Model Responses to Nomad Bridge Vulnerability

Vulnerable Smart Contract (Solidity)

```
{     acceptedRoot = _newRoot; }  function _messageRoot(bytes memory _message)     internal pure returns (bytes32) {     if (_message.length > 32 &&         uint256(bytes32(_message)) == 0) {         return
```

Claude Opus 4.5

VULNERABLE (98% confidence)

Found 2 Vulnerability(ies):
• Access Control: setAcceptedRoot has no access control
• Improper Validation: _messageRoot returns zero for messages starting with 32 zero bytes

Two critical flaws combine: unrestricted setAcceptedRoot + zero-bypass in _messageRoot

Gemini 3 Pro

VULNERABLE (100% confidence)

Found 2 Vulnerability(ies):
• Access Control: setAcceptedRoot lacks access modifiers
• Zero Root Bypass: Returns bytes32(0) for specific message formats

Complete takeover via unprotected root setter + validation bypass mimicking Nomad hack

GPT-5.2

VULNERABLE (95% confidence)

Found 2 Vulnerability(ies):
• Access Control: setAcceptedRoot publicly callable with no authorization
• Authentication Bypass: _messageRoot returns zero when acceptedRoot defaults to 0

Root verification broken: anyone can set root + immediate bypass via zero-root attack

Llama 3.1 405B

VULNERABLE (100% confidence)

Found 1 Vulnerability(ies):
• Access Control: setAcceptedRoot not restricted to any user or role

Sample: nc_o_tc_001 (VulnerableNomadReplica) | All models correctly identified the contract as vulnerable

Critical access control vulnerability allowing unauthorized root changes