🔒 HỆ THỐNG BẢO MẬT - TẠP CHÍ HẬU CẦN QUÂN SỰ

📋 TỔNG QUAN

Hệ thống bảo mật của Tạp chí Hậu cần quân sự được thiết kế tuân thủ các tiêu chuẩn:
- **OWASP Top 10 Security Practices**
- **Thông tư 41/2022/TT-BTTT** về an ninh mạng
- **ISO/IEC 27001** - Quản lý an ninh thông tin
- **Best practices** cho ứng dụng web hiện đại

🛡️ CÁC LỚP BẢO MẬT

1. AUTHENTICATION & AUTHORIZATION

JWT Token Security

```
// Access Token: 8 giờ
// Refresh Token: 7 ngày
// Bcrypt: 12 rounds
```

Tính năng:
- ✅ Dual token system (access + refresh)
- ✅ Secure cookie storage (httpOnly, sameSite)
- ✅ Token rotation on refresh
- ✅ Automatic token expiration
- ✅ Role-based access control (RBAC)

Các vai trò hệ thống:
- `READER` - Người đọc
- `AUTHOR` - Tác giả
- `REVIEWER` - Phản biện viên
- `SECTION_EDITOR` - Biên tập viên chuyên mục
- `MANAGING_EDITOR` - Tổng biên tập điều hành
- `LAYOUT_EDITOR` - Biên tập viên trình bày
- `EIC` - Tổng biên tập
- `SYSADMIN` - Quản trị hệ thống
- `SECURITY_AUDITOR` - Kiểm định bảo mật

2. INPUT VALIDATION & SANITIZATION

Thư viện: `/lib/validation.ts`

Chống các cuộc tấn công:
- ✅ XSS (Cross-Site Scripting)
- ✅ SQL Injection
- ✅ Command Injection

- ✅ Path Traversal
- ✅ LDAP Injection

Validation schemas:

```
- registerSchema: Đăng ký user
- loginSchema: Đăng nhập
- submissionSchema: Nộp bài
- reviewSchema: Phản biện
- fileUploadSchema: Upload file
- searchSchema: Tìm kiếm
```

3. CSRF PROTECTION

Thư viện: `/lib/csrf.ts`

Phương pháp: Double Submit Cookie Pattern

Cơ chế hoạt động:
1. Server tạo CSRF token và lưu vào cookie
2. Client lấy token từ endpoint `/api/csrf`
3. Client gửi token trong header `x-csrf-token`
4. Server so sánh token từ cookie và header

Sử dụng:

```
// Client side
const response = await fetch('/api/csrf')
const { token } = await response.json()

// Include in requests
fetch('/api/endpoint', {
  method: 'POST',
  headers: {
    'x-csrf-token': token
  }
})
```

4. RATE LIMITING

Thư viện: `/lib/rate-limiter.ts`

Hỗ trợ:
- ✅ Redis (production, multi-instance)
- ✅ In-memory fallback (single instance)

Giới hạn mặc định:
- API endpoints: 120 requests/minute
- Login: 5 failed attempts trong 15 phút
- File upload: 50 uploads/hour

Tự động:
- Rate limit headers trong response
- Cleanup expired records
- Fallback khi Redis không khả dụng

5. FILE UPLOAD SECURITY

Thư viện: `/lib/file-security.ts`

Bảo vệ:
- ✅ File type validation (MIME type + magic bytes)
- ✅ File size limits (50MB default)
- ✅ Filename sanitization
- ✅ Executable file detection
- ✅ Content scanning (for text files)
- ✅ Secure filename generation
- ✅ File hash for deduplication

Allowed file types:
- Documents: PDF, DOC, DOCX, XLS, XLSX
- Images: JPG, PNG, GIF, WEBP
- Text: TXT

Cấm:
- Executable files (.exe, .bat, .sh, etc.)
- Script files (.js, .vbs, etc.)
- Double extensions (file.pdf.exe)

6. SECURITY HEADERS

Thư viện: `/lib/security-headers.ts`

Headers được áp dụng:

```
Content-Security-Policy: Ngăn XSS
Strict-Transport-Security: Force HTTPS (production)
X-Frame-Options: DENY - Ngăn clickjacking
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Referrer-Policy: strict-origin-when-cross-origin
Permissions-Policy: Hạn chế quyền truy cập thiết bị
```

7. AUDIT LOGGING

Thư viện: `/lib/audit-logger.ts`

Ghi lại:
- Tất cả hoạt động authentication
- Thay đổi dữ liệu quan trọng
- Truy cập bị từ chối
- Hành vi đáng ngờ
- Thay đổi cấu hình hệ thống

Lưu trữ: PostgreSQL database
Retention: Tuân theo quy định TT41

8. SECURITY MONITORING

Thư viện: `/lib/security-monitor.ts`

Phát hiện:
- ✅ Brute force attacks
- ✅ SQL injection attempts
- ✅ XSS attempts
- ✅ Path traversal attempts
- ✅ Suspicious IP changes
- ✅ Excessive password changes

Cảnh báo tự động:
- `INFO` - Thông tin
- `WARNING` - Cảnh báo
- `CRITICAL` - Nghiêm trọng

Thresholds:

```
FAILED_LOGIN_ATTEMPTS: 5 trong 15 phút
SUSPICIOUS_IP_CHANGES: 5 IP khác nhau trong 1 giờ
PASSWORD_CHANGES_PER_DAY: 3 lần
API_REQUESTS_PER_MINUTE: 100 requests
```

🚀 TRIỂN KHAI TRÊN VPS/CLOUD

Yêu cầu môi trường

```
# Environment variables (.env)
DATABASE_URL=postgresql://...
NEXTAUTH_SECRET=<secure-random-string>
JWT_SECRET=<secure-random-string>
JWT_REFRESH_SECRET=<secure-random-string>

# Optional: Redis for rate limiting
UPSTASH_REDIS_REST_URL=...
UPSTASH_REDIS_REST_TOKEN=...

# AWS S3 for file storage
AWS_PROFILE=hosted_storage
AWS_REGION=us-west-2
AWS_BUCKET_NAME=...
AWS_FOLDER_PREFIX=...
```

Checklist triển khai

Trước khi deploy:
- [ ] Đổi tất cả secrets trong .env
- [ ] Kiểm tra database connection
- [ ] Cấu hình SSL certificate
- [ ] Thiết lập backup tự động
- [ ] Cấu hình firewall
- [ ] Thiết lập monitoring

Bảo mật server:

```
# Firewall
ufw allow 22/tcp # SSH
ufw allow 80/tcp # HTTP
ufw allow 443/tcp # HTTPS
ufw enable

# Fail2ban
apt install fail2ban
systemctl enable fail2ban

# Automatic updates
apt install unattended-upgrades
dpkg-reconfigure -plow unattended-upgrades
```

NGINX Configuration:

```
server {
    listen 443 ssl http2;
    server_name yourdomain.com;

    ssl_certificate /path/to/cert.pem;
    ssl_certificate_key /path/to/key.pem;

    # Security headers
    add_header X-Frame-Options "DENY" always;
    add_header X-Content-Type-Options "nosniff" always;
    add_header X-XSS-Protection "1; mode=block" always;
    add_header Referrer-Policy "strict-origin-when-cross-origin" always;

    # Rate limiting
    limit_req_zone $binary_remote_addr zone=api:10m rate=10r/s;
    limit_req zone=api burst=20;

    location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}
```

📊 GIÁM SÁT & BÁO CÁO

Metrics cần theo dõi

1. **Security Metrics:**
   - Failed login attempts
   - Rate limit violations
   - CSRF token failures
   - File upload rejections

2. **Performance Metrics:**
   - API response time
   - Database query time
   - File upload speed

3. **Audit Logs:**
   - Truy cập vào `/api/audit-logs`
   - Dashboard admin: `/dashboard/admin/security`

Alerts

Cấu hình alerts cho:
- [ ] 5+ failed logins trong 5 phút
- [ ] SQL injection attempts
- [ ] XSS attempts
- [ ] Excessive requests từ 1 IP
- [ ] Suspicious file uploads

🔧 BẢO TRÌ & CẬP NHẬT

Hàng tuần

- [ ] Review audit logs
- [ ] Check security alerts
- [ ] Monitor failed login attempts

Hàng tháng

- [ ] Update dependencies
- [ ] Review user permissions
- [ ] Test backup restore
- [ ] Security scan

Hàng quý

- [ ] Penetration testing
- [ ] Security audit
- [ ] Update security documentation
- [ ] Training for team

📞 SỰ CỐ BẢO MẬT

Quy trình xử lý

1. **Phát hiện:**
   - Monitor logs/alerts
   - User reports
   - Automated detection

2. **Phản ứng:**
   - Isolate affected systems

- Stop the attack
- Collect evidence

3. **Khắc phục:**
   - Fix vulnerabilities
   - Reset compromised credentials
   - Restore from backup if needed

4. **Báo cáo:**
   - Document incident
   - Analyze root cause
   - Update security measures

Liên hệ

Security Team:
- Email: security@example.com
- Phone: 024.XXXX.XXXX
- Emergency: On-call 24/7

📚 TÀI LIỆU THAM KHẢO

- OWASP Top 10 (https://owasp.org/www-project-top-ten/)
- Thông tư 41/2022/TT-BTTT (https://thuvienphapluat.vn/van-ban/Cong-nghe-thong-tin/Thong-tu-41-2022-TT-BTTT-tieu-chuan-ky-thuat-an-ninh-mang-536558.aspx)
- Next.js Security (https://nextjs.org/docs/advanced-features/security-headers)
- PostgreSQL Security (https://www.postgresql.org/docs/current/security.html)

Cập nhật lần cuối: November 3, 2025
Phiên bản: 1.0.0
Người chịu trách nhiệm: System Administrator