BLACK KITE'S
GUIDELINES FOR WRITING LLM-COMPATIBLE CYBERSECURITY CONTROLS

To ensure effective automation and accurate analysis by our AI platform, please follow these best practices when creating custom compliance frameworks.

1. Use Clear, Deterministic Control Statements over Open-Ended or Vague Questions

Avoid vague or open-ended questions as much as possible. Controls should state an expected cybersecurity practice or ask for a specific condition.

- ❌ "How is your network secured?"
- ✅ "The organization uses a firewall to restrict unauthorized access to internal networks."
- ❌ "Explain your approach to endpoint security."
- ✅ "All endpoints are protected with EDR (Endpoint Detection and Response) solutions."

2. Avoid Conditional or Branching Logic

LLMs do not support logic trees like "If yes, go to control X." Break them into separate controls instead.

- ❌ "If using a third-party email service, go to Control 5; else, go to Control 6."
- ✅ "Does the organization use a third-party email provider?"
- ✅ "The third-party email provider enforces SPF, DKIM, and DMARC for domain protection."
- ❌ "Do you use cloud services? If yes, are those services configured securely?"
- ✅ "Does the organization utilize cloud service providers for data storage or processing?"
- ✅ "Cloud environments are configured to restrict public access, enforce encryption at rest and in transit"

3. Split Compound Controls into Atomic Units

Each control should measure one specific requirement.

- ❌ "Is multi-factor authentication enabled for privileged accounts and is remote access monitored?"
- ✅ "Multi-factor authentication is enabled for all privileged accounts."
- ✅ "Remote access is monitored and logged in real time."

4. Use a Consistent Control Format

Each control should include:

- Control ID (e.g., CF-07.2)
- Control Category → Area, Category, Domain information to which the control belongs
- Control Description → Deterministic statement or yes/no question is preferred (unless you require further information from vendors)

5. Be Explicit About the Subject or Object of Each Control

Avoid generic language that could apply to multiple contexts. Clearly state what the control refers to—e.g., personal data, source code, network traffic.

- ❌ "Is encryption in place?"
- ✅ "Personal information stored in databases is encrypted at rest using AES-256."
- ✅ "Source code repositories are encrypted in transit and at rest."

This is especially important when a control with the same phrasing could appear in multiple sections (e.g., "Personal Information" vs. "Coding Practices"). Ambiguity reduces the effectiveness of automated processing.

6. Avoid Ambiguous Language

Replace soft terms like "sufficient", "robust", or "secure enough" with measurable or observable conditions.

- ❌ "The organization has a robust patch management process in place."
- ✅ "Security patches are applied to critical systems within 15 days of release."

7. Avoid Hypothetical or Future-Oriented Conditions

Controls should reflect current state, not potential changes.

- ❌ "If you migrate to the cloud, will you implement encryption?"
- ✅ "Data stored in cloud environments is encrypted using AES-256."

8. Use Cybersecurity Domain Language

Controls should reflect best practices and terminology familiar to cybersecurity professionals.

- ✅ "Audit logs are retained for a minimum of 180 days and reviewed weekly."
- ✅ "Administrative access requires VPN with multi-factor authentication enabled"

9. Group Similar Controls Under Categories or Domains

Organize controls into logical categories (e.g., Access Control, Data Protection, Incident Response) to improve clarity and consistency.

- ❌ Uncategorized list of controls mixing access, logging, and encryption topics.
- ✅ Grouped list with headings like:
  - Access Control
    - "Multi-factor authentication is enabled for all privileged accounts."
    - "Administrative access requires VPN and multi-factor authentication."
  - Data Protection
    - "Data stored in cloud environments is encrypted using AES-256."
    - "Security patches are applied to critical systems within 15 days of release."