Lab 4.1 – Building a Risk Register

1. Lab Overview

1.1 Lab Description

In this lab, you'll learn how to build and maintain a **risk register** — a core tool in cybersecurity governance and risk management. Unlike technical labs focused on system or network tools, this exercise emphasizes **analysis, structured documentation, and strategic thinking**.

You'll capture risks in a structured format, score their likelihood and impact, assign ownership, and make them **visible and actionable**. By the end, you'll have your own working risk register — a document that organizations use daily to guide decisions and strengthen defenses.

1.2 Learning Objectives

After completing this lab, you will be able to:

- Define what a risk register is and explain its purpose.
- Identify the **key components** of a risk register (risk description, likelihood, impact, score, owner, status).
- Translate a real-world threat into a structured risk entry.
- Apply a **risk scoring model** (impact × likelihood).
- Create both technical and human/policy risk entries.
- Understand how risk registers support **strategic security decisions**.

1.3 Prerequisites

Before starting this lab, you should have:

- A basic understanding of **threats, vulnerabilities, and risks** (from earlier modules).
- Familiarity with **risk impact and likelihood scales** (Low, Medium, High).

- A spreadsheet tool (Excel, Google Sheets, or LibreOffice Calc) or a simple table in Word/Docs.
- A notebook (digital or physical) for reflection questions.

1.4 Estimated Completion Time

Activity	**Estimated Time**
Introduction & context	30 minutes
Reviewing key risk register components	45 minutes
Example walkthrough (phishing risk)	30 minutes
Challenge: Create 2 sample entries	60 minutes
Strategic use & discussion	30 minutes
Reflection & wrap-up	15 minutes
Total Estimated Time	**~3.5–4 hours**

Checkpoint
Before moving on, confirm that you:

- Have a spreadsheet tool ready.
- Understand the basics of risks, threats, and vulnerabilities.
- Know you'll be creating **at least two risk entries** (technical + human/policy).
- Can dedicate around **3.5–4 hours** for this lab.

2. Getting Started

2.1 What is a Risk Register?

A **risk register** is a structured document (often a spreadsheet or database) that records, tracks, and manages security risks.

Each entry in the register represents a specific risk and contains information such as:

- What the risk is (description)
- What causes it (threats + vulnerabilities)
- How likely it is to occur (likelihood)
- How damaging it would be if it happened (impact)
- How it is being managed (mitigation, status, owner)

Think of the risk register as your **"control tower"** for monitoring risks — it gives you a complete picture of your organization's risk landscape.

2.2 Why Risk Registers Matter in Cybersecurity

Risk registers are more than documentation; they are **strategic tools**. A well-maintained register helps organizations:

- **Prioritize actions** → High-risk issues get fixed first.
- **Support decision-making** → Leadership sees which risks are critical.
- **Track changes over time** → Monitor if risks are growing, shrinking, or mitigated.
- **Provide evidence for audits and compliance** → Demonstrate a structured approach to risk management.

2.3 Lab Setup Checklist

Before you begin, make sure you have:

A spreadsheet tool (Excel, Google Sheets, or LibreOffice Calc).
A blank risk register template (or create a table with the columns you'll need).

Examples of risks you'd like to document (technical, human, or policy).
Reference notes from earlier modules (threats, vulnerabilities, risk scoring).

Optional: If working in a group or class, be ready to **compare risk entries** — this often reveals different perspectives on the same risk.

Checkpoint
Before moving on:

- Do you understand what a risk register is and why it matters?
- Do you have a spreadsheet or table ready to capture risks?
- Have you reviewed your notes on threats, vulnerabilities, and risk scoring?

3. Key Components of a Risk Register

3.1 Core Columns

A basic risk register can vary in complexity, but most include the following columns:

1. **Risk Description** → A clear statement of the risk (e.g., "Unpatched operating system could be exploited").
2. **Asset Affected** → The system, data, or process at risk (e.g., "Finance server").
3. **Threat Source** → Who or what might cause the risk (e.g., "External attacker," "Insider misuse").
4. **Vulnerability** → The weakness that makes the threat possible (e.g., "Unpatched software").
5. **Likelihood** → How likely the risk is to occur (e.g., Low, Medium, High).
6. **Impact** → The potential consequences if the risk occurs (e.g., Low, Medium, High).
7. **Risk Score** → A numerical or qualitative score (often calculated as **Impact × Likelihood**).
8. **Mitigation Strategy** → What you'll do to reduce the risk (e.g., "Patch system monthly").
9. **Risk Owner** → The person/team responsible for managing the risk.
10. **Status** → The current state (e.g., Open, In Progress, Mitigated, Accepted).

3.2 How Risk Scoring Works

Risk scoring helps prioritize which risks to tackle first.

- **Qualitative model** (common in smaller registers):
    - Likelihood and Impact rated as **Low, Medium, High**.
    - Combine the two to categorize the risk (e.g., "Critical", "Moderate", "Low").
- **Quantitative model** (more advanced):

- Assign numerical values (e.g., Low = 1, Medium = 2, High = 3).
- Calculate Risk Score = Likelihood × Impact.
- Example: Likelihood = 3 (High), Impact = 2 (Medium) → Score = 6.

Even a simple scoring method is useful, because it forces you to **compare risks systematically** rather than subjectively.

3.3 Recap: Turning Risk Data into Actionable Fields

Column	Purpose
Risk Description	Defines the risk clearly
Asset Affected	Shows what could be harmed
Threat Source	Identifies the actor/cause
Vulnerability	Explains the weakness exploited
Likelihood	Rates how probable the event is
Impact	Rates the severity of consequences
Risk Score	Prioritizes risks objectively
Mitigation Strategy	Defines planned defenses
Risk Owner	Assigns accountability
Status	Tracks progress

Checkpoint
Before moving on:

- Do you understand what each column in a risk register means?
- Can you explain how likelihood and impact combine into a risk score?
- Do you feel ready to try filling in your first entry?

4. Example Entry – Phishing Risk

4.1 Walkthrough: Phishing Attack Risk

Let's build an example entry for a common organizational risk: **employees falling for phishing emails**.

Column	Example Entry
Risk Description	Employees may click malicious links in phishing emails.
Asset Affected	Company credentials and email accounts.
Threat Source	External attackers sending phishing campaigns.
Vulnerability	Lack of user awareness and insufficient email filtering.
Likelihood	High
Impact	High
Risk Score	Critical (High × High)
Mitigation Strategy	Conduct security awareness training + implement simulated phishing tests.
Risk Owner	Security Awareness Team
Status	In Progress

4.2 Why This Example Matters

- **Common risk** → Phishing remains one of the top ways attackers compromise organizations.
- **Human factor** → Demonstrates how people can be both the strongest defense and the weakest link.
- **Actionable mitigation** → Training and simulations are practical steps that reduce this risk.

- **Ownership assigned** → Risks are tracked until someone takes responsibility for resolving them.

4.3 Recap: How to Structure Your Own Entry

When creating your own risk entries:

1. Start with a **clear description** (avoid vague terms like "bad security").
2. Link the risk to an **asset, threat, and vulnerability**.
3. Rate **likelihood and impact** consistently (use Low/Medium/High or numbers).
4. Calculate a **risk score** to prioritize.
5. Define a **real mitigation strategy**, not just "fix it later."
6. Assign a **risk owner** to make accountability clear.

Checkpoint
Before moving on:

- Can you explain why phishing is rated High Likelihood, High Impact?
- Do you see how the mitigation directly addresses the vulnerability?
- Are you ready to create your own risk register entries?

5. Lab Challenge – Build Your Own Entries

5.1 Create a Technical Risk Entry

Your first task is to identify and record a **technical risk**. Examples include:

- An **unpatched system** vulnerable to known exploits.
- An **open network port** accessible from the internet.
- A **misconfigured firewall rule** exposing services.

For your chosen risk, fill out all fields:

Column	Your Entry
Risk Description	...
Asset Affected	...
Threat Source	...
Vulnerability	...
Likelihood	...
Impact	...
Risk Score	...
Mitigation Strategy	...
Risk Owner	...
Status	...

5.2 Create a Human or Policy Risk Entry

Next, create a **non-technical risk** related to people, process, or policy. Examples include:

- **Weak password use** among employees.

- **Shared logins** between multiple staff members.
- **Lack of onboarding/offboarding process** for account management.

Use the same structure:

Column	Your Entry
Risk Description	…
Asset Affected	…
Threat Source	…
Vulnerability	…
Likelihood	…
Impact	…
Risk Score	…
Mitigation Strategy	…
Risk Owner	…
Status	…

5.3 Reflection Questions

After completing your entries, reflect on the following:

1. Which risk did you rate as more critical — the technical or the human/policy one? Why?
2. Did you find it easier to describe the **technical side** (vulnerabilities, assets) or the **human/policy side** (processes, behaviors)?
3. If you had to **present your entries to leadership**, how would you explain why these risks deserve attention?
4. What actions could you take in the next 30 days to reduce one of these risks?

Checkpoint
Before moving on:

- Have you created at least **two complete risk entries**?
- Did you assign **likelihood and impact** values consistently?
- Did you include a **clear mitigation strategy** for each risk?
- Did you identify a **risk owner** (even if hypothetical)?

6. Using Risk Registers for Strategic Decisions

6.1 Prioritizing High vs. Low Risks

A risk register isn't just for listing risks — it's for **prioritizing action**.

- **High-risk items** (High likelihood + High impact) should be addressed first.
- **Medium-risk items** may need monitoring or partial mitigation.
- **Low-risk items** might be acceptable and documented as "accepted risks."

This prevents teams from "chasing every problem" and instead focuses effort where it matters most.

6.2 Tracking Risk Over Time

Risks aren't static. New vulnerabilities appear, threats evolve, and mitigations reduce risk levels. A risk register helps you:

- Update **status** (Open → In Progress → Mitigated/Accepted).
- Record when risks were first logged and when they were updated.
- Spot trends, such as whether phishing is decreasing after training or whether unpatched systems remain persistent.

6.3 Reporting Risks to Leadership

Executives and managers rarely want to see raw technical details — but they care deeply about **risk summaries**.
A risk register allows you to:

- Show the **top 5 critical risks** in plain language.
- Demonstrate progress by showing risks that moved from "Critical" to "Medium."
- Provide evidence that risks are **owned and being managed**.

6.4 Compliance and Audit Support

Risk registers are also key for regulatory frameworks such as:

- **ISO 27001** (Information Security Management Systems)
- **NIST Cybersecurity Framework**
- **GDPR** (for data protection risks)

Auditors often ask to see a risk register because it:

- Shows that risks are **documented** and **actively tracked**.
- Proves that the organization isn't ignoring known threats.
- Links risks to mitigation and accountability.

Checkpoint
Before moving on:

- Can you explain how a risk register helps prioritize security actions?
- Do you understand how risk registers support reporting and compliance?
- Have you updated your own entries with **status** and **ownership** so they reflect real progress?

7. Wrap Up

7.1 Key Takeaways

In this lab, you've:
Learned what a **risk register** is and why it matters.
Explored the **core components** (description, asset, threat, vulnerability, likelihood, impact, score, owner, status).
Walked through a **phishing example** to see how risks are structured.
Built **your own risk entries** (technical + human/policy).
Understood how risk registers support **prioritization, tracking, reporting, and compliance**.

7.2 Skills You've Gained

By completing this lab, you now have practical experience in:

- Translating threats and vulnerabilities into structured risk entries.
- Applying **likelihood × impact scoring** to prioritize risks.
- Assigning ownership and tracking the status of risks.
- Using risk registers as **strategic tools**, not just documentation.

7.3 Next Step: Module 4 Final Quiz

You've now reached the end of **Module 4: Risk Management and Security Policies**.

Next, you'll take the **Module 4 Final Quiz**, which will test your knowledge of:

- Risk concepts (threats, vulnerabilities, assets, controls).
- The role of governance and policies in cybersecurity.
- The practical use of a **risk register**.

Final Checkpoint
Before you begin the quiz:

- Do you have at least two complete risk entries saved in your register?
- Can you explain how to calculate a risk score?
- Can you describe how a risk register supports security strategy?

If yes — you're ready for the quiz!

8. Appendix

8.1 Sample Risk Register Template (Spreadsheet Layout)

Risk ID	Risk Description	Asset Affected	Threat Source	Vulnerability	Likelihood	Impact	Risk Score	Mitigation Strategy	Owner	Status
R-001	Employees may click phishing links	Email credentials	External attacker	Lack of awareness	High	High	Critical	Awareness training + phishing simulations	Security Awareness Team	In Progress
R-002	Unpatched server vulnerable to CVE	Finance server	Malware exploit	Missing security updates	Medium	High	High	Monthly patch management cycle	IT Infrastructure Team	Open
R-003	Weak passwords reused by staff	Internal accounts	Insider misuse	Poor password policy	High	Medium	High	Enforce strong password policies, MFA	IAM (Identity Team)	In Progress

8.2 Example Risk Categories

When brainstorming risks, think across these categories:

- **Technical Risks**: unpatched systems, open ports, insecure configurations.
- **Human Risks**: phishing, weak passwords, insider misuse.
- **Policy/Process Risks**: missing policies, lack of training, weak onboarding/offboarding.
- **Third-Party Risks**: vendor breaches, supply chain attacks, outsourced services.

Pro tip: Try to capture **at least one risk per category** for a more balanced register.

8.3 Common Mistakes to Avoid

When students first build risk registers, they often:

- **Write vague risk descriptions** (e.g., “security problem” instead of “unpatched OS vulnerability”).

- **Skip assigning an owner** → no one accountable means no progress.
- **Overestimate everything as High risk** → makes prioritization meaningless.
- **Forget to update** → registers must evolve with new threats.
- **Confuse threats and vulnerabilities** → remember:
    - Threat = potential danger (e.g., attacker).
    - Vulnerability = weakness exploited (e.g., open port).

8.4 Further Resources

- **NIST Cybersecurity Framework (CSF)** – Guidance on risk management: https://www.nist.gov/cyberframework
- **ISO/IEC 27005** – International standard on information security risk management.
- **ENISA Risk Management Guidance** – Practical guides from the EU Agency for Cybersecurity.
- **OWASP Risk Rating Methodology** – For scoring risks in web and app security: https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

Suggested Reading: *Managing Risk in Information Systems* (Darril Gibson) — an accessible introduction to building and maintaining risk registers.