**Vulnerability Assessment Report**  
**For PNexus Web-Based Application**

**Prepared for:**

Department of Social Welfare and Development (DSWD)  
Pantawid Pamilyang Pilipino Program (4Ps)

**Prepared by:**

Alden A. Quiñones  
Information Technology Officer I  
Department of Social Welfare and Development Field Office XII

**Date:**

[Insert Date]

**1. Introduction**

This document presents the findings from the vulnerability assessment conducted on PNexus, a web-based application used within the Pantawid Pamilyang Pilipino Program (4Ps). The assessment aimed to identify security risks, evaluate potential threats, and recommend remediation strategies to enhance the overall security posture of the application.

**2. Executive Summary**

The assessment identified several critical vulnerabilities that could compromise the security and integrity of PNexus. The most pressing issues include:

* **Unencrypted Communication (No HTTPS):** Data transmitted over HTTP is vulnerable to interception.
* **Debug Mode Enabled:** Can expose sensitive information to attackers.
* **Weak Password Policy:** No enforcement of complexity rules, making brute-force attacks easier.
* **Lack of CSRF Protection:** Exposes users to Cross-Site Request Forgery (CSRF) attacks.
* **No CAPTCHA on Login Pages:** Increases risk of automated brute-force attacks.
* **Use of Active Directory Without MFA:** Potential security risk if credentials are leaked.

Immediate action is recommended to address these issues to prevent potential data breaches or unauthorized access.

**3. Assessment Methodology**

The assessment was conducted using the following approach:

1. **Information Gathering:** Understanding application architecture and security controls.
2. **Threat Modeling:** Identifying key security threats and attack vectors.
3. **Automated Scanning:** Using security tools to detect vulnerabilities.
4. **Manual Testing:** Simulating real-world attack scenarios.
5. **Risk Analysis:** Prioritizing findings based on severity and impact.
6. **Report Generation:** Providing detailed findings and recommendations.

**4. Detailed Findings**

**4.1 Unencrypted Communication (No HTTPS)**

* **Severity:** Critical
* **Impact:** Data transmitted over HTTP can be intercepted by attackers, leading to information leakage.
* **Recommendation:** Implement SSL/TLS encryption to secure communication.

**4.2 Debug Mode Enabled**

* **Severity:** High
* **Impact:** Debug mode exposes sensitive system information, making it easier for attackers to exploit vulnerabilities.
* **Recommendation:** Disable debug mode in the production environment.

**4.3 Weak Password Policy**

* **Severity:** High
* **Impact:** Weak passwords increase the likelihood of unauthorized access through brute-force attacks.
* **Recommendation:** Enforce a strong password policy (e.g., at least 12 characters, mix of uppercase, lowercase, numbers, and special characters).

**4.4 Lack of CSRF Protection**

* **Severity:** High
* **Impact:** Attackers can trick authenticated users into executing unintended actions.
* **Recommendation:** Implement CSRF tokens on all form submissions and AJAX requests.

**4.5 No CAPTCHA on Login Pages**

* **Severity:** Medium
* **Impact:** Increases risk of automated brute-force attacks.
* **Recommendation:** Implement CAPTCHA (e.g., Google reCAPTCHA) on login and authentication pages.

**4.6 Use of Active Directory Without MFA**

* **Severity:** Medium
* **Impact:** If an Active Directory account is compromised, an attacker can gain unauthorized access.
* **Recommendation:** Implement Multi-Factor Authentication (MFA) to enhance security.

**5. Risk Rating & Prioritization**

The following table summarizes the identified vulnerabilities and their risk levels:

| **Vulnerability** | **Severity** | **Status** | **Recommended Action** |
| --- | --- | --- | --- |
| Unencrypted Communication (No HTTPS) | Critical | Open | Implement SSL/TLS |
| Debug Mode Enabled | High | Open | Disable debug mode |
| Weak Password Policy | High | Open | Enforce strong passwords |
| Lack of CSRF Protection | High | Open | Implement CSRF tokens |
| No CAPTCHA on Login Pages | Medium | Open | Implement CAPTCHA |
| Use of Active Directory Without MFA | Medium | Open | Enable Multi-Factor Authentication |

**6. Recommendations & Remediation Plan**

To address the identified security risks, the following remediation plan is recommended:

1. **Immediate Fixes (0-3 months):**
   * Enable HTTPS by installing and configuring SSL/TLS certificates.
   * Disable debug mode in the production environment.
   * Implement CSRF protection in all AJAX and form submissions.
   * Enforce strong password policies and reset weak passwords.
2. **Short-Term Fixes (3-6 months):**
   * Implement CAPTCHA on login and authentication pages.
   * Integrate Multi-Factor Authentication (MFA) for Active Directory authentication.
3. **Long-Term Fixes (6+ months):**
   * Consider upgrading to CodeIgniter v4 for improved security and long-term support.
   * Transition from XAMPP to a more secure web server setup (e.g., Nginx or Apache with hardened configurations).
   * Regular vulnerability scanning and penetration testing to identify new risks.

**7. Conclusion**

The PNexus application has several critical security vulnerabilities that must be addressed to prevent potential exploitation. Implementing the recommended security enhancements will significantly improve the system's resilience against cyber threats.

We recommend taking immediate action on high-risk vulnerabilities and following a structured remediation plan to ensure the security of the application.

**Prepared by:**  
Alden A. Quiñones  
Information Technology Officer I  
DSWD Field Office XII