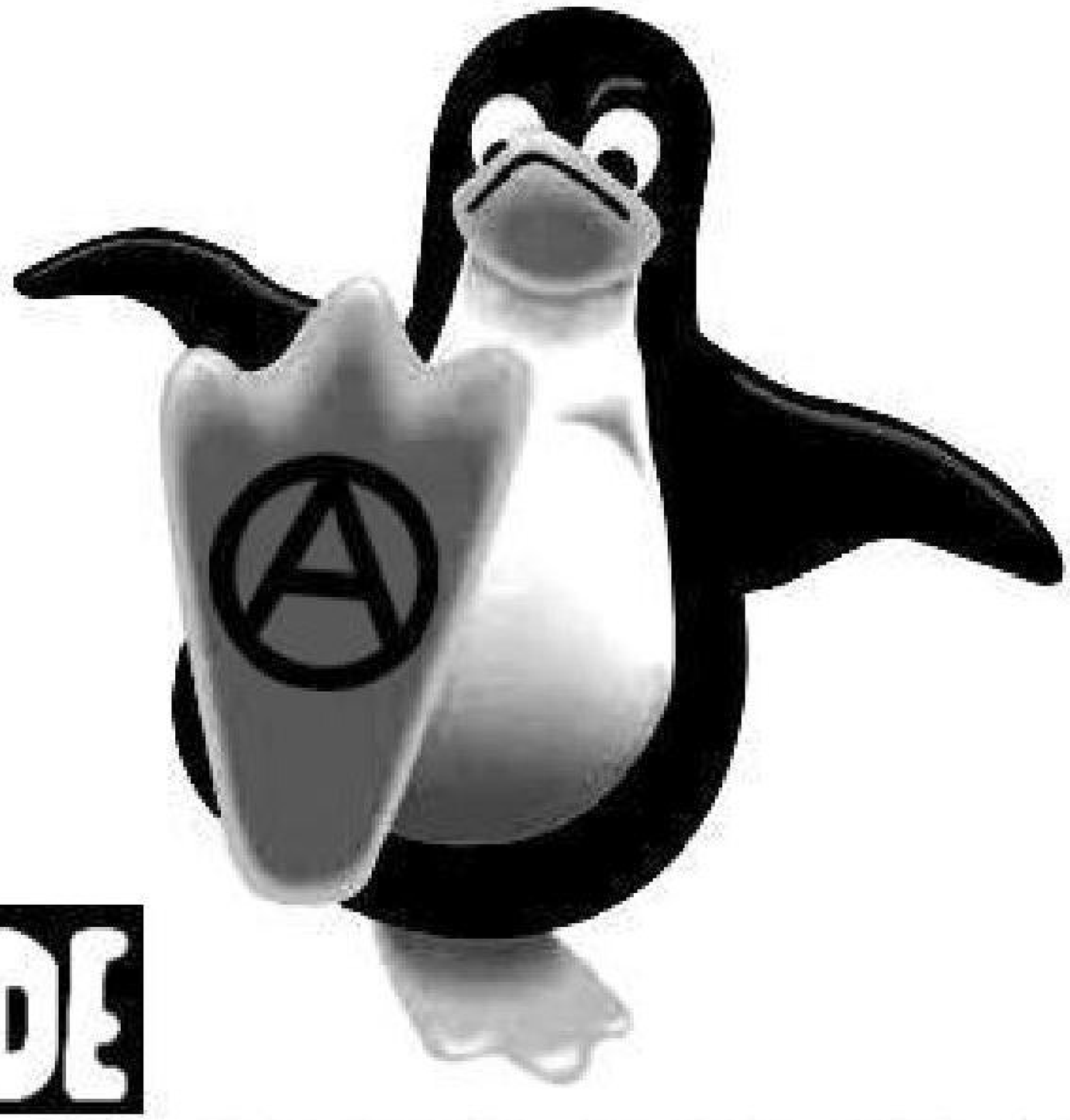


# GUIA DE SOFTWARE LIVRE

## PARA ANARQUISTAS

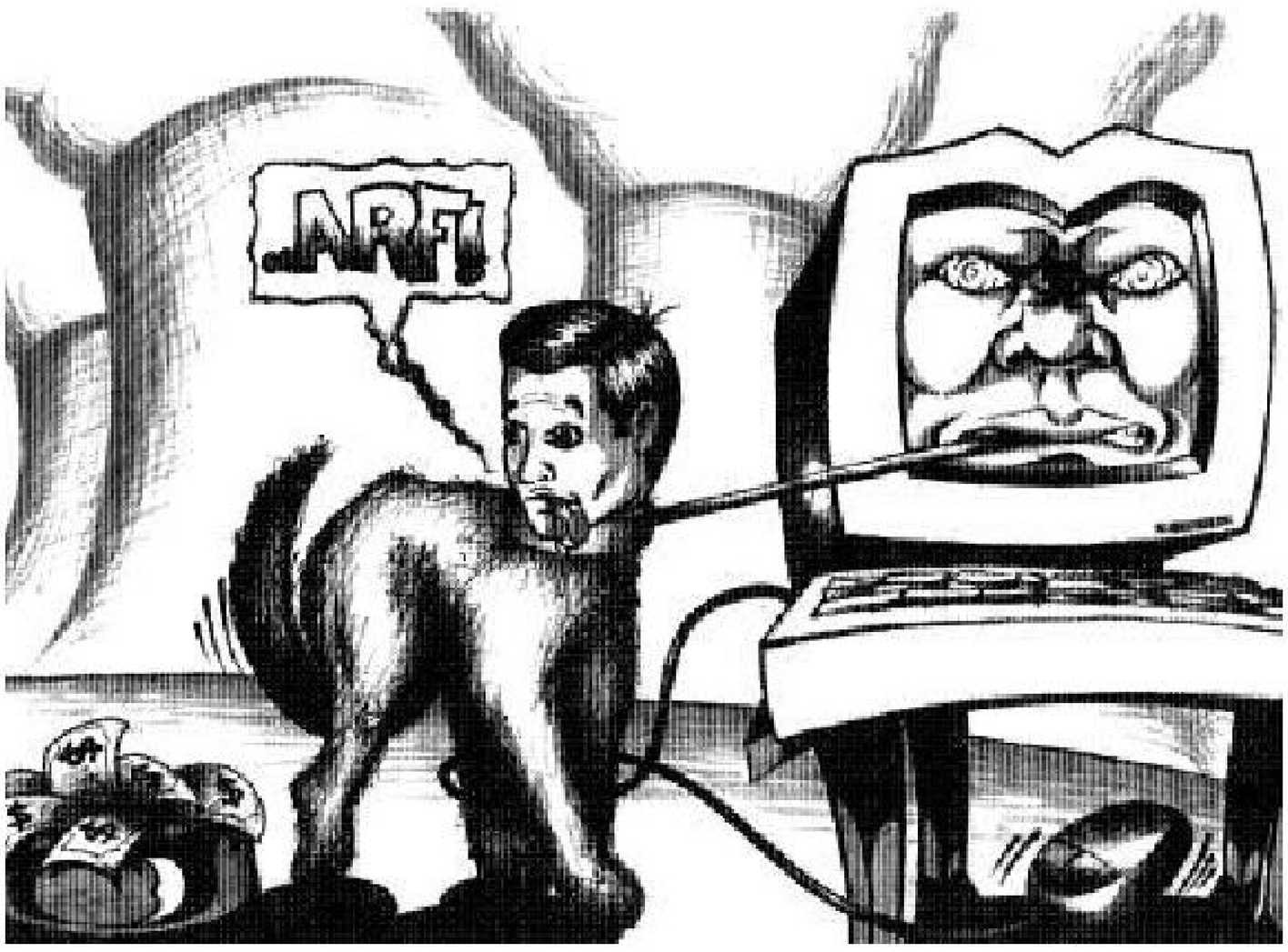
RADICAL TECHNOLOGY COLLECTIVE

# UMA INTRODUÇÃO AO SOFTWARE LIVRE

A ideia central do movimento software livre (em inglês, free software) é, obviamente, a liberdade. O movimento busca nada mais, nada menos que liberdade para x usuárix no campo dos softwares (1). O desejo por liberdade provavelmente não é novo para anarquistas, vendo como ele também é um ponto crucial do movimento. Uma importante distinção a fazer, no entanto, estão entre os dois significados em inglês (2) para a palavra free. Free pode significar tanto "ter liberdade" quanto "sem custos monetários" ("gratuito"). O free em *free software* refere-se apenas a "ter liberdade". Enquanto muitos, senão a maioria, dos softwares livres são distribuídos sem custos, alguns não o são, e continuam livres. A razão para essa distinção é não justificar limitações na disseminação do software por razões de custo (ou então o software livre seria impossível), mas colocar que ser disponível sem custos não faz do software um software livre. O movimento software livre tem um tipo de história de sua criação, e essa é a melhor maneira de apresentá-lo. Começou a muito, muito tempo, no início dos anos 70, quando todo software era livre. No início da computação, todo software era livre. Todo código-fonte era publicamente disponível, porque programas eram escritos em código de máquina - como resultado, o código-"fonte" legível para o humano e o código de "máquina" legível para o computador eram a mesma coisa. Eventualmente, programas começaram a ser escritos em linguagem assembler (3), mas essa não era realmente uma divisão entre código fonte e de máquina, já que uma instrução na linguagem assembly tem uma relação equivalente ao código de máquina no chip do processador. Como resultado, se você tivesse um programa, teria acesso ao código-fonte, e poderia estudar, modificar, e melhorar o programa.

Além disso, nos primeiros dias da computação, a comunidade de usuárixs de computador era muito pequena. Havia apenas algumas companhias envolvidas em fazer computadores e escrever programas, e a maioria das universidades ou corporações que usavam computadores escreviam seus próprios programas. Havia uma consciência de colaboração ao invés da competição dos dias de hoje - se umx programadorx na universidade escrevia código para certo computador, ele iria compartilhá-lo livremente com xs programadorxs na companhia que fez aquele computador, ou com outras universidades, ou até mesmo com outras corporações. Nesse ponto, nos anos 70 e início dos 80, não havia realmente companhias que lidavam apenas com software, e, portanto, nenhuma razão para proteger programas zelosamente - o programa era apenas algo que poderia fazer algo com o hardware, análogo a instruções para construção, digamos, de uma estante desmontada, vinda com uma caixa de ferramentas. Um dos epicentros dessa comunidade foi no Laboratório de Inteligência Artificial do MIT (Instituto de Tecnologia de Massachusetts). O Lab. IA foi terra natal dxs hackers - a

palavra "hacker", originalmente significando prankster (brincalhões; no MIT, brincadeiras mais antigas, ou outras, eram chamadas "hacks"), veio a referir-se a pessoas que se reuniam em torno de computadores no Laboratório de I.A., escrevendo software não apenas como um exercício puramente utilitário, mas como um estilo de vida. Xs hackers contestavam a autoridade, manifestada pelos engenheiros da IBM que os mantinham longe dos monstruosos mainframes(4) do lado de fora do laboratório de IA. Enquanto os pesados computadores da IBM eram mantidos por um grupo de "padres" que limitavam o acesso a todas as outras pessoas, xs hackers tinham seu próprio computador, muito menor, mas mais acessível - computador que eles preferiam. Xs hackers desprezavam o acúmulo de ferramentas ou de equipamentos - elxs eram conhecidxs por invadir escritórios de pessoas que o faziam e "libertar" o que elxs precisavam. Xs hackers tinham uma cultura de compartilhamento - todo o código que escreviam era mantido (em bobinas de papel, porque era assim que o código era armazenado então) em uma mesa no laboratório de IA, dando a qualquer umx a habilidade de aprender daquelxs que vinham antes, e adaptar seus projetos para novas coisas. E xs hackers mantinham uma meritocracia estrita - para ser respeitadx pelxs hackers, não importava a idade, status ou título, mas sim escrever códigos hackers bons, elegantes ou inteligentes. Xs hackers seriam hostis a administradorxs intrometidos que os tentassem expulsar dos computadores para usuárixs "legítimxs", e receberiam bem qualquer umx que provasse sua habilidade (o mais notável exemplo disso é Peter D, um menino de dez anos de idade que se juntou a estudantes de graduação). Eventualmente, entretanto, a comunidade hacker, depois de ir do MIT para a Universidade de Berkley, ruiu. Nos anos 1980, duas companhias se formaram,




buscando lucrar com as Máquinas Lisp (5) para as quais o Laboratório IA escreveu softwares: a Symbolics e a LMI. A LMI era mais aberta e hacker em seus procedimentos. A Symbolics começou a desencorajar a atmosfera de colaboração que fez a comunidade hacker atingir sua grandeza. O laboratório IA começou a se desagregar, com todos xs hackers indo para uma das duas companhias. Um hacker, entretanto, continuou decidido no Laboratório IA. O seu nome era Richard Stallman. Stallman estava enfurecido com as tentativas da Symbolics de terminar a livre troca de ideias que o laboratório IA representava, e, em retaliação às suas ações, reimplementou toda nova característica de softwares que a equipe da Symbolics produzia, lançando o código como software livre (apesar de ele não ser chamado assim

ainda). Ele era capaz de continuar clonando a produção da Symbolics Inc. por anos. Entretanto, neste momento, o gênio estava fora da lâmpada. Com o advento de linguagens de mais alto nível com uma definitiva separação de código-fonte e código de máquina, foi possível às distribuidoras de software coibir que usuários modificassem seus softwares, e a guerra entre a LMI e a Symbolics deixou claro que o software não-livre estava se tornando a norma. Com isso em mente, Stallman começou a trabalhar em um sistema operacional livre, o GNU, e, anos depois, fundou a Fundação Software Livre. Stallman pode ganhar mais crédito que qualquer outra pessoa pelo começo do movimento do software livre - ele foi a primeira pessoa a perceber que o software livre, outrora o modo padrão do uso de software, precisaria de um movimento para defendê-lo. A história do começo do movimento de software livre é trágica, porque também é a história do fim do estilo de vida do software livre. No entanto, desde os anos 1980, o movimento progrediu mais do que até mesmo Stallman poderia ter pensado. Hoje, é possível usar um sistema operacional somente com softwares livres, e é até mesmo possível usar um computador inteiro, dos níveis mais baixos de hardware até o sistema operacional, apenas com software livre. O Movimento de Software Livre voltou ao lugar que ocupava décadas atrás - uma pessoa pode usar um computador e ter liberdade. Entretanto, "ter liberdade" é um modo muito ambíguo de definir o software livre. Como tal, o padrão de facto para o que faz o software livre é a Definição de Software Livre, originalmente escrita por Stallman, e mantida pela Fundação de Software Livre. A Definição em si é um documento de tamanho moderado, mas, fundamentalmente, concentra quatro Liberdades que o software tem de ter se é livre. Como a Definição foi escrita por programadorxs, e programadorxs contam a partir do zero (já que é assim que os computadores contam), as liberdades estão numeradas de zero a quatro.

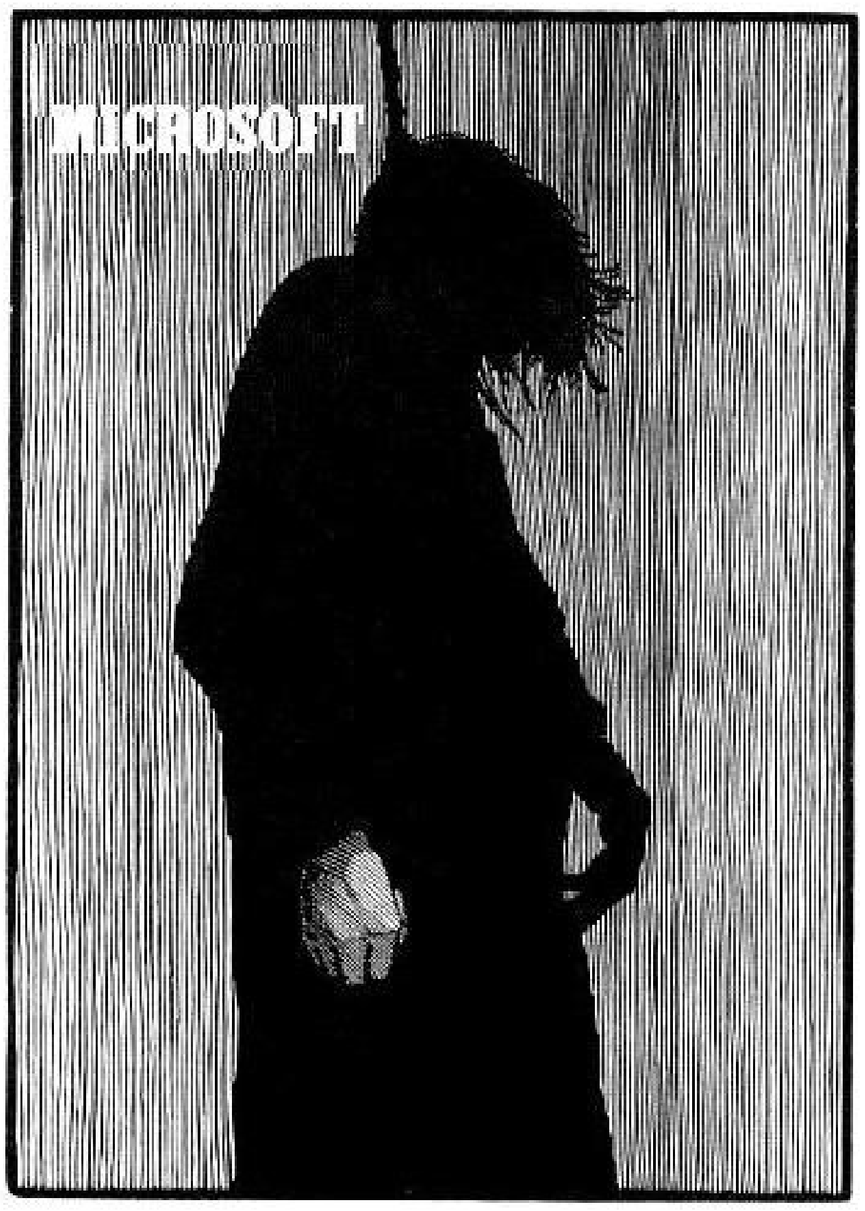
- Liberdade 0: X usuárix é livre para usar o software para qualquer objetivo.
- Liberdade 1: X usuárix é livre para estudar e modificar o software.
- Liberdade 2: X usuárix é livre para redistribuir (compartilhar) o software.
- Liberdade 3: X usuárix é livre para redistribuir (compartilhar) modificações ou versões modificadas do software.

Se x usuárix tem todas essas liberdades em consideração a uma dada parte de software, esse software é livre; e se x usuárix usa somente software livre, essx usuárix é livre. Softwares que estão disponíveis gratuitamente, como "sharewares" ou softwares "piratas" claramente não são livres, já que, mesmo que x usuárix não tenha de pagar pelo software, elx ainda não tem aquelas quatro liberdades e, como resultado, ainda está digitalmente presx à/ao autorx do software quando o usa. A tecnologia, como regra, é um gênio que não volta para dentro da lâmpada. Não importa o quanto pudermos ter desejado, não houve retorno à indústria arcaica desde o início da revolução industrial, nem retorno a caça e coleta depois do início da revolução da agricultura, e não haverá retorno ao analógico depois da revolução digital. As estruturas das quais a própria civilização depende mudarão, como todo o resto, para se incorporarem a esse novo mundo digital. Em suma, tudo vai ser, em algum nível, um computador. Todos os computadores devem rodar softwares. Se o software é livre, x usuárix

também o é. Se aquele não o é, x usuárix não o é - a liberdade que umx usuárix deve ser capaz de exercer transforma-se em um poder que a entidade controladora do software é capaz de usar para atacar x usuárix. Como todas as tecnologias, o computador é neutro. Ele pode ser usado para coordenar protestos e disseminar nossa mensagem, ou pode ser usado como uma ferramenta para o estado aumentar sua opressão. Cabe a nós assegurar que, à medida que o mundo entra na era Digital, as pessoas desse mundo não deixam sua liberdade no passado.

## Por que anarquistas deveriam usar software livre

Quando falamos sobre destruir o estado, geralmente nos referimos às ferramentas e táticas que usamos para expressar dissidência. Usualmente, os assuntos que são levantados são Black Blocs, mercados realmente livres, zines, canos de pvc, espaços libertários e coleta de materiais do lixo. Cada uma dessas ferramentas tem um lugar único em apoiar nossos esforços de nos sustentar, para nos proteger, para espalhar informação. Mas uma ferramenta raramente mencionada que tem a capacidade de nos proteger e nos libertar das instituições do velho mundo é o software. Como qualquer outra coisa, a tecnologia da Idade da Informação foi adotada em vários níveis pela comunidade anarquista. Enquanto algumas/alguns são demasiadamente cautelosxs com a tecnologia, não a usando para se comunicar, tendo menos eficiência, algumas/alguns usam a tecnologia muito liberalmente, pondo a comunidade sob perigo de vigilância. Mas segurança e autopreservação são objetivos importantes para anarquistas, não o objetivo principal, que é criar anarquia. É bem sabido que a tecnologia pode ameaçar nossa segurança, mas o que não é bem sabido é que ela pode ameaçar nossos ideais. No coração do sonho anarquista está a liberdade. Esforçamo-nos para criar um mundo livre de coerção, de um estado opressivo, de gênero e raça e qualquer outra hierarquia, um mundo no qual possamos ser livres. Como tal, já que o




anarquismo, como todo outro movimento, entrou na idade tecnológica, é imperativo que evoluamos e nos adaptemos para o mundo em mudanças, mas temos de fazê-lo de modos que não traiam nossos objetivos. O único modo de fazê-lo é com Software Livre. O "livre" em Software Livre significa livre como em "liberdade". Um programa individual, ou um grupo de programas agindo como um sistema operacional, pode ser chamado de Software Livre, mas o que o Software Livre é primariamente é um movimento social dedicado a preservar a liberdade de usuárixs de computadores. Enquanto softwares não-livres (como produtos da Microsoft ou da Apple) forçam você a usar seu computador nos

termos deles, o software livre (como o GNU/Linux) permite ativa participação em uma vibrante comunidade. Em um mundo construído sobre a ganância, acumular informação é vantajoso para as classes que nos oprimem. Ser escravizadx pelo software delas, confiando nos programas ineficientes e com falhas de segurança delas é exatamente o que elas querem. Elas querem que você seja forçado a comprar mais produtos, ser atacado por adwares e spywares (os delas, é claro, não aqueles de "criminosos" que invadem suas plataformas de merda), e elas amam que não possamos mudar ou mesmo entender o software que usamos. Elas amam que elas sejam os ditadores do seu mundo de software - a única entidade permitida para distribuir, proibindo-nos de compartilhar, a única entidade permitida para modificar, proibindo-nos de arrumar e melhorar as ferramentas que usamos. Mas em um mundo construído sobre reciprocidade e a honorável atitude *faça você mesmo*, compartilhar e melhorar software é uma das mais básicas liberdades. Uma das funções primárias do capitalismo é forçar humanxs a ficar em um papel passivo de consumidorxs, assim como Bill Gates quer, e como Steve Jobs quer, e como qualquer outrx monarca de software quer. Com software livre, x usuárix é umx participante no desenvolvimento e distribuição de programas seguros, não-corporativos, e, o mais importante. livres. Então, se rejeitamos arte corporativa, mídia corporativa, estilos de vida corporativos, ganância, subúrbios, fazendas de criação intensiva, capitalismo e autoritarismo, por que quereríamos o software que os apoia? Por que confiamos no software do capitalismo autoritário quando poderíamos estar usando software que tem um sistema de valores realmente compatível com o nosso. Temos um sistema melhor agora. É tempo de construir um novo mundo fora do monitor e do teclado do antigo.

## Por que você deveria dizer 'GNU/Linux' ao invés de 'Linux'

Quando conto às pessoas que uso GNU ou GNU/Linux, na maior parte do tempo elas não sabem do que estou falando. Mas quando digo "rodo Linux", elas sabem bem. A revolução "código aberto" foi representada na maior parte pelo Linux, e esse é o termo que as pessoas conhecem. Quando alguém se refere ao Linux ou ao GNU/Linux, se refere à mesma coisa: o sistema operacional livre, usando Linux como um núcleo (6) e os utilitários do GNU como uma base. Mas é importante dizer GNU/Linux, ao invés de Linux, especialmente para radicais. O Linux foi lançado no começo dos anos 1990. Em 1992, ele foi lançado sob os termos da GNU GPL (7), fazendo-o um software livre em copyleft. Ele ganhou popularidade durante o começo da bolha da internet (8) nos Estados Unidos, e o Linux logo ganhou muita publicidade pelo que era: software livre, mas não apenas livre, mas também estável e seguro. Companhias rodando o novo sistema operacional do Linux em seus servidores proliferaram livremente, e as pessoas cogitaram a "morte da Microsoft".

É importante saber que o termo Linux não foi popularizado pelxs defensorxs do software livre (o movimento de código aberto não existia naquele tempo), mas pela mídia de massa, que precisava de um nome para o sistema que estava fervilhando a informática. O Projeto GNU esteve por aí por algum tempo, desde

1983, mas eles não ligavam pra isso. O Linux era a nova coisa legal. O fato de que ele era a última peça em um sistema que esteve sendo feito por anos e não foi citado. Entretanto, havia outras razões pelas quais a mídia decidiu usar o termo Linux.

Agora vamos derrubar um pouco as coisas. Os meios de comunicação não estão realmente no ramo para propagar informação para a população. Eles estão lá para fazer dinheiro. Como os meios de comunicação fazem dinheiro? Anunciando. No mundo da tecnologia, quem paga por anúncios? Na maior parte? Companhias de software. Companhias de software não gostam de software livre; não porque significaria tirá-los do trabalho, mas porque companhias de software precisam que xs usuárixs sintam que não estão no controle (veja o artigo "Por que Software Livre?" para mais detalhes), que elxs não são livres. Nesse contexto, vamos comparar duas pessoas “atrás” do projeto Linux e do projeto GNU. No Linux, temos Linus Torvalds. Linus era um estudante na Universidade de Helsinki quando escreveu as primeiras versões do Linux. Linus era extremamente apolítico, e um dos primeiros apoiadores do movimento Código Aberto (pesquise "Por que NÃO Código Aberto" para detalhes sobre o movimento Código Aberto em geral). Ele diz que "código aberto é o melhor modo de fazer coisas", mas é mais pragmaticamente do que eticamente afiliado ao código aberto. Linus previu a incorporação de código não-livre ao Linux, e forçou hackers do núcleo (kernel) a usar, programas não-livres para acessar a fonte. Ele era apolítico até onde vai a liberdade. No GNU, temos Richard Stallman. Stallman era um estudante de graduação no MIT durante a era de ouro dos hackers do MIT, mas era um dos últimos hackers a habitar o Laboratório de IA do MIT. Durante a metade dos anos 1980, uma série de eventos ocorreu em sucessão, os quais provaram uma coisa a Stallman: o software tem de ser livre. Durante os dias de hacker do Laboratório de IA, todo código era compartilhado e livre. Mas agora, Stallman encontrava mais e mais acordos de não divulgação do código que o impediam de ajudar sua comunidade ou seus próprios interesses com software. Ele começou o Projeto GNU em 1983. Em completo contraste a Torvalds, Stallman era extremamente político. Falava contra software não-livre, seus abastecedores, e até mesmo contra o movimento apolítico de código aberto. Agora vamos imaginar: se você fosse o chefe de um monólito midiático, ou melhor ainda, de um monólito de mídia tecnológica, e você dependesse de milhões de dólares de receita vindos de propagandas de companhias de software proprietário, como você chamaria esse novo sistema operacional livre? Você o chamaria de GNU/Linux, o nome mais tecnicamente correto, mas politicamente perigoso, ou Linux, o nome apolítico e amistoso ao software proprietário? Como ativistas, é-nos importante colocar a liberdade antes dos dólares de anúncios e duplipensar a "indústria" de software. Dizer GNU/Linux coloca a liberdade em primeiro plano e apoia os NOSSOS objetivos, não os delxs. Assim, na próxima vez em que você estiver falando a um amigo sobre seu novo sistema operacional, não diga que você instalou o Linux. Diga-lhe que instalou o GNU/Linux. Veja também: http://www.gnu.org/gnu/why-gnu-linux.html

## Por que *radicais* não deveriam usar software não-livre ou software não-livre considerado nocivo

A maior razão pela qual anarquistas não deveriam usar software não-livre é o simples fato de que ele nega-lhes sua liberdade. Usando esse software, elxs estão implicitamente defendendo-o e expandindo o poder que o desenvolvedor do software tem sobre xs usuárixs. As razões pelas quais anarquistas devem usar somente software que respeita a liberdade são óbvias para qualquer anarquista que valoriza a liberdade, mas o que muitxs não sabem é o quão destrutivo, subversivo, e prejudicial para o software não-livre pode ser para o movimento. A maior parte dxs usuárixs de computadores, incluindo xs anarquistas, não dão muita bola, se dão alguma, ao software instalado em seu computador, além do resultado imediato. Essa ignorância tanto de liberdade quanto de segurança assegura que corporações como Apple, Adobe e Microsoft sejam os padrões de facto. No entanto, como todo software não-livre, o software produzido por essas corporações não é controlado pelas pessoas que finalmente o usam, mas pelxs autorxs iniciais do software, um grupo fechado e privado de desenvolvedores isolados dentro de uma estrutura corporativa. Esses desenvolvedores têm poder absoluto sobre usuárixs de computador. O código que escrevem, e que usuárixs rodam, pode e fará qualquer coisa, desde introduzir falhas de segurança não-intencionais até abrir portas do computador, de logs (registros) agressivos que podem dar informações sensíveis a espionagem intencional a usuárixs.

Embora anarquistas não devam usar software não-livre primeiramente porque ele nega-lhes liberdade, a segunda razão mais imediata é o risco real que seu uso tem sobre suas lutas, sejam quais forem. O software não-livre é uma "caixa escura", uma entidade desconhecida - não há conhecimento do que ele pode fazer, e o que ele pode fazer quase não tem limite. No entanto, em notáveis ocasiões, as atividades de corporações vieram à luz como sendo abertamente danosas. Este artigo tenta documentar as piores ofensas em termos de corporações abusando da confiança de usuários através de software não-livre.

Embora seja importante saber o que aconteceu no passado, é muito melhor estar ciente do que pode acontecer no futuro. Como diz o ditado, é melhor prevenir do que remediar. Não leve isso como uma lista definitiva de "maus" fornecedores de software não-livre - nunca confie em nada que não lhe permite liberdade. A Microsoft é frequentemente demonizada na comunidade do software livre, e não sem razão. A Microsoft chegou à proeminência repreendendo usuárixs por ousarem compartilhar o Atari BASIC, irritando um grande número de usuárixs que, por hobby, disseminava uma versão pré-lançada do programa aguardada por muito tempo e muito atrasada. No curso de seu crescimento, a Microsoft veio a ser conhecida como uma das mais cruéis em um círculo de cruéis, e, em 2000, foi realmente condenada nos Estados Unidos por monopólio (apesar de isso nunca ter dado em nada, dando à Microsoft uma efetiva imunidade antitruste). Os softwares da Microsoft têm historicamente um registro de merda que faz rastreio de segurança. Isso se origina do fato de que os primeiros sistemas

operacionais da Microsoft (e os posteriores, até por volta do Windows 98) eram baseados em um DOS, um sistema para umx únicx usuárix. Interrompemos este artigo para trazer-lhe um fato divertido sobre o DOS. Tipicamente, se você perguntar a alguém o que significa "DOS", a pessoa lhe responderá "Disk Operating System" ("Sistema Operacional de Disco", em tradução livre). Isso vem do PC-DOS, o sistema operacional distribuído pela IBM nos anos 1980, que, na verdade, era apenas uma versão licenciada do 86-DOS da Microsoft. A Microsoft não escreveu o 86-DOS - ela o comprou por uma ninharia do que eventualmente viria a valer de uma companhia muito menor. Seu nome original deve dar-lhe uma ideia da qualidade do software da Microsoft - QDOS, "Quick and Dirty Operating System" ("Sistema Operacional Rápido e Sujo", em tradução livre).

Todos os programas no DOS rodavam no modo mais privilegiado no processador central - isso significa, entre outras coisas, que uma falha em um único programa poderia derrubar todo o sistema. Como o DOS foi projetado para ser usado por umx usuárix sentado à frente do equipamento, não havia muito esforço para fazê-lo seguro, e quando a Microsoft reescreveu o Windows do zero para fazer o Windows NT, ele foi sobrecarregado, tendo de manter alguma compatibilidade com códigos antigos. Como tal, rodar um programa ou um sistema operacional, supondo nenhuma malícia da parte da Microsoft (uma suposição ingênua), expõe você a grandes quantidades de falhas de segurança. É por isso que há um próspero mercado para antivírus, antispyware e outros softwares de "segurança" para o Windows. A Microsoft é uma companhia gigante de softwares, com quase um monopólio no ramo do mercado de sistemas operacionais. Como tal, é ingenuidade supor que tal companhia, com seu software rodando em tantos computadores - número completamente incontável para qualquer um fora de Redmond (9) -, não seria próxima do governo dos Estados Unidos. Como qualquer empresa capitalista, a Microsoft não tem obrigação moral para com suas/seus usuárixs - apenas obrigações financeiras para com suas/seus acionistas. Portanto, se um acordo entre a Microsoft e, digamos, a NSA (10), ou o FBI, fosse lucrável, a Microsoft não teria nenhuma objeção lógica. No entanto, não precisamos confiar em especulações para mostrar o envolvimento da Microsoft com o estado. Suas próprias ações falam mais abrangentemente que a especulação jamais poderia. Em 1999, Andrew Fernandes estava analisando o mecanismo de criptografia no Service Pack 5 (11) do Windows NT. Esse service pack foi enviado a usuárixs sem ter seus símbolos separados, o que significava que coisas como nomes de variáveis e de funções do código fonte ainda estavam no código binário. Isso significa que pedaços previamente incompreensíveis em hexadecimal estavam etiquetados e categorizados. Em certo ponto, havia um pacote de hexadecimal que era uma chave criptográfica - seu nome foi marcado como _NSAKEY. A Microsoft imediatamente negou qualquer conluio com a NSA, afirmando que a chave era meramente uma chave secundária usada para assinalar módulos a serem carregados dentro do mecanismo criptográfico. As leis dos Estados Unidos proíbem a exportação de "criptografia forte" - isso é na maior parte nominal, mas companhias de software como a Microsoft ainda precisam obedecer a lei.

Parte de sua condescendência reside no fato de que módulos criptográficos podem ser carregados somente para dentro do sistema de criptografia do Windows NT se estiverem "assinados" criptograficamente por uma das chaves no sistema - seja pela chave da Microsoft, ou pela _NSAKEY, ou por uma misteriosa chave terciária que fosse encontrada posteriormente. Os que possuem essas chaves são as únicas pessoas capazes de inserir software criptografado dentro do Windows. Embora a NSA pudesse usar sua chave para carregar suas próprias cripto-rotinas supersecretas em suas cópias de Windows, eles poderiam também usá-la para carregar cripto-módulos danosos ou de monitoramento em sistemas comprometidos - por exemplo, valeria a pena espiar decisões de grupos dissidentes do governo. Combinado com o fato de que o FBI é conhecido por hackear computadores daqueles que investiga e instalar seu próprio rootkit (12), a possibilidade de um cripto-módulo malicioso assinado pela chave da NSA não é remota. A Microsoft, entretanto, não confia nas agências do governo para produzir programas de vigilância para seu sistema operacional - ela os faz por eles. Para ajudar as agências de execução das leis, a Microsoft coloca junto o COFEE - Computer Online Forensic Evidence Extractor ("Extrator Online de Evidências Forenses de Computadores", em tradução livre). O COFFE combina cento e cinquenta ferramentas para extrair senhas, logs (registros) de navegadores de internet, e outras informações que podem ajudar o estado a monitorar seus alvos. De acordo com a Microsoft, todas as ferramentas no COFEE estão publicamente disponíveis e não monitoram o Windows - mas o COFEE é disponível apenas a agências de execução das leis, então ninguém pode verificar a afirmação da Microsoft. Na verdade, é provável que se o COFEE não explora nenhuma vulnerabilidade especialmente projetada no Windows, ele provavelmente explora uma miríade de falhas em um sistema Windows. A melhor proteção contra os negócios da Microsoft com o estado é não usar a Microsoft - apesar das lições aprendidas desses dois incidentes serem facilmente transferíveis para qualquer companhia de software não-livre. O software não-livre é opaco para x usuárix e para o mundo em geral. Ele não presta contas a ninguém, salvo para o grupo que o produziu. Especialmente porque a maioria dos produtores de software não-livre são corporações e investem pesado na manutenção do status quo, xs anarquistas não deveriam confiar nunca neles para prover plataformas neutras nas quais possamos trabalhar livre e seguramente. O software livre, apesar de desenvolvido e com copyright registrado por pessoas ou entidades individuais, pode ser verificado como benigno pela comunidade, e se algum monitoramento fosse detectado, ele seria rápida e facilmente removido.

Embora o estado seja a maior ameaça a qualquer anarquista, temos de estar igualmente preocupadxs com delitos cometidos não para cidadãos obedientes da lei ou para o interesse do estado de algum modo, mas meramente por lucro. A Adobe é o melhor exemplo de uma companhia de software não-livre que ameaça a segurança e a privacidade de suas/seus usuárixs pelo interesse no lucro. O primeiro exemplo de medidas agressivas antipirataria da Adobe são os "Flash Cookies"(13), ou "Local Shared Objets" ("Objetos Compartilhados Localmente", em tradução livre). Eles agem de uma maneira similar aos cookies dos navegadores, com uma exceção - são modificáveis somente pelo Flash Player,

programa não-livre da Adobe. Isso significa que o navegador de internet dx usuárix não consegue detectá-los, ou mesmo contar a/ao usuárix que existem. Como tal, os *flash cookies* são imunes a qualquer modo de "navegação privativa". As implicações disso para a privacidade dx usuárix são óbvias - enquanto um cookie normal é excluído por restrições de privacidade do navegador, os *flash cookies* não o são. Os *flash cookies* são difíceis de serem excluídos, o que pode ser feito somente através de certos programas editores de flash. Enquanto sítios têm a leitura restrita a seus próprios *flash cookies* (um flash cookie armazenado, digamos, pelo google.com, não pode ser lido pelo yahoo.com), a única garantia que xs usuárixs têm do flash cookie é a palavra da Adobe. Não há modo de verificar sua afirmação sem acessar o código fonte do Adobe Flash Player, e, é desnecessário dizer, ninguém tem acesso a esse código fonte, exceto a própria Adobe. Sempre que software não-livre roda em seu computador, você está dando a ele as conhecidas chaves para o castelo. Quase não há limite para o que ele pode fazer sem o conhecimento dx usuárix. Usuárixs de programas CS3 da Adobe aprenderam isso quando o mais vigilante entre eles percebeu que o programa estava fazendo conexões de rede para um sítio chamado "192.168.112.2o7.net" (duas-letras-o-ponto-net). Claro, um programa não-livre discando para casa não é exatamente novo. Programas não-livres prendem usuárixs em uma única fonte de distribuição, assim elxs têm de ligar para a Central de Controle para informações sobre atualizações e ajustes de segurança (quando aqueles vêm e você os pega sem pagar). Mas há algo especial sobre o endereço "192.168.112.2o7.net". A internet é uma rede de IPs (Internet protocols - "protocolos de internet") globais. Toda máquina na rede tem um endereço IP na forma de um número especial com pontos e números decimais - quatro números variando de zero a 255, assim de 0.0.0.0 até 255.255.255.255 são todos os endereços IP válidos. No entanto, algumas variações de endereços de IP são somente para tráfego interno de não-internet - assim você pode ter sua própria rede sem tirar espaço da rede global. Suas variações? 10.xxx.xxx.xxx, 172.16-32.255.255 e - você adivinhou - 192.168.xxx.xxx. Assim quando a Adobe tinha seu software conectado a "192.168.112.2o7.net", ela estava deliberadamente tentando enganar pessoas analisando o uso da rede por elas, com um firewall ou outra ferramenta pensando que o tráfego que ia para a internet (para o sítio 2o7.net) estava indo para sua rede local. A 2o7.net é propriedade da Omniture, uma "firma de analítica comportamental" - em outras palavras, uma companhia que compra e vende informações de usuárixs. Daqui, podemos inferir que a Adobe está registrando o que usuárixs estão fazendo, em seus softwares e possivelmente além deles (a menos que você tome medidas muito rigorosas, qualquer programa em um computador pode "ver" qualquer outro e contar, até certo ponto, o que este está fazendo), e enviando esses dados para uma casa coletora onde eles podem ser vendidos a anunciantes - ou a qualquer um com dinheiro. Embora esses dados possam nunca provocar mais danos do que enviar spam sobre assuntos que você acessa, não há limite nos usos desses dados, e não há modo dx usuárix controlar esses usos. Nesse ponto, deve ser óbvio o quão nocivo o software não-livre é para qualquer umx cônscio de segurança ou realmente, a qualquer umx com uma esperança de privacidade. Softwares não-livres agem como espiões para o estado e para xs capitalistas, e, ao permitir que

esses softwares tenham livre reinado sobre o computador, suas/seus usuárixs provavelmente prejudicam bastante suas comunidades. O software é, fundamentalmente, uma ferramenta. Mas não podemos cair na armadilha de pensá-lo como uma ferramenta comum, uma ferramenta pontual - um martelo que vai bater somente no que mira, uma navalha que vai cortar qualquer coisa que colocarmos sob ela. O software é uma ferramenta inteligente - uma ferramenta que pode servir você lealmente, ou trair você sem nem um traço de culpa. O software não-livre é um conjunto pérfido de correntes, pois quem o fez se tornou adepto ao fazer a inconsciência escrava de suas correntes, até mesmo aceitando-as. Mas, a qualquer momento, na virada de um bit, no sacudir de um dedo, essas correntes podem apertar tão forte quanto qualquer outra.

Não há razão para viver acorrentado quando há a possibilidade de viver livremente. Para a segurança de nossas comunidades, para nossa própria segurança, e, o mais importante de tudo, para ambas continuarem livres, temos de nos desprender de nossas correntes.

## O que está errado com a 'pirataria'* de software

*Nota: "Pirataria é a termo da propaganda fascista do copyright para difamar o compartilhamento. O usamos no título para fazer o assunto deste artigo reconhecível, mas dentro dele, referir-nos-emos à específica ação acontecendo. Há uma falácia no que diz respeito à luta do software livre que vale rebater aqui. Essa falácia é o argumento "não posso pagar por isso, então isso é livre", onde tipicamente o meio usado para obter o software é um sistema peer-to-peer (14) ou outros meios ilícitos. O modo mais explícito pelo qual essa falácia pode ser negada é simplesmente com linguística. Embora o software possa ter sido gratuito, ele não era livre - usar um programa não-livre sem custo monetário inicial não dá liberdade a você. Há razões prementes maiores para não usar software livre além da questão do custo, e, de fato, o custo nem mesmo é discutido no movimento de software livre, já que é completamente tangencial. O uso propagandístico do termo "pirataria" para significar "compartilhamento proibido" não foi sem razão. Embora o termo carregue fortes conotações negativas para o "cidadão comum", para a juventude ele não é um termo negativo, mas positivo. Para x jovem, o pirata não é uma figura a ser temida, mas um ícone de liberdade pessoal. Se a pirataria fosse universalmente desagradável, podemos imaginar que os "Piratas do Caribe" teriam menos sequências. A atração dxs antiautoritárixs ao universo pirata também é naturalmente ligada ao mundo on-line de "pirataria", e elxs são encorajados pelo movimento pró-pirataria que adotou a terminologia de seu inimigo e as imagens de seus homônimos. O Partido Pirata tornou-se uma das facetas centrais do BitTorrent - aproximadamente metade de todos os torrents são rastreados por seus servidores. Essas imagens não são escolhidas arbitrariamente. O compartilhamento de arquivo provê possivelmente a melhor propaganda para software não-livre - ele permite que usuárixs cresçam acostumados a produtos de software não-livre e os cultiva até o não-livre se tornar padrão de facto. O Adobe Photoshop não teria se tornado um verbo não estivesse disponível

prontamente nas redes peer-to-peer e por outros meios de distribuição, permitindo que estudantes de desenho gráfico e outrxs que não pudessem ser capazes de obtê-lo legalmente o usassem e se tornassem dependentes dele.

A estratégia de marketing de longo prazo do software não-livre, como qualquer outra substância sedutora, sempre foi "mirar as crianças". Há grandes descontos de Microsoft Windows e outros softwares para escolas, para criar elos inquebráveis entre o uso do computador e software não-livre. Companhias que produzem software de edição multimídia se fecham os olhos para o compartilhamento de arquivos, sabendo que se professores suficientes ignorarem a descarga ilegal de programas pelxs estudantes, gerações de artistas digitais aprenderão ferramentas não generalizáveis como suas contrapartes fora do computador, mas ficarão presas a certos programas: Photoshop, Illustrator e Avid, todxs ganharão de longe mais lucros criando usuárixs para uma vida inteira do que perdem não condenando o compartilhamento de arquivos por estudantes.

Quando uma pessoa em uma comunidade usa um desses programas não-livres, o dano é mínimo à comunidade - sua falta de liberdade não se transmite. No entanto, problemas invariavelmente surgem sempre que esses programas são usados em um conjunto colaborativo, já que, tipicamente, usuárixs dependentes de programas não-livres rejeitarão qualquer alternativa livre. Mesmo se não houver conflito sobre o uso de software antiético, surgirão problemas quando um programa livre tornar-se disponível à comunidade - como qualquer outro software não-livre, o software não-livre obtido ilicitamente não faz conversões para qualquer outro formato, e quebra a corrente desse simples processo. Geralmente, uma quantidade substancial de propagação tem de ocorrer antes de uma migração ser possível - a propagação que não precisaria ocorrer se um programa livre fosse usado desde o primeiro dia. O fato de que esses programas possam ser obtidos sem pagar a quantia da licença é completamente irrelevante. Isso não os faz livres. O compartilhamento ilegal expõe xs ativistas a problemas legais que enfraquecem o movimento. Problemas de segurança, causados por bugs acidentais ou métodos que deliberadamente rastreiam x usuárix (comuns em produtos Adobe, entre outros), enfraquecem a segurança provida por outro software, fazendo o computador essencialmente uma plataforma hostil. Esses programas não podem ser adotados pela comunidade, já que podem ser alterados apenas por um grupo fechado de desenvolvedores corporativos. Eles não permitem que xs usuárixs exerçam sua liberdade!

## Liberdade na rede: por que anarquistas não deveriam usar o Facebook

Há uma concepção equivocada em círculos radicais, em relação ao software livre, que levou a maioria de nós, senão todxs, a nos perder. Essa concepção é a confusão entre acesso gratuito e acesso livre. Embora isso afete o uso de sistemas operacionais livres, isso é apenas a metade do problema. A maioria dxs ativistas anarquistas e também de software livre, não sabe e até recentemente ignorava completamente uma nova ameaça à sua liberdade: serviços de redes não-livres.

Um serviço de rede, ou "software como um serviço", é uma instalação de software que é completamente acessível a usuárixs em uma rede. Ao invés de usar software rodando em seus computadores, xs usuárixs se conectam ao software por meio de uma rede. Exemplos desse paradigma incluem o Facebook, Twitter, GMail, GoogleDocs e AIM - serviços de email, serviços de mensagens instantâneas e sítios de redes sociais são todos exemplos mais gerais de serviços de rede. Os ideais do software livre, mais especificamente as *'quatro liberdades'*, são irrelevantes no contexto de serviços de rede, porque a única pessoa "usando" o software é a pessoa que realmente o está rodando em seu computador, e nenhumx dxs usuárixs de um serviço de rede o está - elxs estão apenas interagindo com elx através de uma rede. Como tal, a maioria das licenças predominantes de software livre, incluindo a GNU GPL, podem ser "exploradas" por esses buracos - o fato de o público não ser um "usuário" real de software significa que um serviço de rede pode pegar código de software livre, adicionar modificações de proprietárix que seriam ilegais para distribuição, e então rodar o serviço de rede. O Meebo é um bom exemplo disso - ele é baseado na biblioteca libpurple, que está sob licença GPL. Se o Meebo fosse uma aplicação tradicional, rodando nos computadores daqueles que realmente o usam, ele teria de ser software livre para usar a libpurple, mas como ele é um serviço de rede, pode continuar não-livre. A Fundação do Software Livre, entre outras na comunidade do software livre, entendeu o perigo desse buraco apresentado ao "Mundo Livre", e, em 2007, quando lançou a terceira versão da Licença Pública Geral GNU (GLP GNU), lançou a Licença Pública Geral Affero GNU (AGPL GNU). A principal diferença entre a GPL e a AGPL é que a AGPL obriga acesso livre ao código fonte para xs usuárixs do serviço de rede. Isso é obviamente importante para a liberdade no mundo dos serviços de rede. Infelizmente, entretanto, o livre acesso ao código fonte de um serviço de rede é apenas metade da batalha. Ter a possibilidade de hospedar (sob custo pessoal) uma cópia alternativa de um software de serviço de rede é irrelevante se todos os dados que você acumulou no serviço de rede estão inacessíveis. Pegue o Facebook, por exemplo. Para que uso serviria ser capaz de criar Facebooks alternativos, se você não pode levar suas/seus amigxs, suas fotos e suas mensagens consigo? Que utilidade haveria se você estiver em um jardim cercado, incapaz de se comunicar com alguém fora dele? Os critérios para liberdade em um serviço de rede são claramente diferentes dos de software tradicional. Para um serviço de rede ser livre, umx usuárix tem de ter acesso a duas coisas:

- Liberdade para a fonte: a fonte correspondente ao software do serviço de rede sob uma licença livre, assim elx pode ter ao menos todas as liberdades do software tradicional.

- Liberdade para os dados: acesso irrestrito a todos os seus dados em um serviço de rede, e a capacidade de exportá-los em um formato padronizado e portátil tal que não fiquem presos a uma instância particular de um serviço de rede.

Essas duas liberdades são inexistentes na vasta maioria de serviços de rede usados por anarquistas. Facebook, Twitter, AIM, MSN, todos são serviços de rede que nos negam nossas liberdades. Embora certamente seja conveniente para anarquistas usar esses serviços, e alguns possam ser úteis para organização ou protestos, eles nos negam nossas liberdades e, como tal, são danosos. Serviços de rede não-livres podem também provar-se traiçoeiros, além de serem meramente não-livres. A maior parte dos serviços de rede usados por anarquistas são providos por corporações operando dentro de fronteiras, e de leis, dos Estados Unidos - isso significa duas coisas. Primeira, os provedores desses serviços de rede não vão estar interessados em ética, eles vão estar interessados em lucro. Segunda, os provedores desses serviços de rede vão cooperar com o estado contra anarquistas se isso for lucrável. Devemos nós, como oponentes do estado e do capitalismo, ceder condescendentemente nossas comunicações e nossas redes sociais a nossos inimigos? Devemos arriscar o choque de perder nossa infraestrutura, se aqueles inimigos estavam para arrancá-la de baixo de nós? O Twitter é um exemplo perfeito de um serviço de rede não-livre. O Twitter é obviamente um software não-livre - nenhumx usuárix pode ver como ele funciona, ou criar suas próprias instâncias. O Twitter não permite que usuárixs acessem seus dados de um modo exportável. Esses dois fatores significam que usuárixs estão presxs ao Twitter, e são não podem levar suas contas do Twitter para outro lugar se o desejarem. Além disso, o Twitter é um "jardim cercado": xs usuárixs do Twitter podem somente comunicar com outrxs usuárixs do Twitter. Embora anarquistas tenham usado com sucesso o Twitter para comunicarem ações, mais notavelmente a Convenção Nacional Republicana de 2008, esse sucesso atraiu a atenção do Departamento de Defesa dos EUA e do Departamento de Segurança Interior (MINILUV, na contração em inglês). O Twitter mostrou sua disposição de cooperar com os pedidos do Governo dos EUA atrasando horários de pico para ajudar dissidentes iranianos a disseminar suas mensagens (que veio a ser paralela à do Governo dos EUA) - ele não cooperaria com um esforço para capturar esses "terroristas domésticos"? O Twitter também é um exemplo perfeito de um problema resolvido, já que uma alternativa livre existe e é livre em todos os sentidos em que o Twitter não é. Esse software é chamado Laconica. O Laconica é um software livre licenciado sob a GNU AGPL. Todos os dados dos usuários estão armazenados em no formato padrão aberto FOAF ("friend-to-friend", ou "amigo-a-amigo", em tradução livre), permitindo que usuárixs exportem seus dados em um único arquivo. Além disso, o Laconica é baseado em um protocolo federado, o Open Microblogging Protocol ("Protocolo Aberto de Microblogging"), permitindo que

usuárixs de uma instalação do Laconica possam se comunicar com usuárixs de outras. O Laconica implementa a interface do programa do Twitter, assim qualquer software escrito para o Twitter funcionará no Laconica. Agora mesmo, o maior sítio rodando o Laconica em seus servidores é o identi.ca, mas se anarquistas quiserem rodar o Laconica em seus próprios servidores, e portanto ter controle total sobre o serviço de rede, seria simples fazê-lo. Combinado com tecnologias que aumentam a privacidade como o Tor, anarquistas podem criar instâncias totalmente anônimas e não-rastreáveis do Laconica para uso em um único dia de ação direta - removendo a influência de capitalistas e do estado em nossa infraestrutura. O microblog é um fenômeno relativamente novo, e enquanto o sistema está ainda em sua juventude, nós como anarquistas temos uma oportunidade de influenciar a sociedade como um todo a adotar o sistema livre, mais do que o atravancado, e criar o melhor modo de fazê-lo é dar o exemplo.

Claramente, serviços de rede não-livres não são um obstáculo intransponível. Como todas as outras questões na luta do software livre, o problema real é a simples inércia humana: até mesmo anarquistas acham difícil despertar-se o suficiente para romper as correntes do Twitter ou do Facebook. Mas se vamos continuar lutando por liberdade, pelo consenso acima da coerção, pela autonomia acima do controle, temos de romper essas correntes - para nosso próprio benefício, e pela segurança de nossas comunidades. Agora mesmo, a liberdade do serviço de rede está no mesmo estado em que a liberdade de software estava em 1988 aproximadamente: o chamado por liberdade foi ouvido, mas poucxs já prestaram atenção nele. Com tempo, e com esforço, o mundo de serviços de rede pode se tornar parte do "mundo livre" - mas temos de fazer esse esforço para fazer acontecer. Anarquistas lutaram por liberdade por todo o século XX - temos de manter essa luta viva pelo século XXI, e tomá-la para todo novo campo de batalha que surgir.

Notas:

↑ N.T.: Softwares são os programas de computador. Por questão de conveniência, a palavra software foi mantida sem itálico no restante do texto; o mesmo foi feito com a palavra hacker e suas derivações.

↑ N.T.: O texto inicial foi escrito em inglês, daí a referência ao idioma. Em inglês, "software livre" é chamado de "free software", e é a essa ambiguidade a que o autor se refere.

↑ N.T.: Assembler: linguagem de montagem de programas dita um nível acima da linguagem de máquina, mas um nível abaixo de linguagens de programação amplamente usadas atualmente.

↑ N.T.: Mainframe: computador de grande porte, que processa um volume grande de informações.

↑ N.T.: Computadores projetados para rodar eficientemente a linguagem de programação Lisp.

↑ N.T: O núcleo é a camada do sistema operacional entre os aplicativos e o hardware.

↑ N.T.: GNU GPL: GNU General Public License, ou Licença Pública Geral GNU, é uma licença do software livre criada por Stallman.

↑ N.T: Bolha especulativa que fez subir o preço das ações de muitas empresas de informática no final da década de 1990.

↑ N.T.: Sede das instalações da corporação.

↑ N.T.: National Security Agency, Agência de Segurança Nacional.

↑ N.T.: Service Pack: "Pacote de Serviço", uma espécie de atualização de Windows lançada pela Microsoft com a justificativa de arrumar falhas do sistema operacional.

↑ N.T.: Rootkit: tipo de software que se "esconde" no computador, impedindo que seja achado, e intercepta solicitações feitas ao sistema operacional, procedendo com uma invasão “escondida”.

↑ N.T: Cookie: conjunto de dados trocados entre o computador do usuário e o servidor. Geralmente armazena informações do usuário sobre um sítio web em um arquivo no computador do usuário. Podem ser usados também para rastreamento ou para roubar informações do usuário, como o artigo explica.

↑ N.T.: Sistema peer-to-peer: sistema de rede descentralizada em que cada usuário pode mandar arquivos diretamente para outro. É muito usado para compartilhamento de arquivos, em plataformas como o torrent, por exemplo.

Quando falamos sobre destruir o estado, geralmente nos referimos às ferramentas e táticas que usamos para expressar dissidência. Usualmente, os assuntos que são levantados são Black Blocs, mercados realmente livres, zines, canos de pvc, espaços libertários e coleta de materiais do lixo. Cada uma dessas ferramentas tem um lugar único em apoiar nossos esforços de nos sustentar, para nos proteger, para espalhar informação. Mas uma ferramenta raramente mencionada que tem a capacidade de nos proteger e nos libertar das instituições do velho mundo é o software. Como qualquer outra coisa, a tecnologia da Idade da Informação foi adotada em vários níveis pela comunidade anarquista. Enquanto algumas/alguns são demasiadamente cautelosxs com a tecnologia, não a usando para se comunicar, tendo menos eficiência, algumas/alguns usam a tecnologia muito liberalmente, pondo a comunidade sob perigo de vigilância. Mas segurança e autopreservação são objetivos importantes para anarquistas, não o objetivo principal, que é criar anarquia. É bem sabido que a tecnologia pode ameaçar nossa segurança, mas o que não é bem sabido é que ela pode ameaçar nossos ideais. No coração do sonho anarquista está a liberdade. Esforçamos-nos para criar um mundo livre de coerção, de um estado opressivo, de gênero e raça e qualquer outra hierarquia, um mundo no qual possamos ser livres. Como tal, já que o anarquismo, como todo outro movimento, entrou na idade tecnológica, é imperativo que evoluamos e nos adaptemos para o mundo em mudanças, mas temos de fazê-lo de modos que não traiam nossos objetivos. O único modo de fazê-lo é com Software Livre...