# Harnessing LLMs for Application Security

## Course Overview

This comprehensive course is designed for developers and cybersecurity professionals seeking to harness the power of Generative AI and Large Language Models (LLMs) to enhance software security and development practices. Participants will gain a deep understanding of LLM functionality, strengths, and weaknesses, and learn to craft effective prompts for diverse use cases. The curriculum covers essential topics such as embeddings, vector stores, and Langchain, offering insights into document loading, code analysis, and custom tool creation using Agent Executors.

### Course Highlights:

1. Hands-on techniques like Retrieval-Augmented Generation (RAG) and Few-Shot Prompting for secure code analysis and threat modeling.
2. Integration of AI into security tasks to identify vulnerabilities and improve overall application security.

## Instructors

**Seth Law** is the Founder & Principal of Redpoint Security and **Ken Johnson** is the Co-Founder and CTO of **DryRun Security**. Both Seth & Ken utilize LLMs heavily in their work and both Seth and Ken have a wealth of real world applicable skills to share in applying LLMs to the application security domain.

### Ken Johnson

*CTO & Co-Founder, DryRun Security*

## Course Outline

1. **Introductions & Overview**
    a. Introduction to Generative AI Concepts
    b. Understanding LLMs: Functionality, Strengths, Weaknesses
2. **Lab Setup**
    a. Ensure all students systems work and can reach our LLM and Vector DB
3. **Langchain**

    a. Overview & Components
    b. Explanation of documentation and concepts
4. **Prompt Engineering**
    a. Types of Prompts: User, System, AI
    b. Few-Shot Prompting: Importance & Usage
    c. Prompt Engineering Frameworks - CO-STAR, CLARITY, SMART, etc.
    d. Exercise: Craft prompts using one of the preferred frameworks
5. **Context**
    a. About
    b. Use Cases & Types of Context
    c. Length / Window
    d. Exercise: Use context to improve prompt performance
6. **Embeddings & Vector Stores**
    a. Background: Formats, Documents, Metadata
    b. Use Cases: Similarity Searches, Chaining
    c. Exercise: Use vector store as context
7. **Exploring LLMs**
    a. Types of LLMs: Open Source & Commercial (HuggingFace, Anthropic, OpenAI, etc.)
    b. Hosting options - Hosted vs Transactional (Ex: Sagemaker vs Bedrock)
    c. Considerations
8. **Chatbot / AppSec Assistant**
    a. Background & Use Cases
    b. Retrieval Augmented Generation (RAG) Techniques
    c. Exercise: Build an AI Assistant that uses your company's documentation to answer questions for developers
9. **Source Code Analysis**
    a. Recommended Approaches
    b. Code Splitting, Tree-sitter & Langchain Support
    c. Few Shot Prompting for Tuning Results
    d. Building a Knowledge-base
    e. Compositional Analysis
        i. Exercise: Build Information Gathering Tool
    f. Vulnerability Analysis & Discovery
        i. Exercise: Build Vulnerability Scanning Tool
10. **Agent Executors & Custom Tools**
    a. Use Cases: Compositional & Behavioral Analysis
    b. Langchain ReAct
    c. Langraph Usage

    d. Exercise: Build a “Chain of Thought” so that the LLM uses reasoning and additional lookups in source code to find the answers it needs to validate a vulnerability (ex: validate insecure direct object reference finding)
11. **Testing Framework**
    a. Background & Why
    b. Integration vs Unit-Testing w/ LLMs
    c. Exercise: Use PyTest to build Integration Test
12. **Threat Modeling**
    a. Attack Tree Analysis
    b. Diagram Generation
    c. Risk Assessment
    d. Exercise: Build an automated Threat Modeling tool

## Key Takeaways

1. **Practical Mastery of AI-Driven Development Tools:** Gain hands-on experience with technologies like Langchain, embeddings, and vector stores.
2. **Advanced Prompt Engineering Techniques:** Learn to craft effective prompts and leverage Few-Shot Prompting.
3. **Enhanced Security Practices Through AI:** Apply AI for secure code analysis, threat modeling, and DevSecOps.

## Who Should Attend

1. **Software Developers and Engineers:** Looking to integrate AI and LLMs into development processes and improve application security.
2. **Cybersecurity Professionals:** Focused on application security, threat modeling, and DevSecOps, seeking to leverage AI-driven tools for vulnerability identification and risk assessment.

## Contact Information

For further inquiries, please reach out to the instructors via their provided contact details.

This structure aligns with the professional look of the provided PDF and clearly conveys the essential information about the course. Let me know if you’d like any more adjustments!