Contents

1 SECURITY DOCUMENTATION **2**

1.1 Document Information 2

1.2 Security Overview 2

1.3 1. AUTHENTICATION & AUTHORIZATION 2

1.3.1 1.1 Authentication Mechanism 2

1.3.2 1.2 Password Policy 3

1.3.3 1.3 Role-Based Access Control (RBAC) 3

1.4 2. APPLICATION SECURITY 4

1.4.1 2.1 OWASP Top 10 Protection 4

1.4.2 2.2 CSRF Protection 6

1.4.3 2.3 XSS Prevention 6

1.4.4 2.4 Input Validation 6

1.5 3. DATA SECURITY 7

1.5.1 3.1 Data Classification 7

1.5.2 3.2 Data Encryption 7

1.5.3 3.3 Data Retention & Deletion 7

1.6 4. NETWORK SECURITY 8

1.6.1 4.1 Firewall Configuration 8

1.6.2 4.2 DDoS Protection 8

1.6.3 4.3 API Security 9

1.7 5. MONITORING & AUDIT 9

1.7.1 5.1 Security Monitoring 9

1.7.2 5.2 Audit Logging 9

1.7.3 5.3 Security Metrics 10

1.8 6. INCIDENT RESPONSE 10

1.8.1 6.1 Incident Classification 10

1.8.2 6.2 Incident Response Plan 10

1.8.3 6.3 Emergency Contacts 10

1.9 7. BACKUP & DISASTER RECOVERY 11

1.9.1 7.1 Backup Strategy 11

1.9.2 7.2 Disaster Recovery 11

1.10 8. COMPLIANCE & POLICIES 12

1.10.1 8.1 Privacy Policy 12

1.10.2 8.2 Acceptable Use Policy 12

1.10.3 8.3 Third-Party Security 12

1.11 9. SECURITY TESTING 12

1.11.1 9.1 Penetration Testing 12

1.11.2 9.2 Code Review 13

1.12 10. SECURITY TRAINING 13

1.12.1 10.1 User Security Awareness 13

1.12.2 10.2 Developer Security Training 13

1.13 SECURITY CONTACT 13

Daftar Isi

1 fi DOKUMENTASI KEAMANAN 2

1.1 ff Informasi Dokumen 2
1.2 ff Ikhtisar Keamanan 2
1.3 ff 1. AUTENTIKASI & OTORISASI 2
1.3.1 1.1 Mekanisme Autentikasi 2
1.3.2 1.2 Kebijakan Kata Sandi 3
1.3.3 1.3 Kontrol Akses Berbasis Peran (RBAC) 3
1.4 ff 2. KEAMANAN APLIKASI 4
1.4.1 2.1 Perlindungan OWASP Top 10 4
1.4.2 2.2 Perlindungan CSRF 6
1.4.3 2.3 Pencegahan XSS 6
1.4.4 2.4 Validasi Input 6
1.5 ff 3. KEAMANAN DATA 7
1.5.1 3.1 Klasifikasi Data 7
1.5.2 3.2 Enkripsi Data 7
1.5.3 3.3 Retensi & Penghapusan Data 7
1.6 ff 4. KEAMANAN JARINGAN 8
1.6.1 4.1 Konfigurasi Firewall 8
1.6.2 4.2 Perlindungan DDoS 8
1.6.3 4.3 Keamanan API 9
1.7 ff 5. MONITORING & AUDIT 9
1.7.1 5.1 Pemantauan Keamanan 9
1.7.2 5.2 Pencatatan Audit 9
1.7.3 5.3 Metrik Keamanan 10
1.8 ff 6. TANGGAP DARURAT 10
1.8.1 6.1 Klasifikasi Insiden 10
1.8.2 6.2 Rencana Tanggap Insiden 10
1.8.3 6.3 Kontak Darurat 10
1.9 ff 7. CADANGAN & PEMULIHAN BENCANA 11
1.9.1 7.1 Strategi Cadangan 11
1.9.2 7.2 Pemulihan Bencana 11
1.10 ff 8. KEPATUHAN & KEBIJAKAN 12
1.10.1 8.1 Kebijakan Privasi 12
1.10.2 8.2 Kebijakan Penggunaan yang Dapat Diterima 12
1.10.3 8.3 Keamanan Pihak Ketiga 12
1.11 ff 9. PENGUJIAN KEAMANAN 12
1.11.1 9.1 Pengujian Penetrasi 12
1.11.2 9.2 Tinjauan Kode 13
1.12 ff 10. PELATIHAN KEAMANAN 13
1.12.1 10.1 Kesadaran Keamanan Pengguna 13
1.12.2 10.2 Pelatihan Keamanan Pengembang 13
1.13 ff KONTAK KEAMANAN 13

1 SECURITY DOCUMENTATION

1.1 Document Information

Application: Koperasi Karyawan SKF
Version: 2.0.0
Security Level: Confidential
Last Updated: 17 January 2026
Next Review: 17 July 2026 (6 months)

1.2 Security Overview

Aplikasi Koperasi Karyawan SKF menerapkan **multi-layer security approach** untuk melindungi: - Data keuangan anggota (simpanan, pinjaman, SHU) - Data pribadi (NIK, alamat, foto, kontak) - Transaksi bisnis (POS, pembelian, expense) - Dokumen resmi koperasi

Security Standards Compliance: - OWASP Top 10 Protection - PCI-DSS Level 2 (Payment handling) - ISO 27001 Guidelines - UU No. 27 Tahun 2022 (Perlindungan Data Pribadi - Indonesia)

1.3 1. AUTHENTICATION & AUTHORIZATION

1.3.1 1.1 Authentication Mechanism

Primary Method: Session-based authentication (Laravel Sanctum)

Login Process:

```
User credentials (email + password)
    ↓
Validation
    ↓
Bcrypt hash comparison
    ↓
Session creation (120 min lifetime)
    ↓
CSRF token generation
    ↓
Access granted
```

Security Features: - Password hashing: **Bcrypt** (cost factor: 12) - Session timeout: **120 minutes** (configurable) - CSRF protection: **Enabled** on all forms - Rate limiting: **60 requests/minute** per IP - Login throttling: **5 failed attempts** = account locked (15 mins)

---

1 DOKUMENTASI KEAMANAN

1.1 Informasi Dokumen

Aplikasi: Koperasi Karyawan
SKFVersi: 2.0.0
Tingkat Keamanan: Rahasia
Terakhir Diperbarui: 17 Januari 2026Tinjauan
Berikutnya: 17 Juli 2026 (6 bulan)

1.2 Ikhtisar Keamanan

Aplikasi Koperasi Karyawan SKF menerapkan pendekatan keamanan multi-layer untuk melindungi:- Data keuangan anggota (simpanan, pinjaman, SHU) - Data pribadi (NIK, alamat, foto, kontak) -Transaksi bisnis (POS, pembelian, pengeluaran) - Dokumen resmi koperasi

Kepatuhan Standar Keamanan: - ff Perlindungan OWASP Top 10 - ff PCI-DSS Level 2 (Penanganan pembayaran) - ff Pedoman ISO 27001 - ff UU No. 27 Tahun 2022 (Perlindungan Data Pribadi -Indonesia)

1.3 1. AUTENTIKASI & OTORISASI

1.3.1 1.1 Mekanisme Autentikasi

Metode Utama: Autentikasi berbasis sesi (Laravel Sanctum)

Proses Login:

Kredensial pengguna (email + kata sandi)
↓
Validasi
↓
Perbandingan hash
Bcrypt↓
Pembuatan sesi (masa hidup 120 menit)
↓
Generasi token CSRF↓

Akses diberikanFitur Keamanan: - ff Hashing kata sandi: Bcrypt (faktor biaya: 12) - ff Waktu habis sesi: 120

menit (dapat dikonfigurasi) - ff Perlindungan CSRF: Diaktifkan di semua formulir - ff Pembatasan laju: 60 permintaan/menit per IP - ff Pembatasan login: 5 upaya gagal = akun terkunci (15 menit)

1.3.2 1.2 Password Policy

Requirements (Enforced): - Minimum length: **8 characters** - Must contain: - At least 1 uppercase letter - At least 1 lowercase letter - At least 1 number - Recommended: Include special characters (!@#$%^&*)

Password Reset: - Reset link valid for: **60 minutes** - Sent via: Encrypted email (TLS) - Token: Single-use, one-time only

Password Storage:

```
// NEVER stored in plain text
// Hashed using Bcrypt with salt
password_hash($password, PASSWORD_BCRYPT, ['cost' => 12]);
```

Forbidden Practices: - Default/weak passwords (e.g., "password", "123456") - Password sharing between users - Storing passwords in browser without encryption - Sending passwords via unencrypted channels

---

1.3.3 1.3 Role-Based Access Control (RBAC)

Roles Hierarchy:

```
System Admin (Highest privilege)
      Admin
         Pengurus
            Manager Toko
               Kasir
                  Anggota (Lowest privilege)
```

Access Control Matrix:

Feature	Admin	Pengurus	Manager	Kasir	Anggota
Dashboard	Full	Full	Limited	Limited	View
Manage Members					
Approve Loans					
View All Loans					Own only
POS Operations					
Generate Reports			Limited		
SHU Calculation					
Settings/Config					
Backup/Restore					

1.3.2 1.2 Kebijakan Kata Sandi Persyaratan (Diterapkan): - Panjang minimum: 8 karakter -

Harus mengandung: - Setidaknya 1
huruf besar - Setidaknya 1 huruf kecil - Setidaknya 1 angka - Disarankan: Sertakan karakter khusus (!@#$%^&*)

Reset Kata Sandi: - Tautan reset berlaku selama: 60 menit - Dikirim melalui: Email terenkripsi (TLS) - Token: Sekali pakai, hanya sekali

Penyimpanan Kata Sandi:

```
// TIDAK PERNAH disimpan dalam teks biasa//
Di-hash menggunakan Bcrypt dengan salt

password_hash($password, PASSWORD_BCRYPT, ['cost' => 12]);
```

Praktik Terlarang: - ff Kata sandi default/lemah (misalnya, “password”, “123456”) - ff Berbagi kata sandi antara pengguna - ff Menyimpan kata sandi di browser tanpa enkripsi - ff Mengirim kata sandi melalui saluran yang tidak terenkripsi

---

1.3.3 1.3 Kontrol Akses Berbasis Peran (RBAC)

Hierarki Peran:

Admin Sistem (Hak istimewa
tertinggi) Admin
Pengurus
Manager Toko
Kasir
Anggota (Hak istimewa terendah)

Matriks Kontrol Akses:

Fitur	Admin	Pengurus	Manajer	Kasir	Anggota
Dasbor	Penuh	Penuh	Terbatas	Terbatas	Lihat
Kelola Anggota					
Setujui Pinjaman					
Lihat Semua Pinjaman					Hanya milik sendiri
POS Operasi					
Hasilkan Laporan			Terbatas		
SHUCalculation					
Settings/Config					
ffBackup/Restore ff					

Permission Enforcement: - **Middleware Level:** Route protection - **Controller Level:** Authorization checks - **View Level:** Conditional rendering - **Database Level:** Query scoping

Example Implementation:

```
// Middleware
Route::middleware(['auth', 'role:admin,pengurus'])->group(function() {
    Route::resource('members', MemberController::class);
});

// Controller
public function approve(Loan $loan) {
    $this->authorize('approve', $loan);
    // ...
}

// Policy
public function approve(User $user, Loan $loan) {
    return $user->role === 'admin' || $user->role === 'pengurus';
}
```

1.4 2. APPLICATION SECURITY

1.4.1 2.1 OWASP Top 10 Protection

1.4.1.1 A01: Broken Access Control PROTECTED

- All routes protected by authentication middleware
- Authorization checks on every sensitive operation
- No direct object reference without validation

1.4.1.2 A02: Cryptographic Failures PROTECTED

- HTTPS/TLS 1.3 enforced (production)
- Sensitive data encrypted at rest
- Password hashed with bcrypt
- Database connection encrypted

1.4.1.3 A03: Injection PROTECTED

- **SQL Injection:** Eloquent ORM (parameterized queries)
- **XSS:** Blade auto-escaping `{{ $var }}`
- **Command Injection:** No shell_exec/system calls with user input

Example:

```
//   SAFE - Eloquent ORM
User::where('email', $request->email)->first();
```

Penegakan Izin: - Tingkat Middleware: Perlindungan rute - Tingkat Controller: Pemeriksaan otorisasi - Tingkat Tampilan: Rendering bersyarat - Tingkat Basis Data: Penentuan kueriContoh Implementasi:

```
// Middleware
Route::middleware(['auth', 'role:admin,pengurus'])->group(function() {
    Route::resource('members', MemberController::class);
});

// Controller
public function approve(Loan $loan) {$this->
authorize('approve', $loan);
    // ...
}

// Kebijakan
public function approve(User $user, Loan $loan) {
    return $user->role === 'admin' || $user->role === 'pengurus';
}
```

1.4 2. KEAMANAN APLIKASI

1.4.1 2.1 Perlindungan OWASP Top 10

1.4.1.1 A01: Kontrol Akses yang Rusak fi PROTECTED

- Semua rute dilindungi oleh middleware otentikasi
- Pemeriksaan otorisasi pada setiap operasi sensitif
- Tidak ada referensi objek langsung tanpa validasi

1.4.1.2 A02: Kegagalan Kriptografi fi PROTECTED

- HTTPS/TLS 1.3 diterapkan (produksi)
- Data sensitif dienkripsi saat tidak aktif
- Kata sandi di-hash dengan bcrypt
- Koneksi database dienkripsi

1.4.1.3 A03: Penyisipan fi PROTECTED

- Penyisipan SQL: Eloquent ORM (kueri terparameter)
- XSS: Blade auto-escaping {{ $var }}
- **Penyisipan Perintah: Tidak ada shell_exec/panggilan sistem**

dengan input penggunaContoh:

```
//  AMAN - Eloquent ORM
User::where('email', $request->email)->first();
```

```
//  UNSAFE - Raw SQL (avoided)
DB::select("SELECT * FROM users WHERE email = '$email'");
```

1.4.1.4 A04: Insecure Design MITIGATED

- Security requirements defined upfront
- Threat modeling conducted
- Security reviews in design phase

1.4.1.5 A05: Security Misconfiguration PROTECTED

- `APP_DEBUG=false` in production
- Error messages sanitized (no stack traces to users)
- Unnecessary services disabled
- Default credentials changed

1.4.1.6 A06: Vulnerable Components MONITORED

- Dependencies updated regularly (`composer update`)
- Security advisories monitored
- Laravel framework kept up-to-date

Check for vulnerabilities:

```
composer audit
npm audit
```

1.4.1.7 A07: Identification & Authentication Failures PROTECTED

- Strong password policy enforced
- Multi-factor authentication (planned)
- Session management secure
- Credential stuffing prevention (rate limiting)

1.4.1.8 A08: Software & Data Integrity Failures PROTECTED

- Code signing (Git commits)
- Dependency integrity (composer.lock)
- Auto-update disabled (manual review required)

1.4.1.9 A09: Security Logging & Monitoring IMPLEMENTED

- All authentication events logged
- Failed login attempts tracked
- Audit trail for critical operations
- Log retention: 90 days

1.4.1.10 A10: Server-Side Request Forgery (SSRF) PROTECTED

- URL validation before external requests
- Whitelist of allowed domains

```
// TIDAK AMAN – SQL Mentah (dihindari)
DB::select("SELECT ∗ FROM users WHERE email = '$email'");
```

1.4.1.4 A04: Desain Tidak Aman fi DIMITIGASI

- Persyaratan keamanan ditentukan di awal
- Pemodelan ancaman dilakukan
- Tinjauan keamanan pada fase desain

1.4.1.5 A05: Kesalahan Konfigurasi Keamanan fi TERLINDUNGI

- APP_DEBUG=false di produksi
- Pesan kesalahan disanitasi (tidak ada jejak tumpukan untuk pengguna)
- Layanan yang tidak perlu dinonaktifkan
- Kredensial default diubah

1.4.1.6 A06: Komponen Rentan fi DIMONITOR

- Ketergantungan diperbarui secara berkala (composer update)
- Pemberitahuan keamanan dimonitor
- Kerangka kerja Laravel diperbarui

Periksa kerentanan:

```
composer
auditnpm audit
```

1.4.1.7 A07: Kegagalan Identifikasi & Autentikasi fi TERLINDUNGI

- Kebijakan kata sandi yang kuat diterapkan
- Autentikasi multi-faktor (direncanakan)
- Manajemen sesi yang aman
- Pencegahan pengisian kredensial (pembatasan laju)

1.4.1.8 A08: Kegagalan Integritas Perangkat Lunak & Data fi PROTECTED

- Penandatanganan kode (komit Git)
- Integritas ketergantungan (composer.lock)
- Pembaruan otomatis dinonaktifkan (tinjauan manual diperlukan)

1.4.1.9 A09: Pencatatan & Pemantauan Keamanan fi IMPLEMENTED

- Semua peristiwa otentikasi dicatat
- Upaya login yang gagal dilacak
- Jejak audit untuk operasi kritis
- Retensi log: 90 hari

1.4.1.10 A10: Pemalsuan Permintaan Sisi Server (SSRF) fi PROTECTED

- Validasi URL sebelum permintaan eksternal
- Daftar putih domain yang diizinkan

- No user-controlled URLs in APIs

1.4.2 2.2 CSRF Protection

Enabled Globally: - All POST/PUT/PATCH/DELETE requests require CSRF token - Token included in all forms via `@csrf` directive - Token validated by `VerifyCsrfToken` middleware

Example:

```
<form method="POST" action="/loans">
    @csrf
    <!-- CSRF token auto-included -->
    <input type="text" name="amount">
    <button type="submit">Submit</button>
</form>
```

AJAX Requests:

```
axios.defaults.headers.common['X-CSRF-TOKEN'] =
    document.querySelector('meta[name="csrf-token"]').content;
```

1.4.3 2.3 XSS Prevention

Blade Auto-Escaping:

```
{{--   SAFE - Auto-escaped --}}
{{ $user->name }}

{{--   DANGEROUS - Raw output (only use if necessary) --}}
{!! $htmlContent !!}
```

Content Security Policy (CSP):

```
Content-Security-Policy:
    default-src 'self';
    script-src 'self' 'unsafe-inline' https://app.midtrans.com;
    img-src 'self' data: https:;
```

1.4.4 2.4 Input Validation

Server-Side Validation (Mandatory):

```
$request->validate([
    'email' => 'required|email|unique:users',
    'amount' => 'required|numeric|min:0|max:100000000',
    'nik' => 'required|digits:16',
    'phone' => 'required|regex:/^62[0-9]{9,12}$/',
]);
```

- Tidak ada URL yang dikendalikan pengguna dalam API

1.4.2 2.2 Perlindungan CSRF

Diaktifkan Secara Global: - Semua permintaan POST/PUT/PATCH/DELETE memerlukan token CSRF - Token disertakan dalam semua formulir melalui direktif @csrf - Token divalidasi oleh middleware VerifyCsrfToken

Contoh:

```
<form method="POST" action="/loans">
    @csrf
    <!-- Token CSRF otomatis disertakan -->
    <input type="text" name="amount">
    <button type="submit">Kirim</button>
</form>
```

Permintaan AJAX:

```
axios.defaults.headers.common['X-CSRF-TOKEN'] =
    document.querySelector('meta[name="csrf-token"]').content;
```

1.4.3 2.3 Pencegahan XSS

Blade Auto-Escaping:

```
{{-- AMAN - Auto-escaped --}}
{{ $user->name }}

{{-- BAHAYA - Output mentah (hanya gunakan jika perlu) --}}
{!! $htmlContent !!}
```

Kebijakan Keamanan Konten (CSP):

```
Content-Security-Policy:
    default-src 'self';
    script-src 'self' 'unsafe-inline' https://app.midtrans.com;img-src
    'self' data: https:;
```

1.4.4 2.4 Validasi Input

Validasi Sisi Server (Wajib):

```
$request->validate([
    'email' => 'required|email|unique:users',
    'amount' => 'required|numeric|min:0|max:100000000',
    'nik' => 'required|digits:16',
    'phone' => 'required|regex:/^62[0-9]{9,12}$/',
]);
```