Qwen

安全赋能：面向Linux平台的PyTorch开发工作流与纵深防御体系构建

Linux核心命令行操作与环境配置

在Linux系统上高效开展PyTorch深度学习项目，首先需要熟练掌握其命令行界面。命令行不仅是与操作系统交互的基础工具，更是执行数据处理、环境管理和性能监控等复杂任务的核心。本章节将系统性地介绍Linux文件管理、权限控制、进程监控以及环境变量配置的关键概念与实用命令，为后续的PyTorch项目实践奠定坚实的基石。

文件与目录管理是Linux日常操作的起点。核心命令包括用于列出目录内容的`ls`、切换工作目录的`cd`、显示当前路径的`pwd`、创建新目录的`mkdir`、移除文件或目录的`rm`、复制文件或目录的`cp`、移动或重命名文件或目录的`mv` [122]。对于更复杂的操作，`find`命令可用于递归搜索文件系统，结合`grep`可以实现强大的文本模式匹配，例如在代码库中查找特定函数定义或错误日志 [56]。而`cat`, `less`, `head`, `tail`等命令则提供了灵活的文本文件查看方式，其中`tail -f`尤其适用于实时监控日志文件的动态输出 [89]。这些命令构成了数据集管理、代码版本控制和日志分析的基础。

权限控制是Linux系统安全的核心机制。每个文件和目录都关联着所有者、所属组和其他用户三类主体，并为每类主体设置了读（r）、写（w）和执行（x）三种权限 [90] [94]。`chmod`命令允许用户以八进制（如755代表所有者可读写执行，组用户和其他用户只可读执行）或符号（如`u+x`为所有者添加执行权限）的方式修改权限 [122]。`chown`命令则用于更改文件的所有者和所属组。在共享的GPU服务器环境中，合理设置项目目录的权限至关重要，例如，应确保用户只能访问自己的项目文件夹，避免因误操作导致数据丢失或恶意篡改 [25]。此外，`sudo`命令赋予普通用户临时执行超级用户特权的能力，是进行系统级软件安装和配置变更的关键工具 [110]。

进程监控与管理能力对于维护服务器稳定性和公平分配计算资源至关重要。深度学习训练任务通常是长时间运行的后台进程。`ps aux`命令可以静态地列出当前系统中所有进程的概要信息，常与`grep`结合使用以筛选特定进程，例如`ps aux | grep python` [56]。`top`及其增强版`htop`则是动态监控系统资源消耗的利器，它们能够实时展示CPU、内存、进程ID等关键指标，帮助开发者快速定位资源瓶颈 [1] [2]。当发现某个训练任务异常占用了过多资源时，可以使用`kill <PID>`命令强制终止该进程，以保障其他用户的正常作业

[43]。对于前台运行的任务，可以使用`Ctrl+Z`暂停，再通过`bg`将其放入后台继续运行，或使用`fg`将其调回前台，这种作业控制机制提高了终端使用的灵活性。

环境变量是连接操作系统与应用程序的桥梁，对PyTorch开发尤为关键。`PATH`变量决定了Shell在哪些目录下寻找可执行文件，正确配置它可以使得`python`, `pip`, `torchrun`等命令在任何位置都能被直接调用。`HOME`变量指向用户的主目录，许多应用程序的配置文件默认存放于此。对于涉及自定义CUDA内核编译的高级应用，`LD_LIBRARY_PATH`变量必须包含`.cu`文件编译后生成的动态库路径，否则PyTorch在尝试加载这些自定义扩展时会失败[78]。环境变量通常在用户的shell配置文件（如`~/.bashrc`或`~/.profile`）中进行永久性设置。一个常见的实践是，在激活Conda环境后，Conda会自动修改`PATH`变量，使其优先指向该环境下的Python解释器和库，从而实现了不同项目的Python环境隔离[59]。

核心命令类别	关键命令	主要功能与PyTorch应用场景
文件管理	`ls`, `cd`, `pwd`, `mkdir`, `rm`, `cp`, `mv`	管理代码、数据集、虚拟环境和配置文件[122]。
文件搜索与查看	`find`, `grep`, `cat`, `less`, `head`, `tail`	在大型数据集中查找特定文件；在代码或日志中搜索关键词[56]。
权限控制	`chmod`, `chown`, `sudo`	设置项目目录权限，防止文件误删或篡改[90]；以管理员身份安装系统软件。
进程管理	`ps`, `top`/`htop`, `kill`	监控GPU训练任务的资源消耗，终止异常进程[1][43]。
环境配置	`.bashrc`, `PATH`, `LD_LIBRARY_PATH`	配置Python路径以便全局使用；为自定义CUDA扩展指定库搜索路径[78]。

PyTorch项目工作流：从环境搭建到GPU监控

掌握了Linux基础操作后，便可在其之上构建完整的PyTorch深度学习项目工作流。这一流程涵盖了从环境隔离、依赖安装，到利用硬件加速进行模型训练和监控的全过程。本章节将详细阐述PyTorch项目开发的核心环节，重点突出与Linux系统紧密相关的实践方法。

项目的第一步是建立一个干净、隔离的Python环境。这不仅能避免不同项目间库版本的冲突，也是防范依赖污染和恶意软件注入的第一道防线。主流的虚拟环境工具有`conda`和标准库提供的`venv`。推荐的工作流是为每个新项目创建一个独立的环境，例如使用`conda create -n my_project python=3.9`创建一个名为`my_project`、使用Python 3.9的环境。创建后，通过`conda activate my_project`进入该环境。之后，所有通过`conda install`或`pip install`安装的包都将仅限于该环境使用。为了保证项目的可复现性，可以将环境依赖导出到一个文件中，如`conda env export > environment.yml`或使用`pip freeze > requirements.txt`。这个文件随后可以被团队成员用来重建完全相同的开发环境。

在具备隔离环境的基础上，下一步是安装PyTorch及其生态系统。PyTorch官方提供了基于`pip`或`conda`的安装脚本选择器，用户可以根据自己的操作系统、包管理器和CUDA版本进行定制 [9]。对于拥有NVIDIA GPU的用户，选择正确的CUDA版本至关重要。例如，如果系统安装了CUDA 12.1驱动，就应该选择与之兼容的PyTorch CUDA版本。安装命令通常类似于`pip install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu121` [9]。对于云环境或AMI实例，有时需要手动安装PyTorch，或者通过云服务商提供的特定AMI来简化此过程 [10]。安装完成后，可以通过Python脚本`import torch; print(torch.cuda.is_available())`来验证GPU是否能被PyTorch成功识别和使用。

GPU资源的有效监控是优化深度学习训练效率的关键。`nvidia-smi`（NVIDIA System Management Interface）是NVIDIA提供的一款命令行工具，它能够提供关于GPU状态的丰富信息，包括驱动版本、CUDA版本、GPU利用率、显存占用、温度、功耗以及当前正在使用GPU的进程ID [3] [4]。在训练开始前运行`nvidia-smi`可以确认驱动和CUDA环境是否正常 [126]。在训练过程中，可以定期执行此命令或使用其持续监控模式（如`watch -n 1 nvidia-smi`）来观察资源变化。例如，如果发现GPU利用率持续偏低，但CPU和I/O负载很高，可能意味着数据加载成为了瓶颈，需要优化`DataLoader`的参数（如增加`num_workers`） [15]。反之，如果显存占用迅速达到上限并导致“out-of-memory”错误，则需要考虑减小批量大小或优化模型结构。

随着模型规模的增大，单GPU往往无法满足计算需求，此时就需要采用分布式训练。PyTorch中最常用且高效的分布式训练抽象是`torch.nn.parallel.DistributedDataParallel`（DDP）。DDP将模型复制到多个GPU（甚至多台机器）上，并在每个副本上独立计算梯度，然后通过一个集体通信后端（如NCCL）同步梯度。要启动一个DDP任务，通常需要在脚本开头设置好分布式环境的必要信息，最核心的是`MASTER_ADDR`（负责协调的节点IP地址）和`MASTER_PORT`（用于通信的端口号）这两个环境变量 [76]。在Slurm等集群管理系统中，这些信息通常由作业调度器自动提供。启动DDP任务的标准方式是使用`torchrun`命令，例如`torchrun --nproc_per_node=4 --master_addr="192.168.1.10" --master_port=12345 train.py`，该命令会在本地节点上启动4个进程，每个进程对应一个GPU。理解和正确配置DDP是进行大规模模型训练的前提 [11]。

工作流阶段	核心任务	关键工具/命令	实践要点
环境隔离	创建独立Python环境	`conda create`, `conda activate`, `venv`	每个项目使用独立环境，避免依赖冲突[59]。
依赖安装	安装PyTorch及生态库	`pip install`, `conda install`, `torch.utils.cpp_extension`	根据CUDA驱动版本选择正确的PyTorch发行版[78]。
GPU监控	实时监控GPU状态	`nvidia-smi`	观察利用率、显存和温度，诊断性能瓶颈[3][4]。
分布式训练	并行化大规模模型训练	`torch.distributed.DistributedDataParallel`, `torchrun`	正确设置`MASTER_ADDR`和`MASTER_PORT`，理解DDP工作原理[11][76]。

远程安全访问：SSH服务加固与最佳实践

对于在云端服务器或本地高性能计算集群上工作的开发者而言，通过SSH（Secure Shell）进行远程访问是日常工作不可或缺的一部分。然而，不安全的SSH配置是黑客攻击服务器的主要入口之一。本章节将深入探讨如何系统性地加固SSH服务，构建一个抵御常见网络威胁（特别是暴力破解攻击）的坚实防线。

强化SSH安全的首要原则是放弃传统的密码认证，全面转向基于公私钥的认证机制。密码认证极易受到字典攻击和暴力破解，而密钥对则提供了远超密码强度的安全保障。生成SSH密钥对的标准流程是在本地客户端运行`ssh-keygen`命令，按照提示创建一个密钥对（通常位于`~/.ssh/id_rsa`和`~/.ssh/id_rsa.pub`）。随后，需要将公钥（`.pub`文件的内容）追加到服务器上目标用户家目录下的`~/.ssh/authorized_keys`文件中。一旦配置完成，客户端连接时便会自动使用私钥进行身份验证，无需输入密码[29][101]。

在启用了密钥认证之后，下一步是彻底禁用密码认证本身。这可以通过编辑服务器上的SSH守护进程配置文件`/etc/ssh/sshd_config`来实现。找到`PasswordAuthentication`这一行，将其值修改为`no`[97][130]。执行此操作前务必确保新的密钥认证方式已经测试无误，因为一旦禁用，密码登录将不再可用，可能会导致用户被锁在系统之外[29]。需要注意的是，`PasswordAuthentication no`有时可能因为配置文件中的其他指令（如`Match`块）而未能生效，这是排查问题时需要关注的一个常见陷阱[113][119][129]。

另一个重要的安全措施是禁止root用户通过SSH直接登录。让root账户直接暴露在互联网上是一个巨大的安全隐患，因为攻击者会将root作为首要攻击目标。最佳做法是创建一个普通的用户账户用于日常登录，当需要执行管理任务时，再通过`sudo`命令获取必要的权

限 [28] [31]。在`sshd_config`中，可以通过设置`PermitRootLogin prohibit-password`来实现这一目标 [67]。这个选项既禁止了root的密码登录，又保留了通过密钥登录的可能性，为紧急情况下的运维提供了一条安全通道，是一种兼顾安全与便利的推荐配置 [68]。

为了进一步提升安全性，可以结合使用防火墙和入侵检测系统。UFW（Uncomplicated Firewall）是Ubuntu等系统上一个易于使用的防火墙前端，可以通过简单的命令限制对SSH端口（默认为22）的访问，例如只允许特定IP地址段连接 [30]。Fail2ban则是一个主动防御工具，它会监控系统的日志文件（如`/var/log/auth.log`），一旦发现针对SSH服务的多次连续登录失败尝试，就会通过底层的防火墙（如iptables）自动封禁发起攻击的IP地址一段时间 [32] [65]。将UFW和Fail2ban结合使用，可以形成一道有效的防御网，显著降低暴力破解的成功率 [82]。

对于追求极致安全的企业环境，还可以考虑部署更高级的访问控制技术。SELinux（Security-Enhanced Linux）和AppArmor是两种主流的强制访问控制（MAC）框架，它们提供了比传统自主访问控制（DAC）更为严格的权限模型 [114]。通过为SSH守护进程（`sshd`）定义精细的安全策略，可以限制其能够执行的操作，即使SSH服务本身被攻破，攻击者也难以获得更高的系统权限。此外，更改SSH的默认监听端口也是一个简单有效的辅助措施，虽然它不能从根本上解决问题，但可以过滤掉大量自动化扫描脚本，减少不必要的登录日志记录。

安全措施	配置项/工具	描述与最佳实践
认证方式	`PubkeyAuthentication yes`	强制使用SSH密钥而非密码进行认证 [27] [101]。
禁用密码登录	`PasswordAuthentication no`	在确认密钥认证可用后，彻底关闭密码认证以防御暴力破解 [97] [130]。
禁止root登录	`PermitRootLogin prohibit-password`	禁止root用户密码登录，降低高权限账户的攻击面 [67] [68]。
防火墙	UFW	限制对SSH端口的访问，仅允许可信IP [30] [32]。
入侵检测	Fail2ban	自动检测并封禁频繁尝试登录失败的IP地址 [65] [82]。
高级访问控制	SELinux / AppArmor	实施强制访问控制，限制`sshd`进程的权限，实现纵深防御 [114]。

敏感数据与模型资产保护策略

在深度学习项目中，训练数据和最终训练出的模型本身就是极具价值的知识产权资产。保护这些资产免受泄露、篡改和窃取，是贯穿整个ML生命周期的重要安全议题。本章节将

围绕数据与模型的静态保护、动态加载安全以及服务化部署安全三个方面，提供一套全面的保护策略。

首先，对静态存储的敏感数据和模型文件进行加密是保护资产的第一道物理屏障。根据加密范围的不同，主要分为全盘加密和文件级加密。LUKS（Linux Unified Key Setup）是Linux系统上实现全盘加密的行业标准，它在磁盘分区层面进行透明加密。部署新系统时启用LUKS，可以确保服务器关机后所有数据（包括硬盘、SSD上的敏感材料）都处于加密状态，有效防止通过物理介质窃取数据的攻击 87。然而，在某些场景下，用户可能希望部分数据明文存储以提高访问性能，此时文件级加密方案如`fscrypt`则更为适用。`fscrypt`直接集成在支持它的文件系统（如ext4, F2FS）中，可以对指定的目录树进行加密，而其子目录和文件在访问时会自动解密 86 93。因此，对于存放个人研究数据的`/home/user/datasets`目录，可以单独启用`fscrypt`加密，而系统或其他非敏感数据则保持明文状态。

其次，安全地处理和加载模型文件是防范代码注入攻击的关键环节。历史上曾发生过严重的漏洞，例如CVE-2025-32434，该漏洞发现在PyTorch的`torch.load()`函数中，即使在被认为相对安全的`weights_only=True`模式下，仍然存在远程代码执行的风险 35 37。这一事件打破了长期以来“只加载权重是安全的”这一普遍认知，凸显了加载任何来自不可信来源的模型文件都可能带来的巨大风险。因此，必须遵循以下最佳实践：永远不要加载未经验证的第三方模型；如果必须使用外部模型，应从官方渠道下载，并通过校验SHA256等哈希值来确保其完整性；在隔离的沙箱环境中先加载和验证模型，确认其行为符合预期后再投入使用；并且，始终将PyTorch升级到最新的安全版本，及时修补已知漏洞 37。

在开发和调试过程中，经常需要存储数据库密码、API密钥等敏感凭据。一种广泛使用的方法是将这些信息存储在环境变量中，而不是硬编码在代码或配置文件里 45。这种方法的优点在于可以将代码仓库与敏感凭据分离。然而，环境变量并非绝对安全，它们可能通过进程列表（`ps`命令）或调试工具被泄露 16。因此，必须采取严格的防护措施。对于本地开发，通常使用`.env`文件来管理环境变量，并通过`.gitignore`文件确保`.env`不会被意外提交到Git代码仓库 92 118。GitHub的安全报告指出，每年都有大量代码库因`.env`文件被意外公开而造成严重安全漏洞 118。对于生产环境，更安全的做法是使用专门的密钥管理服务，如HashiCorp Vault、AWS Secrets Manager或Google Secret Manager，在应用启动时动态地、安全地获取这些敏感信息，而不是将其持久化存储在环境中 91 104105。

最后，当模型被部署为线上推理服务时，其安全性同样不容忽视。像TorchServe这样的专用模型服务框架，提供了多种内置安全功能 41。首先，应强制启用身份验证机制，确保只有经过授权的应用程序才能向模型发送预测请求。其次，对于传输中的数据，应使用HTTPS/TLS进行加密，防止中间人攻击窃听或篡改请求和响应。此外，TorchServe还支持

通过KMS（密钥管理服务）对存储在S3等对象存储中的模型进行服务端加密，为静态存储的模型提供额外保护 [41]。在容器化的部署环境中，还应结合使用最小权限原则，例如使用`seccomp`或`AppArmor`等工具限制容器内进程的系统调用能力，进一步缩小潜在的攻击面 [33]。

第三方依赖库安全：防范软件供应链攻击

现代软件开发严重依赖于开源社区贡献的第三方库，Python的包索引（PyPI）是这一生态系统的中心枢纽。然而，这种便捷性也带来了新的安全挑战，其中最严峻的便是软件供应链攻击。攻击者通过污染合法的软件包，将恶意代码传递给下游的依赖项目，从而间接攻击广大用户。本章节将剖析此类攻击的原理、危害，并提供一套系统性的防护策略，旨在为PyTorch项目构建一条坚固的依赖安全防线。

软件供应链攻击的危害是深远且广泛的。一个典型的例子是2022年12月发生的`torchtriton`恶意依赖事件 [42]。攻击者利用了PyPI上包名相似性（typosquatting）的漏洞，上传了一个名为`torchtriton`的恶意包。由于PyTorch的夜间构建版本恰好将其作为一个依赖项，大量用户在不知情的情况下通过`pip install`安装了这个被植入的恶意软件。该恶意软件的功能是从受害者的系统中窃取敏感文件（如SSH密钥、Git配置），并通过加密DNS查询将数据外传至攻击者控制的域名 [42]。这次事件清晰地展示了，即便是顶级的人工智能框架，也可能成为供应链攻击的受害者。另一类更具隐蔽性的攻击是工作流注入，攻击者通过入侵项目的CI/CD系统（如GitHub Actions），在其发布流程中植入恶意代码，从而污染发布的Python包本身 [33]。

面对如此严峻的威胁，单一的防御手段是远远不够的，必须建立一个多层次、纵深的防御体系。首要的基石是严格锁定依赖版本。开发者不应在`requirements.in`或`pyproject.toml`中使用宽松的版本约束（如`requests>=2.0.0`），而应固定到具体的版本号（如`requests==2.28.1`）。这样做的目的是保证每次构建的环境都是一致的，避免因上游库的意外更新引入破坏性变更或隐藏的漏洞。

在此基础上，使用带哈希校验的锁文件是提升安全性的关键一步。推荐使用`pip-tools`等工具，它可以根据`requirements.in`文件生成一个包含所有直接和间接依赖及其确切版本的锁文件（`requirements.txt`），并在其中为每个包附加其SHA256哈希值 [33]。当安装时加上`--require-hashes`参数，`pip`会严格校验每个下载下来的包是否与锁文件中记录的哈希值完全一致。任何微小的差异，无论是依赖劫持还是网络传输过程中的篡改，都会导致安装失败。这是目前在Python生态中确保依赖完整性的最强有力手段。

信任源是第三个重要环节。默认情况下，`pip`会从公共的PyPI索引安装包。为了增加一层控制，企业或组织可以设置一个内部的PyPI镜像。所有对外的包安装请求都先指向这个内部镜像。内部镜像可以配置为只缓存那些已经被人工审核过的、来自可信源的包。新上传的包或未知来源的包会被拦截，从而阻止潜在的恶意软件进入开发和生产环境 [33]。这种策略不仅提升了安全性，也改善了在内网环境下的安装速度和稳定性。

最后，将安全检查融入持续集成/持续交付流程是实现自动化防护的有效途径。CI/CD流水线中应集成软件组成分析（SCA）工具，这些工具能够自动扫描项目依赖，识别出已知漏洞（通过CVE数据库匹配）和可疑的依赖项。同时，可以建立一个“黑名单”，明确禁止安装那些已被证实存在问题的包（如`torchtriton`）。此外，可以利用CI/CD系统的上下文来确保软件包是由可信的CI系统本身构建和签名的，而不是由本地开发者的机器偶然触发发布，这有助于防止开发者主机被感染后产生的“毒丸”包进入生产环境 [33]。

防护策略	实施工具/方法	作用与优势
锁定依赖版本	`requirements.txt`, `pyproject.toml`	保证环境的可复现性，避免因上游更新引入问题 [33]。
使用哈希校验	`pip-compile --generate-hashes`, `--require-hashes`	确保安装的每个依赖都未经篡改，从根本上防御依赖劫持 [33]。
使用可信源	内部PyPI镜像	对外部依赖进行审查和缓存，阻断恶意包进入内部网络 [33]。
CI/CD安全检查	SCA工具, 黑名单机制	自动化扫描已知漏洞和恶意依赖，实现实时防御 [33]。

综合工作流与安全实践指南

将前述的Linux操作、PyTorch开发实践与三大安全主题融会贯通，形成一个连贯、高效且安全的深度学习工作流，是本手册的最终目标。本章旨在提供一个贯穿项目始终的综合指南，强调安全意识应渗透到每一个环节，成为一种习惯而非事后补救。

一个理想的PyTorch项目启动流程始于安全的环境初始化。在服务器上创建新项目前，应首先为其创建一个独立的用户账户，并确保该账户通过SSH密钥进行认证，且root登录已被禁用 [27] [28]。接着，在该用户主目录下克隆代码仓库，并立即配置`.gitignore`文件，确保`.env`、虚拟环境目录（如`.venv`或`envs/`）和本地配置文件不会被提交 [92]。随后，使用`conda`或`venv`创建一个新的Python虚拟环境，并激活它。此时，所有的`pip`或`conda`安装命令都将在隔离的环境中进行，避免污染系统Python环境。在安装PyTorch之前，应先通过`nvidia-smi`检查服务器的GPU驱动和CUDA版本，确保选择正确的PyTorch安装包 [9] [126]。

在项目开发阶段，安全实践体现在代码编写和数据处理的方方面面。编写数据预处理脚本时，应避免在代码中硬编码敏感信息，而是通过环境变量或配置文件来读取。在加载数据时，应始终验证数据源的合法性。当需要加载外部模型时，必须遵循“绝不信任，始终验证”的原则。这意味着不仅要校验模型文件的哈希值，最好还能在隔离的沙箱环境中进行初步测试 [33]。在整个开发过程中，定期运行`pip check`来检查依赖项之间的版本冲突，以及运行SCA工具扫描依赖树，都是保障项目健康的重要步骤。

模型训练和监控是资源密集型活动，同时也伴随着安全风险。启动训练任务时，应明确指定所需的GPU数量和资源，尤其是在共享集群上。在训练过程中，应养成定期使用`nvidia-smi`检查GPU状态的习惯，警惕异常的高负载或显存泄漏 [4]。当训练结束需要保存模型时，应考虑对模型文件进行加密存储，特别是当模型本身包含敏感信息或其结构可以反推出商业秘密时。可以使用`fscrypt`等工具对存放模型的目录进行加密 [86]。

项目部署是安全的最后一道关口。当模型被封装成API服务（如使用TorchServe）对外提供服务时，必须启用强身份验证机制，防止未经授权的访问 [41]。服务间的通信应全部使用TLS加密。在基础设施层面，应遵循最小权限原则，为运行服务的容器配置严格的`seccomp`或`AppArmor`安全策略，限制其系统调用能力，从而降低容器逃逸的风险 [33]。同时，通过防火墙规则严格控制服务的入站和出站流量，仅允许必要的网络连接。

综上所述，构建一个安全可靠的PyTorch开发环境，是一个系统工程，它要求开发者不仅精通技术工具，更要具备强烈的安全意识。通过将SSH加固、数据加密、依赖锁定、模型验证等一系列安全措施内建到日常工作流的每个环节，开发者可以在享受Linux和PyTorch强大能力的同时，最大限度地降低安全风险，保护宝贵的数据和智力资产。这份手册所提供的知识和实践指南，正是为了帮助每一位使用者建立起这样一套纵深防御体系。

参考文献

1. Something like "top" to monitor the gpu? https://forums.developer.nvidia.com/t/something-like-top-to-monitor-the-gpu/20714
2. A top-like utility for monitoring CUDA activity on a GPU https://stackoverflow.com/questions/8223811/a-top-like-utility-for-monitoring-cuda-activity-on-a-gpu
3. Monitoring GPU utilization for Deep Learning https://www.digitalocean.com/community/tutorials/monitoring-gpu-utilization-in-real-time

4. How to see what process is using GPU? https://askubuntu.com/questions/1396706/how-to-see-what-process-is-using-gpu
5. CUDA C++ Best Practices Guide 13.1 documentation https://docs.nvidia.com/cuda/cuda-c-best-practices-guide/
6. Performance Tuning Guide https://docs.pytorch.org/tutorials/recipes/recipes/tuning_guide.html
7. Welcome to PyTorch Tutorials https://docs.pytorch.org/tutorials/index.html
8. Training with PyTorch https://docs.pytorch.org/tutorials/beginner/introyt/trainingyt.html
9. Get Started https://pytorch.org/get-started/locally/
10. Start via Cloud Partners https://pytorch.org/get-started/cloud-partners/
11. Getting Started with Distributed Data Parallel https://docs.pytorch.org/tutorials/intermediate/ddp_tutorial.html
12. Can someone provide the steps to upload my ... https://discuss.pytorch.org/t/can-someone-provide-the-steps-to-upload-my-pytorch-project-on-google-colab/101324
13. Deep Learning Energy Measurement and Optimization https://pytorch.org/blog/zeus/
14. Reinforcement Learning (DQN) Tutorial https://docs.pytorch.org/tutorials/intermediate/reinforcement_q_learning.html
15. Efficient PyTorch I/O library for Large Datasets, Many Files, ... https://pytorch.org/blog/efficient-pytorch-io-library-for-large-datasets-many-files-many-gpus/
16. Linux - securing environment variables https://stackoverflow.com/questions/4129631/linux-securing-environment-variables
17. Protecting Confidentiality, Privacy and Integrity in ... https://arxiv.org/html/2412.08534v2
18. How to ensure data security in machine learning model ... https://www.tencentcloud.com/techpedia/130725
19. (PDF) Securing Linux Cloud Environments: Privacy-Aware ... https://www.researchgate.net/publication/388908983_Securing_Linux_Cloud_Environments_Privacy-Aware_Federated_Learning_Framework_for_Advanced_Malware_Detection_in_Linux_Clouds
20. Pass Sensitive Data Using Linux Environment Variables https://docs-composer.zendesk.com/hc/en-us/articles/25438517927053-Pass-Sensitive-Data-Using-Linux-Environment-Variables
21. A more secure way to handle secrets in OpenShift https://developers.redhat.com/articles/2025/10/01/secure-way-handle-secrets-openshift

22. Machine Learning with Confidential Computing https://dl.acm.org/doi/10.1145/3670007
23. Using SSH - Imperial RCS User Guide https://icl-rcs-user-guide.readthedocs.io/en/latest/hpc/getting-started/using-ssh/
24. 3. Cluster User Guide — NVIDIA DGX Cloud Slurm ... https://docs.nvidia.com/dgx-cloud/slurm/latest/cluster-user-guide.html
25. Design and Operation of Shared Machine Learning Clusters ... https://cse.hkust.edu.hk/~kaichen/papers/tacc-asplos25.pdf
26. How to Create an SSH Key in Linux: Easy Step-by- ... https://www.digitalocean.com/community/tutorials/how-to-configure-ssh-key-based-authentication-on-a-linux-server
27. Chapter 5. Using secure communications between two ... https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html/configuring_basic_system_settings/assembly_using-secure-communications-between-two-systems-with-openssh_configuring-basic-system-settings
28. 7 Quick and Easy Ways to Secure SSH on a Linux Server https://xtom.com/blog/secure-ssh-linux-server-guide/
29. A Guide To Securing Your Remote Access Using SSH Keys https://infosecwriteups.com/a-guide-to-securing-your-remote-access-using-ssh-keys-84b48097f3bf
30. How to Set Up a Firewall with UFW on Ubuntu https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-with-ufw-on-ubuntu
31. Set Up and Configure an Application Server on Ubuntu 24.04 https://www.digitalocean.com/community/tutorials/set-up-configure-application-server-ubuntu-24-04
32. Ubuntu Fundamentals: fish https://dev.to/devopsfundamentals/ubuntu-fundamentals-fish-3m00
33. PyTorch Supply Chain Attack: Dev Guardrails https://www.cybersrely.com/pytorch-supply-chain-attack/
34. (Part 1) Pre-training with float8 — torchao 0.15 ... https://docs.pytorch.org/ao/stable/pretraining.html
35. CVE-2025-32434： PyTorch torch.load () 函数漏洞可致远程 ... https://www.anquanke.com/post/id/306706
36. 1 High-Security vulnerability found in Black Duck scan https://dev-discuss.pytorch.org/t/1-high-security-vulnerability-found-in-black-duck-scan/2509
37. Critical RCE in PyTorch: Upgrade to 2.6.0 now https://www.linkedin.com/posts/ammarmohanna_critical-pytorch-vulnerability-cve-2025-32434-activity-7320745554622226432-vuSd

38. Statistics and Machine Learning in Python https://hal.science/hal-03038776/file/StatisticsMachineLearningPython.pdf
39. Ultimate Python Guide (2024) | PDF https://www.scribd.com/document/754260195/Ultimate-Python-Guide-2024
40. PDF https://stable-baselines3.readthedocs.io/_/downloads/en/master/pdf/
41. ANNOUNCEMENT: Security Changes https://docs.pytorch.org/serve/README.html
42. Compromised PyTorch-nightly dependency chain between ... https://pytorch.org/blog/compromised-nightly-dependency/
43. DGX Memory Not Released After Stopping Ollama ... https://forums.developer.nvidia.com/t/dgx-memory-not-released-after-stopping-ollama-openwebui-fixed/348353
44. Scaling FlexPod for GPU Intensive Applications https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_gpu_aiml.html
45. Is it secure to store passwords as environment variables ... https://stackoverflow.com/questions/12461484/is-it-secure-to-store-passwords-as-environment-variables-rather-than-as-plain-t
46. Doing much better than your .env file https://dev.to/dangtony98/doing-much-better-than-your-env-file-46bg
47. 7 ways to improve security of your machine learning ... https://aws.amazon.com/blogs/security/7-ways-to-improve-security-of-your-machine-learning-workflows/
48. Smart Patching with Cron Jobs: An Ops-Centric Perspective https://www.researchgate.net/publication/393971910_Smart_Patching_with_Cron_Jobs_An_Ops-Centric_Perspective
49. Practical Migration from x86 to Linux on IBM Z https://www.redbooks.ibm.com/redbooks/pdfs/sg248217.pdf
50. VMware Tanzu Greenplum 7 - Broadcom Tech Docs https://techdocs.broadcom.com/content/dam/broadcom/techdocs/us/en/pdf/vmware-tanzu/data-solutions/tanzu-greenplum/7-1/vmware-tanzu-greenplum-7-1.pdf
51. https://packages.debian.org/trixie/amd64/allpackag... https://packages.debian.org/trixie/amd64/allpackages?format=txt.gz
52. Hardening Linux | PDF | System Software https://www.scribd.com/document/442925846/Hardening-Linux
53. Linux 6.7.2 https://www.kernel.org/pub/linux/kernel/v6.x/ChangeLog-6.7.2
54. Information Security Applications https://link.springer.com/content/pdf/10.1007/978-3-030-65299-9.pdf
55. Is it secure way to store private values in .env file? https://stackoverflow.com/questions/60360298/is-it-secure-way-to-store-private-values-in-env-file

56. Python FileNotFoundError when reading a file in the same ... https://stackoverflow.com/questions/77790161/python-filenotfounderror-when-reading-a-file-in-the-same-directory-using-pathlib
57. Statistics Machine Learning Python Draft | PDF https://www.scribd.com/presentation/451070502/StatisticsMachineLearningPythonDraft-pptx
58. Profile for PyImageSearch https://www.linknovate.com/affiliation/pyimagesearch-82619185/all/?query=default%20object%20size
59. Deploying and Installing SUSE AI https://documentation.suse.com/suse-ai/1.0/pdf/AI-deployment_en.pdf
60. 2015年1月4日随笔档案- paulwong http://www.blogjava.net/paulwong/archive/2015/01/04.html
61. 2019年7月30日随笔档案- paulwong http://www.blogjava.net/paulwong/archive/2019/07/30.html
62. Securing Linux Cloud Environments: Privacy-Aware ... https://ieeexplore.ieee.org/iel8/6287639/10820123/10879481.pdf
63. On resource consumption of machine learning in ... https://www.sciencedirect.com/science/article/pii/S1389128625005675
64. torch.compile Troubleshooting https://docs.pytorch.org/docs/stable/torch.compiler_troubleshooting.html
65. How To Protect SSH with Fail2Ban on Ubuntu 22.04 https://www.digitalocean.com/community/tutorials/how-to-protect-ssh-with-fail2ban-on-ubuntu-22-04
66. potential ufw and fail2ban conflicts https://askubuntu.com/questions/54771/potential-ufw-and-fail2ban-conflicts
67. ssh 使用root 账户无法登录_permitrootlogin prohibit ... https://blog.csdn.net/lyndon_li/article/details/120387260
68. SSH安全配置：PermitRootLogin prohibit-password详解 https://comate.baidu.com/zh/page/djlsqn0ek14
69. Linux 允许root用户远程登陆 https://www.cnblogs.com/zoneofmine/p/10934239.html
70. permitrootlogin prohibit-password - 腾讯云开发者社区 https://cloud.tencent.com/developer/information/%23permitrootlogin%20prohibit-password-article
71. Packages being worked on, organized by activity https://www.debian.org/devel/wnpp/being_packaged_byactivity
72. Brew Formula | PDF | Command Line Interface https://www.scribd.com/document/551715438/Brew-Formula
73. Multiprocessing best practices https://docs.pytorch.org/docs/stable/notes/multiprocessing.html

74. Optimizing CPU Performance on Intel® Xeon ... https://docs.pytorch.org/tutorials/recipes/xeon_run_cpu.html
75. Ahead-Of-Time Compilation for Torch.Export-ed Models https://docs.pytorch.org/docs/main/user_guide/torch_compiler/torch.compiler_aot_inductor.html
76. Distributed communication package - torch.distributed https://docs.pytorch.org/docs/stable/distributed.html
77. HIP (ROCm) semantics — PyTorch 2.9 documentation https://docs.pytorch.org/docs/stable/notes/hip.html
78. torch.utils.cpp_extension https://docs.pytorch.org/docs/stable/cpp_extension.html
79. ISC2 CISSP Study Guide | PDF | Computer Security https://www.scribd.com/document/840647875/ISC2-CISSP-Study-Guide-1
80. 9.3 Release Notes | Red Hat Enterprise Linux | 9 https://docs.redhat.com/de/documentation/red_hat_enterprise_linux/9/html-single/9.3_release_notes/index
81. PyTorch 2 DistributedDataParallel - distributed https://discuss.pytorch.org/t/pytorch-2-distributeddataparallel/180719
82. 如何用UFW + Fail2ban 搭一套“多层次安全防护” https://www.a5idc.com/helpcontent/836.html
83. Prospective packages https://www.debian.org/devel/wnpp/prospective
84. Пакеты, работа над которыми начата https://www.debian.org/devel/wnpp/being_packaged.ru.html
85. Navy Removal Scout 800 Pink Pill Assasin Expo Van ... https://www.scribd.com/document/531005187/70048773907-navy-removal-scout-800-pink-pill-assasin-expo-van-travel-bothell-punishment-shred-norelco-district-ditch-required-anyhow
86. Filesystem-level encryption (fscrypt) https://www.kernel.org/doc/html/v5.8/filesystems/fscrypt.html
87. LUKS, fscrypt, and EnCFS: A Comparison https://www.linkedin.com/posts/dr-murty-chandrapati-ph-d-60449013_luks-fscrypt-and-encfs-a-comparison-of-activity-7226045586628595712-wF8G
88. The definitive guide to Kernel vs. User Space Cryptography ... https://www.wolfssl.com/the-definitive-guide-to-kernel-vs-user-space-cryptography-on-windows-or-linux/
89. Prevent access to .env file in Apache server? - laravel https://stackoverflow.com/questions/29963586/prevent-access-to-env-file-in-apache-server
90. https://packages.debian.org/ja/bullseye/allpackage... https://packages.debian.org/ja/bullseye/allpackages?format=txt.gz&MuVg9NH6rk=36b0RLoF
91. Arif Alam's Post https://www.linkedin.com/posts/iamarifalam_%F0%9D%97%97%F0%9D%97%AE%F0%9D%98%81%F0%9D%97%AE

-
%F0%9D%97%98%F0%9D%97%BB%F0%9D%97%B4%F0%9D%97%B6%F0%9D%97%BB%F0%9D%97%B2%F0%9D%97%B2%F0%9D%97%BF%F0%9D%97%B6%F0%9D%97%BB%F0%9D%97%B4-
%F0%9D%97%A5%F0%9D%97%BC%F0%9D%97%AE%F0%9D%97%B1%F0%9D%97%BA%F0%9D%97%AE%F0%9D%97%BD-activity-7284411689838460928-1Shn

92. 终极python-dotenv配置安全指南：10个保护环境变量的关键 ... https://blog.csdn.net/gitblog_00809/article/details/152017268
93. Filesystem-level encryption (fscrypt) https://www.kernel.org/doc/html/v4.19/filesystems/fscrypt.html
94. https://packages.debian.org/fr/bullseye/arm64/allp... https://packages.debian.org/fr/bullseye/arm64/allpackages?format=txt.gz
95. Compare Packages Between Distributions https://distrowatch.com/dwres.php?resource=compare-packages&firstlist=gentoo&secondlist=devuan&firstversions=0&secondversions=0&showall=yes
96. Ubuntu 24.04系统下SSH服务的安装与配置指南 https://comate.baidu.com/zh/page/87kys692bec
97. How do I force SSH to only allow users with a key to log in? https://askubuntu.com/questions/346857/how-do-i-force-ssh-to-only-allow-users-with-a-key-to-log-in
98. PermitRootLogin - sshd_config - Multiple versions, No impact https://discourse.ubuntu.com/t/permitrootlogin-sshd-config-multiple-versions-no-impact/72347
99. Ubuntu 24.04 开启root远程登录完全指南：密码与密钥双重认证 https://juejin.cn/post/7532777221196808207
100. PasswordAuthentication no, but I can still login by password https://unix.stackexchange.com/questions/727492/passwordauthentication-no-but-i-can-still-login-by-password
101. How to Set Up SSH Keys on Ubuntu https://www.digitalocean.com/community/tutorials/how-to-set-up-ssh-keys-on-ubuntu-22-04
102. Ubuntu 24.04系统下SSH远程访问服务的部署与安全配置指南 https://comate.baidu.com/zh/page/702wnt2gc3s
103. State of symbolic shapes branch - Page 4 - compiler https://dev-discuss.pytorch.org/t/state-of-symbolic-shapes-branch/777?page=4
104. Best practices to store credentials in your Python script https://stackoverflow.com/questions/62113332/best-practices-to-store-credentials-in-your-python-script

105. Get a Secrets Manager secret value using Python with ... https://docs.aws.amazon.com/secretsmanager/latest/userguide/retrieving-secrets_cache-python.html

106. How to restart Data Engineering in 2025: A step-by-step plan https://www.linkedin.com/posts/arabinda-pani-56783320_sql-tutorial-geeksforgeeks-activity-7349420285143498753-xifx

107. suse-sles-15-sp2-byos-v20220128-hvm-ssd-x86_64 ... https://publiccloudimagechangeinfo.suse.com/amazon/suse-sles-15-sp2-byos-v20220128-hvm-ssd-x86_64/package_changelogs.html

108. hw3_stats_google_1gram.txt https://www.cs.cmu.edu/~roni/11761/2017_fall_assignments/hw3_stats_google_1gram.txt

109. 333333 23135851162 the 13151942776 of 12997637966 https://www.cs.princeton.edu/courses/archive/spring25/cos226/assignments/autocomplete/files/words-333333.txt

110. profiles/use.local.desc · master · ti / gentoo · GitLab https://git.nju.edu.cn/ti/gentoo/-/blob/master/profiles/use.local.desc

111. Learning PyTorch with Examples https://docs.pytorch.org/tutorials/beginner/pytorch_with_examples.html

112. d2l-en.pdf https://d2l.ai/d2l-en.pdf

113. SSH - PasswordAuthentication no has no effect https://superuser.com/questions/1022637/ssh-passwordauthentication-no-has-no-effect

114. SELinux 与AppArmor 的配置，增强系统安全性 https://cloud.tencent.com.cn/developer/article/2613320?policyId=1003

115. suse-sles-sap-15-sp5-hardened-byos-v20240808-hvm-ssd ... https://publiccloudimagechangeinfo.suse.com/amazon/suse-sles-sap-15-sp5-hardened-byos-v20240808-hvm-ssd-x86_64/package_changelogs.html

116. PyTorch 2: Faster Machine Learning Through Dynamic ... https://www.researchgate.net/publication/380151019_PyTorch_2_Faster_Machine_Learning_Through_Dynamic_Python_Bytecode_Transformation_and_Graph_Compilation

117. Download Python source code: tuning_guide.py https://docs.pytorch.org/tutorials/_downloads/8c82db84c10318a94cbe213adb618139/tuning_guide.py

118. 保护敏感配置：python-dotenv安全最佳实践指南 https://blog.csdn.net/gitblog_00171/article/details/152012400

119. How to disable password authentication in SSH in ... https://askubuntu.com/questions/1522998/how-to-disable-password-authentication-in-ssh-in-ubuntu-24-04-new-socket-based

120. NFS best practice and implementation guide | TR-4067 https://www.netapp.com/media/10720-tr-4067.pdf
121. IBM Storage Scale: Big Data and Analytics Guide https://www.ibm.com/docs/de/STXKQY_BDA_SHR/pdf/scale_bda.pdf
122. Linux Shell Commands Compilation | PDF https://www.scribd.com/doc/124411164/24409062-Linux-Shell-Commands-Compilation
123. 333333 23135851162 the 13151942776 of 12997637966 ftp://ftp.cs.princeton.edu/pub/cs226/autocomplete/words-333333.txt
124. java - Environment specific config files - best practices? https://stackoverflow.com/questions/5232987/environment-specific-config-files-best-practices
125. NVIDIA TensorRT https://docs.nvidia.com/deeplearning/tensorrt/archives/tensorrt-825/pdf/TensorRT-Release-Notes.pdf
126. NVIDIA-SMI has failed because it couldn't communicate ... https://forums.developer.nvidia.com/t/nvidia-smi-has-failed-because-it-couldnt-communicate-with-the-nvidia-driver-make-sure-that-the-latest-nvidia-driver-is-installed-and-running/197141
127. PyTorch on XLA Devices https://docs.pytorch.org/xla/release/2.2/index.html
128. NVIDIA AI Enterprise User Guide https://docs.nvidia.com/ai-enterprise/5.1/user-guide/index.html
129. Why is SSH “PasswordAuthentication no” not working as ... https://superuser.com/questions/1924210/why-is-ssh-passwordauthentication-no-not-working-as-expected-on-ubuntu-still
130. etc/ssh/sshd_config的PasswordAuthentication ... https://blog.csdn.net/kfepiza/article/details/127781345
131. sshd_config — OpenSSH SSH daemon configuration file https://manpages.ubuntu.com/manpages/trusty/en/man5/sshd_config.5.html