

Bot Distribution Guide for APBeeper

This guide covers making your APBeeper Discord bot available to other Discord servers, including OAuth2 setup, verification process, and user onboarding.

Table of Contents

- [OAuth2 Invite Link Generation](#)
- [Discord Bot Verification](#)
- [Website Integration](#)
- [User Onboarding](#)
- [Bot Listing Platforms](#)
- [Analytics and Monitoring](#)
- [Support and Documentation](#)
- [Legal and Privacy Considerations](#)

OAuth2 Invite Link Generation

1. Understanding Discord Permissions

APBeeper requires specific permissions to function properly. Here's the breakdown:

Essential Permissions (Required)

- **Send Messages** (2048): Basic message sending
- **Read Message History** (65536): Access to message history
- **Use Slash Commands** (2147483648): Modern Discord interactions
- **Embed Links** (16384): Rich message embeds
- **Attach Files** (32768): File attachments for logs/exports

Moderation Permissions (Recommended)

- **Manage Messages** (8192): Delete/edit messages for moderation
- **Kick Members** (2): Remove members from server
- **Ban Members** (4): Ban members from server
- **Manage Roles** (268435456): Role management features
- **Manage Channels** (16): Channel management features
- **View Audit Log** (128): Access server audit logs

Administrative Permissions (Optional)

- **Administrator** (8): Full server control (not recommended for public distribution)
- **Manage Server** (32): Server settings management

2. Calculate Permissions Integer

Use the Discord Permissions Calculator or calculate manually:

Recommended Permission Set:

- Send Messages: 2048
- Read Message History: 65536

- Use Slash Commands: 2147483648
- Embed Links: 16384
- Attach Files: 32768
- Manage Messages: 8192
- Kick Members: 2
- Ban Members: 4
- Manage Roles: 268435456
- Manage Channels: 16
- View Audit Log: 128

Total: 2147616870

3. Generate Invite Links

Basic Invite Link

```
https://discord.com/api/oauth2/authorize?client_id=YOUR_CLIENT_ID&permissions=2147616870&scope=bot%20applications.commands
```

Advanced Invite Link with Guild Selection

```
https://discord.com/api/oauth2/authorize?client_id=YOUR_CLIENT_ID&permissions=2147616870&scope=bot%20applications.commands&guild_id= Guild_ID&disable_guild_select=true
```

Invite Link Generator Script

Create `scripts/generate_invite.js` :

```

const CLIENT_ID = process.env.CLIENT_ID || 'YOUR_CLIENT_ID';

// Permission calculator
const permissions = {
  SEND_MESSAGES: 2048,
  READ_MESSAGE_HISTORY: 65536,
  USE_SLASH_COMMANDS: 2147483648,
  EMBED_LINKS: 16384,
  ATTACH_FILES: 32768,
  MANAGE_MESSAGES: 8192,
  KICK_MEMBERS: 2,
  BAN_MEMBERS: 4,
  MANAGE_ROLES: 268435456,
  MANAGE_CHANNELS: 16,
  VIEW_AUDIT_LOG: 128
};

// Calculate total permissions
const totalPermissions = Object.values(permissions).reduce((a, b) => a + b, 0);

// Generate invite URLs
const baseUrl = 'https://discord.com/api/oauth2/authorize';
const params = new URLSearchParams({
  client_id: CLIENT_ID,
  permissions: totalPermissions.toString(),
  scope: 'bot applications.commands'
});

console.log('APBeeper Bot Invite Links:');
console.log('=====');
console.log(`Full Permissions: ${baseUrl}?${params.toString()}`);

// Minimal permissions for basic functionality
const minimalPermissions = permissions.SEND_MESSAGES +
  permissions.READ_MESSAGE_HISTORY + permissions.USE_SLASH_COMMANDS + permis-
  sions.EMBED_LINKS;
params.set('permissions', minimalPermissions.toString());
console.log(`Minimal Permissions: ${baseUrl}?${params.toString()}`);

```

Run with:

```
node scripts/generate_invite.js
```

Discord Bot Verification

1. Verification Requirements

For bots in 100+ servers, Discord requires verification:

Technical Requirements

- Bot must use slash commands (APBeeper supports this)
- Proper error handling and rate limiting
- No spam or abuse functionality
- Secure token handling

Documentation Requirements

- Clear bot description and functionality
- Privacy policy
- Terms of service
- Support server/contact information

2. Verification Process

Step 1: Prepare Documentation

Create the following documents:

Bot Description:

APBeeper - Advanced Discord Server Management

APBeeper is a comprehensive Discord bot designed for multi-server management with advanced features including:

Core Features

- **Role Management**: Automated role assignment and management
- **Channel Management**: Dynamic channel creation and organization
- **Moderation Tools**: Advanced moderation with logging and appeals
- **Multi-Server Support**: Manage multiple Discord communities
- **Custom Commands**: Server-specific command customization
- **Analytics**: Server activity and engagement metrics

Use Cases

- Large Discord communities requiring advanced management
- Gaming servers with complex role structures
- Educational servers with organized channels
- Business communities with moderation needs

Privacy & Security

- No personal data storage beyond Discord IDs
- All data encrypted and securely stored
- GDPR compliant data handling
- Regular security audits and updates

Step 2: Create Privacy Policy

```
# APBeeper Privacy Policy

## Data Collection
We collect only the minimum data necessary for bot functionality:
- Discord server IDs and names
- Discord user IDs (for command usage)
- Server configuration settings
- Command usage statistics (anonymized)

## Data Usage
Data is used exclusively for:
- Providing bot functionality
- Improving user experience
- Generating anonymized usage statistics

## Data Storage
- All data stored securely with encryption
- Regular backups with secure storage
- Data retention limited to active usage period
- Automatic deletion of inactive server data after 90 days

## Data Sharing
We do not share, sell, or distribute user data to third parties.

## User Rights
Users can request data deletion by contacting support.

## Contact
For privacy concerns: privacy@yourdomain.com
```

Step 3: Submit Verification Application

1. Go to Discord Developer Portal
2. Select your application
3. Navigate to “Bot” section
4. Click “Request Verification”
5. Fill out the verification form with:
 - Bot description
 - Privacy policy URL
 - Terms of service URL
 - Support server invite

3. Verification Checklist

- ☐ Bot uses slash commands
- ☐ Proper error handling implemented
- ☐ Rate limiting configured
- ☐ Privacy policy created and hosted
- ☐ Terms of service created
- ☐ Support server set up
- ☐ Bot description is clear and accurate
- ☐ No spam or malicious functionality

- [] Secure token and data handling

Website Integration

1. Create Bot Landing Page

Create a simple website for your bot at `website/index.html` :

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>APBeeper - Advanced Discord Bot</title>
  <meta name="description" content="APBeeper is a powerful Discord bot for server
management, moderation, and community building.">

  <style>
    * { margin: 0; padding: 0; box-sizing: border-box; }
    body { font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif; line-
height: 1.6; color: #333; }
    .container { max-width: 1200px; margin: 0 auto; padding: 0 20px; }

    header { background: linear-gradient(135deg, #667eea 0%, #764ba2 100%); color:
white; padding: 2rem 0; text-align: center; }
    h1 { font-size: 3rem; margin-bottom: 1rem; }
    .subtitle { font-size: 1.2rem; opacity: 0.9; }

    .invite-section { background: #f8f9fa; padding: 3rem 0; text-align: center; }
    .invite-btn { display: inline-block; background: #5865f2; color: white;
padding: 15px 30px; text-decoration: none; border-radius: 8px; font-size: 1.1rem; font-
weight: bold; transition: background 0.3s; }
    .invite-btn:hover { background: #4752c4; }

    .features { padding: 3rem 0; }
    .features-grid { display: grid; grid-template-columns: repeat(auto-fit,
minmax(300px, 1fr)); gap: 2rem; margin-top: 2rem; }
    .feature { background: white; padding: 2rem; border-radius: 8px; box-shadow: 0
2px 10px rgba(0,0,0,0.1); }
    .feature h3 { color: #5865f2; margin-bottom: 1rem; }

    footer { background: #2c2f33; color: white; padding: 2rem 0; text-align:
center; }
    .footer-links { margin-top: 1rem; }
    .footer-links a { color: #7289da; text-decoration: none; margin: 0 1rem; }
  </style>
</head>
<body>
  <header>
    <div class="container">
      <h1> APBeeper</h1>
      <p class="subtitle">Advanced Discord Server Management Made Simple</p>
    </div>
  </header>

  <section class="invite-section">
    <div class="container">
      <h2>Add APBeeper to Your Server</h2>
      <p>Get started with advanced Discord server management in seconds</p>
      <a href="https://discord.com/api/oauth2/authorize?cli-
ent_id=YOUR_CLIENT_ID&permissions=2147616870&scope=bot%20applications.commands"
class="invite-btn" target="_blank">
        Invite APBeeper
      </a>
    </div>
  </section>

```

```

<section class="features">
  <div class="container">
    <h2>Why Choose APBeeper?</h2>
    <div class="features-grid">
      <div class="feature">
        <h3> Advanced Moderation</h3>
        <p>Comprehensive moderation tools with logging, appeals, and auto-
mated actions to keep your server safe and organized.</p>
      </div>
      <div class="feature">
        <h3> Role Management</h3>
        <p>Sophisticated role assignment system with reaction roles,
temporary roles, and hierarchical management.</p>
      </div>
      <div class="feature">
        <h3> Channel Organization</h3>
        <p>Dynamic channel creation, categorization, and management tools
for growing communities.</p>
      </div>
      <div class="feature">
        <h3> Multi-Server Support</h3>
        <p>Manage multiple Discord servers from a single bot with server-
specific configurations.</p>
      </div>
      <div class="feature">
        <h3> Analytics & Insights</h3>
        <p>Detailed server analytics, member engagement metrics, and growth
tracking.</p>
      </div>
      <div class="feature">
        <h3> High Performance</h3>
        <p>Built for scale with efficient database operations and optimized
Discord API usage.</p>
      </div>
    </div>
  </div>
</section>

<footer>
  <div class="container">
    <p>&copy; 2025 APBeeper. All rights reserved.</p>
    <div class="footer-links">
      <a href="/privacy">Privacy Policy</a>
      <a href="/terms">Terms of Service</a>
      <a href="https://discord.gg/your-support-server">Support Server</a>
      <a href="https://github.com/yourusername/apbeeper-bot">GitHub</a>
    </div>
  </div>
</footer>

<script>
  // Track invite button clicks
  document.querySelector('.invite-btn').addEventListener('click', function() {
    // Add analytics tracking here if needed
    console.log('Invite button clicked');
  });
</script>

```



```
</body>
</html>
```

2. Host the Website

You can host the website on:

- **GitHub Pages** (free)
- **Netlify** (free tier available)
- **Vercel** (free tier available)
- **Your VPS** (using Nginx)

Hosting on Your VPS with Nginx

```
# Install Nginx
sudo apt install nginx

# Create website directory
sudo mkdir -p /var/www/apbeeper

# Copy website files
sudo cp -r website/* /var/www/apbeeper/

# Create Nginx configuration
sudo nano /etc/nginx/sites-available/apbeeper
```

Nginx configuration:

```
server {
    listen 80;
    server_name yourdomain.com www.yourdomain.com;
    root /var/www/apbeeper;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }

    # Security headers
    add_header X-Frame-Options "SAMEORIGIN" always;
    add_header X-XSS-Protection "1; mode=block" always;
    add_header X-Content-Type-Options "nosniff" always;
}
```

Enable the site:

```
sudo ln -s /etc/nginx/sites-available/apbeeper /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl reload nginx
```

User Onboarding

1. Welcome Message System

Implement a welcome system in your bot:

```
// Example welcome message when bot joins a server
client.on('guildCreate', async (guild) => {
  const welcomeEmbed = {
    color: 0x5865f2,
    title: ' Welcome to APBeeper!',
    description: 'Thank you for adding APBeeper to your server!',
    fields: [
      {
        name: ' Getting Started',
        value: 'Use `/help` to see all available commands\nUse `/setup` to
configure your server settings'
      },
      {
        name: ' Documentation',
        value: '[View Full Documentation](https://yourdomain.com/docs)\n[Join
Support Server](https://discord.gg/your-support-server)'
      },
      {
        name: '⚙ Quick Setup',
        value: '1. Run `/setup` to configure basic settings\n2. Set up modera-
tion with `/moderation setup`\n3. Configure roles with `/roles setup`'
      }
    ],
    footer: {
      text: 'Need help? Join our support server or check the documentation!'
    }
  };

  // Try to send to system channel or first available channel
  const channel = guild.systemChannel || guild.channels.cache.find(ch => ch.type ===
'GUILD_TEXT' && ch.permissionsFor(guild.members.me).has('SEND_MESSAGES'));

  if (channel) {
    await channel.send({ embeds: [welcomeEmbed] });
  }
});
```

2. Setup Command

Create a comprehensive setup command:

```
// /setup command implementation
const { SlashCommandBuilder } = require('discord.js');

module.exports = {
  data: new SlashCommandBuilder()
    .setName('setup')
    .setDescription('Interactive setup wizard for APBeeper'),

  async execute(interaction) {
    if (!interaction.member.permissions.has('MANAGE_GUILD')) {
      return interaction.reply({ content: 'You need Manage Server permission to run setup.', ephemeral: true });
    }

    const setupEmbed = {
      color: 0x5865f2,
      title: ' APBeeper Setup Wizard',
      description: 'Let\'s configure APBeeper for your server!',
      fields: [
        {
          name: '1 Basic Configuration',
          value:
            '\n/config prefix` - Set command prefix\n\n/config timezone` - Set server timezone'
        },
        {
          name: '2 Moderation Setup',
          value:
            '\n/moderation setup` - Configure moderation settings\n\n/moderation logs` - Set up moderation logging'
        },
        {
          name: '3 Role Management',
          value:
            '\n/roles setup` - Configure role management\n\n/roles reaction` - Set up reaction roles'
        },
        {
          name: '4 Channel Management',
          value:
            '\n/channels setup` - Configure channel management\n\n/channels auto` - Set up auto-channels'
        }
      ],
      footer: {
        text: 'Run each command to configure that feature. Use /help for detailed information.'
      }
    };

    await interaction.reply({ embeds: [setupEmbed] });
  }
};
```

3. Documentation Integration

Create comprehensive help system:

```
// Enhanced help command
const { SlashCommandBuilder, EmbedBuilder } = require('discord.js');

module.exports = {
  data: new SlashCommandBuilder()
    .setName('help')
    .setDescription('Get help with APBeeper commands')
    .addStringOption(option =>
      option.setName('category')
        .setDescription('Command category')
        .addChoices(
          { name: 'Moderation', value: 'moderation' },
          { name: 'Roles', value: 'roles' },
          { name: 'Channels', value: 'channels' },
          { name: 'Utility', value: 'utility' }
        )
    ),

  async execute(interaction) {
    const category = interaction.options.getString('category');

    if (!category) {
      // Show general help
      const helpEmbed = new EmbedBuilder()
        .setColor(0x5865f2)
        .setTitle(' APBeeper Help')
        .setDescription('APBeeper is a comprehensive Discord server management bot.')
        .addFields(
          { name: ' Categories', value: 'Use `/help category:<name>` to see commands in each category' },
          { name: ' Moderation', value: 'Advanced moderation tools and logging' },
          { name: ' Roles', value: 'Role management and assignment' },
          { name: ' Channels', value: 'Channel creation and organization' },
          { name: ' Utility', value: 'Server utilities and information' }
        )
        .addFields(
          { name: ' Links', value: '[Documentation](https://yourdomain.com/docs) | [Support Server](https://discord.gg/your-support) | [Invite Bot](https://yourdomain.com/invite)' }
        )
        .setFooter('');

      return interaction.reply({ embeds: [helpEmbed] });
    }

    // Show category-specific help
    const categoryHelp = getCategoryHelp(category);
    await interaction.reply({ embeds: [categoryHelp] });
  }
};
```

Bot Listing Platforms

1. Top.gg (Discord Bot List)

- **URL:** <https://top.gg>
- **Requirements:** Detailed description, screenshots, invite link

- **Benefits:** Large user base, voting system, analytics

Submission Checklist for Top.gg:

- ☐ Bot description (minimum 300 characters)
- ☐ High-quality bot avatar
- ☐ Screenshots of bot in action
- ☐ Proper categorization
- ☐ Invite link with correct permissions
- ☐ Support server link
- ☐ Website URL (optional but recommended)

2. Discord.bots.gg

- **URL:** <https://discord.bots.gg>
- **Requirements:** Similar to Top.gg
- **Benefits:** Clean interface, good SEO

3. Bots on Discord

- **URL:** <https://bots.ondiscord.xyz>
- **Requirements:** Basic bot information
- **Benefits:** Simple submission process

4. Discord Bot List

- **URL:** <https://discordbotlist.com>
- **Requirements:** Comprehensive bot details
- **Benefits:** Detailed analytics, review system

5. Submission Template

Use this template for bot listings:

APBeeper - Advanced Discord Server Management

Short Description

Comprehensive Discord bot for server management, moderation, and community building with multi-server support.

Long Description

APBeeper is a powerful, feature-rich Discord bot designed to help server administrators manage their communities effectively. With advanced moderation tools, sophisticated role management, dynamic channel organization, and comprehensive analytics, APBeeper scales from small communities to large Discord servers.

Key Features:

****Advanced Moderation****

- Automated moderation with customizable rules
- Comprehensive logging and audit trails
- Appeal system for moderation actions
- Temporary and permanent punishment options

****Role Management****

- Reaction roles with multiple message support
- Temporary role assignments
- Role hierarchies and permissions
- Automated role assignment based on activity

****Channel Management****

- Dynamic channel creation and deletion
- Category organization and management
- Voice channel automation
- Channel templates and cloning

****Multi-Server Support****

- Manage multiple Discord servers
- Server-specific configurations
- Cross-server analytics and insights
- Centralized administration panel

****Analytics & Insights****

- Member engagement tracking
- Server growth analytics
- Command usage statistics
- Activity heatmaps and trends

< ****Performance & Reliability****

- 99.9% uptime guarantee
- Optimized for large servers
- Rate limit handling
- Automatic error recovery

Commands Preview

- `/moderation ban @user reason`` - Ban a user with reason
- `/roles add @user @role`` - Assign role to user
- `/channels create #name category`` - Create new channel
- `/analytics server`` - View server statistics
- `/help`` - Comprehensive help system

Setup Instructions

1. Invite APBeeper to your server

2. Run ``/setup`` for interactive configuration
3. Configure moderation with ``/moderation setup``
4. Set up roles with ``/roles setup``
5. Join our support server for assistance

Support

- Support Server: <https://discord.gg/your-support-server>
- Documentation: <https://yourdomain.com/docs>
- Website: <https://yourdomain.com>
- GitHub: <https://github.com/yourusername/apbeeper-bot>

Privacy & Security

APBeeper is committed to user privacy and data security. We collect only necessary data for bot functionality and never share user information with third parties. All data is encrypted and securely stored with regular backups.

Tags

moderation, management, roles, channels, analytics, multi-server, automation, utility, admin, community

Analytics and Monitoring

1. Bot Usage Analytics

Implement analytics tracking in your bot:

```

// Analytics tracking system
const analytics = {
  commandUsage: new Map(),
  serverCount: 0,
  userCount: 0,

  trackCommand(commandName, guildId) {
    const key = `${commandName}:${guildId}`;
    this.commandUsage.set(key, (this.commandUsage.get(key) || 0) + 1);

    // Log to database or external service
    this.logToDatabase({
      type: 'command_usage',
      command: commandName,
      guild_id: guildId,
      timestamp: new Date()
    });
  },

  updateStats() {
    this.serverCount = client.guilds.cache.size;
    this.userCount = client.guilds.cache.reduce((acc, guild) => acc + guild.memberC
ount, 0);
  },

  getTopCommands(limit = 10) {
    return Array.from(this.commandUsage.entries())
      .sort(([,a], [,b]) => b - a)
      .slice(0, limit);
  }
};

// Track command usage
client.on('interactionCreate', async (interaction) => {
  if (!interaction.isChatInputCommand()) return;

  analytics.trackCommand(interaction.commandName, interaction.guildId);

  // Your existing command handling code
});

```

2. External Analytics Integration

Consider integrating with:

- **Google Analytics** for website tracking
- **Mixpanel** for detailed event tracking
- **DataDog** for infrastructure monitoring
- **Sentry** for error tracking

3. Dashboard Creation

Create an analytics dashboard:


```
// Simple analytics API endpoint
app.get('/api/stats', async (req, res) => {
  const stats = {
    servers: client.guilds.cache.size,
    users: client.guilds.cache.reduce((acc, guild) => acc + guild.memberCount, 0),
    commands_today: await getCommandsToday(),
    uptime: process.uptime(),
    version: require('./package.json').version
  };

  res.json(stats);
});
```

Support and Documentation

1. Support Server Setup

Create a dedicated Discord server for support:

Server Structure:

```
Information
├── rules
├── announcements
└── faq

Support
├── create-ticket
├── general-support
├── bug-reports
└── feature-requests

Community
├── general-chat
├── showcase
└── server-configs

Development
├── bot-status
├── updates
└── beta-testing
```

2. Ticket System

Implement a ticket system for support:

```

// Ticket creation system
const { ActionRowBuilder, ButtonBuilder, ButtonStyle } = require('discord.js');

// Create ticket button
const ticketButton = new ActionRowBuilder()
  .addComponents(
    new ButtonBuilder()
      .setCustomId('create_ticket')
      .setLabel('Create Support Ticket')
      .setStyle(ButtonStyle.Primary)
      .setEmoji(' ')
  );

// Handle ticket creation
client.on('interactionCreate', async (interaction) => {
  if (!interaction.isButton()) return;

  if (interaction.customId === 'create_ticket') {
    // Create private channel for user
    const ticketChannel = await interaction.guild.channels.create({
      name: `ticket-${interaction.user.username}`,
      type: ChannelType.GuildText,
      parent: 'SUPPORT_CATEGORY_ID',
      permissionOverwrites: [
        {
          id: interaction.guild.id,
          deny: [PermissionFlagsBits.ViewChannel]
        },
        {
          id: interaction.user.id,
          allow: [PermissionFlagsBits.ViewChannel, PermissionFlags-
Bits.SendMessages]
        }
      ]
    });

    await interaction.reply({
      content: `Ticket created: ${ticketChannel}`,
      ephemeral: true
    });
  }
});

```

3. FAQ System

Create comprehensive FAQ:

Frequently Asked Questions

General Questions

Q: How do I add APBeeper to my server?

A: Click the invite link on our website or use this direct link: [**Invite APBeeper**](https://discord.com/api/oauth2/authorize?client_id=YOUR_CLIENT_ID&permissions=2147616870&scope=bot%20applications.commands)

Q: What permissions does APBeeper need?

A: APBeeper needs specific permissions to function properly. The invite link includes all necessary permissions. You can review them in the Discord authorization screen.

Q: Is APBeeper free to use?

A: Yes! APBeeper is completely free to use with all features available to all servers.

Setup Questions

Q: How do I configure APBeeper for my server?

A: Run the `/setup` command after inviting the bot. This will guide you through the configuration process.

Q: Can I customize APBeeper's settings?

A: Yes! APBeeper offers extensive customization options. Use `/config` commands to modify settings for your server.

Troubleshooting

Q: APBeeper isn't responding to commands

A: Check that:

- APBeeper has the necessary permissions
- You're using slash commands (type `/` to see available commands)
- APBeeper is online (check our status page)

Q: How do I report a bug?

A: Join our support server and use the `#bug-reports` channel, or create a support ticket.

Feature Questions

Q: Does APBeeper support multiple servers?

A: Yes! APBeeper can manage multiple Discord servers with server-specific configurations.

Q: Can I use APBeeper for moderation?

A: Absolutely! APBeeper includes comprehensive moderation tools including bans, kicks, mutes, and logging.

Legal and Privacy Considerations

1. Privacy Policy Template

APBeeper Privacy Policy

****Last Updated:**** June 23, 2025

Introduction

This Privacy Policy describes how APBeeper ("we," "our," or "us") collects, uses, and protects information when you use our Discord bot service.

Information We Collect

Automatically Collected Information

- Discord server IDs and names
- Discord user IDs (for command execution)
- Command usage data (anonymized)
- Server configuration settings
- Message IDs (for moderation features only)

Information You Provide

- Server configuration preferences
- Custom command settings
- Moderation settings and rules

How We Use Information

We use collected information to:

- Provide and maintain bot functionality
- Improve user experience and bot performance
- Generate anonymized usage statistics
- Provide customer support
- Ensure security and prevent abuse

Data Storage and Security

- All data is encrypted in transit and at rest
- Data is stored on secure servers with regular backups
- Access to data is limited to authorized personnel only
- We implement industry-standard security measures

Data Retention

- Active server data is retained while the bot is in use
- Inactive server data is automatically deleted after 90 days
- Users can request immediate data deletion

Data Sharing

We do not sell, trade, or share your data with third parties except:

- When required by law
- To protect our rights or safety
- With your explicit consent

Your Rights

You have the right to:

- Request access to your data
- Request correction of inaccurate data
- Request deletion of your data
- Withdraw consent for data processing

Children's Privacy

APBeeper is not intended for users under 13 years of age. We do not knowingly collect personal information from children under 13.

Changes to This Policy

We may update this Privacy Policy periodically. Changes will be posted on our website and in our support server.

Contact Us

For privacy-related questions or requests:

- Email: privacy@yourdomain.com
- Discord: Join our support server
- Website: <https://yourdomain.com/contact>

2. Terms of Service Template

APBeeper Terms of Service

****Last Updated:**** June 23, 2025

Acceptance of Terms

By using APBeeper, you agree to these Terms of Service and our Privacy Policy.

Description of Service

APBeeper is a Discord bot that provides server management, moderation, and community building features.

User Responsibilities

You agree to:

- Use the service in compliance with Discord's Terms of Service
- Not use the bot for illegal or harmful activities
- Not attempt to exploit or abuse the service
- Respect other users and communities

Prohibited Uses

You may not use APBeeper to:

- Spam or harass other users
- Distribute malicious content
- Violate Discord's community guidelines
- Circumvent Discord's rate limits or restrictions

Service Availability

- We strive for 99.9% uptime but cannot guarantee uninterrupted service
- We may perform maintenance that temporarily affects availability
- We reserve the right to modify or discontinue features

Limitation of Liability

APBeeper is provided "as is" without warranties. We are not liable for:

- Service interruptions or data loss
- Actions taken by server administrators using the bot
- Indirect or consequential damages

Termination

We may terminate or suspend access to APBeeper for:

- Violation of these terms
- Abuse of the service
- Legal requirements

Changes to Terms

We may modify these terms at any time. Continued use constitutes acceptance of changes.

Contact Information

For questions about these terms:

- Email: legal@yourdomain.com
- Discord: Join our support server
- Website: <https://yourdomain.com/contact>

3. GDPR Compliance

Ensure GDPR compliance with:

```

// GDPR data handling functions
const gdprCompliance = {
  // Handle data deletion requests
  async deleteUserData(userId) {
    await database.deleteUserData(userId);
    await database.anonymizeUserLogs(userId);
    console.log(`GDPR: Deleted data for user ${userId}`);
  },

  // Export user data
  async exportUserData(userId) {
    const userData = await database.getUserData(userId);
    return {
      user_id: userId,
      servers: userData.servers,
      command_usage: userData.commands,
      created_at: userData.created_at,
      last_active: userData.last_active
    };
  },

  // Handle data portability requests
  async generateDataExport(userId) {
    const data = await this.exportUserData(userId);
    return JSON.stringify(data, null, 2);
  }
};

// GDPR command for users
const gdprCommand = {
  data: new SlashCommandBuilder()
    .setName('gdpr')
    .setDescription('GDPR data management')
    .addSubcommand(subcommand =>
      subcommand
        .setName('export')
        .setDescription('Export your data'))
    .addSubcommand(subcommand =>
      subcommand
        .setName('delete')
        .setDescription('Delete your data')),

  async execute(interaction) {
    const subcommand = interaction.options.getSubcommand();

    if (subcommand === 'export') {
      const data = await gdprCompliance.generateDataExport(interaction.user.id);
      // Send data as file attachment
      await interaction.reply({
        content: 'Your data export:',
        files: [{ attachment: Buffer.from(data), name: 'my_data.json' }],
        ephemeral: true
      });
    } else if (subcommand === 'delete') {
      await gdprCompliance.deleteUserData(interaction.user.id);
      await interaction.reply({
        content: 'Your data has been deleted from our systems.',
        ephemeral: true
      });
    }
  }
};

```



```

    });
  }
};

```

Launch Checklist

Pre-Launch

- ☐ Bot functionality thoroughly tested
- ☐ Documentation completed and hosted
- ☐ Privacy policy and terms of service created
- ☐ Support server set up and configured
- ☐ Website created with invite links
- ☐ Analytics and monitoring implemented

Launch Day

- ☐ Submit to bot listing platforms
- ☐ Announce in relevant Discord communities
- ☐ Share on social media platforms
- ☐ Monitor for issues and user feedback
- ☐ Respond to support requests promptly

Post-Launch

- ☐ Gather user feedback and iterate
- ☐ Monitor analytics and usage patterns
- ☐ Regular updates and feature additions
- ☐ Community building and engagement
- ☐ Consider Discord verification if growing rapidly

Success Tip: Focus on providing excellent user experience and support. Happy users are your best marketing tool!