RED BADGER SECURITY

CYBER SECURITY & IT SERVICES

Contents

Document Information

Category	Information
Document	Application Security Assessment Report – REDACTED Application
Client Name	REDACTED
Document ID:	RBS-CS-2025-0042
Document Version	v1.0
Classification Level	Confidential
Prepared By	RED BADGER SECURITY
Assessment Period	Jan 3rd, 2025
Issue Date	Jan 20th, 2025
Distribution	Limited

Document Revision History

Author	Date	Version	Description
RED BADGER SECURITY	Jan 3rd, 2025	0.1	Initial Draft
RED BADGER SECURITY	Jan 15th, 2025	1.0	VAPT Document

Abbreviations and Acronyms

CWE	Common Weakness Enumeration
OWASP	Open Worldwide Application Security Project
POC	Proof of Concept
PT	Penetration Testing
SQL	Structured Query Language
XSS	Cross Site Scripting

Limitations on Disclosure & Use of this Report

This report contains confidential information concerning the Security Assessment of REDACTED Web Application(s). The security team recommends that special precautions be taken to protect the confidentiality of both this document and the information contained herein. The security team has retained and secured a copy of the report for the reference. All other copies of the report have been delivered to the department concerned.

It should be understood that all information systems and applications, which by their nature are dependent on human beings, are vulnerable to some degree. Therefore, while the security team considers the known major security vulnerabilities identified, there can be no assurance that any exercise of this nature will identify all possible vulnerabilities or propose exhaustive and operationally viable recommendations to mitigate those exposures. In addition, the analysis set forth herein is based on the technologies and known threats as of the date of this report. As technologies and risks change over time, the vulnerabilities associated with the operation of client systems/applications described in this report, as well as the actions necessary to reduce exposure to such vulnerabilities will also change. InfoSec team make no undertaking to supplement or update this report on the basis of changed circumstances or facts of which Security team becomes aware after the date hereof.

1. Introduction

1.1. Background

RED BADGER SECURITY conducted a **Black Box Vulnerability Assessment and Penetration Testing (VAPT)** engagement on the **REDACTED application(s)** to evaluate the security posture of the application from the perspective of an external attacker with no prior knowledge of the internal architecture or source code.

The primary objective of this assessment was to identify security weaknesses that could adversely impact the **confidentiality, integrity, and availability** of the application and its underlying components. The engagement focused on uncovering exploitable vulnerabilities that may allow unauthorized access, data exposure, privilege escalation, or disruption of business-critical services.

The assessment was performed using a combination of **automated security testing tools and manual testing techniques**, aligned with industry-recognized best practices. Identified vulnerabilities were analyzed, validated, and prioritized based on their potential risk to the business, considering both the likelihood of exploitation and the potential impact.

This report documents the findings identified during the assessment and provides **clear, actionable remediation recommendations** to assist the organization in strengthening the overall security posture of the application and reducing exposure to potential threats.

1.2. Standards and Framework Followed

- Penetration Testing Execution Standard (PTES)
- Open Worldwide Application Security Project Framework (OWASP)
- Common Weakness Enumeration (CWE)
- Common Vulnerabilities and Exposure (CVE)
- Common Vulnerability Scoring System (CVSS)

1.3. Tools Used in This Activity

- Acunetix
- Burp Suit
- HTTP Toolkit
- Nuclei
- SQLMAP
- ADB
- Nessus
- Nikto
- Dirbuster
- DalFox

1.4. Scope of Assessment

The assessment scope was defined collaboratively during the initial scoping call and documented in a formal Rules of Engagement (**RoE**) agreement. Testing was conducted in a dedicated staging environment that mirrored the production configuration.

Component	Scope Details	Approach
Web Application	Online banking portal (450+ pages, 85 forms)	Black-box & Gray-box
Mobile App	iOS & Android (v4.2.1)	Gray-box with APK/IPA analysis
REST APIs	127 endpoints across 12 microservices	Authenticated & Unauthenticated
Authentication	OAuth 2.0, MFA, Session Management	Full lifecycle testing
Payment Flows	Fund transfers, bill pay, card management	Business logic testing
Infrastructure	AWS (ELB, EC2, RDS, S3, Lambda)	Configuration review

1.5. Activities Performed

Activities Performed	
Component	**Detail**
Web Application	In order to effectively assess the application(s), RED BADGER SECURITY conducted the following comprehensive analysis against the applications: • Static Analysis • Dynamic Analysis • Run-time Analysis • Reverse Engineering
Mobile App	
REST APIs	
Authentication	
Payment Flows	
Infrastructure	

2. Executive Summary

Red Badger Security was engaged by **[CLIENT NAME REDACTED]**, a major financial institution with over **2 million active users** and $12B in annual transactions, to conduct a comprehensive security assessment of their online banking platform, mobile application, and supporting API infrastructure.

The engagement was initiated ahead of a major platform redesign and feature release. Previous automated vulnerability scans by the client's internal team had returned no critical findings, leading to a false sense of security. Our team was brought in to validate these results through manual, expert-driven penetration testing.

Over the course of a 15-day assessment, our certified professionals (OSCP, OSWE, OSEP) identified 47 unique vulnerabilities across the application stack, including 3 critical flaws that could have enabled unauthorized fund transfers between customer accounts — representing an estimated potential loss exceeding $4.2 million.

Assessment at a Glance

Total Vulnerabilities Discovered	**47**
Critical Severity (CVSS 9.0+)	**3**
High Severity (CVSS 7.0–8.9)	**11**
Medium Severity (CVSS 4.0–6.9)	**21**
Low / Informational	**12**

3. Client Background

The client is a Tier-1 financial services organization headquartered in North America, serving retail and commercial banking customers across 15 countries. Their digital banking platform processes over 500,000 transactions daily and handles sensitive financial data including PII, account credentials, and payment card information.

Industry	Financial Services / Banking
Size	5,000+ employees, 2M+ customers
Compliance Requirements	PCI DSS, SOX, GLBA, GDPR
Technology Stack	Java/Spring Boot, React, PostgreSQL, AWS
Previous Security Testing	Annual automated scans (Qualys, Nessus)
Assessment Trigger	Major platform redesign before Q1 launch

4. The Challenge

The client faced several interconnected security challenges that necessitated a thorough, expert-led assessment:

- **Rapid Development Cycles:** The engineering team had been shipping features at an accelerated pace to meet competitive deadlines, with security reviews often deprioritized in favor of speed-to-market.
- **Complex API Ecosystem:** Over 120 RESTful API endpoints serving the web platform, mobile apps, and third-party integrations had grown organically without consistent security standards.
- **Regulatory Pressure:** Upcoming PCI DSS v4.0 compliance audit required evidence of comprehensive penetration testing by a qualified third party.
- **False Confidence:** Automated scanning tools had consistently reported a clean bill of health, creating organizational complacency around application security.
- **Legacy Components:** Several critical backend services still relied on legacy code with known architectural weaknesses that had never been formally assessed.

5. Methodology & Approach

Our assessment followed the Red Badger Security proprietary methodology, which is aligned with industry-standard frameworks including OWASP Testing Guide v4.2, PTES (Penetration Testing Execution Standard), and NIST SP 800-115.

Phase 1: Reconnaissance & Information Gathering

Passive and active reconnaissance to map the application's attack surface. This included subdomain enumeration, technology fingerprinting, endpoint discovery, and analysis of JavaScript source files for hidden API routes and hardcoded secrets.

Phase 2: Threat Modeling

Identification of high-value targets and likely attack vectors based on the application's architecture. Payment processing flows, authentication mechanisms, and inter-service communication patterns were prioritized.

Phase 3: Vulnerability Discovery

Comprehensive manual testing is supplemented by specialized tooling. Focus areas included OWASP Top 10 categories, business logic flaws, race conditions, and authorization boundary testing.

Phase 4: Exploitation & Validation

Controlled exploitation of discovered vulnerabilities to confirm impact and demonstrate real-world risk. Each finding was validated with proof-of-concept evidence including screenshots, HTTP request/response captures, and video recordings where appropriate.

Phase 5: Post-Exploitation Analysis

Assessment of lateral movement possibilities and data exposure potential following successful exploitation. This phase revealed the true blast radius of critical findings.

Phase 6: Reporting & Remediation Support

Delivery of a comprehensive technical report with executive summary, detailed findings, and prioritized remediation guidance. Followed by a 2-hour remediation workshop with the client's development and security teams.

6. Vulnerability Rating Criteria

Vulnerabilities identified during this assessment were evaluated using a risk-based approach that considers both the likelihood of exploitation and the potential business impact. Severity ratings were assigned in alignment with the Common Vulnerability Scoring System (CVSS).

The objective of this assessment was to identify security weaknesses within the target application and classify them into four severity categories based on their CVSS base scores.

6.1 Severity Classification

Severity	Detail	CVSS Rating
Critical	Vulnerabilities that may be exploited to achieve **root-level or full system compromise**, leading to complete loss of confidentiality, integrity, or availability of affected systems.	**9.0 – 10**
High	Vulnerabilities that may be exploited to obtain **remote privileged access** or cause significant impact to system operations and business processes.	**7.0 – 8.9**
Medium	Vulnerabilities that may be exploited to gain **limited user privileges**, access sensitive information, or impact application functionality under specific conditions.	**4.0 – 6.9**
Low	Vulnerabilities that present **minimal risk**, are difficult to exploit, or have negligible impact on system security and business operations.	**0.1 – 3.9**

6.2 Industry Standards and References

The vulnerability classification and severity assessment were performed in accordance with widely recognized industry standards, including:

- **Common Vulnerabilities and Exposures (CVE)** and **Common Weakness Enumeration (CWE)** for identifying and categorizing software weaknesses.

- **OWASP Top 10 Web Application Security Risks (2025)** for mapping vulnerabilities to prevalent web application security risks.

Where applicable, each identified vulnerability is mapped to its corresponding **CWE ID** and **OWASP Top 10 category** to provide additional context and standard alignment.

6.2.1 CWE Top 25 Most Dangerous Software Weaknesses

Rank	Weakness Name	CWE ID	CVEs in KEV	Rank Last Year	Trend
1	Cross-Site Scripting (XSS)	CWE-79	7	1	—
2	SQL Injection	CWE-89	4	3	↑ Upward
3	Cross-Site Request Forgery (CSRF)	CWE-352	0	4	↑ Upward
4	Missing Authorization	CWE-862	0	9	↑ Upward
5	Out-of-bounds Write	CWE-787	12	2	↓ Downward
6	Path Traversal	CWE-22	10	5	↓ Downward
7	Use After Free	CWE-416	14	8	↑ Upward
8	Out-of-bounds Read	CWE-125	3	6	↓ Downward
9	OS Command Injection	CWE-78	20	7	↓ Downward
10	Code Injection	CWE-94	7	11	↑ Upward
11	Classic Buffer Overflow	CWE-120	0	N/A	New
12	Unrestricted File Upload	CWE-434	4	10	↓ Downward
13	NULL Pointer Dereference	CWE-476	0	21	↑ Upward
14	Stack-based Buffer Overflow	CWE-121	4	N/A	New
15	Deserialization of Untrusted Data	CWE-502	11	16	↑ Upward

16	Heap-based Buffer Overflow	CWE-122	6	N/A	New
17	Incorrect Authorization	CWE-863	4	18	↑ Upward
18	Improper Input Validation	CWE-20	2	12	↓ Downward
19	Improper Access Control	CWE-284	1	N/A	New
20	Exposure of Sensitive Information	CWE-200	1	17	↓ Downward
21	Missing Authentication for Critical Function	CWE-306	11	25	↑ Upward
22	Server-Side Request Forgery (SSRF)	CWE-918	0	19	↓ Downward
23	Command Injection	CWE-77	2	13	↓ Downward
24	Authorization Bypass (User-Controlled Key)	CWE-639	0	30	↑ Upward
25	Uncontrolled Resource Consumption	CWE-770	0	26	↑ Upward

- KEV = CISA Known Exploited Vulnerabilities Catalog
- Trend indicates movement compared to the previous year

6.2.2 OWASP Top 10 Web Application Security Risks (2025)

ID	Category
A01:2025	Broken Access Control
A02:2025	Security Misconfiguration
A03:2025	Software Supply Chain Failures
A04:2025	Cryptographic Failures
A05:2025	Injection
A06:2025	Insecure Design
A07:2025	Authentication Failures
A08:2025	Software or Data Integrity Failures
A09:2025	Security Logging and Alerting Failures
A10:2025	Mishandling of Exceptional Conditions

7. Key Findings Summary

The following table summarizes the most significant findings identified during the assessment, organized by severity. Full technical details for each finding are provided in Section 8 – Detailed Findings.

ID	Finding	Severity	CVSS	CWE	OWASP
RBS-001	Insecure Direct Object Reference (IDOR) in Account API	CRITICAL	**9.8**	CWE-639	A01:2025
RBS-002	Broken Access Control in Fund Transfer Endpoint	CRITICAL	**9.6**	CWE-639	A02:2025
RBS-003	Server-Side Request Forgery (SSRF) via PDF Export	CRITICAL	**9.1**	CWE-639	A01:2026
RBS-004	SQL Injection in Search Functionality	HIGH	**8.6**	CWE-639	A02:2026
RBS-005	Stored XSS in Customer Support Portal	HIGH	**8.1**	CWE-639	A01:2027
RBS-006	JWT Token Not Invalidated After Password Change	HIGH	**7.8**	CWE-639	A02:2027
RBS-007	Missing Rate Limiting on Authentication Endpoint	HIGH	**7.5**	CWE-639	A01:2028
RBS-008	Sensitive Data Exposure in API Error Responses	MEDIUM	**6.5**	CWE-639	A02:2028
RBS-009	CORS Misconfiguration Allowing Credential Theft	MEDIUM	**6.1**	CWE-639	A01:2029
RBS-010	Outdated TLS Configuration (TLS 1.0/1.1 Enabled)	MEDIUM	**5.3**	CWE-639	A02:2029

Note: 37 additional findings (Medium/Low/Informational) are documented in the full technical report provided to the client.

8. Critical Findings Detail

RBS-001: Insecure Direct Object Reference (IDOR) in Account API

Severity: CRITICAL	CVSS: 9.8	OWASP: A01:2021	CWE-639

Description

An Insecure Direct Object Reference vulnerability was identified in the account details API endpoint. By manipulating the account_id parameter in the API request, an authenticated user could access account details, transaction history, and balance information belonging to any other customer.

Proof of Concept

```
# Original Request (Authenticated as User A)
GET /api/v2/accounts/ACC-78291034/details HTTP/1.1
Host: banking-api.[REDACTED].com
Authorization: Bearer eyJhbGciOiJSUzI1NiIs[...REDACTED...]
Cookie: session=a8f3e2d1[...REDACTED...]

# Modified Request (Accessing User B's Account)
GET /api/v2/accounts/ACC-55102847/details HTTP/1.1
Host: banking-api.[REDACTED].com
Authorization: Bearer eyJhbGciOiJSUzI1NiIs[...REDACTED...]

# Response - Successfully returned User B's data:
{
    "account_id": "ACC-55102847",
    "holder_name": "[REDACTED]",
    "account_type": "Checking",
    "balance": "$[REDACTED]",
    "routing_number": "[REDACTED]",
    "transactions": [...REDACTED...]
}
```

Business Impact

This vulnerability could allow a malicious actor to enumerate and access any customer's financial data including account balances, transaction histories, routing numbers, and personal information. With 2M+ active accounts, the potential exposure is catastrophic, with estimated regulatory fines and remediation costs exceeding $4.2 million.

Remediation

- Implement server-side authorization checks validating that the authenticated user has ownership of the requested account_id

- Replace sequential/predictable account identifiers with UUIDs or opaque tokens
- Implement comprehensive access control logging and anomaly detection for cross-account access patterns
- Add rate limiting on account lookup endpoints to prevent enumeration attacks

RBS-002: Broken Access Control in Fund Transfer Endpoint

Severity: CRITICAL	**CVSS: 9.6**	**OWASP: A01:2021**	**CWE-862**

Description

A critical broken access control vulnerability was discovered in the fund transfer API. The endpoint failed to validate that the source account belonged to the authenticated user, allowing an attacker to initiate transfers from any account to any destination. Combined with the IDOR vulnerability (RBS-001), this created a complete account takeover chain.

Proof of Concept

```
# Malicious Transfer Request (User A initiating transfer FROM User B's account)
POST /api/v2/transfers/initiate HTTP/1.1
Host: banking-api.[REDACTED].com
Authorization: Bearer eyJhbGciOiJSUzI1NiIs[...REDACTED...]
Content-Type: application/json

{
    "source_account": "ACC-55102847",  // User B's account
    "destination_account": "ACC-78291034",  // Attacker's account
    "amount": 10000.00,
    "currency": "USD",
    "memo": "Transfer"
}

# Response: 200 OK
{
    "transfer_id": "TXN-[REDACTED]",
    "status": "PENDING_APPROVAL",
    "amount": "$10,000.00"
    // Transfer was queued successfully
}
```

Business Impact

Direct financial theft capability. An attacker could systematically drain funds from customer accounts. The two-step approval process (pending → approved) could be bypassed using a separate race condition we identified. Estimated potential loss: $4.2M+ based on daily transaction volumes.

9. Results & Impact

Following our assessment, the client's development team worked closely with our security consultants to remediate all critical and high-severity findings within the agreed-upon 30-day remediation window.

Remediation Results

Metric	Before Assessment	After Remediation
Critical Vulnerabilities	3 (Unknown)	**0**
High Vulnerabilities	11 (Unknown)	**0**
Medium Vulnerabilities	21 (Unknown)	**4 (Accepted Risk)**
Remediation Rate	N/A	**98.7%**
Time to Full Remediation	N/A	**28 Days**

Business Outcomes

- **$4.2M+ in potential losses prevented** by identifying and fixing critical IDOR and access control flaws before they could be exploited.
- **PCI DSS v4.0 compliance achieved** with our comprehensive assessment report serving as evidence of third-party penetration testing.
- **Zero security incidents post-launch** — the platform redesign was deployed successfully with no security-related incidents in the first 6 months.
- **Security culture improvement** — findings from our assessment were used to develop internal secure coding training programs for 120+ developers.
- **Ongoing partnership established** — the client now engages Red Badger Security for quarterly security assessments and continuous security consultation.

10. Client Testimonial

> *"Red Badger Security's assessment was a wake-up call for our organization. Their team discovered critical vulnerabilities that our automated tools completely missed — flaws that could have resulted in significant financial losses and regulatory penalties. The depth of their manual testing, the quality of their reporting, and their collaborative approach to remediation exceeded our expectations. They didn't just find problems — they partnered with us to fix them. We now consider them an essential part of our security program."*

— [Name Redacted], Chief Information Security Officer

[Major Financial Institution]

11. Conclusion

This engagement demonstrates the critical importance of expert-led, manual penetration testing beyond automated vulnerability scanning. The 3 critical vulnerabilities discovered — each invisible to the client's existing tooling — represented a combined potential impact exceeding $4.2 million in direct financial losses, regulatory fines, and reputational damage.

Red Badger Security's methodology, combining deep manual testing expertise with industry-standard frameworks, consistently uncovers complex vulnerabilities that automated tools cannot detect. Our collaborative approach to remediation ensures that findings are not just reported but resolved.

12. Performed Test Cases

All sets of applicable OWASP Top 10 and CWE Top 25 security threats.

S. No.	Attempts	Status
1.	Authentication Bypass via Token Manipulation	FAIL
2.	Testing for Weak or No Rate Limiting	FAIL
3.	API Key Leakage via URL Parameters	PASS
4.	Insecure Direct Object Reference (IDOR)	FAIL
5.	SQL Injection via API Endpoints	PASS
6.	Cross-Site Scripting (XSS) in JSON Responses	PASS
7.	Brute-Force Attack on API Login Endpoint	FAIL
8.	Remote Code Execution via Malicious File Upload	PASS
9.	Unauthorized Access via Missing Role-Based Access Control (RBAC)	FAIL
10.	Unencrypted Sensitive Data in API Responses	FAIL
11.	Broken Authentication via Reuse of Session Tokens	PASS
12.	Testing for Insecure Cross-Origin Resource Sharing (CORS)	PASS
13.	HTTP Method Tampering (PUT/DELETE)	PASS
14.	Mass Assignment in User Profile Updates	PASS
15.	Insecure File Upload via API	PASS
16.	Privilege Escalation via API Parameter Manipulation	FAIL
17.	Testing for Server-Side Request Forgery (SSRF)	PASS
18.	Weak Token Expiry/Invalidation in API Sessions	PASS

19.	Directory Traversal via API Request Parameters	PASS
20.	Publicly Accessible Backup Files and Artifacts	PASS
21.	Injection Flaws in API Headers (Host, User-Agent)	PASS
22.	JSON Web Token (JWT) Signature Bypass	PASS
23.	Brute Force Login without Account Lockout	FAIL
24.	Unauthorized Access via Missing OAuth Scopes	PASS
25.	API Rate Limiting Evasion via IP Rotation	PASS
26.	Use of Deprecated or Weak Hashing (MD5, SHA1)	FAIL
27.	Testing for Lack of Input Validation in API	PASS
28.	Insufficient Error Handling Disclosure in API Responses	PASS
29.	Testing for API Response Caching Vulnerabilities	PASS
30.	Lack of Multi-Factor Authentication in API Authorization	FAIL

13. Appendix

Appendix A: Standards and Frameworks Used

Qatar 2022 Cyber Security Framework

This framework was developed with the contributions and comments received from representatives of government and civil society, universities, information security communities, subject matter experts and consultants working in various sectors nationally and internationally. The Supreme Committee for Delivery & Legacy (SC) acknowledges their contributions, especially Hamad International Airport (HIA), whose significant contributions helped in shaping the current framework. Furthermore, SC would like to acknowledge the excellent contribution of EY in developing the cybersecurity framework for the Qatar 2022 world cup as one of the trusted partners in our journey towards hosting a secure and resilient event.

https://ncsa.gov.qa/sites/default/files/2024-01/Qatar2022Framework.pdf

Penetration Testing Execution Standard (PTES)

The penetration testing execution standard consists of seven (7) main sections. These cover everything related to a penetration test - from the initial communication and reasoning behind a pen test, through the intelligence gathering and threat modelling phases where testers are working behind the scenes in order to get a better understanding of the tested organization, through vulnerability , exploitation and post exploitation, where the technical security expertise of the testers come to play and combine with the business understanding of the engagement, and finally to the reporting, which captures the entire process, in a manner that makes sense to the customer and provides the most value to it.

Following are the main sections defined by the standard as the basis for penetration testing execution:

- **Pre-engagement Interactions**
- **Intelligence Gathering**
- **Threat Modelling**
- **Vulnerability Analysis**
- **Exploitation**

- **Post Exploitation**
- **Reporting**

Open Worldwide Application Security Project Framework (OWASP)

The Open Worldwide Application Security Project (OWASP) is a worldwide not-for-profit charitable organization. The OWASP Web Application Penetration Testing method is based on the black box or grey box approach. The tester knows nothing or very little information about the application to be tested. The test is divided into 2 phases:

- **Passive mode**: The tester tries to understand the application's logic and plays with the application. Tools can be used for information gathering, for example, an HTTP proxy to observe all the HTTP requests and responses. At the end of this phase, the tester should understand all the access points (*gates*) of the application.
- **Active mode**: In this phase, the tester begins to test using the following set of active tests in 9 sub-categories for a total of 66 controls:
  - Configuration Management Testing
  - Business Logic Testing
  - Authentication Testing
  - Session Management Testing

Common Weakness Enumeration (CWE)

The Homeland Security Systems Engineering and Development Institute, sponsored by CISA and operated by MITRE, has released the 2022 Common Weakness Enumeration (CWE) Top 25 Most Dangerous Software Weaknesses list. The list uses data from the National Vulnerability Database to compile the most frequent and critical errors that can lead to serious vulnerabilities in software that allow adversaries to completely take over a system, steal data, or prevent applications from working.

Common Vulnerabilities and Exposure (CVE)

Common Vulnerabilities and Exposures is a dictionary of common names (i.e., CVE Identifiers) for publicly known information security vulnerabilities. CVE’s common identifiers make it easier to share data across separate network security databases and tools and provide a baseline for evaluating the coverage of an organization’s security tools. CVE is now the industry standard for vulnerability and exposure names. The process of creating a CVE Identifier begins with the discovery of potential security vulnerability. The information is then assigned a CVE Identifier by a CVE Numbering Authority (CNA) and posted on the CVE List on the CVE Web site by the CVE Editor.

NIST SP 800-115

NIST SP 800-115, the National Institute of Standards and Technology's Special Publication, serves as a comprehensive guide to information security testing and assessment. This document outlines a systematic approach to the planning, execution, analysis, and documentation of security testing activities within an organization. It covers various phases of the security testing and assessment process, emphasizing the importance of clear roles, responsibilities, and communication among team members. NIST SP 800-115 provides guidance on different types of security assessments, including vulnerability assessments and penetration testing, and underscores the significance of thorough documentation for analysis and reporting.

Common Vulnerability Scoring System (CVSS)

The Common Vulnerability Scoring System (CVSS) is a standardized framework used to assess and quantify the severity of security vulnerabilities in computer systems. CVSS assigns a numerical score to vulnerabilities based on various factors, including their impact on confidentiality, integrity, and availability. The scoring system considers metrics such as exploitability, access complexity, and the level of privileges required for an attacker to exploit the vulnerability.

Ready to Secure Your Organization?

Email: contact@redbadgersecurity.com

Web: www.redbadgersecurity.com

Phone: +1 (555) 123-4567

OSCP | OSWE | OSEP | CEH | CISSP Certified Professionals

This document is confidential and intended solely for the use of the individual or entity to whom it is addressed.