# Natural Language Query System Documentation

## Overview

The Natural Language Query System provides strategic conversational AI for cybersecurity, transforming complex cybersecurity data analysis into simple, natural language conversations. This system enables non-technical stakeholders to access deep security insights through ChatGPT-style interactions.

## Base URL

http://localhost:3000/api/v1/nl-query

## Authentication & Permissions

* **Authentication**: JWT token required
* **Permission**: nl\_query:create for processing queries, nl\_query:read for history and analytics

## Core Features

### 🤖 Advanced Natural Language Processing

#### **Conversational AI Interface**

* **ChatGPT-style Interface**: Natural, conversational interactions with cybersecurity data
* **Context-Aware Responses**: AI understanding of organizational context and security posture
* **Multi-Turn Conversations**: Complex queries broken down into manageable conversation flows
* **Natural Language to SQL**: Automatic conversion of questions into database queries

**Endpoint:** POST /process

**Request Example:**

{  
 "query": "Show me all critical vulnerabilities affecting our web servers",  
 "conversationContext": {},  
 "includeVisualization": true,  
 "includeRecommendations": true  
}

**Response Structure:**

{  
 "data": {  
 "queryId": 123,  
 "conversationalResponse": {  
 "mainResponse": "I found 15 critical vulnerabilities that require immediate attention. These affect your web server systems.",  
 "insights": [  
 "12 critical severity vulnerabilities identified",  
 "Some vulnerabilities affect up to 8 assets"  
 ],  
 "businessImpact": "High business impact: 12 critical vulnerabilities pose immediate risk to business operations...",  
 "recommendations": [  
 {  
 "priority": "High",  
 "action": "Prioritize patching critical vulnerabilities with CVSS scores ≥ 9.0",  
 "timeline": "Within 72 hours"  
 }  
 ],  
 "executiveSummary": "Security Assessment: 15 vulnerabilities identified, 12 critical. Immediate action required for critical items."  
 },  
 "confidence": 0.85,  
 "executionTime": 1.23,  
 "suggestedFollowUps": [  
 "Which assets are most affected by these vulnerabilities?",  
 "What is the remediation timeline for these critical vulnerabilities?"  
 ]  
 }  
}

### 🔄 Multi-Turn Conversations

#### **Context-Aware Follow-ups**

Continue conversations with full context from previous queries.

**Endpoint:** POST /continue

**Request Example:**

{  
 "followUpQuery": "Which of these vulnerabilities affect our most critical systems?",  
 "originalQueryId": 123  
}

### 📊 Comprehensive Query Capabilities

#### **Vulnerability Analysis**

* **Critical Vulnerability Discovery**: “Show me all critical vulnerabilities affecting our web servers”
* **Trend Analysis**: “What vulnerability trends have we seen over the last quarter?”
* **Impact Assessment**: “Which vulnerabilities pose the highest risk to our operations?”
* **Asset Correlation**: “Which assets are most affected by recent vulnerabilities?”

#### **Compliance Inquiries**

* **Status Assessment**: “What is our current NIST 800-53 compliance status?”
* **Gap Analysis**: “Which controls are not implemented and need immediate attention?”
* **POAM Tracking**: “What is the status of POAMs due this month?”
* **Control Effectiveness**: “How effective are our current security controls?”

#### **Risk Assessment**

* **Asset Risk Ranking**: “Which systems pose the highest risk to our organization?”
* **Risk Trend Analysis**: “How has our risk posture changed over the last quarter?”
* **Risk Factor Analysis**: “What are the top risk factors in our environment?”
* **Business Impact**: “What security issues have the highest business impact?”

#### **Executive-Level Intelligence**

* **Strategic Insights**: “How has our security posture improved over the last quarter?”
* **Business Impact Analysis**: “What is the business impact of our current security issues?”
* **Predictive Analytics**: “What security trends should we prepare for?”
* **Comparative Analysis**: “How does our security posture compare to industry standards?”

## AI Processing Pipeline

### 🧠 Natural Language Processing Workflow

1. **Intent Classification**: Determine query intent and category
2. **Entity Extraction**: Extract structured entities (severities, asset types, time periods)
3. **Context Integration**: Incorporate conversation context and user profile
4. **SQL Generation**: Convert natural language to optimized SQL queries
5. **Query Execution**: Execute generated queries with performance monitoring
6. **Response Formatting**: Generate conversational responses with insights
7. **Recommendation Generation**: Provide actionable recommendations
8. **Follow-up Suggestions**: Suggest relevant follow-up questions

### 🎯 Intent Classification

The system recognizes various intents:

* **find\_critical\_vulnerabilities**: Identify high-priority security issues
* **compliance\_status**: Check regulatory compliance status
* **highest\_risk\_assets**: Identify most vulnerable systems
* **poam\_status**: Track remediation progress
* **vulnerability\_trends**: Analyze security trends over time
* **web\_server\_query**: Focus on web server security
* **general\_inquiry**: Handle miscellaneous questions

### 📈 Entity Extraction

Automatically extracts structured data:

{  
 "severities": ["critical", "high"],  
 "assetTypes": ["web server", "database"],  
 "timePeriod": "quarter",  
 "frameworks": ["nist"],  
 "quantity": 10,  
 "contextual": {...}  
}

## Response Generation

### 💬 Conversational Response Components

#### **Main Response**

Human-readable summary of findings tailored to the query intent.

#### **Insights**

Key analytical insights derived from the data: - Pattern recognition - Statistical analysis - Trend identification - Anomaly detection

#### **Business Impact Analysis**

Translation of technical findings into business context: - Risk assessment - Operational impact - Financial implications - Compliance considerations

#### **Actionable Recommendations**

Prioritized recommendations with timelines:

{  
 "priority": "High",  
 "action": "Prioritize patching critical vulnerabilities with CVSS scores ≥ 9.0",  
 "timeline": "Within 72 hours"  
}

#### **Executive Summary**

Concise summary suitable for leadership reporting.

#### **Data Visualization Suggestions**

Recommended charts and visualizations:

{  
 "type": "bar\_chart",  
 "title": "Vulnerabilities by Severity",  
 "xAxis": "severity",  
 "yAxis": "count",  
 "description": "Shows distribution of vulnerabilities across severity levels"  
}

## Query Management

### 📚 Query History

**Endpoint:** GET /history

Track all user queries with filtering options: - Status filtering (completed, failed, etc.) - Query type filtering - Time-based filtering - Confidence scoring

### 👍 Feedback System

**Endpoint:** POST /{queryId}/feedback

Collect user feedback for continuous improvement: - Helpfulness ratings - Accuracy assessment - Improvement suggestions - Comment collection

### 📊 Analytics Dashboard

**Endpoint:** GET /analytics (Admin only)

System-wide analytics: - Query volume and success rates - Average confidence scores - Popular query types - User satisfaction metrics

## AI Capabilities

### 🎯 Supported Query Types

#### **Asset Search**

* Asset inventory queries
* Hardware/software discovery
* Asset security status

#### **Cost Analysis**

* Financial reporting and analysis
* Budget tracking and forecasting
* Cost optimization queries

#### **Vulnerability Report**

* Security vulnerability analysis
* Risk assessment queries
* Compliance reporting

#### **Compliance Check**

* Regulatory compliance verification
* Policy adherence monitoring
* Audit preparation queries

#### **Lifecycle Planning**

* Asset lifecycle management
* Replacement planning
* Warranty and EOL tracking

#### **Operational Metrics**

* Performance monitoring
* Utilization analysis
* Operational efficiency queries

#### **Risk Assessment**

* Risk analysis and scoring
* Threat assessment
* Security posture evaluation

### 💡 Query Suggestions

**Endpoint:** GET /suggestions

Context-aware query suggestions by category: - Vulnerability management - Compliance monitoring - Risk assessment - Asset management - Trending topics

## Integration Architecture

### 🔗 Database Integration

* **Vulnerabilities**: CVE data, CVSS scores, affected assets
* **Assets**: Inventory, configurations, relationships
* **Compliance**: Controls, POAMs, assessment results
* **Risk**: Risk scores, assessments, mitigation status

### 🤖 AI/ML Integration

* **Intent Classification**: Rule-based and ML-based classification
* **Entity Extraction**: Named entity recognition
* **Query Generation**: Template-based and AI-generated SQL
* **Response Optimization**: Continuous learning from feedback

## Security & Privacy

### 🔒 Data Protection

* **Query Sanitization**: Prevent SQL injection
* **Result Filtering**: Apply user permissions
* **Audit Logging**: Complete query tracking
* **Data Masking**: Protect sensitive information

### 👤 User Context

* **Permission Integration**: Respect access controls
* **Role-based Responses**: Tailor responses to user role
* **Department Filtering**: Scope queries appropriately
* **Data Sovereignty**: Respect data boundaries

## Usage Examples

### Basic Vulnerability Query

curl -X POST "http://localhost:3000/api/v1/nl-query/process" \  
 -H "Authorization: Bearer YOUR\_TOKEN" \  
 -H "Content-Type: application/json" \  
 -d '{  
 "query": "Show me all critical vulnerabilities affecting our web servers",  
 "includeRecommendations": true  
 }'

### Continue Conversation

curl -X POST "http://localhost:3000/api/v1/nl-query/continue" \  
 -H "Authorization: Bearer YOUR\_TOKEN" \  
 -H "Content-Type: application/json" \  
 -d '{  
 "followUpQuery": "Which of these have patches available?",  
 "originalQueryId": 123  
 }'

### Get Query Suggestions

curl "http://localhost:3000/api/v1/nl-query/suggestions?category=vulnerability\_management&limit=5" \  
 -H "Authorization: Bearer YOUR\_TOKEN"

## Best Practices

### 📋 Query Optimization

1. **Be Specific**: Include specific asset types, time ranges, or severity levels
2. **Use Context**: Leverage multi-turn conversations for complex analysis
3. **Provide Feedback**: Rate responses to improve AI accuracy
4. **Follow Suggestions**: Use suggested follow-ups for deeper insights

### 🎯 Executive Communication

1. **Focus on Business Impact**: Emphasize business implications
2. **Use Executive Summaries**: Leverage generated executive summaries
3. **Include Recommendations**: Always request actionable recommendations
4. **Visualize Data**: Use suggested visualizations for presentations

### 🔧 System Administration

1. **Monitor Analytics**: Track system usage and performance
2. **Review Feedback**: Analyze user feedback for improvements
3. **Update Templates**: Maintain and improve query templates
4. **Performance Tuning**: Optimize SQL generation and execution

## Testing

### 🧪 API Testing

# Test the complete NL Query system  
npm run test:nl-query-api

### 📊 Performance Testing

* Query processing time monitoring
* SQL execution optimization
* Response generation efficiency
* Concurrent user handling

This Natural Language Query System transforms cybersecurity data analysis into intuitive conversations, enabling stakeholders at all levels to access critical security insights through natural language interactions.