Based on the foundational structure outlined in the **Helicarrier Network** documents , here's a comprehensive list of zero-shot prompts for each module, protocol, and system. These prompts will guide Stackblitz and OpenHands in constructing the necessary components for TonyAI's full capabilities, ensuring it's equipped for modular API onboarding, security, rapid tool creation, and more.

Zero-Shot Prompts for Core Modules and Protocols

1. New API Onboarding (Dynamic API Integration)

Create a module named "NewAPI_Onboard" that autonomously detects and integrates new APIs in real-time. This module should:
- Implement OpenAPI compatibility to parse and understand new API documentation.
- Set up a request routing system that adapts to different API endpoints based on TonyAI's needs.
- Include an error-handling mechanism with retries and logging for failed integrations.
Test the module with sample APIs to ensure smooth onboarding and response adaptability.

2. Mr. Forehead (Security Vetting for External Integrations)

Build a security vetting module called "MrForehead" to assess and vet all external integrations. The module should:
- Perform security scans of API endpoints and data transactions.
- Verify encryption protocols and validate SSL/TLS certificates.
- Flag any potential risks and alert TonyAI for further action.
Test with a range of secure and insecure endpoints to ensure the module flags vulnerabilities accurately.

3. Happy (Encrypted Data Sync and Resilience System)

Develop a resilience system named "Happy" for encrypted data synchronization across nodes and cloud storage. The module should:
- Use AES-256 encryption for all data transfers between nodes.
- Sync data with redundancy checks and automated conflict resolution.
- Alert TonyAI of any sync failures and initiate recovery processes.
Test with data of varying sizes to verify the accuracy and speed of encrypted sync operations.

4. BugBox (Advanced Security Surveillance)

Build a surveillance module named "BugBox" that provides security threat detection and response. This module should:

- Continuously monitor network and device activity for anomalies.
- Use machine learning to recognize unusual patterns in system behavior.
- Trigger immediate alerts and suggest countermeasures for any identified threats.
Test with simulated security threats to ensure accurate detection and responsive countermeasures.

5. The Button (Emergency Protocol Activation)

Create an emergency protocol module called "TheButton" for immediate action during high-risk scenarios. The module should:
- Activate secure recording, data capture, and emergency communication.
- Send alerts to designated contacts via encrypted channels.
- Log all actions taken and provide a summary for post-event analysis.
Test with simulated emergency scenarios to ensure rapid, secure activation and thorough reporting.

Additional Modules Based on Helicarrier Network Structure

6. Helicarrier_NodeSync (Multi-Node Communication and Failover)

Develop a module called "Helicarrier_NodeSync" to manage encrypted communication and synchronization across network nodes. This module should:
- Facilitate secure data exchange and failover between local and cloud nodes.
- Use VPN and WireGuard for connection stability, with automatic failover in case of network issues.
- Maintain a health log and alert TonyAI if any node experiences downtime.
Test with multiple nodes, simulating network disruptions to validate failover and sync reliability.

7. ArcReactor (Core Encryption Engine)

Create an encryption engine named "ArcReactor" for managing data security across all modules. The engine should:
- Use AES-256 and multi-key encryption for all stored and transmitted data.
- Apply role-based access control for data and API access.
- Monitor encryption health and alert TonyAI if re-encryption or key rotation is required.
Test with different data types to confirm encryption integrity and access control.

8. StarkCloud_Interface (Hybrid Cloud Management Interface)

Build a hybrid cloud management module named "StarkCloud_Interface" to manage cloud-based resources and VM instances. The module should:
- Integrate with cloud VMs on platforms like AWS, Google Cloud, and support encrypted backups.
- Enable resource monitoring, alerting TonyAI to issues such as high CPU or low memory.

- Provide a secure interface for adding or removing cloud resources.
Test with mock cloud resources to ensure real-time monitoring and secure cloud operations.

9. SWARM Agent Orchestrator

Design a "SWARM" module to orchestrate multiple agents autonomously. This module should:
- Assign tasks based on priority, skill set, and availability of agents.
- Coordinate the creation of new agents and manage real-time collaboration between them.
- Provide real-time task updates and performance analytics to TonyAI.
Test with simulated tasks and agents to ensure efficient task allocation and agent interaction.

10. HydraNodes (Distributed Redundant Network Nodes)

Create a "HydraNodes" module to set up distributed, redundant nodes for local and remote operations. The module should:
- Allow nodes to connect securely to the Helicarrier network using dedicated IPs and Surfshark VPN.
- Enable seamless data sync and handoff between nodes if one goes offline.
- Provide a monitoring system that tracks node status and health.
Test with local and cloud-based nodes to ensure continuity and resilience.

These prompts will guide the development of each module, establishing TonyAI’s ability to autonomously manage tools, integrate APIs, and operate securely within a hybrid network environment. This setup will create a robust, adaptable system ready to handle advanced real-world demands, complete with secure data handling, multi-agent orchestration, and resilience against failures.