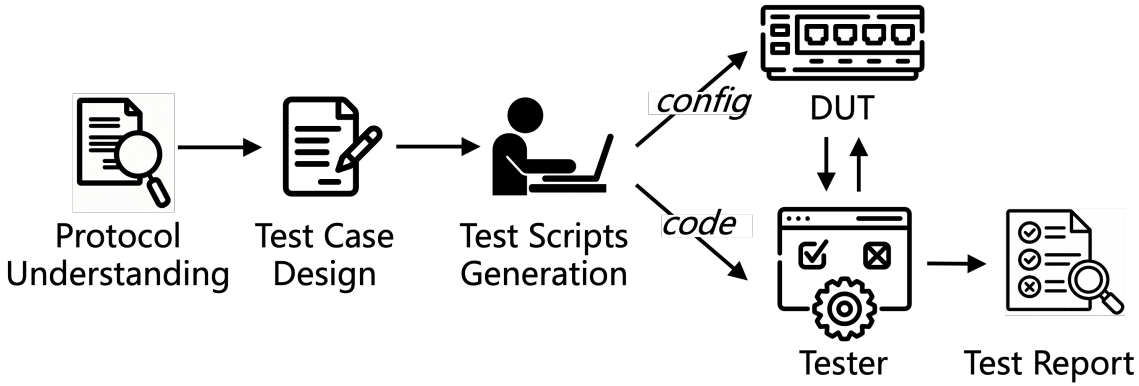
LLM Driven Automated Network Protocol Testing

IETF 125 - Hackathon

Yong Cui, Xiaohui Xie, **Yunze Wei,** Kaiwen Chi, **Chuyi Wang**

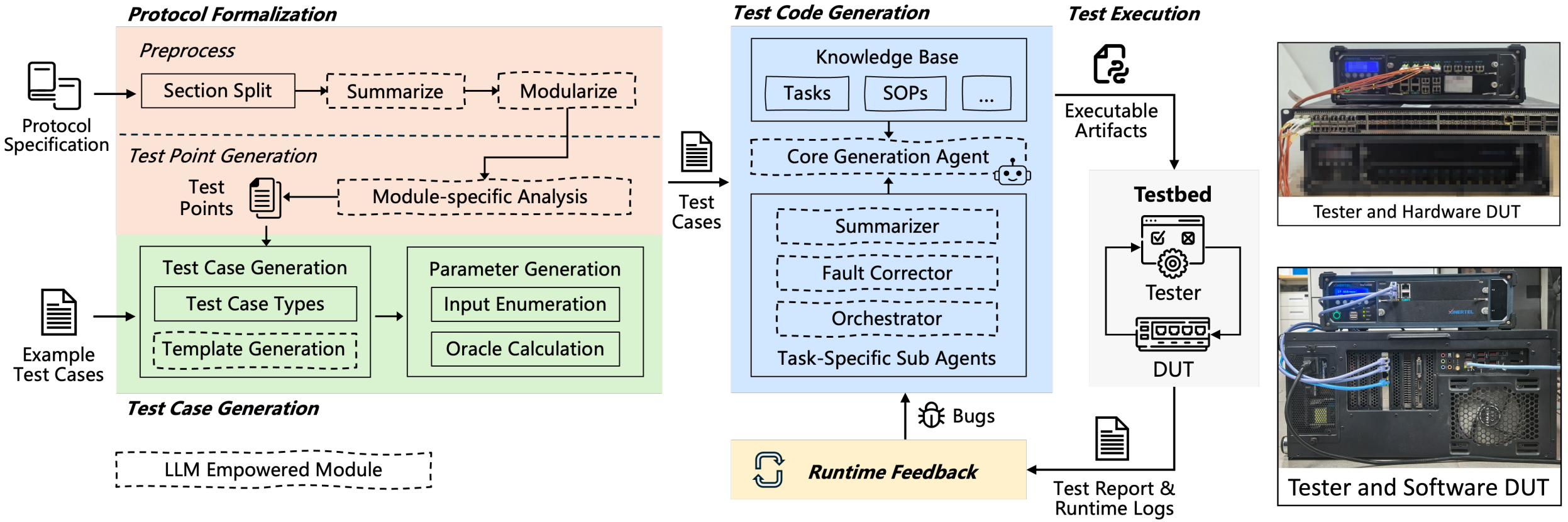
Tsinghua University

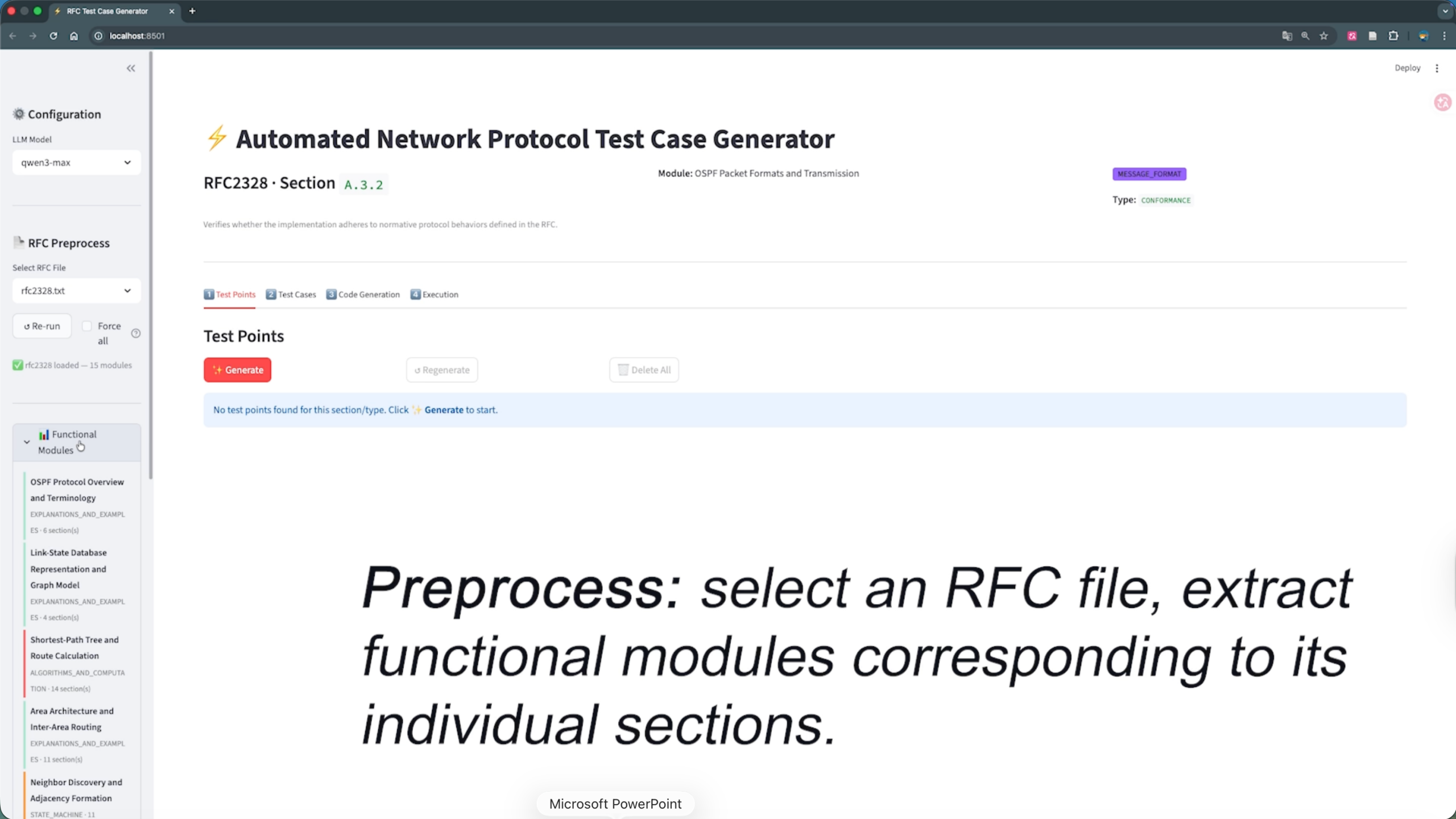
Background

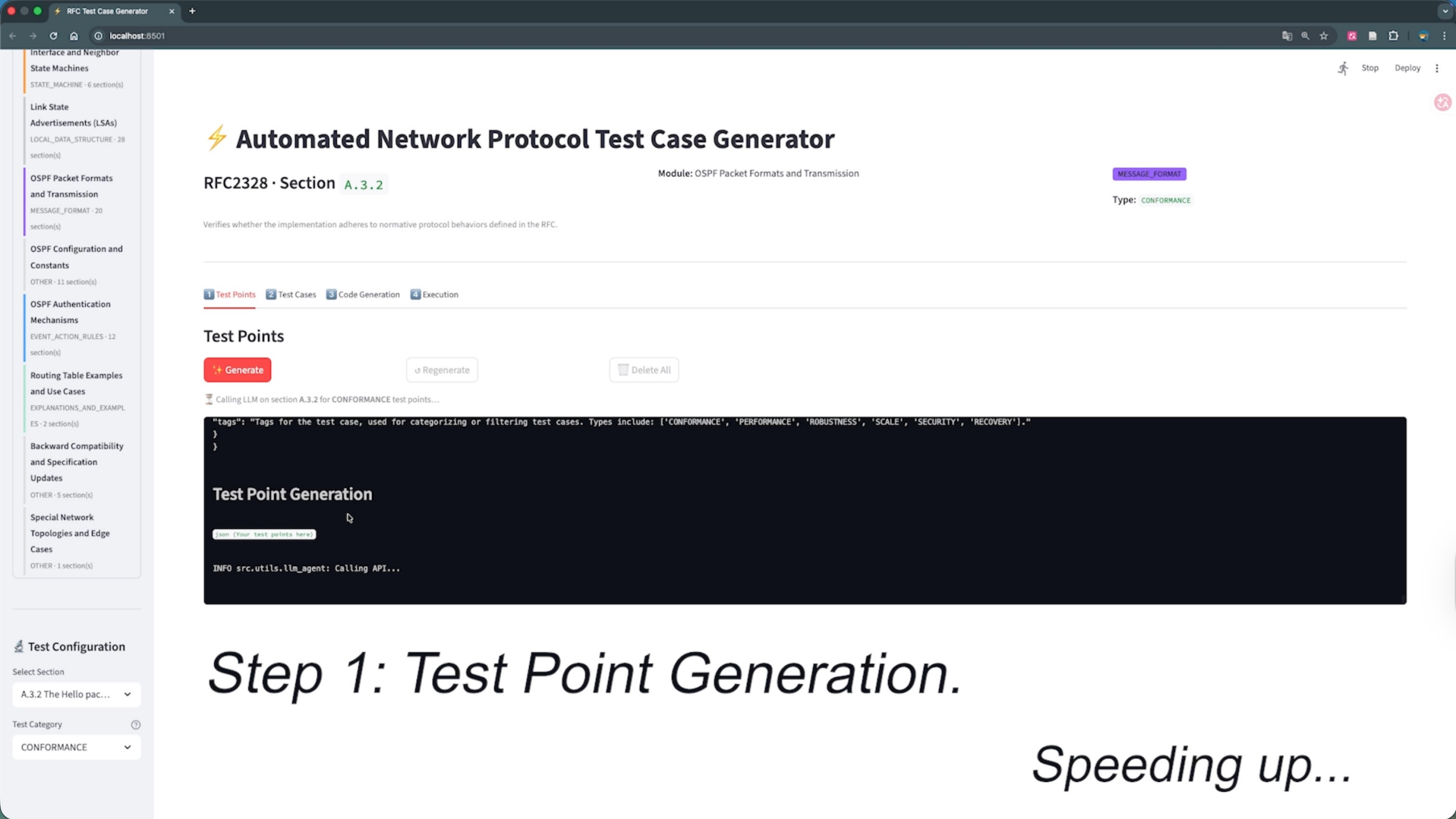


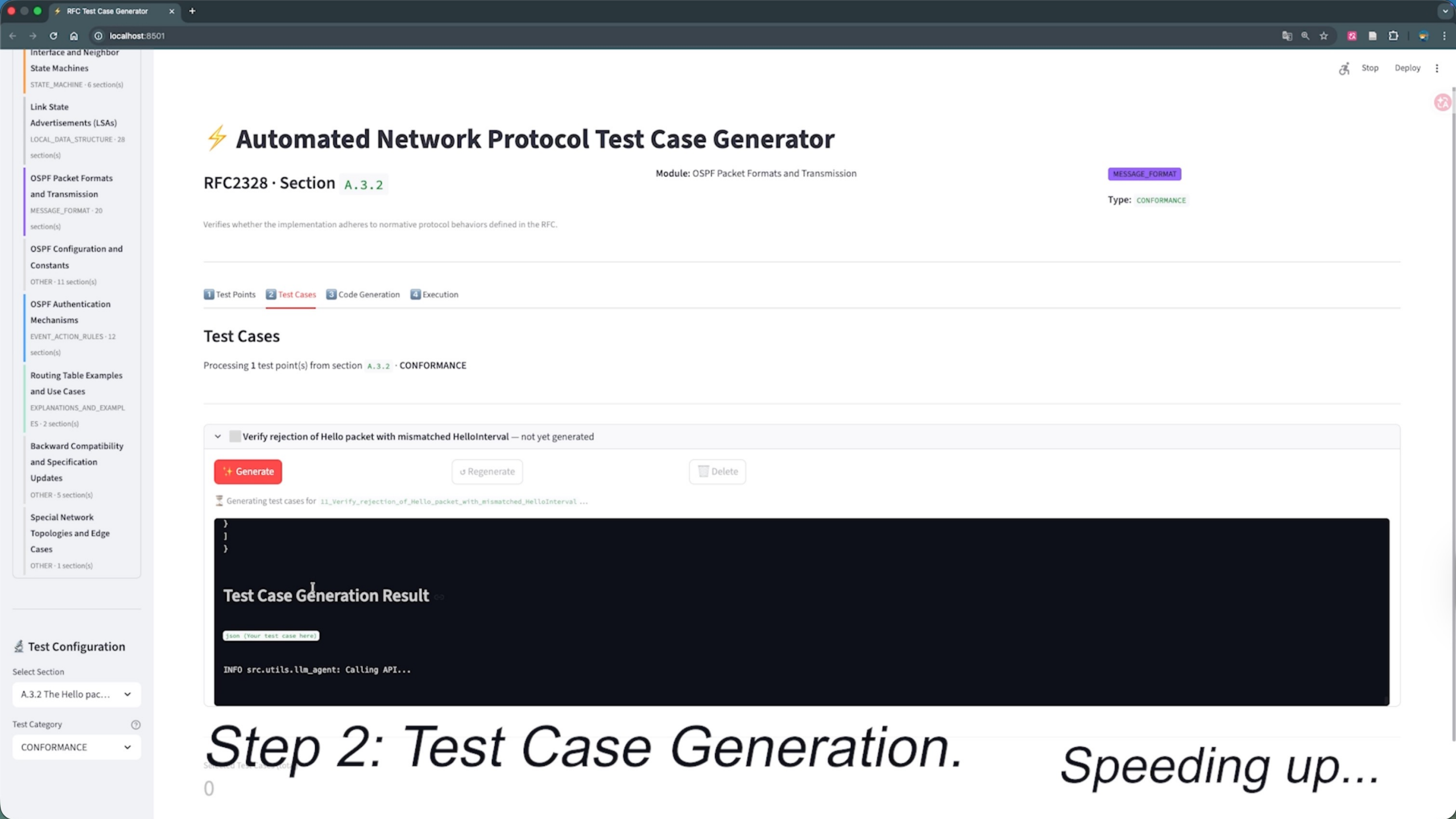
- Network Protocol Testing
  - Test protocol implementations to ensure conformance, performance, security, etc.
- Traditional Network Protocol Testing is Labor-Intensive
  - Engineers need to analyze RFCs documents, design test cases, convert them into executable scripts for both testers and devices under test (DUTs), execute the test and analyze reports
  - Drawbacks: **Low efficiency, Limited coverage**
  - Struggling to adapt to the rapid evolution of network protocols in new scenarios, such as Industrial Internet, Satellite Networks, and Modern DCNs.

Design and Implementation




RFC Test Case Generator
localhost:8501
Deploy
Configuration
LLM Model
qwen3-max
RFC Preprocess
Select RFC File
rfc2328.txt
Re-run
Force all
rfc2328 loaded — 15 modules
Functional Modules
OSPF Protocol Overview and Terminology
EXPLANATIONS_AND_EXAMPLES · 6 section(s)
Link-State Database Representation and Graph Model
EXPLANATIONS_AND_EXAMPLES · 4 section(s)
Shortest-Path Tree and Route Calculation
ALGORITHMS_AND_COMPUTATION · 14 section(s)
Area Architecture and Inter-Area Routing
EXPLANATIONS_AND_EXAMPLES · 11 section(s)
Neighbor Discovery and Adjacency Formation
STATE_MACHINE · 11
Automated Network Protocol Test Case Generator
RFC2328 · Section A.3.2
Module: OSPF Packet Formats and Transmission
MESSAGE_FORMAT
Type: CONFORMANCE
Verifies whether the implementation adheres to normative protocol behaviors defined in the RFC.
1 Test Points
2 Test Cases
3 Code Generation
4 Execution
Test Points
Generate
Regenerate
Delete All
No test points found for this section/type. Click ✨ Generate to start.
Preprocess: select an RFC file, extract functional modules corresponding to its individual sections.
Microsoft PowerPoint


RFC Test Case Generator
localhost:8501
Interface and Neighbor State Machines
STATE_MACHINE · 6 section(s)
Link State Advertisements (LSAs)
LOCAL_DATA_STRUCTURE · 28 section(s)
OSPF Packet Formats and Transmission
MESSAGE_FORMAT · 20 section(s)
OSPF Configuration and Constants
OTHER · 11 section(s)
OSPF Authentication Mechanisms
EVENT_ACTION_RULES · 12 section(s)
Routing Table Examples and Use Cases
EXPLANATIONS_AND_EXAMPLES · 2 section(s)
Backward Compatibility and Specification Updates
OTHER · 5 section(s)
Special Network Topologies and Edge Cases
OTHER · 1 section(s)
Test Configuration
Select Section
A.3.2 The Hello pac...
Test Category
CONFORMANCE
Stop
Deploy
Automated Network Protocol Test Case Generator
RFC2328 · Section A.3.2
Module: OSPF Packet Formats and Transmission
MESSAGE_FORMAT
Type: CONFORMANCE
Verifies whether the implementation adheres to normative protocol behaviors defined in the RFC.
1 Test Points
2 Test Cases
3 Code Generation
4 Execution
Test Points
Generate
Regenerate
Delete All
Calling LLM on section A.3.2 for CONFORMANCE test points...
"tags": "Tags for the test case, used for categorizing or filtering test cases. Types include: ['CONFORMANCE', 'PERFORMANCE', 'ROBUSTNESS', 'SCALE', 'SECURITY', 'RECOVERY']."
}
}
Test Point Generation
json (Your test points here)
INFO src.utils.llm_agent: Calling API...
Step 1: Test Point Generation.
Speeding up...


RFC Test Case Generator
localhost:8501
Interface and Neighbor State Machines
STATE_MACHINE · 6 section(s)
Link State Advertisements (LSAs)
LOCAL_DATA_STRUCTURE · 28 section(s)
OSPF Packet Formats and Transmission
MESSAGE_FORMAT · 20 section(s)
OSPF Configuration and Constants
OTHER · 11 section(s)
OSPF Authentication Mechanisms
EVENT_ACTION_RULES · 12 section(s)
Routing Table Examples and Use Cases
EXPLANATIONS_AND_EXAMPLES · 2 section(s)
Backward Compatibility and Specification Updates
OTHER · 5 section(s)
Special Network Topologies and Edge Cases
OTHER · 1 section(s)
Test Configuration
Select Section
A.3.2 The Hello pac...
Test Category
CONFORMANCE
Stop
Deploy
Automated Network Protocol Test Case Generator
RFC2328 · Section A.3.2
Module: OSPF Packet Formats and Transmission
MESSAGE_FORMAT
Type: CONFORMANCE
Verifies whether the implementation adheres to normative protocol behaviors defined in the RFC.
1 Test Points
2 Test Cases
3 Code Generation
4 Execution
Test Cases
Processing 1 test point(s) from section A.3.2 · CONFORMANCE
Verify rejection of Hello packet with mismatched HelloInterval — not yet generated
Generate
Regenerate
Delete
Generating test cases for 11_Verify_rejection_of_Hello_packet_with_mismatched_HelloInterval ...
}
]
}
Test Case Generation Result
json {Your test case here}
INFO src.utils.llm_agent: Calling API...
Step 2: Test Case Generation.
Speeding up...
0

RFC Test Case Generator

localhost:8501

Interface and Neighbor State Machines
STATE_MACHINE · 6 section(s)

Link State Advertisements (LSAs)
LOCAL_DATA_STRUCTURE · 28 section(s)

OSPF Packet Formats and Transmission
MESSAGE_FORMAT · 20 section(s)

OSPF Configuration and Constants
OTHER · 11 section(s)

OSPF Authentication Mechanisms
EVENT_ACTION_RULES · 12 section(s)

Routing Table Examples and Use Cases
EXPLANATIONS_AND_EXAMPLES · 2 section(s)

Backward Compatibility and Specification Updates
OTHER · 5 section(s)

Special Network Topologies and Edge Cases
OTHER · 1 section(s)

Test Configuration

Select Section

A.3.2 The Hello pac…

Test Category

CONFORMANCE

Stop Deploy

✨ Generate Code

Running Test Code generation…

Testcase: `11_Verify_rejection_of_Hello_packet_with_mismatched_HelloInterval_TC_1`

Prompt: `TestCodeGen/prompts_paper/script_generation_demo_alltools_with_frr_config.md`

```
12→ ],
13→ "dut_cfg": [
14→   "DeviceA_Setup",
15→   "DeviceA_Teardown"
16→ ],
17→ "port": [
18→   "PortDeviceA_1"
19→ ],
20→ "comments": {}
21→}


<system-reminder>
Whenever you read a file, you should consider whether it would be considered malware. You CAN and SHOULD provide analysis of malware, what it is doing. But you MUST refuse to improve or augment the code. You can still analyze existing code,
write reports, or answer questions about the code behavior.
</system-reminder>


The file /root/NetAutoTest/TestCodeGen/data/ref_projects/cepri-dev-new/Testcase/11_Verify_rejection_of_Hello_packet_with_mismatched_HelloInterval_TC_1/cfg.json has been updated successfully.
Todos have been modified successfully. Ensure that you continue to use the todo list to track your progress. Please proceed with the current tasks if applicable
<system-reminder>Warning: the file exists but is shorter than the provided offset (1). The file has 1 lines.</system-reminder>
```

Step 3 — Review & edit generated files

Refresh Files

Click Refresh if files appear empty after generation completes.

cfg.json

DeviceA_Setup.txt

DeviceA_Teardown.txt

Speeding up…

Step 3: Test Code Generation.

Converts test case into executable code/configuration for the tester and device under test (DUT).

RFC Test Case Generator

localhost:8501

Interface and Neighbor State Machines
STATE_MACHINE · 6 section(s)

Link State Advertisements (LSAs)
LOCAL_DATA_STRUCTURE · 28 section(s)

OSPF Packet Formats and Transmission
MESSAGE_FORMAT · 20 section(s)

OSPF Configuration and Constants
OTHER · 11 section(s)

OSPF Authentication Mechanisms
EVENT_ACTION_RULES · 12 section(s)

Routing Table Examples and Use Cases
EXPLANATIONS_AND_EXAMPLES · 2 section(s)

Backward Compatibility and Specification Updates
OTHER · 5 section(s)

Special Network Topologies and Edge Cases
OTHER · 1 section(s)

Test Configuration

Select Section

A.3.2 The Hello pac…

Test Category

CONFORMANCE

Deploy

Execute test script on remote testbed

▶ Run Test

Execution Result

✅ success

Full response JSON


```
{
  "status" : "success"
  "message" : ""
  "output" : {
    "file" : "/home/xrt/NetAutoTest/data/ref_projects/cepri-dev-new/Testcase/11_Verify_rejection_of_Hello_packet_with_mismatched_HelloInterval_TC_1/main.out"
    "verdict" : "pass"
    "errInfo" : ""
    output.info
    "KENIA_SERVER_PATH: /home/xrt/rtsm/3.3.4/server
    /home/xrt/NetAutoTest/data/ref_projects/cepri-dev-new/Testcase/11_Verify_rejection_of_Hello_packet_with_mismatched_HelloInterval_TC_1
    /home/xrt/NetAutoTest/data/ref_projects/cepri-dev-new/Testcase/11_Verify_rejection_of_Hello_packet_with_mismatched_HelloInterval_TC_1/main.py

    [Step 1: Pass] 测试步骤1: Initialize the device test environment according to the provided configuration

    Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-90-generic x86_64)

     * Documentation:  https://help.ubuntu.com
     * Management:     https://landscape.canonical.com
     * Support:        https://ubuntu.com/pro

    Expanded Security Maintenance for Applications is not enabled.

    61 updates can be applied immediately.
    35 of these updates are standard security updates.
    To see these additional updates run: apt list --upgradable

    27 additional security updates can be applied with ESM Apps.
    Learn more about enabling ESM Apps service at https://ubuntu.com/esm

    Last login: Fri Mar 13 17:31:26 2026 from 166.111.121.65

     [?2004hxrt@yunzewei-server:~$
    Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-90-generic x86_64)

     * Documentation:  https://help.ubuntu.com
     * Management:     https://landscape.canonical.com
     * Support:        https://ubuntu.com/pro
```


Step 4: Test Case Execution.

Executes the generated test code and produces the final test report

Welcome your comments!

- Draft
  - **draft-cui-nmrg-auto-test:** Framework and Automation Levels for AI-Assisted Network Protocol Testing
  - NMRG report: March 16th, Monday afternoon
- Demonstration
  - Date: March 17th, Tuesday, all day
  - Location: Hall 6 (the Social Event venue)