

ollama 安装流程

- 1 #建议 docker 安装，比较方便
- 2 #1、拉取镜像
- 3 #docker hub 地址，选则自己需要的版本
- 4 #<https://hub.docker.com/r/ollama/ollama/tags>
- 5 #我安装的是最新版
- 6 `docker pull ollama/ollama:latest`
- 7
- 8 #2、启动 docker
- 9 `docker run -d --gpus=all -v ollama:/root/.ollama -p 11434:11434`
`--name ollama ollama/ollama`
- 10
- 11 #3、运行模型
- 12 #其中 `ollama run llama3` 为运行模型的命令，如果需要使用其他模型，可以参考
官方提供模型版本 <https://github.com/ollama/ollama>，将 llama3 替换掉即可
- 13 `docker exec -it ollama ollama run llama3`

Langchain 安装

```
1  conda create --name rag python=3.10
2  conda activate rag
3  pip install langchain
4  pip install streamlit
5  pip install chromadb
6  pip install pypdf
7  pip install langchain_community
8  pip install chromadb
```

核心代码

```
1  #核心代码主要包含两个模型，检索模型和生成模型，详见下文备注，模型地址为
   使用上面 ollama 启动的模型地址
2
3  from langchain.chains import RetrievalQA
4  from langchain.callbacks.streaming_stdout import
   StreamingStdOutCallbackHandler
5  from langchain.callbacks.manager import CallbackManager
6  from langchain_community.llms import Ollama
7  from langchain_community.embeddings.ollama import
   OllamaEmbeddings
8  from langchain_community.vectorstores import Chroma
9  from langchain.text_splitter import RecursiveCharacterTextSplitter
10 from langchain_community.document_loaders import PyPDFLoader
11 from langchain.prompts import PromptTemplate
12 from langchain.memory import ConversationBufferMemory
13 import streamlit as st
14 import os
15 import time
16
17 if not os.path.exists('files'):
18     os.mkdir('files')
19
20 if not os.path.exists('jj'):
21     os.mkdir('jj')
22
```

- 1 #执行这个命令运行即可启动服务
- 2 streamlit run app.py
- 3 #接着在浏览器访问服务地址
- 4 http://localhost:8501

上传 PDF 即可进行问答对话

PDF Chatbot

Upload your PDF



Drag and drop file here

Limit 200MB per file • PDF

Browse files



漏洞自动修复研究综述_徐同同.pdf 1.2MB



软件漏洞是什么



software漏洞是指一种安全漏洞，指的是在软件或系统的设计、编程或开发过程中存在的错误或者缺陷，使得攻击者能够利用这些漏洞来进行未授权的访问、修改、删除或其他非法操作。Software vulnerabilities是一种常见的安全风险，它们可以通过各种方式被exploited，例如：1. **注入攻击（Injection Attack）**：Attackers可以利用软件不安全地处理输入数据来获得控制。2. **SQL注射**：Attackers可以利用未正确处理的SQL请求来获取或修改数据。3. **cross-site scripting（XSS）**：Attackers可以利用未正确验证和编码的输入，例如在网页界面中执行JavaScript脚本。4. **Buffer overflow**：Attackers可以利用软件中的缓冲区溢出来执行恶意代码。software漏洞可以来自各种地方，包括：1. **软件开发过程中存在的错误** 2. **未正确更新或维护的软件** 3. **未使用安全特性** Software漏洞的修复非常重要，它们通常需要专门的工具和技术来进行发现、分析和修复。一些常见的漏洞扫描工具包括Nessus、OpenVAS等。为了防止软件漏洞，以下是几种有效策略：1. **遵守安全标准** 2. **定期更新和维护** 3. **使用安全编程实践** 4. **进行安全测试** 通过这些措施，可以帮助减少software漏洞的风险，并提高系统的安全性。