Contents

Security Policy

Supported Versions

Version	Supported
4.17.x	:white_check_mark:
< 4.0	:x:

Reporting a Vulnerability

We take security vulnerabilities seriously. If you discover a security issue, please report it responsibly.

How to Report

DO NOT open a public GitHub issue for security vulnerabilities.

Instead, please email: **security@radiant.example.com**

Include: - Description of the vulnerability - Steps to reproduce - Potential impact - Suggested fix (if any)

What to Expect

1. **Acknowledgment**: Within 24 hours
2. **Initial Assessment**: Within 72 hours
3. **Status Updates**: Every 7 days
4. **Resolution**: Varies by severity

Severity Levels

Level	Response Time	Examples
Critical	24 hours	RCE, authentication bypass
High	72 hours	SQL injection, privilege escalation
Medium	1 week	XSS, CSRF
Low	2 weeks	Information disclosure

Security Best Practices

For Contributors

1. **Never commit secrets**

    - Use environment variables
    - Use AWS Secrets Manager
    - Check for secrets in pre-commit hooks

2. **Always use parameterized queries**

```
// Good
executeStatement('SELECT * FROM users WHERE id = :id', [
  { name: 'id', value: { stringValue: userId } }
]);

// Bad
executeStatement(`SELECT * FROM users WHERE id = '${userId}'`);
```

3. **Validate all input**

    - Use Zod or similar for validation
    - Sanitize user input
    - Validate on both client and server

4. **Use RLS for tenant isolation**

```
-- Always set tenant context
SET app.current_tenant_id = 'tenant-id';
```

5. **Follow least privilege**

    - Minimal IAM permissions
    - Role-based access control
    - Regular permission audits

For Operators

1. **Enable MFA** for all admin accounts
2. **Rotate credentials** regularly
3. **Monitor audit logs** for suspicious activity
4. **Keep dependencies updated**
5. **Use WAF** in production

6. **Enable encryption** at rest and in transit

Security Features

Authentication

- AWS Cognito with MFA support
- JWT tokens with short expiration
- Secure session management

Authorization

- Role-based access control (RBAC)
- Tenant-level isolation (RLS)
- API key scoping

Data Protection

- TLS 1.3 for transit encryption
- AES-256 for data at rest
- Field-level encryption for PII

Monitoring

- CloudWatch for logging
- GuardDuty for threat detection
- Config for compliance

Compliance

- HIPAA-ready (Tier 3+)
- SOC 2 controls
- GDPR data handling

Disclosure Policy

We follow responsible disclosure:

1. Reporter contacts us privately
2. We verify and assess impact
3. We develop and test fix
4. We release patch
5. We credit reporter (if desired)
6. We publish advisory

Bug Bounty

We currently do not have a formal bug bounty program. However, we appreciate all security reports and will acknowledge contributors in our security advisories.

Security Updates

Subscribe to security updates: - Watch this repository - Follow our security advisories