Security Scan Report for test_1.py
Scan ID: 2

Scan Findings:

Issue: Consider possible security implications associated with pickle module.
Severity: LOW
Line Number: 3
Suggestion: Refer to Bandit documentation for remediation steps.

Issue: Possible hardcoded password: 'password123'
Severity: LOW
Line Number: 15
Suggestion: Refer to Bandit documentation for remediation steps.

Issue: Possible SQL injection vector through string-based query construction.
Severity: MEDIUM
Line Number: 30
Suggestion: Refer to Bandit documentation for remediation steps.

Issue: Pickle and modules that wrap it can be unsafe when used to deserialize untrusted data, possible se
Severity: MEDIUM
Line Number: 49
Suggestion: Refer to Bandit documentation for remediation steps.

Issue: Call to requests without timeout
Severity: MEDIUM
Line Number: 78
Suggestion: Refer to Bandit documentation for remediation steps.

Issue: A Flask app appears to be run with debug=True, which exposes the Werkzeug debugger and allows
Severity: HIGH
Line Number: 132
Suggestion: Refer to Bandit documentation for remediation steps.

Issue: python.django.security.injection.sql.sql-injection-using-db-cursor-execute.sql-injection-db-cursor-exe
Severity: WARNING
Line Number: 26
Suggestion: User-controlled data from a request is passed to 'execute()'. This could lead to a SQL injection

```
----------------------------------------
Issue: python.django.security.injection.sql.sql-injection-using-db-cursor-execute.sql-injection-db-cursor-exe
Severity: WARNING
Line Number: 27
Suggestion: User-controlled data from a request is passed to 'execute()'. This could lead to a SQL injection

----------------------------------------
Issue: python.django.security.injection.tainted-sql-string.tainted-sql-string
Severity: ERROR
Line Number: 30
Suggestion: Detected user input used to manually construct a SQL string. This is usually bad practice beca

----------------------------------------
Issue: python.flask.security.injection.tainted-sql-string.tainted-sql-string
Severity: ERROR
Line Number: 30
Suggestion: Detected user input used to manually construct a SQL string. This is usually bad practice beca

----------------------------------------
Issue: python.lang.security.audit.formatted-sql-query.formatted-sql-query
Severity: WARNING
Line Number: 31
Suggestion: Detected possible formatted SQL query. Use parameterized queries instead.

----------------------------------------
Issue: python.sqlalchemy.security.sqlalchemy-execute-raw-query.sqlalchemy-execute-raw-query
Severity: ERROR
Line Number: 31
Suggestion: Avoiding SQL string concatenation: untrusted input concatenated with raw SQL query can res

----------------------------------------
Issue: python.flask.security.dangerous-template-string.dangerous-template-string
Severity: ERROR
Line Number: 42
Suggestion: Found a template created with string formatting. This is susceptible to server-side template inj

----------------------------------------
Issue: python.django.security.injection.raw-html-format.raw-html-format
Severity: WARNING
Line Number: 42
Suggestion: Detected user input flowing into a manually constructed HTML string. You may be accidentally

----------------------------------------
Issue: python.flask.security.injection.raw-html-concat.raw-html-format
Severity: WARNING
Line Number: 42
Suggestion: Detected user input flowing into a manually constructed HTML string. You may be accidentally
```

```
----------------------------------------
Issue: python.flask.security.audit.render-template-string.render-template-string
Severity: WARNING
Line Number: 43
Suggestion: Found a template created with string formatting. This is susceptible to server-side template inj

----------------------------------------
Issue: python.flask.security.insecure-deserialization.insecure-deserialization
Severity: ERROR
Line Number: 49
Suggestion: Detected the use of an insecure deserialization library in a Flask route. These libraries are pro

----------------------------------------
Issue: python.lang.security.deserialization.pickle.avoid-pickle
Severity: WARNING
Line Number: 49
Suggestion: Avoid using `pickle`, which is known to lead to code execution vulnerabilities. When unpicklin

----------------------------------------
Issue: python.flask.security.audit.directly-returned-format-string.directly-returned-format-string
Severity: WARNING
Line Number: 63
Suggestion: Detected Flask route directly returning a formatted string. This is subject to cross-site scriptin

----------------------------------------
Issue: python.flask.security.audit.debug-enabled.debug-enabled
Severity: WARNING
Line Number: 132
Suggestion: Detected Flask app with debug=True. Do not deploy to production with this flag enabled as it 

----------------------------------------
Issue: No Vulnerability Detected
Severity: LOW
Suggestion: No action needed.
```