

Contact Form Guide

Ape Framework - Contact Form Guide

Introduction

The Contact Form is a public-facing feature that allows website visitors to send messages to site administrators. This guide covers the anti-spam protection measures implemented and how to configure message recipients.

Contact Form Overview

The contact form is accessible to all visitors (authenticated or not) and provides a simple way for users to reach out with questions, feedback, or inquiries.

Form Fields

Field	Purpose
Name	Sender's name
Email	Sender's email address (for replies)
Subject	Message topic
Message	The actual message content

Location

- **URL:** /Info/Contact
 - **Menu:** Available under "Contact" in the navigation bar
-

Anti-Spam Protection

Spam submissions are a significant problem for public contact forms. Bots constantly crawl the web looking for forms to abuse, sending unwanted advertisements, phishing attempts, and malicious content.

The Ape Framework implements a **triple-layer protection system** to combat spam while maintaining a smooth user experience for legitimate visitors.

Layer 1: Honeypot Fields

What are honeypot fields?

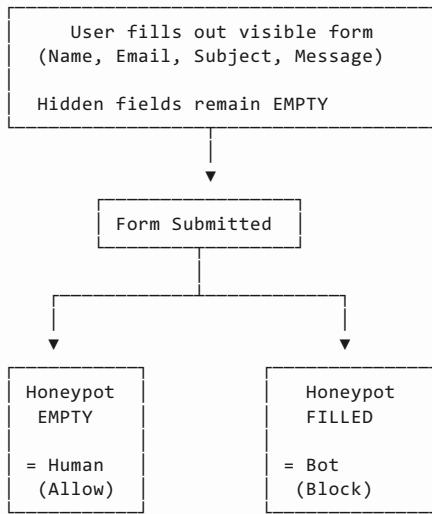
Honeypot fields are hidden form inputs that are invisible to human users but visible to automated bots. When a bot fills out a form, it typically populates every field it finds. By including hidden fields that humans never see, we can detect bot submissions.

Implementation:

The contact form includes four honeypot fields: - Comment - Website - Company - Url

These fields are hidden using CSS (`display: none` or positioned off-screen). Legitimate users never see or interact with them.

How it works:



Why it's effective:

- Invisible to humans - no CAPTCHA friction
- Bots can't distinguish real fields from honeypots
- Zero impact on user experience
- No external dependencies

Layer 2: JavaScript Token

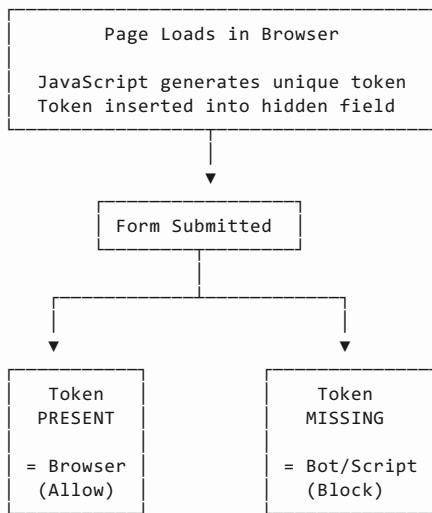
What is the JavaScript token?

A JavaScript-generated token that proves the form was rendered in a real browser with JavaScript enabled. Automated bots often don't execute JavaScript, making this an effective filter.

Implementation:

When the page loads, JavaScript generates a unique token and inserts it into a hidden form field. The server validates this token on submission.

How it works:



Why it's effective:

- Many spam bots don't execute JavaScript
- Simple bots that scrape HTML miss the token
- Headless browsers can be detected
- Adds another validation layer

Layer 3: Timing Validation

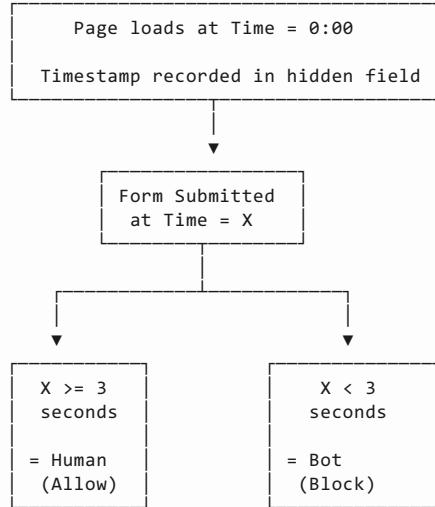
What is timing validation?

A check that ensures a minimum amount of time has passed between when the form was loaded and when it was submitted. Humans need time to read and fill out forms; bots submit instantly.

Implementation:

When the page loads, a timestamp is recorded. On submission, the server checks if at least 3 seconds have elapsed. Instant submissions are rejected.

How it works:

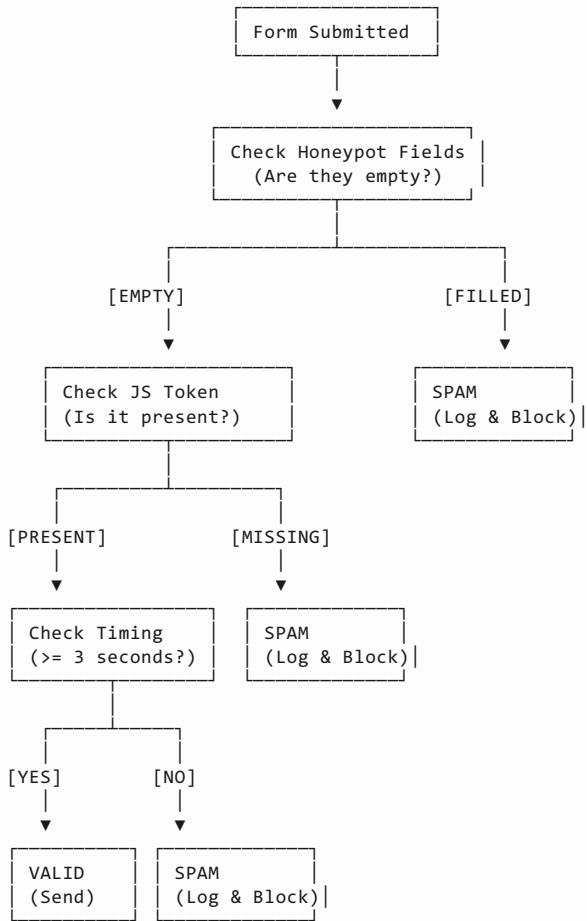


Why it's effective:

- Humans physically cannot fill forms in under 3 seconds
- Automated submissions happen in milliseconds
- No user interaction required
- Simple but highly effective

Combined Protection Flow

When a form is submitted, all three layers are checked:



Spam Logging

All detected spam attempts are logged for monitoring:

- **IP Address** - Source of the attempt
- **Timestamp** - When it occurred
- **Failure Reason** - Which protection layer caught it

This data helps identify attack patterns and persistent offenders.

Configuring Contact Form Recipients

One of the key administrative features is the ability to control who receives contact form messages without modifying code.

Why Configurable Recipients?

Different organizations have different needs:

- **Small teams** - All messages go to one person
- **Departments** - Route to support, sales, or admin teams
- **Multiple stakeholders** - Several people need visibility
- **Staff changes** - Update recipients without code deployments

Setting Up Recipients

Step 1: Access Contact Form Settings

1. Log in as an Administrator
2. Navigate to **Manage** → **Email Configuration** → **Contact Form Recipients**

Or go directly to: /ContactFormSettings

Step 2: Enter Recipient Addresses

Enter email addresses separated by commas:

admin@example.com, support@example.com, manager@example.com

Step 3: Save Changes

Click **Save** to update the settings immediately.

How It Works

The contact form recipient list is stored in the `SystemSettings` table:

Setting Key	Value
ContactFormEmails	admin@example.com, support@example.com

When a contact form is submitted:

1. System retrieves the `ContactFormEmails` setting
2. Parses the comma-separated list
3. Sends the message to all listed recipients
4. Logs the email in the Email Log

Fallback Behavior

If no recipients are configured: - The system falls back to the `SMTP_USERNAME` credential - This ensures messages aren't lost during initial setup

Best Practices

1. **Use distribution lists** - If your email system supports it, use a distribution list (e.g., `contact@example.com`) that forwards to multiple people. This simplifies management.
2. **Include a backup** - Always have at least two recipients in case one person is unavailable.
3. **Monitor the Email Log** - Regularly check the Email Log to ensure messages are being delivered.
4. **Test after changes** - After updating recipients, submit a test message to verify delivery.

Email Delivery

Contact form messages are sent using the site's email system:

Email Flow

1. **User submits form** → Validation passes
2. **System formats message** → HTML email template
3. **Email sent via Azure** → Primary provider
4. **If Azure fails** → SMTP fallback
5. **Logged to database** → Email Log entry created

Email Template

Messages are formatted professionally:

Subject: Contact Form: [User's Subject]
From: [User's Name] <[User's Email]>
Message:
[User's Message]

Sent from the Contact Form at [Site Name]

Viewing Sent Messages

All contact form emails appear in the Email Log:

1. Navigate to **Manage** → **Email Configuration** → **Email Log**
 2. Filter by **Email Type: ContactForm**
 3. View delivery status, timestamps, and any errors
-

Troubleshooting

Messages Not Being Received

1. **Check recipient configuration** - Verify email addresses in Contact Form Settings
2. **Check Email Log** - Look for delivery errors
3. **Check spam folders** - Messages may be filtered
4. **Test email system** - Use Email Test page to verify setup

Spam Getting Through

If spam is bypassing protection:

1. **Review Email Log** - Check for patterns in successful spam
2. **Consider additional measures:**
 - Rate limiting by IP
 - CAPTCHA for suspicious IPs
 - Block known spam sources

Legitimate Messages Blocked

If real users report issues:

1. **Ensure JavaScript is required** - Some privacy browsers block JS
 2. **Check timing threshold** - 3 seconds should be sufficient
 3. **Review honeypot CSS** - Ensure fields are truly hidden
-

Security Considerations

Data Handling

- User input is sanitized before display
- HTML is encoded to prevent XSS attacks
- Email addresses are validated for format

Privacy

- Contact form data is not stored in the database (only in email)
- IP addresses are logged only for spam attempts
- No tracking cookies are used

Rate Limiting

Consider implementing rate limiting if abuse becomes an issue: - Limit submissions per IP per hour - Add CAPTCHA after multiple submissions - Temporary blocks for repeat offenders

Summary

The Ape Framework's contact form provides:

1. **Triple-layer spam protection**
 - Honeypot fields (catch bots filling hidden fields)
 - JavaScript token (require real browser)
 - Timing validation (require human-speed interaction)
 2. **Configurable recipients**
 - Change who receives messages through admin UI
 - No code changes required
 - Supports multiple recipients
 3. **Reliable delivery**
 - Azure Communication Services primary
 - SMTP fallback for reliability
 - Full logging in Email Log
 4. **Easy monitoring**
 - All messages logged
 - Spam attempts tracked
 - Delivery status visible
-

Version: 1.0.0 **Framework:** Ape Framework **Site:** <https://Illustrate.net>