 이루어지는 것은 아니다. 오히려 여러분이 갖고 있는 것을 나누어 줌으로써 얻어지는 신분과 명성인 것이다. 특히, 여러분의 시간, 창의력, 취득된 기술의 결과물을 나누어 줌으로써 말이다.

해커들에게서 존경 받는 일을 할 수 있는 것 중 가장 기본이 되는 5 가지를 제시한다.

1. 소스 공개 소프트웨어를 작성하라.

우선(가장 핵심적이고 전통적인 것) 다른 해커들이 생각하는 프로그램이 재미있고 유용하도록 작성한다. 그리고 그 프로그램 소스를 전해커 문화가 사용할 수 있도록 나누어 준다.

우리는 이것을 “공개(무료, free) 소프트웨어”라고 부른다. 그러나 “무료(free)”라는 말이 정확히 어떤 의미를 내포하는지를 모르는 대다수의 사람들에게는 혼동만 가중시키는 꼴이 되었다. 우리 대다수의 해커들은 이 용어를 “소스 공개” 소프트웨어라고 부르기를 좋아한다.

해커 왕국에서 가장 존경 받는 인물(demigod)들은 포괄적인 필요에 걸 맞는 대형, 성능이 우수한 프로그램을 작성하여, 모든 사람들이 이용할 수 있도록 나누어 주는 사람이다.

2. 소스 공개 소프트웨어의 분석과 디버그를 할 수 있도록 도와라.

또한 소스 공개 소프트웨어를 유지하고 디버그하는 것에 봉사한다. 우리가 사는 이 불완전한 세상에서 어쩔 수 없이 우리는 디버깅하느라고 대부분의 소프트웨어 개발시간을 허비한다. 소스 공개 제작자들이 여러분에게 훌륭한 베타 테스터(beta-tester, 소프트웨어 증상을 명확하게 기술하고 문제점들을 정확히 집어내며, 줄속적으로 내놓은 판이기 때문에 어느 정도 버그들이 있을 수 있다는 것을 수용하고 아주 간단한 진단 루틴들을 기꺼이 적용할 수

있는 사람들)는 루비라는 보석에 그 가치를 덧붙이는 사람이라고 말하는 이유가 여기에 있다. 심지어 이들 테스터들은 지지부진하고 소모적인 악몽과 같은 디버깅 국면과 그저 유익한 뉘앙스가 있는 디버깅 국면의 차이점을 변별할 수 있는 사람들이다.

만약 여러분이 신참(newbie)이라면 흥미를 갖고 훌륭한 베타 테스터가 될 수 있는, 현재 개발 중인 프로그램을 찾아보기 바란다. 이는 프로그램을 테스트하는 것을 돕는 것에서 디버깅할 수 있도록 돕고, 이를 수정할 수 있도록 돕는 자연스럽고 진보적인 절차이다. 이를 통해서 많은 것을 배울 것이며, 후에 여러분을 돕게 될 사람들과 좋은 인연을 만드는 것이기도 하다.

3. 유용한 정보를 출간하라.

또 하는 선행으로서는 유용하고 재미있는 정보를 수집하고 여과하여, 웹 페이지나 FAQs와 같은 문건으로 만들어서 누구든지 이용할 수 있게 만드는 일이다.

주요 기술적인 FAQs 를 유지,개선하는 사람들은 소스 공개 저작들 만큼이나 존경을 받는다.

4. 하부구조가 견실히 작동할 수 있도록 도와라.

해커 문화(그리고 인터넷의 공학적 개발)는 지원자들에 의해 운영된다. 메일링 리스트 관리, 뉴스그룹의 조정작업, 대형 소프트웨어 아카이브 사이트를 유지하고, RFC 와 기술 표준을 개발하는 것과 같이 반드시 유지,지속되어야 하는 것이 있지만 그리 매력적인 일은 아니다.

이러한 일을 제대로 하는 사람들은 대단히 존경을 받는다. 왜냐하면 이러한 작업은 엄청나게 많은 시간이 소요되며, 코드를 갖고 노는 것처럼 흥미로운 것이 아니기 때문이다. 철저한 헌신적 봉사가 아니라면 이룰 수 없는 일이다.

5. 해커 문화 그 자체에 봉사하라.

마지막으로 여러분은 해커 문화 자체에 봉사하고, 그 사상을 전파할 수 있다(예로, 해커가 되는 법에 관한 정확한 입문서를 작성하는 것 따위). 이러한 위치에 도달하는 것은 앞에서 기술한 것들을 제대로 이해하고 실천해야만 비로소 이루어질 수 있다.

엄밀히 말해서 해커 문화는 지도자들이 없다. 그러나 그 문화의 영웅들, 부족 원로들, 그리

고 역사가와 대표자들은 있다. 여러분이 이러한 최전선의 진지를 지켜왔을 때에야 비로소 해커 문화의 일원으로 성장할 것이다.

경고: 해커들은 그들의 부족 원로들 중에서 떠들썩한 자들을 믿지 않는다. 일종의 명성을 위한 가시적인 접근은 위험한 것이다. 오히려 이러한 속 빈 강정들과 싸워라. 여러분은 여러분 자신들만의 지위를 획득해야만 명예를 얻을 수 있다. 그런 후 여러분의 신분에 관해서 겸손과 자비로운 마음을 항상 마음에 새겨 실천해야 한다.

해커(hacker)와 숙맥(nerd)간의 관계

대중적인 신화와는 달리, 해커가 되려고 숙맥이 도리 필요는 없다. 그러나 도움은 된다. 사실상 많은 해커들이 숙맥들이기는 하다. 사교적인 모임에서 버림받았다면 생각이나 해킹 처럼 정녕 중요한 일들에 집중할 수 있는 여건을 마련해 주는 셈은 된다.

이러한 연유로 많은 해커들에게는 ‘숙맥(nerd)’이라는 딱지가 붙는다. 심지어는 자궁의 상징으로써 좀더 험한 용어인 ‘변태(geek)’라는 말도 사용한다. 이는 정상적인 사회적 기대감에서 자신들은 독립적이라는 것을 선언하는 방식이다. 좀더 확장된 논의를 원하면 “The Geek Page(<http://samsara.circus.com/~omni/geek.html>)”를 방문하시오.

여러분이 해킹에 익숙해질 만큼 충분히 어떻게 해서든 집중할 수 있고, 여전히 삶을 영위하고 있다면 다행이다. 지금은 내가 1970 년대의 신참 시절을 보냈을 때보다는 훨씬 수월하게 보낼 수 있다. 다시 말해 주류 문화가 이전보다는 기술적 숙맥(techno-nerd)들에게 훨씬 호의를 보이고 있기 때문이다. 해커들은 종종 괜찮은 연인이자 배우자임을 깨닫는 사람이 점점 많아지고 있다. 좀더 깊이 있는 내용을 알려면, “Girl’s Guide to Geek Buys(<http://bunnyhop.com/BH5.geekguys.html>)”를 방문하시오.

생활을 영위할 수 없다는 이유로 해킹에 끌린다면 그것도 괜찮다. 적어도 집중에 곤란을 겪지 않을 테니까 말이다. 아마도 여러분은 나중에 해커가 될 것이다.

해커 품격 갖추기

해커가 되려면 여러분은 해커의 사고 방식과 태도를 체득해야 한다. 여러분이 컴퓨터로 도움을 줄 수 없을 때, 몇 가지 할 수 있는 일이 있다. 그렇다고 이것도 해킹의 대체물일 수는 없다. 그 어떤 것도 해킹과 대체할 수 있는 것은 없다. 그러나 많은 해커들은 다음과 같은 노력을 기울이고 있으며, 일부 기본적인 처세술과 해킹의 본질은 그 맥을 같이 하고 있

다고 느낀다.

- 과학 소설을 읽어라. 과학 소설 집회에 참석하라(해커와 원형 해커들을 만날 수 있는 좋은 방법이다).
- 선을 익히거나 병법을 익혀라(정신 수양은 무엇보다도 중요하다. 즉, 노자의 도덕경, 장자 등)
- 음악을 통해 분석적 청각을 개발하라. 특별한 음악을 감상하는 법을 배워라. 또한 악기로 연주하는 법과 창법을 배워라.
- 동음 이의적인 익살과 어휘적 유희의 변별력을 개발하라.
- 여러분의 모국어로 좋은 글을 쓰는 법을 배워라(거의 대다수의 해커들은 훌륭한 작가가 될 소질이 있다).

상기 언급한 것을 이미 행하고 있다면, 여러분은 타고난 해커적 제목감이라 할 수 있다. 언급한 내용들은 단독적으로 적용되는 것은 아니다. 서로 얼기설기 얽혀져 있다. 우리의 두뇌 좌/우반구를 모두 개발해야 한다(해커들은 논리적인 추론과 문제의 명확한 논리 외의 단계로까지 끌어올릴 수 있는 능력이 필요하다).

마지막으로 해서는 ‘안 될 것’을 적어 둔다.

- 어리석고 뽐내는 듯한 ID 나 스크린명을 사용하지 마라.
- 유즈넷 상에서 flame wars 에 끼어들지 마라(그 어디서든지).
- 여러분 자신을 ‘cyberpunk’라고 부르지 말 것이며, 쓸데없이 시간을 낭비하는 사람들에게 여러분의 시간을 할애하지 마라.
- 오자투성이에다 문법에 어긋나는 글을 게시하거나 전자 우편으로 보내지 마라.

이러한 일을 하게 되면, 명성은 하루 아침에 가을에 떨어지는 낙엽과 같은 꼴이 된다. 해커들은 상당히 좋은 기억력의 소유자들이다. 어떤 것을 받아들이면 최소한 수년 동안 잊지 않는다.

기타 자원

“Loginataka”는 유닉스 해커들의 적절한 훈련법과 자세에 관해 알려 준다.

나는 “해커 왕국의 소사(A Brief History of Hackerdom, <http://www.ccil.org/esr/faqs/hacker-hist.html>)”에 관한 글을 썼다.

피터 시바흐(Peter Seebach)는 해커와 함께 일하는 방법을 모르는 관리자들을 위해서 홀

를한 “Hacker FAQ, <http://www.solon.com/~seebs/faqs/hacker.html>”를 작성해 두고 있다.

나는 리눅스와 소스 공개 문화가 어떠한 것인지에 관해서 묘사한 글인 “대성당과 자선 시(The Cathedral and the Baszr)”라는 글을 썼다.

빈번한 질문들(FAQ)

질문: 나에게 해킹하는 법을 가르쳐 줄 수 있소?

이 웹 페이지가 개설되고 난 후에 나는 “해킹에 관해 모든 것을 가르쳐 달라”는 요청을 받아 왔다. 불행히도 나는 이러한 일에 투입할 여력이 없다. 나는 내 해킹 프로젝트를 수행하는 데 내 능력과 시간의 거의 110%를 쏟아 붓고 있다.

내 처지가 이러하다. 그러나 해킹이란 본질적으로 독학으로 얻어지는 자제이자 기술인 것이다. 만약 진정한 해커들이 당신을 돕기를 원할 때, 여러분이 그들이 알고 있는 모든 것을 어린 아이에게 음식을 먹여 주듯이 일일이 떠먹여 준다면, 그들은 당신을 존경하지 않을 것이다. 존경 받지 못하는 해커가 무슨 의미가 있는가!

작은 것부터 시작하라. 스스로 학습할 수 있는 능력이 있음을 보여라. 그리고 나서 해커들에게 의문 나는 점을 질문하라.

질문: 어디에 가면 대화를 나눌 수 있는 진정한 해커를 만날 수 있을까요?

IRC 가 아닌 것만은 확실하다. 여기는 눈을 씻고 봐도 flamer 들과 크래커들이 들끓는 곳이다. 여러분의 지역의 유닉스나 리눅스 사용자 그룹을 찾아보고, 그들의 회합에 참여하는 것이 가장 좋은 방법이다(sunsite 의 리눅스 사용자 그룹(<http://sunsite.unc.edu/linux/1에/lugs.html>)) 웹 페이지에서 유용한 사용자 그룹 목록을 찾을 수 있을 것이다.

질문: 가장 먼저 익혀야 할 언어는 무엇인가?

HTML. 이 언어를 모른다면 말이다. 걸만 번지르르하고 과장으로 가득 찬 불량 HTML 서적들이 산재해 있다. 서글프게도 HTML 에 대한 양서는 드물다. 내가 좋아하는 서적 중 하나는 HTML: The Definitive Guide(O'Reilly)이다.

프로그래밍을 시작할 준비가 되어 있다면, PERL(<http://www.perl.com>)과 Python(<http://www.python.org>)으로 시작할 것을 권한다. C 언어는 중요하다. 그러나 익히기가 좀 더 어렵다.

질문: 소스 공개 소프트웨어를 작성하여 나누어 준다면, 프로그래머들은 어떻게 생활을 꾸려나갈 수 있는가?

지금까지는 그런 것 같지 않다. 소스 공개 소프트웨어 산업은 프로그래머들을 내몰기보다는 일을 창출하는 것 같다. 작성된 프로그램을 갖고 있는 것이 전혀 작성하지 않았을 때보다 순경제적으로 이득이라면, 프로그래머들은 그 프로그램이 완성된 후에 그것이 무료이든 안되었든지 간에 보상을 받게 될 것이다. 그리고 아무리 “무료”로 소프트웨어를 작성했다 하더라도, 언제나 새롭고 주문적인 응용프로그램에 대한 좀더 많은 수요가 창출된다고 볼 수 있다.

질문: 어떻게 시작할 수 있는가? 어디서 공개 유닉스를 구할 수 있는가?

어디서든 구할 수 있다. 해커가 되지 위해서는 여러분은 자신을 교육할 수 있는 동기, 창의력과 능력이 필요하다.

esr@snark.thyrsus.com