

General Guide

How To Ask Questions The Smart Way

<http://forumubuntu.org.uk/viewtopic.php?f=7&t=3755>

- ✓ Use meaningful, specific subject headers
- ✓ Make it easy to reply
- ✓ Write in clear, grammatical, correctly-spelled language
- ✓ Send questions in accessible, standard formats
- ✓ Be precise and informative about your problem
- ✓ Volume is not precision
- ✓ Don't claim that you have found a bug
- ✓ Grovelling is not a substitute for doing your homework
- ✓ Describe the problem's symptoms, not your guesses
- ✓ Describe your problem's symptoms in chronological order
- ✓ Describe the goal, not the step
- ✓ Don't ask people to reply by private e-mail
- ✓ Be explicit about your question
- ✓ When asking about code
- ✓ Don't post homework questions
- ✓ Prune pointless queries
- ✓ Don't flag your question as "Urgent", even if it is for you
- ✓ Courtesy never hurts, and sometimes helps
- ✓ Follow up with a brief note on the solution

If you're reading this document because you need help, and you walk away with the impression you can get it directly from the authors of this document, you are one of the idiots in question. Don't ask us questions. We'll just ignore you. We are here to show you how to get help from people who actually know about the software or hardware you're dealing with, but 99.9% of the time that will not be us. Unless you know for certain that one of the authors is an expert on what you're dealing with, leave us alone and everybody will be happier.

Introduction

In the world of hackers, the kind of answers you get to your technical questions depends as much on the way you ask the questions as on the difficulty of developing the answer. This guide will teach you how to ask questions in a way more likely to get you a satisfactory answer.

Now that use of open source has become widespread, you can often get as good answers from other, more experienced users as from hackers. This is a Good Thing; users tend to be just a little bit more tolerant of the kind of failures newbies often have. Still, treating experienced users like hackers in the ways we recommend here will generally be the most effective way to get useful answers out of them, too.

The first thing to understand is that hackers actually like hard problems and good, thought-provoking questions about them. If we didn't, we wouldn't be here. If you give us an interesting question to chew on we'll be grateful to you; good questions are a stimulus and a gift. Good questions help us develop our understanding, and often reveal problems we might not have noticed or thought about otherwise. Among hackers, "Good question!" is a strong and sincere compliment. Despite this, hackers have a reputation for meeting simple questions with what looks like hostility or arrogance. It sometimes looks like we're reflexively rude to newbies and the ignorant. But this isn't really true.

What we are, unapologetically, is hostile to people who seem to be unwilling to think or to do their own homework before asking questions. People like that are time sinks — they take without giving back, and they waste time we could have spent on another question more interesting and another person more worthy of an answer. We call people like this “losers” (and for historical reasons we sometimes spell it “lusers”).

We realize that there are many people who just want to use the software we write, and who have no interest in learning technical details. For most people, a computer is merely a tool, a means to an end; they have more important things to do and lives to live. We acknowledge that, and don't expect everyone to take an interest in the technical matters that fascinate us. Nevertheless, our style of answering questions is tuned for people who do take such an interest and are willing to be active participants in problem-solving. That's not going to change. Nor should it; if it did, we would become less effective at the things we do best.

We're (largely) volunteers. We take time out of busy lives to answer questions, and at times we're overwhelmed with them. So we filter ruthlessly. In particular, we throw away questions from people who appear to be losers in order to spend our question-answering time more efficiently, on winners. If you find this attitude obnoxious, condescending, or arrogant, check your assumptions. We're not asking you to genuflect to us — in fact, most of us would love nothing more than to deal with you as an equal and welcome you into our culture, if you put in the effort required to make that possible. But it's simply not efficient for us to try to help people who are not willing to help themselves. It's OK to be ignorant; it's not OK to play stupid.

So, while it isn't necessary to already be technically competent to get attention from us, it is necessary to demonstrate the kind of attitude that leads to competence — alert, thoughtful, observant, willing to be an active partner in developing a solution. If you can't live with this sort of discrimination, we suggest you pay somebody for a commercial support contract instead of asking hackers to personally donate help to you.

If you decide to come to us for help, you don't want to be one of the losers. You don't want to seem like one, either. The best way to get a rapid and responsive answer is to ask it like a person with smarts, confidence, and clues who just happens to need help on one particular problem.

Before You Ask

Before asking a technical question by e-mail, or in a newsgroup, or on a website chat board, do the following:

1. Try to find an answer by searching the archives of the forum you plan to post to.
2. Try to find an answer by searching the Web.
3. Try to find an answer by reading the manual.
4. Try to find an answer by reading a FAQ.
5. Try to find an answer by inspection or experimentation.
6. Try to find an answer by asking a skilled friend.
7. If you're a programmer, try to find an answer by reading the source code.
- 8.

When you ask your question, display the fact that you have done these things first; this will help establish that you're not being a lazy sponge and wasting people's time. Better yet, display what you have learned from doing these things. We like answering questions for people who have demonstrated they can learn from the answers.

Use tactics like doing a Google search on the text of whatever error message you get (searching Google groups as well as Web pages). This might well take you straight to fix documentation or a mailing list thread answering your question. Even if it doesn't, saying “I googled on the following phrase but didn't get anything that looked promising” is a good thing to do in e-mail or news

postings requesting help, if only because it records what searches won't help. It will also help to direct other people with similar problems to your thread by linking the search terms to what will hopefully be your problem and resolution thread.

Take your time. Do not expect to be able to solve a complicated problem with a few seconds of Googling. Read and understand the FAQs, sit back, relax and give the problem some thought before approaching experts. Trust us, they will be able to tell from your questions how much reading and thinking you did, and will be more willing to help if you come prepared. Don't instantly fire your whole arsenal of questions just because your first search turned up no answers (or too many).

Prepare your question. Think it through. Hasty-sounding questions get hasty answers, or none at all. The more you do to demonstrate that having put thought and effort into solving your problem before seeking help, the more likely you are to actually get help.

Beware of asking the wrong question. If you ask one that is based on faulty assumptions, J. Random Hacker is quite likely to reply with a uselessly literal answer while thinking "Stupid question...", and hoping the experience of getting what you asked for rather than what you needed will teach you a lesson.

Never assume you are entitled to an answer. You are not; you aren't, after all, paying for the service. You will earn an answer, if you earn it, by asking a substantial, interesting, and thought-provoking question — one that implicitly contributes to the experience of the community rather than merely passively demanding knowledge from others.

On the other hand, making it clear that you are able and willing to help in the process of developing the solution is a very good start. "Would someone provide a pointer?", "What is my example missing?", and "What site should I have checked?" are more likely to get answered than "Please post the exact procedure I should use." because you're making it clear that you're truly willing to complete the process if someone can just point you in the right direction.

When You Ask

Choose your forum carefully

Be sensitive in choosing where you ask your question. You are likely to be ignored, or written off as a loser, if you:

- * post your question to a forum where it's off topic
- * post a very elementary question to a forum where advanced technical questions are expected, or vice-versa
- * cross-post to too many different newsgroups
- * post a personal e-mail to somebody who is neither an acquaintance of yours nor personally responsible for solving your problem

Hackers blow off questions that are inappropriately targeted in order to try to protect their communications channels from being drowned in irrelevance. You don't want this to happen to you.

The first step, therefore, is to find the right forum. Again, Google and other Web-searching methods are your friend. Use them to find the project webpage most closely associated with the hardware or software giving you difficulties. Usually it will have links to a FAQ (Frequently Asked Questions) list, and to project mailing lists and their archives. These mailing lists are the final places to go for help, if your own efforts (including reading those FAQs you found) do not find you a solution. The project page may also describe a bug-reporting procedure, or have a link to one; if so, follow it. Shooting off an e-mail to a person or forum which you are not familiar with is risky at best. For example, do not assume that the author of an informative webpage wants to be your free consultant. Do not make optimistic guesses about whether your question will be welcome — if you're unsure,

send it elsewhere, or refrain from sending it at all.

When selecting a Web forum, newsgroup or mailing list, don't trust the name by itself too far; look for a FAQ or charter to verify your question is on-topic. Read some of the back traffic before posting so you'll get a feel for how things are done there. In fact, it's a very good idea to do a keyword search for words relating to your problem on the newsgroup or mailing list archives before you post. It may find you an answer, and if not it will help you formulate a better question. Don't shotgun-blast all the available help channels at once, that's like yelling and irritates people. Step through them softly.

Know what your topic is! One of the classic mistakes is asking questions about the Unix or Windows programming interface in a forum devoted to a language or library or tool portable across both. If you don't understand why this is a blunder, you'd be best off not asking any questions at all until you get it.

In general, questions to a well-selected public forum are more likely to get useful answers than equivalent questions to a private one. There are multiple reasons for this. One is simply the size of the pool of potential respondents. Another is the size of the audience; hackers would rather answer questions that educate many people than questions serving only a few.

Understandably, skilled hackers and authors of popular software are already receiving more than their fair share of mis-targeted messages. By adding to the flood, you could in extreme cases even be the straw that breaks the camel's back — quite a few times, contributors to popular projects have withdrawn their support because collateral damage in the form of useless e-mail traffic to their personal accounts became unbearable.

Web and IRC forums directed towards newbies often give the quickest response. Your local user group, or your Linux distribution, may advertise a Web forum or IRC channel where newbies can get help. (In non-English-speaking countries newbie forums are still more likely to be mailing lists.) These are good first places to ask, especially if you think you may have tripped over a relatively simple or common problem. An advertised IRC channel is an open invitation to ask questions there and often get answers in real time.

In fact, if you got the program that is giving you problems from a Linux distribution (as common today), it may be better to ask in the distro's forum/list before trying the program's project forum/list. The project's hackers may just say, "use our build".

Before posting to any Web forum, check if it has a Search feature. If it does, try a couple of keyword searches for something like your problem; it just might help. If you did a general Web search before (as you should have), search the forum anyway; your Web-wide search engine might not have all of this forum indexed recently.

There is an increasing tendency for projects to do user support over a Web forum or IRC channel, with e-mail reserved more for development traffic. So look for those channels first when seeking project-specific help.

As a second step, use project mailing lists

When a project has a development mailing list, write to the mailing list, not to individual developers, even if you believe you know who can best answer your question. Check the documentation of the project and its homepage for the address of a project mailing list, and use it. There are several good reasons for this policy:

* Any question good enough to be asked of one developer will also be of value to the whole group. Contrariwise, if you suspect your question is too dumb for a mailing list, it's not an excuse to harass

individual developers.

* Asking questions on the list distributes load among developers. The individual developer (especially if he's the project leader) may be too busy to answer your questions.

* Most mailing lists are archived and the archives are indexed by search engines. If you ask your question on-list and it is answered, a future querent could find your question and the answer on the Web instead of asking it again.

* If certain questions are seen to be asked often, developers can use that information to improve the documentation or the software itself to be less confusing. But if those questions are asked in private, nobody has the complete picture of what questions are asked most often.

If a project has both a “user” and a “developer” (or “hacker”) mailing list or Web forum, and you are not hacking on the code, ask in the “user” list/forum. Do not assume that you will be welcome on the developer list, where they're likely to experience your question as noise disrupting their developer traffic.

However, if you are sure your question is non-trivial, and you get no answer in the “user” list/forum for several days, try the “developer” one. You would be well advised to lurk there for a few days before posting to learn the local folkways (actually this is good advice on any private or semi-private list).

If you cannot find a project's mailing list address, but only see the address of the maintainer of the project, go ahead and write to the maintainer. But even in that case, don't assume that the mailing list doesn't exist. Mention in your e-mail that you tried and could not find the appropriate mailing list. Also mention that you don't object to having your message forwarded to other people. (Many people believe that private e-mail should remain private, even if there is nothing secret in it. By allowing your message to be forwarded you give your correspondent a choice about how to handle your e-mail.)

Use meaningful, specific subject headers

On mailing lists, newsgroups or Web forums, the subject header is your golden opportunity to attract qualified experts' attention in around 50 characters or fewer. Don't waste it on babble like “Please help me” (let alone “PLEASE HELP ME!!!!”; messages with subjects like that get discarded by reflex). Don't try to impress us with the depth of your anguish; use the space for a super-concise problem description instead.

One good convention for subject headers, used by many tech support organizations, is “object - deviation”. The “object” part specifies what thing or group of things is having a problem, and the “deviation” part describes the deviation from expected behavior.

Stupid:

HELP! Video doesn't work properly on my laptop!

Smart:

X.org 6.8.1 misshapen mouse cursor, Fooware MV1005 vid. chipset

Smarter:

X.org 6.8.1 mouse cursor on Fooware MV1005 vid. chipset - is misshapen

The process of writing an “object-deviation” description will help you organize your thinking about the problem in more detail. What is affected? Just the mouse cursor or other graphics too? Is this specific to the X.org version of X? To version 6.8.1? Is this specific to Fooware video chipsets? To model MV1005? A hacker who sees the result can immediately understand what it is that you are having a problem with and the problem you are having, at a glance.

More generally, imagine looking at the index of an archive of questions, with just the subject lines showing. Make your subject line reflect your question well enough that the next guy searching the archive with a question similar to yours will be able to follow the thread to an answer rather than posting the question again.

If you ask a question in a reply, be sure to change the subject line to indicate that you're asking a question. A Subject line that looks like "Re: test" or "Re: new bug" is less likely to attract useful amounts of attention. Also, pare quotation of previous messages to the minimum consistent with cluing in new readers.

Do not simply hit reply to a list message in order to start an entirely new thread. This will limit your audience. Some mail readers, like mutt, allow the user to sort by thread and then hide messages in a thread by folding the thread. Folks who do that will never see your message. Changing the subject is not sufficient. Mutt, and probably other mail readers, looks at other information in the e-mail's headers to assign it to a thread, not the subject line. Instead start an entirely new e-mail.

On Web forums the rules of good practice are slightly different, because messages are usually much more tightly bound to specific discussion threads and often invisible outside those threads. Changing the subject when asking a question in reply is not essential. Not all forums even allow separate subject lines on replies, and nearly nobody reads them when they do. However, asking a question in a reply is a dubious practice in itself, because it will only be seen by those who are watching this thread. So, unless you are sure you want to ask only the people currently active in the thread, start a new one.

Make it easy to reply

Finishing your query with "Please send your reply to..." makes it quite unlikely you will get an answer. If you can't be bothered to take even the few seconds required to set up a correct Reply-To header in your mail agent, we can't be bothered to take even a few seconds to think about your problem. If your mail program doesn't permit this, get a better mail program. If your operating system doesn't support any e-mail programs that permit this, get a better operating system. In Web forums, asking for a reply by e-mail is outright rude, unless you believe the information may be sensitive (and somebody will, for some unknown reason, let you but not the whole forum know it). If you want an e-mail copy when somebody replies in the thread, request that the Web forum send it; this feature is supported almost everywhere under options like "watch this thread", "send e-mail on answers", etc.

Write in clear, grammatical, correctly-spelled language

We've found by experience that people who are careless and sloppy writers are usually also careless and sloppy at thinking and coding (often enough to bet on, anyway). Answering questions for careless and sloppy thinkers is not rewarding; we'd rather spend our time elsewhere. So expressing your question clearly and well is important. If you can't be bothered to do that, we can't be bothered to pay attention. Spend the extra effort to polish your language. It doesn't have to be stiff or formal — in fact, hacker culture values informal, slangy and humorous language used with precision. But it has to be precise; there has to be some indication that you're thinking and paying attention.

Spell, punctuate, and capitalize correctly. Don't confuse "its" with "it's", "loose" with "lose", or "discrete" with "discreet". Don't TYPE IN ALL CAPS; this is read as shouting and considered rude. (All-smalls is only slightly less annoying, as it's difficult to read. Alan Cox can get away with it, but you can't.)

More generally, if you write like a semi-literate boob you will very likely be ignored. So don't use instant-messaging shortcuts. Spelling "you" as "u" makes you look like a semi-literate boob to save two entire keystrokes. Worse: writing like a l33t script kiddie hax0r is the absolute kiss of death and guarantees you will receive nothing but stony silence (or, at best, a heaping helping of scorn and sarcasm) in return.

If you are asking questions in a forum that does not use your native language, you will get a limited amount of slack for spelling and grammar errors — but no extra slack at all for laziness (and yes, we can usually spot that difference). Also, unless you know what your respondent's languages are, write in English. Busy hackers tend to simply flush questions in languages they don't understand, and English is the working language of the Internet. By writing in English you minimize your chances that your question will be discarded unread.

Send questions in accessible, standard formats

If you make your question artificially hard to read, it is more likely to be passed over in favor of one that isn't. So:

- * Send plain text mail, not HTML. (It's not hard to turn off HTML.)

- * MIME attachments are usually OK, but only if they are real content (such as an attached source file

or patch), and not merely boilerplate generated by your mail client (such as another copy of your message).

- * Don't send e-mail in which entire paragraphs are single multiply-wrapped lines. (This makes it too difficult to reply to just part of the message.) Assume that your respondents will be reading mail on 80-character-wide text displays and set your line wrap accordingly, to something less than 80.

- * However, do not wrap data (such as log file dumps or session transcripts) at any fixed column width. Data should be included as-is, so respondents can have confidence that they are seeing what you saw.

- * Don't send MIME Quoted-Printable encoding to an English-language forum. This encoding can be necessary when you're posting in a language ASCII doesn't cover, but many e-mail agents don't support it. When they break, all those =20 glyphs scattered through the text are ugly and distracting — or may actively sabotage the semantics of your text.

- * Never, ever expect hackers to be able to read closed proprietary document formats like Microsoft Word or Excel. Most hackers react to these about as well as you would to having a pile of steaming pig manure dumped on your doorstep. Even when they can cope, they resent having to do so.

- * If you're sending e-mail from a Windows machine, turn off Microsoft's stupid “Smart Quotes” feature. This is so you'll avoid sprinkling garbage characters through your mail.

- * In Web forums, do not abuse “smiley” and “HTML” features (when they are present). A smiley or two is usually OK, but colored fancy text tends to make people think you are lame. Seriously overusing smileys and color and fonts will make you come off like a giggly teenage girl, which is not generally a good idea unless you are more interested in sex than answers.

If you're using a graphical-user-interface mail client such as Netscape Messenger, MS Outlook, or their ilk, beware that it may violate these rules when used with its default settings. Most such clients have a menu-based “View Source” command. Use this on something in your sent-mail folder, verifying sending of plain text without unnecessary attached crud.

Be precise and informative about your problem

- * Describe the symptoms of your problem or bug carefully and clearly.

- * Describe the environment in which it occurs (machine, OS, application, whatever). Provide your vendor's distribution and release level (e.g.: “Fedora Core 7”, “Slackware 9.1”, etc.).

- * Describe the research you did to try and understand the problem before you asked the question.

- * Describe the diagnostic steps you took to try and pin down the problem yourself before you asked the question.

- * Describe any possibly relevant recent changes in your computer or software configuration.

Do the best you can to anticipate the questions a hacker will ask, and answer them in advance in your request for help.

Simon Tatham has written an excellent essay entitled [How to Report Bugs Effectively](#). I strongly recommend that you read it.

Volume is not precision

You need to be precise and informative. This end is not served by simply dumping huge volumes of code or data into a help request. If you have a large, complicated test case that is breaking a program, try to trim it and make it as small as possible.

This is useful for at least three reasons. One: being seen to invest effort in simplifying the question makes it more likely you'll get an answer, Two: simplifying the question makes it more likely you'll get a useful answer. Three: In the process of refining your bug report, you may develop a fix or workaround yourself.

Don't claim that you have found a bug

When you are having problems with a piece of software, don't claim you have found a bug unless you are very, very sure of your ground. Hint: unless you can provide a source-code patch that fixes the problem, or a regression test against a previous version that demonstrates incorrect behavior, you are probably not sure enough. This applies to webpages and documentation, too; if you have found a documentation "bug", you should supply replacement text and which pages it should go on. Remember, there are many other users that are not experiencing your problem. Otherwise you would have learned about it while reading the documentation and searching the Web (you did do that before complaining, didn't you?). This means that very probably it is you who are doing something wrong, not the software.

The people who wrote the software work very hard to make it work as well as possible. If you claim you have found a bug, you'll be impugning their competence, which may offend some of them even if you are correct. It's especially undiplomatic to yell "bug" in the Subject line.

When asking your question, it is best to write as though you assume you are doing something wrong, even if you are privately pretty sure you have found an actual bug. If there really is a bug, you will hear about it in the answer. Play it so the maintainers will want to apologize to you if the bug is real, rather than so that you will owe them an apology if you have messed up. Grovelling is not a substitute for doing your homework

Some people who get that they shouldn't behave rudely or arrogantly, demanding an answer, retreat to the opposite extreme of grovelling. "I know I'm just a pathetic newbie loser, but...". This is distracting and unhelpful. It's especially annoying when it's coupled with vagueness about the actual problem.

Don't waste your time, or ours, on crude primate politics. Instead, present the background facts and your question as clearly as you can. That is a better way to position yourself than by grovelling. Sometimes Web forums have separate places for newbie questions. If you feel you do have a newbie question, just go there. But don't grovel there either.

Describe the problem's symptoms, not your guesses

It's not useful to tell hackers what you think is causing your problem. (If your diagnostic theories were such hot stuff, would you be consulting others for help?) So, make sure you're telling them the raw symptoms of what goes wrong, rather than your interpretations and theories. Let them do the interpretation and diagnosis. If you feel it's important to state your guess, clearly label it as such and describe why that answer isn't working for you.

Stupid:

I'm getting back-to-back SIG11 errors on kernel compiles, and suspect a hairline crack on one of the motherboard traces. What's the best way to check for those?

Smart:

My home-built K6/233 on an FIC-PA2007 motherboard (VIA Apollo VP2 chipset) with 256MB Corsair PC133 SDRAM starts getting frequent SIG11 errors about 20 minutes after power-on during the course of kernel compiles, but never in the first 20 minutes. Rebooting doesn't restart the

clock, but powering down overnight does. Swapping out all RAM didn't help. The relevant part of a typical compile session log follows.

Since the preceding point seems to be a tough one for many people to grasp, here's a phrase to remind you: "All diagnosticians are from Missouri." That US state's official motto is "Show me" (earned in 1899, when Congressman Willard D. Vandiver said "I come from a country that raises corn and cotton and cockleburs and Democrats, and frothy eloquence neither convinces nor satisfies me. I'm from Missouri. You've got to show me.") In diagnosticians' case, it's not a matter of skepticism, but rather a literal, functional need to see whatever is as close as possible to the same raw evidence that you see, rather than your surmises and summaries. Show us.

Describe your problem's symptoms in chronological order

The clues most useful in figuring out something that went wrong often lie in the events immediately prior. So, your account should describe precisely what you did, and what the machine and software did, leading up to the blowup. In the case of command-line processes, having a session log (e.g., using the script utility) and quoting the relevant twenty or so lines is very useful.

If the program that blew up on you has diagnostic options (such as `-v` for verbose), try to select options that will add useful debugging information to the transcript. Remember that more is not necessarily better; try to choose a debug level that will inform rather than drowning the reader in junk.

If your account ends up being long (more than about four paragraphs), it might be useful to succinctly state the problem up top, then follow with the chronological tale. That way, hackers will know what to watch for in reading your account.

Describe the goal, not the step

If you are trying to find out how to do something (as opposed to reporting a bug), begin by describing the goal. Only then describe the particular step towards it that you are blocked on. Often, people who need technical help have a high-level goal in mind and get stuck on what they think is one particular path towards the goal. They come for help with the step, but don't realize that the path is wrong. It can take substantial effort to get past this.

Stupid:

How do I get the color-picker on the FooDraw program to take a hexadecimal RGB value?

Smart:

I'm trying to replace the color table on an image with values of my choosing. Right now the only way I can see to do this is by editing each table slot, but I can't get FooDraw's color picker to take a hexadecimal RGB value.

The second version of the question is smart. It allows an answer that suggests a tool better suited to the task.

Don't ask people to reply by private e-mail

Hackers believe solving problems should be a public, transparent process during which a first try at an answer can and should be corrected if someone more knowledgeable notices that it is incomplete or incorrect. Also, helpers get some of their reward for being respondents from being seen to be competent and knowledgeable by their peers.

When you ask for a private reply, you are disrupting both the process and the reward. Don't do this. It's the respondent's choice whether to reply privately — and if he does, it's usually because he thinks the question is too ill-formed or obvious to be interesting to others.

There is one limited exception to this rule. If you think the question is such that you are likely to get

many answers that are all closely similar, then the magic words are “e-mail me and I'll summarize the answers for the group”. It is courteous to try and save the mailing list or newsgroup a flood of substantially identical postings — but you have to keep the promise to summarize.

Be explicit about your question

Open-ended questions tend to be perceived as open-ended time sinks. Those people most likely to be able to give you a useful answer are also the busiest people (if only because they take on the most work themselves). People like that are allergic to open-ended time sinks, thus they tend to be allergic to open-ended questions.

You are more likely to get a useful response if you are explicit about what you want respondents to do (provide pointers, send code, check your patch, whatever). This will focus their effort and implicitly put an upper bound on the time and energy a respondent must allocate to helping you.

This is good.

To understand the world the experts live in, think of expertise as an abundant resource and time to respond as a scarce one. The less of a time commitment you implicitly ask for, the more likely you are to get an answer from someone really good and really busy.

So it is useful to frame your question to minimize the time commitment required for an expert to field it — but this is often not the same thing as simplifying the question. Thus, for example, “Would you give me a pointer to a good explanation of X?” is usually a smarter question than “Would you explain X, please?”. If you have some malfunctioning code, it is usually smarter to ask for someone to explain what's wrong with it than it is to ask someone to fix it.

When asking about code

Don't ask others to debug your broken code without giving a hint what sort of problem they should be searching for. Posting a few hundred lines of code, saying "it doesn't work", will get you ignored. Posting a dozen lines of code, saying "after line 7 I was expecting to see <x>, but <y> occurred instead" is much more likely to get you a response.

If you simply want a code review, say as much up front, and be sure to mention what areas you think might particularly need review and why.

Don't post homework questions

Hackers are good at spotting homework questions; most of us have done them ourselves. Those questions are for you to work out, so that you will learn from the experience. It is OK to ask for hints, but not for entire solutions.

If you suspect you have been passed a homework question, but can't solve it anyway, try asking in a user group forum or (as a last resort) in a “user” list/forum of a project. While the hackers will spot it, some of the advanced users may at least give you a hint.

Prune pointless queries

Resist the temptation to close your request for help with semantically-null questions like “Can anyone help me?” or “Is there an answer?” First: if you've written your problem description halfway competently, such tacked-on questions are at best superfluous. Second: because they are superfluous, hackers find them annoying — and are likely to return logically impeccable but dismissive answers like “Yes, you can be helped” and “No, there is no help for you.”

In general, asking yes-or-no questions is a good thing to avoid unless you want a yes-or-no answer.

Don't flag your question as “Urgent”, even if it is for you

That's your problem, not ours. Claiming urgency is very likely to be counter-productive: most hackers will simply delete such messages as rude and selfish attempts to elicit immediate and special attention.

There is one semi-exception. It can be worth mentioning if you're using the program in some high-profile place, one that the hackers will get excited about; in such a case, if you're under time pressure, and you say so politely, people may get interested enough to answer faster.

This is a very risky thing to do, however, because the hackers' metric for what is exciting probably differs from yours. Posting from the International Space Station would qualify, for example, but posting on behalf of a feel-good charitable or political cause would almost certainly not. In fact, posting "Urgent: Help me save the fuzzy baby seals!" will reliably get you shunned or flamed even by hackers who think fuzzy baby seals are important.

If you find this mysterious, re-read the rest of this how-to repeatedly until you understand it before posting anything at all.

Courtesy never hurts, and sometimes helps

Be courteous. Use "Please" and "Thanks for your attention" or "Thanks for your consideration". Make it clear you appreciate the time people spend helping you for free.

To be honest, this isn't as important as (and cannot substitute for) being grammatical, clear, precise and descriptive, avoiding proprietary formats etc.; hackers in general would rather get somewhat brusque but technically sharp bug reports than polite vagueness. (If this puzzles you, remember that we value a question by what it teaches us.)

However, if you've got your technical ducks in a row, politeness does increase your chances of getting a useful answer.

(We must note that the only serious objection we've received from veteran hackers to this HOWTO is with respect to our previous recommendation to use "Thanks in advance". Some hackers feel this connotes an intention not to thank anybody afterwards. Our recommendation is to either say "Thanks in advance" first and thank respondents afterwards, or express courtesy in a different way, such as by saying "Thanks for your attention" or "Thanks for your consideration".)

Follow up with a brief note on the solution

Send a note after the problem has been solved to all who helped you; let them know how it came out and thank them again for their help. If the problem attracted general interest in a mailing list or newsgroup, it's appropriate to post the followup there.

Optimally, the reply should be to the thread started by the original question posting, and should have 'FIXED', 'RESOLVED' or an equally obvious tag in the subject line. On mailing lists with fast turnaround, a potential respondent who sees a thread about "Problem X" ending with "Problem X - FIXED" knows not to waste his/her time even reading the thread (unless (s)he personally finds Problem X interesting) and can therefore use that time solving a different problem.

Your followup doesn't have to be long and involved; a simple "Howdy — it was a failed network cable! Thanks, everyone. - Bill" would be better than nothing. In fact, a short and sweet summary is better than a long dissertation unless the solution has real technical depth. Say what action solved the problem, but you need not replay the whole troubleshooting sequence.

For problems with some depth, it is appropriate to post a summary of the troubleshooting history. Describe your final problem statement. Describe what worked as a solution, and indicate avoidable blind alleys after that. The blind alleys should come after the correct solution and other summary material, rather than turning the follow-up into a detective story. Name the names of people who helped you; you'll make friends that way.

Besides being courteous and informative, this sort of followup will help others searching the archive of the mailing-list/newsgroup/forum to know exactly which solution helped you and thus may also help them.

Last, and not least, this sort of followup helps everybody who assisted feel a satisfying sense of closure about the problem. If you are not a techie or hacker yourself, trust us that this feeling is very important to the gurus and experts you tapped for help. Problem narratives that trail off into unresolved nothingness are frustrating things; hackers itch to see them resolved. The goodwill that scratching that itch earns you will be very, very helpful to you next time you need to pose a question.

Consider how you might be able to prevent others from having the same problem in the future. Ask yourself if a documentation or FAQ patch would help, and if the answer is yes send that patch to the maintainer.

Among hackers, this sort of good followup behavior is actually more important than conventional politeness. It's how you get a reputation for playing well with others, which can be a very valuable asset.

How To Interpret Answers

RTFM and STFW: How To Tell You've Seriously Screwed Up

There is an ancient and hallowed tradition: if you get a reply that reads "RTFM", the person who sent it thinks you should have Read The Fucking Manual. He or she is almost certainly right. Go read it.

RTFM has a younger relative. If you get a reply that reads "STFW", the person who sent it thinks you should have Searched The Fucking Web. He or she is almost certainly right. Go search it. (The milder version of this is when you are told "Google is your friend!")

In Web forums, you may also be told to search the forum archives. In fact, someone may even be so kind as to provide a pointer to the previous thread where this problem was solved. But do not rely on this consideration; do your archive-searching before asking.

Often, the person telling you to do a search has the manual or the web page with the information you need open, and is looking at it as he or she types. These replies mean that he thinks (a) the information you need is easy to find, and (b) you will learn more if you seek out the information than if you have it spoon-fed to you.

You shouldn't be offended by this; by hacker standards, your respondent is showing you a rough kind of respect simply by not ignoring you. You should instead be thankful for this grandmotherly kindness.

If you don't understand...

If you don't understand the answer, do not immediately bounce back a demand for clarification. Use the same tools that you used to try and answer your original question (manuals, FAQs, the Web, skilled friends) to understand the answer. Then, if you still need to ask for clarification, exhibit what you have learned.

For example, suppose I tell you: "It sounds like you've got a stuck zentry; you'll need to clear it." Then: here's a bad followup question: "What's a zentry?" Here's a good followup question: "OK, I read the man page and zentries are only mentioned under the -z and -p switches. Neither of them says anything about clearing zentries. Is it one of these or am I missing something here?"

Dealing with rudeness

Much of what looks like rudeness in hacker circles is not intended to give offense. Rather, it's the product of the direct, cut-through-the-bullshit communications style that is natural to people who are more concerned about solving problems than making others feel warm and fuzzy.

When you perceive rudeness, try to react calmly. If someone is really acting out, it is very likely a

senior person on the list or newsgroup or forum will call him or her on it. If that doesn't happen and you lose your temper, it is likely that the person you lose it at was behaving within the hacker community's norms and you will be considered at fault. This will hurt your chances of getting the information or help you want.

On the other hand, you will occasionally run across rudeness and posturing that is quite gratuitous. The flip-side of the above is that it is acceptable form to slam real offenders quite hard, dissecting their misbehavior with a sharp verbal scalpel. Be very, very sure of your ground before you try this, however. The line between correcting an incivility and starting a pointless flamewar is thin enough that hackers themselves not infrequently blunder across it; if you are a newbie or an outsider, your chances of avoiding such a blunder are low. If you're after information rather than entertainment, it's better to keep your fingers off the keyboard than to risk this.

(Some people assert that many hackers have a mild form of autism or Asperger's Syndrome, and are actually missing some of the brain circuitry that lubricates “normal” human social interaction. This may or may not be true. If you are not a hacker yourself, it may help you cope with our eccentricities if you think of us as being brain-damaged. Go right ahead. We won't care; we like being whatever it is we are, and generally have a healthy skepticism about clinical labels.)

In the next section, we'll talk about a different issue; the kind of “rudeness” you'll see when you misbehave.

On Not Reacting Like A Loser

Odds are you'll screw up a few times on hacker community forums — in ways detailed in this article, or similar. And you'll be told exactly how you screwed up, possibly with colourful asides. In public.

When this happens, the worst thing you can do is whine about the experience, claim to have been verbally assaulted, demand apologies, scream, hold your breath, threaten lawsuits, complain to people's employers, leave the toilet seat up, etc. Instead, here's what you do: Get over it. It's normal. In fact, it's healthy and appropriate.

Community standards do not maintain themselves: They're maintained by people actively applying them, visibly, in public. Don't whine that all criticism should have been conveyed via private e-mail: That's not how it works. Nor is it useful to insist you've been personally insulted when someone comments that one of your claims was wrong, or that his views differ. Those are loser attitudes.

There have been hacker forums where, out of some misguided sense of hyper-courtesy, participants are banned from posting any fault-finding with another's posts, and told “Don't say anything if you're unwilling to help the user.” The resulting departure of clueful participants to elsewhere causes them to descend into meaningless babble and become useless as technical forums. Exaggeratedly “friendly” (in that fashion) or useful: Pick one.

Remember: When that hacker tells you that you've screwed up, and (no matter how gruffly) tells you not to do it again, he's acting out of concern for (1) you and (2) his community. It would be much easier for him to ignore you and filter you out of his life. If you can't manage to be grateful, at least have a little dignity, don't whine, and don't expect to be treated like a fragile doll just because you're a newcomer with a theatrically hypersensitive soul and delusions of entitlement.

Sometimes people will attack you personally, flame without an apparent reason, etc., even if you don't screw up (or have only screwed up in their imagination). In this case, complaining is the way to really screw up.

These flamers are either lamers who don't have a clue but believe themselves to be experts, or would-be psychologists testing whether you'll screw up. The other readers either ignore them, or find ways to deal with them on their own. The flamers' behavior creates problems for themselves,

which don't have to concern you.

Don't let yourself be drawn into a flamewar, either. Most flames are best ignored — after you've checked whether they are really flames, not pointers to the ways in which you have screwed up, and not cleverly ciphered answers to your real question (this happens as well).

Questions Not To Ask

Here are some classic stupid questions, and what hackers are thinking when they don't answer them.

Q: Where can I find program or resource X?

Q: How can I use X to do Y?

Q: How can I configure my shell prompt?

Q: Can I convert an AcmeCorp document into a TeX file using the Bass-o-matic file converter?

Q: My {program, configuration, SQL statement} doesn't work

Q: I'm having problems with my Windows machine. Can you help?

Q: My program doesn't work. I think system facility X is broken.

Q: I'm having problems installing Linux or X. Can you help?

Q: How can I crack root/steal channel-ops privileges/read someone's e-mail?

Q: Where can I find program or resource X?

A: The same place I'd find it, fool — at the other end of a web search. God, doesn't everybody know how to use Google yet?

Q: How can I use X to do Y?

A: If what you want is to do Y, you should ask that question without pre-supposing the use of a method that may not be appropriate. Questions of this form often indicate a person who is not merely ignorant about X, but confused about what problem Y they are solving and too fixated on the details of their particular situation. It is generally best to ignore such people until they define their problem better.

Q: How can I configure my shell prompt?

A: If you're smart enough to ask this question, you're smart enough to RTFM and find out yourself.

Q: Can I convert an AcmeCorp document into a TeX file using the Bass-o-matic file converter?

A: Try it and see. If you did that, you'd (a) learn the answer, and (b) stop wasting my time.

Q: My {program, configuration, SQL statement} doesn't work

A: This is not a question, and I'm not interested in playing Twenty Questions to pry your actual question out of you — I have better things to do. On seeing something like this, my reaction is normally of one of the following:

* do you have anything else to add to that?

* oh, that's too bad, I hope you get it fixed.

- and this has exactly what to do with me?

•

Q: I'm having problems with my Windows machine. Can you help?

A: Yes. Throw out that Microsoft trash and install an open-source operating system like Linux or BSD.

Note: you can ask questions related to Windows machines if they are about a program that does have an official Windows build, or interacts with Windows machines (i.e., Samba). Just don't be surprised by the reply that the problem is with Windows and not the program, because Windows is so broken in general that this is very often the case.

Q: My program doesn't work. I think system facility X is broken.

A: While it is possible that you are the first person to notice an obvious deficiency in system calls

and libraries heavily used by hundreds or thousands of people, it is rather more likely that you are utterly clueless. Extraordinary claims require extraordinary evidence; when you make a claim like this one, you must back it up with clear and exhaustive documentation of the failure case.

Q: I'm having problems installing Linux or X. Can you help?

A: No. I'd need hands-on access to your machine to troubleshoot this. Go ask your local Linux user group for hands-on help. (You can find a list of user groups here.)

Note: questions about installing Linux may be appropriate if you're on a forum or mailing list about a particular distribution, and the problem is with that distro; or on local user groups forums. In this case, be sure to describe the exact details of the failure. But do careful searching first, with "linux" and all suspicious pieces of hardware.

Q: How can I crack root/steal channel-ops privileges/read someone's e-mail?

A: You're a lowlife for wanting to do such things and a moron for asking a hacker to help you.

Good and Bad Questions

Finally, I'm going to illustrate how to ask questions in a smart way by example; pairs of questions about the same problem, one asked in a stupid way and one in a smart way.

Stupid: Where can I find out stuff about the Foonly Flurbamatic?

This question just begs for "STFW" as a reply.

Smart: I used Google to try to find "Foonly Flurbamatic 2600" on the Web, but I got no useful hits.

Can I get a pointer to programming information on this device?

This one has already STFWed, and sounds like he might have a real problem.

Stupid: I can't get the code from project foo to compile. Why is it broken?

The querent assumes that somebody else screwed up. Arrogant git...

Smart: The code from project foo doesn't compile under Nulix version 6.2. I've read the FAQ, but it doesn't have anything in it about Nulix-related problems. Here's a transcript of my compilation attempt; is it something I did?

The querent has specified the environment, read the FAQ, is showing the error, and is not assuming his problems are someone else's fault. This one might be worth some attention.

Stupid: I'm having problems with my motherboard. Can anybody help?

J. Random Hacker's response to this is likely to be "Right. Do you need burping and diapering, too?" followed by a punch of the delete key.

Smart: I tried X, Y, and Z on the S2464 motherboard. When that didn't work, I tried A, B, and C. Note the curious symptom when I tried C. Obviously the florbish is grommicking, but the results aren't what one might expect. What are the usual causes of grommicking on Athlon MP motherboards? Anybody got ideas for more tests I can run to pin down the problem?

This person, on the other hand, seems worthy of an answer. He/she has exhibited problem-solving intelligence rather than passively waiting for an answer to drop from on high.

In the last question, notice the subtle but important difference between demanding "Give me an answer" and "Please help me figure out what additional diagnostics I can run to achieve enlightenment."

In fact, the form of that last question is closely based on a real incident that happened in August 2001 on the linux-kernel mailing list (lkml). I (Eric) was the one asking the question that time. I was seeing mysterious lockups on a Tyan S2462 motherboard. The list members supplied the critical information I needed to solve them.

By asking the question in the way I did, I gave people something to chew on; I made it easy and attractive for them to get involved. I demonstrated respect for my peers' ability and invited them to consult with me as a peer. I also demonstrated respect for the value of their time by telling them the blind alleys I had already run down.

Afterwards, when I thanked everyone and remarked how well the process had worked, an lkml member observed that he thought it had worked not because I'm a "name" on that list, but because I asked the question in the proper form.

Hackers are in some ways a very ruthless meritocracy; I'm certain he was right, and that if I had behaved like a sponge I would have been flamed or ignored no matter who I was. His suggestion that I write up the whole incident as instruction to others led directly to the composition of this guide.

If You Can't Get An Answer

If you can't get an answer, please don't take it personally that we don't feel we can help you.

Sometimes the members of the asked group may simply not know the answer. No response is not the same as being ignored, though admittedly it's hard to spot the difference from outside.

In general, simply re-posting your question is a bad idea. This will be seen as pointlessly annoying. Have patience: the person with your answer may be in a different time-zone and asleep. Or it may be that your question wasn't well-formed to begin with.

There are other sources of help you can go to, often sources better adapted to a novice's needs.

There are many online and local user groups who are enthusiasts about the software, even though they may never have written any software themselves. These groups often form so that people can help each other and help new users.

There are also plenty of commercial companies you can contract with for help, both large and small (Red Hat and SpikeSource are two of the best known; there are many others). Don't be dismayed at the idea of having to pay for a bit of help! After all, if your car engine blows a head gasket, chances are you would take it to a repair shop and pay to get it fixed. Even if the software didn't cost you anything, you can't expect that support to always come for free.

For popular software like Linux, there are at least 10,000 users per developer. It's just not possible for one person to handle the support calls from over 10,000 users. Remember that even if you have to pay for support, you are still paying much less than if you had to buy the software as well (and support for closed-source software is usually more expensive and less competent than support for open-source software).

How To Answer Questions in a Helpful Way

Be gentle. Problem-related stress can make people seem rude or stupid even when they're not.

Reply to a first offender off-line. There is no need of public humiliation for someone who may have made an honest mistake. A real newbie may not know how to search archives or where the FAQ is stored or posted.

If you don't know for sure, say so! A wrong but authoritative-sounding answer is worse than none at all. Don't point anyone down a wrong path simply because it's fun to sound like an expert. Be humble and honest; set a good example for both the querent and your peers.

If you can't help, don't hinder. Don't make jokes about procedures that could trash the user's setup — the poor sap might interpret these as instructions.

Ask probing questions to elicit more details. If you're good at this, the querent will learn something — and so might you. Try to turn the bad question into a good one; remember we were all newbies once.

While muttering RTFM is sometimes justified when replying to someone who is just a lazy slob, a

pointer to documentation (even if it's just a suggestion to google for a key phrase) is better. If you're going to answer the question at all, give good value. Don't suggest kludgy workarounds when somebody is using the wrong tool or approach. Suggest good tools. Reframe the question. Help your community learn from the question. When you field a good question, ask yourself "How would the relevant documentation or FAQ have to change so that nobody has to answer this again?" Then send a patch to the document maintainer.

If you did research to answer the question, demonstrate your skills rather than writing as though you pulled the answer out of your butt. Answering one good question is like feeding a hungry person one meal, but teaching them research skills by example is showing them how to grow food for a lifetime.

Related Resources

If you need instruction in the basics of how personal computers, Unix, and the Internet work, see The Unix and Internet Fundamentals HOWTO.

When you release software or write patches for software, try to follow the guidelines in the Software Release Practice HOWTO.

Acknowledgements

Evelyn Mitchell contributed some example stupid questions and inspired the "How To Give A Good Answer" section. Mikhail Ramendik contributed some particularly valuable suggestions for improvements.

How to enable root login (for newbies)

<http://forumubuntu.com/viewtopic.php?f=7&t=3657>

Navigate through System-->Administration--> Login Window, I was presented with the Login Window Preferences window. On the Security tab I found an empty checkbox labelled 'Allow local system administrator login'. I ticked this box and pressed 'Close'. I had already set a root password xxxxxx as described below (In Green), and when I rebooted Ultimate, I only had to enter Username: root and Password: xxxxxx in the opening screen to open Ultimate with full root privileges.

To set the password for 'root':- Go to System-->Administration-->Users and Groups. Enter your User password. Select 'root' in the Users settings window and press 'Properties, Set and confirm' your new password xxxxxx for root in the Account 'root' Properties window. Press 'OK' and 'Close'. Done.

Tor on Ultimate Edition 9.04

<http://forumubuntu.com/viewtopic.php?f=7&t=3534>

Time to get Tor working on Ultimate Edition Jaunty, the new release. I'll be doing this on a pretty recent

install. If you don't know what Tor is, let me explain.

Tor (The Onion Router) is a anonymity network, which is meant to make you more anonymous.

Because of the way Tor works, people won't get to know your original IP address if you set it up right. Please note that the government will still be able to monitor what you do, since they can monitor both sides of the connection. If you want to share files securely with high encryption, try Freenet.

Anyway, it's time to get going. Add these lines to '/etc/apt/sources.list' with your favorite text-editor.

```
deb http://mirror.noreply.org/pub/tor jaunty main
```

```
deb-src http://mirror.noreply.org/pub/tor jaunty main
```

Next open the terminal and do the following to make sure the system sees the new repos as 'trusted'.

```
gpg --keyserver subkeys.pgp.net --recv 94C09C7F
```

```
gpg --fingerprint 94C09C7F
```

gpg --export 94C09C7F | sudo apt-key add -

And update with 'sudo aptitude update'. Now install Tor with 'sudo aptitude install tor'.

Congratulations, you've just installed Tor! Now it's time to make Tor work with FireFox.

Install FoxyProxy and restart FireFox. Right click on 'FoxyProxy: Disabled' and then click on 'Add new proxy'. Give it a name, like Tor, and click on the tab 'Proxydetails', the second tab. Since Tor is a SOCKS proxy working on localhost, select 'SOCKS Proxy?' and click on SOCKS 5. The IP address is 127.0.0.1 and the port is 9050.

If you get questions from FoxyProxy about Tor, read the messages and select the default settings. Remember to disable cookies, javascript, java, set the 'offline' buffer to 0 MB (Settings -> Advanced -> Network) and disable automatic update from FireFox (Settings -> Advanced -> Updates). You can enable and disable Tor by clicking right-clicking on 'FoxyProxy: Disabled' and select the Tor proxy. Make sure Tor is running, and if it's not, go to the terminal and type 'sudo /etc/init.d/tor start'.

To check if Tor works, go to <http://torcheck.xenobite.eu/>. For you really paranoid, Modify Headers is able to disable and edit some browser headers, like user-agent and referrer.

Howto Find Maximum memory capacity of Your Computer

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=3506>

Before Upgrading your Computers Memory capacity, You have to check maximum memory capacity of your system supports.

How to check your maximum memory capacity
it is simple

```
$ sudo dmidecode | grep Maximum  
output
```

```
Maximum Size: 64 KB
```

```
Maximum Size: 4096 KB
```

```
Maximum Size: 0 KB
```

```
Maximum Capacity: 4 GB
```

Solve missing gpg key error

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2720>

Solve missing gpg key error ie: W: GPG error: <http://ppa.launchpad.net/intrepid> Release: The following signatures couldn't be verified because the public key is not available: NO_PUBKEY 778978B00F7992B0

W: You may want to run apt-get update to correct these problems

I made a perl script that:

- Can detect and fix any launchpad PPA link in /etc/apt/sources.list
- Backs up the original /etc/apt/sources.list as /etc/apt/sources.list.backup
- Imports GPG keys for the links detected in /etc/apt/sources.list

Requirements:

- Administrative privileges
- Perl
- Internet connection
- HTML::Parser

Open a terminal Applications>acesories>terminal & paste this

Select all text in blue

```
sudo apt-get install libhtml-parser-perl
```

```
mkdir gpg-fix && cd gpg-fix
```

```
wget http://savvas.radevic.com/launchpad/change.tar.gz -O change.tar.gz
```

```
tar xzvf change.tar.gz
```

```
perl /*.pl
cd ..
rm -Rf gpg-fix
sudo apt-get update ( wait for a while while it gets the keys ) coffee yum ....then press enter when it
shows apt get update
```

PulseAudio Fixes & System-Wide Equalizer Support

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=3077>

Note 1. Jaunty users: do not follow this guide . Thanks to the hard work of the Ultimate Edition developers,

PulseAudio is optimally configured in the Jaunty release.

Note 2. Intrepid users: due to a bug in the ALSA libraries, your PCM mixer may occasionally become muted or reset to 0% volume. If you cannot hear sound - or hear a faint crackling - refer to Part C, Step 3.

Note 3: OSSv4 users: PulseAudio does not support OSSv4, so this guide will serve no purpose to you. If you have chosen to install OSSv4 and experience issues, you should seek guidance within the threads dedicated to OSSv4. I do not recommend users to install OSSv4 due to compatibility and support issues.

PulseAudio is an advanced sound server which has been included in Ultimate Edition (i.e. the standard GNOME version) since the release of Hardy Heron (8.04). Unfortunately, Hardy shipped with a suboptimal configuration of PulseAudio which has resulted in users experiencing various problems, ranging from sporadic crashes in Firefox to sound mixing being completely broken. PulseAudio in Intrepid should work by default, but it is quite possible that your configuration is suboptimal. For more information, refer to the FAQ below.

When you are ready to follow this guide, this is all you need to know:

Hardy users: Follow Part A & B.

Intrepid users: Follow Part A & C.

Additionally:

* Appendix A gives general troubleshooting tips - if you have problems, start here.

* Appendix B gives useful information on the more advanced/technical features of PulseAudio.

* Appendix C gives information on how to properly configure specific applications that may not work with PulseAudio by default, including (but not limited to): WINE, Skype, and all OSS applications.

* Appendix D will show you how to enable equalized audio for **all** applications on your system - this is especially useful for laptop users who experience poor audio quality with their internal speakers.

Frequently Asked Questions

The most common queries are answered here.

Q. What exactly is PulseAudio?

A. From the homepage:

Quote:

PulseAudio is a sound server for POSIX and Win32 systems. A sound server is basically a proxy for your sound applications. It allows you to do advanced operations on your sound data as it passes between your application and your hardware. Things like transferring the audio to a different machine, changing the sample format or channel count and mixing several sounds into one are easily achieved using a sound server.

Simplified: PulseAudio is responsible for playback and mixing of audio on your system. It is not a sound driver - in fact, it runs on top of the Advanced Linux Sound Architecture (ALSA). Aside from all the cool effects PulseAudio provides, it serves as a replacement for ALSA's virtual sound mixing device (DmixPlugin, or "dmix") - thus allowing multiple applications to share access to your sound card.

Q. PulseAudio? Bleh! I don't want it on my system.

A. Well... tough! PulseAudio is already installed and active on Hardy and Intrepid by default; it replaces ESD (ESound Daemon) for system sounds, and most of Ultimate Edition's default applications already use it (Totem, Rhythmbox, and any other applications using the GStreamer framework). Although some high-profile applications support PulseAudio natively (such as VLC and mplayer), most applications use plain ALSA or OSS output, and thus don't have native PulseAudio support.

Q. If PulseAudio is already installed, why do I need this guide?

A. While PulseAudio has been installed by default since Hardy Heron (8.04), we dropped the ball when it came to the configuration part. A quote from the main PulseAudio developer, Lennart Pöttering:

Quote:

Some distributions did a better job adopting PulseAudio than others. On the good side I certainly have to list Mandriva, Debian, and Fedora. OTOH Ultimate Edition didn't exactly do a stellar job. They didn't do their homework. Adopting PA in a distribution is a fair amount of work, given that it interfaces with so many different things at so many different places. The integration with other systems is crucial. The information was all out there, communicated on the wiki, the mailing lists and on the PA IRC channel. But if you join and hang around on neither, then you won't get the memo. To my surprise when Ultimate Edition adopted PulseAudio they moved into one of their 'LTS' releases rightaway. Which I guess can be called gutsy -- on the background that I work for Red Hat and PulseAudio is not part of RHEL at this time. I get a lot of flak from Ultimate Edition users, and I am pretty sure the vast amount of it is undeserving and not my fault.

When PulseAudio is running, it requires exclusive access to your sound card in order to work correctly as it assumes responsibility for mixing application's sounds instead of ALSA's "dmix" device. If you launch a "regular" application that does not have explicit PulseAudio support, it will most likely try to open the "Dmix" device - and this will deprive PulseAudio of control over the sound card. From the user's perspective, they will observe that audio mixing between applications is broken.

PulseAudio includes ALSA plugins (within the package "libasound2-plugins") which are designed make regular ALSA applications remap audio to the PulseAudio server (and thus avoid mixing problems as described above). Unfortunately, Hardy Heron shipped without these plugins enabled (or even installed) by default, which is causing many, many audio mixing issues for users. To compound the problem, the version of these PulseAudio ALSA plugins in the Hardy repositories do not function correctly, so updated versions are required for ALSA applications to work correctly with PulseAudio.

By following this guide, your system will be configured to use these PulseAudio ALSA plugins for Hardy users (and updated versions of necessary packages will get installed from my PPA). Although Intrepid has these plugins installed and configured by default, following this guide is still worthwhile because a) it will ensure you have a clean PulseAudio configuration, and b) you will hopefully gain a better understanding of how PulseAudio works.

Q. I'm glad to hear these issue are fixed in Intrepid, but why the hell aren't they fixed in Hardy already?

A. The simplest answer to this question is: complexity. Hardy is a LTS (Long Term Support) release, and there is a very strict policy towards updates (SRU; even the most trivial of bugfixes are entered into a code review). In order to fix Hardy, many components will require updates and changes, including but not limited to: libflashsupport, ia32-libs, pulseaudio, libasound2, libasound2-plugins, flashplugin-nonfree, nspluginwrapper...

Up until the last moment of Hardy's development cycle, the PulseAudio ALSA plugins weren't

functioning correctly, and Flash 9 absolutely would not work without the "evil" libflashsupport library (I say evil, because it caused frequent random crashes in Firefox) - and so it wasn't possible to enact the required changes before the final release. It's possible now, but there would require a huge amount of effort to review and apply these changes.

Q. I followed your guide and PulseAudio still doesn't work!

A. Refer to Appendix A and provide the requested information in your post.

Q. I can't get Skype/WINE/an OSS application/XYZ working correctly with PulseAudio, what can I do?

A. Some applications require some extra configuration, and some applications don't work with PulseAudio - please refer to Appendix C for information on specific applications.

Q. Where can I find the appropriate bug reports related to these issues?

A. If you click on a step number it will link to the appropriate bug report, if one exists.

Part A: Common instructions (Hardy & Intrepid)

All users must must follow the steps in this section to guarantee a fully working PulseAudio configuration.

1. Backup (and then delete) your previous configuration files:

Code:

```
$ mkdir ~/pulse-backup && cp -r ~/.pulse ~/.asound* /etc/asound.conf /etc/pulse -t ~/pulse-backup/
$ sudo rm -r ~/.pulse ~/.asound* /etc/asound.conf
```

Note: Don't worry if some of these files did not exist on your system.

2. Ensure you have Adobe Flash & the necessary PulseAudio libraries and configuration utilities installed:

Code:

```
$ sudo apt-get install libasound2-plugins padevchooser libao-pulse libsdl1.2debian-pulseaudio
flashplugin-nonfree
```

3. Ensure the evil "libflashsupport" library is not installed:

Code:

```
$ sudo apt-get remove --purge libflashsupport flashplugin-nonfree-extrasound
```

4. Open System/Preferences/Sound. In the Devices section, ensure that all "Sound playback" options are set to Autodetect. Set the "Sound capture" item to "ALSA", or the appropriate hardware definition. Close the application when you're finished.

Note: Choosing PulseAudio for sound capture may result in crashes, so you are advised to choose the "direct" ALSA device instead.

5. Open the PulseAudio Volume Control application ("pavucontrol", or you can launch "Applications/Sound & Video/PulseAudio Device Chooser" and select Volume Control from this applet's menu). In the Output Devices section you will see a listing of the playback devices available on your system. Right-click on the entry that you desire to be made the default playback device on your system and enable the "Default" checkmark. Similarly, navigate to Input Devices, then right-click on the device you wish to set as your default input device (microphone), and ensure the "Default" setting is checked. Close the application when you're finished.

Note: If you are greeted with the error "Connection failed: Connection refused", manually launch PulseAudio before opening the PulseAudio Volume Control application:

Code:

```
$ pulseaudio & pavucontrol
```

6. Continue to Part B if you are running Hardy Heron (8.04), or Part C if you are running Intrepid Ibex (8.10).

Part B: Hardy Heron (8.04)

Follow the steps in this section only if you are running the Hardy Heron release.

Disclaimer: My PPA contains the necessary packages backported from Intrepid for: PulseAudio, ALSA, Flash 10 and nspluginwrapper. If you upgrade to a newer distribution, or Hardy receives official updates for any of these packages, you will not experience any issues.

0. 64-bit users only: Install "getlibs" and some extra 32-bit libraries that are required for Flash 10 & Skype to function properly:

Code:

```
$ wget http://www.boundlessupremacy.com/Cappy ... bs-all.deb && sudo dpkg -i getlibs-all.deb && rm getlibs-all.deb
```

```
$ sudo getlibs -p libnss3-1d libnspr4-0d libcurl3 libasound2-plugins
```

1. Edit /etc/apt/sources.list:

Code:

```
$ gksudo gedit /etc/apt/sources.list
```

If they don't already exist, add the following lines to the end of this file and save:

Code:

```
# PulseAudio Fixes - http://ubuntuforums.org/showthread.php?t=789578
```

```
deb http://ppa.launchpad.net/psyke83/Ultimate Edition hardy main
```

```
deb-src http://ppa.launchpad.net/psyke83/Ultimate Edition hardy main
```

2. Update your repositories and upgrade packages:

Code:

```
$ sudo apt-get update && sudo apt-get dist-upgrade
```

Note: You may be warned that some packages cannot be authenticated - that's normal. Simply press "y" to confirm installation of unsigned packages.

3. Enable the PulseAudio ALSA plugins:

Code:

```
$ asoundconf set-pulseaudio
```

Note: Please wait until you have upgraded packages in the previous step before running this command. My packages have a patch for "asoundconf" to ensure it enables the PulseAudio plugins correctly.

4. Log out & back in for changes to take effect!

Part C: Intrepid Ibex (8.10)

Follow the steps in this section only if you are running the Intrepid Ibex release.

Disclaimer: Currently there are no updated packages for Intrepid in my PPA (except for an updated nspluginwrapper package for i386 users). I have decided to keep this step in the guide in case I upload any important updates that will not make it into the official repositories. I will never upload any "risky" packages (i.e. highly untested backports), only upgrades that seem compelling and relatively stable.

1. Edit /etc/apt/sources.list:

Code:

```
$ gksudo gedit /etc/apt/sources.list
```

If they don't already exist, add the following lines to the end of this file and save:

Code:

```
# PulseAudio Fixes - http://ubuntuforums.org/showthread.php?t=789578
```

```
deb http://ppa.launchpad.net/psyke83/Ultimate Edition intrepid main
```

```
deb-src http://ppa.launchpad.net/psyke83/Ultimate Edition intrepid main
```

2. Update your repositories and upgrade packages:

Code:

```
$ sudo apt-get update && sudo apt-get dist-upgrade
```

Note: You may be warned that some packages cannot be authenticated - that's normal. Simply press "y" to confirm installation of unsigned packages.

3. Ensure that your sound card's PCM mixer is not muted or set to 0% volume (this appears to be a bug in Intrepid):

Code:

```
$ alsamixer -Dhw
```

Note: When the PulseAudio ALSA plugins are active, you must explicitly specify your hardware device in alsamixer (marked in blue above), otherwise it will open the PulseAudio mixer.

4. Log out & back in for changes to take effect!

Appendix A - General Troubleshooting

This section will outline some general troubleshooting steps you can perform to help identify your problem, and the information I need to help with your issues:

1. Close all applications that may be accessing the sound card.
2. Open the "PulseAudio Device Chooser" from Applications/Sound & Video. Click on the applet icon, and click "Volume Control...". Click on the "Playback" tab.
3. Launch the application you wish to test and allow it to play sound.
4. Check PulseAudio Volume Control's Playback tab and see if the application displays an entry while the application is (or should be) playing audio.

Here are the possible results within the PulseAudio Volume Control's Playback tab:

1. The application does play audio and does list an entry in the Playback tab;
- the application is using PulseAudio correctly.
2. The application does play audio and does not list an entry in the Playback tab;
- the application is either accessing your sound card directly or playing sound via ALSA's dmix device. This will prevent PulseAudio from working correctly & cause audio mixing errors.
3. The application does not play audio and does list an entry in the Playback tab;
- the application is using PulseAudio but there is a problem, such as: a bug in PulseAudio, a problem with your ALSA kernel module or libraries, or your PCM/Master volume is muted.
4. The application does not play audio and does not list an entry in the Playback tab;
- the application is trying either to access your sound card directly or to play sound via ALSA's dmix device, but the sound card is already in use. This is the opposite case to result B, which will also cause mixing errors.
5. The PulseAudio Volume Control application displays the error: "Connection failed: Connection refused";
- the PulseAudio daemon is not running.
6. Other (please specify).

Note: Unless the application you are testing is known to be incompatible with PulseAudio, you should always expect result "A" on a properly configured system.

If you require assistance with a particular application - or simply cannot get PulseAudio to work - provide the following information:

1. Your distribution version and architecture (e.g. Hardy Heron i386, Intrepid Ibex amd64, etc.).
2. A listing of your sound devices:

Code:

```
$ aplay -l
```

3. The verbose output from pulseaudio on your system:

Code:

```
$ pkill pulseaudio; sleep 2; pulseaudio -vv
```

4. If you are having a problem only with a specific application, specify the application's name and result you received from the instructions above (A-F).

Appendix B: Advanced PulseAudio Configuration

This appendix will explain some of the more advanced/technical features of PulseAudio.

Q. Where are the PulseAudio configuration utilities?

A. Providing you've followed this guide, you can gain access to all the utilities by launching "Applications/Sound & Video/PulseAudio Device Chooser". The applet's icon will appear in your notification tray - left-click to see the options.

The main applications you will want to check are the Manager (to see the current server status) and Volume Control (to manipulate the volume and stream settings).

Note: Please don't mess with any of the options until you have established a working configuration by following this guide - otherwise it'll be a nightmare to isolate your problem.

Q. How can I tell if an application is using PulseAudio successfully?

A. The application will give you result "A" from the troubleshooting steps of Appendix A.

Q. How can I change the default playback and/or recording device for my system?

A. Refer to Part A, Step 5.

Q. Is it possible to change to a different playback and/or recording device for an individual application?

A. Yes. Launch the desired application and play some sound (or begin recording), and open the PulseAudio Volume Control. Click the Playback or Recording tab (depending on which you want to change) and right-click on the desired application's entry. Choose the option "Move Stream..." and select the desired output device.

Note: The volume level, sink (playback) and source (recording) settings will be saved automatically for every application you run, so PulseAudio should remember your settings. If you wish to view or delete these saved settings, they're stored in the file "`~/pulse/volume-restore.table`".

Q. If I plug in my USB/Bluetooth headset, my external speakers no longer work!

A. This is normal behaviour, as PulseAudio supports "hotplugging" of audio devices. If you plug in a new device, PulseAudio may choose it as the default sink (audio device). Also see Part A, Step 5.

Q. PulseAudio is working correctly, but I am noticing some stuttering on my system. Is there anything I can do to help?

A. Edit the file `/etc/pulse/daemon.conf`:

Code:

```
$ gksudo gedit /etc/pulse/daemon.conf
```

Find the following lines (usually at the bottom):

Code:

```
default-fragments = 8
```

```
default-fragment-size-msec = 10
```

Try experimenting with different values for both of these entries. I can't tell you what values are optimal for your system, as each sound card has different buffer sizes and characteristics - therefore you'll need to use trial & error. The default fragment amount and size used by an untweaked PulseAudio installation is 4 and 25, respectively.

Note 1: you must restart pulseaudio for any configuration changes to take effect.

Note 2: If your system was stuttering in versions of Ultimate Edition prior to Hardy, then you could be

suffering from an ALSA kernel issue - these instructions probably won't help.

Q. I'm unhappy about the audio quality / CPU usage of PulseAudio. How do I change this?

A. For the record, PulseAudio should offer superior quality audio playback than ALSA's by default, as PulseAudio uses a higher quality audio resampler. This also means that PulseAudio may use more CPU usage compared to ALSA, unfortunately. If you wish to change the resampler:

Edit `/etc/pulse/daemon.conf`:

Code:

```
$ gksudo gedit /etc/pulse/daemon.conf
```

Find the following line:

Code:

```
resample-method = speex-float-l1
```

You can change the resampler to any of the following, listed in descending order, from highest quality to lowest quality (and therefore, CPU usage):

Quote:

```
src-sinc-best-quality, src-sinc-medium-quality, src-sinc-fastest, speex-float-{10-0}, speex-fixed-{10-0}, ffmpeg, src-zero-order-hold, src-linear, trivial
```

Appendix C: Application Compatibility Guide

This appendix will explain how to configure specific applications that may require manual configuration to work with PulseAudio.

OSS applications: You need to launch the application using the "padsdp" wrapper. For more

information, see "man padsp".

Skype: Open Skype's Options, then go to Sound Devices. You need to set "Sound Out" and "Ringing" to the "pulse" device, and set "Sound In" to the hardware definition of your microphone. For example, my laptop's microphone is defined as "plughw:182801DBICH4,0".

WINE: Open the Wine Configuration application ("winecfg"). On the Audio tab, choose the ALSA driver, and leave everything else to default. If your sound stutters, choose the OSS driver instead, and use the "padsp" wrapper to launch the wine executable (via the terminal, or edit your shortcuts).

Audacity (updated 11/12): From the PerfectSetup page:

Quote:

Audacity has now been packaged with a proper "alsa: pulse" device listed, in a ppa for Ultimate Edition intrepid. See <https://launchpad.net/~diwic/+archive>

Everything else: refer to the Third Party Applications section of the upstream PerfectSetup wiki page.

Note: Many of the instructions on the PerfectSetup page are either outdated, too distro-agnostic to work on Ultimate Edition, or have been obsoleted by the fixes in this guide. You should ignore the advice

for: ALSA applications, Flash, ESOUND, GNOME, GStreamer Applications, SDL, and Skype.

Appendix D: System-Wide Equalizer

In this section, we will configure PulseAudio to use equalized output, which is especially useful for laptops speakers which have very poor frequency response. If your sound is "tinny" or distorted at high ranges, then this will most likely enhance the quality of your audio.

Warning 1: Do not attempt to set up the equalizer until you have followed the other parts of this guide and verified that PulseAudio works correctly using "unmodified" output.

Warning 2: The equalizer may not work on 64 bit systems, as extra 32-bit libraries could be required. If it doesn't work, simply revert the changes you made.

1. Install the required LADSPA plugins and tools:

Code:

```
$ sudo apt-get install swh-plugins ladspa-sdk
```

2. Edit ~/.asoundrc:

Code:

```
$ gedit ~/.asoundrc
```

Append the following text to the bottom of this file and save:

Code:

```
pcm.equalized {  
  type plug  
  slave.pcm "equalizer";  
}
```

```
pcm.equalizer {  
  type ladspa
```

```
# The output from the EQ can either go direct to a hardware device  
# (if you have a hardware mixer, e.g. SBLive/Audigy) or it can go  
# to the software mixer shown here.
```

```
slave.pcm "plughw"
```

```
#slave.pcm "plug:dmix"
```

```
# Sometimes you may need to specify the path to the plugins,  
# especially if you've just installed them. Once you've logged  
# out/restarted this shouldn't be necessary, but if you get errors  
# about being unable to find plugins, try uncommenting this.
```

```
path "/usr/lib/ladspa"
```

```
plugins [
```

```
{  
  label mbeq
```

```
id 1197
input {
#this setting is here by example, edit to your own taste
#bands: 50hz, 100hz, 156hz, 220hz, 311hz, 440hz, 622hz, 880hz,
# 1250hz, 1750hz, 25000hz, 50000hz, 10000hz, 20000hz
#range: -70 to 30
controls [ -1 -1 -1 -1 -5 -10 -20 -17 -12 -7 -6 -5 -5 0 0 ]
}
}
]
}
```

Note 1: If you are on Intrepid, this file will not exist - that's ok. Simply paste into the new file and save normally.

Note 2: If you have multiple sound cards, the section of this text marked in blue may need to be modified slightly. If you have problems, ask in the thread.

3. Edit /etc/pulse/default.pa:

Code:

```
$ gksudo gedit /etc/pulse/default.pa
```

Find the following line marked in blue:

Code:

```
### Load audio drivers statically (it's probably better to not load
### these drivers manually, but instead use module-hal-detect --
### see below -- for doing this automatically)
#load-module module-alsa-sink
#load-module module-alsa-source device=hw:1,0
#load-module module-oss device="/dev/dsp" sink_name=output source_name=input
```

Change the line in blue to the following, and save:

Code:

```
load-module module-alsa-sink device=equalized
```

Note: Be sure to also remove the comment (#) at the beginning of the line.

4. Log out & back in for changes to take effect!

That's all folks!

Howto: Alt Ctrl Del - Ultimate Edition Task Manager

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2638>

Damn I installed Ultimate Edition and the only complaint I had was no alt-ctrl-del for a task manager in Ultimate Edition.

To bind your system monitor to pop up when you use ctrl+al+del use the following commands in xterm/gnome-terminal:

remove the spaces between these "<" ">"

```
gconftool-2 -t str --set /apps/metacity/global_keybindings/run_command_9 "< Control << Alt >>Delete"
```

```
gconftool-2 -t str --set /apps/metacity/keybinding_commands/command_9 "gnome-system-monitor"
58 Cool Hacks
```

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=3155>

One of the great things about Linux is that there's usually more than one way to do something.

Chances are, the way you are doing a particular task right now isn't necessarily the best way either.

This collection of hacks is the result of a lot of effort: some are functional, some are fun, but they are all cool! So, start reading, and find out what you could be doing with Linux...

DISCLAIMER: Obviously, anything you do to your computer is your responsibility, and while we have made every effort to test these hacks, your mileage may vary. With most of them, it is highly unlikely that you can do any damage - but the expert tips carry a higher risk of something going wrong, especially if you don't follow the instructions carefully.

Table of contents [showhide]

- 1 Fix a wonky terminal
- 2 Creating Mozilla keywords
- 3 Running multiple X sessions
- 4 Faster browsing
- 5 Backup your website easily
- 6 Keeping your clock in time
- 7 Finding the biggest files
- 8 Nautilus shortcuts
- 9 Defrag your databases
- 10 Quicker emails
- 11 Parallelise your build
- 12 Save battery power
- 13 Wireless speed management
- 14 Unclog open ports
- 15 Faster Hard drives
- 16 Uptime on your hands
- 17 Grabbing a screenshot without X
- 18 Access your programs remotely
- 19 Making man pages useful
- 20 Talk to your doctor!
- 21 Super cow powers
- 22 Generating package relationship diagrams
- 23 Unmount busy drives
- 24 Text file conversion
- 25 Listing today's files only
- 26 Avoid common mistypes and long commands
- 27 Alter Mozilla's secret settings
- 28 A backdrop of stars
- 29 Open an SVG directly
- 30 Editing without an editor
- 31 Backup selected files only
- 32 Merging columns in files
- 33 Case sensitivity
- 34 Macros in Emacs
- 35 Replacing same text in multiple files
- 36 Simple spam killing
- 37 Read OOo docs without OOo
- 38 Find and execute
- 39 Use the correct whois server
- 40 Where did that drive mount?
- 41 Autorun USB devices
- 42 Rename and resize images
- 43 Secure logout
- 44 Transferring files without ftp or scp
- 45 Backing up a Debian package list
- 46 Hardening ssh
- 47 Stop replying to pings
- 48 Slow down ping rates
- 49 Clean up KDE on logout
- 50 Password-less ssh
- 51 Using rsync over ssh
- 52 Asset scanning
- 53 Backup your bootsector
- 54 Protect log files

55 Automatically encrypted connections

56 Eliminate suid binaries

57 Mac filtering Host AP

58 Check processes not run by you

[edit]

Fix a wonky terminal

* Difficulty: Easy

* Application: bash

We've all done it - accidentally used less or cat to list a file, and ended up viewing binary instead. This usually involves all sorts of control codes that can easily screw up your terminal display. There will be beeping. There will be funny characters. There will be odd colour combinations. At the end of it, your font will be replaced with hieroglyphics and you don't know what to do. Well, bash is obviously still working, but you just can't read what's actually going on! Send the terminal an initialisation command:

```
reset
```

and all will be well again.

[edit]

Creating Mozilla keywords

* Difficulty: Easy

* Application: Firefox/Mozilla

A useful feature in Konqueror is the ability to type gg onion to do a Google search based on the word onion. The same kind of functionality can be achieved in Mozilla by first clicking on Bookmarks>Manage Bookmarks and then Add a New Bookmark. Add the URL as:

```
http://www.google.com/search?q=%s
```

Now select the entry in the bookmark editor and click the Properties button. Now enter the keyword as gg (or this can be anything you choose) and the process is complete. The %s in the URL will be replaced with the text after the keyword. You can apply this hack to other kinds of sites that rely on you passing information on the URL.

Alternatively, right-click on a search field and select the menu option "Add a Keyword for this Search...". The subsequent dialog will allow you to specify the keyword to use.

[edit]

Running multiple X sessions

* Difficulty: Easy

* Application: X

If you share your Linux box with someone and you are sick of continually logging in and out, you may be relieved to know that this is not really needed. Assuming that your computer starts in graphical mode (runlevel 5), by simultaneously pressing the keys Control+Alt+F1 - you will get a login prompt. Insert your login and password and then execute:

```
startx -- :1
```

to get into your graphical environment. To go back to the previous user session, press Ctrl+Alt+F7, while to get yours back press Ctrl+Alt+F8.

You can repeat this trick: the keys F1 to F6 identify six console sessions, while F7 to F12 identify six X sessions. Caveat: although this is true in most cases, different distributions can implement this feature in a different way.

[edit]

Faster browsing

* Difficulty: Easy

* Application: KDE

In KDE 3.2, a little-known but useful option has been added to speed up your web browsing experience. Start the KDE Control Center and choose System > KDE performance from the sidebar. You can now select to preload Konqueror instances. Effectively, this means that Konqueror is run

on startup, but kept hidden until you try to use it. When you do, it pops up almost instantaneously.

Bonus!

[edit]

Backup your website easily

* Difficulty: Easy

* Application: Backups

If you want to back up a directory on a computer and only copy changed files to the backup computer instead of everything with each backup, you can use the rsync tool to do this. You will need an account on the remote computer that you are backing up from. Here is the command:

```
rsync -vare ssh jono@192.168.0.2:/home/jono/importantfiles/* /home/jono/backup/
```

Here we are backing up all of the files in /home/jono/importantfiles/ on 192.168.0.2 to /home/jono/backup on the current machine.

[edit]

Keeping your clock in time

* Difficulty: Easy

* Application: NTP

If you find that the clock on your computer seems to wander off the time, you can make use of a special NTP tool to ensure that you are always synchronised with the kind of accuracy that only people that wear white coats get excited about. You will need to install the ntpdate tool that is often included in the NTP package, and then you can synchronise with an NTP server:

```
ntpdate ntp.blueyonder.co.uk
```

A list of suitable NTP servers is available at www.eecis.udel.edu/~mills/ntp/clock1b.html. If you modify your boot process and scripts to include this command you can ensure that you are perfectly in time whenever you boot your computer. You could also run a cron job to update the time.

[edit]

Finding the biggest files

* Difficulty: Easy

* Application: Shell

A common problem with computers is when you have a number of large files (such as audio/video clips) that you may want to get rid of. You can find the biggest files in the current directory with:

```
ls -lSrhc
```

(the r causes the large files to be listed at the end, the h gives human readable output (MB and such))

You could also search for the biggest MP3/MPEGs:

```
ls -lSrhc *.mp*
```

You can also look for the largest directories with:

```
du -kx | egrep -v "\./+/" | sort -n
```

[edit]

Nautilus shortcuts

* Difficulty: Easy

* Application: Nautilus

Although most file managers these days are designed to be used with the mouse, it's also useful to be able to use the keyboard sometimes. Nautilus has a few keyboard shortcuts that can have you flying through files:

* Open a location - Ctrl+L

* Open Parent folder - Ctrl+Up

* Arrow keys navigate around current folder.

You can also customise the file icons with 'emblems'. These are little graphical overlays that can be applied to individual files or groups. Open the Edit > Backgrounds and Emblems menu item, and drag-and-drop the images you want.

[edit]

Defrag your database

* Difficulty: Easy

* Application: MySQL

Whenever you change the structure of a MySQL database, or remove a lot of data from it, the files can become fragmented resulting in a loss of performance, particularly when running queries. Just remember any time you change the database to run the optimiser:

```
mysqlcheck -o <databasename>
```

You may also find it worth your while to defragment your database tables regularly if you are using VARCHAR fields: these variable-length columns are particularly prone to fragmentation.

[edit]

Quicker emails

* Difficulty: Easy

* Application: KMail

Can't afford to waste three seconds locating your email client? Can't be bothered finding the mouse under all those gently rotting mountains of clutter on your desk? Whatever you are doing in KDE, you are only a few keypresses away from sending a mail. Press Alt+F2 to bring up the 'Run command' dialog. Type:

```
mailto:plop@ploppypants.com
```

Press return and KMail will automatically fire up, ready for your words of wisdom. You don't even need to fill in the entire email address. This also works for Internet addresses: try typing www.slashdot.org to launch Konqueror.

[edit]

Parallelise your build

* Difficulty: Easy

* Application: GCC

If you're running a multiprocessor system (SMP) with a moderate amount of RAM, you can usually see significant benefits by performing a parallel make when building code. Compared to doing serial builds when running make (as is the default), a parallel build is a vast improvement. To tell make to allow more than one child at a time while building, use the -j switch:

```
make -j4; make -j4 modules
```

[edit]

Save battery power

* Difficulty: Intermediate

* Application: hdparm

You are probably familiar with using hdparm for tuning a hard drive, but it can also save battery life on your laptop, or make life quieter for you by spinning down drives.

```
hdparm -y /dev/hdb
```

```
hdparm -Y /dev/hdb
```

```
hdparm -S 36 /dev/hdb
```

In order, these commands will: cause the drive to switch to Standby mode, switch to Sleep mode, and finally set the Automatic spindown timeout. This last includes a numeric variable, whose units are blocks of 5 seconds (for example, a value of 12 would equal one minute).

Incidentally, this habit of specifying spindown time in blocks of 5 seconds should really be a contender for a special user-friendliness award - there's probably some historical reason for it, but we're stumped. Write in and tell us if you happen to know where it came from!

[edit]

Wireless speed management

* Difficulty: Intermediate

* Application: iwconfig

The speed at which a piece of radio transmission/receiver equipment can communicate with another depends on how much signal is available. In order to maintain communications as the available signal fades, the radios need to transmit data at a slower rate. Normally, the radios attempt to work out the available signal on their own and automatically select the fastest possible speed. In fringe areas with a barely adequate signal, packets may be needlessly lost while the radios continually renegotiate the link speed. If you can't add more antenna gain, or reposition your equipment to achieve a better enough signal, consider forcing your card to sync at a lower rate. This will mean fewer retries, and can be substantially faster than using a continually flip-flopping link. Each driver has its own method for setting the link speed. In Linux, set the link speed with iwconfig:

```
iwconfig eth0 rate 2M
```

This forces the radio to always sync at 2Mbps, even if other speeds are available. You can also set a particular speed as a ceiling, and allow the card to automatically scale to any slower speed, but go no faster. For example, you might use this on the example link above:

```
iwconfig eth0 rate 5.5M auto
```

Using the auto directive this way tells the driver to allow speeds up to 5.5Mbps, and to run slower if necessary, but will never try to sync at anything faster. To restore the card to full auto scaling, just specify auto by itself:

```
iwconfig eth0 rate auto
```

Cards can generally reach much further at 1Mbps than they can at 11Mbps. There is a difference of 12dB between the 1Mbps and 11Mbps ratings of the Orinoco card - that's four times the potential distance just by dropping the data rate!

[edit]

Unclog open ports

* Difficulty: Intermediate

* Application: netstat

Generating a list of network ports that are in the Listen state on a Linux server is simple with netstat:

```
root@catlin:~# netstat -lnp
```

Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name

```
tcp 0 0 0.0.0.0:5280 0.0.0.0:* LISTEN 698/perl
```

```
tcp 0 0 0.0.0.0:80 0.0.0.0:* LISTEN 217/httpd
```

```
tcp 0 0 10.42.3.2:53 0.0.0.0:* LISTEN 220/named
```

```
tcp 0 0 10.42.4.6:53 0.0.0.0:* LISTEN 220/named
```

```
tcp 0 0 127.0.0.1:53 0.0.0.0:* LISTEN 220/named
```

```
tcp 0 0 0.0.0.0:22 0.0.0.0:* LISTEN 200/sshd
```

```
udp 0 0 0.0.0.0:32768 0.0.0.0:* 220/named
```

```
udp 0 0 10.42.3.2:53 0.0.0.0:* 220/named
```

```
udp 0 0 10.42.4.6:53 0.0.0.0:* 220/named
```

```
udp 0 0 127.0.0.1:53 0.0.0.0:* 220/named
```

```
udp 0 0 0.0.0.0:67 0.0.0.0:* 222/dhcpd
```

```
raw 0 0 0.0.0.0:1 0.0.0.0:* 7 222/dhcpd
```

That shows you that PID 698 is a Perl process that is bound to port 5280. If you're not root, the system won't disclose which programs are running on which ports.

[edit]

Faster Hard drives

* Difficulty: Expert

* Application: hdparm

You may know that the hdparm tool can be used to speed test your disk and change a few settings. It can also be used to optimise drive performance, and turn on some features that may not be enabled by default. Before we start though, be warned that changing drive options can cause data corruption, so back up all your important data first. Testing speed is done with:

hdparm -Tt /dev/hda

You'll see something like:

```
/dev/hda:
```

```
Timing buffer-cache reads: 128 MB in 1.64 seconds =78.05 MB/sec
```

```
Timing buffered disk reads: 64 MB in 18.56 seconds = 3.45MB/sec
```

Now we can try speeding it up. To find out which options your drive is currently set to use, just pass hdparm the device name:

```
hdparm /dev/hda
```

```
/dev/hda:
```

```
multcount = 16 (on)
```

```
I/O support = 0 (default 16-bit)
```

```
unmaskirq = 0 (off)
```

```
using_dma = 0 (off)
```

```
keepsettings = 0 (off)
```

```
readonly = 0 (off)
```

```
readahead = 8 (on)
```

```
geometry = 40395/16/63, sectors = 40718160, start = 0
```

This is a fairly default setting. Most distros will opt for safe options that will work with most hardware. To get more speed, you may want to enable dma mode, and certainly adjust I/O support. Most modern computers support mode 3, which is a 32-bit transfer mode that can nearly double throughput. You might want to try

```
hdparm -c3 -d1/dev/hda
```

Then rerun the speed check to see the difference. Check out the modes your hardware will support, and the hdparm man pages for how to set them.

[edit]

Uptime on your hands

* Difficulty: Expert

* Application: Perl

In computing, wasted resources are resources that could be better spent helping you. Why not run a process that updates the titlebar of your terminal with the current load average in real-time, regardless of what else you're running?

Save this as a script called tl, and save it to your ~/bin directory:

```
#!/usr/bin/perl -w
```

```
use strict;
```

```
#++;
```

```
my $host=`/bin/hostname`;
```

```
chomp $host;
```

```
while(1) {
```

```
open(LOAD, "/proc/loadavg") || die "Couldn't open /proc/loadavg: $!\n";
```

```
my @load=split(/ /, <LOAD>);
```

```
close(LOAD);
```

```
print "\033]0;";
```

```
print "$host: $load[0] $load[1] $load[2] at ", scalar(localtime);
```

```
print "\007";
```

```
sleep 2;
```

```
}
```

When you'd like to have your titlebar replaced with the name, load average, and current time of the machine you're logged into, just run tl&. It will happily go on running in the background, even if you're running an interactive program like Vim.

[edit]

Grabbing a screenshot without X

* Difficulty: Easy

* Application: Shell

There are plenty of screen-capture tools, but a lot of them are based on X. This leads to a problem when running an X application would interfere with the application you wanted to grab - perhaps a game or even a Linux installer. If you use the venerable ImageMagick import command though, you can grab from an X session via the console. Simply go to a virtual terminal (Ctrl+Alt+F1 for example) and enter the following:

```
chvt 7; sleep 2; import -display :0.0 -window root sshot1.png; chvt 1;
```

The chvt command changes the virtual terminal, and the sleep command gives it a while to redraw the screen. The import command then captures the whole display and saves it to a file before the final chvt command sticks you back in the virtual terminal again. Make sure you type the whole command on one line.

This can even work on Linux installers, many of which leave a console running in the background - just load up a floppy/CD with import and the few libraries it requires for a first-rate run-anywhere screen grabber.

[edit]

Access your programs remotely

* Difficulty: Easy

* Application: X

If you would like to lie in bed with your Linux laptop and access your applications from your Windows machine, you can do this with SSH. You first need to enable the following setting in /etc/ssh/sshd_config:

```
X11Forwarding yes
```

We can now run The GIMP on 192.168.0.2 with:

```
ssh -X 192.168.0.2 gimp
```

[edit]

Making man pages useful

* Difficulty: Easy

* Application: man

If you are looking for some help on a particular subject or command, man pages are a good place to start. You normally access a man page with man <command>, but you can also search the man page descriptions for a particular keyword. As an example, search for man pages that discuss logins:

```
man -k login
```

When you access a man page, you can also use the forward slash key to search for a particular word within the man page itself. Simply press / on your keyboard and then type in the search term.

[edit]

Talk to your doctor!

* Difficulty: Easy

* Application: Emacs

To say that Emacs is just a text editor is like saying that a Triumph is just a motorcycle, or the World Cup is just some four-yearly football event. True, but simplified juuuust a little bit. An example? Open the editor, press the Esc key followed by X and then enter in doctor: you will be engaged in a surreal conversation by an imaginary and underskilled psychotherapist. And if you want to waste your time in a better way

```
Esc-X tetris
```

will transform your 'editor' into the old favourite arcade game.

Does the madness stop there? No! Check out your distro's package list to see what else they've bundled for Emacs: here at LXF Towers we've got chess, Perl integration, IRC chat, French translation, HTML conversion, a Java development environment, smart compilation, and even something called a "semantic bovinator". We really haven't the first clue what that last one does, but we dare you to try it out anyway! (Please read the disclaimer first!)

[edit]

Super cow powers

* Difficulty: Easy

* Application: Debian

A strange but endearing hidden feature within the highly regarded apt-get tool in Debian is its secret cow powers. Type the following command to experience the wrath of the super cow powers:

```
apt-get moo
```

Some people really have too much time on their hands...

[edit]

Generating package relationship diagrams

* Difficulty: Easy

* Application: Debian

The most critical part of the Debian system is the ability to install a package and have the dependencies satisfied automatically. If you would like a graphical representation of the relationships between these packages (this can be useful for seeing how the system fits together), you can use the Graphviz package from Debian non-free (apt-get install graphviz) and the following command:

```
apt-cache dotty > debian.dot
```

The command generated the graph file which can then be loaded into dotty:

```
dotty debian.dot
```

[edit]

Unmount busy drives

* Difficulty: Easy

* Application: bash

You are probably all too familiar with the situation - you are trying to unmount a drive, but keep getting told by your system that it's busy. But what application is tying it up? A quick one-liner will tell you:

```
lsof +D /mnt/windows
```

This will return the command and process ID of any tasks currently accessing the /mnt/windows directory. You can then locate them, or use the kill command to finish them off.

[edit]

Text file conversion

* Difficulty: Easy

* Application: recode

recode is a small utility that will save you loads of effort when using text files created on different platforms. The primary source of discontent is line breaks. In some systems, these are denoted with a line-feed character. In others, a carriage return is used. In still more systems, both are used. The end result is that if you are swapping text from one platform to another, you end up with too many or too few line breaks, and lots of strange characters besides.

However, the command parameters of recode are a little arcane, so why not combine this hack with HACK 26 in this feature, and set up some useful aliases:

```
alias dos2unix='recode dos/CR-LF..11'
```

```
alias unix2win='recode 11..windows-1250'
```

```
alias unix2dos='recode 11..dos/CR-LF'
```

There are plenty more options for recode - it can actually convert between a whole range of character sets. Check out the man pages for more information.

[edit]

Listing today's files only

* Difficulty: Easy

* Application: Various

You are probably familiar with the problem. Sometime earlier in the day, you created a text file, which now is urgently required. However, you can't remember what ridiculous name you gave it, and being a typical geek, your home folder is full of 836 different files. How can you find it? Well, there are various ways, but this little tip shows you the power of pipes and joining together two powerful shell commands:

```
ls -al --time-style=+%D | grep `date +%D`
```

The parameters to the ls command here cause the datestamp to be output in a particular format. The cunning bit is that the output is then passed to grep. The grep parameter is itself a command (executed because of the backticks), which substitutes the current date into the string to be matched. You could easily modify it to search specifically for other dates, times, file sizes or whatever.

Combine it with HACK 26 to save typing!

[edit]

Avoid common mistypes and long commands

* Difficulty: Easy

* Application: Shell

The alias command is useful for setting up shortcuts for long commands, or even more clever things. From HACK 25, we could make a new command, lsnew, by doing this:

```
alias lsnew="ls -al --time-style=+%D | grep `date +%D` "
```

But there are other uses of alias. For example, common mistyping mistakes. How many times have you accidentally left out the space when changing to the parent directory? Worry no more!

```
alias cd..="cd .."
```

Alternatively, how about rewriting some existing commands?

```
alias ls="ls -al"
```

saves a few keypresses if, like us, you always want the complete list.

To have these shortcuts enabled for every session, just add the alias commands to your user .bashrc file in your home directory.

[edit]

Alter Mozilla's secret settings

* Difficulty: Easy

* Application: Mozilla

If you find that you would like to change how Mozilla works but the preferences offer nothing by way of clickable options that can help you, there is a special mode that you can enable in Mozilla so that you can change anything. To access it, type this into the address bar:

```
about:config
```

You can then change each setting that you are interested in by changing the Value field in the table.

Other interesting modes include general information (about:), details about plugins (about:plugins), credits information (about:credits) and some general wisdom (about:mozilla).

[edit]

A backdrop of stars

* Difficulty: Easy

* Application: KStars

You may already have played with KStars, as it was included with the astronomy software that featured in LXF50's Roundup; but how about creating a KStars backdrop image that's updated every time you start up?

KStars can be run with the --dump switch, which dumps out an image from your startup settings, but doesn't load the GUI at all. You can create a script to run this and generate a desktop image, which will change every day (or you can just use this method to generate images).

Run KStars like this:

```
kstars --dump --width 1024 --height 768 --filename = ~/kstarsback.png
```

You can add this to a script in your `~/kde/Autostart` folder to be run at startup. Find the file in Konqueror, drag it to the desktop and select 'Set as wallpaper' to use it as a randomly generated backdrop.

[edit]

Open an SVG directly

* Difficulty: Easy

* Application: Inkscape

You can run Inkscape from a shell and immediately edit a graphic directly from a URL. Just type:

```
inkscape http://www.somehost.com/graphic.svg
```

Remember to save it as something else though!

[edit]

Editing without an editor

* Difficulty: Intermediate

* Application: Various

Very long files are often hard to manipulate with a text editor. If you need to do it regularly, chances are you'll find it much faster to use some handy command-line tools instead, like in the following examples.

To print columns eg 1 and 3 from a file file1 into file2, we can use awk:

```
awk '{print $1, $3}' file1 > file2
```

To output only characters from column 8 to column 15 of file1, we can use cut:

```
cut -c 8-15 file1 > file2
```

To replace the word word1 with the word word2 in the file file1, we can use the sed command:

```
sed "s/word1/word2/g" file1 > file2
```

This is often a quicker way to get results than even opening a text editor.

[edit]

Backup selected files only

* Difficulty: Intermediate

* Application: tar

Want to use tar to backup only certain files in a directory? Then you'll want to use the `-T` flag as follows. First, create a file with the file you want to backup:

```
cat >> /etc/backup.conf
```

```
# /etc/passwd
```

```
# /etc/shadow
```

```
# /etc/yp.conf
```

```
# /etc/sysctl.conf
```

```
EOF
```

Then run tar with the `-T` flag pointing to the file just created:

```
tar -cjf bck-etc-`date +%Y-%m-%d`.tar.bz2 -T /etc/backup.conf
```

Now you have your backup.

[edit]

Merging columns in files

* Difficulty: Intermediate

* Application: bash

While splitting columns in files is easy enough, merging them can be complicated. Below is a simple shell script that does the job:

```
#!/bin/sh
```

```
length=$(wc -l $1 | awk '{print $1}')
```

```
count=1
```

```
[ -f $3 ] && echo "Optionally removing $3" && rm -i $3
```

```
while [ "$count" -le "$length" ]; do
```

```
a=`head -${count} $1 | tail -1`  
b=`head -${count} $2 | tail -1`  
echo "$a $b" >> $3  
count=`expr $count + 1`  
done
```

Give to this script the name `merge.sh` and make it executable with:

```
chmod u+x merge.sh
```

Now, if you want to merge the columns of `file1` and `file2` into `file3`, it's just matter of executing `/path/to/merge.sh file1 file2 file3`

where `/path/to/` has to be replaced with the location of `merge.sh` in your filesystem.

[edit]

Case sensitivity

* Difficulty: Intermediate

* Application: bash

Despite the case of a word not making any difference to other operating systems, in Linux "Command" and "command" are different things. This can cause trouble when moving files from Windows to Linux. `tr` is a little shell utility that can be used to change the case of a bunch of files.

```
#!/bin/sh
```

```
for i in `ls -l`; do
```

```
file1=`echo $i | tr [A-Z] [a-z]`
```

```
mv $i $file1 2>/dev/null
```

```
done
```

By executing it, `FILE1` and `file2` will be renamed respectively `file1` and `file2`.

[edit]

Macros in Emacs

* Difficulty: Intermediate

* Application: Emacs

When editing files, you will often find that the tasks are tedious and repetitive, as LXF's Production Editor knows only too well! To spare your time, you can record a macro. In Emacs, you will have to go through the following steps:

1. Press `Ctrl+X` to start recording.
2. Insert all the keystrokes and commands that you want
3. Press `Ctrl+X` to stop when you're done.

Now, you can execute that with

```
Ctrl-u <number> Ctrl-x e
```

where `<number>` is the number of times you want to execute the macro. If you enter a value of 0, the macro will be executed until the end of the file is reached. `Ctrl-x e` is equivalent to `Ctrl-u 1 Ctrl-x e`.

[edit]

Replacing same text in multiple files

* Difficulty: Intermediate

* Application: find/Perl

If you have text you want to replace in multiple locations, there are several ways to do this. To replace the text `Windows` with `Linux` in all files in current directory called `test[something]` you can run this:

```
perl -i -pe 's/Windows/Linux/;' test*
```

To replace the text `Windows` with `Linux` in all text files in current directory and down you can run this:

```
find . -name '*.txt' -print | xargs perl -pi -e's/Windows/Linux/ig' *.txt
```

Or if you prefer this will also work, but only on regular files:

```
find -type f -name '*.txt' -print0 | xargs --null perl -pi -e 's/Windows/Linux/'
```

Saves a lot of time and has a high guru rating!

[edit]

Simple spam killing

* Difficulty: Intermediate

* Application: KMail

Spam, or unsolicited bulk email, is such a widespread problem that almost everyone has some sort of spam protection now, out of necessity. Most ISPs include spam filtering, but it isn't set to be too aggressive, and most often simply labels the spam, but lets it through (ISPs don't want to be blamed for losing your mails). The result is that, while you may have anti-spam stuff set up on the client-side, you can make its job easier by writing a few filters to remove the spam that's already labelled as such. The label is included as a header. In KMail, you can just create a quick filter to bin your mail, or direct it to a junk folder. The exact header used will depend on the software your ISP is using, but it's usually something like X-Spam-Flag = YES for systems like SpamAssassin. Simply create a filter in KMail, choose Match Any of the Following and type in the header details and the action you require. Apply the filter to incoming mail, and you need never be troubled by about half the volume of your spam ever again.

[edit]

Read OOo docs without OOo

* Difficulty: Intermediate

* Application: OpenOffice.org

Have you ever been left with an OOo document, but no OpenOffice.org in which to read it?

Thought you saved it out as plain text (.txt), but used the StarOffice .sxw format instead? The text can be rescued. Firstly, the sxw file is a zip archive, so unzip it:

```
unzip myfile.sxw
```

The file you want is called 'content.xml'. Unfortunately, it's so full of xml tags it's fairly illegible, so filter them out with some Perl magic:

```
cat content.xml | perl -p -e "s/<[^>]*>/ /g;s/\n/ /g;s/ +/ /;"
```

It may have lost lots of formatting, but at least it is now readable.

[edit]

Find and execute

* Difficulty: Intermediate

* Application: find

The find command is not only useful for finding files, but is also useful for processing the ones it finds too. Here is a quick example.

Suppose we have a lot of tarballs, and we want to find them all:

```
find . -name '*.gz'
```

will locate all the gzip archives in the current path. But suppose we want to check they are valid archives? The gunzip -vt option will do this for us, but we can cunningly combine both operations, using xargs:

```
find . -name '*.gz' | xargs gunzip -vt
```

[edit]

Use the correct whois server

* Difficulty: Intermediate

* Application: whois

The whois command is very useful for tracking down Internet miscreants and the ISPs that are supplying them with service. Unfortunately, there are many whois servers, and if you are querying against a domain name, you often have to use one which is specific to the TLD they are using.

However, there are some whois proxies that will automatically forward your query on to the correct server. One of these is available at <http://whois.geektools.com>

```
whois -h whois.geektools.com plop.info
```

[edit]

Where did that drive mount?

* Difficulty: Intermediate

* Application: bash

A common problem with people who have lots of mountable devices (USB drives, flash memory cards, USB key drives) is working out where that drive you just plugged in has ended up?

Practically all devices that invoke a driver - such as usb-storage - will dump some useful information in the logs. Try

```
dmesg | grep SCSI
```

This will filter out recognised drive specs from the dmesg output. You'll probably turn up some text like:

```
SCSI device sda: 125952 512-byte hdwr sectors (64 MB)
```

So your device is at sda.

[edit]

Autorun USB devices

* Difficulty: Expert

* Application: hotplug scripts

Want to run a specific application whenever a particular device is added? The USB hotplug daemon can help you! This service is notified when USB devices are added to the system. For devices that require kernel drivers, the hotplug daemon will call a script by the same name in /etc/hotplug/usb/, for example, a script called usb-storage exists there. You can simply add your own commands to the end of this script (or better still, tag a line at the end of it to execute a script elsewhere). Then you can play a sound, autosync files, search for pictures or whatever.

For devices that don't rely on kernel drivers, a lookup table is used matching the USB product and manufacturer ID. Many distros already set this up to do something, but you can customise these scripts pretty easily. See <http://jphoto.sourceforge.net/?selected=sync> for an example of what can be done.

[edit]

Rename and resize images

* Difficulty: Expert

* Application: bash

Fond of your new camera but can't put up with the terrible names? Do you want also to prepare them for publishing on the web? No problem, a simple bash script is what you need:

```
#!/bin/sh
```

```
counter=1
```

```
root=mypict
```

```
resolution=400x300
```

```
for i in `ls -l $1/*.jpg`; do
```

```
echo "Now working on $i"
```

```
convert -resize $resolution $i ${root}_${counter}.jpg
```

```
counter=`expr $counter + 1`
```

```
done
```

Save the script in a file called picturename.sh and make it executable with `chmod u+x picturename.sh`

and store it somewhere in your path. Now, if you have a bunch of .jpg files in the directory

/path/to/picdir, all you have to do is to execute

```
picturename.sh /path/to/picdir
```

and in the current directory you'll find mypict_1.jpg, mypict_2.jpg etc, which are the resized versions of your original ones. You can change the script according to your needs.

[edit]

Secure logout

* Difficulty: Easy

* Application: bash

When you are using a console on a shared machine, or indeed, just on your own desktop, you may find that when you logout, the screen still shows a trace of who was logged in and what you were doing. A lot of distros will clear the screen, but some don't. You can solve this by editing your `~/bash_logout` file and adding the command:

```
clear
```

You can add any other useful commands here too.

[edit]

Transferring files without ftp or scp

* Difficulty: Easy

* Application: netcat

Need to transfer a directory to another server but do not have FTP or SCP access? Well this little trick will help out using the netcat utility. On the destination server run:

```
nc -l -p 1234 | uncompress -c | tar xvfp -
```

And on the sending server run:

```
tar cfp - /some/dir | compress -c | nc -w 3 [destination] 1234
```

Now you can transfer directories without FTP and without needing root access.

[edit]

Backing up a Debian package list

* Difficulty: Easy

* Application: Debian

If you are running Debian and have lost track of which packages you are running, it could be useful to get a backup of your currently installed packages. You can get a list by running:

```
dpkg --get-selections > debianlist.txt
```

This will put the entire list in `debianlist.txt`. You could then install the same packages on a different computer with:

```
dpkg --set-selections < debianlist.txt
```

You should bear in mind that you would also need to copy over configuration files from `/etc` when copying your system to a new computer.

To actually install the selections, use:

```
apt-get -u dselect-upgrade.
```

[edit]

Hardening ssh

* Difficulty: Easy

* Application: ssh

Although SSH is a pretty secure way to connect to your server, there are two simple changes you can make that will boost its security even further. First, you almost certainly don't want people logging in directly as root - instead, they should logon as a normal user, then use the `su` command to switch over. You can change this simply in the `/etc/ssh/ssh_config` file by adding the line:

```
PermitRootLogin no
```

Now the only way to get root privileges is through `su`, which means crackers now need to break two passwords to get full access. While you are editing that file, find the line which says:

```
Protocol 2, 1
```

And change it to:

```
Protocol 2
```

This removes the option to fallback on the original SSH protocol, now considered very vulnerable.

[edit]

Stop replying to pings

* Difficulty: Easy

* Application: sysctl

While ping is a very useful command for discovering network topology, the disadvantage is that it does just that, and makes it easier for hackers on the network to target live servers. But you can tell Linux to ignore all pings - the server simply won't respond. There are a number of ways to achieve this, but the best is to use sysctl. To turn off ping replies:

```
sysctl -w net.ipv4.icmp_echo_ignore_all=1
```

To turn it back on, again use:

```
sysctl -w net.ipv4.icmp_echo_ignore_all=0
```

If turning off ping is too severe for you, take a look at the next hack.

[edit]

Slow down ping rates

* Difficulty: Easy

* Application: sysctl

You may want to keep the ability to reply to pings, but protect yourself from a form of attack known as a 'ping flood'. So how can you manage such a feat? The easiest way is to slow down the rate at which the server replies to pings. They are still valid, but won't overload the server:

```
sysctl -w net.ipv4.icmp_echo_reply_rate=10
```

This slows the rate at which replies are sent to a single address.

[edit]

Clean up KDE on logout

* Difficulty: Easy

* Application: bash

On Windows there are plenty of programs that do stuff like clean out your web cache, remove temporary files and all sorts of other stuff when you logout. Wouldn't it be cool to do this on Linux too? With KDE, you don't need to even install any new software, as the startkde script will automatically run scripts you put in special places.

First, you need to create a directory called shutdown in your .kde directory:

```
mkdir /home/username/.kde/shutdown
```

Now create a script to do any stuff you like on shutdown. Here is an example:

```
#!/bin/bash
```

```
#clear up temp folder
```

```
rm -rf ~/tmp/*
```

```
#clear out caches
```

```
rm -rf ~/.ee/mini/*
```

```
rm -rf ~/.kde/share/cache/http/*
```

```
# delete konqueror form completions
```

```
rm ~/.kde/share/apps/khtml/formcompletions
```

Now make sure you set the correct permissions:

```
chmod ug+x ~/.kde/shutdown/cleanup.sh
```

(or whatever you called it). As well as cleaning up sensitive files, you can also have global shutdown scripts for all users, by placing the script in your default KDE folder, in a subfolder called shutdown. To find out which is your default KDE directory, try:

```
kde-config --path exe
```

[edit]

Password-less ssh

* Difficulty: Intermediate

* Application: ssh

Tired of typing your password every time you log into the server? ssh also supports keys, so you'll only have to type in your password when you log in to the desktop. Generate a keypair on your desktop machine:

```
ssh-keygen -t dsa -C your.email@address
```

Enter a passphrase for your key. This puts the secret key in `~/.ssh/id_dsa` and the public key in `~/.ssh/id_dsa.pub`. Now see whether you have an ssh-agent running at present:

```
echo $SSH_AGENT_PID
```

Most window managers will run it automatically if it's installed. If not, start one up:

```
eval $(ssh-agent)
```

Now, tell the agent about your key:

```
ssh-add
```

and enter your passphrase. You'll need to do this each time you log in; if you're using X, try adding `SSH_ASKPASS=ssh-askpass ssh-add`

to your `.xsession` file. (You may need to install `ssh-askpass`.) Now for each server you log into, create the directory `~/.ssh` and copy the file `~/.ssh/id_dsa.pub` into it as `~/.ssh/authorized_keys`.

If you started the ssh-agent by hand, kill it with

```
ssh-agent -k
```

when you log out.

```
[edit]
```

Using rsync over ssh

* Difficulty: Intermediate

* Application: Shell

Keep large directory structures in sync quickly with rsync. While tar over SSH is ideal for making remote copies of parts of a filesystem, rsync is even better suited for keeping the filesystem in sync between two machines. To run an rsync over SSH, pass it the `-e` switch, like this:

```
rsync -ave ssh greendome:/home/ftp/pub/ /home/ftp/pub/
```

Note the trailing `/` on the file spec from the source side (on greendome.) On the source spec, a trailing `/` tells rsync to copy the contents of the directory, but not the directory itself. To include the directory as the top level of what's being copied, leave off the `/`:

```
rsync -ave ssh bcnu:/home/six .
```

This will keep a copy of the `~/six/` directory on village in sync with whatever is present on `bcnu:/home/six/`. By default, rsync will only copy files and directories, but not remove them from the destination copy when they are removed from the source. To keep the copies exact, include the `--delete` flag:

```
rsync -ave ssh --delete greendome:~one/reports .
```

Now when old reports are removed from `~one/reports/` on greendome, they're also removed from `~/six/public_html/reports/` on the synced version, every time this command is run. If you run a command like this in cron, leave off the `v` switch. This will keep the output quiet (unless rsync has a problem running, in which case you'll receive an email with the error output). Using SSH as your transport for rsync traffic has the advantage of encrypting the data over the network and also takes advantage of any trust relationships you already have established using SSH client keys.

```
[edit]
```

Asset scanning

* Difficulty: Intermediate

* Application: nmap

Normally, when people think of using nmap, they assume it's used to conduct some sort of nefarious network reconnaissance in preparation for an attack. But as with all powerful tools, nmap can be made to wear a white hat, as it's useful for far more than breaking into networks. For example, simple TCP connect scans can be conducted without needing root privileges:

```
nmap rigel
```

nmap can also scan ranges of IP addresses by specifying the range or using CIDR notation:

```
nmap 192.168.0.1-254
```

```
nmap 192.168.0/24
```

nmap can provide much more information if it is run as root. When run as root, it can use special packets to determine the operating system of the remote machine by using the `-O` flag. Additionally,

you can do half-open TCP scanning by using the `-sS` flag. When doing a half-open scan, nmap will send a SYN packet to the remote host and wait to receive the ACK from it; if it receives an ACK, it knows that the port is open. This is different from a normal three-way TCP handshake, where the client will send a SYN packet and then send an ACK back to the server once it has received the initial server ACK. Attackers typically use this option to avoid having their scans logged on the remote machine.

```
nmap -sS -O rigel
```

```
Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
```

```
Interesting ports on rigel.nnc (192.168.0.61):
```

```
(The 1578 ports scanned but not shown below are in state: filtered)
```

```
Port State Service
```

```
7/tcp open echo
```

```
9/tcp open discard
```

```
13/tcp open daytime
```

```
19/tcp open chargen
```

```
21/tcp open ftp
```

```
22/tcp open ssh
```

```
23/tcp open telnet
```

```
25/tcp open smtp
```

```
37/tcp open time
```

```
79/tcp open finger
```

```
111/tcp open sunrpc
```

```
512/tcp open exec
```

```
513/tcp open login
```

```
514/tcp open shell
```

```
587/tcp open submission
```

```
7100/tcp open font-service
```

```
32771/tcp open sometimes-rpc5
```

```
32772/tcp open sometimes-rpc7
```

```
32773/tcp open sometimes-rpc9
```

```
32774/tcp open sometimes-rpc11
```

```
32777/tcp open sometimes-rpc17
```

```
Remote operating system guess: Solaris 9 Beta through Release on SPARC
```

```
Uptime 44.051 days (since Sat Nov 1 16:41:50 2003)
```

```
Nmap run completed -- 1 IP address (1 host up) scanned in 166 seconds
```

With OS detection enabled, nmap has confirmed that the OS is Solaris, but now you also know that it's probably Version 9 running on a SPARC processor.

One powerful feature that can be used to help keep track of your network is nmap's XML output capabilities. This is activated by using the `-oX` command-line switch, like this:

```
nmap -sS -O -oX scandata.xml rigel
```

This is especially useful when scanning a range of IP addresses or your whole network, because you can put all the information gathered from the scan into a single XML file that can be parsed and inserted into a database. Here's what an XML entry for an open port looks like:

```
<port protocol="tcp" portid="22">
```

```
<state state="open" />
```

```
<service name="ssh" method="table" conf="3" />
```

```
</port>
```

nmap is a powerful tool. By using its XML output capabilities, a little bit of scripting, and a database, you can create an even more powerful tool that can monitor your network for unauthorized services and machines.

[edit]

[Backup your bootsector](#)

* Difficulty Expert

* Application Shell

Messing with bootloaders, dual-booting and various other scary processes can leave you with a messed up bootsector. Why not create a backup of it while you can:

```
dd if=/dev/hda of=bootsector.img bs=512 count=1
```

Obviously you should change the device to reflect your boot drive (it may be sda for SCSI). Also, be very careful not to get things the wrong way around - you can easily damage your drive! To restore use:

```
dd if=bootsector.img of=/dev/hda  
[edit]
```

Protect log files

* Difficulty: Expert

* Application: Various

During an intrusion, an attacker will more than likely leave telltale signs of his actions in various system logs: a valuable audit trail that should be protected. Without reliable logs, it can be very difficult to figure out how the attacker got in, or where the attack came from. This info is crucial in analysing the incident and then responding to it by contacting the appropriate parties involved. But, if the break-in is successful, what's to stop him from removing the traces of his misbehaviour?

This is where file attributes come in to save the day (or at least make it a little better). Both Linux and the BSDs have the ability to assign extra attributes to files and directories. This is different from the standard Unix permissions scheme in that the attributes set on a file apply universally to all users of the system, and they affect file accesses at a much deeper level than file permissions or ACLs. In Linux, you can see and modify the attributes that are set for a given file by using the lsattr and chattr commands, respectively. At the time of this writing, file attributes in Linux are available only when using the ext2 and ext3 filesystems. There are also kernel patches available for attribute support in XFS and ReiserFS. One useful attribute for protecting log files is append-only. When this attribute is set, the file cannot be deleted, and writes are only allowed to append to the end of the file.

To set the append-only flag under Linux, run this command:

```
chattr +a filename
```

See how the +a attribute works: create a file and set its append-only attribute:

```
touch /var/log/logfile
```

```
echo "append-only not set" > /var/log/logfile
```

```
chattr +a /var/log/logfile
```

```
echo "append-only set" > /var/log/logfile
```

```
bash: /var/log/logfile: Operation not permitted
```

The second write attempt failed, since it would overwrite the file. However, appending to the end of the file is still permitted:

```
echo "appending to file" >>> /var/log/logfile
```

```
cat /var/log/logfile
```

```
append-only not set
```

```
appending to file
```

Obviously, an intruder who has gained root privileges could realise that file attributes are being used and just remove the append-only flag from our logs by running chattr -a. To prevent this, we need to disable the ability to remove the append-only attribute. To accomplish this under Linux, use its capabilities mechanism.

The Linux capabilities model divides up the privileges given to the all-powerful root account and allows you to selectively disable them. In order to prevent a user from removing the append-only attribute from a file, we need to remove the CAP_LINUX_IMMUTABLE capability. When present in the running system, this capability allows the append-only attribute to be modified. To modify the set of capabilities available to the system, we will use a simple utility called lcap (<http://packetstormsecurity.org/linux/ad...3.tar.bz2>).

To unpack and compile the tool, run this command:

```
tar xvjf lcap-0.0.3.tar.bz2 && cd lcap-0.0.3 && make
```

Then, to disallow modification of the append-only flag, run:

```
./lcap CAP_LINUX_IMMUTABLE  
./lcap CAP_SYS_RAWIO
```

The first command removes the ability to change the append-only flag, and the second removes the ability to do raw I/O. This is needed so that the protected files cannot be modified by accessing the block device they reside on. It also prevents access to `/dev/mem` and `/dev/kmem`, which would provide a loophole for an intruder to reinstate the `CAP_LINUX_IMMUTABLE` capability. To remove these capabilities at boot, add the previous two commands to your system startup scripts (eg `/etc/rc.local`). You should ensure that capabilities are removed late in the boot order, to prevent problems with other startup scripts. Once `lcap` has removed kernel capabilities, they can be reinstated only by rebooting the system.

Before doing this, you should be aware that adding append-only flags to your log files will most likely cause log rotation scripts to fail. However, doing this will greatly enhance the security of your audit trail, which will prove invaluable in the event of an incident.

[edit]

Automatically encrypted connections

* Difficulty: Expert

* Application: FreeS/WAN

One particularly cool feature supported by FreeS/WAN is opportunistic encryption with other hosts running FreeS/WAN. This allows FreeS/WAN to transparently encrypt traffic between all hosts that also support opportunistic encryption. To do this, each host must have a public key generated to use with FreeS/WAN. This key can then be stored in a DNS TXT record for that host. When a host that is set up for opportunistic encryption wishes to initiate an encrypted connection with another host, it will look up the host's public key through DNS and use it to initiate the connection.

To begin, you'll need to generate a key for each host that you want to use this feature with. You can do that by running the following command:

```
ipsec newhostkey --output /tmp/'hostname'.key
```

Now you'll need to add the contents of the file that was created by that command to `/etc/ipsec.secrets`:

```
cat /tmp/'hostname'.key >> /etc/ipsec.secrets
```

Next, you'll need to generate a TXT record to put into your DNS zone. You can do this by running a command similar to this one:

```
ipsec showhostkey --txt @colossus.nnc
```

Now add this record to your zone and reload it. You can verify that DNS is working correctly by running this command:

```
ipsec verify
```

Checking your system to see if IPsec got installed and started correctly

Version check and ipsec on-path [OK]

Checking for KLIPS support in kernel [OK]

Checking for RSA private key (/etc/ipsec.secrets) [OK]

Checking that pluto is running [OK]

DNS checks.

Looking for TXT in forward map: colossus [OK]

Does the machine have at least one non-private address [OK]

Now just restart FreeS/WAN - you should now be able to connect to any other host that supports opportunistic encryption. But what if other hosts want to connect to you? To allow this, you'll need to create a TXT record for your machine in your reverse DNS zone.

You can generate the record by running a command similar to this:

```
ipsec showhostkey --txt 192.168.0.64
```

Add this record to the reverse zone for your subnet, and other machines will be able to initiate opportunistic encryption with your machine. With opportunistic encryption in use, all traffic between the hosts will be automatically encrypted, protecting all services simultaneously.

[edit]

Eliminate suid binaries

* Difficulty: Intermediate

* Application: find

If your server has more shell users than yourself, you should regularly audit the setuid and setgid binaries on your system. Chances are you'll be surprised at just how many you'll find. Here's one command for finding all of the files with a setuid or setgid bit set:

```
find / -perm +6000 -type f -exec ls -ld {} \; > setuid.txt &
```

This will create a file called setuid.txt that contains the details of all of the matching files present on your system. To remove the s bits of any tools that you don't use, type:

```
chmod a-s program
```

```
[edit]
```

Mac filtering Host AP

* Difficulty: Expert

* Application: iwpriv

While you can certainly perform MAC filtering at the link layer using iptables or ebtables, it is far safer to let Host AP do it for you. This not only blocks traffic that is destined for your network, but also prevents miscreants from even associating with your station. This helps to preclude the possibility that someone could still cause trouble for your other associated wireless clients, even if they don't have further network access.

When using MAC filtering, most people make a list of wireless devices that they wish to allow, and then deny all others. This is done using the iwpriv command.

```
iwpriv wlan0 addmac 00:30:65:23:17:05
```

```
iwpriv wlan0 addmac 00:40:96:aa:99:fd
```

```
...
```

```
iwpriv wlan0 maccmd 1
```

```
iwpriv wlan0 maccmd 4
```

The addmac directive adds a MAC address to the internal table. You can add as many MAC addresses as you like to the table by issuing more addmac commands. You then need to tell Host AP what to do with the table you've built. The maccmd 1 command tells Host AP to use the table as an "allowed" list, and to deny all other MAC addresses from associating. Finally, the maccmd 4 command boots off all associated clients, forcing them to reassociate. This happens automatically for clients listed in the table, but everyone else attempting to associate will be denied.

Sometimes, you only need to ban a troublemaker or two, rather than set an explicit policy of permitted devices. If you need to ban a couple of specific MAC address but allow all others, try this:

```
iwpriv wlan0 addmac 00:30:65:fa:ca:de
```

```
iwpriv wlan0 maccmd 2
```

```
iwpriv wlan0 kickmac 00:30:65:fa:ca:de
```

As before, you can use addmac as many times as you like. The maccmd 2 command sets the policy to "deny," and kickmac boots the specified MAC immediately, if it happens to be associated. This is probably nicer than booting everybody and making them reassociate just to ban one troublemaker.

Incidentally, if you'd like to remove MAC filtering altogether, try maccmd 0.

If you make a mistake typing in a MAC address, you can use the delmac command just as you would addmac, and it (predictably) deletes the given MAC address from the table. Should you ever need to flush the current MAC table entirely but keep the current policy, use this command:

```
iwpriv wlan0 maccmd 3
```

Finally, you can view the running MAC table by using /proc:

```
cat /proc/net/hostap/wlan0/ap_control
```

The iwpriv program manipulates the running Host AP driver, but doesn't preserve settings across reboots. Once you are happy with the contents of your MAC filtering table, be sure to put the relevant commands in an rc script to run at boot time.

Note that even unassociated clients can still listen to network traffic, so MAC filtering actually does

very little to prevent eavesdropping. To combat passive listening techniques, you will need to encrypt your data.
[edit]

Check processes not run by you

* Difficulty: Expert

* Application: bash

Imagine the scene - you read our list of Linux Format Awards nominees this month, saw Crack Attack! on there, and took it upon yourself to give it a (very) thorough testing before casting your vote in its favour. Sadly, to your shock, the game drags to a halt just as you're about to beat your uppity subordinate - what could be happening to make your machine so slow? It must be some of those other users, stealing your precious CPU time with their scientific experiments, webserver or other weird, geeky things!

OK, let's list all the processes on the box not being run by you!

```
ps aux | grep -v `whoami`
```

Or, to be a little more clever, why not just list the top ten time-wasters:

```
ps aux --sort=-%cpu | grep -m 11 -v `whoami`
```

It is probably best to run this as root, as this will filter out most of the vital background processes. Now that you have the information, you could just kill their processes, but much more dastardly is to run xeyes on their desktop. Repeatedly!

Retrieved from "http://www.linuxformat.co.uk/wiki/index.php/58_Cool_Hacks"

HOWTO: Flash BIOS, The Ultimate Edition Way

<http://forumubuntu.com/viewtopic.php?f=7&t=3080>

The Quick Disclaimer: You will find I reiterate myself a number of times below. Some may consider it beating the proverbial dead horse. But, this is for good reason. BIOS flashing is destructive. Again, BIOS flashing is destructive. If you don't understand what this means, then please stop reading here.

Many manufacturers now include native linux utilities to assist in a BIOS flash. If you fit in this category then it is recommended to use the method provided by them. If you are feeling daring, know exactly what you are doing, or just want to know a little more about this topic, then please continue reading.

Update, 2009-03-02: This post details a variety of methods. Choose what works best for you. For example, a generic utility called Flashrom also exists (<http://www.coreboot.org/Flashrom>) -- this utility may or may not work for you -- DO YOUR HOMEWORK BEFORE ATTEMPTING A FLASH AS I WILL NOT BE HELD RESPONSIBLE WHEN THINGS GO AWRY.

This post IS NOT a comprehensive tutorial that details every possible way to flash your BIOS -- it describes a few methods and links to others -- USE WHAT WORKS BEST FOR YOU (AND DO YOUR HOMEWORK!!)

Release Compatibility

Ultimate Edition 9.04 Jaunty Jackalope

Ultimate Edition 8.10 Intrepid Ibex

Ultimate Edition 8.04 Hardy Heron

Ultimate Edition 7.10 Gutsy Gibbon

Ultimate Edition 7.04 Feisty Fawn

Ultimate Edition 6.06.2 LTS Dapper Drake (Maintenance release)

Ultimate Edition 6.06.1 LTS Dapper Drake (Maintenance release)

End of Life (EOL) (HOWTO still applies)

Ultimate Edition 6.10 Edgy Eft

Ultimate Edition 5.10 Breezy Badger

Ultimate Edition 5.04 Hoary Hedgehog

Ultimate Edition 4.10 Warty Warthog

As always, do your research before attempting a BIOS upgrade.

The bulk of this guide is an attempt at providing an easy solution to create either a floppy or CD using an Ultimate Edition system which will contain the open source OS FreeDOS as well as the required/necessary files from the BIOS/Hardware manufacturer in order to flash (upgrade) your Motherboard's BIOS. (Sites currently suspended but should be up soon I hope)

Changes made to your BIOS are permanent in the sense that you are overwriting your previous BIOS with a new one. The procedure below will replace your current BIOS ROM (or Flash ROM) with a new version meant specifically for your BIOS by way of its make, model, and manufacturer -- you will need to grab the necessary files to complete the tutorial and you can find them respectively at your BIOS or Hardware manufacturers' web sites.

Flashing your BIOS is a potentially dangerous activity that can render your Motherboard (and computer for that matter) inoperable. Please proceed with caution and due diligence, and fully understand what you are doing before attempting the commands below! The only way to reverse the commands below, to revert back to your original setup, is to make a copy of your BIOS configuration and re-flash (if your BIOS supports that) or by reinstalling the original BIOS (or newer) to your Motherboard.

Again, proceed with CAUTION!!! If you don't know what you are doing or feel unsafe at this point, please ask someone with more experience to help you.

If you want to take a look at an alternative way of doing this, when an ISO for flashing has already been created for you, but for some reason you need to modify the ISO file by adding other files into it, for example, then please take a look here: [Read-only issues with ISO files](#)

WARNING: Some motherboards have what are called "CMOS Jumpers" which in effect may prevent you from properly flashing your BIOS. If you believe you fall into this category, it is then necessary to read your BIOS flashing instructions from either your BIOS or hardware manufacturer for proper instructions on how to physically remove this jumper to allow for BIOS flashing. These instructions will be unique to each BIOS/hardware manufacturer. To get you started in finding out this information, here is a link to Google. You should modify this search query with your specific BIOS/hardware manufacturer's name to get unique hits/results.

Setting up

In order to create the boot disk you will need a boot disk image. For this guide we will use a FreeDOS image. FreeDOS is open source, and a pre-made usable image can be found at <http://www.fdos.org>. There are several images to choose from, but for this guide we'll assume that you want to upgrade your BIOS, in which case you'll want nothing but the OS plus the new BIOS and flashing tool provided by the motherboard manufacturer on your floppy. In this case you should download the file named "FDOEM.144.gz". (Once downloaded you can keep it on your hard disk, and use it whenever you need a boot floppy.)

Here's a link to what you need in case you can't find it: [FDOEM.144](#)

Creating the disk (floppy-method)

To create the disk you must first unpack the image file by changing directories to where the image was downloaded and issuing the following:

Code:

```
gunzip FDOEM.144.gz
```

Next insert the floppy you want to write to into the floppy drive. (Note that any data on it will be irreversibly erased.) Then run the following command:

Code:

```
dd if=FDOEM.144 of=/dev/fd0
```

If your floppy (for some odd reason) isn't /dev/fd0 then you need to replace that part with the correct path. Once the command has finished executing, you may then mount the floppy and copy

the BIOS + flash-tool onto it. That's all there is to it. Reboot, flash, and enjoy your new BIOS. If you need more assistance in unzipping your .EXE file, then refer to the heading below titled Unzipping the .EXE located under the Creating the disk (CD-Method).

Creating the disk (CD-method)

WARNING: If you don't have a floppy-drive, and use this method to create a bootable CD for BIOS-flashing, please note that the BIOS-flash utility will not be able to save a backup of the old BIOS, since it won't have access to any writeable media.

To create the disk you must first unpack the image file by changing directories to where the image was downloaded and issuing the following:

Because the image will be modified when using this method, you may want to make a backup copy of it and store it in a safe place, ready to be reused when needed

Code:

gunzip FDOEM.144.gz

Next open up a root-console. I like to run a quick `sudo -i` command. Then, if you don't yet have a directory to mount items temporarily, create one:

Code:

mkdir /mnt/temp

If you compiled the loopback device support as a module you will then need to issue the following command (if it's compiled in-kernel you can skip it):

Code:

modprobe loop

Next, it's time to mount the freedos-image:

Code:

mount -o loop -t vfat FDOEM.144 /mnt/temp

After the image has been mounted, you need to then copy the flashing tool and the new BIOS file to /mnt/temp (or whichever working temp directory you choose to use). The code isn't listed, because the filenames vary depending on the motherboard manufacturer, model and BIOS in question. Just in case you can't figure out what to do, here's a quick primer on grabbing the necessary files from what is almost always in the form of an .EXE file

Unzipping the .EXE

Okay. So you've downloaded an .EXE that contains your BIOS manufacturers flash program as well as the new BIOS ROM file. In addition, you'll probably find a Readme.txt file of some sort that will tell you exactly the syntax they want you to use to flash your BIOS. You can then unzip the .EXE file -- you can grab unzip by going to <http://packages.Ultimate Edition.com>, selecting your release, and then choosing the unzip application.

If you want or need a much more robust set of archiving apps, then you can run the following in a terminal (universe and multiverse will need to be enabled. See Enabling Extra Repositories, written by aysiu, for information on how to enable extra repositories)

Code:

sudo aptitude update && sudo aptitude install cabextract rar unace unrar p7zip arj unzoo lha

libarchive1 libarchive-tar-perl libarchive-zip-perl dpkg-dev

The unzipping process would look something like this:

unzip ~/Desktop/hpbios.exe

Various files have now been extracted to your Desktop

View the Readme file with cat or less: `cat Readme.txt` or `less Readme.txt`

Copy the new flashing utility and the new BIOS ROM to /mnt/temp [1]

Move to the next step of unmounting the image

[1] which now should include your flash program and the new BIOS ROM file as well...in addition,

there may be other files that you wish to copy over, such as an Autoexec.bat file to automate the process. If you don't want to enter any commands manually upon reboot, then copy over all files that were extracted to your Desktop and make sure they are part of FDOEM.144 located under the /mnt/temp directory. You can check to make sure that all files were copied over successfully by changing directories into /mnt/temp and viewing the newly copied files from the "deflated" .EXE

Next, unmount the image:

Code:

```
umount /mnt/temp
```

We are now ready to proceed with creating an ISO-image from the floppy-image. First we make sure that mkisofs is currently installed on our system:

Code:

```
sudo aptitude update && sudo aptitude install mkisofs
```

Then we create an ISO-image:

Code:

```
mkisofs -o Bootable-CD-BIOS-Image.iso -b FDOEM.144 FDOEM.144
```

The image is now created and can be burnt to disc.

Option 1

You can burn your new ISO straight from the command line (remember to put a blank CD into your drive before you issue the command).

Code:

```
cdrecord -v Bootable-CD-BIOS-Image.iso
```

Option 2

Burn the image with your favorite CD-burning software, then reboot from the CD, and then flash your BIOS with the commands provided by the BIOS manufacturer!

Enjoy your new BIOS!

Send pictures and files by email in nautilus

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=3086>

<http://ubuntuforums.org/showthread.php?t=1131770>

Runs on x64 looks good, trying now <http://razerraz.free.fr/mailpictures-0.93.deb>

You must log out/log on your session to have the nautilus menu entry

BitDefender A new free antivirus for Unix/Linux platform

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=3059>

Today I'd like to introduce to you all a brand new antivirus for Unix/Linux platform from a famous company, BitDefender. A couple of days ago, BitDefender introduced a new antivirus for Unix/Linux OS called BitDefender Antivirus Scanner for Unices.

Just because you use Linux, it doesn't mean your computer doesn't have viruses or worms. They are just lying dormant, embedded in the EXE files on the NTFS partitions, or hiding beside those DLLs on the dual-boot computers, waiting for you to send them to your Windows-using friends.

Unless you sadistically enjoy seeing your non-Linux peers suffer, you should act responsibly and get yourself an anti-virus scanner that runs on Linux. One such is the latest BitDefender Antivirus Scanner For Unices. If you agree to use it on your home computers only, you can have it for free - that's free as in freeware, not Richard Stallman free.

Virus is a catch-all phrase, and BitDefender's designed to catch them all - from executable viruses, script viruses, macro viruses, to backdoors, trojans, spyware, adware, diallers, and more.

BitDefender looks inside files created by over 70 packers, compressors, and installers, from the most common ones such as Zip, 7-Zip, and .tar.gz to more exotic ones such as UPX, ASPack, PEXcrypt, etc. We found that it even picked up an infected file inside an archive split into multiple volumes. To avoid being caught out by Zip bombs, BitDefender has an adjustable compression depth.

Product Description

BitDefender Antivirus Scanner for Unices is a versatile on-demand scanner built for Linux and FreeBSD systems. It provides antivirus and antispyware scanning for both UNIX-based and Windows-based partitions.

BitDefender Antivirus Scanner for Unices is highly customizable and capable of script and extension-based integration with various applications such as file managers and mail clients.

Licensing

BitDefender Antivirus Scanner for Unices may be used free of charge at home or on your personal computer. In case you want to use BitDefender Antivirus for Unices for business purposes, a registration key must be purchased through the BitDefender Online Store or from BitDefender certified partners

Features and Benefits

- * On-demand antivirus and antispyware protection
- * Script and extension-based integration with various applications and services:
 - o Mail clients (e.g. Pine, Evolution) and Mail Server services
 - o Scheduling services (e.g. Cron) ensuring scan and update automation
- * Classic command line scanner complete with a graphical user interface for better integration with desktop environments
- * Automatic addition of the scanner's GUI to the system menu
- * Three popular file manager plugins (the GPL-ed sources) included in the GUI package: Konqueror (KDE), Nautilus (GNOME) and Thunar (Xfce)

- * Action setting based on scan result type

System Requirements

Operating System: Linux, FreeBSD

Linux Kernel: 2.4.x or 2.6.x (recommended)

FreeBSD: 5.4 (or newer with compat5x)

glibc: version 2.3.1 or newer, and libstdc++5 from gcc 3.2.2 or newer

Processor: x86 compatible 300 MHz; i686 500MHz; amd64(x86_64)

Minimum Memory: 64MB (128MB recommended)

Minimum Free Disk Space: 100MB

Supported Distributions:

- * RedHat Enterprise Linux 3 or newer
- * SuSE Linux Enterprise Server 9 or newer
- * Fedora Core 1 or newer
- * Debian GNU/Linux 3.1 or newer
- * Slackware 9.x or newer
- * Mandrake/Mandriva 9.1 or newer
 - FreeBSD 5.4 or newer

Terminal Command check / Update / Upgrade

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=3069>

What do we type into a terminal to check our repositories, make updates where needed and upgrade the packages

What is the difference between updating and upgrading?

sudo apt-get update && sudo apt-get upgrade

The difference, Update brings the list of packages up to date, Upgrade actually installs new versions

of the packages.

You need to update first before upgrading.

So should I type:

Code:

sudo apt-get upgrade && sudo apt-get update

No, it's the other way around:

Code: **sudo apt-get update && sudo apt-get upgrade**

Creating a Trusted Local Repository

<http://forumubuntu.org.uk/viewtopic.php?f=7&t=3079>

Creating a Trusted Local Repository from which Software Updates can be installed.

Abstract.

If you manage multiple PCs running Ultimate Edition, you will likely want to keep them all updated. Thus, you will want to install the Ultimate Edition updates to each of them as they become available, and you will have each PC individually download all of the updates from the Ultimate Edition repositories on the internet.

This may, however, be impractical to you—particularly if, e.g., you are on a rather slow internet connection, or if your monthly data transfer volume is severely limited, or if you simply prefer to save the bandwidth.

If you would like to download the updates just once, and further distribute the downloaded files locally, you can set up a Local Repository. Basically, a "repository" is a directory that contains a set of software packages (i.e., ".deb" files, in the case of Debian-based distributions—such as Ultimate Edition),

plus an index file that lists the packages that are available in the directory. Such a repository, however, is considered "untrusted": the software installer (i.e., the "APT" system—which includes the "apt-get" command, the "Synaptic Package Manager," the "Update Manager," etc.) will ignore any package that is present in the repository whenever another copy of the package is available in another, trusted, repository—i.e., most likely, on the internet. To create a trusted repository, you need to take a few extra steps, in addition to creating the index file:

- * Create a "release" file, which is a small text file that contains, among other details, a checksum for the index file;

- * Generate a digital signature for the release file, to prevent tampering by unauthorised individuals.

Whenever the software installer attempts to use the trusted repository, it will verify if the digital signature for the release file is correct—i.e., if it corresponds to the current contents of the file. If the digital signature is correct, then the installer will trust the contents of the release file—which includes the checksum for the index file. If, subsequently, that checksum corresponds to the current contents of the index file, then the contents of that file can be trusted as well. Finally, the index file includes a checksum for each of the packages made available by the repository, so the packages themselves can be verified too.

Thus, before you can set up a trusted local repository, you will have to make sure that you can generate a digital signature for its release file. To that end, you will have to create a "GPG Key Pair" for yourself, and register it (or, more precisely, its public key part) as a trusted key for APT to use—so that APT can verify any digital signatures that are generated with this key pair (or, more precisely, with its private key part).

Step 1: Generating your GPG Key Pair.

To generate a GPG key pair, start the following command:

Code:

```
gpg --gen-key
```

After some introductory text, the program will ask you which type of key you want to create:

Code:

Please select what kind of key you want:

(1) DSA and Elgamal (default)

(2) DSA (sign only)

(5) RSA (sign only)

Since you will be using your key only for generating digital signatures, you don't need option 1, "DSA and Elgamal," (although it will work fine), but you can get by with a "sign only" key—i.e., option 2, "DSA," or option 5, "RSA." Of these two, RSA is the more powerful, so I suggest you select that:

Code:

Your selection? 5

If you opt for an RSA key, the program will next ask you what size you want for the key; for maximum security, you will most likely want the longest possible key—i.e., 4096 bits (however, longer keys will require more computing resources, so you may prefer a shorter key if your computer is not all too powerful):

Code:

RSA keys may be between 1024 and 4096 bits long.

What keysize do you want? (2048) 4096

Next, the program will ask you if you want to attach an expiration date to the key; if you don't, then the key will remain valid indefinitely:

Code:

Please specify how long the key should be valid.

0 = key does not expire

<n> = key expires in n days

<n>w = key expires in n weeks

<n>m = key expires in n months

<n>y = key expires in n years

For a high-risk key (e.g., a key that you will use to digitally prove your identity), it may be critical to set an appropriate expiration date, but the type of key that you are currently generating doesn't really need an expiration date. The default option—i.e., a key that won't expire—is, therefore, perfectly acceptable. Alternatively, you may prefer the key to expire in, say, 5 years or so:

Code:

Key is valid for? (0) 5y

The program will show you the expiration date and time for the key, and ask you to confirm if that is what you want:

Code:

Key expires at Thu 06 Mar 2014 21:12:03 CET

Is this correct? (y/N) Y

Now it's time to specify the "user id" for your new key:

Code:

You need a user ID to identify your key; the software constructs the user ID from the Real Name, Comment and E-mail Address in this form:

"Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

The program will ask you for the following three items:

* Real name

For a key that you will use to prove your identity, you should specify your own real name here.

In this case, however, you should really avoid using your own name. Technically, you can give the key any name that you like—even something totally unrelated to the function of the key, like "Bart Bogus" or "Teri Hatcher is the Greatest!" or even "Who the Hell is Teri Hatcher, anyway?" will do. Since you will be using the key to sign your local Ultimate Edition repository, you should probably name it

"Local Ultimate Edition Repository" or (if you want to mention the specific Ultimate Edition release for which you will be using the key) "Local Ultimate Edition Intrepid Ibx Repository" or some such.

* E-mail address

You needn't specify an e-mail address for this key.

* Comment

You needn't specify a comment either.

So, for example:

Code:

Real name: Local Ultimate Edition Intrepid Ibx Repository

E-mail address:

Comment:

You selected this USER-ID:

"Local Ultimate Edition Intrepid Ibx Repository"

You will get a chance to modify the user id; if you're satisfied with what you entered, then select Okay to continue:

Code:

Change (N)ame, (C)omment, (E)-mail or (O)kay/(Q)uit? O

At this point, the program has all the data that it needs to generate your new key. The key will actually consist of two parts:

* A private (or secret) key—which (as the name suggests) you should keep secret, to prevent "digital identity theft."

In this case, you don't really reveal your identity, but anyone who gains access to your secret key will be able to create falsely "trusted" software repositories that could be installed on any computer that knows about your key.

* A public key—which you are supposed to distribute to anyone who should be able to verify your identity.

In this particular case, you will add this public key to the list of trusted keys on any computer on which you want to install the packages from your local repository.

It should be obvious that the secret key should really be kept secret. Therefore, the program will ask you for a "passphrase" (or "password"), which will protect your secret key:

Code:

You need a Passphrase to protect your secret key.

Enter passphrase:

Select a sufficiently complex passphrase, but one that you can remember—if you ever forget it, then you will lose access to your secret key, and you will no longer be able to generate any digital signatures with it.

You will have to repeat your passphrase:

Code:

Repeat passphrase:

Finally, the program will begin to compute your new key pair:

Code:

We need to generate a lot of random bytes. It is a good idea to perform some other action (type on the keyboard, move the mouse, use the disks) during the prime generation; this gives the random number generator a better chance to gain enough entropy.

You should really follow the advice that the program gives you, and keep your system busy. If you don't, then the program will complain that it cannot generate enough random bytes for your key—and it will make you wait until it can:

Code:

...+++++

Not enough random bytes available. Please do some other work to give

the OS a chance to collect more entropy! (Need 182 more bytes)

.....+++++

Once your new key pair is ready, the program will produce the following output:

Code:

```
gpg: /home/luvr/.gnupg/trustdb.gpg: trustdb created
gpg: key xxxxxxxx marked as ultimately trusted
public and secret key created and signed.
gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: next trustdb check due at 2014-03-06
pub 4096R/xxxxxxx 2009-03-07 [expires: 2014-03-06]
Key fingerprint = xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
uid Local Ultimate Edition Intrepid Ibex Repository
```

Note that this key cannot be used for encryption. You may want to use the command "--edit-key" to generate a subkey for this purpose.

The bold, underlined and coloured text, xxxxxxxx, represents the "key id"—which you will have to specify when you export the (public) key, next.

Note:

You can list your keys at any time, with the following command:

Code:

```
gpg --list-keys
```

The output will include the key id, as shown below:

Code:

```
/home/luvr/.gnupg/pubring.gpg
```

```
-----
pub 4096R/xxxxxxx 2009-03-07 [expires: 2014-03-06]
```

```
uid Local Ultimate Edition Intrepid Ibex Repository
```

Step 2: Exporting your newly generated Public Key.

Now that your GPG key pair is ready, you will have to export the public key to a text file, using the following command:

Code:

```
gpg --output pubkey-export-file --armor --export xxxxxxxx
```

You should, obviously, replace the pubkey-export-file parameter string with an appropriate name for the output file. Furthermore, the xxxxxxxx parameter represents the key id, as discussed above.

Step 3: Importing your Public Key into the APT List of Trusted Keys.

Note:

You will have to perform this step on every computer on which you want to install software packages from your (yet to be created) local software repository.

You can perform this step at any time, and on as many computers as you deem appropriate.

Make sure that you have the pubkey-export-file (which you created in Step 2, above) available, and execute the following command to import it into the list of keys that will be trusted by APT:

Code:

```
sudo apt-key add pubkey-export-file
```

Note:

To verify if the key was successfully imported into APT, you can run the following command:

Code:

```
sudo apt-key list
```

Alternatively, you could start the "Software Sources" utility (from the "System" → "Administration" menu) and display its "Authentication" tab page.

Moreover, you can also start the "Software Sources" utility from within the "Synaptic Package Manager," through its "Settings" → "Repositories" menu.

Step 4: Installing the Debian Package Development Tools.

Return to the computer on which you generated your key pair (i.e., where you ran steps 1 and 2 above).

Execute the following command to install the Debian Package Development Tools:

Code:

```
sudo apt-get install dpkg-dev
```

Once this package is installed, you will be able to create software package repositories on this system.

Step 5: Creating the Directory Structure for your Local Repository.

You will be setting up a simple repository—i.e., one that consists of just one directory. Complex repositories, consisting of a deeper directory hierarchy, are out of the scope of this document.

As an example, the following commands will create a "LocalRepository" subdirectory under your personal home directory,

Code:

```
cd  
mkdir LocalRepository  
cd LocalRepository
```

(The first "cd" command will set your current directory to your home location, after which the second command will create the "LocalRepository" subdirectory, and the third command will enter the newly created directory.)

To kickstart your new local repository, you can load it with the software packages that you have recently installed. Whenever you install new software or updates through APT, the required packages will be loaded into the APT cache—i.e., the "/var/cache/apt/archives" directory on your system. You can now copy these packages to your local repository:

Code:

```
cp /var/cache/apt/archives/*.deb .
```

(Notice the lone dot (".") at the end of this command—it is required, and is a placeholder for the "current directory." The command effectively copies all Debian packages from the APT cache into your current directory—i.e., into your local repository.)

Note:

If your APT cache does not contain any packages, then be sure to verify the "Temporary Files" setting in the "Synaptic Package Manager," as follows:

- * Open "Synaptic Package Manager" ("System" → "Administration" → "Synaptic Package Manager");
- * Display its "Preferences" dialogue ("Settings" → "Preferences");
- * Display the "Files" tab page;
- * Under the "Temporary Files" header, ensure that either "Leave all downloaded packages in the cache" or "Only delete packages which are no longer available" is selected.

Step 6: Creating the Index File for your Local Repository.

Once your local repository contains all the packages that you want to make available through it, you are ready to create the index file for the repository. To that end, run the following command:

Code:

```
dpkg-scanpackages ./dev/null > Packages
```

This will create a text file, "Packages," which is the plain-text, uncompressed version of the index file. You should also create a compressed version of the index file, using the following command:

Code:

```
gzip -9c Packages > Packages.gz
```

This will create the compressed version, "Packages.gz," of the index file, without deleting the original, uncompressed file.

Step 7: Creating the Release File for your Local Repository.

The release file is a small text file that should contain:

* A header.

A detailed description of the format of this section is out of the scope of this document, but the following is a simple example of a perfectly valid header:

Code:

```
Archive: intrepid
```

```
Origin: Ultimate Edition
```

```
Label: Local Ultimate Edition Intrepid Repository
```

```
Architecture: i386
```

```
MD5Sum:
```

* Two detail lines—one for the uncompressed "Packages" file, and one for the compressed "Packages.gz" file. The detail lines should be formatted as follows:

- o One blank space;

- o The MD5 checksum for the file, as calculated by the "md5sum" command;

- o One blank space;

- o The size, in bytes, of the file, right-aligned and padded to the left with blanks, in a 16-character field;

- o One blank space;

- o The name of the file.

The following example shows two valid detail lines:

Code:

```
d8cd948e0371a338025d3d99f5f9f304 454617 Packages
```

```
7e43f7e45a7f49dbb01e659c25997446 109619 Packages.gz
```

Obviously, it would be an awfully error-prone process if you were to try and manually type in these lines. Hence, an automated process is highly desirable—and, fortunately, not too hard to develop.

For example, you can construct the command to generate the detail line for the "Packages" file as follows:

* To calculate the MD5 checksum, run the "md5sum" command:

Code:

```
$ md5sum Packages
```

```
d8cd948e0371a338025d3d99f5f9f304 Packages
```

* You should keep only the checksum value, and drop the file name from the output of the "md5sum" command. To that end, pipe the output through the "cut" command, keeping only the first blank-delimited field:

Code:

```
$ md5sum Packages | cut --delimiter=' ' --fields=1
```

```
d8cd948e0371a338025d3d99f5f9f304
```

* To obtain the size of the file, you can use the "wc" command with the "--bytes" option:

Code:

```
$ wc --bytes Packages
```

```
454617 Packages
```

* Again, you need keep only the first blank-delimited field of the output:

Code:

```
$ wc --bytes Packages | cut --delimiter=' ' --fields=1
```

```
454617
```

* To generate the properly formatted output line, you can use the "printf" command.

The checksum can be inserted directly into the format string thanks to the "\$(...command...)" construct supported by the shell; for the file size, the format string can specify a 16-character decimal number field, and the value can be given as an argument to be substituted into the format string:

Code:

```
$ printf ' $(md5sum Packages | cut --delimiter=' ' --fields=1) ' %16d Packages\n \
```

```
> $(wc --bytes Packages | cut --delimiter=' ' --fields=1)
```

```
d8cd948e0371a338025d3d99f5f9f304 454617 Packages
```

Based on the above description, the following is a command sequence to fully automate the creation

of the release file. To generate the header, a so-called "here" document is used--i.e., the input text is copied directly into the command sequence, with a specified marker string (in this case, "EOF") used to mark the end of the document.

Code:

```
cat > Release <<EOF
Archive: intrepid
Origin: Ultimate Edition
Label: Local Ultimate Edition Intrepid Repository
Architecture: i386
MD5Sum:
EOF
printf '$(md5sum Packages | cut --delimiter=' ' --fields=1)' %16d Packages\n \
$(wc --bytes Packages | cut --delimiter=' ' --fields=1) >> Release
printf '$(md5sum Packages.gz | cut --delimiter=' ' --fields=1)' %16d Packages.gz' \
$(wc --bytes Packages.gz | cut --delimiter=' ' --fields=1) >> Release
```

Step 8: *Generating the Digital Signature for the Release File.*

The digital signature for the release file should be written to a file named "Release.gpg," using the following command:

Code:

```
gpg --armor --detach-sign --output Release.gpg Release
```

With this step, your local repository is ready for use.

Step 9: *Adding your Local Repository as an APT Source.*

Now that your local repository is fully set up, you can copy it to any computer on which you want to use it, and add it to its APT source configuration file, "/etc/apt/sources.list."

You will need root privileges to edit the "/etc/apt/sources.list." file—therefore, if you want to start up your text editor (e.g., "gedit") from a terminal, you should run the command as follows:

Code:

```
sudo gedit /etc/apt/sources.list
```

Alternatively, you can start the text editor from the GNOME desktop environment by pressing <Alt>-<F2> to display the "Run Application" dialog, and typing the following command:

Code:

```
gksudo gedit /etc/apt/sources.list
```

At the top of the file, add a definition for the local repository, as follows:

Code:

```
deb file:///home/luvr/LocalRepository /
```

(Note that you should, of course, specify the correct path to your repository on the "file:/" protocol.)

Finally, you should let APT update its internal package index:

Code:

```
sudo apt-get update
```

(Alternatively, you can execute the "Reload" function in "Synaptic Package Manager.")

Epilogue: Keeping your Local Repository Up-To-Date.

From now on, whenever you perform any further software installations through APT, you should perform the following steps to keep your local repository updated:

* Copy the downloaded packages from the APT cache to your local repository—e.g.:

Code:

```
cd ~/LocalRepository
cp /var/cache/apt/archives/*.deb .
```

How to Optimize your Internet Connection using MTU and RWIN

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=3078>

TCP Maximum Transmission Unit (MTU)

The TCP Maximum Transmission Unit (MTU) is the maximum size of a single TCP packet that can pass through a TCP/IP network.

An easy way to figure out what your MTU should be is to use ping where you specify the payload size:

Code:

```
ping -s 1464 -c1 google.com
```

Note though that the total IP packet size will be $1464+28=1492$ bytes since there is 28 bytes of header info. Thus if the packet gets fragmented for payload above 1464, then you should set your MTU=1492. Ping will let you know when it becomes fragmented with something like the following:

Code:

```
john@TECH5321:~$ ping -s 1464 -c1 google.com
```

```
PING google.com (72.14.207.99) 1464(1492) bytes of data.
```

```
64 bytes from eh-in-f99.google.com (72.14.207.99): icmp_seq=1 ttl=237 (truncated)
```

```
--- google.com ping statistics ---
```

```
1 packets transmitted, 1 received, 0% packet loss, time 0ms
```

```
rtt min/avg/max/mdev = 118.672/118.672/118.672/0.000 ms
```

```
john@TECH5321:~$ ping -s 1465 -c1 google.com
```

```
PING google.com (64.233.167.99) 1465(1493) bytes of data.
```

```
From adsl-75-18-118-221.dsl.sndg02.sbcglobal.net (75.18.118.221) icmp_seq=1 Frag needed and DF set (mtu = 1492)
```

```
--- google.com ping statistics ---
```

```
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms
```

In other words, to find your correct MTU, you would first start with a small packet size, and then gradually increase it until you see fragmentation; the cutoff point will be what to use for your MTU (using the formula $\text{payload} + 28 = \text{MTU}$). Note in the first case shown above where the payload size is 1464, the packet was transmitted fine, but in the second case where the payload size is 1465, ping complains "Frag needed"; to clarify, that means any packet with a payload of 1464 or less will be sent just fine, but a payload size of 1465 or above will end up being fragmented. Therefore, 1464 is the maximum payload, and that means the MTU is $1464+28=1492$.

To set the MTU temporarily (will be lost after a reboot), you can do:

Code:

```
sudo ifconfig <interface> mtu 1492
```

Note that unfortunately some NICs do not allow you to change their MTU. You can use "ifconfig" by itself to see what the MTU is for your NIC and whether the MTU changes when you use the above command.

Or to make the change permanent, you can add it to /etc/network/interfaces:

Code:

```
gksudo gedit /etc/network/interfaces
```

And then add "mtu <value>" in it for the particular interface. Here's an example of mine that uses my wireless interface wlan0:

Code:

```
iface wlan0 inet static
```

```
address 192.168.1.23
```

```
netmask 255.255.255.0
```

```
gateway 192.168.1.1
```

```
wireless-essid John's Home WLAN
```

```
mtu 1492
```

TCP Receive Window (RWIN)

In computer networking, RWIN (TCP Receive Window) is the maximum amount of data that a computer will accept before acknowledging the sender. In practical terms, that means when you download say a 20 MB file, the remote server does not just send you the 20 MB continuously after you request it. When your computer sends the request for the file, your computer tells the remote

server what your RWIN value is; the remote server then starts streaming data at you until it reaches your RWIN value, and then the server waits until your computer acknowledges that you received that data OK. Once your computer sends the acknowledgement, then the server continues to send more data in chunks of your RWIN value, each time waiting for your acknowledgment before proceeding to send more.

Now the crux of the problem here is with what is called latency, or the amount of time that it takes to send and receive packets from the remote server. Note that latency will depend not only on how fast the connection is between you and the remote server, but it also includes all additional delays, such as the time that it takes for the server to process your request and respond. You can easily find out the latency between you and the remote server with the ping command. When you use ping, the time that ping reports is the round-trip time (RTT), or latency, between you and the remote server. When I ping google.com, I typically get a latency of 100 msec. Now if there were no concept of RWIN, and thus my computer had to acknowledge every single packet sent between me and google, then transfer speed between me and them would be simply the (packet size)/RTT. Thus for a maximum sized packet (my MTU as we learned above), my transfer speed would be:

Code:

```
1492 bytes/.1 sec = 14,920 B/sec or 14.57 KiB/sec
```

That is pathetically slow considering that my connection is 3 Mb/sec, which is the same as 366 KiB/sec; so I would be using only about 4% of my available bandwidth. Therefore, we use the concept of RWIN so that a remote server can stream data to me without having to acknowledge every single packet and slow everything down to a crawl.

Note that the TCP receive window (RWIN) is independent of the MTU setting. RWIN is determined by the BDP (Bandwidth Delay Product) for your internet connection, and BDP can be calculated as:

Code:

```
BDP = max bandwidth of your internet connection (Bytes/second) * RTT (seconds)
```

Therefore RWIN does not depend on the TCP packet size, and TCP packet size is of course limited by the MTU (Maximum Transmission Unit).

Before we change RWIN, use the following command to get the kernel variables related to RWIN:

Code:

```
sysctl -a 2> /dev/null | grep -iE "_mem |_rmem|_wmem"
```

Note the space after the `_mem` is deliberate, don't remove it or add other spaces elsewhere between the quotes.

You should get the following three variables:

Code:

```
net.ipv4.tcp_rmem = 4096 87380 2584576
```

```
net.ipv4.tcp_wmem = 4096 16384 2584576
```

```
net.ipv4.tcp_mem = 258576 258576 258576
```

The variable numbers are in bytes, and they represent the minimum, default, and maximum values for each of those variables.

Code:

```
net.ipv4.tcp_rmem = Receive window memory vector
```

```
net.ipv4.tcp_wmem = Send window memory vector
```

```
net.ipv4.tcp_mem = TCP stack memory vector
```

Note that there is no exact equivalent variable in Linux that corresponds to RWIN, the closest is the `net.ipv4.tcp_rmem` variable. The variables above control the actual memory usage (not just the TCP window size) and include memory used by the socket data structures as well as memory wasted by short packets in large buffers. The maximum values have to be larger than the BDP (Bandwidth Delay Product) of the path by some suitable overhead.

To try and optimize RWIN, first use ping to send the maximum size packet your connection allows (MTU) to some distant server. Since my MTU is 1492, the ping command payload would be $1492-28=1464$. Thus:

Code:

john@TECH5321:~\$ ping -s 1464 -c5 google.com

PING google.com (64.233.167.99) 1464(1492) bytes of data.

64 bytes from py-in-f99.google.com (64.233.167.99): icmp_seq=1 ttl=237 (truncated)

64 bytes from py-in-f99.google.com (64.233.167.99): icmp_seq=2 ttl=237 (truncated)

64 bytes from py-in-f99.google.com (64.233.167.99): icmp_seq=3 ttl=237 (truncated)

64 bytes from py-in-f99.google.com (64.233.167.99): icmp_seq=4 ttl=237 (truncated)

64 bytes from py-in-f99.google.com (64.233.167.99): icmp_seq=5 ttl=237 (truncated)

--- google.com ping statistics ---

5 packets transmitted, 5 received, 0% packet loss, time 3999ms

rtt min/avg/max/mdev = 101.411/102.699/105.723/1.637 ms

Note though that you should run the above test several times at different times during the day, and also try pinging other destinations. You'll see RTT might vary quite a bit.

But for the above example, the RTT average is about 103 msec. Now since the maximum speed of my internet connection is 3 Mbits/sec, then the BDP is:

Code:

$(3,000,000 \text{ bits/sec}) * (.103 \text{ sec}) * (1 \text{ byte}/8 \text{ bits}) = 38,625 \text{ bytes}$

Thus I should set the default value in net.ipv4.tcp_rmem to about 39,000. For my internet connection, I've seen RTT as bad as 500 msec, which would lead to a BDP of 187,000 bytes.

Therefore, I could set the max value in net.ipv4.tcp_rmem to about 187,000. The values in net.ipv4.tcp_wmem should be the same as net.ipv4.tcp_rmem since both sending and receiving use the same internet connection. And since net.ipv4.tcp_mem is the maximum total memory buffer for TCP transactions, it is usually set to the the max value used in net.ipv4.tcp_rmem and net.ipv4.tcp_wmem.

And lastly, there are two more kernel TCP variables related to RWIN that you should set:

Code:

sysctl -a 2> /dev/null | grep -iE "rcvbuf|save"

which returns:

Code:

net.ipv4.tcp_no_metrics_save = 1

net.ipv4.tcp_moderate_rcvbuf = 1

Note enabling net.ipv4.tcp_no_metrics_save (setting it to 1) means have Linux optimize the TCP receive window dynamically between the values in net.ipv4.tcp_rmem and net.ipv4.tcp_wmem.

And enabling net.ipv4.tcp_moderate_rcvbuf removes an odd behavior in the 2.6 kernels, whereby the kernel stores the slow start threshold for a client between TCP sessions. This can cause undesired results, as a single period of congestion can affect many subsequent connections.

Before you change any of the above variables, try going to <http://www.speedtest.net> or a similar website and check the speed of your connection. Then temporarily change the variables by using the following command with your own computed values:

Code:

sudo sysctl -w net.ipv4.tcp_rmem="4096 39000 187000" net.ipv4.tcp_wmem="4096 39000

187000" net.ipv4.tcp_mem="187000 187000 187000" net.ipv4.tcp_no_metrics_save=1

net.ipv4.tcp_moderate_rcvbuf=1

Then retest your connection and see if your speed improved at all.

Once you tweak the values to your liking, you can make them permanent by adding them to /etc/sysctl.conf as follows:

Code:

net.ipv4.tcp_rmem=4096 39000 187000

net.ipv4.tcp_wmem=4096 39000 187000

net.ipv4.tcp_mem=187000 187000 187000

net.ipv4.tcp_no_metrics_save=1

net.ipv4.tcp_moderate_rcvbuf=1

And then do the following command to make the changes permanent:

Code:

sudo sysctl -p

If everything went well, now you can enjoy a faster internet connection!

Simple guide to Sound Solutions for Hardy,Intrepid and Jaunty

<http://forumubuntu.org.uk/viewtopic.php?f=7&t=3068>

If you have just upgraded to or installed Hardy or Intrepid or Jaunty and you have some sound somewhere, but not everywhere for everything, this is a fast way to get all the missing stuff you need and give you some tools to figure out what is going on. If you hear the startup sound but nothing else, or if some applications work and others don't, this is for you. If you have no sound at all, there is a link at the end of this post for more extensive troubleshooting help but you should try this first anyway as it may solve your problem and will not make matters worse. Make sure that your system is fully updated.

Credit goes to markbuntu forum article <http://ubuntuforums.org/showthread.php?t=997506>

Required Packages

First you need to get some missing packages with Synaptic. These packages were not installed by default but are important for getting your sound working properly:

asoundconf-gtk

This is a little Default Sound Card application for choosing the default sound card for alsa.

gnome-alsamixer

This is a gui mixer, far easier to use than the command line alsamixer.

alsa-oss

This is the wrapper for oss applications so they will use alsa instead of grabbing the sound card all for themselves

libasound2

libasound2-plugins

These are the plugins for alsa

pavucontrol

This is the Pulse Audio device chooser and will pull in the pavucontrol which is the Pulse Audio Volume Control and papref which is the Pulse Audio Preferences along with the Pulse Audio

Volume Meters

gststreamer0.10-pulseaudio

This is the gststreamer plugin for pulseaudio

Ultimate Edition-restricted-extras

This is the package with all the restricted codecs and java and flashplayer so you can watch youtube and play your mp3s,etc..

If you have other applications like mplayer, vlc, amarok, or audacious be sure to get any extra packages available for them also.

Single command to install

**sudo apt-get install asoundconf-gtk alsa-oss libasound2 libasound2-plugins pavucontrol
gststreamer0.10-pulseaudio Ultimate Edition-restricted-extras**

Setting things up

Once you have all these packages installed, close any application that may be trying to use sound and go to System/Preferences/Sound and set all the preferences from automatic to PulseAudio except Default Mixer Tracks which you should set to your sound card. Go to

System/Preferences/Default Sound Card and choose pulseaudio.

Next, right click on the little speaker on the top panel, that is the Panel Volume Control. Click Open Volume Control and make sure it is set to the same thing as the Default Mixer Tracks. Click on Preferences and make sure that Master and PCM and whatever else you want to control are selected. Make sure that any boxes labeled SPDIF or IEC958 are not checked. Close the Preferences box. Push up the sliders in the volume control and make sure the little speakers do not have little red mute marks on them. Go to Applications/Sound and Video/GNOME ALSA Mixer and see if there is anything you missed because sometimes, for some cards, not all the options are in the Panel Volume Control.

Go to Applications/Sound and Video and select Pulse Audio Device Chooser. This will put a little icon on the panel near the Panel Volume Control. Click on the new icon and choose Volume Control. This will open the Pulse Audio Volume Control. Go to Output Devices and see if your sound card is there, it will be listed as ALSA PCM on front:...(ALC8 via DMA or whatever your sound card is. If you have a usb device it will be listed as ALSA PCM on front:...(USB Audio) via DMA or something like that. Make sure the sliders are up and the device is not muted. If any of the above is giving you problems, try rebooting.

Now, open Rythmbox and play something. If you have nothing handy just play one of the radio stations, you should hear something. In the Pulse Audio Volume Control/Playback you should see something like this Rythmbox: Playback Stream and some Volume sliders that you can adjust more than one device.

If you have more than one device listed in Output Devices, Rythmbox may be playing in the wrong one if you do not hear anything so right click on the stream and choose move stream and move it to another device.

If you have more than one device and you want to use them all, like a usb headset and your speakers, go back to the Pulse Audio Device chooser on the panel and select Configure Local Sound Server/Simultaneous Output and click the box:

“Add virtual output device for simultaneous output on all local sound cards”

Now you can right click on the stream and move it to your new device. You should have sound from all your sound devices now or at least a clue about how it is supposed to work.

Other stuff

Another thing you may need to do, Check in System/Administration/Users and Groups that your users and root are enabled as members of the following groups:

pulse

pulse-access

pulse-rt

This seems to be a particular problem for some people after getting recent updates.

If you still have problems look in to the following references (They are very good)

<http://ubuntuforums.org/showthread.php?t=843012>

<http://ubuntuforums.org/showthread.php?p=5931543>

50 amazing Ultimate Edition time-saving tips

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2692>

A Small post

System performance

1. Shorten the boot menu timeout

If you're fed up of waiting for the boot menu to timeout before your favourite operating system launches, open '/boot/grub/menu.lst' with a text editor and look for the line starting with 'timeout'. Just lower the number to its the right. This is the number of seconds the menu system will wait before booting the default operating system (0 or 1 is not recommended).

2. Monitor boot performance

One of the best utilities you can install for checking your system's performance is called 'bootchart'. After installation and a reboot, 'bootchart' will create a complex graph of everything that's running and taking up resources as your system boots, and place an image of the graph in the /var/log/bootgraph folder.

3. Improve boot speed

When the boot menu appears (you might have to press escape) select the default Ultimate Edition boot option and press 'e'. Cursor down to the line starting with 'kernel' and press 'e' again. You're now editing the boot parameters, and you need to press space and add the word 'profile'. Press return followed by 'b' to boot. Disk access during your boot sequence will now be profiled, which means that subsequent booting should be faster.

4. Trim unwanted services

The default Ultimate Edition installation takes an over cautious approach to background services. Bluetooth tools may be running, for example, even if you don't have the hardware. Disable the services you don't need by opening the Services window from the System>Administration menu. Be careful not to disable services you rely on.

5. Monitor CPU usage

You might think that CPU monitors are purely for geeks trying to steal a few extra cycles from their overclocked processors. But this isn't true. A discreet CPU monitor is the best way detecting a wayward process that's slowing down the rest of the system. Right click on the desktop panel, and select 'System Monitor' for our favourite. There's a similar applet for KDE.

6. Manage your processes

If you do detect a process on your system that's stealing more CPU cycles than it really should, then you need to end that process to get those cycles back. Save all your work, and use the Ultimate Edition process manager. This is part of the System Monitor tool, and this can be opened from the System>Administration menu.

7. Be nice to one another

If you use the System Monitor to manage your running tasks, you might have noticed the 'nice' column. 'nice' is basically a task's priority, and ranges between -20 to 19. If you have a CPU heavy task running, such as a 3D calculation for example, increasing the nice will lower its priority, and make your system feel more responsive.

The default Gnome desktop

8. Enable Gnome Auto login

A lot of us are the sole users of our computers, and it makes little sense navigating through a login screen before getting to our desktops. You can enable auto-login for a default account on your Ultimate Edition machine by selecting 'Login Window' from the System> Administration window. Switch to the 'Security' page, enable 'Automatic Login' and select the user.

10. Remove the menu popup delay

HCI gurus insist that there should be a delay between when you click on a menu and when it appears, but if it's speed you're after, you can remove the delay. Open a terminal, and type 'nano ~/.gtkrc-2.0', then add a single line 'gtk-menu-popup-delay = 0'. Save this by pressing escape and typing 'Y', and after a restart you should find your menus are ultra quick.

11. Add More Workspaces

Workspaces are one of the best things about Linux. They're a great way of organising your applications onto different virtual screens. By default, Ultimate Edition sets up only two, but you can adjust

this number by right clicking on the workspace switcher in the bottom right corner of the display and opening the Preferences window.

12. Use Workspaces more effectively

Use 'Ctrl alt' and either cursor left or right to switch between adjacent workspaces, and if you hold down the shift key, the active window will move to the new desktop too. For better control, right click on any windows top border to open a context menu, and from here you can choose to move the window to another workspace.

13. Don't start everything

As with system services, the average Ultimate Edition installation runs lots of different programs at startup.

You can remove those you don't need by launching the Sessions window from the Preferences menu. If you don't use the desktop search, for instance, disable 'Tracker'. Other likely candidates for removal include Bluetooth, the Evolution Alarm Notifier and the Print Queue Applet.

14. Remember the running session

Another neat feature of the setting manager is that you can configure your desktop to remember the applications that were running when you shutdown your machine. This is a great way of quickly launching into your working environment. Just switch to the Session Options page and enable the 'Automatically Remember' option.

15. Fine tune the Gnome desktop

Application shortcuts are hidden behind the Gnome equivalent of the Windows registry editor. This can be launched from the command-line by typing 'gconf-editor'. But be careful, settings changed here could mess up your desktop. If you do, then the desktop can be restored to its default state by deleting the '.gconf' and '.gconfd' folders from your home directory.

16. Launching applications with a key combination

One of the settings hidden in Gconf is the ability to launch applications with a key combination. Navigate to 'apps>metacity>key_binding_commands', double click on one of the 'command_' entries and enter the launch command for the application you want to run. To set the key, double click on the same entry in 'apps>metacity> global_keybindings' and press a key. Holding 'Ctrl Shift alt' and that key will now launch the application.

17. Use pervasive searches

Ultimate Edition comes with an excellent utilities for searching through the contents of files and emails, but

it's not enabled by default. Open the Search and Indexing window from the Preferences menu, and enable both indexing and watching. After the index has been created, you can search through your files using the 'Tracker Search Tool' in the 'Applications>Accessories' menu.

18. Switch To A Faster Desktop

Ultimate Edition uses the Gnome desktop by default. It's a good choice because Gnome is powerful, capable

and popular. But it's not streamlined or particularly efficient. A faster alternative is XFCE, the source of Xubuntu, and this can be installed through the Synaptic package manager by searching for the 'xubuntu-desktop' meta-package.

20. KDE auto-login

If you prefer Kubuntu or the KDE desktop, then you'll need to use a different configuration panel to enable auto-login. Open KDE's System Settings application, and switch to the Advanced page and open the Login Manager. Enter your root password and switch to the convenience page. From here you can choose to enable a user for auto login.

21. Pre-load Konqueror

If you're always launching KDE's file and web browser, you can pre-load several instances of it to speed up launch time. Open the Settings>Configure Konqueror window, and switch to the Performance page. Increase the number of instances from 1 to something like 4 or 5. Each instance takes extra system memory, but each session of Konqueror will now load almost instantly.

22. Use Konqueror shortcuts

Konqueror is a great file and web browser with plenty of shortcuts for the power user. Our favourite is the ability to use shortcuts in the location field to perform online searches. Typing 'wp:linux' will search Wikipedia for Linux, while 'gg:linux' and 'ggi:linux' will search Google and Google Images. System-wide

23. Launch OpenOffice.org faster

The default OpenOffice.org configuration errs on the side of caution. There are 100 levels of undo, for example, and reducing this number will reduce the amount of memory it uses. This setting can be found from the Options window by switching to the Memory page. Try reducing the undo steps to 30.

24. Use the quick launch toolbar

In both Gnome and KDE, you can drag applications from the launch menu onto the desktop and onto the toolbar. Clicking on these icons is the quickest way of launching your most used applications, short of holding down a certain key combination.

25. Replace slow applications

One of the best things about open source is that there's always an alternative, and switching to one can vastly improve your system's performance. Try Abiword instead of OpenOffice.org's Writer, Thunar instead of Nautilus and Opera instead of Firefox. All are broadly compatible with their alternatives, and perform faster.

26. Rapid application launch

If you know the name of the application or tool you want to launch, you can quickly start it by pressing Alt and F2. This displays a single command-line prompt in a window, and into this you type your application name. Type 'firefox' and its icon will appear. Pressing enter will launch it.

27. Take a screenshot

Pressing the Print key will take a screenshot and bring up the save file window. Being able to take a screenshot at any moment is incredibly useful, and is great for saving online order details, for example, or just your high score in Crack Attack. Pressing Alt and Print will take a screenshot of the currently active window.

28. Quickly restart the desktop

Occasionally, you may find that your desktop hangs and you can no longer use the keyboard or mouse. Fortunately, the desktop process is entirely independent of the rest of the operating system, and you can reset the desktop by holding down the Ctrl Alt and backspace keys. But you will still lose any unsaved data, so be careful.

29. Jump to a console

Another option if your desktop has crashed is jumping to a command-line console. Pressing Ctrl

and alt, followed by F1-F6 will switch the display to one of six different consoles. From here, you can login and try to kill the process causing trouble, before switching back to your desktop by pressing Ctrl Alt and F7.

31. *Tweak your Nvidia settings*

After installing the proprietary driver, Nvidia graphics hardware provides exceptional 3D and 2D acceleration for the Linux desktop. You can fine-tune your Nvidia hardware by installing an application called 'nvidia-settings', from which you can edit your monitor settings, enable twin displays and add a drop shadow to the cursor.

32. *Track down large unused files*

Large and scattered files can start to slow your desktop down, as well as any applications that rely on reading the contents of a directory. The best tool we've found for consolidating and deleting unused files is called Filelight. It uses a pie chart to show where the largest files are located, and you can easily delete directories of junk from the right click menu.

33. *Enable vertical sync in Compiz*

Compiz, the 3D whizzy desktop effects application, can be either a resource hog or even an acceleration tool. It depends on the power of your graphics hardware. But we've nearly always had better more responsive results on the desktop by enabling the vertical sync option in the general option page of the Compiz settings manager.

34. *Don't Compiz*

On the other hand, the wonderful effects that Compiz produces can't really be described as functional, although they do provide some improved usability for some. You can free up plenty of resources by disabling the desktop effects from the Visual Effects page of the Preferences>Appearance window.

35. *Get packages off a CD or DVD*

Even in these times of pervasive internet, you sometimes need to be able to install a package without having an internet connection. Fortunately, the Synaptic package manager can read the contents of an Ultimate Edition installation CD, and add those packages to the database for installation from the drive. Open the Software Sources window from the Administration menu, switch to the 'Third Party' page and click on the 'Add CD-ROM' button.

36. *Boost load speed with Preload*

Preload is a tool you can install through the Synaptic package manager. It will run silently in the background, from where it will try to guess which libraries you're likely to use before you use them. It will then load these into memory so that your applications load quicker. The effects seem to be minimal with recent releases of Ultimate Edition, but it's worth a try.

37. *Use a virtual desktop*

If you enjoy trying different distributions, but have always been put off by the installation, try Virtual Box from the official Ultimate Edition repositories. It's easy to use and lets you install a virtual version of almost any Linux installation (and even Windows) right on your desktop, and running at close to native speeds.

38. *Boot into text mode*

Sometimes, a graphical environment is unnecessary, especially if you use your machine as a server. Which is exactly why there's a version of Ultimate Edition called the Server edition. By default, Server has no graphical desktop. But in all other ways, it's the same Ultimate Edition. This makes it perfect as

a web or media streaming server.

39. Suspend your system

Why wait for your system to boot when you can resume your session from hibernation. This is quicker than booting, and you can continue where you left off. But it's also dependent on your hardware behaving itself. Just give it a go to see if your hardware supports the feature. Click on the logout button, and if hibernate appears as an option, it should work.

41. Try it, it's really not that bad

The command-line really is your friend. After opening Terminal from the Applications>Accessories menu (or Konsole in KDE), you can accomplish many common tasks much more effectively than from any desktop GUI. To copy folder, for example, type 'cp -rf source destination', rename a file with 'mv' and edit a text file using a command called 'nano'.

42. Easy command shortcuts

You can press the tab key while using the command-line to automatically complete command names as well as system paths. You can also cursor up through your command history, and use 'Ctrl

- r' to search for a command starting with the characters you begin to type.

43. Replace heavy GUI applications with command-line equivalents

There are command-line versions of most desktop applications. You could install and use 'pine' for your email and news, for instance. Or try 'lynx' for web browsing and 'wget' or 'ncftp' for downloading files. 'mc', short for Midnight Commander, is a feature-full file manager, and all of these tools will run on hardly any memory with hardly any CPU requirements.

44. Create an ISO image from a CD or a DVD

You can create an ISO image from optical media, and most attached devices, by using a single command on the terminal. Type 'dd if=/dev/cdrom of=disk.iso bs=1024' to make a raw copy of the data and drop it into the disk.iso file. You may need to unmount the drive first, by typing 'sudo umount /dev/cdrom'.

45. Read an ISO disc image without burning it

If you've downloaded an ISO disc image, and you want to access the files on it without wasting an optical disk, you can create a virtual drive from the image with a single command. Open the terminal from the Accessories menu. Type 'sudo mkdir /mnt/image', followed by 'sudo mount -o loop disk.iso /mnt/image'. You can now browse the disc by pointing a file browser at the '/mnt/image' folder.

46. Use the 'screen' command

After you've got used to the command-line, one of the best commands to learn is called 'screen'. It's the command equivalent to virtual desktops, and it lets you run several sessions at once, as well as suspend and resume a session. Type 'screen' to start, then press 'Ctrl a' followed by 'c' to create a new session. 'Ctrl a' and 'n' or 'p' will switch through the active sessions. 'Ctrl a' and 'd' will detach from the session, while typing 'screen -r' will resume one.

47. Access your Ultimate Edition machine from anywhere

The best thing about the command-line is that you can use it to access your machine securely from anywhere on the internet. The key to this is something called 'SSH' - the secure shell. Install 'openssh-server' through Synaptic and use a tool called 'putty' on a Windows machine, or 'ssh' on Linux, to access the command-line through your user accounts on your Ultimate Edition box.

48. Transfer files between computers files quickly and securely

With the open SSH server installed and running, you can quickly and securely transfer files to and

from the remote computer using the 'sftp' command. It works just like FTP, and accepts both 'put' and 'get' for file transfers. If you prefer a GUI, we recommend using Filezilla on Windows, or 'sftp://' as a protocol in KDE.

49. Avoid typing 'sudo'

You might have noticed that for almost every important configuration command you type, you need to precede it with 'sudo' and your password. This can be a real pain if you're typing one sudo command after the other. Avoid this hassle by typing 'sudo bash', this transparently replaces the current shell with a new one, complete with administrator privileges.

50. Create a root account

If you find yourself spending more and more time requiring system administration privileges, you may as well enable the root account. Just type 'sudo passwd root', and enter your password followed by a new one for the root account. You can now type 'su root' to login as root, but you should only use this mode for essential system maintenance.

Speed up web surfing in Ultimate Edition 8.10 (Intrepid) Should work for all though
<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2394>

Everyone wants to have a faster web browsing experience! In this short How-To I'm going to cover an easy way of doing this using the recently released Ultimate Edition 8.10 Intrepid distribution. My recommendation in this article is to install a proxy server on your local computer. This stores local copies (caching) of web sites on your computers hard drive.

When you surf to a site it checks the cache first and if it finds the page or image there, it loads directly from the local hard drive copy. This is much faster than downloading again from the Internet, especially if you don't have a fast connection speed and it has the added benefit of reducing downloads from the Internet.

To install and use this in Ultimate Edition 8.10 is incredibly simple. Go to the System menu, select Administration and then Synaptic Package Manager. Click on search and type in squid.

In the main part of the window, you'll need to go about two-thirds of the way down the page until you find squid. Right-click on it and select Mark for Installation. Click Apply. This will then install squid and any dependencies it has (squid-common from memory). As this is a relatively tiny application this really only takes a few seconds.

When this is done, go to Firefox and click on Edit, Preferences, Advanced, Network, Settings. Click on Manual proxy configuration, and type into the box marked "Http proxy" the word localhost. In the port number, type in 3128. You can also tick the "Use this proxy for all protocols box".

If web pages don't load after following the instructions, try the following trouble-shooting tips and tricks.

Firstly, as squid installs itself as a service that loads on start-up, we are going to check if the service is actually running. The easiest way to do this is...(queue the dramatic music!)...using the Command Line Interface. Go to Applications, Accessories, Terminal.

Type in the following: `sudo service squid status`, and hit Enter and then your password. If you get a message that the service is running, then it should be working through Firefox. Reboot your computer and try Firefox again.

If however, it says the service is stopped, then type in the same command (or hit the up arrow on your keyboard) and replace "status" with "start" or "restart" and then Enter. It should only take a few seconds to get it going again. If you try the "status" command again, it should tell you that the service is running.

The second trouble-shooting tip is to disable IPv6 on your computer. This is also incredibly easy. In

the Command Line Interface terminal, type in the following: `sudo gedit /etc/modprobe.d/aliases`, followed by your password. This opens up a simple text editor, which is very similar to Windows Notepad.

Change the line: "alias net-pf-10 ipv6" to "alias net-pf-10 off" (so, simply delete the "ipv6" and replace with "off"). This disables IPv6. Save the file, close the editor window and the terminal, reboot and try again.

If you still can't get it going (and I'd be really surprised), then let me know here and I'll do what I can to help you!

Note that these instructions only work for 8.10. If you are using a previous version then they are a bit different. You have to edit the `/etc/squid/squid.conf` file to uncomment and create the `visible_hostname` field and then run `sudo squid -z` to create the directories.

So, there you have it. A simple way to help speed up your web browsing and lower your downloads.

KDE tips and tricks

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2571>

The K Desktop Environment (KDE) is incredibly popular in the world of GNU/Linux. Distributions such as SUSE and Mandrakelinux use it by default. KDE has some useful features that, while easily accessible, are less prominent. Just as a camera inexplicably makes a cell phone more fun to use, KDE's cool but unnoticed details may make it more attractive to prospective users. Read on to learn about a few such features may help you every day.

The K Desktop Environment (KDE) is incredibly popular in the world of GNU/Linux. Distributions such as SUSE and Mandrakelinux use it by default. KDE has some useful features that, while easily accessible, are less prominent. Just as a camera inexplicably makes a cell phone more fun to use, KDE's cool but unnoticed details may make it more attractive to prospective users. Read on to learn about a few such features may help you every day.

KWin is KDE's window manager, which means it organizes the placement of applications into windows so that you may easily use many of them simultaneously.

To shade a window, right-click on its border and click Shade in the resulting menu. After you do, the window will roll up whenever you aren't using it. When you want to see it again, moving your cursor over its border will restore it.

To keep a window on top, right-click on its border, select the Advanced sub-menu, and click Keep Above Others. This will keep the window visible as you work in the windows below it.

To resize a window to the full-screen, right-click on its border, select the Advanced sub-menu, and click Fullscreen. To return the window to its previous size, press Alt-Space and disable the Fullscreen option.

To hide a window's border, right-click on its border, select the Advanced sub-menu, and click No Border. To restore the window's border, press Alt-Space and disable the option. Removing the border will save you a few pixels of valuable screen space.

Creating and using hot keys

Hot keys are keyboard shortcuts to commands. They allow you easy access to any action without fumbling through menus, by simply pressing a short sequence of keys. The KDE Control Center

(KControl) allows you to create hot keys you can use with any application.

Let's assign one of the actions in the previous section to a hot key. Open KControl (use the console command `kcontrol` or find it in the K Menu), expand the Regional and Accessibility branch, and select Keyboard Shortcuts. You will be presented with dozens of actions to assign hot keys to. Scroll down to (or search for) the Shade action and select it. In the Shortcut for Selected Action frame, select the Custom button and enter a key sequence -- for instance, `Ctrl-Alt-S` -- then click the Apply button. Press the sequence to try shading KControl.

Next, let's assign a keyboard shortcut to a command using the Desktop Communications Protocol (DCOP). Still in KControl under the Regional and Accessibility branch, select the KHotKeys module. From here you can assign keyboard shortcuts and mouse gestures to commands that the Keyboard Shortcuts module does not provide.

For the purposes of this example we will set a key to pause the Kaffeine media player. Press the New Action button to begin, and set the Action type to Keyboard Shortcut -> DCOP Call (simple). Select the Keyboard Shortcut tab, then click the lone button to follow the familiar steps of assigning a key sequence. Now select the DCOP Call Settings tab and type `kaffeine` for Remote Application. Set the Remote object to `KaffeineIface` and the Called function to `pause`, then click Apply. Your keyboard shortcut will now toggle a running Kaffeine player between pause and play. You can do something similar for other applications; you can generally find instructions in each application's documentation.

Most applications designed for integration with KDE have a Configure Shortcuts dialog accessible through their Settings menu that will let you create hot keys that work specifically for that application, and only when its window is selected.

Using Konqueror's advanced features

Konqueror is KDE's file manager and Web browser. You can turn on its sidebar by going to the 'Window' menu and selecting Show Navigation Panel. The Navigation Panel will appear as a sidebar to Konqueror, with tabs to switch between tools. You can use it to Quick Browse your filesystem (Folder utility), mount CDs (Services utility), or even play music (Sidebar Media Player utility). To have more than one panel open at once, right-click on one of the tabs, select the Configure Navigation Panel sub-menu, and then the Multiple Views option. You can also add new tabs from this menu. Also, clicking Show Terminal Emulator in the Window menu will open an integrated Konsole.

Konqueror supplies many different ways to view your files. Along with the standard Icon, Tree, and Detailed List views is the excellent File Size view, which displays files and folders as graphical sections of your Konqueror page. Larger files are represented by larger blocks, allowing you to visualize your file system. You can configure how this information is presented to you in the bottom section of the View menu.

In Konqueror's Window menu are options to Split View horizontally and vertically. Splitting a window's view lets you see two pages at once, which can be useful for filling in forms at one Web page with information from another, or easily dragging and dropping files from one folder to another.

If you've created a combination of Konqueror options that suits you, you can save it for future use. Go to the Settings menu and click Save View Profile, enter a name for it, and click Save. To load it again later, go back to the Settings menu, the Load View Profile submenu, and click on the name you saved.

KIO slaves are a much-touted, and oft under-appreciated, aspect of KDE. KIO slaves allow you to access a number of APIs as if they were Internet protocols in Konqueror. In the same way that you

would attach `http://` to a Web site's URL, you can access your configured devices by going to `devices://` in Konqueror. You can find a list of the KIO slaves on your system in the Protocols section of KInfoCenter. A few you might appreciate are `locate://`, `man://`, `settings://`, and `print://`. The `audiocd:/` KIO slave is especially well implemented. Visiting it with an audio CD in a drive presents you with a list of the files on the CD, any of which can be double-clicked to play. There will also be MP3 and Ogg Vorbis folders that you can simply copy to your hard drive to rip the CD to either format.

GTK

One of the most unfortunate side effects of choice is incompatibility. You may have noticed that applications meant for KDE (which are written using Trolltech's Qt toolset) look different from those meant for GNOME (which uses the GIMP Toolkit (GTK+)). If you want a more consistent look, you can turn to the GTK-Qt project, which creates a KControl module for simple configuration. Installing and enabling (through KControl) GTK-Qt forces GTK+ applications to call Qt to render their widgets, effectively making GNOME software look like KDE software. I hope these tips may help those of you who are partial to KDE squeeze more use out of your desktop experience.

GUI CPU Speed Selector

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2639>

So, recently I saw an article about using the "cpufreq-selector" utility to set the CPU scaling speed of my Core 2 Duo. As a person who likes simplicity, I decided it might be useful to have a graphical interface for Ultimate Edition users who prefer to avoid the command prompt. Here's what I came up with.

First, Ultimate Edition installs Zenity by default. Zenity is a utility which allows you to integrated graphical

elements into shell scripts. Using Zenity, one could create a selection dialogue which allows the user to choose the SpeedStep frequency for their CPU. Here's how:

1. Open a new instance of gedit (Applications->Accessories->Text Editor).
2. Enter the following code:

Code:

```
#!/bin/bash
```

```
CURRENT=`cat /proc/cpuinfo | grep "^cpu MHz.*" | awk -F": " '{print $2}' | sed 's@\.@@g' | uniq`  
if [ "${CURRENT}" == "1000000" ]; then  
  SETA='TRUE';  
  SETB='FALSE';  
  SETC='FALSE';  
fi  
if [ "${CURRENT}" == "1333000" ]; then  
  SETA='FALSE';  
  SETB='TRUE';  
  SETC='FALSE';  
fi  
if [ "${CURRENT}" == "1667000" ]; then  
  SETA='FALSE';  
  SETB='FALSE';  
  SETC='TRUE';  
fi  
ans=$(zenity --list --text "Select CPU Speed" --radiolist --column "" --column "Speed" ${SETA}  
"1.0 GHz" ${SETB} "1.33 GHz" ${SETC} "1.67 GHz");  
VALUE=1000000  
if [ "${ans}" == "1.0 GHz" ]; then  
  VALUE=100000;  
fi  
if [ "${ans}" == "1.33 GHz" ]; then
```

```
VALUE=1333000;  
fi  
if [ "${ans}" == "1.67 GHz" ]; then  
VALUE=1667000 ;  
fi  
gksu "cpufreq-selector -f ${VALUE}"
```

3. Save the new file as "cpuScaler" and give the file execute permissions.
- 4.

4. Review your CPU seppings for you particular computer. My computer has 3 settings of 1, 1.3, and 1.66 GHz. Yours may differ, but you should easily be able to figure it our using cpufreq-selector and looking at the contents of /proc/cpuinfo.

5. Modify the seppings to suit your computer.

BleachBit.....

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2654>

BleachBit deletes unnecessary files to free valuable disk space, maintain privacy, and remove junk. Rid your system of old clutter including cache, temporary files, cookies, and broken shortcuts. Designed for Linux systems, it wipes clean Bash, Beagle, Epiphany, Firefox, Flash, Java, KDE, OpenOffice.org, Opera, RealPlayer, rpmbuild, VIM, XChat, and more.

Install BleachBit in Ultimate Edition

use the following command

```
wget http://downloads.sourceforge.net/bleach ... ntu810.deb
```

Now you have bleachbit_0.2.0-1_all_ubuntu810.deb package.

Install this .deb package using the following command

```
sudo dpkg -i bleachbit_0.2.0-1_all_ubuntu810.deb
```

You must check for updates once installed to make sure you have the latest version

Use with caution

Get Internet via your Bluetooth-connected mobile phone

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2234>

I've knocked up a guide to using your mobile phone as a modem device connected to your PC via Bluetooth. Once again, this is a menagerie of various guides already out there, but merged into one coherent mass with modifications to suit newbies.

Purpose: To access the Internet on your Ultimate Edition Linux laptop via your mobile phone's data service,

and do it via Bluetooth.

Scenario: You're on the move, you need access to the Internet. You can't make use of an unsecured access point because you're travelling on a train/bus/car/segway/pony. What to do?

Solution: You reach into your pocket. Your mobile phone has data services on it. Your phone and laptop also have Bluetooth. Your mind slowly does the math: Mobile + Bluetooth + Laptop = Mobile Internet Access. Why not use that?

NOTE: FOR ALL Ultimate Edition INTREPID USERS - There is an even simpler solution for you using

Intrepid's new Mobile Broadband feature. The current solution utilises the cabled USB connection to your phone rather than Bluetooth, but it's quick, simple and it works like a charm.

Prerequisites:

* A mobile phone with Bluetooth enabled. In this tutorial, I'm using my Nokia N95.

* Data enabled on your plan. In my case I'm using the Vodafone \$79 plan that includes data at a rate of \$0.90 per 5 minutes regardless of what is downloaded. This comes off the included monthly call allowance, ie: you are not paying extra on top of your call allowance unless you've used it all up

already.

* A laptop. In my case, a legtop - my EeePC 701.

* Bluetooth interface to your PC. This can be inbuilt, such as on a modern notebook PC, or a USB dongle such as the one I use.

* Ultimate Edition Hardy 8.04 (I'm using 8.04.1 32-bit in this example).

Pros:

* Internet access anywhere, anytime, as long as you have mobile coverage.

* Convenient - you don't need to do anything on your mobile to connect (other than enable Bluetooth) - it's all controlled by your lappy, and your phone can stay in your pocket/bag.

* Reasonably fast - my tests at the time of writing saw me pull between 0.82mbps and 1.04mbps via Oz Speed Test (Internode mirror) on three bars reception. More than adequate for general surfing, and as it turns out, more than enough to watch YouTube videos too.

* Much easier to surf the 'net on a lappy than it is on your mobile's diminutive screen.

Cons:

* Battery draining - Your lappy and mobile phone will sap more energy to generate that Bluetooth connection, but that said, it should be less draining than WiFi.

* Speed - I wouldn't bother trying to download a distro ISO. The speed is great for general surfing, but nothing bigger unless you are prepared to wait.

* Potentially expensive - On Vodafone, \$0.90 cents is for a five minute block of time, whether I am idle or not. There is no data limit. That makes it reasonably pricey, but it is a small price to pay for drop-in access. Keep an eye on your usage otherwise you may run out of credit. Based on my speed tests, I can download about 37MB of data, assuming constant transfer rate, in five minutes.

These instructions are fairly generic and should be adaptable to other distros, mobile phones and mobile networks.

1. Fire up/install Ultimate Edition as normal.

2. Plug in or enable your Bluetooth adapter. Your Bluetooth adapter will be automatically detected and drivers loaded - there is nothing for you to do manually here.

3. If you have NEVER used Bluetooth on your Ultimate Edition setup before, then go to the next step,

otherwise skip to Step 11 because you're probably already setup properly.

4. Get into a terminal.

5. Verify that your Bluetooth adapter is running with:

Code:

```
$ hciconfig -a
```

If you get details about hci0 listed including manufacturer's name, then your adapter is working.

.

6. Type in the following to edit your Bluetooth configuration file:

Code:

```
$ sudo gedit /etc/bluetooth/hcid.conf
```

This will bring up the Bluetooth configuration into the GEdit text editor.

.

7. Near the top of the file you will see the following:

Code:

```
# HCId options
```

```
options {
```

```
# Automatically initialize new devices
```

```
autoinit yes;
```

```
# Security Manager mode
```

```
# none - Security manager disabled
```

```
# auto - Use local PIN for incoming connections
```

```
# user - Always ask user for a PIN
```

```
#
```

```
security user;
```

Change the security user line to security auto

8. A few lines beneath this is a section that reads as follows:

Code:

```
# Default PIN code for incoming connections  
passkey "1234";
```

Change the 1234 to something else, eg: 4493. This is the pin number required for other Bluetooth devices to connect to you and it would be insecure to leave it at the default.

9. Save and exit.

10. Now restart Bluetooth by typing in:

Code:

```
$ sudo /etc/init.d/bluetooth restart
```

When you do this, an informational bubble will appear in your task bar saying <hostname>-0 Device has been made connectable, eg: if your PC's name is "gordon", the message would say "gordon-0 Device has been made connectable".

11. If it isn't enabled already, turn on Bluetooth on your mobile phone. On a Nokia N95, you bring up the main menu, then go to Tools->Bluetooth->Change OFF to ON.

12. In your terminal, type in the following:

Code:

```
$ hcitool scan
```

Your PC will now scan for local Bluetooth devices and your mobile phone should appear in the resulting list after a few seconds (along with anyone else's Bluetooth-enabled mobile phones that are in range). The output will look something like:

Code:

```
$ hcitool scan
```

```
Scanning ...
```

```
00:11:22:AA:BB:CC Nokia N95
```

```
$
```

In this example, my PC has found my Nokia mobile phone and shown me the MAC address for it.

13. Highlight and copy the MAC address of the phone to the clipboard using your mouse and CTRL + SHIFT + C.

14. Now we need to find the channel that your phone uses for Dial-Up Networking. Type in:

Code:

```
$ sdptool browse 00:11:22:AA:BB:CC
```

(Remember to paste, using CTRL + SHIFT + V, the MAC address of YOUR phone in place of the example above that you copied earlier).

15. You will see a whole lot of output as the services on your phone are queried and displayed.

Scroll back a bit and look for an entry that looks similar to the following:

Code:

```
Service Name: Dial-Up Networking
```

```
Service RecHandle: 0x10022
```

```
Service Class ID List:
```

```
"Dialup Networking" (0x1103)
```

```
Protocol Descriptor List:
```

```
"L2CAP" (0x0100)
```

```
"RFCOMM" (0x0003)
```

```
Channel: 2
```

```
Language Base Attr List:
```

```
code_ISO639: 0x454e
```

```
encoding: 0x6a
```

```
base_offset: 0x100
```

Profile Descriptor List:

"Dialup Networking" (0x1103)

Version: 0x0100

...and note down the Channel number. In the case of my Nokia N95, the channel used is 2.

16. Now type in:

Code:

```
sudo gedit /etc/bluetooth/rfcomm.conf
```

Add the following to the end of the file (NOTE: if you already have an rfcomm0 defined for another device, then simply increment the number, eg: rfcomm1, for your mobile phone entry):

Code:

```
rfcomm0 {  
bind no;  
device 00:11:22:AA:BB:CC;  
channel 2;  
comment "Nokia N95 via Bluetooth";  
}
```

(Remember to paste down YOUR MAC address and set YOUR channel number as derived from your phone).

17. Save and exit.

18. Now test the connection to your phone using the following commands:

Code:

```
$ rfcomm release 0
```

```
$ rfcomm connect 0
```

At this point, if your phone wasn't paired with Ultimate Edition already, your phone will prompt for the PIN

number to access Ultimate Edition and once accepted, Ultimate Edition will then prompt you for the mobile phone's

PIN. Enter them accordingly.

NOTE: If you setup anything other than rfcomm0 earlier, change the number 0 in the connect command to the number of the device you created, eg: rfcomm1 would mean typing in rfcomm connect 1 instead.

After a brief delay, you will see the following:

Code:

```
Connected /dev/rfcomm0 to 00:11:22:AA:BB:CC on channel 2
```

```
Press CTRL-C for hangup
```

Hooray! You have a data connection to your phone, but not to the Internet yet. Press CTRL + C to break the connection to your mobile.

19. Now we need to setup PPP (Point to Point Protocol). This is exactly what your home ADSL/Cable modem uses to connect to your Internet Provider, though in our case, the "provider" is Vodafone. Type in:

Code:

```
$ sudo gedit /etc/ppp/peers/vodafone
```

...and a blank file called "vodafone" will be created. You can name it anything you want really, but once created, populate the file with the following:

Code:

```
# Vodafone PPP initialisation/termination script  
noauth  
connect "/usr/sbin/chat -v -f /etc/chatscripts/vodafone-connect"  
disconnect "/usr/sbin/chat -v -f /etc/chatscripts/vodafone-disconnect"  
silent  
debug
```

```
/dev/rfcomm0
115200
defaultroute
usepeerdns
```

(Remember to change the rfcomm device number if you're not using zero).

To reduce the number of messages going to /var/log/messages, you can remove the "debug" line too.

.
20. Save and exit.

21. Now type in:

Code:

```
$ sudo gedit /etc/chatscripts/vodafone-connect
```

...and you are presented with another blank file. Populate it with:

Code:

```
# Vodafone PPP CONNECT script
TIMEOUT 5
ECHO ON
ABORT '\nBUSY\r'
ABORT '\nERROR\r'
ABORT '\nNO ANSWER\r'
ABORT '\nNO CARRIER\r'
ABORT '\nNO DIALTONE\r'
ABORT '\nRINGING\r\n\r\nRINGING\r'
" \rAT
TIMEOUT 12
OK ATE1
OK 'AT+cgdcont=1,"IP","vfinetnet.au"'
OK ATD*99***1#
```

If you are not in Australia (or indeed not with Vodafone), you will need to change the "vfinetnet.au" bit on the second last line above to your local country's/provider's Access Point Name. Refer to your provider for assistance, or you can try and see if your country and provider is listed on this nifty site I came across, and get the Access Point name from that.

.
22. Save and close.

23. Now type in:

Code:

```
$ sudo gedit /etc/chatscripts/vodafone-disconnect
```

...and populate the empty file with:

Code:

```
# Vodafone PPP DISCONNECT script
ABORT "BUSY"
ABORT "ERROR"
ABORT "NO DIALTONE"
SAY "\nSending break to the modem\n"
"" "K"
"" "K"
"" "K"
"" "+++ATH"
"" "+++ATH"
"" "+++ATH"
SAY "\nPDP context detached\n"
```

24. Save and exit.

25. Now we are ready to rock and/or roll. Let's try and connect to the Internet! First, let's reconnect to the phone via Bluetooth:

Code:

```
$ rfcomm connect 0
```

26. Once the connection is established, open up a second terminal window and type in:

Code:

```
$ pon vodafone
```

27. Now let's see if we picked up the network settings. Type in:

Code:

```
$ ifconfig
```

...and you should see an entry similar to the following:

Code:

```
ppp0 Link encap:Point-to-Point Protocol
inet addr:10.120.11.197 P-t-P:10.6.6.6 Mask:255.255.255.255
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:4 errors:1 dropped:0 overruns:0 frame:0
TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:3
RX bytes:64 (64.0 B) TX bytes:97 (97.0 B)
```

Yes, we have an IP address!

NOTE: If you cannot see the entry for "ppp0" in your ifconfig output, your phone may be taking a little longer to negotiate the connection. Wait a couple of seconds and try issuing the ifconfig command again. Failing that, re-issue the pon vodafone command and check ifconfig again. If you still cannot get ppp0 up, it is possible you might not have data enabled on your phone/plan. Refer to your provider for assistance, but essentially if you can surf the 'net from your mobile phone itself, then ppp0 should appear if the rfcomm connection was successful.

Anyway, if you can see ppp0, you can now try and hit the Internet:

Code:

```
$ ping -c4 www.overclockers.com.au
```

...and you should get output similar to the following:

Code:

```
PING overclockers.com.au (203.220.0.228) 56(84) bytes of data.
64 bytes from chips.ocau.ausgamers.com (203.220.0.228): icmp_seq=1 ttl=49 time=159 ms
64 bytes from chips.ocau.ausgamers.com (203.220.0.228): icmp_seq=2 ttl=49 time=134 ms
64 bytes from chips.ocau.ausgamers.com (203.220.0.228): icmp_seq=3 ttl=49 time=127 ms
64 bytes from chips.ocau.ausgamers.com (203.220.0.228): icmp_seq=4 ttl=49 time=136 ms
```

```
--- overclockers.com.au ping statistics ---
```

```
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
```

```
rtt min/avg/max/mdev = 25.838/25.885/25.957/0.045 ms
```

```
$
```

Congratulations - you've connected to the Internet via your mobile!

28. Now bring up Firefox (or other browser) and try and hit a website. If you get an "offline" error, make sure your browser is not operating in offline mode. In Firefox, uncheck the "Work Offline" box in the File menu.

29. Once you've gotten over how exciting this all is, you need to disconnect from Vodafone to ensure you're not going to rack up a huge phone bill, so go back into the second terminal window and enter:

Code:

```
$ poff vodafone
```

...and this will automatically break the rfcomm0 connection without needing to press CTRL + C in the first terminal window.

30. If you'd like to automate the commands entered to initiate PPP with your phone, create a new text file on your desktop (or wherever) as follows:

Code:

```
$ gedit ~/Desktop/Start_Vodafone_Internet
```

...and then populate the file with the following:

Code:

```
#!/bin/bash
rfcomm release 0
rfcomm connect 0 &
sleep 5
pon vodafone
read -p "Press Enter to terminate your session."
```

This will simply ensure the rfcomm device is released, then will connect to the phone, wait for 5 seconds to ensure the connection is up, then start the PPP to Vodafone. It will then sit there and wait for you to hit Enter at which point it will terminate the PPP.

31. Save and exit.

32. Now at the terminal, make the file executable with:

Code:

```
$ chmod a+x ~/Desktop/Start_Vodafone_Internet
```

...and now you can just double-click on the icon on your desktop, then Run In Terminal, to connect to your mobile phone in one hit.

Once finished, issue the poff vodafone command again, or hit Enter in the terminal window running the script above.

Fix Frozen Desktop or Application In Ultimate Edition Linux

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2529>

Linux is Very Stable operating system unlike windows. It has very less crashes, stucks and freezes which is very common with windows. But at rare cases this may occur in Linux.

We have 3 Quick ways to fix this problem without switching off your computer or rebooting it.

1) If only some applications are not responding not the whole desktop is frozen

Take system -> Administration -> System Monitor.

Then select the frozen application. Now click Kill process.

2) Try the key combination of Ctrl + Alt + Backspace to logout. Then login.

3) Use Virtual console

This way is little complicated than the first one but better method. Press

Ctrl + Alt + F2 to take the virtual console. Then log in and type:

```
ps -ax | grep startx
```

This command will return the PID of the Xserver.

Kill Xserver

```
kill -9 PID_Number
```

Return to your first console.

Press Alt + F1.

How to use Magic System Request Keys in Ultimate Edition Linux

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=2395>

The magic SysRq keys are key combinations within the Linux kernel that allows the user to perform various low level commands regardless of the system's state, except during kernel panics or freezes. It is often used to recover from X-Server freezes, or to reboot a computer without corrupting the file system.

Restarting Ultimate Edition safely when it is frozen

If anyone faces a freeze with Ultimate Edition where you cannot do anything, then this will certainly be

helpful if you want to reboot the OS as cleanly as possible without damaging their HDD's or losing their data.

In case of a freeze where you cannot do anything, simply press Alt+PrintScreen+R+E+I+S+U+B, keep in mind that the underlined keys must be kept pressed through the rest of the sequence AND that you will need to keep holding the sequence keys for a small period of time before going to the next one so that their actions can be carried out properly (For example, hold the R key for about 1-2 seconds before moving on to S). If the sequence does not work at first, then increase the time period between each sequence key press and try again.

If anyone requires a good way of remembering the sequence R+E+I+S+U+B, just remember “Reboot Even If System Utterly Broken”.

Raw (take control of keyboard back from X), tErminate (kill -15 programs, allowing them to terminate gracefully), kIll (kill -9 untermiated programs), Sync (flush data to disk), Unmount (remount everything read-only), reBoot.

NOTE:- This keystroke does not work in the event of a kernel freeze as the keystroke sequence depends on the kernel in order to unmount and make the required steps before the restart.

Safely shutting down Ultimate Edition when it is frozen

The key sequence to achieve this does not differ from the one used to restart Ultimate Edition safely except for the last key. So here it is as follows:-

Alt+PrintScreen+R+S+E+I+U+O, keep in mind that as in the previous sequence, the underlined keys must be kept pressed through the rest of the sequence AND that you will need to keep holding the sequence keys for a small period of time before going to the next one so that their actions can be carried out properly.

If someone requires a good description on what each keystroke here does, there is not much of a difference from the last one, except(Once again),the final key:-

Raw (take control of keyboard back from X), tErminate (kill -15 programs, allowing them to terminate gracefully), kIll (kill -9 untermiated programs), Sync (flush data to disk), Unmount (remount everything read-only), shutdown.

NOTE:- This keystroke does not work in the event of a kernel freeze as the keystroke sequence depends on the kernel in order to unmount and make the required steps before the shutdown.

Brief descriptions about the keys you can use in magic SysRq sequences

0 - 9 - sets the console log level, controlling which kernel messages will be printed to your console so that you don't get flooded.

B - restarts the system without making steps to ensure that the conditions are good for a safe reboot, using this key alone is like doing a cold reboot.

E - sends SIGTERM to all processes except init. This means that an attempt is done to end the current processes except init, safely, e.g. saving a document.

F - call oom_kill(Out Of Memory Killer), which will kill a process that is consuming all available memory.

H - displays help about the SysRq keys on a terminal though in actuality you can use any key except for the ones specified, to display help.

I - sends SIGKILL to all processes except init. This means that all the processes except for init are killed, any data in processes that are killed will be lost.

K - kills all processes on the current terminal. It is a bad idea to do this on a console where X is running as the GUI will stop and you can't see what you type, so you will need to switch to a tty after doing the magic SysRq.

L - sends SIGKILL to all processes, including init. This means that every process including init will be killed, using this key will render your system non-functional and no further magicSysRq keys can be used. So in this case you will have to cold reboot it.

M - dumps memory info to your console.

O - shuts down the system via ACPI or in older systems, APM. As in key “B”, using this key alone is like a cold reboot(Or in this case, a cold shut down).

P - dumps the current registers and flags to your console.

Q - dumps all timers info to your console.

R - takes keyboard and mouse control from the X server. This can be useful if the X-Server crashed, you can change to a console and kill the X-Server or check the error log.

NOTE:- The documentation refers to this key’s task as “Turns off keyboard raw mode and sets it to XLATE”, but we suppose it’s safe enough to assume that it takes back control from X.

S - writes all data from the disc cache to the hard-discs, it is a sync and is necessary to reduce the chances of data corruption.

T - dumps a list of current tasks and info to your console.

U - remounts all mounted filesystems read-only. After using this key, you can reboot the system with Alt+SysRq+B without harming the system.

W - dumps uninterruptable (blocked) state tasks.

Controlling the use of SysRq keys

There are some ways of controlling the use of SysRq keys(i.e. what can be used, enabling or disabling them completely), two ways of doing this are:-

1) Configuring the SysRq keys during kernel compilation itself.

There isn’t much here since you can only disable SysRq keys and not actually control or define what you can and can’t use. The option you are looking for is:-

MAGIC_SYSRQ

2) Using proc sysrq trigger calls.

This is much more flexible than changing the configuration of the kernel but this has one downside with security which is explained after(since it is very minor). You use the echo command to achieve this for ease but you could also use any normal text editor to achieve this. Now the command is(you will need root permissions):-

echo * > /proc/sys/kernel/sysrqwhere “*” is a number, which can be any one of these:-

0 - disable sysrq keys completely

1 - enable all functions of sysrq

2 - enable control of console logging level

4 - enable control of keyboard (SAK, unraw)

8 - enable debugging dumps of processes etc.

16 - enable sync command

32 - enable remount read-only

64 - enable signalling of processes (term, kill, oom-kill)

128 - allow reboot/poweroff

256 - allow nicing of all RT tasks(control the nice level(priority) of Real Time tasks)

So you can define what SysRq keys can be used, and also define whether they are all on or off.

Now for the “downside”. For example you disable SysRq keys when you want to stop people(local) from doing key presses and then shutting down or messing up the PC during an important task.

Now with configuring the kernel, you can stop SysRq keys from being used at all from the beginning of the boot process right upto the end, with calling the proc sysrq triggers however, your option only takes place when it is executed(i.e. after the system has booted up) so there is a certain area of vulnerability with calling the triggers whereas there is no such thing in configuring the kernel, some people are that desperate to secure their systems to care about a few seconds, however do not blame me for it.

Something about the magicSysRq keys is that they can be used in any sequence and in any way to achieve the required objective, for example you can just press Alt+SysRq+B to do something like a cold reboot.

Enabling Temperature Sensors in Linux

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=1988>

Most computers these days come with a myriad of sensors to monitor the temperature of your computer. These sensors are generally located on the processor and the motherboard, and you might also have sensors on your video card. On top of that, all S.M.A.R.T-enabled hard drives have built-in temperature monitoring.

The temperature of your computer is a vital thing to keep track of - heat and computers don't mix very well. Unfortunately, Ultimate Edition doesn't setup your computer's sensors automatically; but you can follow these steps to enable the temperature sensors in your computer in Ultimate Edition, or any other version of Linux. While sensor-monitoring is somewhat hardware dependant, this guide will work for most users. It involves heavy use of the command-line, but don't worry - I will walk you through it step-by-step.

1. Installing the sensor libraries

First thing's first - you need to install the libraries that allow Linux to read your sensors. To do this, install the lm-sensors library, by running the command:

```
sudo apt-get install lm-sensors
```

This will install the libraries for your motherboard's sensors. For your hard-disk sensors, you'll want to install hddtemp:

```
sudo apt-get install hddtemp
```

In Ultimate Edition, the install will ask you several questions. First it will ask if it should run SUID root,

select "yes." It will then ask you for an interval for logging the temperature to a file; since we are going to have an applet display our system temperatures for us, this isn't necessary, so most users will be fine leaving the default of '0' and pressing enter; if you wish to log this data, however, I'd recommend a value between 2 and 10 seconds. Next, it will ask if it should run as a daemon; select yes, and leave the default values for hostname and port. Finally, it will ask if you wish for it to run on startup - select "yes."

2. Running sensors-detect

Now that your sensor libraries are installed, you need to detect your sensors! Run the command:

```
sudo sensors-detect
```

Which will probe your system for sensors. Answer "YES" to all questions! Don't just hit enter, type "YES", because at the end there will be a question for which the default answer is "no", and we'll want to answer in the affirmative.

The sensors-detect program will scan your system, and then give you a summary, stating which sensors it has found. It will then say: I will now generate the commands needed to load the required modules. After you hit ENTER to continue, it will ask, Do you want to add these lines to /etc/modules automatically? (yes/NO) This is the question we want to make sure we answer YES to.

3. Loading the modules

Since we answered YES to the previous question, our sensor modules will be loaded by default the next time we start up. But since we don't want to have to reboot, we're going to use the information we got from the sensors-detect script to load the modules ourselves, this time only. Right above the last question will appear a list of modules that you should load, in the form of:

```
#----cut here----
```

```
# Chip drivers
```

```
smc47m1
```

```
#----cut here----
```

You may have more, or different, items listed - that's fine! What we want to do now, to load these modules, is use the modprobe command, as follows:

```
sudo modprobe [module name]
```

So, in my case, I would type:

```
sudo modprobe smsc47m1
```

If all goes well, you should be returned to the command-line, without any output.

4. *Monitoring the sensors!*

Wow, that was a lot of work! Now, let's see the rewards. On the command line, you can simply run the sensors command; this will output the information from your motherboard's sensors.

However, we'd rather have a graphical interface for checking up on our hardware, so let's install an applet for our Gnome desktop to keep an eye on our system's temperature. Run the command:

sudo apt-get install sensors-applet to install the applet. Now, add the applet by right-clicking on your desktop panel, selecting "Add to Panel," and you will now see a "Hardware Sensors Monitor" applet in the System & Hardware section. Click and drag this to your panel to add it.

The applet will now say that you haven't enabled any sensors; right click on the applet and open its preferences. The first screen is for general settings:

Howto Setup Bluetooth Keyboard and Mouse in Ultimate Edition

<http://forumubuntusoftware.info/viewtopic.php?f=7&t=1990>

We need the MAC address (e.g. 00:00:00:00:00) of the mouse and keyboard. You can use KEYBOARD_ADDR and MOUSE_ADDR where you should find the addresses for the keyboard and mouse respectively. Press the button on the mouse that makes it visible to be found by the computer. Do the same for the keyboard. Now open a terminal window and run the following command

```
hcitool scan
```

```
Scanning ...
```

```
KEYBOARD_ADDR Microsoft Wireless Keyboard
```

```
MOUSE_ADDR Microsoft Mouse
```

```
Adding the Keyboard and Mouse
```

Now we need to add the keyboard and mouse to the bluetooth configuration files. Run the following command to pop up GEdit

```
sudo gedit /etc/bluetooth/hcid.conf
```

You may be asked for your password, this is because we used sudo.

At the end of the file, add the following (replacing KEYBOARD_ADDR and MOUSE_ADDR for the keyboard and mouse MAC addresses as found earlier)

```
device KEYBOARD_ADDR {  
name "Microsoft Wireless Keyboard";  
auth enable;  
encrypt enable;  
}
```

```
device MOUSE_ADDR {  
name "Microsoft Mouse";  
}
```

Now you need to restart the bluetooth subsystem so that it refreshes it's configuration file.

```
sudo /etc/init.d/bluez-utils restart
```

```
* Restarting Bluetooth services... [ ok ]
```

Pairing the Devices

You now need to pair the devices with the computer. Do not press any buttons on the keyboard as we'll need to use it to enter a passcode so we can pair. Run the following command

```
sudo hidd --search
```

```
Searching ...
```

```
Connecting to device MOUSE_ADDR
```

```
Connecting to device KEYBOARD_ADDR
```

They could pair with the computer in any order, you will need to remember which one is the

keyboard. As soon as Connecting to device KEYBOARD_ADDR appears you must enter a PIN code into the keyboard. It must consist of numbers not using the numpad, somewhere between 4 and 8 should be fine. Type this number in to the keyboard and press Return.

A window should pop up on your computer asking you for the number you just entered on the keyboard.

You should now be set up. The devices should automatically reconnect when they go to sleep and when your computer boots up.

Troubleshooting

If you have followed all the steps above and you find your mouse or keyboard don't automatically reconnect, we can fix it. Run the following command in a terminal

```
sudo gedit /etc/default/bluetooth
```

Find the following lines

```
HID_ENABLED=0
```

```
HID_OPTIONS=""
```

Change them to

```
HID_ENABLED=1
```

```
HID_OPTIONS="--master --connect KEYBOARD_ADDR --connect MOUSE_ADDR --server"
```

Now reboot and hopefully they'll automatically connect (give them a few seconds to connect after you move the mouse/press a key).

10 things you didn't know you could do in Ultimate Edition

<http://forumubuntusoftware.info/viewforum.php?f=7&start=125>

Create website links that automatically install software

Sometimes if you're trying to help somebody fix a problem, you'll have to tell them how to install software. Yet for some Ultimate Edition newbies, even this can be confusing. The solution is to create a

"software install" hyperlink within a web page (such as a forum posting), new email window, or Pidgin message window. To do this, simply click the "create link" button on the web page or within the email (the precise name of this will vary depending on the software/website used), and then type apt:packagename in the URL field, replacing packagename with the precise name of the package as listed in Synaptic.

For example, let's say you want to tell somebody how to install the thunar package, a lightweight file manager. If you're creating an email with the instructions, ensure the new mail uses HTML (ensure HTML is checked on the Format menu), and then click Insert -> Link. In the URL field, delete what's there, and type apt:thunar. Don't worry about the Description field—leave it with the default contents that will probably mirror what's in the URL field. Then click the Close button.

Note

that there's a slight bug in Evolution that means, for some reason, the hyperlink won't actually appear as a link until you type some more into the new mail window, or click the Send button.

Perhaps it goes without saying that should you ever receive such a link in an email, or see one on a website, you should be very wary (especially if there are also additional instructions telling you to add a new software repository). It would be easy to disguise a malicious link as something seemingly benign, although you will always be prompted to confirm the choice of software before installation.

Do stuff without touching the mouse

If you want to run an application, just hit Alt + F2 and then type the name of the program. If it needs to run with root privileges, just type gksu beforehand. For example, to run GNOME Terminal, you would type terminal. To run Gedit, type gedit. If the program is command line, check the Run In Terminal box (use the Tab key to move from field to field in the dialog box and hit the Space to select a field). This will then open a terminal window and run the command, but be aware that the terminal window will then close as soon as the command has finished, so you won't be able to inspect the output. Want to browse to a file system location, but too lazy to grab the mouse and click

the Places menu? Hit the forward slash (/), and then type the path into the dialog box that appears. Want to rename a file but don't want to use the mouse? Just ensure the file is highlighted (use the cursor keys to highlight it if necessary) and hit F2. Then type the new filename. To change the file extension too (which isn't highlighted for deletion by default), just hit Ctrl + a and then type the complete new filename.

Instantly Search Google for Any Word or Phrase

Have you ever been reading a document and wanted to look up something in Google? In Firefox you can just highlight the word or phrase, right-click it, and select Search Google. However, what if you're reading, say, a PDF file? Or a man page in a terminal window? A very simple but effective solution is Googlizer, which you can install using Synaptic. Once installed, it's added to the Applications -> Internet menu, so you'll have to manually drag and drop it to a blank spot on the panel for quicker access. How it works is simple. Highlight any text, in any application, and then click Googlizer's icon to instantly search Google. If a Firefox window is open, a new tab will be added showing the search results. Otherwise, Firefox will be started, and the search results shown. Give it a try. It's one of those simple things that might just change the way you work forever.

Googlizer can be personalized so that it searches the version of Google localized to your country, or even a non-Google search engine. To do this, you'll need to discover the search URL for the engine you want to use. To do so, just perform a search using either the localized version of Google (for example, <http://www.google.co.uk>, if you live in the UK) or a different search engine. Then look at the URL for the part where your search term appears, and highlight/copy all that comes before. If I search for Ultimate Edition Kung Fu using <http://www.google.co.uk>, I get the following URL for the search

results page:

[http://www.google.co.uk/search?hl=en&q= ... arch&meta=](http://www.google.co.uk/search?hl=en&q=...arch&meta=)

...so I chop the end off, from the Ultimate Edition+Kung+Fu part onward, and I'm left with following,

which I copy into the clipboard (highlight the text, and hit Ctrl + c):

<http://www.google.co.uk/search?hl=en&q=>

Once you have the information, right-click the Googlizer panel icon, and select Properties. In the Command text field, add `--url` after googlizer, and then paste your Google URL. For example, I ended up with the following:

`googlizer --url http://www.google.co.uk/search?hl=en&q=`

You can also change the icon if you want by clicking the icon preview at the top left of the dialog box. When finished, click the Close button, and then test the new localized search. Here are some URLs that will make Googlizer use other search engines—just add these addresses after the `--url` part of the Command text field, as described earlier:

Yahoo.com: <http://search.yahoo.com/search?p=>

Ask.com: <http://www.ask.com/web?q=>

Microsoft Live: <http://search.live.com/results.aspx?q=>

Create a File Delete Command That Uses the Trash

The `rm` command doesn't have a trash facility. Once files are deleted, they're gone forever.

However, you can create your own trash command, which, when used at the prompt, will move files and/or folders to Ultimate Edition's standard trash folder. The files can then be recovered, if desired, or

permanently deleted in the usual way by emptying the Trash folder. To add the new command, you'll have to create an alias. You'll need to edit the `.bashrc` file in your `/home` folder and add a line to the bottom, as follows:

1. Open a terminal window, and type `gedit ~/.bashrc`.

2. At the bottom of the file that appears, add the following new line:

```
alias trash="mv -t ~/.local/share/Trash/files --backup=t"
```

3. Save the file, close Gedit, and open a new terminal window to test your new command. To delete `filename.doc`, for example, you would type `trash filename.doc`. The new command will work on

folders too, and multiple files/folders can be specified one after the other (for example, trash filename1.doc filename2.doc).

Repair Windows from Within Ultimate Edition

If Windows is refusing to boot, for whatever reason, you can try repairing the file system from within Ultimate Edition. Use Synaptic to search for the ntfsprogs package. Once it's installed, unmount your Windows partition (if it's mounted), and type `sudo ntfsfix /dev/sda1` to check and fix the partition (assuming your Windows partition is /dev/sda1—likely if you installed Ultimate Edition in a dual-boot configuration on a computer already running Windows). This tip is also useful if you see the “Cannot mount volume” error when attempting to access your Windows partition from within Ultimate Edition.

Dump the Text on a Virtual Console to a File

If you're trying to fix a problem, you might want to capture the output of a command for reproduction on a website forum, along with the command you typed to get the results. If you're working in a terminal window, you can cut and paste, but what if you're working at a virtual console? If you simply want to capture the result of a command, just redirect the output:

```
ls > output.txt 2>&1
```

This will send both the output and error output (if any) of the `ls` command to `output.txt`. If you want to capture the command you typed and any other command-line detritus (including output), use the `screendump` command. The following will send everything currently on the current screen (command-line prompts included) to a text file called `output.txt`:

```
sudo screendump > output.txt
```

The command has to be issued as root because of permission issues, but the resulting file will be owned by you.

Instantly Hide a File or Folder

Any file or folder whose name is preceded with a period (.) is hidden from view in Nautilus and also won't appear in the list of shell commands such as `ls`, unless the user specifically chooses to view hidden files (`ls -a`, or clicking View → Show Hidden Files in Nautilus). So to hide a file or folder, just rename it (select it and hit F2), and then put a period in front of the filename. Gone. If the file doesn't vanish, hit F5 to refresh the file listing. To return the file to view, just remove the period. If you want to make a file disappear from Nautilus' view of files (including the desktop) but still appear in command-line listings, add a tilde symbol (~) to the end. For example, to hide `partypicture.jpg`, change its filename to `partypicture.jpg~`. To hide text file, change its name to `text file~`.

Print at the Command Line

You can quickly send text or configuration files to the printer using the `lp` command. For example, to print the `/etc/fstab` configuration file, you would type `lp /etc/fstab`. The formatting of the printed page is rough (no margins, and non-proportional font used), but it's OK for quick hard copy viewing. If you want you can set a top page margins using the `-o page-top=` command option. The following will print the same file with a one-inch (72 pica) margin at the top:

```
lp -o page-top=72 /etc/fstab
```

Note that for the `lp` command to work, you'll need to first make your printer the system default (even if it's the only one attached). To do so, click System → Preferences → Default Printer. Select your printer, and then click Set Default. Then click Close.

Listen to MP3s when no GUI is running

So you've tweaked Ultimate Edition into a state of disrepair. Any hope of a GUI is a pipe-dream, at least for the moment. While you hack away fixing things, wouldn't it be nice to have some music to console you at the console?

Just switch to an unused virtual console, login, and type **sudo apt-get install vlc**.

VLC is a GUI media playback application normally used on the desktop, but it can also run with a text-mode interface—just start it with the `-I ncurses` command option (note that's a capital I, not L). For example, to play back `filename.mp3`, I would type `vlc -I ncurses filename.mp3`. Multiple files can be specified one after the other, thus creating a playlist, or a wildcard can be used to playback all files in a particular folder (ie `vlc -I ncurses ~/Music/*.mp3`). Use the `a` and `z` keys to alter the volume. If playback is still too quiet (or too loud), switch to another virtual console and type `alsamixer`. Hey, presto—primitive but useful text-mode faders. Use the left and right cursor keys to move between faders. Use the up and down keys to change the values. Hit `Escape` to quit.

Turn your desktop into your /home folder

Do you use your desktop as a dumping ground for files, and pretty much ignore your actual `/home` folder, which is where you should store things? If so, you might be interested in this tweak, which effectively makes Ultimate Edition use your `/home` folder for the desktop, instead of the actual `/home/username/Desktop` folder. Anything saved to the desktop, such as files/folders dropped there, will be placed in your `/home` folder. Additionally, anything in your `/home` folder will appear on the desktop. To give this a try, start `gconf-editor` (hit `Alt + F2` and type `gconf-editor`) and navigate to `/apps/nautilus/preferences` and put a check alongside `desktop_is_home_dir`. Then log out and back in again. Remember that the contents of your desktop haven't vanished. They're still in the Desktop folder in your `/home` folder.

BONUS TIP! Quickly Set Wallpaper with One Click

The easiest way of setting your own picture as a desktop wallpaper is to click and drag the image to the desktop using the middle mouse button (if the image is already on the desktop, then click and drag it a few inches to the left/right). On most modern mice, the middle mouse button is the scroll wheel, which also doubles as a third mouse button. On the menu that appears when you release the button, click `Set as Background`. If that sounds a little too unorthodox for you (it can be hard to use the middle mouse button), you can also use `Synaptic` to install the `nautilus-wallpaper` package, which adds a simple `Set as Wallpaper` option to the menu that appears when you right-click an image file. After installation, you'll need to log out and then in again before the option becomes visible.

Have fun !!!