Contributor Onboarding Guide

Welcome to the Audityzer community! This guide will help you get started as a contributor to our Web3 security testing platform.

Welcome to Audityzer

Audityzer is an open-source platform dedicated to making Web3 applications more secure through advanced testing, AI-powered vulnerability detection, and community-driven security research.

Our Mission

To democratize Web3 security by providing accessible, comprehensive, and cutting-edge security testing tools for developers, auditors, and security researchers worldwide.

Our Values

- **Security First**: Every decision prioritizes security and user safety
- **Community Driven**: We believe in the power of collective intelligence
- **Innovation**: We embrace cutting-edge technologies and methodologies
- **Transparency**: Open development, clear communication, honest feedback
- **Inclusivity**: Everyone is welcome, regardless of background or experience level

Getting Started

1. Join Our Community

Discord Server

Join our Discord community at https://discord.gg/audityzer (https://discord.gg/audityzer)

Key Channels:
- `#welcome` - Introduce yourself
- `#general` - General discussions
- `#development` - Development discussions
- `#security-research` - Security research and findings
- `#help` - Get help from the community
- `#announcements` - Important updates

Other Platforms

- **Reddit**: r/audityzer (https://reddit.com/r/audityzer)
- **Twitter**: @audityzer (https://twitter.com/audityzer)
- **GitHub**: Audityzer/audityzer (https://github.com/Audityzer/audityzer)

2. Understand the Codebase

Repository Structure

```
audityzer/
├── src/
│   ├── core/           # Core security testing modules
│   ├── ai/             # AI-powered analysis
│   ├── web/            # Web dashboard
│   ├── cli/            # Command-line interface
│   └── utils/          # Utility functions
├── tests/
│   ├── e2e/            # End-to-end tests
│   ├── integration/    # Integration tests
│   └── security/       # Security test suite
├── docs/               # Documentation
├── community/          # Community tools and resources
└── examples/           # Usage examples
```

Key Technologies

- **Backend**: Node.js, TypeScript, Express
- **Frontend**: React, Material-UI, D3.js
- **AI/ML**: OpenAI API, TensorFlow.js, Python
- **Blockchain**: Web3.js, Ethers.js, Hardhat
- **Testing**: Jest, Playwright, Mocha
- **Infrastructure**: Docker, Kubernetes, Prometheus

3. Development Environment Setup

Prerequisites

```
# Required software
- Node.js >= 16.0.0
- npm or yarn
- Git
- Docker (optional)
- Python 3.8+ (for AI components)
```

Quick Setup

```
# 1. Fork the repository on GitHub
# 2. Clone your fork
git clone https://github.com/your-username/audityzer.git
cd audityzer

# 3. Add upstream remote
git remote add upstream https://github.com/Audityzer/audityzer.git

# 4. Install dependencies
npm install

# 5. Copy environment file
cp .env.example .env

# 6. Run setup wizard
npm run setup

# 7. Start development server
npm run dev
```

Environment Configuration

```
# .env file configuration
NODE_ENV=development
PORT=3000

# Blockchain networks (get free keys from Infura/Alchemy)
ETHEREUM_RPC_URL=https://mainnet.infura.io/v3/your-key
POLYGON_RPC_URL=https://polygon-mainnet.infura.io/v3/your-key

# AI configuration (optional for basic development)
OPENAI_API_KEY=your-openai-key

# Database (SQLite for development)
DATABASE_URL=sqlite:./dev.db
```

Contribution Areas

1. Security Research

Perfect for: Security researchers, auditors, penetration testers

What you can do:
- Discover new vulnerability patterns
- Improve detection algorithms
- Add support for new protocols
- Create test cases for edge cases

Getting started:

```
# Explore security modules
cd src/core/security/
ls -la

# Run security tests
npm run test:security

# Add new vulnerability detection
cp template-detector.js new-vulnerability-detector.js
```

2. AI/ML Development

Perfect for: Data scientists, ML engineers, AI researchers

What you can do:
- Improve vulnerability detection models
- Enhance pattern recognition
- Optimize model performance
- Create new AI-powered features

Getting started:

```
# Explore AI modules
cd src/core/ai/
ls -la

# Run AI tests
npm run test:ai

# Train models locally
npm run ai:train
```

3. Frontend Development

Perfect for: Frontend developers, UI/UX designers

What you can do:
- Improve dashboard visualizations
- Create new UI components
- Enhance user experience
- Add accessibility features

Getting started:

```
# Start frontend development
npm run dev:frontend

# Explore dashboard components
cd src/web/components/
ls -la

# Run frontend tests
npm run test:frontend
```

4. Backend Development

Perfect for: Backend developers, DevOps engineers

What you can do:
- Improve API performance
- Add new endpoints
- Enhance security features
- Optimize database queries

Getting started:

```
# Start backend development
npm run dev:backend

# Explore API routes
cd src/routes/
ls -la

# Run backend tests
npm run test:backend
```

5. Documentation

Perfect for: Technical writers, developers who love documentation

What you can do:
- Improve existing documentation
- Create tutorials and guides
- Write API documentation
- Translate documentation

Getting started:

```
# Explore documentation
cd docs/
ls -la

# Build documentation locally
npm run docs:dev

# Contribute to docs
# Edit markdown files and submit PR
```

6. Community Building

Perfect for: Community managers, evangelists, educators

What you can do:
- Help newcomers in Discord
- Create educational content
- Organize community events
- Moderate discussions

Getting started:
- Join Discord and introduce yourself
- Help answer questions in #help channel
- Share your Audityzer experience on social media
- Propose community initiatives

Your First Contribution

Good First Issues

Look for issues labeled `good first issue` on GitHub:
- Documentation improvements
- Simple bug fixes
- Test case additions
- Code cleanup tasks

Step-by-Step Guide

1. Find an Issue

```
# Browse issues on GitHub
https://github.com/Audityzer/audityzer/issues?q=is%3Aissue+is%3Aopen+label%3A%22good+first+issue%22

# Or create a new feature
# Discuss in Discord first for larger changes
```

2. Create a Branch

```
# Update your fork
git checkout develop
git pull upstream develop

# Create feature branch
git checkout -b feature/your-feature-name

# Make your changes
# ... edit files ...

# Test your changes
npm test
npm run lint
```

3. Submit a Pull Request

```
# Commit your changes
git add .
git commit -m "feat: add new vulnerability detection for XYZ"

# Push to your fork
git push origin feature/your-feature-name

# Create PR on GitHub
# Fill out the PR template
# Request review from maintainers
```

PR Template

```
## Description
Brief description of changes

## Type of Change
- [ ] Bug fix
- [ ] New feature
- [ ] Breaking change
- [ ] Documentation update

## Testing
- [ ] Unit tests added/updated
- [ ] Integration tests added/updated
- [ ] Manual testing completed

## Checklist
- [ ] Code follows style guidelines
- [ ] Self-review completed
- [ ] Documentation updated
- [ ] No breaking changes (or documented)
```

Contributor Levels

Community Contributor

Requirements: First merged PR
Benefits:
- Contributor badge on Discord
- Listed in contributors file
- Access to contributor channels

Regular Contributor

Requirements: 5+ merged PRs over 3 months
Benefits:
- Priority code review
- Input on roadmap decisions
- Early access to new features

Core Contributor

Requirements: 20+ merged PRs, consistent contributions
Benefits:
- Write access to repository
- Mentorship opportunities
- Speaking opportunities at events

Maintainer

Requirements: Invitation by existing maintainers
Benefits:
- Full repository access
- Release management
- Community leadership role

Learning Resources

Web3 Security Fundamentals

- Smart Contract Security Best Practices (https://consensys.github.io/smart-contract-best-practices/)
- DeFi Security Summit (https://defisecuritysummit.org/)
- Ethereum Security Resources (https://ethereum.org/en/developers/docs/security/)

Auditzer-Specific Learning

```
# Explore examples
cd examples/
ls -la

# Run tutorials
npm run tutorial:basic
npm run tutorial:advanced

# Read architecture docs
open docs/architecture.md
```

Recommended Courses

- **Blockchain Security**: ConsenSys Academy (https://consensys.net/academy/)
- **Smart Contract Auditing**: OpenZeppelin Defender (https://defender.openzeppelin.com/)
- **DeFi Security**: DeFi Security Course (https://defisecurity.io/)

Community Guidelines

Code of Conduct

We follow the Contributor Covenant (CODE_OF_CONDUCT.md):
- Be respectful and inclusive
- Welcome newcomers
- Provide constructive feedback
- Focus on what's best for the community

Communication Guidelines

- **Be patient**: Everyone is learning
- **Be helpful**: Share knowledge freely
- **Be constructive**: Provide actionable feedback
- **Be respectful**: Treat everyone with kindness

Recognition and Rewards

Contribution Recognition

- Monthly contributor highlights
- Annual contributor awards
- Conference speaking opportunities
- Swag and merchandise

Hall of Fame

Outstanding contributors are featured in our Hall of Fame (../tests/security/HALL_OF_FAME.md).

Bounty Program

We offer bounties for:
- Critical security vulnerabilities
- Major feature implementations
- Comprehensive documentation
- Community initiatives

Development Workflow

Branch Strategy

```
main (production)
├── unified-main (stable + latest)
├── roadmap-exec (cutting-edge)
└── develop (integration)
    └── feature/your-feature
```

Commit Convention

```
# Format: type(scope): description
feat(security): add reentrancy detection
fix(dashboard): resolve loading issue
docs(api): update endpoint documentation
test(bridge): add cross-chain tests
```

Code Style

```
# Linting
npm run lint
npm run lint:fix

# Formatting
npm run format

# Type checking
npm run type-check
```

Advanced Development

Running Tests

```
# All tests
npm test

# Specific test suites
npm run test:unit
npm run test:integration
npm run test:e2e
npm run test:security

# With coverage
npm run test:coverage

# Watch mode
npm run test:watch
```

Debugging

```
# Debug mode
npm run dev:debug

# VS Code debugging
# Use provided launch.json configuration

# Browser debugging
npm run dev:inspect
```

Performance Profiling

```
# Profile application
npm run profile

# Memory analysis
npm run memory:analyze

# Performance benchmarks
npm run benchmark
```

Internationalization

Adding Translations

```
# Add new language
npm run i18n:add-language es

# Update translations
npm run i18n:extract
npm run i18n:update

# Validate translations
npm run i18n:validate
```

Translation Guidelines

- Use clear, concise language
- Maintain technical accuracy
- Consider cultural context
- Test UI with longer text

Metrics and Analytics

Contribution Metrics

We track:
- Code contributions (commits, PRs)
- Community engagement (Discord, forums)
- Documentation improvements
- Bug reports and fixes
- Security research contributions

Personal Dashboard

Contributors get access to:
- Contribution statistics
- Impact metrics
- Recognition badges
- Learning progress

Events and Initiatives

Regular Events

- **Weekly Office Hours**: Every Wednesday 3 PM UTC
- **Monthly Community Call**: First Friday of each month
- **Quarterly Hackathons**: Themed security challenges
- **Annual Conference**: AudityzerCon (virtual/hybrid)

Special Initiatives

- **Security Research Grants**: Funding for research projects
- **Student Program**: Mentorship for students

- **Open Source Fridays**: Dedicated contribution time
- **Bug Bounty Events**: Focused security testing

Getting Help

Where to Ask Questions

1. **Discord #help**: Quick questions and real-time help
2. **GitHub Discussions**: Detailed technical discussions
3. **Stack Overflow**: Tag questions with `adityzer`
4. **Office Hours**: Weekly live Q&A sessions

Mentorship Program

New contributors can request mentorship:
- Pair programming sessions
- Code review guidance
- Career development advice
- Technical skill building

Common Issues and Solutions

Setup Problems

```
# Clear npm cache
npm cache clean --force

# Reinstall dependencies
rm -rf node_modules package-lock.json
npm install

# Check Node.js version
node --version  # Should be >= 16.0.0
```

Test Failures

```
# Update test snapshots
npm run test:update-snapshots

# Run tests in isolation
npm test -- --runInBand

# Debug specific test
npm test -- --testNamePattern="your test name"
```

Career Development

Skills You'll Develop

- **Technical**: Blockchain, security, AI/ML, full-stack development
- **Soft Skills**: Communication, collaboration, project management
- **Domain Expertise**: Web3 security, DeFi protocols, smart contracts

Career Opportunities

Contributing to Adityzer can lead to:
- Security auditor positions
- Blockchain developer roles
- DevSecOps opportunities
- Technical writing careers
- Community management roles

Certification and Recognition

- Adityzer Contributor Certificates
- LinkedIn skill endorsements
- Conference speaking opportunities
- Industry recognition

Next Steps

Immediate Actions

1. **Join Discord**: Introduce yourself in #welcome
2. **Set up development environment**: Follow the setup guide
3. **Find your first issue**: Look for `good first issue` labels
4. **Make your first contribution**: Submit a small PR
5. **Engage with community**: Participate in discussions

Medium-term Goals

1. **Become a regular contributor**: Make consistent contributions
2. **Specialize in an area**: Choose security, AI, frontend, etc.
3. **Mentor newcomers**: Help others get started
4. **Lead initiatives**: Propose and lead new projects

Long-term Vision

1. **Become a core contributor**: Take on significant responsibilities
2. **Shape the roadmap**: Influence project direction
3. **Build expertise**: Become a recognized expert in Web3 security
4. **Give back**: Mentor others and contribute to the ecosystem

Contact Information

Maintainers

- **Lead Maintainer**: @leadmaintainer (Discord)
- **Security Lead**: @securitylead (Discord)
- **Community Manager**: @communitymanager (Discord)

Support Channels

- **Email**: contributors@adityzer.com
- **Discord**: https://discord.gg/adityzer (https://discord.gg/adityzer)
- **GitHub**: Adityzer/adityzer (https://github.com/Adityzer/adityzer)

Welcome to the Audityzer family! We're excited to have you on this journey to make Web3 more secure.

This guide is a living document. Help us improve it by suggesting changes or additions!