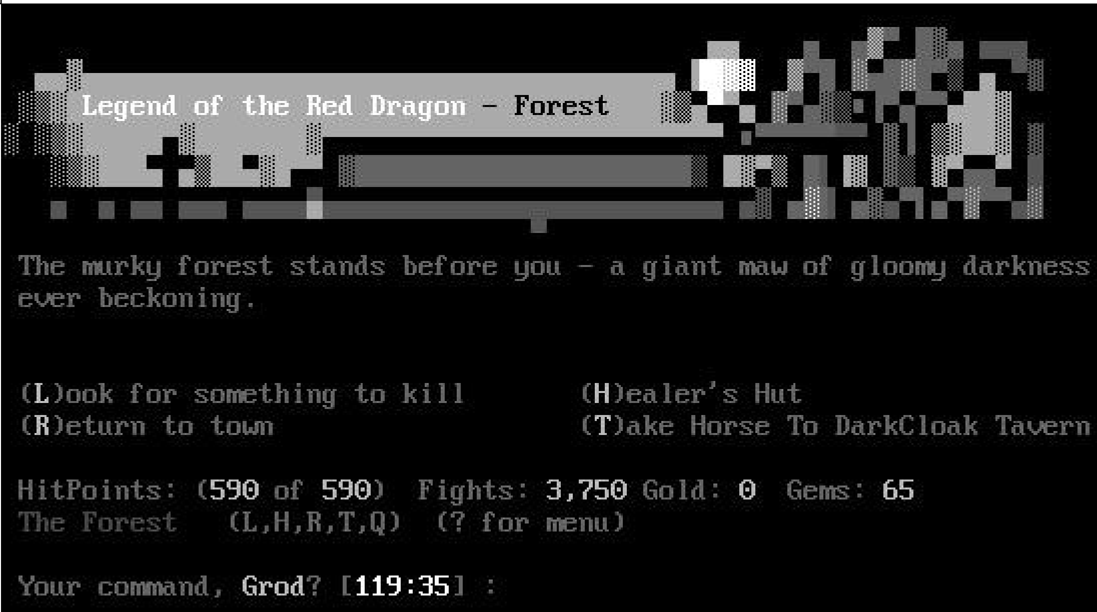


```
Legend of the Red Dragon - Forest

The murky forest stands before you - a giant maw of gloomy darkness
ever beckoning.


(L)ook for something to kill        (H)ealer's Hut
(R)eturn to town                    (T)ake Horse To DarkCloak Tavern

HitPoints: (590 of 590)  Fights: 3,750 Gold: 0  Gems: 65
The Forest   (L,H,R,T,Q)  (? for menu)

Your command, Grod? [119:35] :
```

scrape

two.o0|twenty.one

1/edit/two.o0/well_i_just_dont_know_anymore

well it is here, i nearly didnt make it I've been staring at blank vi terminals for weeks with ideas but no words to go with them

but through some murrrkal here we are.

this issue was fueled by anxiety and distraction, i dunno wtf goes on in peoples head ... social media urrrgh ... of course i'll leverage social media when this issue heads out no doubt.

anyway back on topic, this entire issue was written in a day.5 and what we've got for you this month is the expected curated selection of random stuff almost written properly.

been missing lord since norc dumped his net247 bbs so grabbed it and set it up locally.

gearing up to automatically deploy some servers on libvirt, qemu, kvm so i ramble about what i've learned.

we also take a look at a new bbs doco thats out from gnu2tux

this issue is a little bit larger than the last because i wouldn't shutup

- ed

want to write something?

contact us with your idea.
: tech, gaming, retro stuff
: anything else (discuss)
: we are after cover art.
: submissions welcomed

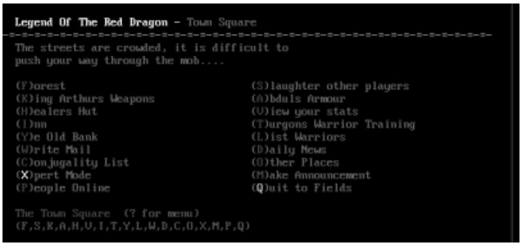
@@scrapezine_at_mameau.com@@

this issue

cover: legend of the red dragon
author: sairuk
date: now

2/re/1989/bbs/lord_a_masterclass_in_door_games

i was around before the internet when bbs'ing was a thing. back when you'd dial a system on the phone it was a one user to one line relationship. these systems sometime had games available called door games.



```
Legend Of The Red Dragon - Town Square
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
The streets are crowded, it is difficult to
push your way through the mob....

(F)orest                                (S)laughter other players
(K)ing Arthurs Weapons                  (A)bduls Armour
(H)ealers Hut                           (V)iew your stats
(I)nn                                   (T)urgons Warrior Training
(Y)e Old Bank                           (L)ist Warriors
(W)rite Mail                            (D)aily News
(C)onjugality List                      (O)ther Places
(X)pert Mode                            (M)ake Announcement
(P)eople Online                         (Q)uit to Fields

The Town Square  (? for menu)
(F,S,K,A,H,V,I,T,Y,L,W,D,C,O,X,M,P,Q)
```

mindful of the systems user base, gaming sessions were limited to a number of turns per day, so to progress you had to call back daily.

it was on these bbses that i was introduced to (l)egend (o)f (t)he (r)ed dragon, developed by seth able, gameplay was simple enough, your character needs to slay a dragon that has been terrorising the town.

stores, healers, inns and a bank where available to the player, to progress visit your current master. a character was one of 3 classes which determined what kind of tests your peers put you through.

random events in the forest would keep the gameplay fresh, meeting olivia, saving people from penyon manor or running into those annoying dwarves who want gamble with you are just a few of events available.

the dragon wouldn't appear until you hit a minimum level at which point you could query the bartender about it, that being said the bartender wouldn't even speak with you about anything of use until you made a name for yourself.

visit the inn to hear a tune from seth the bard (who?) which may impact your game for the day

the game could be extended by sysops (owners of bbses) through the use of igm's providing additional locations and battles for the players to explore although i rarely ran across bbses that made use of these.

your character concludes their journey once the dragon is slain which resets the game, prompts you to create a new character while carrying over some of the skills you had aquired. equipment is reset, gold is reset but the carry over is there as the means to master characater classes.

lord was a multiplayer door game. on an active system every day the story for a player will have progressed when you logged on. you could battle other players, message each other, get married and this is well before the internet was thing.

all of this comes together to keep the game fresh a good 20+ years later which i personally continue to play locally almost daily.

- sairuk

3/poc/2015/steam_controller/rejected_by_a_lack_imagination

ahhh the steam controller, that is a thing. you either love it or hate it, there doesn't seems to be much middle ground between the factions. i would pose that the the steam controller is probably close to one of, if not the most flexible controllers available atm and as such you are only limited by your imagination. lets first look into my major complaints about the controller. I am going to approach this article independant of the steam application and am just using this controller on linux but don't do the whole ... oh ok linux, dude isn't a gamer ... if thats your first thought you may as well stop reading here.

the dpad or lack there of, i don't understand that decision based on the growing popularity of retro stuff and indies leaning towards retro during the entire development time of the controller, while you can use dpad functionality it really was an odd decision to omit a real dpad completely or not take some kind of interchangable peripheral approach to the left pad

the button placement of the 4 face buttons is off; while only slightly off its still enough to be uncomfortable moving it slightly north-east by a couple of mm and losing some throw on the pads wouldn't been a better solution i think.

not rechargable, i just don't get this. it is not a rechargable device, again this is odd. the actual battery placement makes sense as it balances the controller to a point where it falls back into your grip of its own accord but this could've easily been a rechargable unit and not having rechargable circuitry built in feels like a cop out. this also increases the weight of the device.

these gripes aside lets take into account what the controller does well

comfort, it is comfortable and well thought out, the battery position puts the weight into the bottom of the controller which forces it to rest into your grip leaving the main control face to essentially float above your grip leaving you free to operate it. this is contrary to the ds4 where it isn't weighted properly at all and just feels like you need to constantly hold onto it instead it sitting in place, thats ergonomics i suppose

precision, i don't think you could find a more precise _analog_ controller outside of a mouse on the market, while i hate playing fps with this it is a step up from using any stick based analog solutions.

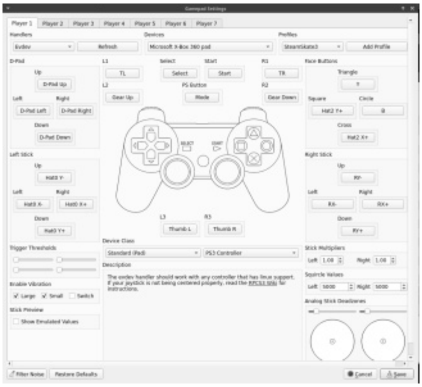
grips, the underside grips are something that open up the flexibility of the controller dramatically. Yes they are referenced as gear l/r in the official layout so the intent is clear but if you look at them as grap l/r it starts to change your thinking of how they can be leveraged ingame

the triggers also have a nice throw, we won't put this up against the 'new fangdangled' haptics based dualsense triggers because haptics support is not included in the linux driver released by sony anyway. after working in a haptics lab for a few years tho ... meh, i'm sure it'll add to the experience but being nearly thrown around a room by haptics feedback in the tech demos we used to run i am not expecting as much as others.

taking these pros and cons into account lets talk about a new approach to the much lauded control scheme in skate 3.

while niche, skateboarding game controls will either make or break the series. if you've never skated before that's fair enough you may just be happy with what you are given by the dev but lets demonstrate an alternate layout for the steam controller that opens up the controls in skate 3 dramatically. for this we'll use skate 3 running on rpcs3.

there are two main changes here



Handlers: Evdev
Devices: Microsoft X-Box 360 pad
Profiles: SteamSkate3

D-Pad
Up: D-Pad Up
Left: D-Pad Left
Right: D-Pad Right
Down: D-Pad Down

L1: TL
L2: Gear Up
Select: Select
Start: Start
PS Button: Mode
R1: TR
R2: Gear Down

Face Buttons
Triangle: Y
Square: Hat2 Y+
Circle: B
Cross: Hat2 X+

Left Stick
Up: Hat0 Y-
Left: Hat0 X-
Right: Hat0 X+
Down: Hat0 Y+

Right Stick
Up: RY-
Left: RX-
Right: RX+
Down: RY+

L3: Thumb L
R3: Thumb R

Trigger Thresholds

Device Class: Standard (Pad) PS3 Controller

Description
The evdev handler should work with any controller that has linux support. If your joystick is not being centered properly, read the RPCS3 Wiki for instructions.

Stick Multipliers
Left: 1.00 Right: 1.00

Squircle Values
Left: 5000 Right: 5000

Analog Stick Deadzones

Enable Vibration
☑ Large ☑ Small ☐ Switch

Stick Preview
☐ Show Emulated Values

analog sticks
move the steam controller pads providing true flick-it style controls

grabbing
the board moves to the gear grips under the controller

additionally we move most other interactions to the shoulders and triggers so the player can minimise their movement around the controller.

unlearning the official control scheme takes some work of course this is a more realistic control scheme than the default.

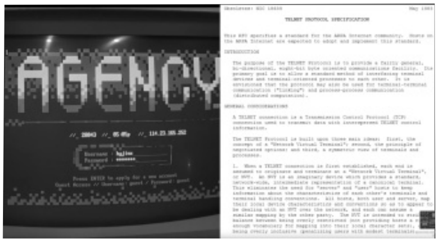
this is an example of making use of the steam controller to dramatically increase your interactivity with a game. i hope you decide to give it a go and try out some other games instead of letting the controller collect dust in a cupboard somewhere.

- sairuk

4/laz/2020/bbs/back_to_the_bbs_documentary

for many years the only real attempt to capture the scene from the bbs days has been jscotts efforts in the bbs documentary which is available under the creative commons license or was available to purchase as a box set

this has served as a staple on my media center for many years getting constant rewatches but now we see a newcomer capturing what it meant to be up and online back in the day



// 28843 // 05:46p // 114.23.185.252

Username : bgjlee
Password : ********

Press ENTER to apply for a new account
Guest Access // Username: guest / Password: guest

Obsoletes: NIC 18639 May 1983

TELNET PROTOCOL SPECIFICATION

This RFC specifies a standard for the ARPA Internet community. Hosts on the ARPA Internet are expected to adopt and implement this standard.

INTRODUCTION

The purpose of the TELNET Protocol is to provide a fairly general, bi-directional, eight-bit byte oriented communications facility. Its primary goal is to allow a standard method of interfacing terminal devices and terminal-oriented processes to each other. It is envisioned that the protocol may also be used for terminal-terminal communication ("linking") and process-process communication (distributed computation).

GENERAL CONSIDERATIONS

A TELNET connection is a Transmission Control Protocol (TCP) connection used to transmit data with interspersed TELNET control information.

The TELNET Protocol is built upon three main ideas: first, the concept of a "Network Virtual Terminal"; second, the principle of negotiated options; and third, a symmetric view of terminals and processes.

1. When a TELNET connection is first established, each end is assumed to originate and terminate at a "Network Virtual Terminal", or NVT. An NVT is an imaginary device which provides a standard, network-wide, intermediate representation of a canonical terminal. This eliminates the need for "server" and "user" hosts to keep information about the characteristics of each other's terminals and terminal handling conventions. All hosts, both user and server, map their local device characteristics and conventions so as to appear to be dealing with an NVT over the network, and each can assume a similar mapping by the other party. The NVT is intended to strike a balance between being overly restricted (not providing hosts a rich enough vocabulary for mapping into their local character sets), and being overly inclusive (penalizing users with modest terminals).

a new series called back to the bbs is currently being created by the channel als geek lab on youtube (gnu2tux on reddit). so far we have seen two episodes released.

I had the opportunity to have a quick chat with gnu2tux on reddit yesterday and found out they originally planned for 3 episodes but based on the content they have gathered to date it looks like we will see more than originally planned.

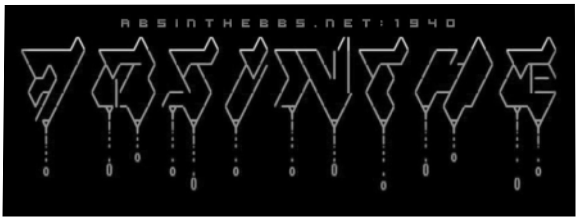
looking at gnu2tuxs main page it lists episodes topics covering mods, hardware, server software, privacy, ansi, demoscene and the underground. which is quite exciting.

so far these two episodes have achieved good topical coverage from general bbs info, ansi, how to use a bbs and some interviews with sysops to newer users. the documentary comes across as a passion project one that the author seems legitimately passionate about the bbsing, in that regard i felt i should reach out to gnu2tux with a couple of questions to get a feel for the motivation behind it.

the production quality of the doco is top notch and particularly when jumping into the door games episode (ep.2) you will learn a lot about the modern day efforts going into developing new titles, some of which i'd heard of being developed but haven't yet had the time to play.

really if you never experienced the bbs door gaming days i couldn't recommend a better introduction than the first 15m of episode 2.

I asked gnu2tux about the motivations behind this series; the passion for the topic was apparent in their reply. driven from the frustrations of the modern day internet they recalled the oldschool days of bbsing and after quite a few years away from the scene (like a lot of us) they dropped into the absinthe board after learning that telnet boards were a thing.



the sysop there was awesome and supportive and had a strong community. this just reinforced what a digital community was supposed to be and the freedoms that came with it. unlike the corporatisation of the modern day net.

every board is a new experience like visiting a new town where estate developments haven't taken hold, from the occasional generic board to the fully modded, each is unique.

i asked them about hopes for the doco and the answer was simple, it is an effort to provide an opportunity for people to understand and experience what bbsing was and is in the current day. when asking about the possiblity of a physical release they said is not currently planned due to the costs and production completexities involved which all makes sense to me. while it would be nice to have the doco sitting on shelf, i am still trying to track down a copy of jscotts efforts since i missed the run of the physicals

if you were in the scene, still are in the scene or completely missed it i couldn't recommend this series highly enough it'll drive you back to the bbs and as such is aptly named.

it is my opinion that this doco acts a voice for what digital community should be.

- sairuk

5/boot/2020/m.2/just_what_were_they_thinking

picked up my first m.2 drive after upgrading my pc to a ryzen 7 system w/32G ram.

not a bad upgrade from the i7 2600k w/16G i managed to destroy before christmas and damn sight quicker than the rpi4 i covered in the last issue for sure.

m.2 drives, what an odd space to enter unaware of whats going on. i picked up a kingston drive from amazon.

it was advertised as an A2000M M.2 2280 Nvme Internal SSD PCie and even i had to question if i'd purchased the correct model for my board based on the supported specs for m.2 from the manual

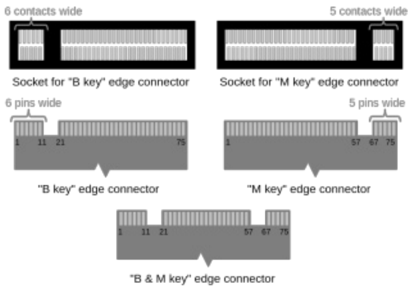
The manual states m.2 socket 3 with m key type 2242/2260/2280/22110 pcie 3&4.

uh yeh sure thing, i ordered my drive based on the 2280 and m.2, pcie info and all is well ... i thought

anyway then of course the minor heart attack kicks in about m key as i soon discovered there were 3 key types for m.2 drives

m key, b key and b+m key, oddly the key type shaped like an m is not the m key type but the m+b key type, the m type notch is to the right of the device (front) with 5 pins not 6 like the b key with the notch on the right ...

holy spit i'd rather go back to ribbon cables without the red pin 1 print at this rate



this is insane and just screams lack of forethought again on the part of component manufacturers

the keying was probably the most under mentioned (i make up phrases now) spec on every ad for parts i looked at.

then of course there is the length of the device which is what the 2280 refers to. you need to ensure it'll fit your board mounts before purchase.

the really interesting part is that if you go this way, probably dependant on your mobo, you will lose functionality after installing an m.2.

on my board that meant that the 1st m.2 slot disabled sata ports 5&6 while the utilizing the second m.2 slot restricted the pcie 16x slot back to 8x fast storage is one thing but it needs a better implementation.

really this is just a rant about the state of m.2 after coming into it

- sairuk

6/doc/2021/virtualization/intro_open_source_virtualization

i have been playing with kvm, qemu and libvirt of late building a deployment framework for some new servers we are planning to spin up. I'd not delved into this infrastructure before this project and it has been an interesting experience with alot of confusing info out there i am still sifting through. here in this article i'm going to cover what i have discovered to date

there are 3 parts to this deployment

kvm
is the underlying (k)ernal based (v)irtualzation (m)odule build into linux, it requires your board supports hardware virtualization in some manner

qemu
is the (q)uick (emu)lation layer privoding all your emulated hardware support

libvirt
is an api to manage qemu (and other virtualization solutions) it runs a daemon you can interact with providing a virualization environment

there are a number of management tools shipped with libvirt, this is generally your starting point, most tutorials out there focus on the graphical tools which is fine but apart from a quick mention here we wont be covering those

there are other tools available but i haven't personally used them yet so they wont be covered.

virt-manager
is a gui for managing vm creation, editing etc. it connects to the qemu:///system namespace by default.

virt-viewer
is a gui console interface for the vm allowing you to remotely connect

virsh
is a command line tool that will give you a manangement shell within the virtualzation environment where you can commands to manage the virtualized infratructure by default this tool connects to the userspace qemu://session namespace, this behaviour can be changed through the configuration

virt-inst
is a cli tool to deploying hosts, if can be called to create a new vm with a number of commands line options

looking into each component in a bit more detail,

kvm
is the kernel level virtualzation layer, it is a type 2 hypervisor that being one that relies on an os. there is not too much to do here in general use, install it and qemu-kvm then ensure hardware virtuzation is operational in the bios

qemu
noting that you can run qemu without kvm, nor libvirtd but performance may suffer and you will not get the suite of tools libvirt provides, we may look at qemu as a standalone in another article

qemu:///system vs qemu:///session

qemu has two default namespaces; the system namespace vs session is something i'm still getting my head around so take this section with a grain of salt and do your own research.

the user session namespace provides some isolation between users and their own vms it also restrics access to certain operations (e.g. qemu can't commission a macvtap network device) so depending on the use case you may choose to deploy here

where as the system namespace provides additional functionality, operations and privilege (e.g. create network bridge devices at deploy time) the libvirt doco eludes to servers running the system namespace for servers, continuing my research on that though.

libvirt
provides the api for managing qemu (and other) deployments. deploying a host with virt-install is genrelly trivial for now here are some commands to get you going

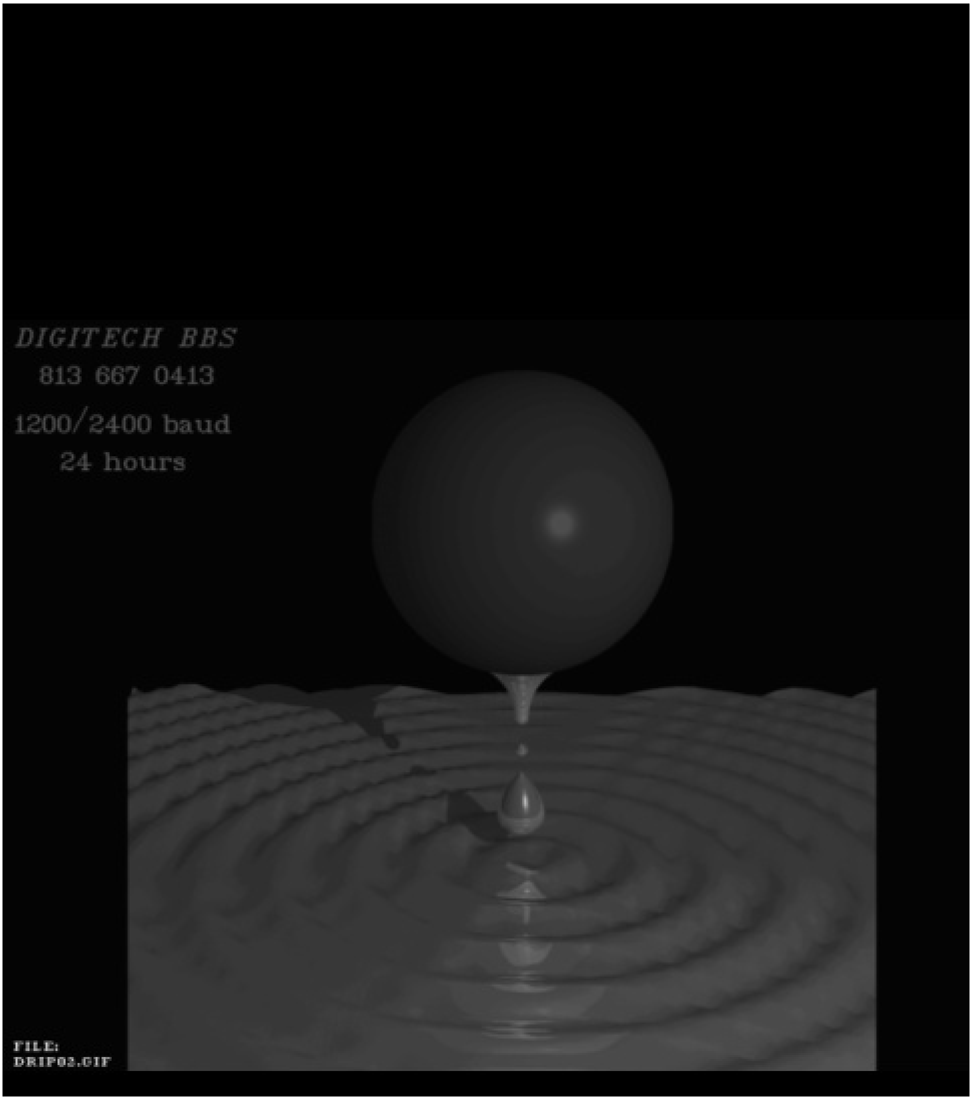
```
virsh
 connect to default namespace
(remember system vs session)
list:
 show you list of all vms
define
 create a vm
dumpxml
 dump the domain configuration to
stdout
net-list
 show a list of all configure netowkr
net-define
 creaete a network
net-dumpxml
 dump the network configuration to
stdout
```

an installation command example:

```
 virt-install \
 --name ${NAME} \
 --ram ${RAM} \
 --disk path=$
{QCOWR},boot_order=1,size=${DISK} \
 --vcpus ${VCPU} \
 --os-type linux \
 --network network=${NETWORK} \
 --graphics none \
 --console pty,target_type=serial \
 --location "${REPO}"
```

as for a disclaimer, this is what i've pulled together over the last few days of looking into this stack so run with it but do your own research

- sairuk



About

Members

ed - sairuk (@sairukau)

Greets (no order)

slateman
blahjedi
norc
gnu2tux

Creds

whomever did the m.2 diagram

- https://en.wikipedia.org/wiki/M.2

Al's Geek Lab

- https://alsgeeklab.com/