

## Lab 2: Intro to Positive Security¶

In WAF141 we created and tested some of the negative security aspects of the Application Security Policy including Ip Intelligence Enforcement, Geolocation, Signature Based Bot Detection and a Transparent Policy focused on Attack Signatures. If you were following along in successive fashion and building your own environment, the configurations below logically pick up in the policy right where 141 left off. We will be creating a new policy for this lab that starts with “2nd day” best-practice configurations for **elevating** your WAF policy up a level but in reality you should always start your approach as recommended in the 141 class and build from there.

1. Navigate to **Security > Application Security > Security Policies > Policies List > Create** and configure a policy named **juiceshop\_blocking**
2. Choose Policy Template: **Rapid Deployment Policy**, Enforcement Mode: **Blocking** and click **Save**.

The screenshot shows the 'Create New Policy' form in the F5 Security console. The breadcrumb trail is 'Security > Application Security > Security Policies > Policies List > Create New Policy...'. The form has a 'Save' button and a 'Cancel' button. On the left is a 'Security Policy Configuration' sidebar with tabs for 'General Settings' (selected), 'Microservices', 'Attack Signatures', 'Threat Campaigns', and 'Response and Blocking Pages'. The main form area contains the following fields:

- Policy Name:** juiceshop\_blocking (Partition: Common)
- Description:** (empty text area)
- Policy Type:** Security (selected), Parent (disabled)
- Policy Template:** Rapid Deployment Policy (dropdown menu)
- Virtual Server:** None (dropdown menu)
- Application Language:** Unicode (utf-8) (dropdown menu)
- Learning and Blocking:**
  - Enforcement Mode:** Blocking (selected), Transparent (disabled)
  - Policy Building Learning Mode:** Manual (selected), Automatic (disabled), Disabled (disabled)
  - Auto-Added Signature Accuracy:** Medium (also includes signatures with high accuracy) (dropdown menu)
  - Signature Staging:** Enabled (selected), Disabled (disabled)
  - Enforcement Readiness Period:** 7 days

(../././\_images/juice\_block.png)

3. Navigate to **Security > Application Security > Policy Building > Learning and Blocking Settings**.
4. Under **Cookies** note the default settings. Uncheck **Learn** from **Modified ASM Cookie**. Generally we do not want to allow modification of the WAF cookie and therefore will disable learning suggestions for this.
5. Under **File Types** note the default Learning Mode is set to **Never (wildcard only)** Change that to be **Selective** and enable **Learn and Alarm** for **Illegal file type**.

juiceshop\_blocking Learning Mode: Manual Apply Policy

► Policy Building Settings  Note: Click Save to retain any changes you made on this screen. Blocking Settings...

► Antivirus

► Attack Signatures

► CSRF Protection

► Content Profiles

▼ Cookies

Learn New Cookies  When false positives occur, the system will add/suggest to add an explicit Cookie with relaxed settings that avoid the false positive.

Maximum Learned Cookies

☒ Learn and enforce new unmodified cookies

| <input type="checkbox"/> Learn      | <input type="checkbox"/> Alarm      | <input type="checkbox"/> Block      | Violation                   |
|-------------------------------------|-------------------------------------|-------------------------------------|-----------------------------|
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | Cookie not RFC-compliant ▼  |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Expired timestamp ▼         |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal cookie length ▼     |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | Modified ASM cookie ▼       |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Modified domain cookie(s) ▼ |

☒ Collapse many common Cookies into one wildcard Cookie after  occurrences

► Data Guard

► Evasion technique detected ▼ (0 out of 8 subviolations are enabled) ☒ Learn ☒ Alarm ☒ Block

▼ File Types

Learn New File Types  When false positives occur, the system will add/suggest to add an explicit File Type with relaxed settings that avoid the false positive.

Maximum Learned File Types

| <input type="checkbox"/> Learn      | <input type="checkbox"/> Alarm      | <input type="checkbox"/> Block | Violation                     |
|-------------------------------------|-------------------------------------|--------------------------------|-------------------------------|
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>       | Illegal POST data length ▼    |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>       | Illegal URL length ▼          |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>       | Illegal file type ▼           |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>       | Illegal query string length ▼ |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>       | Illegal request length ▼      |

(../../\_images/cookies.png)

6. Under **Headers** check the box for both Alarm and Block for **Illegal host name** and then it's **very important** to check the bottom box for **Learn New Hostnames**.

7. Under **URLS** enable **Compact Mode** for "Learn New HTTP URLs" instead of **Never (wildcard only)**. Note the description of this mode:

Choose this option if you would like to create a list of top-level URL directories (e.g. /abc/\*) and /, while enforcing all other URLs with a wildcard rule.

▼ Headers

| <input type="checkbox"/> Learn      | <input type="checkbox"/> Alarm      | <input type="checkbox"/> Block      | Violation                          |
|-------------------------------------|-------------------------------------|-------------------------------------|------------------------------------|
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | Host name mismatch ▼               |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal header length ▼            |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | Illegal host name ▼                |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal meta character in header ▼ |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | Illegal method ▼                   |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | Illegal repeated header ▼          |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | Mandatory HTTP header is missing ▼ |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | Learn Host Names                   |

Maximum Learned Host Names: 10000

HTTP protocol compliance failed ▼ (14 out of 19 subviolations are enabled) ☒ Learn ☒ Alarm ☒ Block

IP Addresses and Geolocations

Parameters

Redirection Domains

Server Technologies

Sessions and Logins

Threat Campaigns

▼ URLs

Learn New HTTP URLs: Compact ▼ Choose this option if you would like to create a list of top-level URL directories (e.g. /abc/) and /, while enforcing all other URLs with a wildcard rule.

Maximum Learned HTTP URLs: 100

Learn New WebSocket URLs: Never (wildcard only) ▼ When false positives occur the system will suggest to relax the settings of the wildcard URL.

Maximum Learned WebSocket URLs: 100

| <input type="checkbox"/> Learn      | <input type="checkbox"/> Alarm      | <input type="checkbox"/> Block      | Violation   |
|-------------------------------------|-------------------------------------|-------------------------------------|---|
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Binary content found in text only WebSocket ▼     |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | Disallowed file upload content detected in body ▼ |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | Illegal URL ▼                                     |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal WebSocket binary message length ▼         |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal WebSocket extension ▼                     |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal WebSocket frame length ▼                  |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal cross-origin request ▼                    |
| <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | Illegal entry point ▼                             |

(../../../../\_images/heads.png)

8. Enable Learn and Alarm for **Illegal URL**, click **Save** and **Apply Policy** which is back at the top right of the UI. Accept the popup and check the box for no more confirmations and then **Ok**.

Apply Policy? ✕

Are you sure you want to perform the "Apply Policy" operation on the current edited policy?

☒ Do not ask for confirmation again.  
Note: Confirmation can be changed [here](#).

OK Cancel



(../../../../\_images/pop1.png)

Allow List

Since we will be training the waf for positive security, let's create an allow list. This will help to create high fidelity learning suggestions as events occur.

1. Navigate to **Security > Application Security > IP Addresses > IP Address Exceptions** and click **Create**. Configure the allow list for a 10/8 to allow our internal "trusted" network as shown below and check the box for **Policy Builder trusted IP**.
2. Note in the upper left that this allow list is only associated with the juiceshop\_blocking policy. Allow lists are unique per policy but could be defined at part of a parent policy and delegated down to child policies.
3. Click **Create** and **Apply Policy**.

Security » Application Security : IP Addresses : IP Address Exceptions » New IP Address Exception...

juiceshop\_blocking   Learning Mode: Manual

### IP Address Exception Properties

|  |   |
|--|---|
| IP Address                             | 10.0.0.0                                    |
| Netmask                                | 255.0.0.0                                   |
| Policy Builder trusted IP              | <input checked="" type="checkbox"/> Enabled |
| Ignore in Brute Force Detection        | <input type="checkbox"/> Enabled            |
| Ignore in Learning Suggestions         | <input type="checkbox"/> Enabled            |
| Block this IP Address                  | Policy Default ▼                            |
| Never log traffic from this IP Address | <input type="checkbox"/> Enabled            |
| Ignore IP Intelligence                 | <input type="checkbox"/> Enabled            |
| Description                            |   |

Cancel Create

(../../\_images/list.png)

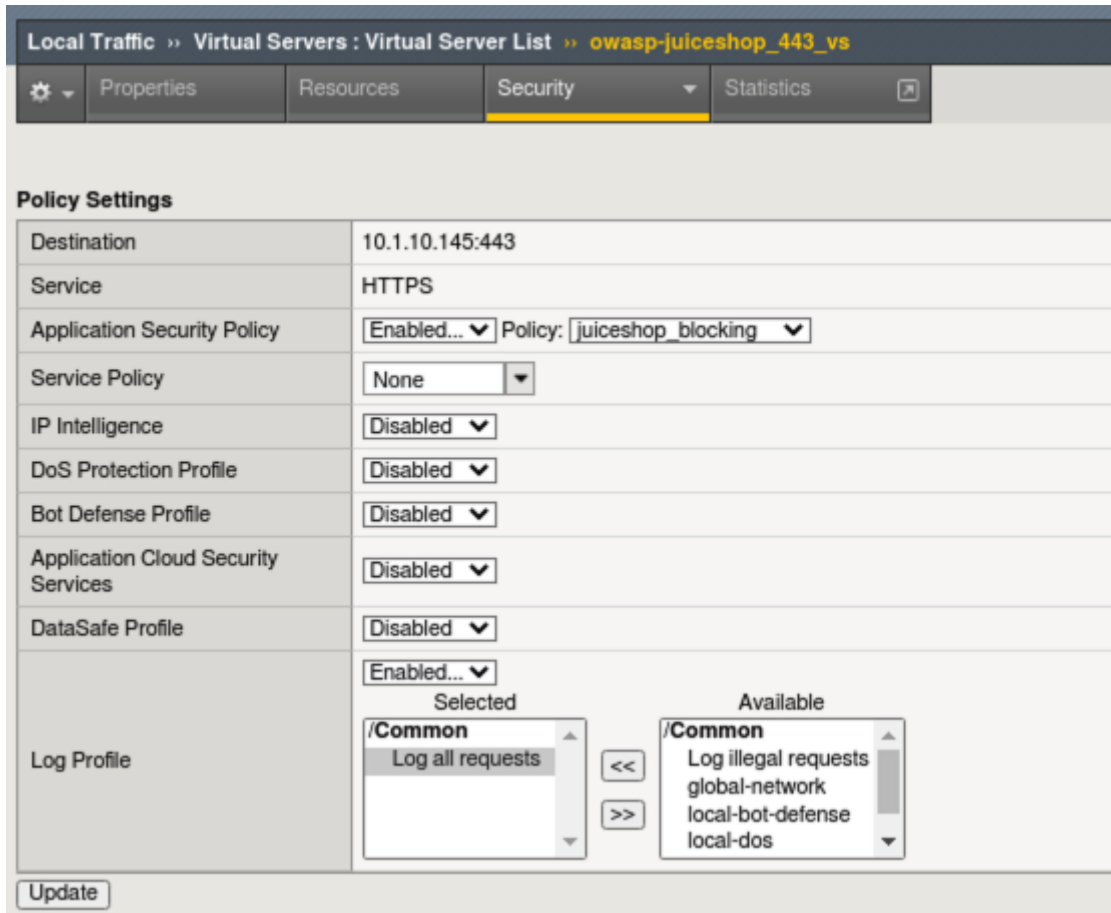
Configure the Virtual Server¶

1. Navigate to **Local Traffic > Virtual Servers > owasp-juiceshop\_443\_vs > Security > Policies**

#### Note

For Demo purposes, we will be removing the Bot Defense Profile to exclusively test the blocking Application Security Policy. In the “real world” these profiles complement each other by providing a layered defense for your application.

2. **Enable** the Application Security Policy **juiceshop-blocking**.
3. **Disable** the Bot Defense profile.
4. **Move** the Balanced\_Bot\_Log from **Selected** to **Available** and move the **Log all requests** profile over to **Selected** and click **Update**.



(../../\_images/virt.png)

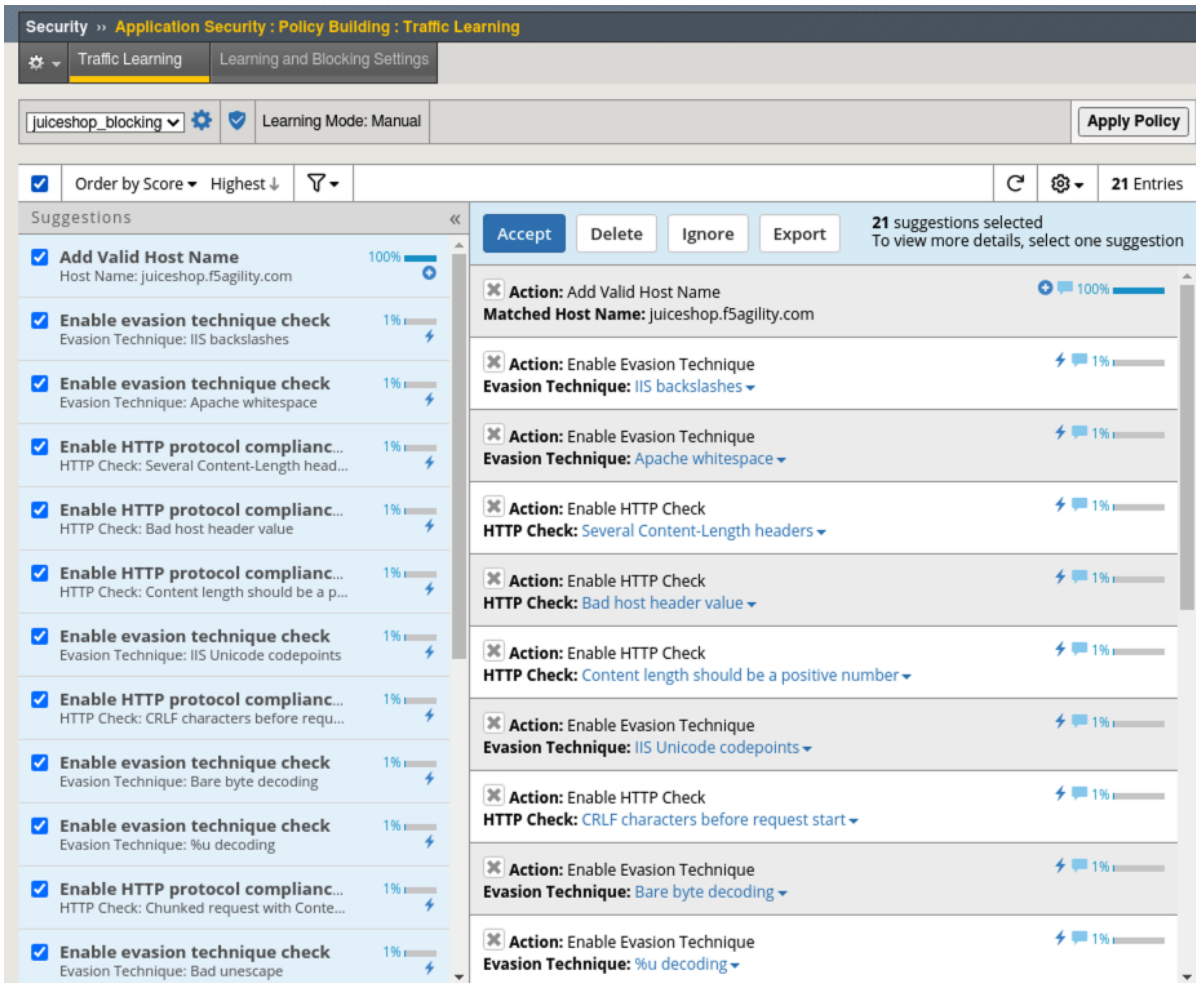
## Exercise the App Part 1¶

1. Close any existing Juice Shop tabs in the browser and open a new one to the **OWASP Juice Shop** bookmark.
2. **Request Rejected!!!** What Happened?

## Investigating an Incident¶

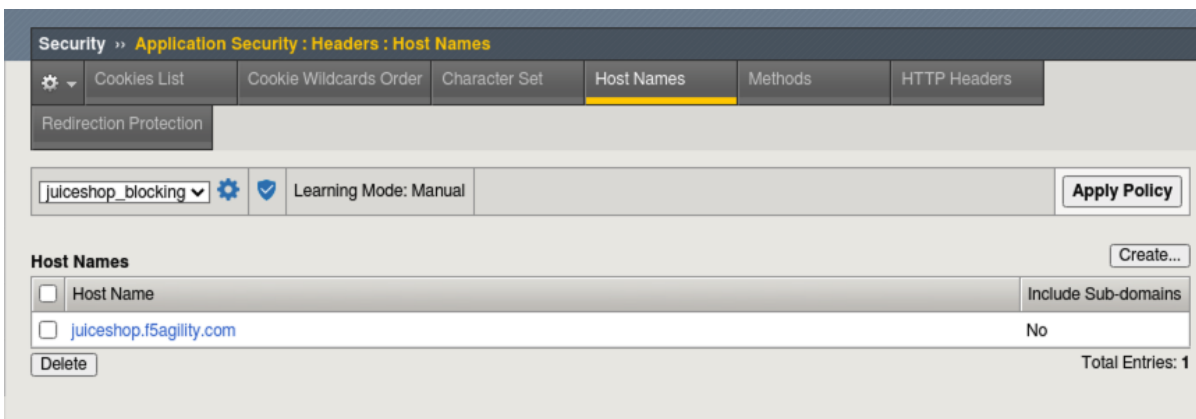
1. Click back on the **Advanced WAF** tab and navigate to **Security > Event Logs > Application > Requests** and review the blocked events. In both cases it was an illegal hostname due to the checks that we enabled under **Headers** in **Learning and Blocking Settings** just moments ago.
2. The problem is that we enabled checking for a hostname but haven't defined what that hostname is yet. This exercise is to draw your attention to the importance of understanding what you are enabling in Learning and Blocking Settings and how to quickly resolve an issue. We can easily add the hostname.
3. Navigate to **Security > Application Security > Policy Building > Traffic Learning** and note the learning suggestions and score. You will see suggestions to add the top level URL and a Valid Hostname. All of the others involve enabling various checks for evasion techniques and http protocol compliancy which are generally a good idea to enable.

4. Click the box to **Select All** suggestions and click **Accept > Accept suggestions** and **Apply Policy**.



(../..../\_images/learn.png)

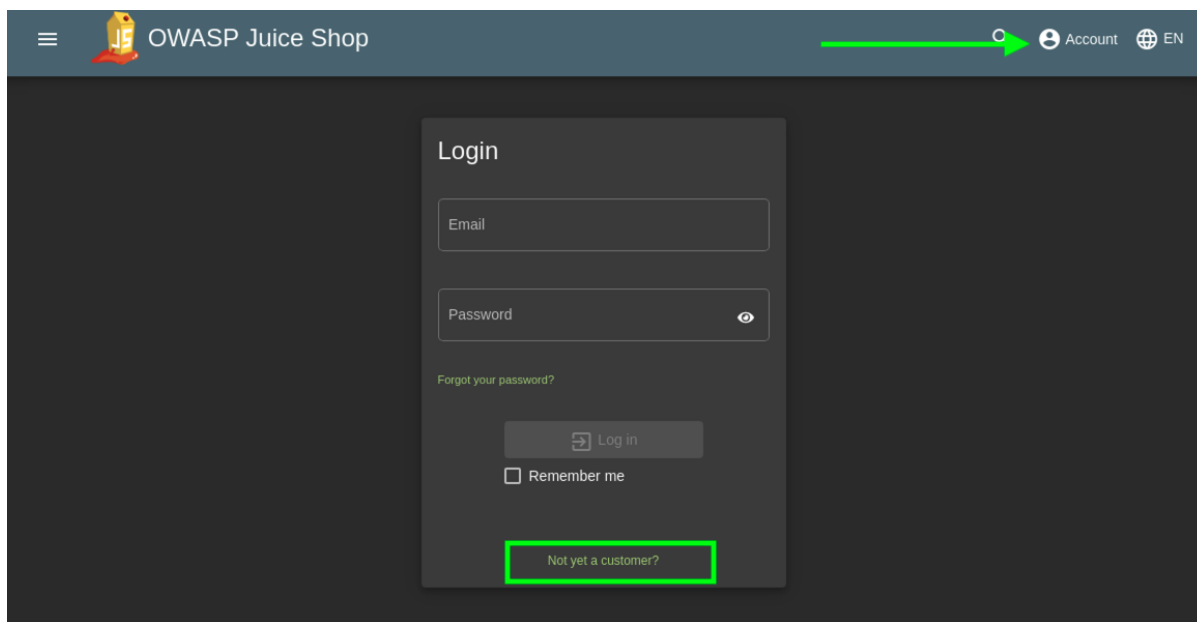
5. Navigate to **Security > Application Security > Headers > Host Names** to review the hostname that was configured when you accepted the learning suggestion.



(../..../\_images/host.png)

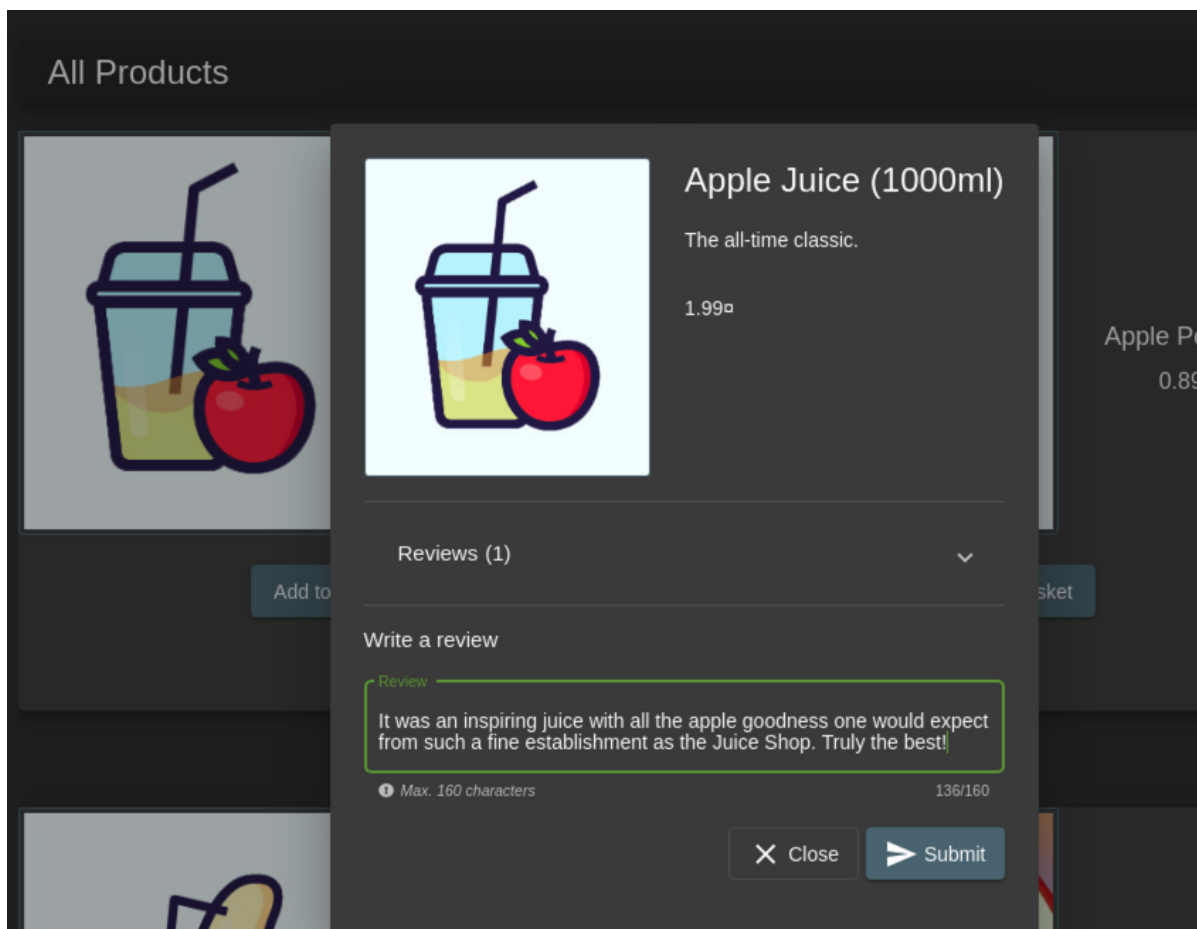
Exercise the App Part 2¶

1. Go back to the Juice Shop tab and do a [Shft + Refresh]
2. If you are continuing with the same deployment from the 141 class, skip to step 4, else click **Account > Login** in the top right and then click **Not yet a customer** on the login form.



(../..../\_images/account.png)

3. Use **f5student@f5agility.com** for email address and the same password you've been using for the labs. Select and complete any of the Security Questions and click **Register**.
4. Login with the new account, click on the **Apple Juice** and leave a short review and click **Submit**.



(../..../\_images/feedback.png)

5. In the Advanced WAF tab navigate to **Security > Event Logs > Application > Requests** and you will see a blocked event for the review you just left.
6. Click on the blocked event and review the Violation. This is an **Illegal method** violation due to "Put" being used as the command to leave feedback. "Put" is not a default allowed



HTTP command per the Rapid Deployment Policy Template.

Security » Event Logs : Application : Requests

Application Protocol Network DoS Bot Defense Logging Profiles

Order by Date Newest 1 - 100 of 122 Entries 1 2

Requests

[HTTPS] /rest/products/1/reviews 10.1.10.100 10:49:17 2021-02-24 N/A

[HTTPS] /socket.io/ 10.1.10.100 10:49:13 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:48:48 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:48:48 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:48:23 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:48:23 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:47:58 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:47:58 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:47:33 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:47:33 2021-02-24 200

[HTTPS] /socket.io/ 10.1.10.100 10:47:08 2021-02-24 200

[HTTPS] /rest/products/1/reviews

Triggered Violations 1

Violation Occurrences Suggestions

Illegal method 1 View...

Request Details Basic All Details

Geolocation N/A

Source IP Address 10.1.10.100:47300

Session ID 131ad9585075f514

Microservice N/A

Time 2021-02-24 10:49:17

Enforcement Action Block

Enforced By Application Security Policy

Violation Rating 3 Request needs further examination

Attack Types Information Leakage

Request

Request actual size: 3709 bytes.

PUT /rest/products/1/reviews HTTP/1.1

Host: juiceshop.f5agility.com

Connection: keep-alive

Content-Length: 185

Accept: application/json, text/plain, \*/\*

Authorization: \*\*\*\*\*

(../../\_images/badreview.png)

7. Click the **Accept** button which will add "Put" to the **Allowed Methods** in **Security > Application Security > Headers > Methods**

8. Navigate to **Security > Application Security > Headers > Methods** to review the addition and click **Apply Policy**.

Apply Policy Configuration

Operation completed successfully.

Security » Application Security : Headers : Methods

Cookies List Cookie Wildcards Order Character Set Host Names Methods HTTP Headers Redirection Protection

juiceshop\_blocking Learning Mode: Manual Apply Policy

Allowed Methods Create...

| Method Name | Act As Method |
|-------------|---------------|
| PUT         | GET           |
| HEAD        | GET           |
| POST        | POST          |
| GET         | GET           |

Delete

(../../\_images/put1.png)

9. Go back to Juice Shop and test leaving a review again. From the left hamburger menu start a support chat and test leaving a complaint.

10. Navigate to **Security > Event Logs > Application > Requests** and you should see all **Allowed Requests** at this point. If you, by rare chance, see a blocked request, take steps similar to the previous to resolve the issue by **Accepting** the blocked request.



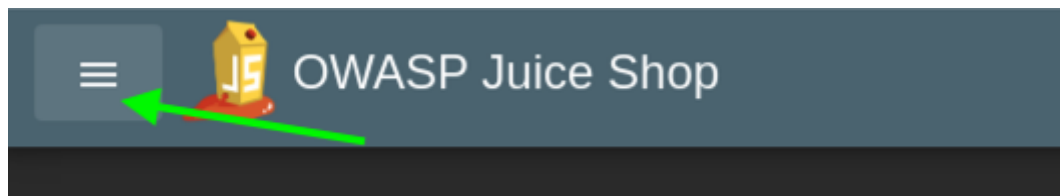
(../../../../\_images/allowed.png)

**This is how to train a waf, ferret out any false positives and why it is critical to get your policies developed from trusted sources during the testing phases of application development.**

## Enforcing File Types¶

File types are low-hanging fruit from a positive security perspective and a great starting point for enhancing your security policy by allowing or disallowing access to known file types or extensions. We will enable Compact mode learning to create a learned list of common file types and enforce against a pre-populated list of disallowed file types.

1. Navigate to **Security > Application Security > Policy Building > Learning and Blocking Settings > File Types** and change the default learning mode from **Selective** to **Compact** and read the description.
2. Click **Save** and **Apply Policy**.
3. Go back to the Juice Shop tab and browse to the **Photo Wall** via the "Hamburger Menu" at the top left.



(../../../../\_images/ham.png)

4. In Advanced WAF go to **Security > Application Security > Policy Building > Traffic Learning** and notice the new file type learning suggestions.
5. Review and then select all of the new suggestions and click **Accept > Accept Suggestions** and **Apply Policy**. There may be additional suggestions that you can safely accept.

Security » Application Security : Policy Building : Traffic Learning

Traffic Learning Learning and Blocking Settings

juiceshop\_blocking Learning Mode: Manual Apply Policy

Order by Score Highest 4 Entries

Suggestions

| Suggestion  | Score | Action                      | Matched File Type   | Matched Wildcard |
|---|-------|-----------------------------|---|------------------|
| <input checked="" type="checkbox"/> Add File Type<br>File Type: no_ext          | 100%  | Accept Delete Ignore Export | Action: Add File Type<br>Matched File Type: no_ext        | 100% *           |
| <input checked="" type="checkbox"/> Add File Type<br>File Type: png             | 100%  |                             | Action: Add File Type<br>Matched File Type: png           | 100% *           |
| <input checked="" type="checkbox"/> Add File Type<br>File Type: jpg             | 100%  |                             | Action: Add File Type<br>Matched File Type: jpg           | 100% *           |
| <input checked="" type="checkbox"/> Classify URL Content<br>HTTP URL: [HTTPS] / | 1%    |                             | Action: End Classification<br>Matched HTTP URL: [HTTPS] / | 1%               |

(../../../../\_images/fileaccept.png)

6. Navigate to **Security > Application Security > File Types > Allowed File Types** and review what was added. Click on the **Disallowed File Types** tab at the top of the GUI and review the default disallowed files for this policy.

Security » Application Security : File Types : Allowed File Types

Allowed File Types Disallowed File Types Wildcards Order

juiceshop\_blocking Learning Mode: Manual Apply Policy

Allowed File Types List

File Type All Enforcement Readiness All Total Entries: 5

Legend: Waiting for additional traffic samples Learning suggestions available Ready to be enforced Create...

| Type   | URL Length | Request Length | Query String Length | POST Data Length | Staging |
|--------|------------|----------------|---------------------|------------------|---------|
| *      | Any        | Any            | Any                 | Any              | No      |
| jpeg   | Any        | Any            | Any                 | Any              | No      |
| jpg    | Any        | Any            | Any                 | Any              | No      |
| no_ext | Any        | Any            | Any                 | Any              | No      |
| png    | Any        | Any            | Any                 | Any              | No      |

Enforce Delete Total Entries: 5

(../../../../\_images/files.png)

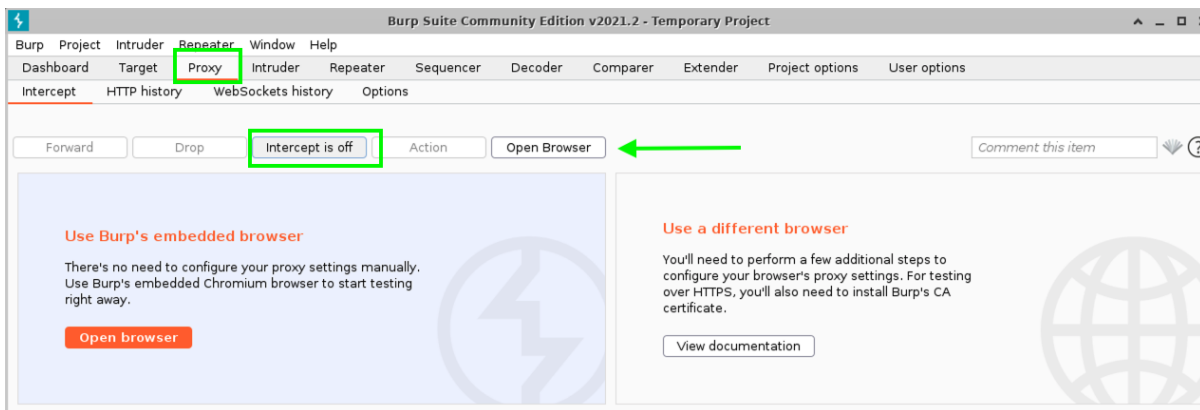
Testing WAF Policy

1. On the Linux Client desktop launch Burp Suite Community Edition. **DO NOT click multiple times. It takes a few moments to load on first launch.**



(../../\_images/burp1.png)

2. Take the default setting of **Temporary project** by clicking **Next** and then click **Start Burp** with the default settings. **Do not accept any offers to update/restart the software.**
3. Click on the **Proxy** tab and click on the "blue" **Intercept is on** button to turn it off.
4. Click the **Open Browser** button and **wait for several moments** for the built-in Burp Browser to open. Your setup should look like this:

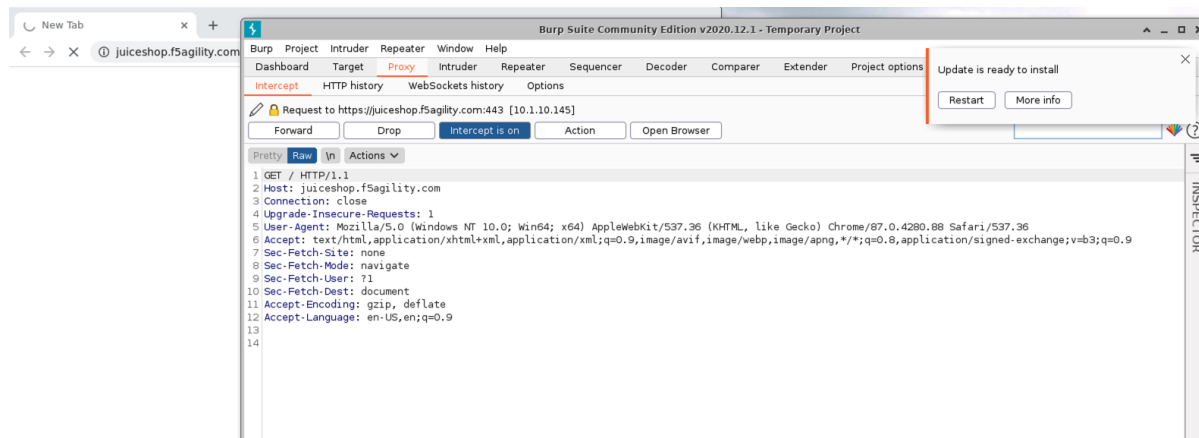


(../../\_images/browser.png)

5. In the Burp browser paste in: `https://juiceshop.f5agility.com/` but do **NOT** hit the Return key yet.
6. Back in Burp Console click on **Intercept is off** button to turn it back on.
7. In Burp browser click in the whitespace of the URL bar twice so the URL is NOT highlighted and hit the **Return** key on your keyboard to send the request for `https://juiceshop.f5agility.com/`
8. You will notice Burp Console will popover the browser with the intercepted request. You can now decide which actions to take real-time before sending the requests.

### **i Important**

You may see some requests mixed in that are Google related (`www.gstatic.com`, `googleapis.com` etc). These are produced automatically by the browser and you can safely forward them until you get to the request for `https://juiceshop.f5agility.com` (`https://juiceshop.f5agility.com`).



(../../../../images/burpjuice.png)

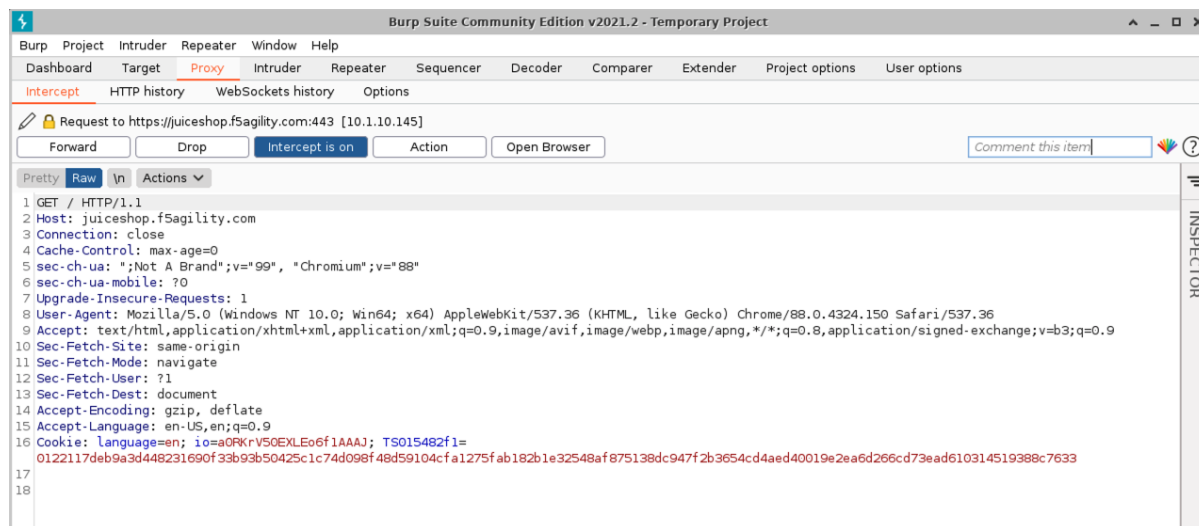
9. Go ahead and choose to **Forward** that request.

10. As you can see Burp is a very powerful proxy that allows you to view each request as it is being made and potentially insert or modify that request before sending.

11. Click on **Intercept is on** to turn it off so the rest of the requests load and then click it again to turn it back on so that it reads **Intercept is on**.

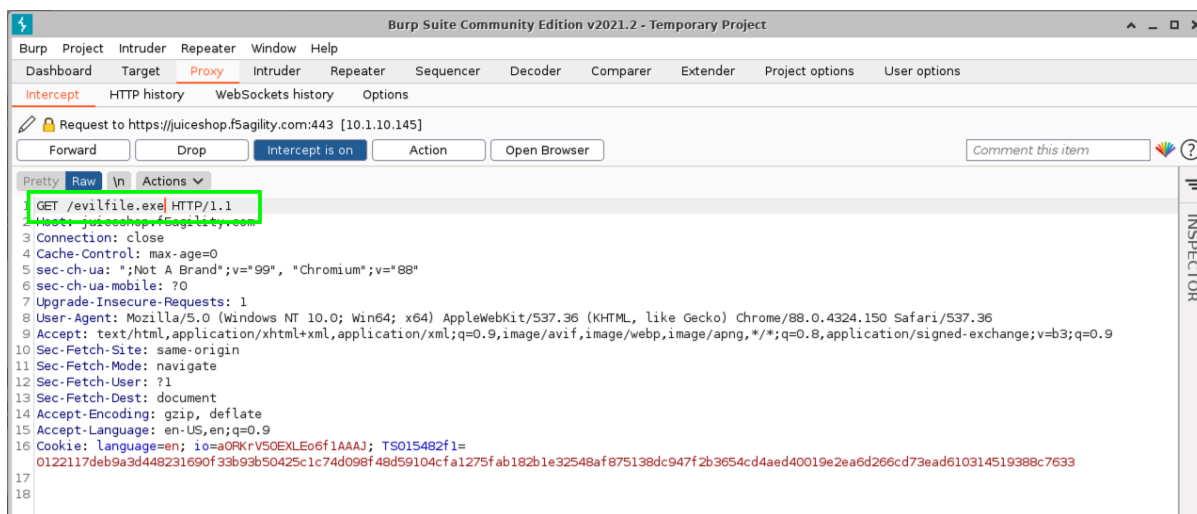
12. Back in the Burp Browser the page should have loaded from the previous requests so just click the **Refresh** button in the browser and **Dismiss** the popup.

13. You can use any of the **GET** requests for **Host: juiceshop.f5agility.com**. Simply forward any of the aforementioned Google related requests should they pop up.



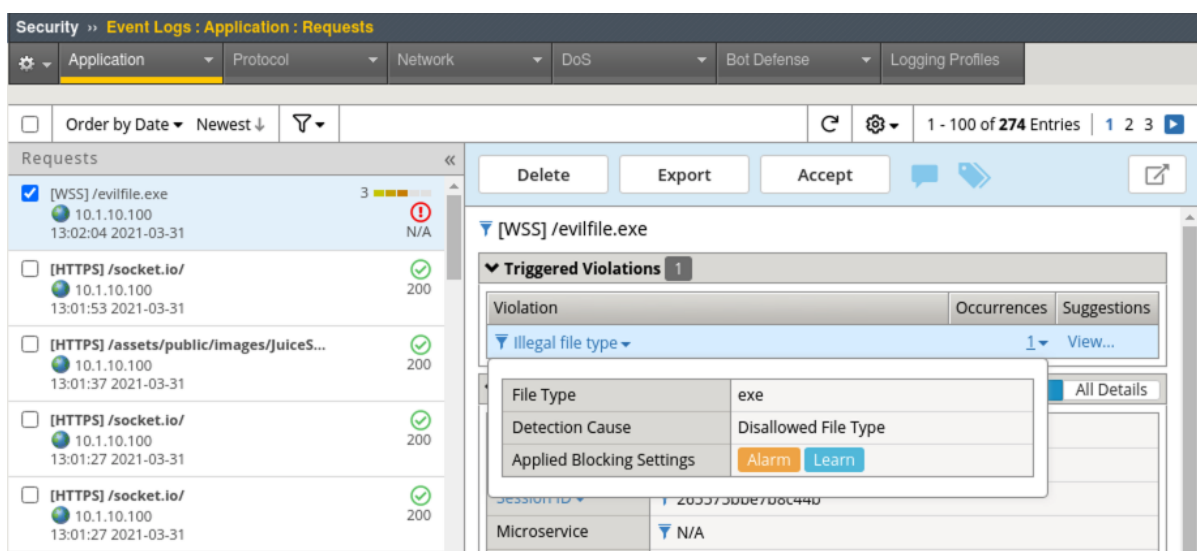
(../../../../images/defaultpage.png)

14. Modify the request to fetch an evilfile.exe file from the server and click **Forward**.



(../../../../\_images/evilfile.png)

15. In Advanced WAF, navigate to **Security > Event Logs > Application > Requests** and review the alert. Was it blocked?



(../../../../\_images/evilalert.png)

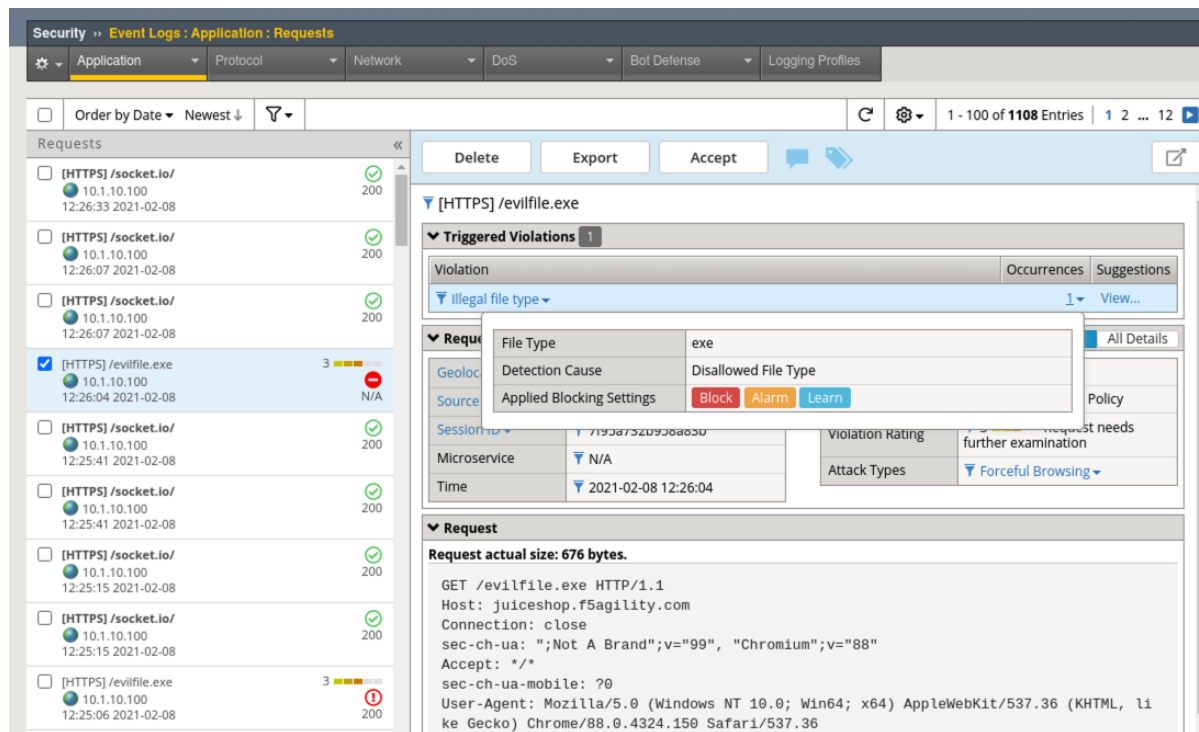
## Note

Even though policy is in blocking mode, individual elements can be very granularly configured to Alarm or Block. In practice you could have a Blocking policy with everything in set in Learning and Blocking settings to only "Alarm". You can then methodically enable blocking for each individual element and validate the application. This gives you the utmost flexibility when moving from a Transparent to Blocking policy.

16. Navigate to **Security > Application Security > Policy Building > Learning and Blocking Settings > File Types** and enable **Block** for **Illegal file type**.
17. Click **Save** and **Apply Policy**.
17. Back in Burp Console modify another request for **evilfile.exe** again and click **Forward**. Just delete whatever URI is currently being called and replace it with **evilfile.exe** **Make sure the Host is set to juiceshop.f5agility.com and not a google site. If it is related to**

google just click forward until you get to the next juiceshop.f5agility.com "Host" request.

18. Navigate to **Security > Event Logs > Application > Requests** and review the alert. Was it blocked this time?



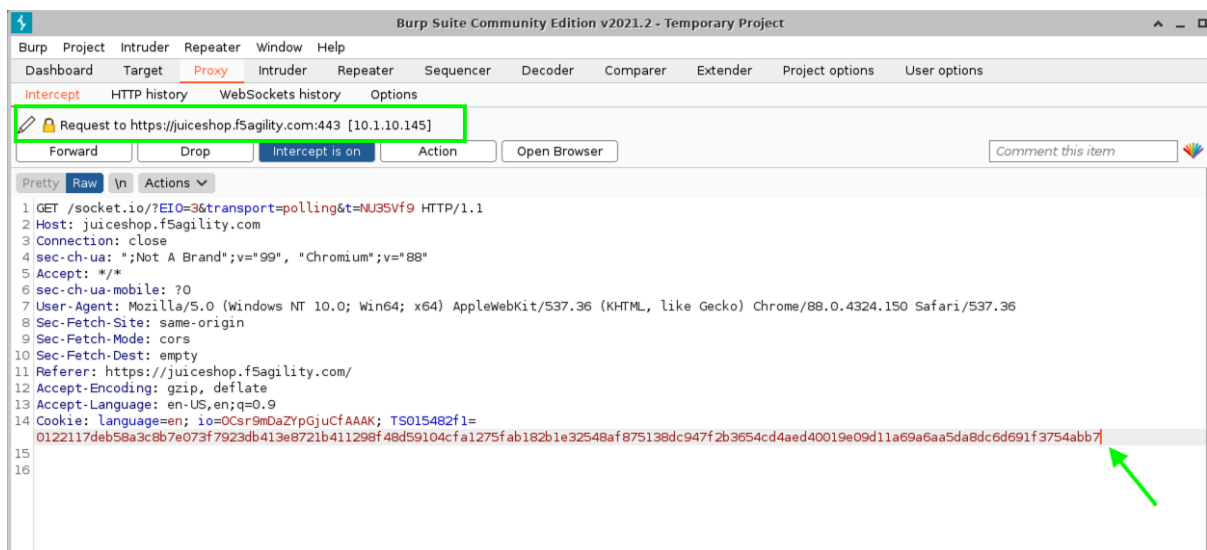
(../\_images/evilblock.png)

Monkeying with the ASM Cookie

1. Back in Burp Console make sure you are looking at a request for Host **juiceshop.f5agility.com**. If not, click forward until you find one.
2. Under **Cookie** notice at least 3 cookies.

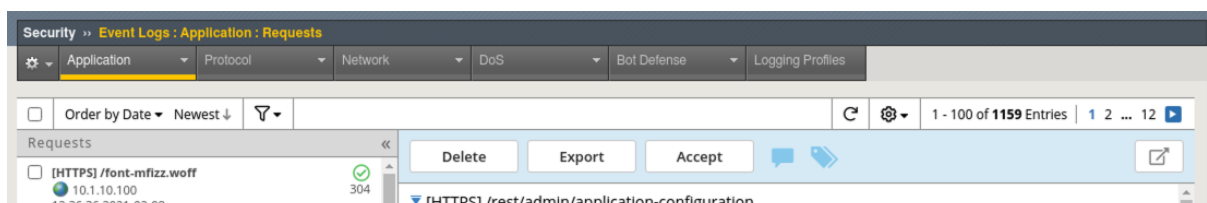
- language (used to set language pref in the browser)
- io (Juice Shop session cookie)
- TS (Set by Advanced WAF and will always be a unique identifier)

3. Add an extra character (7) to the end of the TS cookie value and click **Forward**.



(../../../../\_images/modified.png)

4. Navigate to **Security > Event Logs > Application > Requests** and review the alert.



(../../../../\_images/modified1.png)

5. Back in Burp Console make sure you are looking at a request for Host **juiceshop.f5agility.com**. If not, click forward until you find one.

6. Change the host to the ip address of the Virtual Server: **10.1.10.145** and click **Forward**.





(../../../\_images/iphost.png)

7. Close Burp

8. Back in Advanced WAF, refresh **Security > Event Logs > Application > Requests** and review the alert. What was the violation? How could you add it to the allowed hostnames if required?

**This concludes Lab 2**

**This lab was designed to give you the tools and strategies for building and managing a more complex or "Day 2" WAF policy. You now know how to turn on and test some positive security features that will greatly elevate your application security posture.**