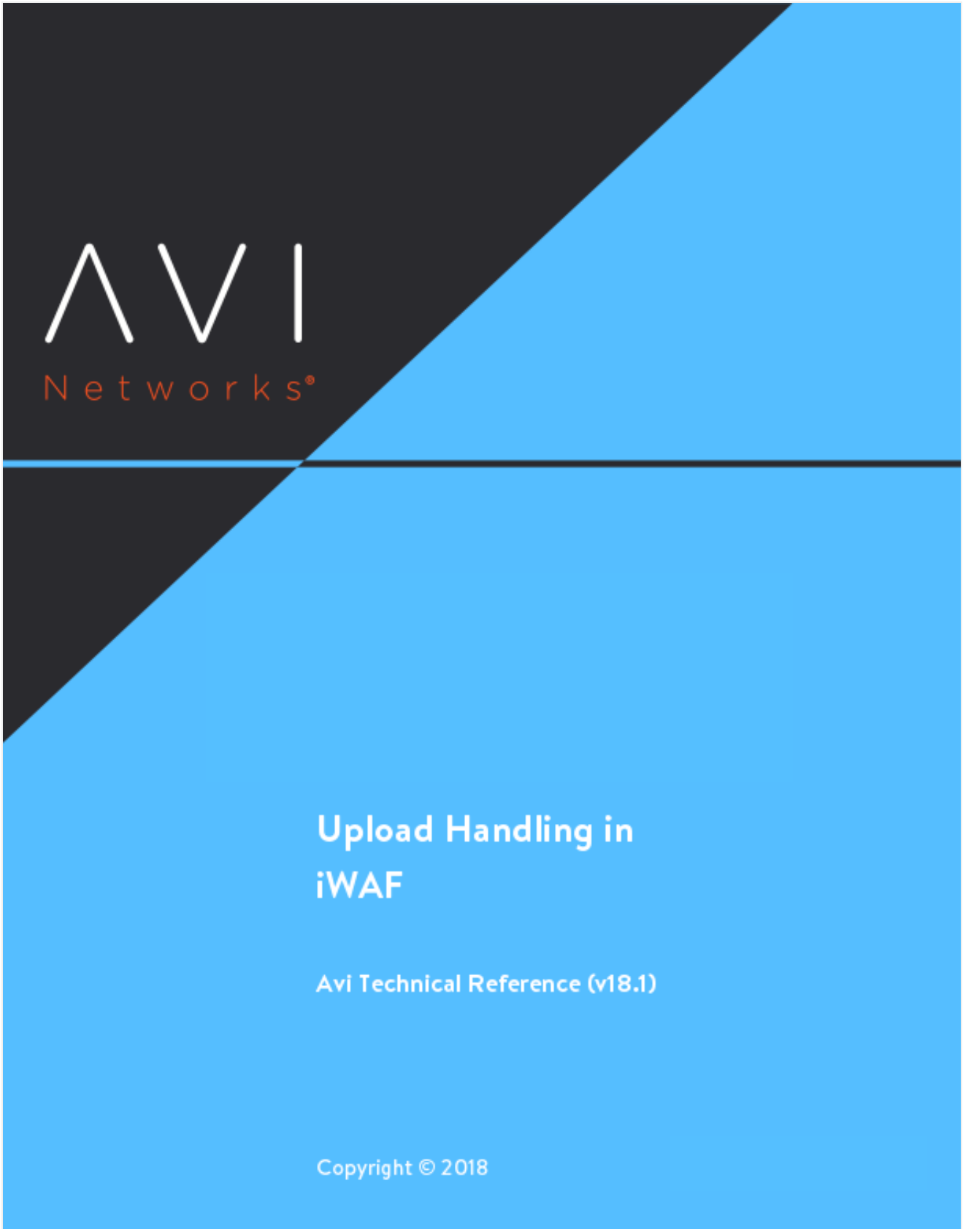


Upload Handling in iWAF

Avi Technical Reference (v18.1)

Upload Handling in iWAF

view online

Overview

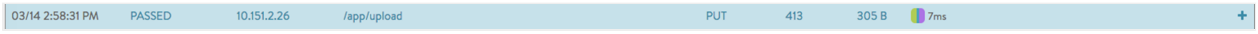
After enabling a WAF policy on a virtual service, certain requests can be blocked with a 413 Request Entity Too Large message. However, there are a few file extensions that are bypassed from WAF check, as they are static content. For more information on configuring static extensions, refer to Configuring WAF Profile.

iWAF could trigger an alert or block uploads for requests such as the following:
* File size exceeding the limit set in either WAF profile or HTTP profile of the virtual service. * Random match of System-Default-Policy rules on binary upload data. * Exceeding regex match limit with rules running too long on the input, due to which iWAF will terminate the execution.

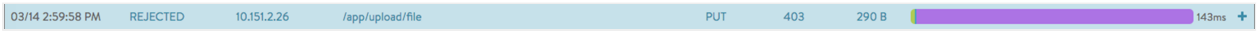
Note: Caching each upload for inspection which will result in a larger Service Engine memory footprint (Default profile allows only 1 MB uploads).

The following are a few examples for specific uploads and the corresponding WAF log entries:

Example 1:
* Request is denied with a 413 message for exceeding the size limit. * As WAF did not inspect the request, WAF status is PASSED.



Example 2:
* Size limit has been increased and so no limit was hit. * Request is denied with a 403 message. * Coincidentally, as parts of the PDF matched the WAF CRS rules, the request is rejected and the status is REJECTED.



To ensure that legitimate requests do not trigger an alert or are not blocked, iWAF needs to be configured to handle large uploads.

Configuration Summary

Configuration changes to the following are recommended to enable upload bypass:

- Client Post Body Size parameter configured for global virtual service HTTP policy under DDoS.
- Maximum file upload size configured in WAF profile used in the virtual service WAF policy. (However, the recommendation is to bypass the upload URLs, as explained later).
- Custom PRE-CRS rule within the attached virtual service WAF policy.

Configuration

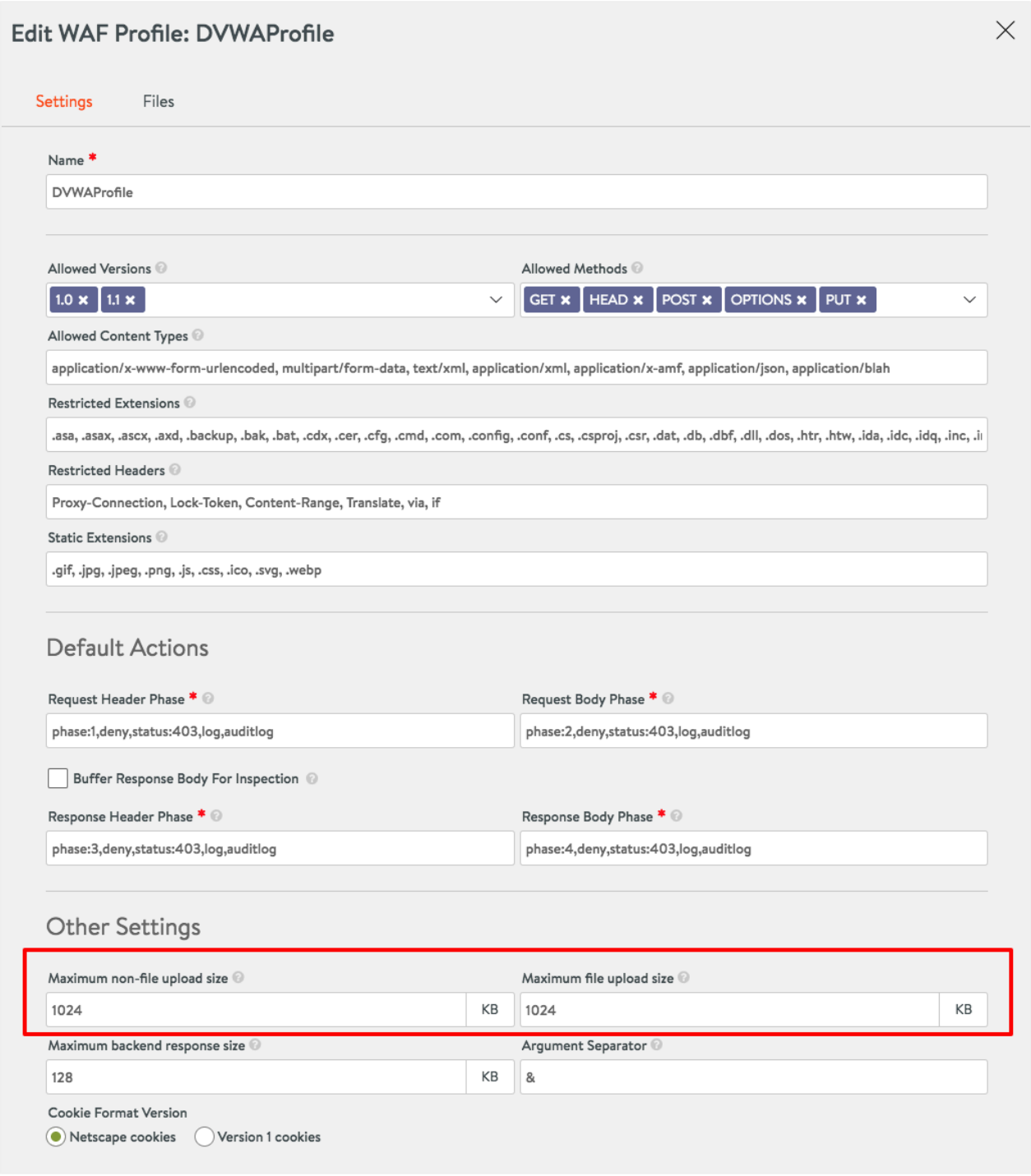
Follow one of the examples provided below to bypass iWAF for large file uploads.

Note: Use application limit guidelines for defining the configured limits.

You can enable upload bypass by changing the parameters for: * WAF profile * WAF policy * VS-HTTP profile

WAF Profile

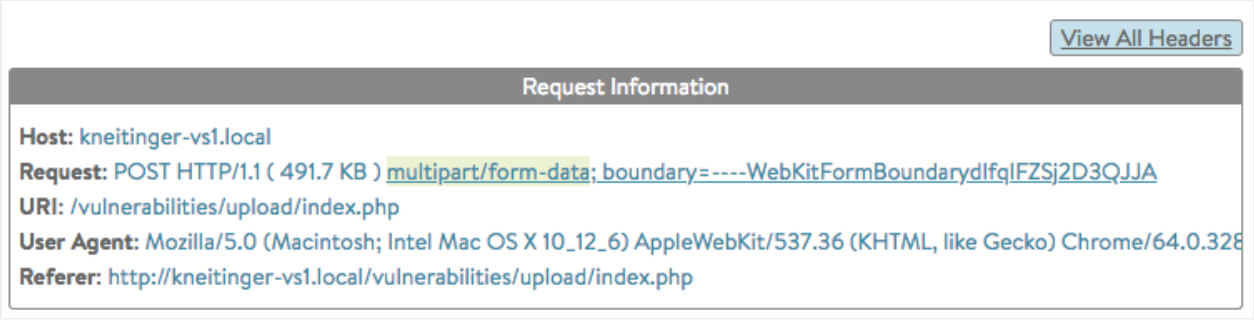
On Avi UI, navigate to Templates > WAF > WAF Profile, and choose the relevant one from the listed profiles.



Under Other Settings, configure the following fields:

- Maximum non-file upload size: This value is used for any request that is not sent with a multipart or form-data header. For instance,

    - PUT requests with application or json.
    - Other POST requests that have a larger body size.
- Maximum file upload size: This value is used only if a request uses multiport or form-data as request Content-Type. For uploads using a form on a web page, the browser will send the Content-Type header, as shown in the screenshot below:



If the Content-Length header of the request is bigger than the supplied Maximum file upload size value, then the request will be blocked with a 413 message.

Recommendation

It is recommended to set the Maximum non-file upload size and Maximum file upload size to the same value. Ideally the application will have its own limit, which can be used for the iWAF setting.

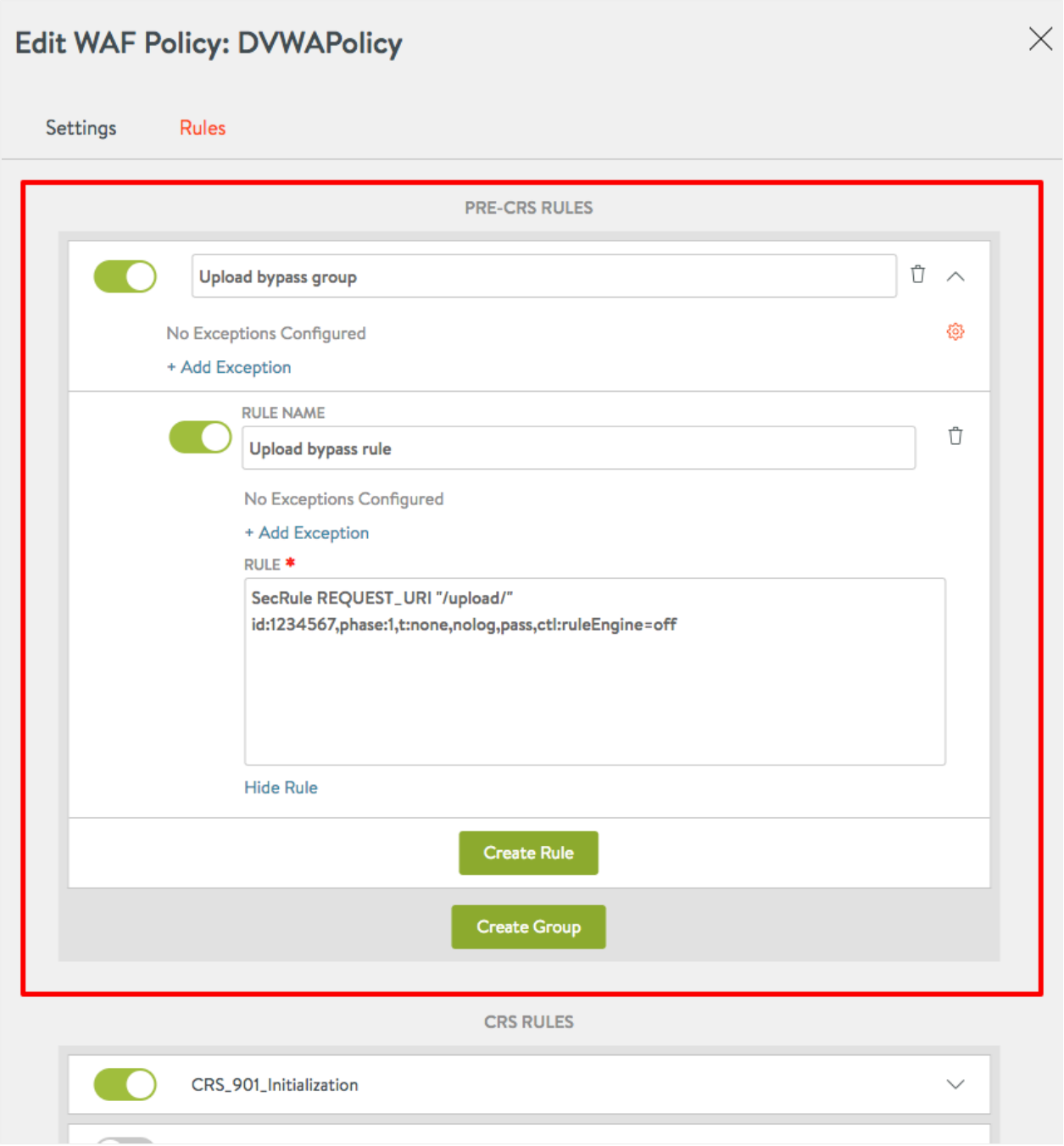
For instance, if the Application size limit is 1024 KB, then the Maximum non-file upload size and Maximum file upload size can also be set to 1024 KB.

This will ensure that the iWAF will not interfere with the application response and will deal with larger uploads before they can reach the application.

WAF Policy

On Avi UI, navigate to Templates > WAF > WAF Policy, and choose the relevant one from the listed policies.

Under Rules > PRE-CRS RULES, create an upload bypass rule as shown in the screenshot below.



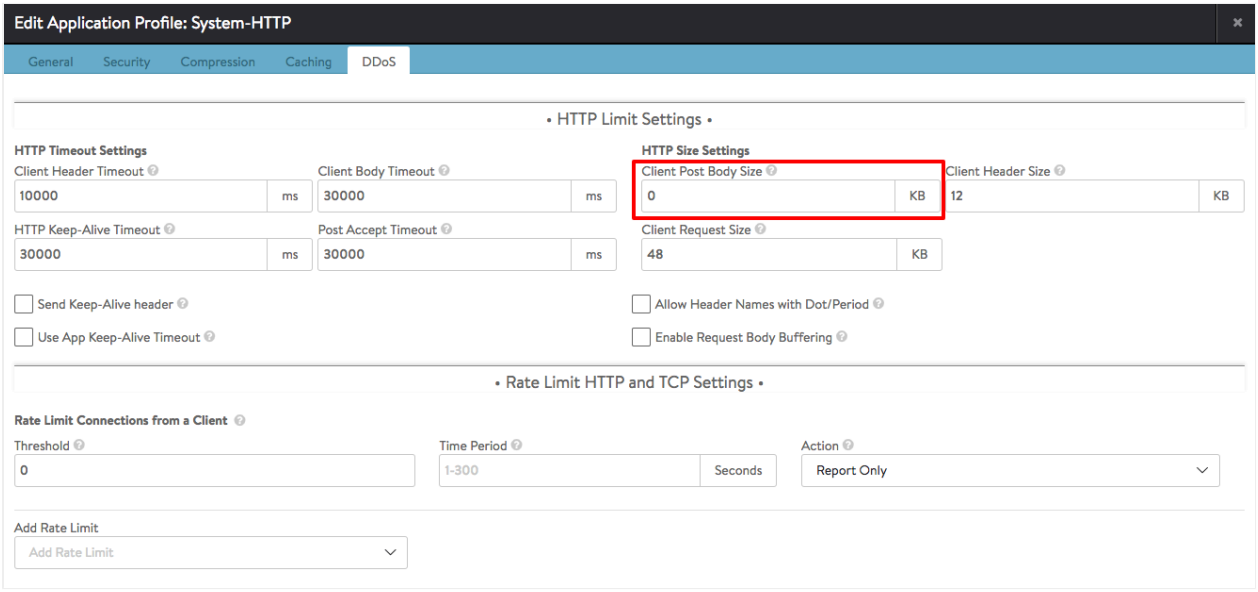
For this custom bypass rule, the ID should be changed to either the local range of 0 to 99.999 or a private reserved range, as explained at the link here.

Recommendation

You can add a PRE-CRS rule with customization to protect your application. For more information on language reference, refer to ModSecurity Handbook.

Virtual Service - HTTP Profile

On Avi UI, navigate to Templates > Profiles. Under Application tab choose System-HTTP. Click on edit and navigate to the DDoS tab.



Client Post Body Size is the maximum body size of a client request. This limits the size of a client POST as a part of a single HTTP request. In case of an iWAF bypass rule, this setting is over ridden and is not considered. If no iWAF bypass rule is configured for uploads, then this setting is considered.

If the Client Post Body Size is configured for a value lesser than the Maximum file upload size, then the buffering will fail with a 413 error message in the proxy, before it reaches the iWAF. To fix this, you need to update the value of Client Post Body Size in the application profile to increase the size, so that it is greater than the Maximum file upload size value configured in an iWAF Profile.

Note: If the Client Post Body Size is set to a default value of 0, which refers to no-limit, then this value will always be greater than the Maximum file upload size limit configured in an iWAF profile.

Modsec Bypass Rules

The following are a few examples for modsec bypass rules. It is recommended to configure this using URLs. The ID should either be within the local range of 0 to 99.999 or a private reserved range, as explained at the link here. The numbers are choosen here to explain the example. Ensure unique rule IDs in your deployment.

Single URL

```
SecRule REQUEST_URI "@rx /app/upload/" id:90001,phase:1,t:none,nolog,pass,ctl:ruleEngine=off
```

Multiple URLs

```
SecRule REQUEST_URI "@rx /app/upload/|/app/upload_two/|/app/upload_three/" id:90002,phase:1,t:none,nolog,pass,ctl:ruleE
```

This rule can be altered using other OPERATORS such as *@contains*, *@startwith*.

By Content-Length

```
SecRule REQUEST_HEADERS:Content-Length ?@gt 1048576? phase:1,id:90003,nolog,pass,ctl:ruleEngine=off
```

Note: It is recommended to configure rules using the URL, instead of Content-Length.

The rules provided should have a number within the Avi Vantage recommended range, as explained in the ModSecurity Handbook.

Frequently Asked Questions

Does enabling upload bypass affect uploads in Detection mode ?
Failure to specify and configure uploads will only result in a block if the size entered within the HTTP profile is exceeded (default is 0, unlimited). This configuration is out of the iWAF's scope and will therefore still block a request.

What is the effect on malicious uploads ?
These are cases where an attacker might want to smuggle a malicious upload onto a server. Such an upload might contain malware, ransomware, viruses, other file based exploits (pdf reader exploits) among many others. It is recommended to use a virus malware scanning tool on the upload directory of the application to detect the attacks and mitigate them.

Will upload bypass affect the security of my application ?
If large binary data bypass is configured with the right scope of uploading requests or URLs, then iWAF will not be able to inspect the data and the impact will be minimal.

Is it not enough to create an exclude rule for the upload parameter?
Large uploads will be cached during traffic processing. Even when a part of the request is not needed and bypassed it will be cached until other parts of that request have been inspected. Therefore, excluding only the upload parameter will not help achieve the best result.