

合首页 / 虚拟化 / 正文

_

虚拟化 synology

PVE直通DG1并使用SA6400驱动DG1教程

2023-12-23 / 0 评论 / 9 点赞 / 414 阅读 / 11758 字



Intel DG1显卡的编解码性能非常的强悍,将其用于NAS中的编解码任务是一个非常好的选择,本文旨在介绍Intel DG1独立显卡在群晖SA6400系统下的编解码安装。但由于DG1的硬件兼容性非常差,如果你没有DG1这张显卡,请先尝试下咨询网友自己的硬件平台是否兼容再决定是否购买尝试,切不可在不知兼容性的情况下直接购买。

介绍视频:

- 全网首发! NAS独显转码新王晋级, Intel DG1驱动适配计划第一阶段完成!
- NAS独显转码新王晋级, Intel DG1驱动第二阶段, PVE内核+群晖SA6400驱动分享!

1.1 名词解释

裸机:物理机器直接安装作业系统 (SA6400) 的统称。

虚拟机直通:物理机安装虚拟机管理器(ESXi、PVE)等,其中作业系统(SA6400)以虚拟机的形式部署,并将某个特定设备(DG1)分配给该虚拟机独享使用。

本教程兼顾裸机安装SA6400或者PVE直通安装SA6400并使用DG1;如果你是**裸机**的方式,请直接跳步问读 二、配置BIOS 和 六、RR配置 ;如果你是**虚拟机直通**的方式,请从头开始阅读。

1.2 虚拟机直通环境介绍



这里我们介绍下我们的老贞上(测试半台);

• CPU : Intel 15-10400

_

- 内存: DDR4 2666 8G x 4
- 显卡: 华硕圣旗DG1被动散热版
- 硬盘: 镁光5100 PRO 960G版本

我们需要准备一个容量大于8G的优盘,用于写入 PVE 安装系统。

当然最开始的还是需要先在不安装 DG1 的情况下配置BIOS。

二、配置BIOS

这里需要你具备一定的基础,需要了解如果进入BIOS,准确找到相关选项,如何修改BIOS并保存等。需要修改的选项如下:

- 1. 打开 above4G 选项。
- 2. 打开 UEFI 引导,关闭CSM兼容模式。
- 3. 打开 resizableBAR 选项 (如果有),没有也不强求,该选项打开会增强一定兼容性,如果你未能成功点亮DG1,可尝试打开该选项再试试。
- 4. 关闭PCle子选项中的ASPM,由于DG1的PCle IP兼容性非常差,且不太符合规范,所以很多高阶 PCle的特性请务必关闭这些经常变换PCle工作状态的特性。

此时请尝试插入DG1,尝试是否可以点亮系统,如果不可以请多尝试诸如更新BIOS,检查bios设备等选项。

有些企业级平台对于PCIe的规范性要求比较严格,DG1极有可能无法通过其PCIe的测试,比如 R730/R730XD明确无法通过,所以也无法点亮。

三、PVE系统安装

PVE系统安装时,请不要插入DG1,首先我们需要准备下载两套软件:

1. 官方pve-7.4下载点我

2. 官方rufus 写入软件下载点我

这里我们教程演示基于 PVE7.4 ,系统足够稳定是 NAS 长久稳定运行的前提,这里我们选择非常稳定 的 PVE7.4 ,不选 PVE8 也是基于稳定的考量。

Rufus 4.3.2090 (Portable)	- 🗆 X				
设备选项					
设备					
CPBA_X64FRE_ZH-CN_DV9 (F:) [3	2 GB] 1~				
引导类型选择					
proxmox-ve_7.4-1.iso	⊘ 选择 2				
分区类型	目标系统类型				
GPT ~	UEFI (북ECSM) · ?				
✔ 显示高级设备选项					
格式化选项 ———					
卷标					
PVE					
文件系统	簇大小				
NTFS ~	4096 字节 (默认) ~				
✔ 显示高级格式化选项					
<u> </u>					
准备就绪					
🔇 () 😤 🔳	开始 3 关闭				
E在使用镜像: proxmox-ve_7.4-1	iso				

写入镜像过程中:

Ξ

蔚<mark>然小</mark>茄

设备选项 ————	
设备	
CPBA_X64FRE_ZH-CN_DV9 (F:) [3	2 GB] ~
引导类型选择	
proxmox-ve_7.4-1.iso	~ ⊘ 选择
分区类型	目标系统类型
GPT ~	UEFI (╡╞ CSM) ~ ?
▼ 显示高级设备选项	
おキルキュ	
俗式化选坝 ———	
卷标	
PVE	
文件系统	簇大小
NTFS ~	4096 字节 (默认) ~ ~
✔ 显示高级格式化选项	
状态	
正在写入钱	竟像: 23.9%
\$\$ (i) ≵ 🔳	开始取消

镜像写入完毕:

Ξ

Q

<mark>蔚然小</mark>茄

Q

	进币							
以田	见坝 ————————————————————————————————————							
设备	设备							
NO_LA	NO_LABEL (磁盘 2) [32 GB] ~ /							
引导类	型选择							
proxm	ox-ve_7.4-1.iso	~ 🕗	选择					
分区类	型	目标系统类型						
GPT	~	UEFI (╡╞ CSM)	~ ?					
✔ 显示	高级设备选项							
格式	化选项 ———							
卷标								
PVE								
文件系	统	箍大小						
FAT32	(默认) ~	16 K 字节 (默认)	\sim					
▼ 显示	高级格式化选项							
状态								
		141						
	/庄奋别?	泊						
S () ≋ 🔳	开始	关闭					
检测到 1	个设备		00:00:52					
此时启动U盘我们就准备	好了,这时我们将写入镜像后的	1优盘插入到系统,启	动U盘中的安装	長系统。				
启动后可以参考佛西大佬	的教程: https://foxi.buduanwa	ng.vip/virtualization/2	229.html/ 搜索					
开始安装pve 可直接定位到目标位置,参考图文一步步进行。								
安装完毕过后我们进入到	」系统中,此时我们需要对电脑进	进行一些配置:		3				

四、配置内核

Ξ

4.1 安装指定内核

其中第一个就是更换为皮蛋熊修改过的内核,为了防止之前系统或存在同样的内核,所以先将其卸载:



Q

_

内核安装完毕后,重启检查是否内核启动完成:



100cmhrc..... aname -a

Linux pve 5.15.131-1-pve #1 SMP PVE 5.15.131-2 (2023-11-14T11:32Z) x86_6

可以看到上面内核已经修改成功了,接下来我们将修改内核参数。

4.2 修改内核参数

下面开始修改内核参数:

Ξ

● ● Bash ▼ 즽 # vim /etc/default/grub # 修改 GRUB_CMDLINE_LINUX_DEFAULT

GRUB_CMDLINE_LINUX_DEFAULT="quiet intel_iommu=on spectre_v2=off spectre_

上面的选项解释如下:

- 1. intel_iommu=on 开启intel的iommu, 这里如果是amd的处理需要修改为: amd_iommu=on
- 2. spectre_v2=off spectre_v1=off 两个CPU BUG的修补,是侧信道攻击,安全类BUG,但修复 会导致CPU性能下降,这里我选择不修补,可根据自己情况进行选择。
- 3. pcie_acs_override=downstream IOMMU分组补丁,用于将设备分组分的更开一点
- 4. pcie_aspm=off 关闭PCle硬件链路节能机制,DG1的PCle IP不太兼容这个,建议关闭,关闭后可以运行的更为稳定。

上面选项修改完毕后,使用 update-grub 更新引导,并重新启动PVE系统;

重新启动后将下面脚本保存为 iommu.sh , 给与可执行权限 chmod +x iommu.sh , 并执行, 即可查看 iommu分组:

```
Bash

#!/bin/bash
for g in `find /sys/kernel/iommu_groups/* -maxdepth 0 -type d | sort
echo "IOMMU Group ${g##*/}:"; \
for d in $g/devices/*; do \
echo -e "\t$(lspci -nns ${d##*/})"; \
done; \
done;
```



确保一切正常后,可进行下面的创建虚拟机的环节。

五、创建虚拟机

开始正式安装群晖SA6400,这里我们将创建一个SA6400的虚拟机:点击创建虚拟机 Create VM :



填入Name SA6400 ,其中 VM ID 是自动生成的:



Ξ

Node: VM ID: Name: Start at boot:	pve 106 SA6400 ☑		Resource Pool: Start/Shutdown order: Startup delay: Shutdown timeout:	any default default	
Help Create: Virtual I General OS	Machine	CPU Memory	Network Conf	Advanced 🕢	Back Next
) disc image file (iso)		Guest OS		
Storag	e: local		Туре:	Linux	
ISO imag			Version:	6.x - 2.6 Kernel	
 O Use physical ■ Do not use a 	I CD/DVD Drive				
				Advanced 🗹	Back N

<mark>蔚黙小</mark>茄

Q

Graphic card:	Default	SCSI Controller:	VirtIO SCSI single		~
Machine:	q35	Qemu Agent:			
Firmware					
BIOS:	Default (SeaBIOS)	Add TPM:			
O Help			Advanced 🖂	Back	Next

至少给200G的空间:

Ξ

Create: Virtual Mach	ine			\otimes
General OS Sy	rstem Disks	CPU Memory Network	Confirm	
scsi0 🛱	Disk Bandwi	idth		
	Bus/Device:	SCSI 🗸 0 🗘	Cache:	Write back 🗸 🗸
	SCSI Controller:	VirtIO SCSI single	Discard:	
	Storage:	local-lvm	IO thread:	
	Disk size (GiB):	200 🗘		
	Format:	Raw disk image (raw $\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \$		
	SSD emulation:		Backup:	
	Read-only:		Skip replication:	
Add			Async IO:	Default (io_uring)
P Help			Adva	nced 🗹 🛛 Back Next

Create: Vir	tual N	lachine						\otimes
General	OS	System	Disks	CPU	Memory			
Sockets:		1				Туре:	host	× ~
Cores:		12				Total cores:	12	
		40			^	0000	400	^
VCPUS:						CPU units:		
CPU limit:						Enable NUMA:		
CPU Affinity	:							
Extra CPU F	-lags:							
Default	-0	•)	md-clear	Re	quired to le	et the guest OS kr	now if MDS is mitigated correctly	
Default	- ()	••	pcid	Me Inte	Itdown fix el CPUs	cost reduction on	Westmere, Sandy-, and lvyBridge	
Default	-0	•0+	spec-ctrl	Allo	ows improv	ved Spectre mitiga	tion with Intel CPUs	
Default	-0	• •	ssbd	Pro	tection for	"Speculative Stor	re Bypass" for Intel models	
Default	-0	•••	ibpb	Allo	ows improv	ved Spectre mitiga	tion with AMD CPUs	
😧 Help							Advanced 🗹 🛛 Back	Next

设置分配内存大小:

Ξ

Create: Virtual Machine						\otimes
General OS System	ı Disks	CPU	Memory	Network	Confirm	
Memory (MiB):	8192		٥			
Minimum memory (MiB):	8192		0			
Shares:						
Ballooning Device:						
					· · · · · · · · · · · · · · · · · · ·	Q)
e Help					Advanced 🖂 🛛 Back 🛛 Next	

蔚林小<mark>跖</mark>

Create: Virtual I	Machine				\otimes				
General OS	System Disks	CPU Memory	Network Cor	nfirm					
🗌 No network de	No network device								
Bridge:	vmbr0		Model:	VirtIO (paravirtualized)	~				
VLAN Tag:			MAC address:						
Firewall:									
Disconnect			Rate limit (MB/s):	unlimited	0				
MTU:	1500 (1 = bridge MTU)		Multiqueue:		\circ				
0 Help				Advanced 🛛 Back	Next				

点击完成:

Ξ

Create: Vir	rtual M	achine							\otimes
General	OS	System	Disks	CPU	Memory	Network	Confirm		
Key ↑		Va	alue						
ide2		n	one,media=	-cdrom					A
machine		q;	35						
memory		8	192						
name		S	A6400						
net0		vi	rtio,bridge=	vmbr0,fi	rewall=1				
nodename	e	Þ١	/e						
numa		0							
onboot		1							
ostype		12	6						
scsi0		lo	cal-lvm:20),iothread	l=on,cache=	writeback			
scsihw		vi	rtio-scsi-sir	ngle					
sockets		1							
vmid		1(06						
Start afte	er creat	ed							
							Advanced	Back	Finish

<mark>蔚</mark>然小<mark>站</mark>

Virtual Machine 106 (Virtual Machine 106 (SA6400) on node 'pve' No Tags 🖋					
Summary	Add ~ Remove Edit	Disk Action V Revert				
>_ Console	📟 Memory	8.00 GiB				
Hardware	💭 Processors	12 (1 sockets, 12 cores) [host]				
📥 Cloud-Init	BIOS	Default (SeaBIOS)				
Options	🖵 Display	Default				
Task History	📽 Machine	q35				
	SCSI Controller	VirtIO SCSI single				
	OD/DVD Drive (ide2)	none,media=cdrom				
🖺 Backup	🖨 Hard Disk (scsi0)	local-lvm:vm-106-disk-0,cache=writeback,iothread=1,size=200G				
13 Replication		virtio=6A:9A:DA:20:71:C8,bridge=vmbr0,firewall=1				
Snapshots						
♥ Firewall ▶						
Permissions						

删除 CD/DVD Driver :

Ξ

Virtual Machine 106 (SA6400) on node 'pve' No Tags 🖋						
🖻 Summary	Add V Remove Edit	Disk Action V Revert				
>_ Console	🚥 Memory	8.00 GiB				
Hardware	💭 Processors	12 (1 sockets, 12 cores) [host]				
📥 Cloud-Init	BIOS	Default (SeaBIOS)				
Options	🖵 Display	Default				
Task History	📽 Machine	q35				
	SCSI Controller	VirtIO SCSI single				
	🖨 Hard Disk (scsi0)	local-lvm:vm-106-disk-0,cache=writeback,iothread=1,size=200G				
🖺 Backup		virtio=6A:9A:DA:20:71:C8,bridge=vmbr0,firewall=1				
13 Replication						
Snapshots						
♥ Firewall ▶						
■ Permissions						

添加直通的DG1设备, 仅添加显卡就可以了, 不用管声卡和周边设备:

Q

蔚然小品

>_ Console	Memory	8.00 GiB
Hardware	Processors	12 (1 sockets, 12 cores) [host]
📥 Cloud-Init	BIOS	Default (SeaBIOS)
Options	🖵 Display	Default
Task History	📽 Machine	q35
	SCSI Controller	VirtIO SCSI single
Monitor	🖨 Hard Disk (scsi0)	local-lvm:vm-106-disk-0,cache=writeback,iothread=1,size=200G
🖺 Backup		virtio=6A:9A:DA:20:71:C8,bridge=vmbr0,firewall=1
17 Replication	📟 PCI Device (hostpci0)	0000:05:00.0,pcie=1
Snapshots		
♥ Firewall ▶		
Permissions		

下载RR引导并解压,并把引导文件 rr_4GB.img 上传到PVE中,这里是上传到 /root/arpl 目录;

PS:目前RR引导是最完善的黑群引导,但因为一些原因已经删库转为私有,关于引导文件的获取,可QQ 频道搜索 矿神NAS研究社 或者 群晖引导编译交流 找寻最新的免费下载地址,这里我就不上传了。

这里我们将RR引导模拟为U盘使用,这里采用jim大佬博客中介绍的方法,详细可点击查看参考文档:修改上面我们创建的虚拟机的配置文件,这里我的虚拟机ID是106,所以地址是

/etc/pve/qemu-server/106.conf ,请根据自己的id进行修改对应路径,并在文件中添加下面的配置选项

Bash Table Bash Tab

上面配置中的 file=/root/arpl/rr_4GB.img , 就是RR引导的具体路径。这里把引导文件 rr 4GB.img 上传到 /root/arpl 目录中,即可正常启动RR并引导黑群晖了。

注意:第一个版本需要先取消RR内部自带的i915驱动(下面会讲述如何取消),后期代码提到RR后下面的操作不需要做,直接驱动。

最后的RR黑裙虚拟机,配置如下:

_

蔚黙小品

	Summary	Add V Remove Edit Disk Action V Revert
∽ 📷 pve		
📦 103 (dg1)	>_ Console	mm Memory 8.00 GiB
DVA3219)	Hardware	Processors 12 (1 sockets, 12 cores) [host]
📮 101 (ubuntu)	🗅 Cloud-Init	BIOS SeaBIOS
102 (windows)	Options	Display Default
105 (WIN10-GAMF)	Task History	📽 Machine q35
Solution (method and president		SCSI Controller VirtIO SCSI
Sallocal-lvm (pve)	Monitor	□ Hard Disk (sata0) local-lvm:vm-104-disk-2,cache=writeback,size=200G
	🖺 Backup	□ Hard Disk (sata1) local-lvm:vm-104-disk-1,cache=writeback,size=200G
	13 Replication	A Network Device (net0) virtio=42:C9:CA:3E:A7:56,bridge=vmbr0,firewall=1
	Snapshots	E PCI Device (hostpci0) 0000:05:00.0,pcie=1
	♥ Firewall ▶	Serial Port (serial0) socket
	-0 Pormissions	□ Unused Disk 0 local-lvm:vm-104-disk-0
	- remissions	

其中直通了DG1, 这里该设备的BDF是 0000:05:00.0 ,这里选择 Q35 还是 i440fx 都可以,没有 任何问题。



Ξ

注意:以下教程裸机或者虚拟群晖通用:选择配置引导 configure loader :



输入 menu.sh 并回车:





选择模块 Modules menu :

Ξ





蔚就小跖

👘 103 (ag1)	>_ 00113010	4		
100 (DVA3219)	Hardware			
101 (ubuntu)	Cloud-Init			
102 (Mildons)	Options			
105 (WIN10-GAME)	Task History			
🗐 🖬 local (pve)	Monitor		Hodu les 1	
U local-lvm (pve)	🖺 Backup		Choose a option	
	13 Replication		Show/Select modules	
	Snapshots		 Defect loaded modules Upload a external module 	
	♥ Firewall ▶		p Priority use of official drivers: false e Exit	
	Permissions			
			L	
			Cancel>	

取消勾选

Ξ

evbug, i915-compat, i915, dmabuf, drm_kms_helper, drm, drm_mipi_dsi,

drm_panel_orientation_quirks, mei-gsc, mei_hdcp, mei_iaf, mei, mei-me, mei_pxp, mei_txe

这么多选项:

	Modules
Select modules to include	
[*] e1000	Intel(R) PRU/1000 Network Driver
[*] e1000e	Intel(R) PRU/1000 Network Driver
[*] eepron_93cx6	EEPROM 93cx6 chip driver
[*] ehci-hcd	USB 2.0 'Enhanced' Host Controller (EHCI) Driver
H*l ehci-nci	FHC1 PC1 platform driver
[] evbug	Input driver event debug module
[*] eudeu	Input driver event char devices
[*] failover	Generic failover infrastructure/interface
[*] fb	Framebuffer base
[*] fixed_phy	Fixed MDIO bus (MDIO bus emulation with fixed PHYs)
[*] hdmi	hdmi.ko
[*] hpsa	Driver for HP Smart Array Controller version 3.4.20-200
[*] huc_console	hvc_console.ko
[*] humon-vid	hwmon-vid driver
[*] i40c	Intel(R) 40-10 Gigabit Ethernet Connection Network Driver
	18042 keyboard and mouse controller driver
L] 1915-compat	Intel Graphics
L J 1915	Intel Graphics
	Intel(R) Ethernet Adaptive Virtual Function Driver
L*J 1b_Cm	InfiniBand Ch
L*J ib_core	core kernel InfiniBand API
L*J 1b_umad	Infiniband userspace MAD packet access
L*J 1b_uverbs	InfiniBand userspace verbs access
	Intel(R) Gigabit Ethernet Linux Driver
L*J 1gbut	Intel(R) Gigabit Virtual Function Network Driver
	Intel(R) 2.56 Ethernet Linux Driver
L*I Intel-gtt	Intel-gtt.ko
1*1 Intel_auxiliary	Huxillary Bus Standalone
Test Theo	IPU6 protocol stack for Linux
	33%
	(Select all) (Cancel) (Decelect all)
	Concer / Concer / Concer / Concert all/





注:后期RR引导适配了dg1后,这些选项务必进行重新勾选,就不再需要单独安装驱动了。

MEI IAF

MEI PXP

Intel(R) Management Engine Interface

Intel(R) Management Engine Interface

MII hardware support library Mellanox ConnectX HCA low-level driver

Mellanox ConnectX HCA Ethernet driver Mellanox ConnectX HCA InfiniBand driver

Multimedia Card (MMC) block device driver

Intel(R) Trusted Execution Environment Interface

Mellanox 5th generation network adapters (ConnectX series) core Mellanox 5th generation network adapters (ConnectX series) IB

一路 OK 后返回重新编译引导:

] mei_iaf

[] mei_pxp

[*] mlx5_core [*] mlx5_ib

[*] mmc_block

E

[] me

] m

[] mei

[*] mlx4_core [*] mlx4_en [*] mlx4_ib

Ξ

<mark>蔚</mark>然小<mark>击</mark>



完成后启动设备:

Ξ



开始群晖系统的初始化设置。。。。

开启SSH:

<mark>蔚黙小</mark>茄

_

へ 文件共享	使用终端机服务登录并管理您的系统。SSH/Telnet 仅支持从属于 administrators 群组的帐户登录。	N.
1 共享文件夹	启动 Telnet 功能	
🔄 文件服务	✓ 启动 SSH 功能	
🚢 用户与群组	端口: 22	
■ 域/LDAP	高级设置	
	禁止使用控制台端口	
▲ 连接性	注意: 建议为登录帐户设置高强度的密码并启用 <u>自动封锁</u> 以获得最强的系统安全性。	
▶ 外部访问		
☆ 网络		
用SSN工具如 mobalterm 动安装步骤:	、xshell、finalshell 寺庄按枏晖:	
• •	Plain text 🔻 🗐	1
cd /volume1/docke curl https://blog	r/ .kkk.rs/upload/i915_debug_pdbear.tar.gz -o i915_debug.	t
# 2. 以普通用户解日	Ē.	
tar -xf i915_debu	g.tar.gz	L
# 3. 切换root用户		L
sudo -i		L
# 4 切换到驱动目录	- -	
cd /volume1/docke	r/i915_debug	
# 5. 安装驱动		L
/install_sh		
•/ •		
面所有执行完毕后,应该	在ssh窗口中有如下信息输出表示一切正常:	
□所有执行完毕后,应该 ●●	在ssh窗口中有如下信息输出表示一切正常: Bash ▼ ♀	
되所有执行完毕后, 应该 ●● HuC firmware: i91	茫在ssh窗口中有如下信息输出表示一切正常: Bash ▼ ♀ 5/dg1_huc_7.9.3.bin	
缸所有执行完毕后, 应该 ● ● HuC firmware: i91 status: R	茫在ssh窗口中有如下信息输出表示一切正常: Bash ▼ ♀ 5/dg1_huc_7.9.3.bin UNNING	

<mark>蔚然小</mark>茄

Q

HuC status: 0x00090001

此时可参考本博客中的其他教程,创建支持intel显卡的docker来使用显卡的解码能力。

注:因为docker或者套件启动时间比驱动加载更前,其加载时系统中并不存在驱动;所以驱动加载后,请 按需重启一下相关套件或者docker容器(该问题在后续将代码合并到RR后会修复)。

七、附件

_

- PVE 内核: https://blog.kkk.rs/upload/5.15.131-pve-pdbear.tar.gz
- dg1 SA6400驱动: https://blog.kkk.rs/upload/i915_debug_pdbear.tar.gz

八、错误排查

因为DG1的PCle IP质量非常糟糕,导致其兼容性堪忧,下面是一些常见的排错手段:

8.1 如果我无法点亮DG1应该怎么办?

- 1. 检查BIOS选项是否设置了上述步骤中的选项
- 2. 根据网友反馈,可以尝试升级或者降级BIOS
- 3. 某些服务器平台R730XD/R730无法过PCIe检测,暂无可解选项

8.2 我可以成功点亮,但是驱动工作看起来不太正常,没有找到 DRM 设备的 card/renderD 节点应该怎么办?

- 1. 检查 dmesg 信息中是否存在 PCI错误或AER错误 ,如果你确定已经关闭 ASPM 选项,仍然出 现这个错误,表示你的主板和DG1不太兼容,目前没有很好的办法。
- 2. 检查 dmesg | grep i915 , 查看是否存在什么错误, 可将错误发到群里, 以便进一步排查。

九、致谢

非常感谢群里的 大熊猫贡贡 、 卡布达 、 刕圐圙 、 长凤大侠 、 jin大佬 、 不浮不 之 、 佳芯 等多位大佬的鼎立支持。更感谢小黄鱼老板 南唐乖乖的角瓜 的鼎立支持赠送的2张个测试 卡,老板承诺不会因为我适配驱动而乱涨价,同时会给报皮蛋熊的网友多送几个 SATA 线,非常感, 领板的慷慨。

virtualization

synology

Ξ	<u> </u>	Q
本文链接:	https://blog.kkk.rs/archives/28	
😪 许可协议:	本文使用《署名-非商业性使用-相同方式共享 4.0 国际 (CC BY-NC-SA 4.0)》协议授权 >	>
上一篇		下一篇
	评论区	
编写评论		
登录	● 提交	评论
0条评论		
	 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	
	2023 © 蔚然小站 • 已运行 149 天 03 时 43 分 50 秒	

∠ 访问量 3.8w □□ 访客量 1.7w
ぷ 本页访客 618