

Chrome拡張機能がPuppeteerで読み込まれない問題の分析

🔍 問題の概要 (Summary of the Issue)

Chrome DevTools MCPプロジェクトにおいて、Puppeteer経由でChromeを起動する際に、`--load-extension` フラグで指定した拡張機能が読み込まれない問題が発生しています。起動コマンドの引数には確かに拡張機能のパスが含まれており、`ignoreDefaultArgs` オプションで `--disable-extensions` も無効化しています。しかし、実際には以下の不具合が確認されています：

- **拡張機能が無効:** Chrome起動後に `chrome://extensions/` ページを確認しても、拡張機能が表示されない。
- **スクリプト未注入:** 対象ウェブページに拡張機能のcontent scriptや背景のスクリプトが全く読み込まれていない。
- **UI非表示:** ブラウザツールバーに拡張機能のアイコンが表示されず、機能が動作しない。

この問題は、Chromeを手動で起動し拡張機能を読み込んだ場合や、通常のユーザープロファイルで拡張機能をロードした場合には発生せず、Puppeteerの自動制御下でChromeを起動した場合にのみ発生しています。

🔍 原因分析 (Root Cause Analysis)

結論から言うと、**Chromeブラウザ (公式ビルド) のバージョン137以降で導入された仕様変更**が原因であると判明しました。Googleはセキュリティと安定性向上のため、**コマンドラインフラグによる拡張機能の読み込み (`--load-extension`) を公式Chromeでは無効化する措置**を取りました ¹ ²。これにより、PuppeteerなどでChrome (公式ビルド) を自動起動する際に `--load-extension` を指定しても、**Chromeがそのフラグを無視するようになっています**。

- **Chrome 137以降の変更:** Chrome 137版以降の公式リリースでは、`--load-extension` フラグによる拡張読み込みが徐々に**非推奨化**され、将来的には無効になる旨がGoogleからアナウンスされています ¹。この変更は悪意のある拡張機能のリスク軽減などが目的です。
- **公式ビルド vs テスト用ビルド:** この制限は**Google公式のChromeブラウザ**に適用され、**Chrome for Testing**や**Chromiumなどの非公式ビルド**では引き続き `--load-extension` が利用可能です ²。公式ビルドのChrome 137では無効でも、開発者向けのテスト用Chromeでは有効、という差異があります。
- **Puppeteer環境への影響:** PuppeteerでChromeを起動する際、既定ではGoogle公式のChrome (またはインストール済みのChrome) を使用します。そのため**Chrome 137+ 環境下では、自動制御モード (`--enable-automation` が有効) のChromeでは拡張機能読み込みフラグが効かず**、拡張機能が一切読み込まれないという現象が発生します。エラーメッセージは表示されず**サイレントに無視**されるため、問題の発見が困難でした。

このように、問題の本質は**Chromeブラウザ自体の仕様変更**であり、Puppeteerや拡張機能側の不備ではありません。実際、他の自動化ツール (WebDriverIOやSeleniumなど) でもChrome 137アップデート以降に同様の報告が相次いでいます ³ ⁴。

補足情報・証拠 (Supporting Evidence)

- Googleの公式発表によれば、「Chrome 137以降、公式Chromeブランドのビルドでは `--load-extension` による拡張読み込みができなくなる。未パッケージの拡張機能を読み込むには、開発者モードで `chrome://extensions/` から手動で読み込む必要がある。開発者は **Chromium** や **Chrome for Testing** などの非ブランド版Chromeでは引き続きコマンドラインフラグを使用できる」ことが明記されています ¹ ²。
- 実際のコミュニティ報告では、「Chrome公式ビルド137で `--load-extension` オプションが削除された。この変更はChrome for TestingやChromiumには影響しない」ことが強調されています ³。この問題は自動テスト界隈でも「重大な事態」 (“Major situation!”) と認識されており、多くのユーザーが影響を受けています ⁵。
- SeleniumやKatalonといったツールでも同様の問題が発生し、開発者たちは回避策を模索しています。例えば、SeleniumベースのテストフレームワークではChrome 137+で拡張機能を動作させるため、内部的に特定のChrome機能フラグを無効化するアップデートが行われました ⁶。

⁶ では、SeleniumBase 4.39.1において `--disable-features=DisableLoadExtensionCommandLineSwitch` というフラグが追加されたことが記されています。このフラグを付与すると、Chrome公式版137でも従来通り拡張機能を有効化できるようになり、Chrome 136以前と同様に動作するとのこと ⁶。

原因のまとめ (Summary of Root Cause)

以上より、本件の根本原因は**Chrome公式ビルドのポリシー変更**です。Chrome 137以降では、コマンドラインからの拡張機能読み込みが標準でブロックされるため、Puppeteer経由の自動実行では拡張機能を読み込まれません。これは**セキュリティ強化策**として意図された仕様であり、Puppeteerや拡張機能のコードに問題があったわけではありません。

解決策と対処法 (Solutions and Workarounds)

この問題を解決・回避するために、以下の対処法が考えられます：

- Chrome公式ビルドの代替を使用:** 自動テスト用途には公式Chromeではなく、「**Chrome for Testing**」(テスト専用のChromeビルド)や**オープンソース版Chromium**を使用します。これらのビルドでは引き続き `--load-extension` フラグが有効なため、Puppeteerでも問題なく拡張機能をロードできます ² ⁵。Googleも公式に、開発者はChrome for Testing等を利用するよう案内しています。
- Chrome起動フラグで強制的に許可:** どうしても公式Chromeを使う必要がある場合は、Chrome起動時のフラグに `--disable-features=DisableLoadExtensionCommandLineSwitch` を追加します。このフラグは「`--load-extension` を無効化するChromeの内部機能」をオフにするもので、これを指定すればChrome 137+の公式ビルドでも従来通り拡張機能を読み込めます ⁷ ⁶。実際、先述のSeleniumベースのプロジェクトではこの方法でChrome 137以降への対応が行われました。
- 一時的なダウングレード:** 短期的な回避策として、Chromeブラウザをバージョン136以下にダウングレードする方法もあります。ただし、これは根本的解決ではなくセキュリティリスクも伴うため推奨されません ⁸。将来的にはChrome 137+に対応した環境への移行が必要です。
- 別ブラウザの利用:** Chrome互換の別ブラウザ(例: Braveブラウザ)ではこの制限がない場合があります ⁵。BraveなどChromium派生ブラウザは `--load-extension` が使用可能なため、Puppeteerの `executablePath` にそれらを指定することで拡張機能を読み込める可能性があります。ただしプロジェクト要件によります。

今後の展望 (Next Steps)

- **プロジェクトでの反映:** Chrome DevTools MCPプロジェクトでは、上記の対策を踏まえてChrome起動方法を見直す必要があります。最も現実的なのは、開発・テスト用途としてGoogle提供の**Chrome for Testing**を導入し、Puppeteerの起動バイナリをそちらに切り替える方法です。これにより最小限の変更で拡張機能を有効化できます。
- **起動オプションの更新:** もし公式Chromeを引き続き使う場合は、Puppeteer起動時の引数に前述の `--disable-features=DisableLoadExtensionCommandLineSwitch` を追加してください。MCPの設定ファイルや `browser.ts` の起動オプションに、このフラグを条件付きで付与する実装を検討すべきです。
- **情報収集と検証:** Chromeブラウザのリリースノートや自動テスト関連のフォーラムをウォッチし、拡張機能周りの仕様変更情報を収集しましょう。今回の変更は各所で大きな話題となっているため、同様のケーススタディやベストプラクティスが今後さらに共有される可能性があります。
- **将来のプロトコル対応:** GoogleはWebDriver BiDiなど新しい自動化インターフェースを推進しており、今後は**公式API経由で拡張機能をインストール**できる手段が提供されるかもしれません⁹。現状では利用できませんが、将来的なアップデートでBiDiプロトコルを活用した拡張機能制御が可能になれば、本問題の根本的な解決につながるでしょう。

まとめ (Conclusion)

今回の調査により、Chrome拡張機能がPuppeteer経由で読み込まれなかったのは、**Chrome公式版 (バージョン137以降) のポリシー変更**によるものと判明しました。Puppeteer自体や拡張機能コードに問題はなく、**Chrome側のセキュリティ措置**が原因です。そのため、Chrome for Testingの使用や特定フラグの付与といった対処で問題を回避・解決できます。

本件はChromeの設計変更に起因するため、今後も類似のセキュリティ強化策が導入される可能性があります。常に最新の公式情報を確認しつつ、プロジェクトのツールチェーンを適切にアップデートして対応していくことが重要です。

参考資料: Chrome拡張機能読み込み無効化に関するGoogle公式ドキュメントおよびコミュニティディスカッション^{1 2 3 6} など。今回判明した知見をもとに、プロジェクトのChrome起動設定を更新してください。

¹ ² ³ ⁴ ⁵ ⁶ ⁷ ⁸ ⁹ Katalon 10.2.1 now using Chrome for Testing? - Katalon Studio - Katalon Community

<https://forum.katalon.com/t/katalon-10-2-1-now-using-chrome-for-testing/175420>