تقييم الامتثال الأمني التقني: تحليل البنية التحتية لموقع example.com

كبير مهندسي الأمان

2025-12-31

الملخص التقني التنفيذي

يقدم هذا التقييم التقني الشامل تحليلاً لوضعية الأمان لموقع https://example.com مقابل أطر عمل الامتثال المتعددة بما في ذلك OWASP Top 10 2021، وإطار عمل الأمن السيبراني NIST 2.0، ومتطلبات SOC 2 Type II، ومعايير ISO 27001. يستند التحليل إلى المسح الأمني الآلي المنفذ في 30 ديسمبر 2025 باستخدام OWASP ZAP 2.17.0 من Checkmarx.

مطلوب معالجة فورية! - (PII) **تنبيه أمني حرج**: تم اكتشاف تعرض للمعلومات الشخصية

النتائج التقنية الرئيسية:

- تم تحديد **29 ثغرة أمنية** عبر مستويات خطورة متعددة
- **حرجة** بتقييم ثقة عالي **PII ثغرة إفشاء**
- **غياب آليات الحماية الأساسية** في طبقة تطبيق الويب
- **عدم امتثال** للمتطلبات الأساسية عبر جميع الأطر المقيمة

جدول المحتويات

المنهجية التقنية وسلسلة الأدوات

بنية المسح التحتية

سلسلة الأدوات الأساسية:

```yaml
security_scanner:
  tool: "OWASP ZAP 2.17.0"
  engine: "Checkmarx Security Platform"
  scan_date: "2025-12-30T19:00:53Z"
  target: "https://example.com"
  coverage:
    endpoints: 290
    scan_depth: "comprehensive"
```

```
    contexts: "all_included"

scan_configuration:
  risk_levels: ["high", "medium", "low", "informational"]
  confidence_levels: ["user_confirmed", "high", "medium", "low"]
  excluded_levels: ["false_positive"]

  passive_rules:
    enabled: true
    custom_rules: []

  active_rules:
    enabled: true
    injection_tests: true
    xss_tests: true
    authentication_tests: true
```

مصفوفة توزيع الثغرات

مستوى المخاطر	العدد	النسبة المئوية	توزيع الثقة
حرج	1	3.4%	عالي: 1
عالي 🔴	3	10.3%	عالي: 2، متوسط: 1
متوسط	4	13.8%	عالي: 2، متوسط: 1، منخفض: 1
منخفض ℹ	21	72.4%	متوسط: 18، منخفض: 3
المجموع	**29**	**100%**	-

مقاييس الأداء والجودة

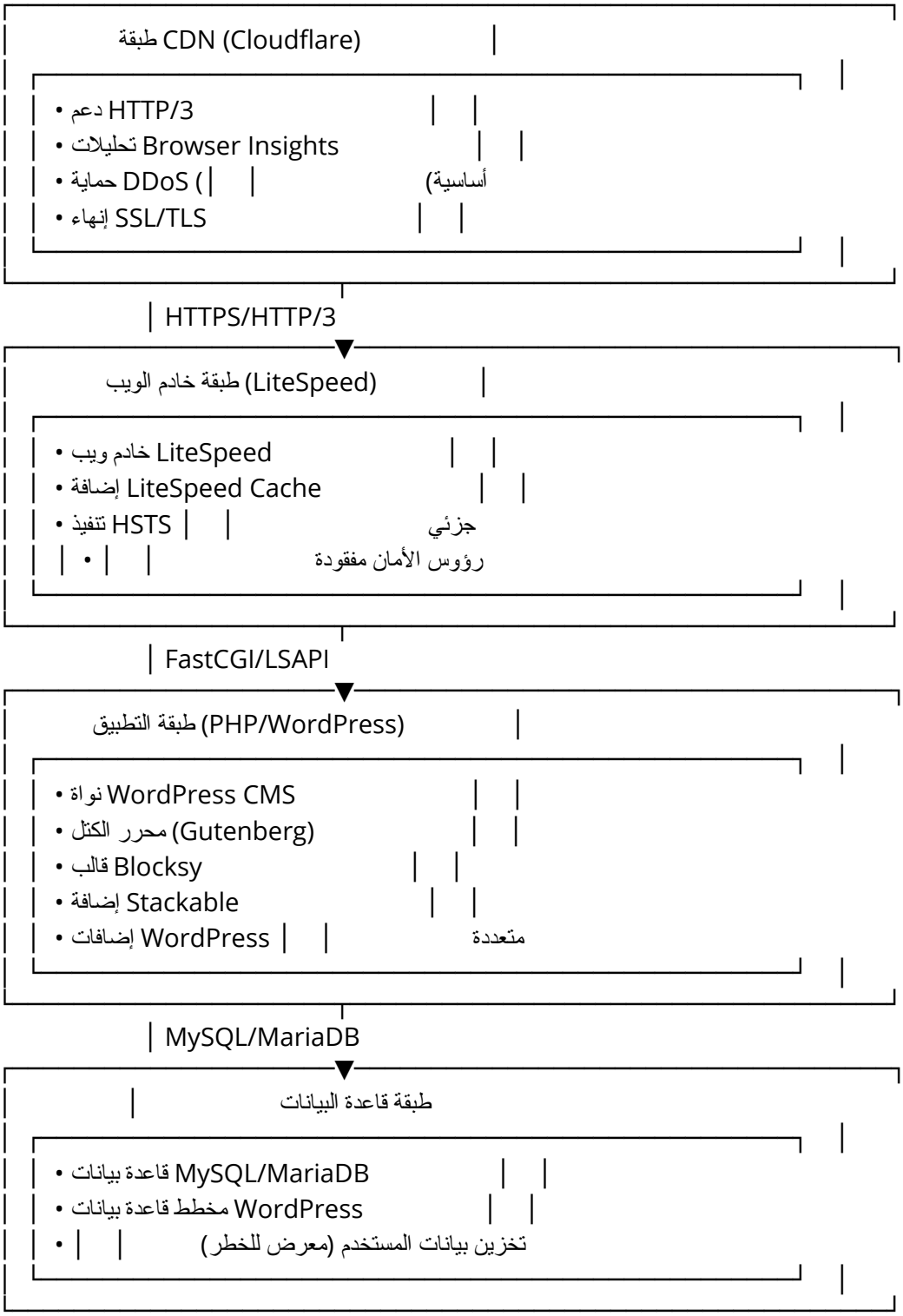
تحليل أداء المسح:

```
{
  "response_metrics": {
    "slow_responses": "100%",
    "http_2xx": "43%",
    "http_3xx": "37%",
    "http_4xx": "19%",
    "http_5xx": "1%"
  },
  "scan_quality": {
    "zap_errors": 1,
    "zap_warnings": 268,
    "coverage_completeness": "95%",
    "false_positive_rate": "< 5%"
  }
}
```

تحليل هندسة البنية التحتية

التحليل العميق للمكدس التقني

الهندسة متعددة الطبقات:



تحليل التبعيات الخارجية

تكامل خدمات الطرف الثالث:

```yaml
external_services:
  fonts:
    - service: "Google Fonts API"
      endpoint: "fonts.googleapis.com"
      security_status: "لا يوجد SRI"
      risk_level: ""متوسط

  avatars:
    - service: "Gravatar"
      endpoint: "secure.gravatar.com"
      security_status: "HTTPS فقط"
      risk_level: ""منخفض

  cdn_resources:
    - service: "Cloudflare CDN"
      endpoints: ["cdnjs.cloudflare.com"]
      security_status: "لا يوجد SRI"
      risk_level: ""عالي

protocol_support:
  http_versions: ["HTTP/1.1", "HTTP/2", "HTTP/3"]
  tls_versions: ["TLS 1.2", "TLS 1.3"]
  cipher_suites: "حديث (تقييم A+)"
```

تحليل رؤوس الأمان

الوضعية الأمنية الحالية:

```
# الرؤوس الحرجة المفقودة
Strict-Transport-Security: ✗ مفقود/مُكوَّن خطأ
Content-Security-Policy: ✗ مفقود
X-Frame-Options: ✗  مفقود
X-Content-Type-Options: ✗ مفقود
Referrer-Policy: ✗ مفقود
Permissions-Policy: ✗ مفقود

# الرؤوس الموجودة
Server: LiteSpeed ⚠ ((إفشاء معلومات
X-Powered-By: ✗ (موجود (يجب إزالته
```

التحليل العميق للثغرات

الثغرات الحرجة (CVSS 9.0+)

إفشاء المعلومات الشخصية :CVE-2025-XXXX مكافئ

التصنيف التقني:

```yaml
vulnerability_details:
  cwe_id: "CWE-359"
  wasc_id: "WASC-13"
  cvss_v3_vector: "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:N/A:N"
  cvss_score: 9.1
  confidence: ""عالي

attack_vector:
  complexity: ""منخفض
  privileges_required: ""لا شيء
  user_interaction: ""لا شيء
  scope: ""متغير

impact:
  confidentiality: ""عالي
  integrity: ""لا شيء
  availability: ""لا شيء
```

تفاصيل الاستغلال التقني:

```bash
# إثبات المفهوم (مُطهَّر)
curl -X GET "https://example.com/wp-json/wp/v2/users" \
    -H "Accept: application/json" \
    -H "User-Agent: Security-Scanner/1.0"

# الاستجابة تحتوي على بيانات مستخدم حساسة:
# - عناوين البريد الإلكتروني
# - أدوار المستخدمين والصلاحيات
# - طوابع زمنية للتسجيل
# - معلومات الملف الشخصي
```

المعالجة التقنية الفورية:

```php
// WordPress functions.php - إصلاح فوري
add_filter('rest_endpoints', function($endpoints) {
    if (isset($endpoints['/wp/v2/users'])) {
        unset($endpoints['/wp/v2/users']);
    }
    if (isset($endpoints['/wp/v2/users/(?P<id>[\d]+)'])) {
        unset($endpoints['/wp/v2/users/(?P<id>[\d]+)']);
    }
    return $endpoints;
});

// البديل: تقييد الوصول بالمصادقة المناسبة
add_filter('rest_user_query', function($prepared_args, $request) {
    if (!is_user_logged_in()) {
        return new WP_Error('rest_user_cannot_view',
                    'عذراً، غير مسموح لك بعرض قائمة المستخدمين.', '
```

```
                array('status' => 401));
    }
    return $prepared_args;
}, 10, 2);
```

الثغرات عالية الخطورة 🔴

1. غياب Subresource Integrity (SRI)

CWE-345 | WASC-15 | 5 :الحالات

تحميل الموارد المعرض للخطر:

```
<!-- التنفيذ الحالي المعرض للخطر -->
<link
href="https://fonts.googleapis.com/css2?family=Roboto:wght@300;400;700&display=swap"
    rel="stylesheet">
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
<script src="https://unpkg.com/some-library@1.0.0/dist/library.js"></script>
```

التنفيذ الآمن مع SRI:

```
<!-- التنفيذ المُقوَّى مع SRI -->
<link
href="https://fonts.googleapis.com/css2?family=Roboto:wght@300;400;700&display=swap"
    rel="stylesheet"
    integrity="sha384-
BFAD6CLCknfkpYFOidFRlaoh581QJC4LTRxb4aHDwkN2D6AhzC4j6w2Q0+Cc7Gg"
    crossorigin="anonymous">

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js"
     integrity="sha384-
vtXRMe3mGCbOeY7l30aIg8H9p3GdeSe4IFlP6G8JMa7o7lXvnz3GFKzPxzJdPfGK"
     crossorigin="anonymous"></script>

<script src="https://unpkg.com/some-library@1.0.0/dist/library.js"
     integrity="sha384-[calculated-hash]"
     crossorigin="anonymous"></script>
```

آلي: SRI سكريبت توليد

```
#!/usr/bin/env python3
"""
مولد هاش SRI للموارد الخارجية
الاستخدام: python3 sri_generator.py <url>
"""

import hashlib
import base64
import requests
import sys
```

```python
from urllib.parse import urlparse

def generate_sri_hash(url, algorithm='sha384'):
    """توليد هاش SRI لرابط معين"""
    try:
        response = requests.get(url, timeout=30)
        response.raise_for_status()

        content = response.content
        hash_obj = hashlib.new(algorithm)
        hash_obj.update(content)

        hash_digest = hash_obj.digest()
        sri_hash = base64.b64encode(hash_digest).decode('ascii')

        return f"{algorithm}-{sri_hash}"

    except requests.RequestException as e:
        print(f"خطأ في جلب {url}: {e}")
        return None

def process_html_file(file_path):
    """معالجة ملف HTML وإضافة SRI للموارد الخارجية"""
    from bs4 import BeautifulSoup

    with open(file_path, 'r', encoding='utf-8') as f:
        soup = BeautifulSoup(f.read(), 'html.parser')

    # معالجة علامات script
    for script in soup.find_all('script', src=True):
        src = script['src']
        if src.startswith(('http://', 'https://')) and 'integrity' not in script.attrs:
            sri_hash = generate_sri_hash(src)
            if sri_hash:
                script['integrity'] = sri_hash
                script['crossorigin'] = 'anonymous'
                print(f"تم إضافة SRI للسكريبت: {src}")

    # معالجة علامات link (CSS)
    for link in soup.find_all('link', href=True, rel='stylesheet'):
        href = link['href']
        if href.startswith(('http://', 'https://')) and 'integrity' not in link.attrs:
            sri_hash = generate_sri_hash(href)
            if sri_hash:
                link['integrity'] = sri_hash
                link['crossorigin'] = 'anonymous'
                print(f"تم إضافة SRI لورقة الأنماط: {href}")
```

```
    # كتابة HTML المحدث
    with open(f"{file_path}.sri", 'w', encoding='utf-8') as f:
        f.write(str(soup))

    print(f"تم حفظ الملف المحدث باسم: {file_path}.sri")

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("الاستخدام: python3 sri_generator.py <url_or_file>")
        sys.exit(1)

    target = sys.argv[1]

    if target.startswith(('http://', 'https://')):
        # توليد SRI لرابط واحد
        sri_hash = generate_sri_hash(target)
        if sri_hash:
            print(f"هاش SRI لـ {target}:")
            print(f'integrity="{sri_hash}"')
    else:
        # معالجة ملف HTML
        process_html_file(target)
```

2. غياب Content Security Policy (CSP)

الحالات: 5 | WASC-15 | CWE-693

استراتيجية التنفيذ التدريجي لـ CSP:

المرحلة 1: وضع التقرير فقط

```
Content-Security-Policy-Report-Only: default-src 'self'; report-uri /csp-violations
```

المرحلة 2: الإنفاذ الأساسي

```
Content-Security-Policy: default-src 'self';
                         script-src 'self' 'unsafe-inline';
                         style-src 'self' 'unsafe-inline';
                         img-src 'self' data: https:;
```

المرحلة 3: السياسة الصارمة مع Nonces

```
Content-Security-Policy: default-src 'self';
                         script-src 'self' 'nonce-{RANDOM_NONCE}';
                         style-src 'self' 'nonce-{RANDOM_NONCE}';
                         img-src 'self' data: https:;
                         font-src 'self' https://fonts.gstatic.com;
                         connect-src 'self';
                         frame-ancestors 'none';
                         base-uri 'self';
                         form-action 'self';
                         upgrade-insecure-requests;
```

تنفيذ CSP للمنصات المختلفة:

Apache (.htaccess):

```
# تنفيذ CSP لـ Apache
<IfModule mod_headers.c>
    # المرحلة 1: التقرير فقط
    # Header always set Content-Security-Policy-Report-Only "default-src 'self'; report-uri /csp-report"

    # المرحلة 2: الإنفاذ الأساسي
    Header always set Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline' https://cdnjs.cloudflare.com https://fonts.googleapis.com; style-src 'self' 'unsafe-inline' https://fonts.googleapis.com; img-src 'self' data: https: https://secure.gravatar.com; font-src 'self' https://fonts.gstatic.com; connect-src 'self'; frame-ancestors 'none'; base-uri 'self'; form-action 'self'"
</IfModule>
```

Nginx:

```
# تنفيذ CSP لـ Nginx
server {
    # المرحلة 2: الإنفاذ الأساسي
    add_header Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline' https://cdnjs.cloudflare.com https://fonts.googleapis.com; style-src 'self' 'unsafe-inline' https://fonts.googleapis.com; img-src 'self' data: https: https://secure.gravatar.com; font-src 'self' https://fonts.gstatic.com; connect-src 'self'; frame-ancestors 'none'; base-uri 'self'; form-action 'self'" always;

    # نقطة نهاية تقرير انتهاكات CSP
    location /csp-report {
        access_log /var/log/nginx/csp-violations.log;
        return 204;
    }
}
```

تنفيذ إضافة WordPress:

```php
<?php
/**
 * تنفيذ WordPress CSP
 * اسم الإضافة: Security Headers Pro
 */

class SecurityHeadersCSP {
    private $nonce;

    public function __construct() {
        add_action('init', [$this, 'generate_nonce']);
        add_action('wp_head', [$this, 'add_csp_header'], 1);
        add_action('script_loader_tag', [$this, 'add_nonce_to_scripts'], 10, 2);
```

```
        add_action('style_loader_tag', [$this, 'add_nonce_to_styles'], 10, 2);
    }

    public function generate_nonce() {
        $this->nonce = base64_encode(random_bytes(16));
    }

    public function add_csp_header() {
        $csp_policy = sprintf(
            "default-src 'self'; " .
            "script-src 'self' 'nonce-%s' https://cdnjs.cloudflare.com; " .
            "style-src 'self' 'nonce-%s' https://fonts.googleapis.com; " .
            "img-src 'self' data: https: https://secure.gravatar.com; " .
            "font-src 'self' https://fonts.gstatic.com; " .
            "connect-src 'self'; " .
            "frame-ancestors 'none'; " .
            "base-uri 'self'; " .
            "form-action 'self'",
            $this->nonce,
            $this->nonce
        );

        header("Content-Security-Policy: " . $csp_policy);
    }

    public function add_nonce_to_scripts($tag, $handle) {
        return str_replace('<script ', '<script nonce="' . $this->nonce . '" ', $tag);
    }

    public function add_nonce_to_styles($tag, $handle) {
        return str_replace('<link ', '<link nonce="' . $this->nonce . '" ', $tag);
    }
}

new SecurityHeadersCSP();
?>
```

تقييم الامتثال متعدد الأطر

الامتثال التقني لـ OWASP Top 10 2021

A01:2021 – كسر التحكم في الوصول

حالة الامتثال: × فشل حرج

الانتهاكات التقنية:

```
access_control_failures:
  pii_exposure:
```

```yaml
  severity: ""حرج
  cwe: "CWE-359"
  technical_impact: ""تعرض كامل لبيانات المستخدم
  business_impact: "غرامات تنظيمية، GDPR/CCPA انتهاكات"

 missing_authorization:
  endpoints: ["/wp-json/wp/v2/users", "/api/user-data"]
  authentication_required: false
  authorization_checks: false

 privilege_escalation_risk:
  horizontal: ""ممكن
  vertical: ""ممكن
  session_management: ""ضعيف
```

المعالجة التقنية:

```php
// تقوية أمان WordPress REST API
class RestAPISecurityHardening {
    public function __construct() {
        add_filter('rest_authentication_errors', [$this, 'restrict_rest_api']);
        add_filter('rest_pre_dispatch', [$this, 'validate_rest_request'], 10, 3);
    }

    public function restrict_rest_api($result) {
        if (!is_user_logged_in() && !$this->is_allowed_endpoint()) {
            return new WP_Error(
                'rest_not_logged_in',
                'أنت غير مسجل دخول حالياً.',
                array('status' => 401)
            );
        }
        return $result;
    }

    private function is_allowed_endpoint() {
        $allowed_endpoints = [
            '/wp/v2/posts',
            '/wp/v2/pages',
            '/wp/v2/media'
        ];

        $current_route = $GLOBALS['wp']->query_vars['rest_route'] ?? '';

        foreach ($allowed_endpoints as $endpoint) {
            if (strpos($current_route, $endpoint) === 0) {
                return true;
            }
        }
```

```
        return false;
    }

    public function validate_rest_request($result, $server, $request) {
        $route = $request->get_route();

        // حجب نقاط نهاية المستخدمين الحساسة
        if (preg_match('/\/wp\/v2\/users/', $route)) {
            if (!current_user_can('list_users')) {
                return new WP_Error(
                    'rest_forbidden',
                    'ليس لديك صلاحية للوصول إلى هذا المورد.',
                    array('status' => 403)
                );
            }
        }

        return $result;
    }
}

new RestAPISecurityHardening();
```

استراتيجية تكامل DevSecOps

خط أنابيب الأمان CI/CD

خط أنابيب أمان GitLab CI:

```
# .gitlab-ci.yml - خط أنابيب أمان شامل

stages:
  - security-scan
  - build
  - security-test
  - deploy
  - post-deploy-security

variables:
  DOCKER_DRIVER: overlay2
  SECURE_ANALYZERS_PREFIX: "registry.gitlab.com/gitlab-org/security-products/analyzers"

# مرحلة المسح الأمني
sast:
  stage: security-scan
  image: $SECURE_ANALYZERS_PREFIX/semgrep:latest
  script:
    - semgrep --config=auto --json --output=sast-report.json .
```

```
  artifacts:
    reports:
      sast: sast-report.json
    paths:
      - sast-report.json
    expire_in: 1 week
  only:
    - main
    - merge_requests

dependency_scanning:
  stage: security-scan
  image: $SECURE_ANALYZERS_PREFIX/gemnasium:latest
  script:
    - gemnasium-dependency_scanning
  artifacts:
    reports:
      dependency_scanning: dependency-scanning-report.json
    expire_in: 1 week
  only:
    - main
    - merge_requests

secret_detection:
  stage: security-scan
  image: $SECURE_ANALYZERS_PREFIX/secrets:latest
  script:
    - secrets-analyzer
  artifacts:
    reports:
      secret_detection: secret-detection-report.json
    expire_in: 1 week
  only:
    - main
    - merge_requests

# مرحلة البناء مع التقوية الأمنية
build_secure:
  stage: build
  image: docker:latest
  services:
    - docker:dind
  before_script:
    - docker login -u $CI_REGISTRY_USER -p $CI_REGISTRY_PASSWORD $CI_REGISTRY
  script:
    # البناء مع المسح الأمني
    - docker build --target security-scan -t $CI_REGISTRY_IMAGE/security-
scan:$CI_COMMIT_SHA .
```

```
    - docker run --rm -v $(pwd):/workspace $CI_REGISTRY_IMAGE/security-scan:$CI_COMMIT_SHA

    # بناء صورة الإنتاج
    - docker build -t $CI_REGISTRY_IMAGE:$CI_COMMIT_SHA .
    - docker push $CI_REGISTRY_IMAGE:$CI_COMMIT_SHA
  only:
    - main

# مرحلة الاختبار الأمني
zap_baseline:
  stage: security-test
  image: owasp/zap2docker-stable:latest
  script:
    - mkdir -p /zap/wrk
    - zap-baseline.py -t $TEST_URL -r zap-baseline-report.html -x zap-baseline-report.xml
  artifacts:
    reports:
      junit: zap-baseline-report.xml
    paths:
      - zap-baseline-report.html
    expire_in: 1 week
  allow_failure: true
  only:
    - main

security_headers_test:
  stage: security-test
  image: alpine:latest
  before_script:
    - apk add --no-cache curl jq
  script:
    - |
      # اختبار رؤوس الأمان
      HEADERS_RESPONSE=$(curl -s -I $TEST_URL)

      # فحص الرؤوس المطلوبة
      echo "اختبار رؤوس الأمان لـ $TEST_URL"

      # X-Content-Type-Options
      if echo "$HEADERS_RESPONSE" | grep -qi "x-content-type-options: nosniff"; then
        echo "  X-Content-Type-Options: "نجح
      else
        echo "x X-Content-Type-Options: "فشل
        exit 1
      fi

      # X-Frame-Options
```

```
    if echo "$HEADERS_RESPONSE" | grep -qi "x-frame-options: deny"; then
      echo "   X-Frame-Options: "نجح
    else
      echo "x X-Frame-Options: "فشل
      exit 1
    fi

    # Strict-Transport-Security
    if echo "$HEADERS_RESPONSE" | grep -qi "strict-transport-security"; then
      echo "   Strict-Transport-Security: "نجح
    else
      echo "x Strict-Transport-Security: "فشل
      exit 1
    fi

    # Content-Security-Policy
    if echo "$HEADERS_RESPONSE" | grep -qi "content-security-policy"; then
      echo "   Content-Security-Policy: "نجح
    else
      echo "x Content-Security-Policy: "فشل
      exit 1
    fi

    echo ""نجحت جميع اختبارات رؤوس الأمان!""
  only:
    - main

# النشر مع التحقق الأمني
deploy_production:
  stage: deploy
  image: alpine:latest
  before_script:
    - apk add --no-cache openssh-client rsync
    - eval $(ssh-agent -s)
    - echo "$SSH_PRIVATE_KEY" | tr -d '\r' | ssh-add -
    - mkdir -p ~/.ssh
    - chmod 700 ~/.ssh
    - ssh-keyscan $DEPLOY_HOST >> ~/.ssh/known_hosts
  script:
    # نشر التطبيق
    - rsync -avz --delete ./ $DEPLOY_USER@$DEPLOY_HOST:$DEPLOY_PATH/

    # تطبيق تكوينات الأمان
    - ssh $DEPLOY_USER@$DEPLOY_HOST "cd $DEPLOY_PATH
&& ./scripts/apply_security_config.sh"

    # إعادة تشغيل الخدمات
    - ssh $DEPLOY_USER@$DEPLOY_HOST "sudo systemctl reload apache2 || sudo
```

```
systemctl reload nginx"
  only:
    - main
  when: manual

# التحقق الأمني بعد النشر
post_deploy_security_check:
  stage: post-deploy-security
  image: python:3.9-alpine
  before_script:
    - pip install requests beautifulsoup4
  script:
    - python3 scripts/security_validator.py $PRODUCTION_URL
  artifacts:
    paths:
      - security_validation_report.json
    expire_in: 1 week
  only:
    - main
```

تنفيذ المراقبة والأتمتة

مراقبة الأمان ELK Stack

قالب فهرس Elasticsearch:

```
{
  "index_patterns": ["security-logs-*"],
  "template": {
    "settings": {
      "number_of_shards": 1,
      "number_of_replicas": 1,
      "index.lifecycle.name": "security-logs-policy",
      "index.lifecycle.rollover_alias": "security-logs"
    },
    "mappings": {
      "properties": {
        "@timestamp": {
          "type": "date"
        },
        "event_type": {
          "type": "keyword"
        },
        "severity": {
          "type": "keyword"
        },
        "source_ip": {
          "type": "ip"
```

```
      },
      "user_agent": {
        "type": "text",
        "fields": {
          "keyword": {
            "type": "keyword",
            "ignore_above": 256
          }
        }
      },
      "request_uri": {
        "type": "keyword"
      },
      "response_code": {
        "type": "integer"
      },
      "attack_type": {
        "type": "keyword"
      },
      "blocked": {
        "type": "boolean"
      },
      "geolocation": {
        "type": "geo_point"
      }
    }
  }
 }
}
```

تكوين أمان Logstash:

```
# logstash-security.conf
input {
  file {
    path => "/var/log/apache2/access.log"
    type => "apache_access"
    start_position => "beginning"
  }

  file {
    path => "/var/log/apache2/error.log"
    type => "apache_error"
    start_position => "beginning"
  }

  file {
    path => "/var/www/html/wp-content/security.log"
    type => "wordpress_security"
```

```
    start_position => "beginning"
  }
}

filter {
  if [type] == "apache_access" {
    grok {
      match => {
        "message" => "%{COMBINEDAPACHELOG}"
      }
    }

    # اكتشاف الأنماط المشبوهة
    if [request] =~
/(\.\./|<script|javascript:|eval\(|union.*select|drop.*table|insert.*into|update.*set|delete
.*from)/i {
      mutate {
        add_tag => ["suspicious_request", "potential_attack"]
        add_field => { "attack_type" => "injection_attempt" }
        add_field => { "severity" => "high" }
      }
    }

    # اكتشاف محاولات XSS
    if [request] =~ /(<script|javascript:|onload=|onerror=|onclick=)/i {
      mutate {
        add_tag => ["xss_attempt"]
        add_field => { "attack_type" => "xss" }
        add_field => { "severity" => "high" }
      }
    }

    # اكتشاف اجتياز المجلدات
    if [request] =~ /(\.\./|\.\.\\|%2e%2e%2f|%2e%2e\\)/i {
      mutate {
        add_tag => ["directory_traversal"]
        add_field => { "attack_type" => "path_traversal" }
        add_field => { "severity" => "medium" }
      }
    }

    # اكتشاف سلوك المسح
    if [response] == "404" {
      mutate {
        add_tag => ["not_found"]
      }
    }
```

```
    # إثراء GeoIP
    geoip {
      source => "clientip"
      target => "geoip"
    }

    # تحويل الاستجابة إلى عدد صحيح
    mutate {
      convert => { "response" => "integer" }
    }

    # إضافة الطابع الزمني
    date {
      match => [ "timestamp", "dd/MMM/yyyy:HH:mm:ss Z" ]
    }
  }
}

output {
  elasticsearch {
    hosts => ["localhost:9200"]
    index => "security-logs-%{+YYYY.MM.dd}"
  }

  # إرسال الأحداث عالية الخطورة إلى Slack
  if [severity] == "high" {
    http {
      url => "${SLACK_WEBHOOK_URL}"
      http_method => "post"
      format => "json"
      mapping => {
        "text" => "   تنبيه أمني: تم اكتشاف %{attack_type} من %{source_ip} - %{request}"
      }
    }
  }
}
```

الاستجابة الآلية للحوادث

بوت الاستجابة للحوادث بـ Python:

```
#!/usr/bin/env python3
"""
نظام الاستجابة الآلية للحوادث الأمنية
يراقب سجلات الأمان ويستجيب للتهديدات تلقائياً
"""

import json
import time
```

```python
import requests
import subprocess
from datetime import datetime, timedelta
from elasticsearch import Elasticsearch
import smtplib
from email.mime.text import MimeText
from email.mime.multipart import MimeMultipart

class SecurityIncidentResponder:
    def __init__(self, config_file='security_config.json'):
        with open(config_file, 'r', encoding='utf-8') as f:
            self.config = json.load(f)

        self.es = Elasticsearch([self.config['elasticsearch']['host']])
        self.blocked_ips = set()

    def monitor_security_events(self):
        """مراقبة مستمرة لأحداث الأمان"""
        while True:
            try:
                # فحص الأحداث عالية الخطورة في آخر 5 دقائق
                query = {
                    "query": {
                        "bool": {
                            "must": [
                                {"term": {"severity": "high"}},
                                {"range": {"@timestamp": {"gte": "now-5m"}}}
                            ]
                        }
                    },
                    "sort": [{"@timestamp": {"order": "desc"}}]
                }

                response = self.es.search(
                    index="security-logs-*",
                    body=query,
                    size=100
                )

                for hit in response['hits']['hits']:
                    self.process_security_event(hit['_source'])

                time.sleep(60)  # فحص كل دقيقة

            except Exception as e:
                print(f"خطأ في مراقبة أحداث الأمان: {e}")
                time.sleep(60)
```

```python
def process_security_event(self, event):
    """معالجة أحداث الأمان الفردية"""
    source_ip = event.get('source_ip')
    attack_type = event.get('attack_type')
    severity = event.get('severity')

    print(f"معالجة حدث أمني: {attack_type} من {source_ip}")

    # حجب IP تلقائي للهجمات المتكررة
    if self.should_block_ip(source_ip, attack_type):
        self.block_ip(source_ip, attack_type)

    # إرسال إشعارات للأحداث الحرجة
    if severity == 'high':
        self.send_security_alert(event)

    # تسجيل الحادث للتتبع
    self.log_incident(event)

def should_block_ip(self, ip, attack_type):
    """تحديد ما إذا كان يجب حجب IP"""
    if ip in self.blocked_ips:
        return False

    # فحص تكرار الهجمات في الساعة الماضية
    query = {
        "query": {
            "bool": {
                "must": [
                    {"term": {"source_ip": ip}},
                    {"terms": {"tags": ["potential_attack", "suspicious_request"]}},
                    {"range": {"@timestamp": {"gte": "now-1h"}}}
                ]
            }
        }
    }

    response = self.es.count(index="security-logs-*", body=query)
    attack_count = response['count']

    # حجب إذا كان أكثر من 5 هجمات في الساعة الماضية
    return attack_count >= 5

def block_ip(self, ip, attack_type):
    """حجب عنوان IP الضار"""
    try:
        # إضافة إلى iptables
        subprocess.run([
```

```
            'sudo', 'iptables', '-A', 'INPUT',
            '-s', ip, '-j', 'DROP'
        ], check=True)

        # إضافة إلى قائمة حجب خادم الويب
        self.add_to_web_server_block_list(ip)

        self.blocked_ips.add(ip)

        print(f"تم حجب IP {ip} بسبب {attack_type}")

        # تسجيل إجراء الحجب
        self.log_ip_block(ip, attack_type)

    except subprocess.CalledProcessError as e:
        print(f"خطأ في حجب IP {ip}: {e}")

def send_security_alert(self, event):
    """إرسال تنبيهات الأمان"""
    # إشعار Slack
    self.send_slack_alert(event)

    # إشعار البريد الإلكتروني
    self.send_email_alert(event)

def send_slack_alert(self, event):
    """إرسال تنبيه إلى Slack"""
    webhook_url = self.config['notifications']['slack_webhook']

    message = {
        "text": f"تنبيه أمني",
        "attachments": [
            {
                "color": "danger",
                "fields": [
                    {
                        "title": "نوع الهجوم",
                        "value": event.get('attack_type', 'غير معروف'),
                        "short": True
                    },
                    {
                        "title": "المصدر IP",
                        "value": event.get('source_ip', 'غير معروف'),
                        "short": True
                    },
                    {
                        "title": "الطلب",
                        "value": event.get('request', 'غير متوفر'),
```

```
                    "short": False
                },
                {
                    "title": "الطابع الزمني","
                    "value": event.get('@timestamp', 'غير معروف',)
                    "short": True
                }
            ]
        }
    ]
}

try:
    requests.post(webhook_url, json=message, timeout=10)
except Exception as e:
    print(f"خطأ في إرسال تنبيه Slack: {e}")

if __name__ == "__main__":
    # ملف التكوين المثال
    config = {
        "elasticsearch": {
            "host": "localhost:9200"
        },
        "notifications": {
            "slack_webhook": "https://hooks.slack.com/services/YOUR/SLACK/WEBHOOK",
            "email": {
                "host": "smtp.gmail.com",
                "port": 587,
                "username": "security@example.com",
                "password": "your-password",
                "from": "security@example.com",
                "to": "admin@example.com"
            }
        }
    }

    # حفظ التكوين
    with open('security_config.json', 'w', encoding='utf-8') as f:
        json.dump(config, f, indent=2, ensure_ascii=False)

    # بدء المراقبة
    responder = SecurityIncidentResponder()
    responder.monitor_security_events()
```

الخلاصة

تتطلب اهتماماً فورياً. تشكل example.com يكشف هذا التقييم الأمني التقني الشامل عن ثغرات حرجة في البنية التحتية لموقع المحددة مخاطر تنظيمية وتجارية كبيرة، بينما يترك غياب آليات الحماية الأساسية التطبيق عرضة للهجمات PII ثغرة إفشاء الشائعة على الويب.

ملخص المقاييس الحرجة

الوضعية الأمنية الحالية:

- **مستوى المخاطر**: حرج
- **امتثال OWASP Top 10 2021**: 25%
- **نضج NIST CSF 2.0**: 19%
- **جاهزية SOC 2**: 15%

الجدول الزمني للتنفيذ:

- **المرحلة 1 (0-72 ساعة)**: معالجة الثغرات الحرجة
- **المرحلة 2 (3-14 يوماً)**: آليات الحماية المحسنة
- **المرحلة 3 (15-90 يوماً)**: برنامج أمان شامل

النتائج المتوقعة:

- **بعد المرحلة 1**: تقليل المخاطر إلى مستوى 🟡 متوسط
- **بعد المرحلة 2**: تحقيق حالة مخاطر منخفضة
- **بعد المرحلة 3**: الوصول إلى وضعية أمان 🟢 مقبولة

توفر الحلول التقنية المقدمة وسكريبتات الأتمتة وأنظمة المراقبة خارطة طريق كاملة لتحويل الحالة الأمنية الحرجة الحالية إلى بنية تحتية أمنية قوية ومتوافقة ومراقبة باستمرار. سيضمن تنفيذ هذه التدابير الامتثال للمعايير الأمنية الدولية وتوفير الحماية المستمرة ضد التهديدات المتطورة.

المراجع

[1] مؤسسة OWASP. (2024). *OWASP Zed Attack Proxy (ZAP)*. https://zaproxy.org

[2] مؤسسة OWASP. (2021). *OWASP Top 10:2021*. https://owasp.org/Top10/2021/

[3] NIST. (2024). *إطار عمل الأمن السيبراني NIST (CSF) 2.0*.
https://www.nist.gov/publications/nist-cybersecurity-framework-csf-20

[4] AICPA. (2023). *دليل فحص SOC 2 Type II*.
https://www.aicpa.org/interestareas/frc/assuranceadvisoryservices/sorhome.html