Linux as a whole is compromised - The Cyberix Network

The Cyberix Network

>Gentoo and Funtoo

> >A new evolution of Funtoo's core ideas is being built from the ground up, unencumbered by legacy toolchains, with a target for an initial public release in early 2026.

They are obviously talking about finally eliminating the python dependency required by portage (and hopefully bash too).

The Funtoo project is probably not possible to maintain anymore due to the fact that Gentoo has been screwed up so badly by the Google employees currently ruining it. The new EAPI specs are terrible. Along with the various retarded changes over the past 4-5 years. Like what they've done to the kernel and OpenRC along with the whole every package now making a user account thing.

Python's a huge pain in the ass to update when it's required to run the basic parts of the OS stack. If you weren't around for the 2.x -> 3.x switch over then consider yourself lucky. Dealing with python targets in Gentoo has always been a lot of trouble. It could have all been easily avoided by using C or even basic sh scripting instead. Sourcemage is much easier to maintain for an example and it basically does the same thing that portage does.

When you're just using it for Portage it isn't too bad (aside from being a slow piece of shit). But when you have say 3 ebuilds all wanting different versions of python then you get into a mess.

This is why python and perl have always been avoided for the "base system" of most Linux distros (to borrow a term from the BSDs). Instead people rely on sh scripts because they're more portable and don't pull in so much garbage that's hard to maintain. Same goes for bash vs. sh. It's easier to write proper portable scripts than to rely on bash. Having a hard requirement on bash just makes it much harder to maintain the system and restricts choice.

If you were going to have something like python in your "base system" using perl would be a better way to go. But perl has its own host of problems. Therefor, anyone with good sense that thinks they need something more than sh scripting will always go for C instead. C is much faster at run time, it's easier to maintain within your "base system" since libc is already a basic requirement for everything else and it's portable.

>bash
The hard requirement on bash means you can't replace the root shell with anything else like fish, zsh, ksh or whatever the fuck else you'd prefer. It should be easy to swap the root shell but 25+ years of legacy cruft has always prevented this.

Portage has always been a slow piece of shit. It's badly designed. There were multiple plans to eventually replace it but like most things people have just added more bullshit on top over the years and no one wants to break existing ebuilds. Hence why the few attempts at replacing it with something written in C have not caught on.

Portage was forgivable for years due to what you got in return (USE flags and the community providing ebuilds). But now that the Gentoo project has been derailed on purpose there is no point in continuing to deal with it.

It would take me far too long to explain what all has been ruined in Gentoo since about 2015. Just know it's really bad now and most of the power users have been leaving in droves for the past 5-7 years because we all saw the writing on the wall. When they started banning people, deleting entire forums and openly bragging about it on the mailing lists we all knew it was over. There was hope that maybe we could take the project back through the council but they've subverted the democratic political system to the point not that it's impossible to vote them out. Hence why Funtoo was started in the first place.

>It would take me far too long to explain what all has been ruined in Gentoo since about 2015
I don't have time to pull up links. But off the top of my head;
>Kernel
Currently the guy maintain the gentoo-kernel is shipping a bunch of patches by default that you do not want on your system. All old timers use vanillia-kernel for this reason.
>init
There is a guy camping the OpenRC project that has refused PRs from the community for years. He made a bunch of bad changes most everyone didn't want. Most old timers run and maintain an older version of OpenRC that you have to get from GURU now
>systemd shims
A bunch of projects we used to maintain inside the project were killed off by this same group on purpose and replaced with stuff pulled from systemd's repo. Usually with underhanded tactics. For example, the guy squatting on OpenRC sat on a PR to fix a bug in opentmpfiles for 2 years. Then declared it was a security risk on day and they replaced it with systemd's tmpfiles replacement. Which had the exact same bug and as far as I know it hasn't been fixed. They did the same thing with consolekit and forced logind on to all systems randomly one day despite the community not wanting it. Along with a bunch of other stuff like that.

Those are just the ones I remember there have been tons of other things they did. The new EAPI stuff is a huge mess no one wanted. They've basically made themselves dictators for life by abusing the voting for the political process. They've silenced and banned anyone that called them out over the years. Google and IBM employees openly brag about this stuff on the mailing lists.

Gentoo is basically Fedora built from source now. The default OpenRC profile even ships a binary blob for Rust's compiler by default these days along with a host of other stuff you don't want.

They try to coast on the reputation the project got 15+ years ago. Anyone that tries to contribute in a positive way gets banned pretty quickly.

One important one I almost forgot is the /usr merge. For years we maintained the system in a way where you had the option to run /usr on its own partition (or even on a network drive). They've recently pushed a policy where the /usr merge is going to be a hard requirement. They're in the process of making that happen right now. First they pushed merged /usr as the new default then they plan to take the option away all together in the coming year. /usr on its own partition is important for a lot of people running systems that aren't "Fedora from source".

The Gentoo project basically got subverted when Google started using it as a base for ChromeOS. You're just beta testing for them at this point. It's their project now.

While they've been doing all of this they've also blocked a lot of things that would have improved the actual distro as a whole. Like fixing the various problems with portage.

In other words; They've done just about everything in their power to remove end user choice. Which is what the project was founded on and what it's supposed to be about. You can't even run a static /dev these days without tons of hacking.

Most of the old timers are only still there because they've invested so many years into the project and there is no where else to really go. We all use local overlays and maintain patches for older ebuilds. Plenty of people have fled or have made plans to leave though. I ended up on the BSDs.

I developed for Gentoo for years. Now I mostly contribute to OpenBSD. I tried working with FreeBSD but it's suffering from a lot of the same problems as the major Linux distros now. So I'm mostly only an end user and don't send back patches.

I have no idea what Drobbins is up to now. I haven't spoken with him since 2020 or so. I did contribute to Funtoo for awhile but I have limited time so I wasn't able to spend as much time on it as I'd like. I also didn't like some of the decisions he made.

> >deleting entire forums and openly bragging about it on the mailing lists we all knew it was over
>AkA. what ended up spawning otw20..

I should have touched on this as well. Yes I'm referring to that.

What happened is this; They started applying heavy handed censorship to the forums. It got to the point where you couldn't talk about a lot of subjects on the support/technical sub-forums without the thread being closed and if you kept up having your account banned. Most of this had to do with the usual stuff that'll get you in trouble on all mainstream Linux forums these days. systemd, rust, IBM/Microsoft/Google influence, FreeDesktop/Red Hat and related subjects. All of this became something you weren't allowed to talk about at all. Usually faggot janny stuff like closing threads and declaring things to be "off topic" and/or "bad for the forum". You know what I mean.

This ended up driving a lot of old timers to the Off Topic (Off the Wall) forum where you could talk about anything. So really long threads ended up there where this stuff was talked about and where people that wanted to run systems that weren't close to the new defaults could gather, work together, share ebuilds/overlays and all that good stuff.

This was obviously a problem for the group that subverted the project. Since users were not only maintaining ebuilds and sharing tips they were also calling these people out by name, archiving their posts on the mailing lists and documenting things like the companies they were working for (Google, M$, IBM/Red Hat etc.) and showing clearly who now had influence and was paying to ruin the project and subvert the democratic process that was supposed to keep it fair for the users.

So what they did was claim that the thread where people had been ranting about politics for years was suddenly this huge problem due to racism and Donald Trump. Then they used that as an excuse to delete the entire forum and ban a bunch of users. Of course it was simply an excuse to nuke those threads.

>Currently the guy maintain the gentoo-kernel is shipping a bunch of patches by default that you do not want on your system



itsaman.png

Its worse than you know.



itsaman2.png

If you can read moon. I encourage you to check out the really shady company he's currently working for in Japan.



nipponspookware.png

Really spooky company.



spooky.png

The rabbit hole is endless.



spookcompany.png

Nothing to see here. It's permanently closed and they just rented office space next to this location because the rent was cheap. It's operating multiple office locations all over Asia despite the parent company being permanently closed.

Don't pay any attention to the fact that one of our employees came out of nowhere and started managing a bunch of Linux kernel projects with no prior qualifications. Oh and don't worry about the fact that they're a ghost on paper and somehow managed to change their last name on a whim while living in a country that's known to be hostile towards gaijin where it's extremely difficult for someone that married a native to get their name listed in the family registry.

There is absolutely nothing strange going on here at all.

>*THIS* is the person is in charge of gentoo now?
It isn't any one person, but yes they hold a lot of sway now. They are in charge of everything to do with the kernel at the moment.

Gentoo has a council where people are supposed to be voted in by the community and not hold life time positions. But around 10-15 years ago a bunch of employees of Google, IBM and these various spook companies like the one shown above gamed the system to get voted into all the positions. Once they secured power they changed a bunch of rules and ensured they could hold life time positions and control the future of the project. Now it's impossible to vote them out because they control the actual voting process. So they just vote themselves in over and over again.

That person is just the tip of the ice berg. I could do it several more times for the council and other important positions within the community structure surrounding the project.

The original author of portage/Gentoo said many years ago that he fucked up really badly when he let the community talk him into bringing in a democratic process instead of remaining BDFL. He left for a few years to wageslave because he couldn't get any donations for Portage/Gentoo to live off of and when he returned and tried to improve Gentoo he was denied access to work on his own creation.

(cont in next post)

Attachments:

>Okay, lets just move out to BSD or NixOS!
Oh yeah? Well I have bad news for you. NixOS is just Fedora from source too. Since the entire project relies on systemd and all the things mentioned above that were forced into Gentoo over the last decade. Just because you're using some autistic package manager to configure it doesn't mean it's any different at the end of the day. Guix suffers from the same problem. Just try finding any example config files that don't rely directly on everything Freedesktop and IBM are pushing hard like Rust dependency for your display manager or a config files offering anything but running a pure wayland set-up. You don't even get any good docs either either (Guix are better than Nix's though). You can't even get help with Guix most of the time because being GNU zealots they refuse to help anyone running one binary blob on their system. So 99% of Linux users.

Not to mention all the other issues with NixOS and Guix system. Anyone that's tried to run it on their day-to-day machine knows exactly what I'm talking about. It's pretty useless for a home users.

That's the main issue with Linux as a whole now. Nothing is geared to the home user. Everything is centered around providing crappy software for shared corporate environment users. Most of this shit like systemd, wayland, PAM, polkit, dbus, logind etc. has no use for a home user. A home user doesn't need multi-seat and shouldn't be concerned about another user logged into the same PC being able to see things they're doing on the machine. 99% of regular users will never use a machine with multiple people using the machine at the same time and if they are they would trust them 99% of the time. UNIX-like OSs were already multi-user anyway since UNIX v1.

They've basically shoved in a bunch of spyware, malware and backdoors by fear mongering about security.

Oh and don't get me wrong. I'm not against IPC because I dislike dbus. I dislike dbus because it's terrible software. I'll spare you the rant just understand it's purely retarded especially when it requires shit like javascript under the hood. It's also a massive security hole. IPC should not run in userspace in the first place. We didn't need something like dbus in the first place since we have pipes.

Same for stuff like polkit. I don't mind being able to grant more privileges to an application that needs to do something normally restricted by the user account that's not root. But this was totally unneeded we already had things like gksudo. But gksudo was deem'd not secure and purged from every last distro's repos in short order for no real reason. Just lies about how it was suddenly not secure for a user to be prompted to enter a password by the GUI.

Then we have the good old lie of suid Xorg being deem'd not-secure because
>so much buggy hard to audit code
their "secure" replacement? Millions of lines of harder to audit code (systemd+logind) running as root at all times.

Then these same people tell you it's not secure for one application being able to see what another application is doing on the same desktop session. When we had multiple ways to deal with that in Xorg already and there were easy ways to improve Xorg's security without something like logind/consolekit/etc managing things. They claim your user account having direct access to your input devices and graphics card is also not secure. Despite their crapware being granted direct access to it while it has thousands of security bugs marked "WONTFIX". These are the people that gave you the ability to gain root by using a user account named "0day" after all.

It's a real mess...

They do not care about security. They care about you not having access to your own hardware and software. They want end users using dumb terminals.

Oh and before the usual X vs. Wayland derail comes along. I'm not a fan of Xlibre either. It has made a ton of dumb decisions starting right off and it feels a lot like controlled op to me. For example, why did they feel the need to go so many versions back to fork off? That and a bunch of other bad decisions. I followed the mailing list in the first months and I saw a lot of things going on that seemed suspect. It feels like it's designed to waste time and give Freedesktop an easy boogeyman to point at while tricking people that would contribute to an actual fork a time sink to waste their effort on.

OpenBSD has maintained a patch set (some call it a fork) that has done several positive things to improve the build system and security of the project. Yet their patches are almost never accepted by upstream.

Then we have the old lie being parroted that
>The original Xorg developers don't even want to work on it
That's not true. Freedesktop (IBM) gained control over the project and have blocked people from sending PRs and patches for years now. Most of the security fear mongering they do does not apply at all to a home user. All their claims about wayland being somehow more secure boil down to things that aren't a problem for a home user that just wants to play games and get shit done while having a stable GUI. Then we have Qt and GTK. Where the GTK project was also taken over and ruined on purpose despite Gnome/GTK v2 being great software. Which was originally started by the community so we wouldn't be tied to Qt's licensing.

Like I said before; There are spooks everywhere. EEE in full effect. The Linux desktop today is in much worse shape than it was in 2004. It's certainly much less usable for the average person.

If someone has already pwn'd the system to the point where you're worried about them seeing other applications running the wayland security theater is not going to help you in the first place.

>If someone has already pwn'd the system to the point where you're worried about them seeing other applications running the wayland security theater is not going to help you in the first place.
(cont.)

The main valid security issue for this would be the web browser engines running on the local system allowing rouge javascripts (and related cancer) seeing other stuff running on your desktop. Well, who's fault is that?

In the 90s-mid 2000s the web browser wasn't this massive security hole by design. Nor were the web based specifications. The web was never designed to be this SaaS cancer where everything is LARPing as a local application. It was intended to be a protocol where you requested an HTML page (with a style sheet later on). A web server would generate a page for you and send it to you. You'd read it. No issue at all here with it being able to exploit your system. As long as you weren't running Microsoft's ActiveX. Even early uses of javascript like AJAX weren't a security exploit waiting to happen since all it was doing was fetching content dynamically in the background and serving it to you without having to reload the entire page. AJAX was considered an ugly hack until something better came along.

That something better wasn't supposed to be running over the http protocol either. It was supposed to be a new protocol. Where we could code applications that interacted with remote servers. Where data would be sent over the wire to the end user and used by an application running locally.

That was sane development and why different stuff ran over different protocols for decades before this web-based garbage came along thanks to Google. But google bought off the W3C and now we no longer have real web standards.

They shove everything over http because it's easy to centralize and spy on what the end user is doing. The web was intended to be a global library of information not for all this crap we're using it for now.

(cont.)

Even with the modern browsers the proper solution wasn't to ruin the desktop. Instead, you simply run them in a jail/chroot/some kind of container. Give it its own X server that can interact with the main X server running on that system so you get basic stuff like copy/paste that works. This is much better security than what something like wayland offers. Since you can keep them separated from each other for real and sandboxed. On FreeBSD the jails were the best option and jails were around for decades before crap like all these modern container formats like Docker came on to the scene.

Or you go the OpenBSD route with pledge/unveil. OpenBSD is the only OS in current use where you can safely run a browser. If any rouge script tries to do something it isn't supposed to it crashes and dumps core. That's running with no container on an X server too. It's easy enough to do it if you give a fuck about the end users for real and your kernel's security aspects aren't coming directly from the NSA. The situation on Linux in this respect is really dire.

The fact that we let the web browsers get this out of control is really the issue. I haven't even touched on what they've done to the web itself since the late 2000s here. What with the https hard requirement for everything, shit like cloudflare and other MitM services being required if you don't want to get ddos'd off the web and the fact that only a handful of spook companies are the ones allowed to issue valid certs to people running websites now.

It's like social media. Everything is centralized now for spying/ad purposes. When we had email, usenet, and all the rest back in the 1970s-1980s. We could have improved and extended protocols like that and they could have never been censored or monitored at this level. But then it would make it much harder for 9,000-eye countries to spy on everyone and control the flow of information over the internet.

>When we had email, usenet, and all the rest back in the 1970s-1980s.
While I'm effort posting about this I might as well mention two things close to my heart: Fidonet and old school web forums.

I put a lot of work into web forums back in the day and I was an avid user of Fidonet. You could call these early "social networks" along with USENET. Let's look at what they did to kill them off.

>Fidonet
A lot of people showed up in the mid-90s to murder this store-and-forward protocol. It's still active today and it's one of the few ways you can talk to people in places like North Korea. I know plenty of people forced to use it today to get real information because the web is so censored in their countries. These are college professors, doctors and people like that. Fidonet can run over any kind of link even very low bandwidth/high latency links like HAM radio and phone lines. Yet no one tries to improve it (or make something new based on this concept) because that obviously wouldn't be good for the people spying on us.

>USENET
It was killed off in the late 2000s by the US Congress. Who demanded all ISPs stop giving users free access because of fear mongering about pizza being on it. I encourage you to look into that if you get the time. Again, impossible to really censor or delete anything once it's on the network. The network has also been centralized in recent years by a handful of companies and if you go looking into them you'll find the same people behind the scenes: spooks everywhere.

(cont.)

>Web forums
These were killed off by Google in the mid-late 2000s through the adsense program. They bought off all the admins running large forums with ad money. People were making $10k+ a month sometimes off ads. But embedding adsense into your forum's code allowed google to spy on all your users and build profiles on them. Then they started kicking people out of the ad program unless they censored discussion. Lots of people gave up running forums because of this when the ad money stopped coming in. I knew plenty of people that quit their jobs in the early-mid 2000s to run forums full time. People were buying and selling them for thousands of dollars on places like webhostingtalk in those days. They didn't care one bit about google being able to spy on their users they just wanted the paycheck much like modern youtubers who all self-censor to stay in google's good graces.

Well in the late 2000s they finally screwed all these sell-outs over for good. First they started to derank them from search results. Claiming they were abusing google's search engine through SEO despite letting large media companies do the same thing without punishing them at all.

After that they started to derank forums all together. Today you can't even get search results from for old forum posts even if you add "forum" or "site:forumurl.tld" to your search query.

They deranked all these forums while promoting websites like major social networks, reddit and all the usual cancer like quora instead.

What I'm saying is forums did not die organically. They were killed off on purpose by hiding them from search results. It was done because they wanted to spy on end users and build better profiles on them by herding them into services they controlled instead of sending them to independent websites. Why pay all that ad money out every month to buy off admins to allow you to spy on their users when you can just run your own website and gather even more data about them?

The segments above may seem unrelated, but here is my greater point. The centralization we saw happen to the web from the mid-2000s and really ramp up through the 2010s is what's happening to Linux distros now.

You have thousands of different distros and claiming to be in-fighting. But if you look beyond the surface you will see the same thing happening to them now that we saw with the web back then.

At their core they're all the same. Same libc, same kernel, same "base system" comprised of the same software all coming from spook sources. Software designed to spy on the end users as much as possible. All coming from the same handful of sources; IBM, Microsoft, Google, Amazon, Microsoft and all the other front companies for the spook community. We even see several patches coming directly from the NSA itself without shame. End users are not aware of what's going on because they do not understand what's happening behind the scenes.

The users feel like they're gaining security, privacy and sticking it to the man. When in reality the same man is in charge of this new OS they're switching to. Meanwhile, all the old timers are either getting very depressed and quitting because they see the writing on the wall or they're checked out all together. Or they've fully sold out for cash (see: Linus, Stallman and many others) because they're getting millions of dollars a year to keep their mouths shut and play along. We have young people the "fucking love FOSS" getting tricked into thinking they're making a difference in the world. All while relying on a software stack that's worse than old M$ ever was and using programming languages being promoted by spooks.

Of course, UNIX was a spook creation to start with along with the internet as a whole. But at least for a short time there was hope that maybe we could take it from them and do something positive with it. But all of that work has been thrown out the window now.

>Or they've fully sold out for cash (see: Linus, Stallman and many others)
MIT is a spook school anon. So is Berkeley and most of the other major tech schools in the west. In fact, Berkeley is the last name of a royal family that has been doing shady stuff for hundreds (probably thousands) of years. Check out the peerage sometime. It's a family tree/list of royal families. You'll find they've been intermarrying for a long time.

Spooks are not a modern creation that started after WW2. They've been around as long as we've had a military (all of human history). Before the CIA there was the OSS. Before the OSS there was the office of naval intelligence. That's just in America. Most spook operations were run out of the Navys of major nations because obviously you'd want your Navy that projects your power all over the world to have a spying organization, right?

Stallman in particular is said not to have written a piece of code or even touched a keyboard for decades now due to health reasons. He didn't even produce emacs despite getting credit for it. He didn't write the GPL license either. Yet he's held a cushy job at the MIT for decades now getting paid a huge yearly salary for producing nothing of value and giving a talk now and again where he says nothing of value. Don't you find that strange?

Don't you also find it strange that despite MIT and Berkeley schools being very close and working so closely together for decades they started a holy war (GPL vs. BSD) that has divided the FOSS community since before the web was a thing? Typical divide and conquer tactics we see everywhere in life. Just like the false right/left divide in politics and in most other aspects of life.

There is a phrase for the above: Ordo ab Chao. Look up what it means.

Oh and please don't take google or whatever AI answer you get at face value. I assure you this phrase is much older than the 1800s and was being widely used before whatever sekrit society they're giving credit for creating it (freemasons or scottish rite would be my guess on what google/AI returns, let me know if I'm correct).

It is a LATIN phrase after all, isn't it? That means it must be pretty damn old. You'll find a lot of Latin in your day-to-day life if you bother looking for it. Check out what the Latin phrases on your $1 bill mean sometime.

These organizations are very very old. They pre-date that Jesus guy they claim walked the Earth as an example. Just like a lot of the Gods from Greek/Roman/whatever times are much older than those societies that are credited for creating them. If you bother to study history (and studying history is really hard since the victors write it) you'll see the same "Gods" being mentioned over and over again by different names through various societies. They're still worshiped today by lots of

people that do not talk about these facts openly.

I would link you to a collection of papers on this type of stuff that would blow your mind. Sadly, if I did I'd eat a 2 week ban instantly. I can't even mention the author here by name.

It isn't just here either. It's every major website on the web these days. The guy has done deep dives on this subject for the past 30-odd years and publishes multiple papers about it every month. In addition to the work he does on other topics like hard science. He's the most censored person I know of on the internet.

All I can do is encourage you to try to seek it out for yourself. As this is kind of off topic. Well not totally because that family literally runs that tech school (go look at who founded MIT while you're at it). But this is more about Gentoo/Linux so I won't say any more about it here.

Spook spam WILL pick up the moment this is posted When they do, it'll prove they actively get alerted when certain keywords are posted anywhere on the web :^). Don't feed them.

Fun fact: They can't post copyrighted material. Since they operate under rules laid out in the Department of Defense Interactive Internet Activities program. Now you know why the frog and wojak spam picked up so heavily and why they rage about people that post anime and Milhouse.

>I think I will move to NetBSD
They haven't left the BSDs alone. As for your second question. They like having control at multiple levels. If people were creating software that actually respected the users at some point a lot of users would start asking hard questions about why the hardware is so pwned. Don't you think?

Moving to BSD is not going to really solve any of your problems despite the fact that I'm personally using it myself. Despite it coming out of a spook school (the license and the original code it forked from) the various projects are run by a lot of good people. I have no reason to suspect Theo at the moment but then again I haven't looked too deeply into his past. So who knows. Also even clueless people can contribute to things that aren't in everyone's best interest while being clueless about it.

At the end of the day Theo is a good coder though and I respect a lot of the decisions he's made over the years. Just like I respect a lot of people that work on NetBSD. They've done some good work. Rump kernels are very interesting for example and offering a alternative to what's been happening in mainstream Linux over the years is admirable. NetBSD in particular has done a lot of great work keeping older hardware alive and keeping things like their Linux emulator for binaries working. It's admirable.

But here is the thing. I'll use OpenBSD as an example here. I run it on my laptop (well one of them, but it's my main laptop for writing). In OpenBSD you get a solid kernel and a leader that cares about security and a community of skilled people working with him to provide something really great. The documentation is great and so is the code.

However, they have to support modern hardware. On my laptop that means supporting the graphics card. Which is a modern-ish AMD chipset. To do this they had to port over the drm drivers from Linux. Well, this driver is comprised of more code than the entire rest of the system: kernel, base system and everything including the 200 or so ports I run on that machine. All this code just to make the graphics work. They can only port over what they have to work with.

There lies the problem. We don't really have control over the hardware or the software. No matter who is in charge nothing will change that fact.

Portage is just a worse version of the BSD's ports at the end of the day.

Portage does provide some value by allowing you to configure compile time options in one place and using stuff like package.provided. It isn't totally useless. But the fact that dependency resolution is so slow due to python and the root shell is forced into being bash are huge issues no matter how much anyone tries to justify it due to legacy cruft.

The old ebuilds aren't even as useful as most people think. Since the current maintainers keep making unneeded (and frankly bad) changes to the EAPI. Which requires that you have to manually edit old ebuilds every few months-years anyway to keep up with the constantly changing requirements. Those of us maintaining local overlays have been forced to sink hours and hours into editing our ebuilds for no reason other than the current maintainers keep changing things for no good reason.

It wasn't that long ago (2020 or so) when half my packages on my system weren't simply ebuilds adding and modifying user accounts for no reason for example. Modern Gentoo is really getting out of control when it comes to things like that. There is really no good reason for it other than the current guy in charge not having any idea wtf he's doing (or more likely being actively malicious).

On my BSD boxes I've been easily able to maintain the same level of customization that USE flags on Gentoo give me without all these issues. Since the ports trees and build system were better designed in the first place and they do not rely on stuff like bash, python or an ever changing EAPI.

It's easy to write a package manager. You don't even need a package manager. If you know how to use make+make install you can run whatever you want on any UNIX-like kernel easily. That was how it was done for years by everyone before the 90s. You'd grab a tarball, run make then go on with your life.

Package managers were simply created to help manage dependencies. Since by the 90s a lot of software projects for POSIX systems started relying on more than a handful of other software in userspace. Or you'd have software A wanting version 1.2 of a dependency while project B wanted version 1.3. Users also started wanting to use more software than they did before since HDDs allowed you to store more data on a local system and people were using home computers instead of mainframes that were being managed by someone else.

Writing a package manager is like CS 101 stuff. Actually, it's probably considered below that level in a proper CS course. A code monkey can write a package manager in a weekend.

But there is really no point in doing it. If you're good you can make due fine with just tarballs+make. If you want a package manager then you're better off using an existing one because the real value in the package manager is the community around it. Since they can help you maintain a large collection of packages and this helps to spread the time/effort burden between all the users instead of everyone having to maintain them on their own. Why would you maintain a package manager just for yourself? You'd be better off just using tarballs+make because it'd be less of a pain once you'd initially setup the system.

I personally use pkgsrc. Since it has a large community behind it. Lets me build from source if I want but the community also provides binaries that I can trust if I want those.

>On Systemd

I don't care to go over all of this, but I'll link you to an old blog post about this from the guy that created musl: https://ewontfix.com/14/

There are far too many bugs and examples of bad design in the various IBM/Red Hat software that makes up lower userspace in Linux to cover. But I'll try to give you an over all example. I've mentioned a few already in prior posts.

PID1 is the most important thing running on any UNIX-like OS after the kernel itself. If PID1 crashes it takes the entire system down (it's as bad as kernel panic). All PID1 should be doing is reaping zombie processes and spawning new processes. PID1 should have as little code as possible in other words and it should be audited often.

IBM went the other way with PID1 (systemd). Instead of making it small, compact and easy to audit they placed millions of lines of code within it and had it spawn many unrelated processes that also run as root at all times (logind) which also comprise thousands-millions of lines of hard to audit code.

In other words they took full advantage of PID1 being the most important process running on your system. Then they used their influence and political maneuvering in various distros (Debian was their first big target) to push systemd hard and started making things rely on it with a hard dependency. In Gentoo (and various other "non-systemd" distros) they gave people controlled op versions of the same thing. You may be running OpenRC or runit for example but it's still spawning every other process from the systemd project as root (e.g. logind) which are required by projects widely used (e.g. KDE, Gnome).

Compare this to something like s6. Where the init itself is _very_ small and easy to audit and the processes it spawns for things like services are also very small and do not require running as root.

So with s6 (and other similar small init systems/service managers) you will have a more stable and by nature more secure system by default. Since it's both easy to audit and replace pieces of the entire service manager with whatever you desire. You will not end up with bugs like
>username 0day gives you root on any system
and
>A STOPPED JOB IS RUNNING!

It's an entirely different approach to software design. IBM's systemd wants to be a wrapper around the kernel and it wants everything else in userspace to depend directly upon it. s6 (and others) simply wants to do what an init is supposed to do: manage sub-processes, reap zombie processes and provide stability. Just compare how logging is done in both of them. In systemd you're forced to use binary log files and you can't swap them out for anything else. You can only re-direct the binary logs to txt logging. In s6 you can use any type of logging you want. In addition, with txt logs you can always get some useful information from logs if the system shits the bed for whatever reason. With IBM's logging you're forced to rely on their tool to read the binary log files and you can never be sure if corruption within them is real or just their own logger shitting the bed. corrupted txt logs still provide you with useful information because you can tell at a glance. With IBM's logging you can never be sure.

s6 can actually get logging up and running faster than any other logging system on UNIX OSs.

IBM also spread a lot of lies about how they've created useful features that are really part of the kernel or things we had before. For example, cgroups and "socket activation" are not unique to systemd. Cgroups are provided by the kernel and can be used with any init/service manager. Socket activation in systemd isn't even real. Same goes for starting up processes in parallel and many other things it claims it invented. Read the link above if you want to know more about that.

(cont.)

In short; IBM took advantage of the unique nature of PID1 to push vendor lock-in. Where s6 and just about all other init/service managers just wanted to give you a useful tool for start-up/shut-down/managing a running system.

Another thing about systemd/IBM is they gobbled up a bunch of existing projects on purpose (e.g. udev) to make it harder for non-IBM UNIX-like OSs to continue using existing software. Which is why we now have the situation we have with things like modern Gnome. Where porting it to say BSD or any non-systemd/logind distro becomes harder and harder with each passing month. Since we're not forced to patch around their garbage software or provide replacements for things most users didn't need/want. A good example of that would be projects like seatd, elogind, eudev and all the others. Where people were forced to fork or provide replacements for things from the IBM repo to continue using stuff like the two most popular DEs on UNIX-like OSs (KDE and Gnome). But it goes even further than that.

A lot of projects these days now claim they need things like dbus running or logind/udev to function. But if you compile them or fake that they're on the system with something like package.provided in Gentoo you'll quickly discover that most of them do not require them at all. For example, it's perfectly possible to use Firefox on systems that do not have dbus running. All it will do if you fake dbus is spam some error messages to console. Otherwise nothing else will break. This is why OSs like OpenBSD can still run Firefox perfectly fine. They simply fake that dbus is running (although you can run it if you want. User has that option in a sane OS/distro). I can give many other examples of this. Most every package claiming to need something from the IBM repo will work perfectly fine without it provided you compile it from source yourself. 9 times out of 10 it'll simply spam some error messages.

I've touched on this before but I'll re-state it. They push a lot of this crapware claiming it's more secure than what we were using for years before. Circa-2004 we had modern desktops running GTK/Gnome 2.x that do everything these modern systemd+wayland+dbus+whatever distros are doing today. They were perfectly fine for 99% of users since 99% of users own their own workstation/laptop and are the only person using it. They were also perfectly secure. Sure there are always bugs and CVEs happen. But in general things were fixed quickly and these distros in that era were certainly much better than say Windows XP as an example.

What IBM did was start pushing standards for the "Linux desktop" through their front organizations like FreeDesktop. They pushed a bunch of broken standards and would change requirements of them often. Forcing FOSS developers to constantly play catch-up. While they were doing that they also took over existing projects like X11 and GTK. Usually, kicking out the existing developers along the way and replacing them with their own. Then after they'd seized control they'd start to ruin them from the inside out, stop accepting PRs from outsiders/anonymous contributors and most importantly breaking portability with other UNIX-like OSs. This resulted in tons of needless work having to be done by everyone. Which wasted tons of time and effort. Then they'd regularly break existing projects whenever they could. Which is how we went from having something stable with GTK v2.x and getting roped into using GTK3.x+. Lots of people got so fed up they gave up on GTK all together. Which pushed them right back to using Qt which people were trying to avoid in the first place. Which is why we developed GTK in the first place.

Embrace, extend, and extinguish.

They also hired a bunch of formally "doing it for free" people through front companies. Spook front companies (all big tech companies are spook companies). So now you had a lot of people with a vested interest into going along with things they knew were bad because they were getting paid to do FOSS as a day-job. They threw tons of money behind this. All of it coming directly from spook companies like IBM, Microsoft, and all the others I've mentioned before.

Red Hat itself openly admits their biggest customer is the US Department of Defense (aka the military). They had an endless supply of money to throw around because all of that money was coming directly from the US tax payer. They had an interest in doing this because their goal was not only seizing control over all the software running all the servers and spook workstations but also so they could charge money to other companies through support contracts. They of course threw a lot of money at kernel hackers as well. Which is how Linus ended up making millions of dollars a year to sit on his ass and read emails. I'm sure you've noticed he's become far less vocal and eager to accept horrible PRs as of late. Especially after getting sent to re-education camp about a decade ago now (man time flies).

Whenever they ran into someone in a position of power they couldn't buy off they did the same thing. First they'd try to kick them out of their own projects through underhanded methods like what we've seen in the last decade with the push for Codes of Conduct enforcement in FOSS projects. If that failed they simply murdered them. Which is why people like Ian Murdock "anhero'd" despite screaming at the top of his lungs for weeks before he "anhero'd" that he was not suicidal, that the local police in his area was stalking him and detained him for no legal reason. He wasn't the only one. A lot of people "anhero'd" around that time but Ian is the most famous.

Man this is getting long, sorry.

You'll see a lot of the same lies parroted by the people behind projects like systemd. I don't want to cover them all in detail here this have been covered over 9,000 times before. But I'll touch on one that comes to mind.

They'll claim all the unrelated projects tied tightly to systemd are somehow "modular". When we all know it's a bold faced lie. They are tightly bound together and are impossible to use separately without doing a ton of hacking. Which is why we had projects like eudev and elogind in Gentoo a few years back which was eventually ditched all together after a few years with the lie that no one wanted to maintain them any longer. Really these type of forks were just projects to waste everyone's time. They didn't care if you used a forked version. They knew they'd be impossible to maintain long term and they would prevent people from writing real replacements.

Then you have them lying openly all of the time. They'll claim stuff like
>If you dislike the project so much you should fork it
>Where is your code? Why don't you contribute if you think the project is bad?
Knowing full well that the entire thing is a non-starter and it'd be useless to do so in the first place.

When they forced systemd into Debian for example they subverted the political process inside the project like they did with Gentoo. They tried to paint the debate as a false choice between upstart and systemd. Claiming the old ways to do init were somehow bad despite the fact they'd worked perfectly fine for decades. They claim shell scripts are "hacky" and unit files are better. When shell scripting had been widely used and is still used in UNIX-like OSs like all of the BSDs. They're fine, they're secure, they're not any slower than systemd's unit files and they are much easier to maintain.

I remember when systemd was first introduced, there was spam everywhere about how it made boot times so fast, and you just had to switch because your computer booted so much faster.

Yeah they pushed a lot lies like this. Boot time was one of the major ones. Of course boot times with systemd are actually slower than even sysvinit. They're slower than OpenRC, runit, s6, dinit and even busybox.

I didn't want to get too into the weeds but I guess I should have mentioned this in my original posts. But there is another key point about this whole boot time lie: How often is the average user rebooting their system anyway? Furthermore, do you care if it takes 2 seconds or 5 seconds?

Funny how they never talk about shut down times isn't it? Considering half of the time it can't even properly shut down :^).

The whole "socket activation" thing is one of my favorites. Where in reality it just spams bullshit while waiting on services to start up. So it really isn't doing anything it's claiming to do. If you read the usual blogs and posts from developers about the various issues with systemd you'll learn a lot. In addition to the one I've already linked I highly suggest reading this one:

https://judecnelson.blogspot.com/2014/09/systemd-biggest-fallacies.html

Do not expect anyone here to argue in good faith about systemd anymore. They're well aware of these break downs and blog posts but they're still parroting the same lies about it that they did back then.

https://judecnelson.blogspot.com/2014/09/systemd-biggest-fallacies.html

---
(cont.)

The real reason they wanted the old way gone is because it meant every last server running Linux was unique. Which meant they had to invest more time into cracking into them. With systemd and all distros being unified at PID1 you can write/use 0day exploits that work globally. Every last system connected to the internet is running the same scripts at the lower levels of userspace. Giving you easy access into any system you want.

An anon before said something about
>why would they do this if they have Intel ME backdoor?
Well here is why. First off al they really dislike using the hardware backdoors if they can avoid it. There would be a major outcry if it was used in a massive pwn and the public found out about it. They prefer to keep those hardware level backdoors for high value targets. Just like they really hate talking about the backdoor they have at the hardware level into everyone's cell phone due to the FCC mandated baseband chip. Instead it's much easier to pwn a system at PID1. You get everything you need and plausible deniability. You can also just say
>whoops we had a bug. Sorry about that here is the patch
Of course, there are 1,000 other bugs marked "WONTFIX" they can use for the next time they need to use one. Rinse and repeat. Can do it over and over again. If your hardware level backdoor is exposed in such a way suddenly you have major scandal and maybe you're on the hook for buying all the normalfags new hardware should they join together for a class action lawsuit or something.

But it's more than that. The DoD can order CPUs with those backdoors turned off/disabled. So can other countries I'm sure. Maybe you want to pwn a system like that. What do you do then? You use your 0day for the kernel or PID1.

That reminds me, SELinux:
It's NSA spookware and it provides you with no real security at all. Avoid at all costs. If you want to run a server and a hardened kernel you're better off using something like OpenBSD.

(cont.)

I'll give you another example of how they operate. Let's talk about zstd compression which came out of yet another spook front company facebook.

This was released years back as a replacement for the community developed xz compression algo. Over night it was everywhere. In the kernel and almost every distro (Arch, Debian, Gentoo and all the others) switched over to it as the new default over night. There were claims it was better than xz compression for various reasons. Most of them were not true. Mostly that it was faster and produced consistent results on multi-core systems. But xz compression was fine and it does the same thing now.

What do we see shortly after zstd became the new default everywhere? We get this massive propaganda push from every tech news outlet about a security bug in xz that never even made it into the repos of most Linux distros. Further more, IT DIDN'T EVEN WORK ON NON-SYSTEMD DISTROS.

But thanks to that fear mongering people now claim it's bad and should no longer be used for anything. It will certainly never be the default for any big distro ever again.

But then we go deeper; Where is zstd being widely used? Well we see it where you'd expect a compression algo to be used: real time compression of RAM, in the kernel and in projects like ZFS. Projects which are tied closely with encryption algos since people want to use encryption on their storage devices. So now we have yet another piece of spook software being used in all parts of the Linux software stack. If you don't want to use it too bad. You have to compile your entire system from source if you don't want to use the zstd default and this is not only a major pain in the ass it also means you can no longer rely on the rest of the community around whatever distro you use to help you maintain your own system. Now you're at the mercy of spookware for lots of things. Like verifying the contents of the binary you want to install from your package manager.

One final note about xz compression and it being replaced by zstd. The person that submitted those PRs for xz came out of nowhere and he's a ghost on paper. So this also destroyed trust in anonymous maintainers and the code they submit to FOSS projects.

What are we seeing lately in the FOSS word? We're seeing major projects no longer accepting code from a submitter without them handing over their full dox. It destroyed public trust in people that want to contribute without handing over their full dox. You can no longer send patches to most projects like the kernel or things in userspace without everyone knowing your real name. Long gone are the days where the code itself was all that mattered. Now you have to be willing to put your real name out there. This is a huge change from how things were being done for decades in the FOSS world. Where the only thing that mattered was the code you sent in and you could build up a reputation by sending in good patches from an anonymous email address using a nick name. In other words; it destroyed trust within the community.

Then we have the whole development for most projects moving over to things like github where your dox is required if you want to be taken seriously. Even FOSS projects maintaining their own git instances became highly restrictive in this respect. You also can't use mailing lists anymore without an email address from an "approved" email provider. Long gone are the days when you could submit work from a throw-away email address. Or use multiple screen names/emails for different projects you worked on anonymously.

Everything needs your full dox now. What do we see on places like LKML happening at the same time? Long time contributors being banned from sending patches to the project due to their country of origin or because they said something on the mailing list 20+ years ago that now violates all these stupid "codes of conduct".

I could go on but I think you get the point.

>More on the BSDs

With OpenBSD it mainly boils down to one of these things:
>Security issue
>Developers working on the project do not own the hardware and therefor are unable to test it themselves
For this one if you donate hardware they're usually happy to port things over. Even more so if you're willing to help with porting the driver yourself.

The final reason something might be rejected from OpenBSD is typically the following. If something is unable to be included within the base system because it would somehow break support for existing platforms being supported it's a no-go.

Last but not least, licensing issues. But this driver seems to be dual licensed already so right now it seems like it falls into the
>no developer owns the hardware and has interest

See this discussion about it on the mailing list(s): https://marc.info/?l=openbsd-misc&m=163734223830612&w=2

If you donate some hardware (and maybe some money) perhaps you could get a developer to port it for you. Better yet. Do it yourself and submit it to the mailing list. People are more than willing to help new developers and get them up to speed.

Another important note about OpenBSD with wifi drivers in particular; In OpenBSD wifi drivers are actually sane and work through ifconfig like everything else. We do not use stuff like wpa_supplicant at all. You can easily connect to wireless networks using ifconfig like all UNIX systems did before all this modern crap like wpa_supplicant came along.

FreeBSD can use all Linux drivers for hardware through the Linux bin emulator and they chose to port and rely on wpa_supplicant (along with a bunch of other Linux garbage). Therefor, they have support for more hardware but the over all system is less coherently designed (and less secure).

(cont.)

I should have also mentioned that code correctness is very important to the OpenBSD project. Which is why they're willing to remove code all of the time they feel hurts security or isn't being properly maintained. Hence why it does not support things like bluetooth, wine, linux binary emulation and a couple of other things. Although for the last one you can always run your linux-only stuff through vmm(4) (VM/Hypervisor).

The project is not opposed to bringing back support for bluetooth, wine, the linux emulator or anything else they might have removed over the years. However, someone must be willing to step up to maintain them, properly document them and they need to be secure to the level of the rest of the project. In addition you must be willing to pass the usual audits that happen frequently and stay on top of any reported security issues that may arise.

Again in general the OpenBSD developers are more than happy to port over drivers. Stuff gets ported over all of the time. Each new release gets support (or improved support) for lots of new hardware. I'm running it on a 5 year old thinkpad right now and support for everything is great. The gfx card is supported as well as it is in Linux and the wifi card in it just got some improvements in the last release.

Also OpenBSD -current is so stable that I've been running it for years now. I don't bother with point releases. If you run -current you get support for new drivers much faster instead of having to wait 6 months. But you need to read the mailing lists because things may break rarely if they need further testing on similar hardware the developer might not have had on hand (minor revisions in hardware all support by one driver may cause minor issues).

I typically update to new -current snapshots weekly. I've rarely had an issue with it and whenever I did they were usually fixed within hours. Of course you need to be proactive and report such things to the mailing list. We're friendly
---

>Rust is a tactic to compromise software even more
Rust is a big part of this, yes. The push for "re-write everything in Rust for security" is coming directly from the US military and spooks. I don't wish to re-hash all the reasons or go into how the Rust Foundation is being funded. I encourage you to go look for yourself to see who is sitting on the board of that organization and who's funding it.

Mixing languages in the kernel is a very bad idea. Which is why C++ was denied in the kernel for so many years. Rust in particular is very bad because it will break support for many older platforms and even 32-bit x86. Mixing C and Rust in the kernel is going to lead to all sorts of bugs and problems. It will also put the ability to compile the kernel on your own machine out of reach of many people soon. Especially as new hardware comes along that will require Rust drivers to function. In addition, a lot of that stuff will probably not be licensed under GPL, BSD and other more friendly licenses that have allowed us little folks to be able to write and use source code on our own machine like we have since the 1980s. They're closing the x86 loophole and putting the genie back in the bottle.

The age of us common folks having access to hardware like we've had since the 1970s or so is coming to an end. The primary reason the Linux kernel is being developed these days is not so people can run it on their own machines. It's so these large data centers they're currently building can run it on the servers that will be used to power the security state we're currently seeing built around us. The future for end users are dumb terminals talking to hardware in "the cloud" that will be beyond our reach.

Another big issue with Rust in the kernel and if it starts to be required for things other than drivers and optional stuff you can exclude at compile time is the following; Let's assume you want your distro to support 32-bit x86 systems and let's say something like Alpha arch. Well. The Rust compiler can't build natively on those systems. So to run that code on your machine you're not unable to build it locally on the machine itself. You either have to have a local machine that can cross-compile the Rust parts of the kernel or you have to rent a build server from someone else. So now you not only have to trust all these new third parties for the compiler you also have to trust a third party somewhere on the internet for the actual hardware.

The Rust community always claims non-AMD64 archs are dead and to "get with the times". Supporting things like an Intel P3 processor, an old m68k machine, an Alpha CPU all things of that nature are not a priority for them. A lot of them aren't even supported with cross-compilation and probably never will be. So a lot of machines that can currently run the Linux kernel are left out in the cold. They're now obsolete for no reason.

I am very skeptical of the claims Rust makes about being safer than C when it comes to things like managing memory. I do not feel the trade off of very increased compile times are worth it. Plus their ecosystem still relies heavily on C as well. I think the proper solution to this problem would be teaching people how to properly use C safely to avoid the common pit falls.

History has proven that you can't avoid bugs of that nature by trying to deal with them using the compiler. You're just kicking the can down the road and I'm sure in 10-20 years time there will be a whole host of issues, bugs and security problems even in things written purely in Rust.

Mixing C and Rust is just a recipe for disaster. It gives these people a false sense of security. Mixing them together like that will end up causing far more issues than simply trying to write code in C that is good.

I am not a fan of the community surrounding it.

(cont.)

Rust aside there have been several things being allowed into the kernel over the last 5 or so years that are very bad. A lot of those patches coming directly from places like the NSA, Microsoft, Amazon, Facebook, IBM and the other spook fronts I've mentioned earlier ITT. For example, Linus has already bent the knee on issues like DRM (digital rights management).

Support for things like TPM are being baked in now which will be used to ID each individual machine and the users using those machines. I fear soon you will not be able to access the internet at all and will be blocked at the ISP level if you are not verified. The verification will be tied to your digital ID. Lots of laws have been passed recently and everyone is being opted in to these digital IDs through drivers licenses in my country. In two years I will no longer be able to legally drive

for example if I don't renew my driver's license. When I renew it I'll have a federal digital ID instead of the state issued ID I've had all of my life. I'll be locked out of being able to use a bank account if I don't go along with this as well and denied access to basic services I need to live.

We're seeing this happening globally in every country. From USA to EU to UK to Japan to China they're all bringing in these systems in lock step. Most of which are all running on top of the Linux kernel.

As I mentioned before tons of people just got blacklisted and are no longer allowed to submit code to the kernel. They used the war in Russia as an excuse. Linus bent the knee because he lives in America and the only way he can continue making money from the kernel is to go along with what the DoD/Military/Government demands. It's now illegal for anyone that isn't doxed to submit code.

I am surprised a hard fork of the Linux kernel hasn't happened already. But it seems there is no one out there with enough balls to do it. Or more likely no one with enough resources to stand against what's happening.

Let me give you an example of what I mean about the latest kernels (since late 2024) being strange since those maintainers were removed from being able to contribute.

It was big news at the time then it seems people just forgot and the usual tech news sites stopped talking about this almost over night.

There was a company called Baikhal Electronics that was producing hardware MIPS and ARM hardware back around 2020. It was the only company that was producing CPUs outside of the US controlled/dominated companies (Intel, AMD, Motorola). Like these companies they were designing and producing the hardware out of the fabrication plants in Taiwan. Under US state sanctions they were denied access to using these fabs and the company was forced to file for bankruptcy in 2021. Then in 2024 when a list of people were banned from contributing to the kernel every last former employee of this company just happened to be on the list. A list that was compiled by US state department lawyers and sent to Linus and friends. Who bent the knee almost instantly.

Some discussion about it on the LKML: https://lore.kernel.org/all/3ace1329d4ef99b87780d0ef07db179d27d04d44.camel@oldum.net/

These people were all removed from contributing on October 18th 2024. Do you know what happened on the 17th of that same month? All versions of the Linux kernel from the version 5.xx and 6.xx series were re-built with massive change logs. I'll leave it up to the reader to decide what to make of this.

I doesn't take a rocket scientist to see what happened here.

We have telemetry and all manner of shady things baked directly into the kernel now. Every month it gets a little worse. It's millions of lines of code.

Have you read through all of it and audited it yourself lately? I know for a fact that I haven't. It's a massive undertaking after all.

Are you okay with the US state department having full control over who is and who isn't allowed to contribute to the kernel? When for decades anyone anonymous could submit any code they wanted and all that mattered was is the code worthy of inclusion.

In the 90s anyone with a pseudonym and an email address could send a patch, have it reviewed and have it included for everyone's benefit as long as Linus signed off on the diff.

We used to take a stand against this sort of thing. People would print encryption algorithms as books and send them in the mail to Canada (and elsewhere) so others could use the software. People would proudly wear DeCSS t-shirts to cons and in public. People played spot the spook at DefCon.

Now DefCon is a spook convention.

>Can we trust Valve to save Linux?
You can trust Valve to push more aggressive DRM and do everything they can to ruin everything they touch. You really shouldn't be using these services. But I understand why people do since they want to play video games with their friends online. I don't allow steam on most of my systems. I have a dedicated gaming PC for it but I rarely play anything on it these days. The games I do play are pretty old now (2016 was when the newest one I play regularly was released). I do not have a large Steam library. I pirate and I encourage other people to do the same. I mostly play older FPS games (mainly Q3A, UT and L4D), fighting games and SHUMPs. The fighting games I do play aren't any of these modern ones (they ruined all three of my favorite series these last few years). I prefer to play on dedicated hardware whenever possible (like my Neo-Geo MVS). My friends and I prefer to play together IRL on the couch.

You get the point. I used to be big into playing console games but I always mod chipped or softmodded them.

As far as this scheduler is concerned I don't see this as anything special. But I'll withhold judgement until I try it. I already run a low latency kernel for doing audio stuff (I make music as a hobby). I don't have any issue with "stuttering" on my machines. But then again I don't run one of these "gaming distros" that went all-in on wayland and every other recent gimmick designed to separate money from dumb fags that don't know any better.

Proton/wine has a lot of issues with various games I play. I still maintain a PC-98 and PCs running Windows 2k/XP on bare metal because wine simply lacks support for a lot of games I like from that era (or their mods). It's just less of a pain to keep those machines going than it is to fuck with wine all of the time and deal with setting up controllers/joysticks on Linux (it's really horrible once you need support for 2-4 at the same time).

Finally, why don't we look at some changes that happened in Gentoo for the last 5 years? Where we can clearly see the agenda they're pushing through their own news updates. To save time we'll only go back to about 2020.

>January 2020
>Stable alpha keywords removed
https://www.gentoo.org/support/news-items/2020-01-23-stable-alpha-keywords-removed.html

Removed stable support for Alpha because they're lazy. You'll see this becoming a trend as we go forward.

>April 2020
>Deprecation of legacy X11 input drivers
https://www.gentoo.org/support/news-items/2020-04-03-deprecation-of-legacy-x11-input-drivers.html

Forced all users to use libinput by default. Claims "upstream" is no longer supporting things that worked for 35+ years. This change forced you to install udev on systems if you wanted working input devices.

>Desktop profile switching USE default to elogind
https://www.gentoo.org/support/news-items/2020-04-14-elogind-default.html

A week later they force (e)logind as the new default and drop even the option to use ConsoleKit2. Claims "upstream" no longer maintains consolekit. Despite it still being maintained by the BSDs and several other linux distros. Again forcing users into using a systemd tool with no warning.

>June 2020
>xorg-server dropping default suid
https://www.gentoo.org/support/news-items/2020-06-24-xorg-server-dropping-default-suid.html

Removed the option to use Xorg without logind. Claims suid is a "major security risk". The solution? Run a systemd tool as root. So much for choice.

>January 2021
>LibreSSL support discontinued
https://www.gentoo.org/support/news-items/2021-01-05-libressl-support-discontinued.html

Removed users ability to run LibreSSL. Claims upstream is not maintaining the project. Removes it entirely from the repos.

(cont.)
>January 2021
>New OpenRC Display Manager Initializer Scripts
https://www.gentoo.org/support/news-items/2021-01-30-display-manager-init.html

Suddenly we have all the developer resources in the world to maintain something ourselves when it allows us to support IBM/Red Hat cancer. I don't really care about this but it's worth noting that they're happy to give you a "choice" when it might encourage you to switch to whatever Red Hat is pushing this week.

>May 2021
>x86 support to be dropped from media-gfx/darktable
https://www.gentoo.org/support/news-items/2021-05-14-darktable-drop-x86.html

Support for a package running on x86 32-bit CPUs totally dropped because "upstream no maintain". Keeping the old working version in the repo for users on this arch? Nah, too much trouble. Wouldn't be "secure".

>July 2021
>systemd-tmpfiles replaces deprecated opentmpfiles
https://www.gentoo.org/support/news-items/2021-07-15-opentmpfiles-deprecation.html

Users on OpenRC forced into using yet another systemd project because the default is changed for "security" reasons. Even though they openly admit it isn't an issue for 99% of users. Same bug exists in the systemd-tmpfiles package. But I

guess they don't care when it's coming from IBM instead of the FOSS community.

>August 2021
>eudev retirement on 2022-01-01
https://www.gentoo.org/support/news-items/2021-08-24-eudev-retirement.html

eudev is dropped for systemd's udev. Claim no one wants to maintain it anymore. Despite multiple other distros maintaining and using it to this day. Starting to see a pattern here...

>September 2021
>busybox removal from system set
https://www.gentoo.org/support/news-items/2021-09-24-busybox-removal-from-system-set.html

Busybox removed from @system set. Just in case you might have got the bright idea to run it instead of systemd. Breaks tons of systems in the wild that had been working fine for years without configuration changes.

(cont.)

>April 2022
>Migration to sys-apps/systemd-utils
https://www.gentoo.org/support/news-items/2022-04-19-systemd-utils.html

All packages that are split out of systemd's repo (udev, systemd-tmpfiles) are bundled into one massive package. Breaking all existing configurations for users not running systemd profiles and making it much harder to use the ::gentoo repos for anyone that had replaced them with the non-systemd (and forked) packaged that were removed from the repo the year before. Why bundle so many unrelated packages like this? There can only be one reason and it isn't "because it makes it easier on the devs".

>July 2022
>PipeWire sound server migration
https://www.gentoo.org/support/news-items/2022-07-29-pipewire-sound-server.html

Pipewire forced by default. Breaks all existing configurations for users that had been using JACK and/or plain ALSA before. They could have easily just added a new desktop profile or did this much cleaner without breaking existing configs. But this is modern Gentoo and no one knows wtf they're doing.

>December 2022
>/usr merge for systemd users
https://www.gentoo.org/support/news-items/2022-07-29-pipewire-sound-server.html

/usr merge forced on to those dumb enough to be using systemd profiles. OpenRC soon to follow I'm sure. So much for "choice". But we're lazy and don't want to do the minimal work to continue allowing this for systemd users.

>May 2023
>Plasma Profile to enable PipeWire, Wayland support
https://www.gentoo.org/support/news-items/2023-05-20-kde-pipewire-wayland.html

Wayland+pipewire forced as new default for KDE profiles. Breaking existing configurations.

>January 2024
>Separate /usr now requires an initramfs
https://www.gentoo.org/support/news-items/2024-01-05-usr-initramfs.html

Initramfs now required to use /usr on separate partitions. Breaking support for systems that had worked fine on all Linux distros going back to 1992. So much for "security" and "choice".

(cont.)
>January 2024
>Merging of installkernel-gentoo and installkernel-systemd
https://www.gentoo.org/support/news-items/2024-01-18-installkernel-merge.html

Merged two tools for no reason other than to make your life a living hell. This and several of the changes mentioned above made it impossible to block all systemd packages with a simple USE flag or through package masking. Broke all existing configurations. This caused them to issue 3 more news items related to GRUB and installkernel because it broke so much shit. I won't bother linking to them as they're all related. Check the archive if you care to read up on what all broke.

>May 2024
>media-video/wireplumber-0.5.2 may break on upgrade
https://www.gentoo.org/support/news-items/2024-05-06-wireplumber-0_5-bump.html

They were forced to issue this news item because of forcing the pipewire default which broke everything as expected just a short time later.

>Changes to dracut kernel module/microcode handling
https://www.gentoo.org/support/news-items/2024-05-17-dracut-ext-kmods.html

They were forced to issue this news item because they stopped maintaining Genkernel and forced dracut (IBM package) default when they switched to providing -bin kernels as default and stopped supporting automated building of custom kernels in the default profiles. This is the kind of high quality engineering you get when your "upstream" is Fedora.

>August 2024
>Gentoo drops IA-64 support
https://www.gentoo.org/support/news-items/2024-08-07-removal-ia64.html

Yet another non-x86-64 arch is dropped despite being maintained for decades with no issues. Thanks new defaults!

>September 2024
>Haskell destabilization
https://www.gentoo.org/support/news-items/2024-09-29-haskell-destabilization.html

We can no longer maintain haskell package support for...reasons lol.

(cont.)

>September 2025
>Stable sparc keywords removed
https://www.gentoo.org/support/news-items/2025-09-01-stable-sparc-keywords-removed.html

Another arch bites the dust. "We don't have enough man power". Meanwhile, must smaller projects with less people (NetBSD/OpenBSD) have no issues maintaining AND building natively on these archs. Rather strange they're having to drop support for so much right after changing all those defaults isn't it?

>Stable hppa keywords removed
https://www.gentoo.org/support/news-items/2025-09-01-stable-hppa-keywords-removed.html

Yet another arch bites the dust. Not enough man power to bump a few packages now and again or test them. Too busy tracking down code of conduct violators on the IRC channel I guess.

>sys-apps/openrc user services introduction
https://www.gentoo.org/support/news-items/2025-09-04-openrc-user-services.html

Forcing you to run a bunch of systemd trash on your OpenRC profile wasn't good enough. So we pushed out a bunch of untested shit in a new OpenRC version maintained by the same guy that's been pushing the /usr merge on the mailing lists for the last 3 years. We all know what's coming next.

>encfs is unmaintained
https://www.gentoo.org/support/news-items/2025-09-16-encfs-is-unmaintained.html

We care a lot about your security. So we've taken away your choice to run something that's worked fine for years again. Same pattern we've seen with a bunch of distros over the past few years. You don't have a "choice" you must migrate and we won't simply change the default and leave the old package alone in the repo. All unmaintained software must be PURGED!

(cont.)
>October 2025
>Cache-enabled sync mirrors only for official repos
https://www.gentoo.org/support/news-items/2025-10-07-cache-enabled-mirrors-removal.html

Maintaining a cache for mirrors and repos we don't like is such a burden guys and we simply don't have enough money despite getting massive donations every year.

Sorry it'll make it harder to sync your system if you've been following all those repos that was hosting and maintaining all that software we removed. We assure you we didn't do this simply to make your life harder. :^)

>January 2026
>Desktop Profile to enable PipeWire support
https://www.gentoo.org/support/news-items/2026-01-15-desktop-profile-pipewire.html

We had to force a pipewire default in ALL desktop profile due to
>long standing complaints
from mystery users that somehow claim audio hasn't been working for them on the desktop profiles for the last 20+ years. Who are these people? No idea. Where did they complain? No idea. But we're going to make your life more difficult and break your existing configuration anyway if you're using ALSA directly, JACK or decided to stay on pulseaudio. We gotta set these defaults as closely to Fedora's after all.

Not these are just news items and the tip of the ice berg. I haven't even gotten into the changes that have been happening since they started offering binaries. Like the new -bin kernel they set as default. Which is using a config straight from Fedora.

Note I don't mind the binaries being an option. But they're changing tons of shit under the hood to conform to the defaults they've set within them. Which are all based on Fedora's of course (hope you like systemd bro).

There have also been many sneaky changes within many key packages themselves that have made it impossible to exclude certain software globally on your system with a one-line make.conf change. systemd being the prime offender.

In short: We can see a clear pattern with news items alone. But this extends into all aspects of the project as this same thing has been happening in individual packages and it has ramped up hard in the past decade or so.

>stop maintaining projects written/forked by Gentoo itself
>replace them with IBM (mostly systemd/freedesktop) packages most users wanted to avoid in the first place (hence why there were so many forks like eudev)
>purge older working software from the official repos. Blame it on "security issues"
>Blame it on "upstream" and "lack of man power"
>Do it in the most annoying way possible which causes users that aren't using the defaults to constantly change their local config files and maintain their own local overlays
>Make it harder for these users to share their custom overlays with the community through various excuses like "no money to mirror them for you sorry"
>Ban anyone that complains or even asks about it using the Code of Conduct and the "Community Relations" team. Where all discussion, votes to ban and the entire process is hidden due to "privacy concerns"
>If anyone points out what is happening label them schizo/crazy/hater/luddite
>Flood all third party places where people discuss Gentoo with your jannys where you shut down all discussion (see: /r/linux, /r/gentoo, HN, and of course /g/)

You'll see the excuse of
>lack of man power
Used over and over again. But if you take the time to dig into the council, community relations team and the forums you'll notice that since 2020 they've added more users than ever. Even the Gentoo sub-reddit is about twice as active as it was 5 years ago. Yet they keep complaining they don't have enough people to maintain packages/ebuilds (which are easy as fuck to learn and maintain) or to maintain the various projects they've ditched like Genkernel.

If people were able to maintain this stuff for the past 15 years during the "dark ages" (2005-2018 or so) when many abandoned Gentoo and went elsewhere due to the snails pace of software entering stable and unstable branches how come they can't get newbies interested now? Where did all those long time users go?

Well it's obvious: They banned most everyone that was doing the actual work. People were trying to contribute and help. Their changes were not being accepted though because it didn't fit with the council's new agenda for the project. Which is sticking as close to Fedora (IBM/Red Hat) as possible. williamh refused to merge in a bug fix that was listed as a "security exploit" in one of the above news item. Yet he pushed untested changes to everyone's machine multiple times over the past 6 years or so. Then you go to the mailing list and you see he admits that HE ONLY OWNS ONE PC and can't properly test things because he'd rick b0rking it. Guess he's too dumb to figure out how VMs work and unwilling to fish machines to test on out of the local dumpster. He's too busy pushing for the /usr merge to deal with reviewing a 3 line .diff to patch that security hole.

They claim to have no time. Then squat on key packages. They claim no one wants to help yet they refuse to leave the council and changed the rules so they couldn't even be voted off of it.

>williamh refused to merge in a bug fix that was listed as a "security exploit" in one of the above news item. Yet he pushed untested changes to everyone's machine multiple times over the past 6 years or so. Then you go to the mailing list and you see he admits that HE ONLY OWNS ONE PC and can't properly test things because he'd rick b0rking it.
Pic related.

This faggot squats on the OpenRC package. Broke users systems several times with untested changes pushed directly to the stable repo. Made multiple horrible changes to the point where lots of users are still running older versions from before he gained control over the project. Yet he doesn't own a second computer to do testing on, doesn't know how to use VMs and refuses to merge patches other people have sent in. Then he sits on a security patch for 2 years and uses it as justification to deprecate the package all together and replace it with something from the systemd repo.

Then you go to the dev mailing lists and he's doing shit like pic related where he's advocating for merged /usr that will break tons of systems in the wild. Systems being run by people that want to do various things that a split /usr directory allow like hosting it on a server on their LAN or allowing you to recover a broken system without having an initramfs. One could argue (I do) that a split /usr is also much more secure when configured correctly and being used in conjunction with a static /dev and kernel.

But I guess williamh wouldn't know about those kind of things because he's stuck developing on one laptop and has probably never used a system that isn't x86-64 or has more than one storage device.

The person maintaining the init for Gentoo doesn't even have his own home server. Let that sink in. These are the kind of "experts" dictating to you how you should be using your own computers.

---

link to mailing list thread. You can find it here: https://archives.gentoo.org/gentoo-dev/570312c8.1469ca0a.30985.5db1@mx.google.com/

If I get some time I might pull up some more stuff like this from the mailing list. Since this group that took over and subverted the project talk pretty openly on it about how they plan to ruin it and take away user choice.

There is a massive thread on there from when they removed an entire forum because they were angry about the old timers teaching new users how to continue running the the distro without using all this cancer they started pushing as defaults. Using "politics" and "the mods refuse to enforce our new Code of Conduct" as their main excuse. In that thread there is an employee of Google/Alphabet openly bragging about how he "owned the Trump supporters" and why that was a great reason to stop allowing freedom of speech on a forum that had existed for over 20 years without issue.

In general though: All these "web 3.0" and "crypto coin censorship free network!" crap that's came out over the past few years are all garbage. IPFS was something I had a lot of hope for. But then the developers sold out and long standing bugs haven't been fixed in 6+ years now. They're all too busy working with big tech companies behind the scenes that wanted to use the protocol. Both of the implementations are horrible and it has managed to murder any router I've placed it behind. Then they tried to tie a crypto-coin to it as well. Shame.

We don't need any new type of censorship resistant protocol. We already had them built and finished in the 1970s and 1980s. It was called email, BBSs, fidonet, and NNTP (usenet). Store-and-forward protocols where once something was out in the wild it couldn't be taken down because it was being hosted on so many systems. At most all we need to do is bring some of these up to date to support sending more data at one time. That way we wouldn't have to break large files into smaller archives like the warez groups have to do on usenet. We can use encryption over these protocols already. For years e-mail could be sent totally anonymously because there were tons of anonymous e-mail servers that would route any message sent to them without verification. The only reason we don't have this stuff anymore is because the big tech companies like google, microsoft and others took over the entire protocol and blacklisted anything outside of their own servers. Well that and the usenet jannys deciding to no longer allow truly anonymous users on to the news servers. We didn't have jannys censoring everything on these networks either. The tech is old, it works and we could be using it right now. It'd be easy to add any quality of life features we wanted.

If you see anyone using the usual buzzwords like schizo and meds you can be sure they're actively malicious or fell for the propaganda. They never engage or argue in good faith. They can't. As laid out in this helpful guide to spot them in the wild.
>>4524

They really hate any thread where some actual people are having a friendly discussion. We don't have to agree to have a friendly discussion either. I love debate and I try to be nice even to those I disagree with. One of the things they attempt to do as stated in the above guide is to drive into an emotional rage in an attempt to make the person posting the information look deranged and angry. If they can drive you to anger they hope it'll make you look bad and cause you to reveal information about yourself.

>It was killed off in the late 2000s by the US Congress. Who demanded all ISPs stop giving users free access because of fear mongering about pizza being on it. I encourage you to look into that if you get the time. Again, impossible to really censor or delete anything once it's on the network. The network has also been centralized in recent years by a handful of companies and if you go looking into them you'll find the same people behind the scenes: spooks everywhere.

Do you have any sources and what bills that were introduced/passed and signed by the president at the time related to this and usenet? i'm interested in looking into this particular statement

---

>>4536
found it

https://techliberation.com/2008/06/11/new-york-ag-pressures-isps-to-cut-off-usenet-access
https://www.wired.com/2008/06/verizon-time-warner-cable-and-sprint-to-block-usenet/
https://www.eff.org/fr/deeplinks/2008/07/more-isps-decide-filter-usenet-newsgroups?language=fr
https://techcrunch.com/2008/07/12/more-ny-isps-agree-to-cut-off-usenet-access-in-response-to-pressure-from-attorney-general/
https://arstechnica.com/uncategorized/2008/09/lights-out-for-usenet-access-through-comcast/

https://en.wikipedia.org/wiki/Communications_Decency_Act
https://en.wikipedia.org/wiki/Child_Online_Protection_Act

close enough

---

walls and walls and walls of text
more text doesn't mean you have more of a point
you really don't know how to make a concise point, do you?



1673619198340299.jpg

It does read like a bit like schizo babbling but I get where you're coming from and I mostly agree, thanks for the efforposting anon. I'm not knowledgeable enough to participate in such a complex discussion so all I can do is ask you a few questions, mostly out of curiosity.

What do you think is the less spooky Linux OS? (if that's even possible in your opinion)

Opinions on the Hyperbola Linux (soon(tm) to be BSD) project?

What would you recommend or suggest to an end user to do? stay on Linux and pray?

Attachments:



what the fuck.jpg

>>4548
>Opinions on the Hyperbola Linux (soon(tm) to be BSD) project?
I have not used that particular distro. But running to another distro is not going to solve this problem. You have to get out of that way of thinking.

>What do you think is the less spooky Linux OS? (if that's even possible in your opinion)
Direct answer: The one you make yourself. Everything below also should be taken into consideration.
>What would you recommend or suggest to an end user to do? stay on Linux and pray?

We lost the kernel somewhere between the LTS 4.xx releases and 6.xx. Linus took the dirty money and screwed us all over. Similarly, Richard Stallman was always a spook working out of a spook school (like most every famous American school with an old school hacker culture).

The hacker ethics are no longer being followed and they haven't been followed in a long time. This war was lost before most of us were born. Somewhere in the mid-late 1980s. Things just didn't get intolerable until the 2010s because most of us were naive about these matters when most of us were young and dumb in the 90s-2000s. A lot of us bought into the lies Stallman promised about the GPL or fell into the BSD camp. They're both just controlled opposition and being funded by the same people pushing this garbage spyware.

The age of regular people having access to a real computer is coming to an end. The x86 loophole we've been exploiting since the late 80s is going to be closed soon. The future is wearable devices and devices under the skin talking to the botnet 24/7. It's why they're spending billions right now to build out all these data centers.

For what it's worth: systemd is just the start of your troubles. You'll want to avoid dbus, policykit, udev, (e)logind, the built-in spyware in modern gnome/kde along with a bunch of other shit that makes up the modern "Linux distros". Yes even the "non-systemd" ones. Replacing PID1 with something that isn't systemd (e.g. runit/OpenRC) then pulling in half of the systemd repo (logind, udev etc.) is just as bad as running Fedora or modern Debian.

You will get nothing good from the Devuan people. They have been funded by shady interests and have done everything possible to stifle any progress for almost a decade now.

Oh and before anyone posts the usual lie again. Yes you can run a GUI (even a "modern" one based on gnome/KDE) without all this shit. I'm not talking about doing everything from console like a retard.

At worse if you rip all that shit I mentioned out (among other things) you'll just get some harmless errors getting thrown to console about this shitty software crying about not being able to talk to dbus. It'll still work just fine.

You will of course want to avoid things like anything that requires Rust compiler as well of course.

Linux is not "open source". BSD isn't "open source". It's all spookware. We should have never wasted so much time giving away code to these projects for free. I'm personally more into other more obscure OSs now. But good luck getting anyone else to use them. We really need a new kernel first then we can port over support for POSIX/NT/Win32/DOS shit later.

IF you do want to stick with Linux though it's still possible to build something from source that works and gives you a GUI without all this crap. But you'll find no one shipping a distro/binaries for that anymore.

You want an actual good service manager on Linux? s6. But for home use you probably don't need it. Since any collection of rc scripts is better than systemd. Anyone with half a brain is not running systemd on their systems because it only serves as a backdoor into POSIX systems. As we've proven over and over again but shills and bots will pretend none of it is an issue. You're a "schizo" if you notice.

My servers are all on Net/OpenBSD now because I got sick of dealing with it. My workstations are a split of Free/OpenBSD and s6+Gentoo. I keep flirting with the idea of building a new Linux distro based around s6+pkgsrc but every time I've tried to get anyone interested in helping the threads are either instantly slid or flooded with bots. Then I got a knock on the door. Faggots even follow me around to small boards with like 10 total users on backwater alt-chans. It's very obvious it's the same people every time because they're always angry as fuck and give themselves away.

There are some blogs and articles that cover most of this stuff. The gentoo forums used to have very active threads about this type of thing. That is before google/ibm got control over the political side of the project and banned everyone for posting wrong-think. Maybe you can find those old threads in the archives somewhere. There was a forum all the old timers fucked off to a few years ago but surprise surprise it got ddos'd off the web in short order.

The problem is much deeper than PID1. It goes all the way down to glibc now I'm afraid. The entire stack can no longer be trusted.
Enjoy these last few years before things get really bad. That's all the advice I have.

This sadly extends out to FreeBSD and anything based on it. Since they seem more concerned with importing stuff from Linux than doing anything original anymore.

Really, UNIX/POSIX should have died off in the 1980s. But every time someone tried to introduce something new for consumer use it was killed off in the womb through politics. The Japanese wanted to introduce TRON and worked really hard on it during the early-mid 80s. They were even ready to introduce it to their public schools to train the next generation of Japanese programmers and users on it. Laws were passed in Japan that would have removed DOS and UNIX based

software from the public school system desktops and replaced them with TRON. But Microsoft, IBM and all the usual suspects lobbied the US Congress to put a stop to it. In the late 80s the USA Government passed a trade agreement and bullied the Japanese into stopping introduction of TRON to the public Japanese schooling systems which effectively killed off the project. They claim there was push back from Japanese teachers that had already written software for DOS but if you dig into Japanese sources (you need to be able to read nip since none of that was translated) you'll discover there was tons of support for a Japanese built OS as you'd expect from their country.

TRON was not just to be limited to desktops. The kernel was supposed to run everything in the country from trains, factories, desktops, TVs/home devices and even handheld devices like cell phones. The entire country was supposed to be integrated under one kernel running everywhere where all of them could communicate with each other over TRON's networking standard.

The US Government killing TRON is what led directly to the Japan lost decades of the 90s-2000s. It killed their economy and stagnated all Japanese growth/business for multiple decades.

This actually happened again during the 2000s. But concerning TRON (or BTRON the desktop variant). One Microsoft and IBM got their way the plan to deploy BTRON desktops in the schools was stopped and replaced with DOS/Windows based desktops. Which is how we ended up with the Japanese version of DOS/Windows computers in Japan in the 80s-90s. The PC-98 standard computers (and clones) that came out of NEC. NEC was more than happy to go along with the anti-TRON side of things because they were basically already owned by the western tech companies of the time. They didn't want to lose their monopoly over domestic desktop computers.

Well later in the 2000s Japan had a large cell phone market consisting of multiple companies all with different hardware, software and networks. It was decided that this was not good for domestic customers because you couldn't use the different networks if you weren't on the correct type of hardware/software. So the Government forced the companies to design a universal OS and to make their networks use the same standard. The Japanese cell phones of the early-mid 2000s already had more advanced features than the so-called smart phones that were introduced by Apple and Google to the US markets in the very late 2000s-early 2010s.

Well not wanting the competition and wanting access to the large Japanese market of users the big US companies lobbied US Congress again. Who threatened to start another trade war with Japan if they didn't put a halt to their cell phone OS project. So that was killed in the womb too. Which handed Apple a monopoly inside Japan and is why so many people use Apple devices within Japan today (Android is a blip on the radar).

Any time Japan innovated in the tech space they were shut down by the US Government fast. But despite all that TRON is still the most widely used OS in the world. TRON is an amazing OS.

The way forward is a new kernel much like the nips attempted to do with TRON. Maybe a microkernel and maybe not. It isn't really important how the kernel is laid out. It's only important that it does a lot of things the Linux and other UNIX kernels can't do right now. Like having the ability to do graphics within the kernel itself. But our problem isn't just the kernel it's also the fact that modern hardware is designed in such a way that the intelligence community gets backdoors into everything for free. So we'd need to be able to build both CPUs and GPUs without such backdoors baked into them.

As far as userspace is concerned instead of the hacky shit we have now we'd need one unified standard for drawing applications to the screen. But more importantly we'd want one way to code those applications.

In short: We've want a kernel written a systems programming language that was fast and easy to port to multiple hardware devices (that leaves you with C in the current era). Then on top of that we'd want something lisp-like. Probably not a lisp and probably something with a syntax closer to C or C-based scripting languages. But the idea would be that the entire GUI and the applications would function much like emacs does. Where the end user could modify everything in real time to their liking through scripting and config files. Then see the results of their modifications in real time without having to reboot or reload the kernel.

Everything "legacy" (*nix, DOS, and Windows) could be run through some kind of translation layer like wine. This way we can run old applications without VMs or having to install those OSs within them. This way the old applications could interact with new ones and be somewhat script-able.

But I am getting off track here.

Linux in truth was always hot garbage. A clone of another POSIX OS (Minix) that only saw wide use because of the UNIX wars that were happening in the late 80s-early 90s when Linus first started working on the project. It's a hacky piece of crap with decades worth of hacky shit thrown on top.

Nothing on Linux's kernel offers the type of isolated between processes possible on FreeBSD or OpenBSD. FreeBSD had jails decades before any of this crap on modern Linux. SELinux comes straight out of the NSA and should never be trusted by anyone. AppArmor is a dirty hack on top of a kernel that's riddled with security exploits.

Your Firefox running under Snap restricts access to folders? That's cute. Meanwhile, on OpenBSD you not only get restricted access to folders through a whitelist. You can restrict if they have read/write/execute access. You also get processes being restricted on where and what they can access. If the browser attempts to do anything outside of that whitelist it crashes. Just like every other application that attempts to do such things (include everything in the base system like ls, mv, cp and all the others).

There is no comparison. Nothing offers the type of security one can get from OpenBSD by default.

In short; Everything in Linux is a glorified container or chroot. While OpenBSD was re-built from the ground up to enforce policy upon everything running within it including the kernel and base system. Furthermore, it places the burden of security on the developers/porters instead of the end user. It isn't an ugly hack in other words. It's baked right into the culture of the project.

If you haven't used OpenBSD, it's reasonable to simply not understand the difference. While FreeBSD jails are nice and much better than anything offered within Linux they too suffer from a similar issue of doing everything via glorified chroots.

You use the right tool for the job. For many Linux is far for the right tool for their day-to-day needs. Linux is rarely the right tool for the job. But it's free and those companies do not care about providing good hardware, software and customer support to their users. Instead they care about sales and make things just good enough to trick people into buying them. Which is why the products are intended to become obsolete and unusable sometimes before the user even gets them. That way they're thrown away quickly to be replaced by whatever garbage is being released next year.

It's actually scary how many devices out there are running out of date linux kernels. For example, Ring cameras are everywhere now but they're effectively useless for the task they're intended for. $20 radio jammer and you can disable them easily. Don't even get me started on modern cars and trucks.

I also don't feel like tracking down anymore fuckery because dealing with the freedesktop fags has been very frustrating for the last 20-odd years I've been helping to develop a UNIX OS that relies on X11 for drawing GUIs. Let's just leave it at any time we've sent code upstream its never been accepted. Then later they've always either fear mongered about "security" problems we fixed decades ago or claim some feature isn't supported because
>no one wants to do it

It's kind of amazing how willing they are to lie to everyone's faces. They even claim to be the
>Xorg developers
when they've done nothing in the last 20 years but deny any patches and tag the same shit over and over again as point releases. They even stopped bothering to tag new releases a few years back in an asinine attempt to push their garbage protocol on to everyone.

The Linux kernel has also been a huge pain to push any patches to that aren't corpo sponsored anti-features. We worked very hard on attempting to push changes into the Linux kernel that would allow applications running on it to fully support both our changes to Xorg in Xenocara and various other security related things within our OS (pledge, unveil, and even basic support for doas). They've always denied them over and over again. Even though we have decades of proof now that they work, do not annoy users and provide meaningful mitigations even on machines that otherwise have hardware bugs that are real security issues.

Instead they push for shit like SELinux instead. A horrible maze of bullshit sponsored and developed by the NSA. Which doesn't matter much when you're running a host of garbage as root with 1,000s of known exploits labeled "WONT FIX".

They've even somehow brainwashed modern developers into thinking shit like suid is an instant pwn. When stuff like logind is far worse.

The Linux ecosystem has basically become a lost cause at this point. IBM has fucked it up so much over the last two decades that it's starting to become impossible to maintain a distro without their hacky bullshit being required. We spend more time hacking around their bullshit than we do producing new software these days. It's a full time job not to run something like Gentoo properly without a bunch of systemd shims running as root.

As much as it pained me I finally had to give up on Gentoo around 2020-2021. The project is now dominated by IBM and Google employees from the inside. They subverted the democratic structure around the project and now they can never be voted out. Once they seized full control around 2015 they quickly started removing user choice and employed harsh censorship on the support forums. This made it much harder to maintain a working system outside of the IBM influence using GURU and overlays. Even if you spend the week required to get everything working properly with none of that crap in your kernel or userspace they'll just make a change next week that wipes out all of your progress.

You would need at least 5 people working together every day to keep pace with this stuff and ship a well designed distro. It is not impossible. But it's very hard. The code and defaults for packages isn't that hard but getting people to help out is quickly becoming a non-starter. Even if you did get it off of the ground you'd probably get instantly ddos'd off the web and no one is ever going to donate a penny or even help maintain ports.

People that aren't porting things between the various UNIX OSs really don't understand just how bad the Linux eco-system is now. We've been forced to do shit like fake a running dbus instance simply to get functioning web browsers. No one on the Linux side is interested in improving anything because they're only concerned with those sweet corp. and military donation money.

Attachments:

We should use Gentoo on Workstations and Net/OpenBSD on servers, can you confirm OP?

I didn't know ZFS uses zstd :(. I wanted to install OpenBSD but it didn't see my hd so I'm stuck with FreeBSD for now.

>>4531

Don't read the Gentoo mailing lists or look at who's running the project. You will not see one western name among them.
Sam James runs the entire goddamn thing like Batman and never sleeps is online in every gentoo IRC channel 24/7/365 and is Fucking Welsh of all goddamn things

It's entirely an AUS/UK shitshow
Go see his GitHub "thesamesam"

I'm not convinced he's actually one person, it's physically impossible for one dude to do that much day in and day out, he has endless tolerance for newbies and stupid questions, average time between commits is something like 5 minutes 18 hours a day. it's gotta be a 3-letter agency somewhere



matturner gentoo 2021.png

I found out why the Off the Wall forum got deleted. You should read the mailing list archives they're hilarious.

What happened was someone called one of the council members a slut. They got so angry that they cried to the council about it being a CoC violation until the council took a vote on deleting the forum. The forum admins and moderators weren't aware of it until the vote had already passed. All discussion happened on a hidden mailing list that no one but the 7 people on the council have access to.

It leaked out so discussion was moved to a public list. All the mods and admins told them that all they had to do was send a report like any other user and the post would have been edited or deleted and the user warned. In this discussion these council members said the forums were on Gentoo's servers and the donators wouldn't agree with the forum making the project look bad. The admins said that's not true someone donated the server the forum was on specifically to host the forums (it used to be independent). Then the council said the forum would get them into legal trouble because of the content on it. The admins told them no we have common carrier status like every other website. Then the council claimed the forums were bad because people were posting hacky solutions instead of reporting bugs and since none of them used the forums they never saw bug reports. The admin showed like 25 threads from the last week where people were solving issues and telling people to report bugs.

This goes on for like 100 posts. Then the council takes a vote to make the forum hidden. After that another to delete it all together.

All because someone called one of them a slut.

I found a post on the mailing list from a council member that referenced /g/ in this thread:

>We should also keep in mind that the current situation hasn't just magically appeared out of nowhere.
>Gentoo has for many years (it already was when I first came in contact with it ca 15 years ago) a meme in certain forums that in the last years have been leaning more and more to the extremes of a certain side of the political spectrum.
>It is easy to see where is this kind of speech coming from and how we ended up where we are.
>To keep such people away we need a significant part of the community to agree and act so that they are not welcome, and this is not just us the developers, this is something where we need most of the users too!

https://public-inbox.gentoo.org/gentoo-project/4cbc0905-b0c2-e207-9929-93e85be19704@gentoo.org/

The Gentoo mailing lists are really something else and so are the chat logs from their council meetings. These people are fucking insane.



mattst88.jpg

Meet the leader of the Gnome for Gentoo package. Member of Freedesktop's X11 team. Former Google employee. Current Netflix employee. All around massive faggot autist.

https://mattst88.com/blog/2008/11/23/Software_Engineering_is_not_Computer_Science/
https://github.com/mattst88
https://wiki.gentoo.org/wiki/User:Mattst88

I like his manifesto.

https://wiki.gentoo.org/wiki/User:Mattst88/Council_Manifesto

>I served as a Council member for the 2020-2021 term. I am pleased that I was able to bring the issue of rampant Social Contract violations in the Off The Wall subforum to the Council's attention, leading to its necessary closure. I was reelected and served the 2021-2022 term.

>I have been a member of the Community Relations team for the last four years . When I was invited to join, I did so not because I wanted to be involved in more disagreements, but because I thought that this was an area that was ripe for improvement. I believe that the difference between ComRel four years ago and today are night and day in an extremely improved way.

His blog is great.

>When I came to college two years ago, I intended to major in Computer Science. I looked forward to learning and being around people who shared my interest in computers. As I quickly found out, (1) there weren't any people here who shared my interest or even anyone who could speak the same language, and (2) the little bit of learning I would be doing in my computer science classes wasn't interesting to me at all.

>By the end of my first year, after attempting to discuss my feelings (read: disappointments) about the computer science curriculum and one professor specifically with the head of the department, I realized it wasn't going to change anything. If I wanted a computer science degree, I would sit in boring classes, be treated as if I weren't competent enough to possibly know how to program, and I would have to do my assignments in Ada.

Oh no not Ada!

>I knew from first class sign-up day that the Computer Science faculty expected all incoming students to have no programming experience, and moreover didn't care one bit if you did have any. Normally, you find someone with common interests, in this case programming, and you will talk to them about the common interest. Here, they treated me as if I was a know-nothing outsider.

Too smart for school.

>I attempted to find a way out of taking this class, so I introduced myself to the head of the Computer Science department. I told him that I knew a few programming languages already (at this point, C, PHP, and x86 assembly) and asked if there was any way I could avoid taking this class but still be able to take higher level courses.
>He replied that he'd give me the final exam and that if I passed, I'd get credit. "OK" I said, "that sounds great." He got a strange look on his face and paused thoughtfully for a moment. He had been trying to call a bluff that wasn't there. He quickly withdrew the offer when he realized I was serious.

>I was forced to rearrange my entire schedule before I'd even set foot in class to accommodate this entirely useless class. In this two-hour class, we literally spent eight class periods on binary numbers. It was a thorough waste of my time, and the frustrating part was that wasn't even the slightest acknowledgment from the professor. To him, I was just another idiot who couldn't comprehend zeros and ones.

>On top of that, there were no other students who felt like I did (To everyone else, this stuff was magic). The final question on the final exam which was the hardest from the entire class was an assignment to write code to add all the even numbers from 1 to 1000. I answered in x86 assembly.
>x86 assembly
LMAO. I'd really like to know how he was graded for that. Normally I'm expected to answer in a specific language.

>Similar episodes occurred throughout the next two years. I also began to realize that the program wasn't Computer Science but rather Software Engineering. Learning to write fault tolerant, rock solid business applications in Ada, while useful, isn't interesting

>I've been a Physics major for a year now, while still retaining a minor in computer science. In contrast to Computer Science, the Physics professors are helpful and responsive, treat students as intelligent creatures (even if they aren't), and are understanding if there's a course scheduling conflict, which mostly are due to, you guessed it, the Computer Science department not thinking of Physics and upper level Math students.

Damn nazi CS professors.

>Best of all, I recently learned that one of my Physics professors has quite an interest in DEC hardware, including Alphas. He's even nice enough to find a cabinet to house my noisy AlphaServer DS20L in the Physics lab in the Science building.

You just know the professor passed him to get him the fuck out of the class as quickly as possible. Imagine you're teaching CS 101 then some autistic faggot shows up demanding you change the entire curriculum and class schedule because he knows PHP and C already (sure buddy). When you're just trying to teach the concept of binary and an intro to Ada to the other 20 students that are actually paying attention.

Then you give a simple assignment and the retard turns in something written in asm just to be a smug little asshole.

I didn't quote it but he was crying about
>wasting my parent's money
then goes on to talk about how he switched majors halfway through and took on more classes. Classes he's apparently not using now since he decided to become a wageslave for IBM, Google and Netflix. Where he does nothing but write Codes of Conducts and go after violators while being a package janny for Gnome.

The main thing he contributed to Gentoo aside from having that forum deleted because someone called his internet girlfriend a slut are:
>Broke Gnome on multiple platforms because the package was lagging two minor versions behind upstream. (he did away with the patches Gentoo developers had been maintaining for years so it'd still work on non-systemd infected systems)
>despite claiming to love Alpha computers he did away with the stable version. Ensuring no one can ever have stable Gentoo on the arch again
>Did the same thing for two or three other architectures
>Forced a change to X11 in Gentoo that broke it on all hardware from before about 2010
>Stalled the Catalyst project to the point where nothing has gotten done in 7 years

There are some other misc. things he did but I can't be bothered to troll through his wiki page where he sucks himself off anymore.

Going through the mailing list I learned most everyone on the council is an ESL. All of these guys are hostile to the users as well. They seem to hate everyone.

If you want to see just how retarded the council is check out the logs from their IRC channel when they take votes. This is the one from when they first tried to vote to delete the OTW forum.

https://projects.gentoo.org/council/meeting-logs/20190210.txt

If you bother reading through it you'll notice quickly that the only person that knows wtf he's doing is NeddySeagoon. A name I've seen many times over the last two decades. Since he's actually helpful and maintains a lot of helpful guides on both the forum and wiki. He posts all the time and he's always helping newbies out. He's the guy that's been around since 2003 and can tell you right off the top of his head why something in Gentoo is the way it is.

He tries to warn them several times that they're being a bunch of power hungry retards. Of course no one will listen. He doesn't get to vote because he's not on the council.

I wasn't aware of just how bad the council had become. I knew it was bad but they've really become tyrants since about 2014. They have changed the rules so they can stay on the council forever basically. There used to be term limits and strict rules on how voting was done. Now they can just vote themselves in over and over again. There is a group of about 10 people that have seized total control of the project by subverting the voting process around the council.

The council has the final say in everything. They discuss everything on a private mailing list and IRC channel. They use the CoC to ban people they don't like for minor infractions (or no infractions at all). But the CoC doesn't apply to them and they regularly violate it everyday.

It's very apparent that most of them can't program or even make simple ebuilds. They do not use the forum. The do not use the public IRC channels. They hate the users. They hate most of the repo jannys too.

>>4553

FYI when using Gentoo: Using -systemd USE flag does not stop all the systemd libs from ending up on your system. Considering they're all installed by default on all desktop profiles. The OpenRC profile uses everything from systemd to make things work now. The only thing you've done is swap systemd init for the awful OpenRC init. The OpenRC project should have been renamed years ago. Considering the current OpenRC isn't even the same thing anymore and it basically a bad systemd clone.

To avoid all this crap now requires using masks in multiple places plus package.provided and other things to exclude libs by file name. Which wasn't the case 5-6 years ago before the current people that ruined the distro started forcing Red Hat software upon everyone and breaking the EAPI and long standing USE flags.

Most users and developers didn't want any of this shit. But they all got banned and silenced. The systemd thread on the Debian mailing list was proof enough of that. But the same thing has been repeated over and over again.

It's always a small core group of IBM shills bullying everyone and using things like the CoC to get rid of anyone that won't fall in-line.

Ultimately, it's a problem of hardware. The FOSS community never had control of the hardware and the people that do have shut the FOSS community down every time it made minimal progress. So we've spent the last three decades chasing our tails attempting to make stuff like
>GPUs
>Modems and wireless cards
>various misc. hardware
work by having to reverse engineer the drivers. The fact that the modern x86 system relies on what is a blackbox to output graphics being the main issue. But they've pulled the same trick on us over and over again.

We will have to maintain UNIX until the end of the time even if we build something new. Since that's what will be required to maintain support for all the legacy software (Windows/DOS shit included). But there is no point in attempting to build a new OS on top of anything until we control the hardware and have fully FOSS drivers all the way down to bare metal.

Just look at what they did to CPUs in the last 20 years. We had fully open CPUs for the most part. The worse thing was just some hidden opcodes that were easy enough to fuzz and figure out. But now there is an entire UNIX-like OS running in Ring -1 beyond our control that we can do nothing about. Even on systems where it's disabled it's still running. No one knows if disabling the ME/PSP really works and we can never be sure. Then on later versions of CPUs they even took that away.

It has just been one dirty trick after another. First with licensing autism to drive in-fighting and fragmentation. Then later all this crap like systemd to drive in-fighting and fragmentation within the GPL side the moment it gained a little progress. On the other end you have the famous split between the BSD advocates. Where now we have things like file systems and GPU drivers that comprise more lines of code than the entire kernel+base system. On the linux side you get horrible design like PID1 being millions of lines of code that's impossible to audit loading things like logind as root.

Attachments:

>>4532
> In OpenBSD you get a solid kernel and a leader that cares about security and a community of skilled people working with him to provide something really great.
Read https://web.archive.org/web/20131222063633/http://www.openbsd.org/faq/faq3.html#Verify
> In the same directory as the installation sets, each mirror includes a file named SHA256 which contains checksums of the various installation files. You can confirm that none of the downloaded files were mangled in transit using the sha256(1) command:
> $ sha256 -c SHA256
> The OpenBSD project does not digitally sign releases. The above command only detects accidental damage, not malicious tampering. If the men in black suits are out to get you, they're going to get you.

I love botposting
https://desuarchive.org/g/search/text/%20They%20are%20obviously%20talking%20about%20finally%20eliminating%20the%20python%20dependency%20required%20by%20portage%20%28and%20hopefully%20bash%20too%29./

More:
https://desuarchive.org/g/thread/107954138/#q107958657
https://desuarchive.org/g/thread/107892690/#q107895880
https://desuarchive.org/g/search/text/%20It%27s%20always%20a%20small%20core%20group%20of%20IBM%20shills%20bullying%20everyone%20and%20using%20things%20like%20the%20CoC%20to%20get%20rid%20of%20anyone%20that%20won%27t%20fall%20in-line./

cringe.

>>4582

>>4583

You cannot trust the DesuArchive for permanent storage of wrongthink. The administrator of that archive has been caught deleting content from it many times before but nobody cares to talk about it because it's unofficial and a privilege that it's free.

Mirroring it here ensures it'll stay archived and placed in front view of a tech literate audience who will benefit from it.
This information is valuable, and mirroring great things like these should be encouraged.

Reply

©2025 The Cyberix Network

cy-x.net has been tested with the following web browsers:

Netscape 7.1, Opera 7.02, Internet Explorer 6.0, and Dillo 3.2.0 at 1024x768 px & 640x360 px