**URL Filtering on**

**Palo Alto PA-220**

Ryan Chen

12/16/2024

**Purpose:**

The primary purpose of this lab is to configure URL Filtering for end-devices on a generic Palo Alto device. In this case, we are using the PA-220 firewall. This configuration involves creating a URL Filtering object, changing appropriate fields to block or unblock, and allowing end-users to override filtering with a password. To do so, root SSL certificates must also be installed on end-devices in order to ensure HTTPS websites can be intercepted to display the block page.

**Background Information:**

URL Filtering is a function on most firewalls (Palo Alto, Fortinet, Juniper, etc.) that allows administrators to configure certain websites to either allow, block, or require a log-in/override page to access. This kind of functionality is desirable for many reasons, in all kinds of environments.

For instance, administrators of work environments may need URL Filtering to block certain game, adult, or phishing websites on employee end-devices to increase both productivity and security. By restricting employee access to potentially dangerous websites, security is hardened through risk avoidance. Alternatively, a school administrator would be likely to block the same categories, including other additional ones like AI websites and social media sites.

Palo Alto implements URL Filtering through their custom block lists. All sites are added by Palo Alto internally into their own categories, and administrators can also create or modify their own lists. Predefined categories can be used with a URL Filtering license, and are listed [here](https://docs.paloaltonetworks.com/advanced-url-filtering/administration/url-filtering-basics/url-categories). Furthermore, Palo Alto provides a website where URLs can be checked for their categorization on their servers, which can be found [here](https://urlfiltering.paloaltonetworks.com/). This is useful for testing sites prior to categorization to ensure correct filtering behavior.

In order to configure URL Filtering on Palo Alto devices, an object must first be created to define the filtering profile. With a valid filtering license, Palo Alto’s predefined categories can be configured to either allow, block, or override. Overriding is a feature where end-users can bypass URL filters temporarily by entering credentials. This is especially useful for cases where end-users may temporarily need to access a blocked website, such as AI. Timers are generally attached to these override functions, granting permissions to the site for a predetermined amount of time. After creating the profile, it can be applied in the policies menu, allowing filtering to be enabled.

Notably, one of the biggest issues with URL Filtering is the handling of HTTP vs HTTPS websites. Hypertext Transfer Protocol, or HTTP is a protocol that handles the core foundation of loading webpages. HTTP requests fetch data from a website, which is then sent to client machines and allows a website to be loaded. The biggest difference between HTTP and HTTPS is security and encryption. HTTP Secure, or HTTPS has been widely adopted ever since the late 90’s, as computer security has become more important. Because of HTTP’s unencrypted nature, numerous situations may arise that allow attackers to intercept data, leaking important data. On the other hand, HTTPS utilizes SSL, or more recently TLS to encrypt requests and responses. TLS implements public and private key cryptography to establish a secure connection and send encrypted data.

The nature of block/filter pages requires the firewall to intercept the request and insert the block page in the place of the response. With HTTP, this is fairly easy to do, however HTTPS renders this impossible (by design) without necessary certificates. SSL/TLS certificates store the previously mentioned public keys for each website, along with other necessary information to establish a connection. Certificates are typically issued by certificate authorities, companies that generate certificates for people hosting websites. Then, these companies root certificates (the base, most trusted certificate) are pre-installed to popular operating systems like Windows, Mac, and Linux. Root certificates are trusted by default and used to verify the authority of other certificates for individual websites. Intercepting the request and inserting data using the firewall is impossible because we would need to decrypt and verify the data using certificates that we do not have access to. As a result, we need to provide our own self-signed root certificate, making our firewall trusted by end-devices. This will allow our connection to go through and show the block page, but it will unfortunately still show that our connection is “Insecure”, due to it being self-signed.

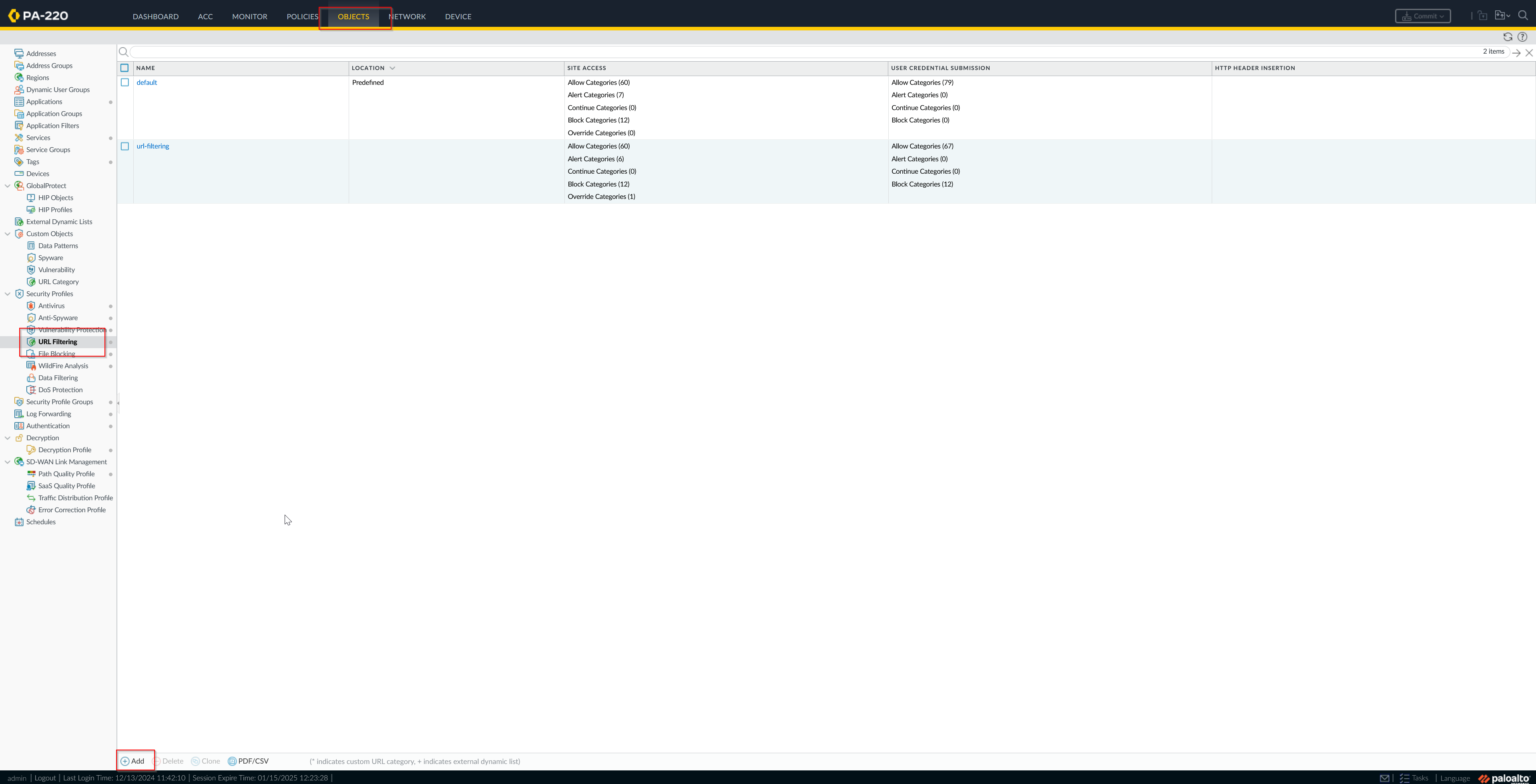
**Lab Summary:**

This lab involves a few separate steps. First, we must create the security object for URL Filtering, which we will later direct our firewall to use. By setting certain predefined Palo Alto categories to allow/block/override, we can grant permissions associated to certain websites. After the object is created, we can use it by setting it to apply to all outbound traffic. This will allow us to filter HTTP traffic.

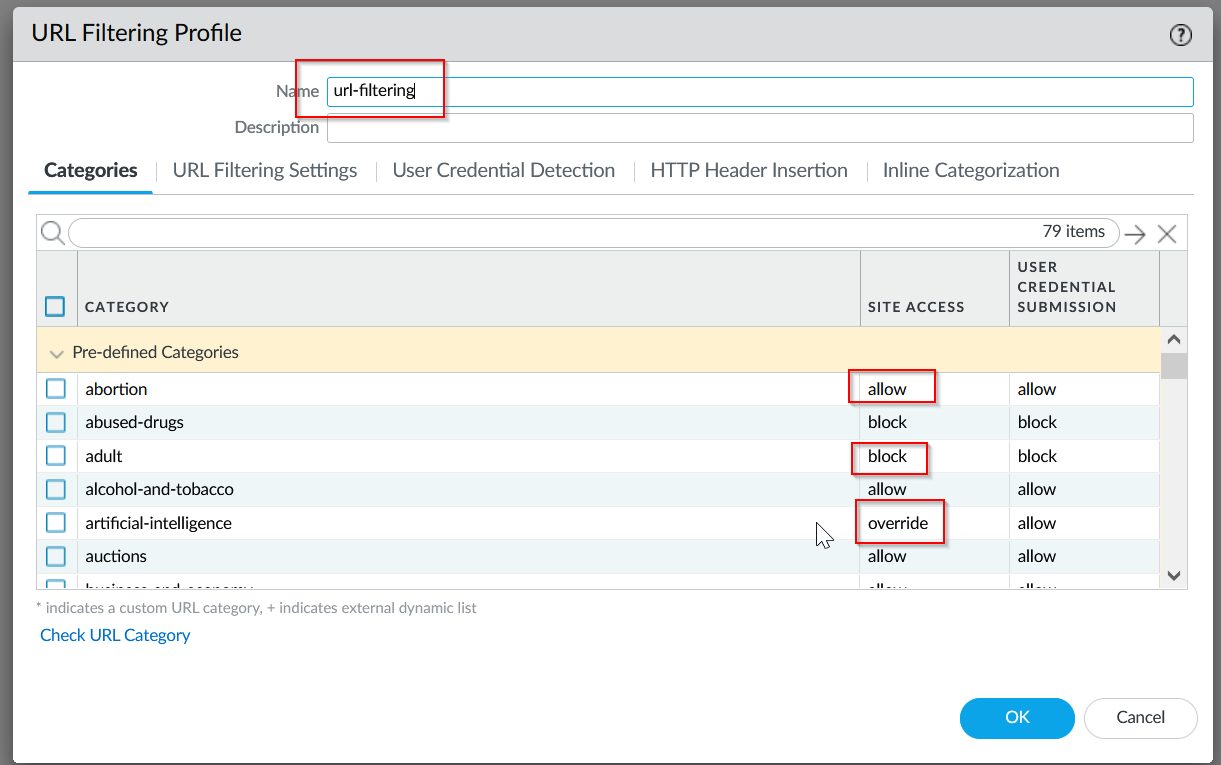
In order to filter HTTPS traffic and enable override pages, we must configure SSL/TLS decryption by generating our own root certificate and installing it into end-device machines. This allows end-devices to trust our firewall, giving us permission to intercept the traffic and instead replace it with the block/override page. Finally, override pages can be configured with specific passwords that end-users can input if they want to access the webpage.

**Lab Commands:**

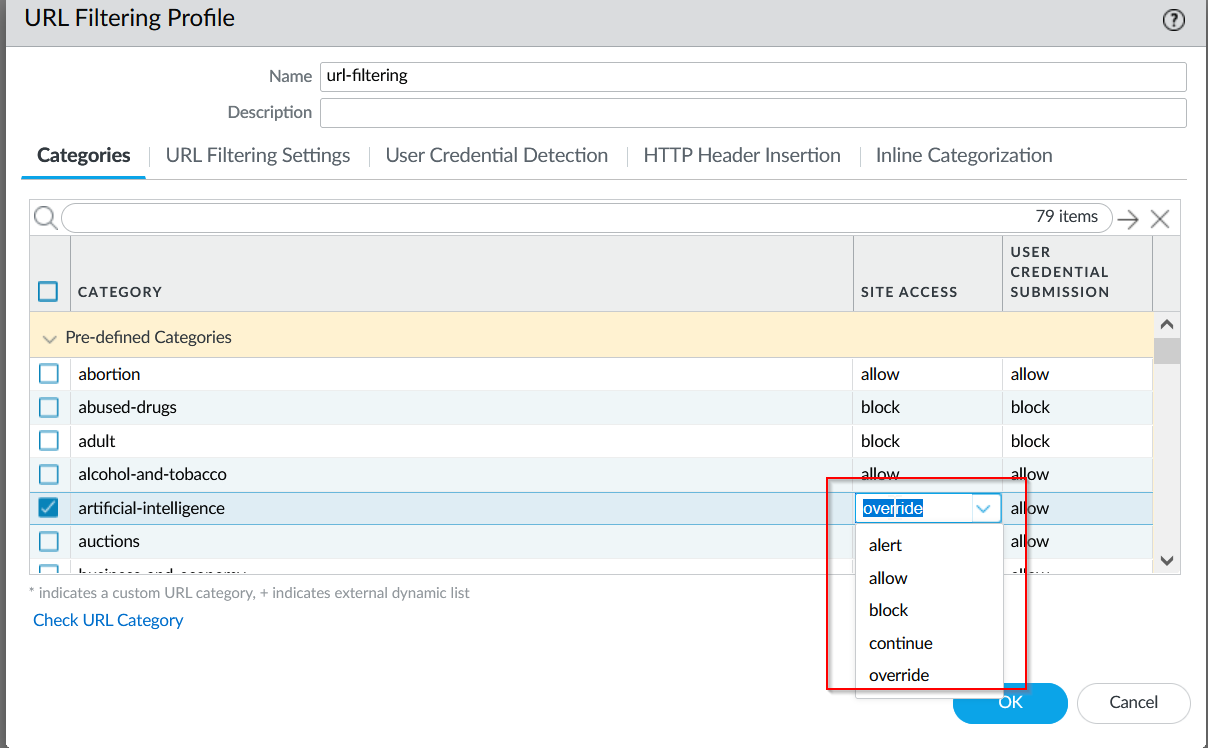
1. Navigate to Objects > Security Profiles > URL Filtering and press Add to create a new URL Filtering profile.



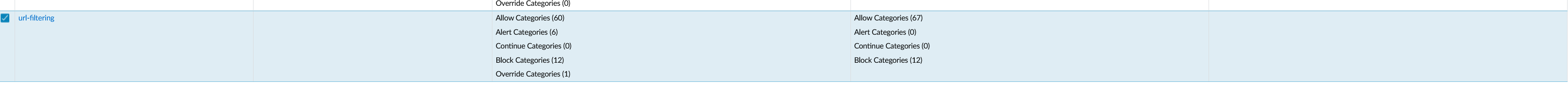
1. Here, you can name the profile and change the responses to certain predefined categories provided by Palo Alto.



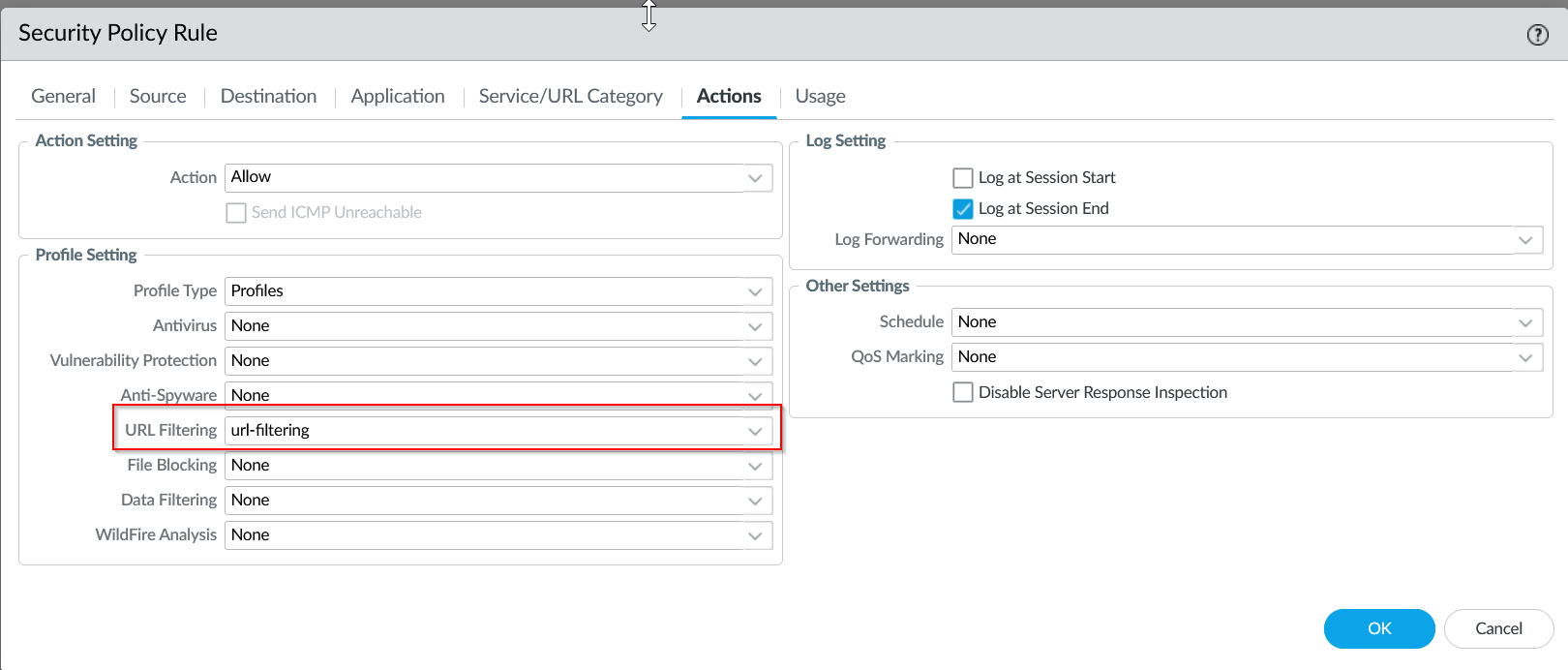
1. For later configuration, to make a category bypass-able with passwords, you should configure it as an “override” site access type.



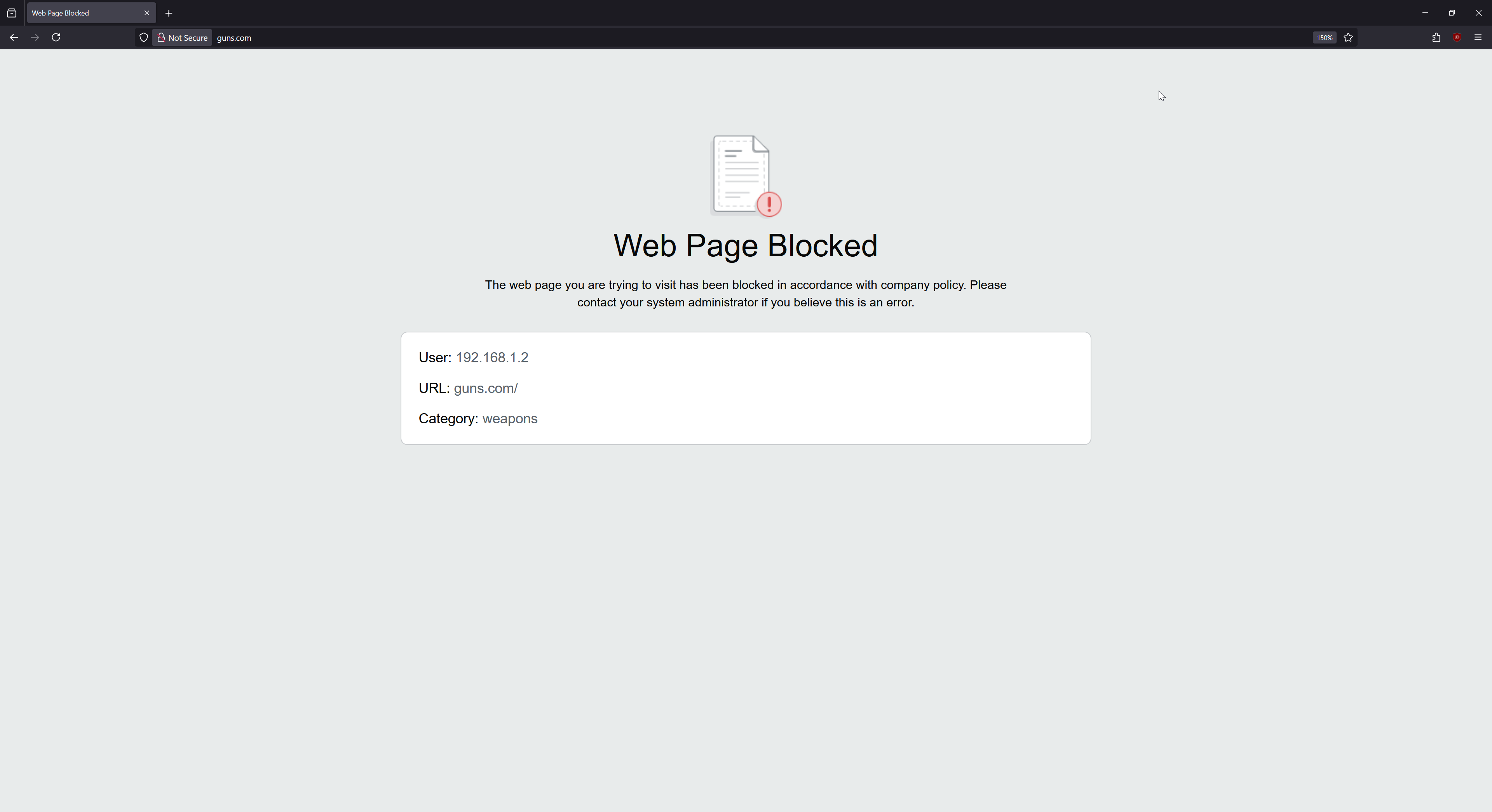
1. Keep the other settings the same and create the profile by pressing “Ok”. It should show up in the list now!



1. Apply the policy by going to Policies > Security now. We will apply the profile to the “outgoing” security policy to filter out websites. Go to the “Actions” section and select the profile you just created.



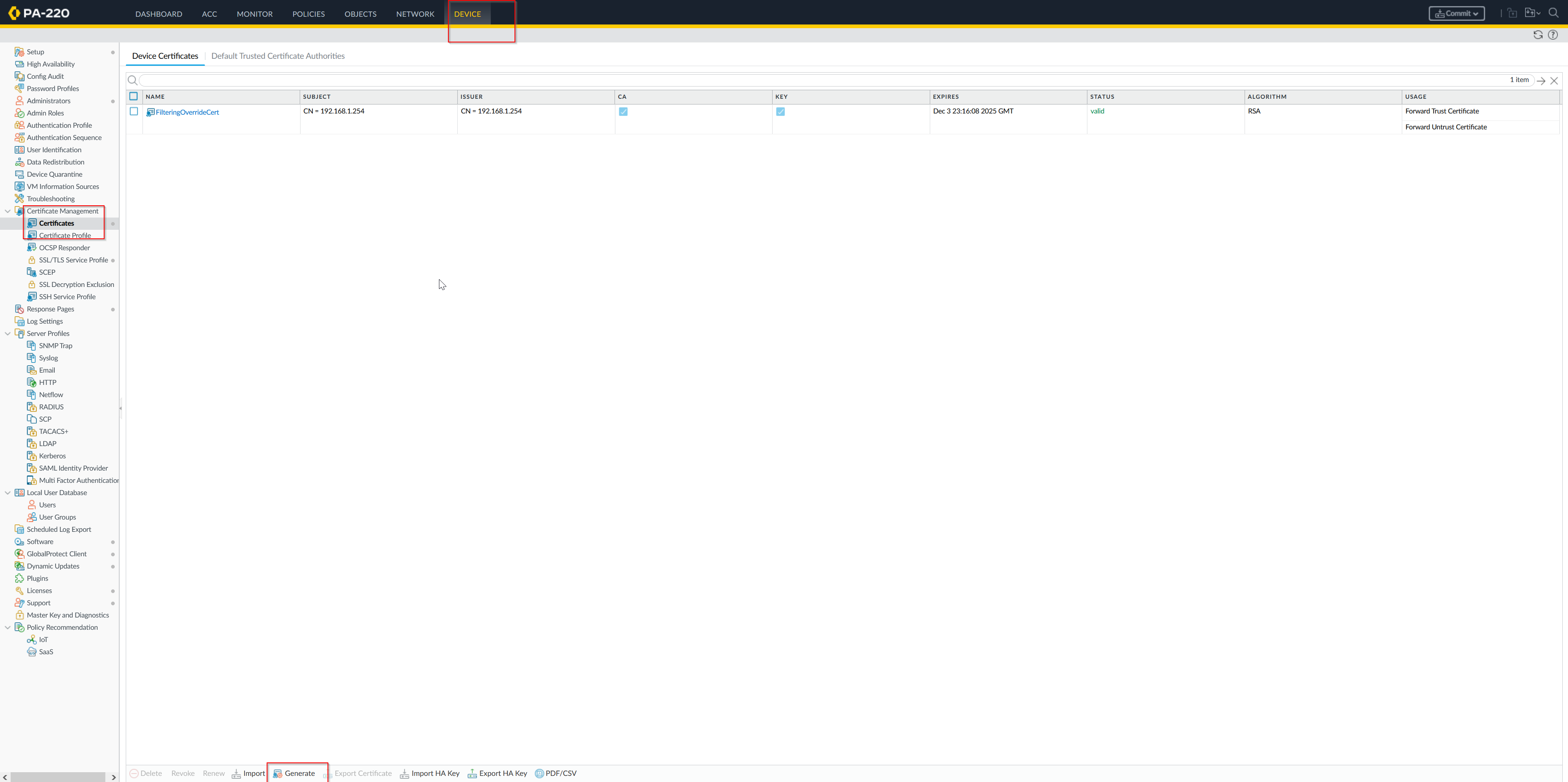
1. Press “Ok” and commit changes. Now, the URL Filtering should work for HTTP websites ONLY. Check that it works like a website like <http://guns.com>. Note that only some websites allow HTTP traffic, others will automatically redirect to HTTPS, so this test will only work on specific websites.



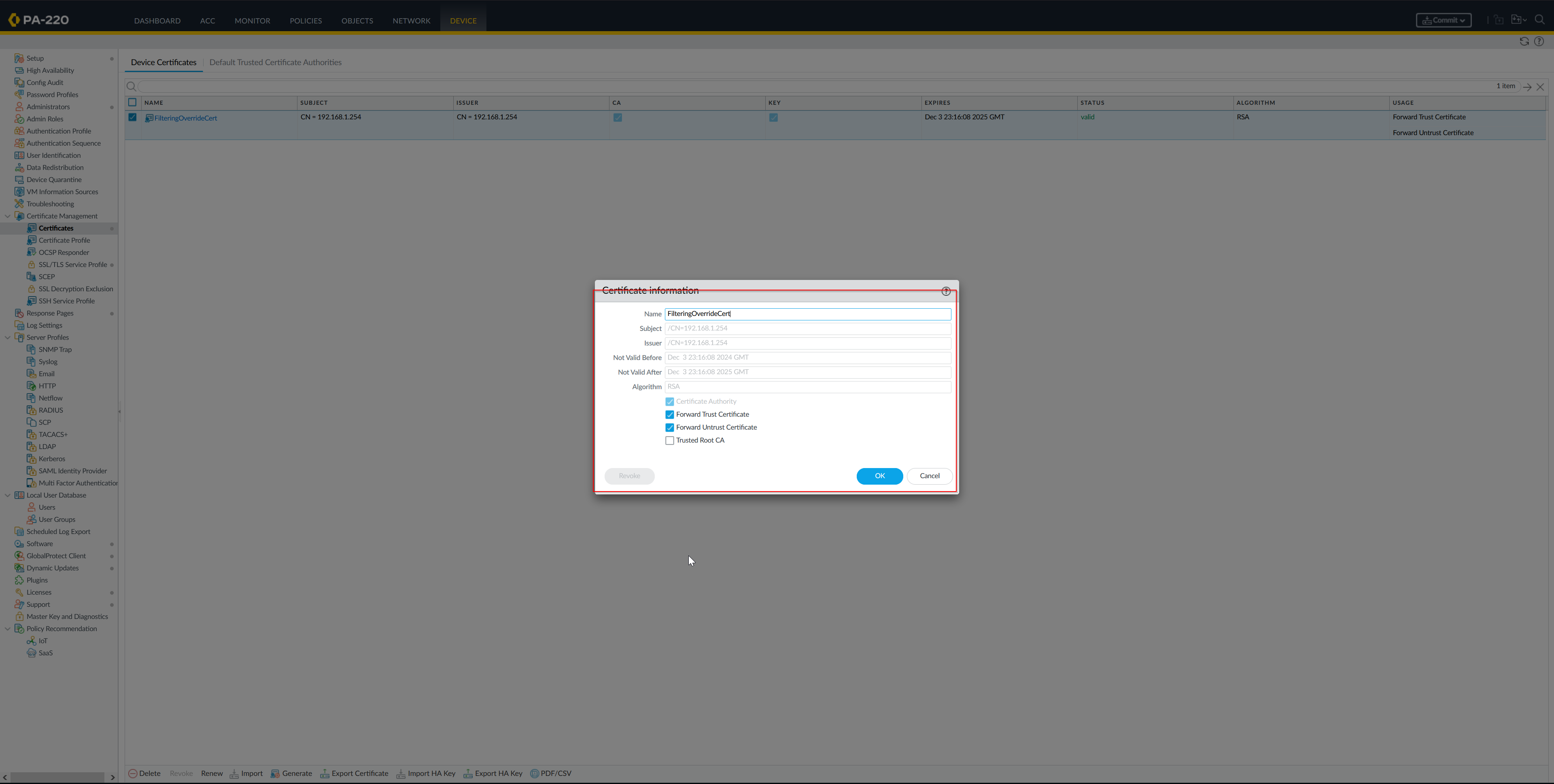
1. Now, we should configure certificates in order to enable HTTPS filtering to work. Console into your firewall and enter this command to enable the firewall to decrypt traffic to insert the block page.

*# set deviceconfig setting ssl-decrypt url-proxy yes*

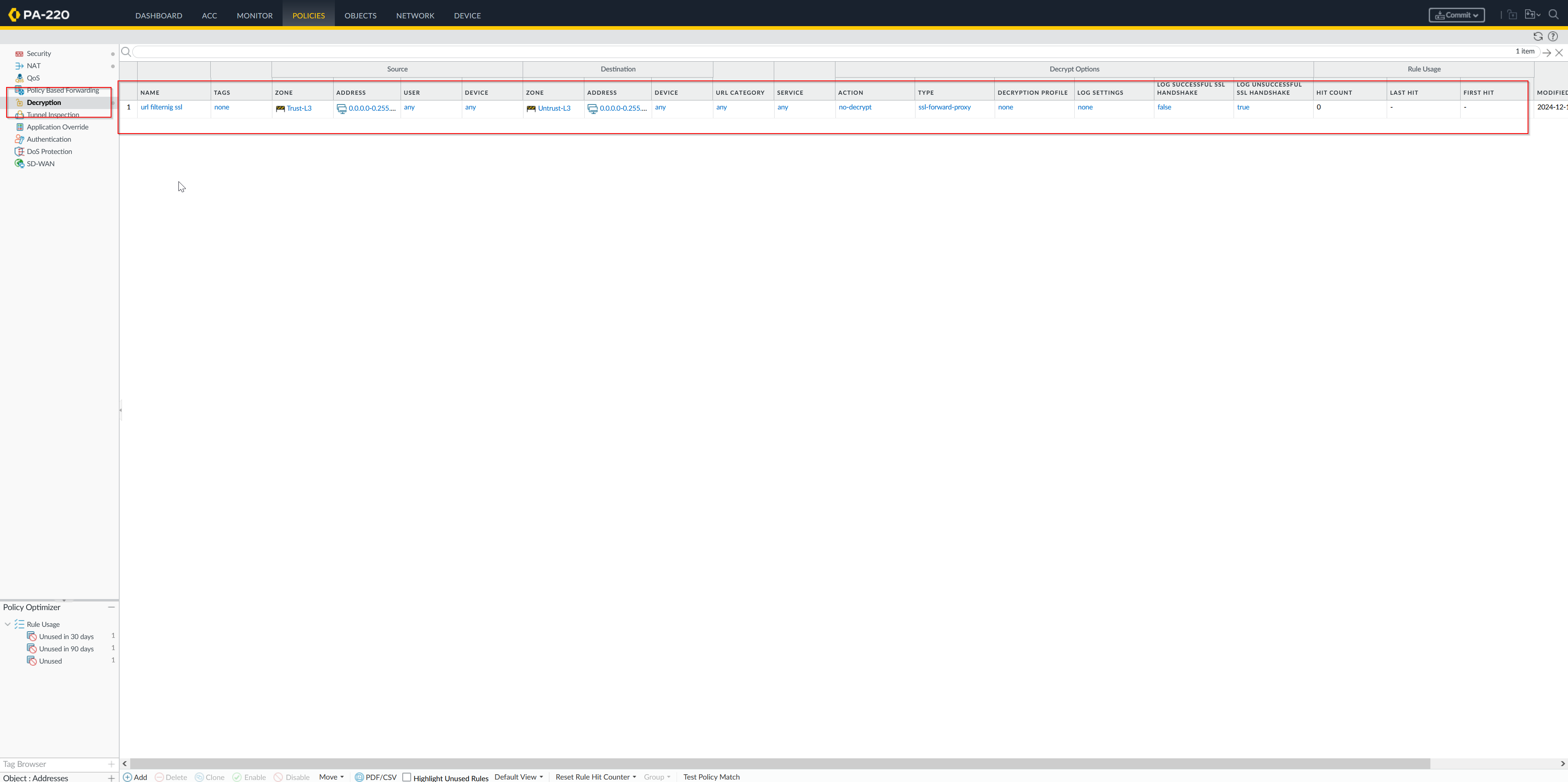
1. Now, we must generate our certificate for the firewall, which will be later imported into our end-devices. Navigate to Device > Certificates > Generate.



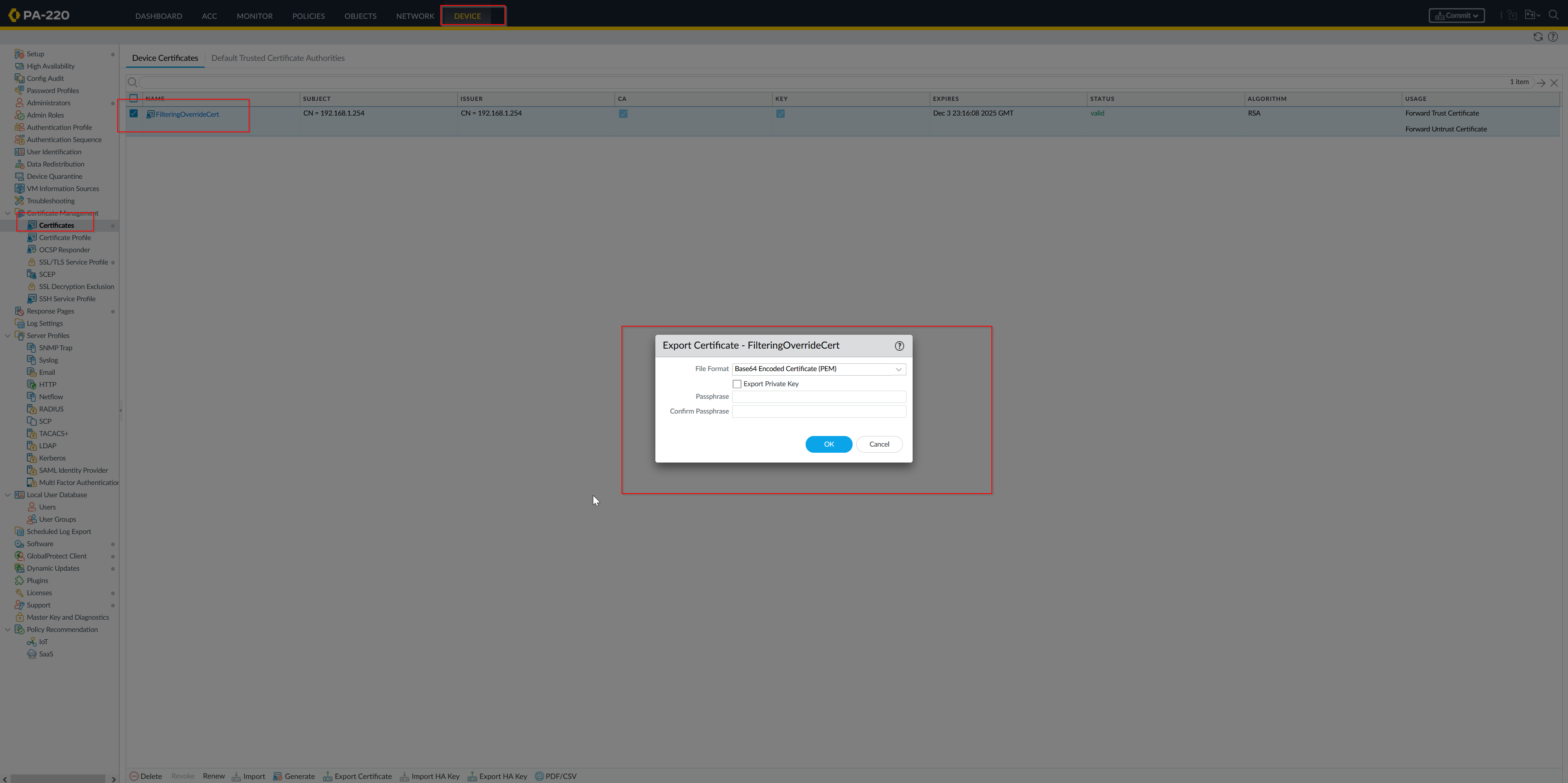
1. Set the following fields. Make sure your default gateway is configured as the correct address.



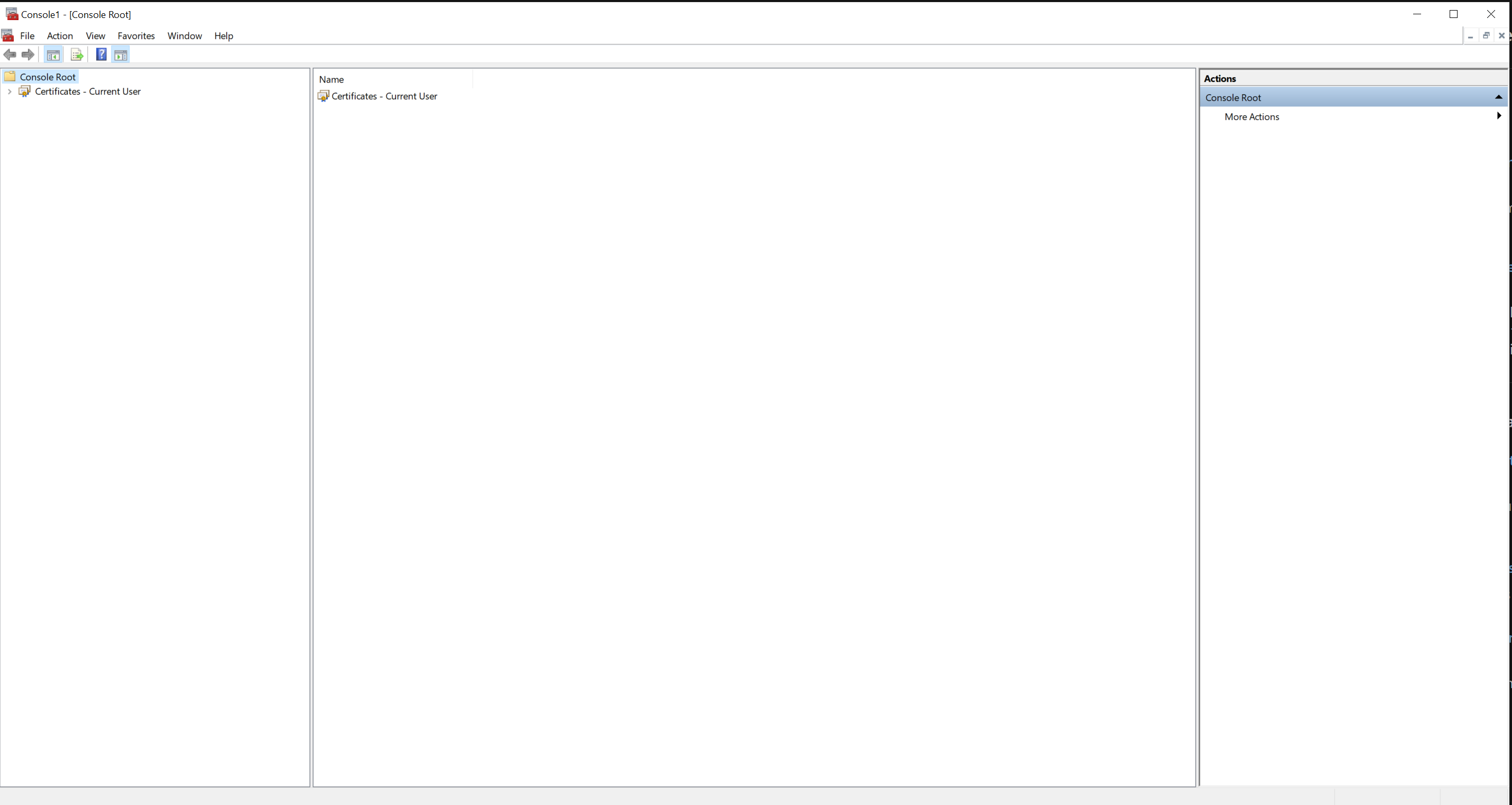
1. Now, go to Policies > Decryption and add a new policy. Make sure it is applied to all inbound and outbound traffic. This tells the firewall to apply decryption to all HTTPS traffic. Set the type as “ssl-decrypt-proxy”. This proxies the HTTPS traffic to our firewall for decryption, where our firewall can then insert the block page.



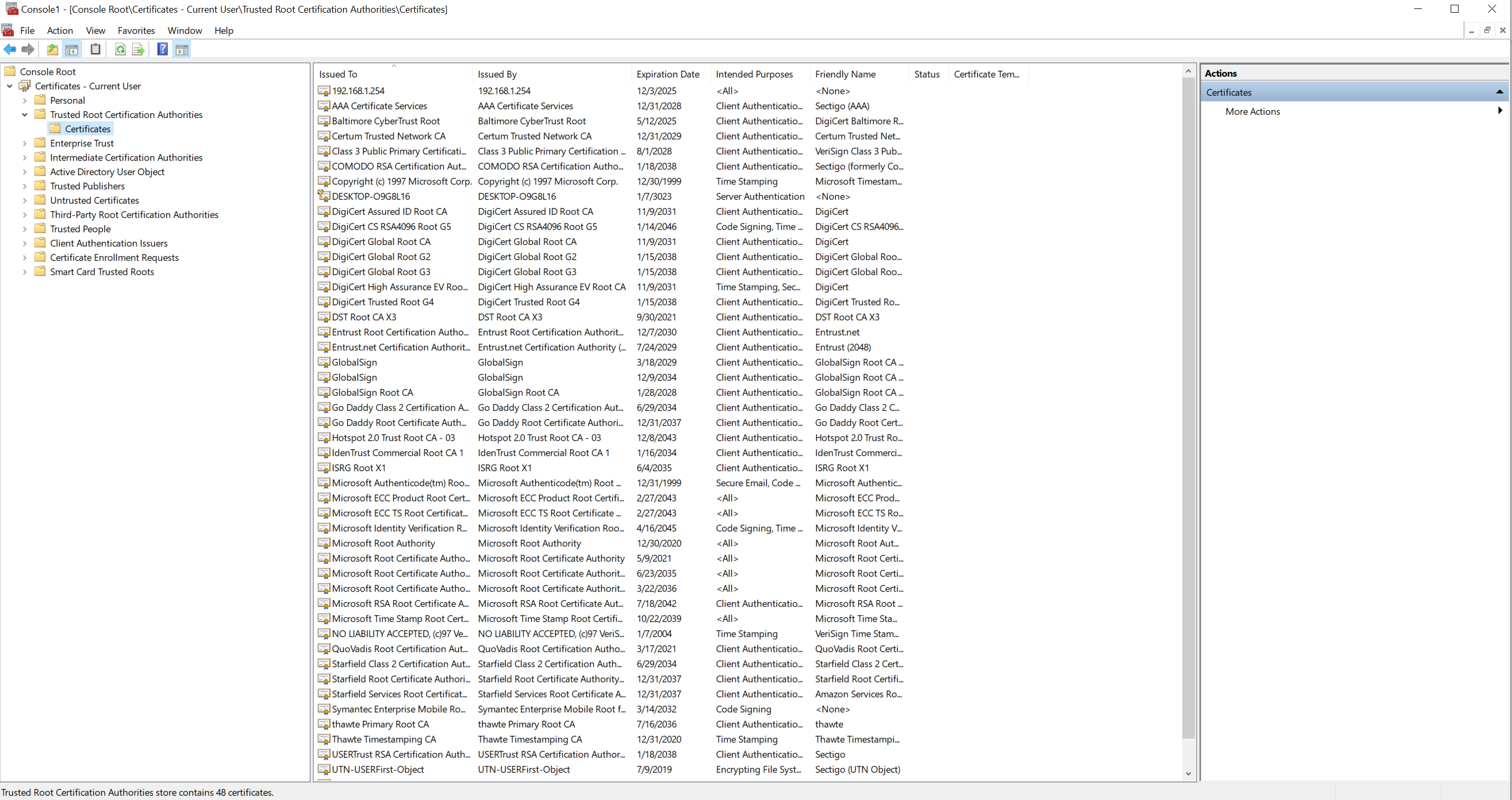
1. Finally, we need to install our Root Certificate on Windows for our end-device. Go back to Device > Certificates > Export Certificate to download the self-signed root certificate.



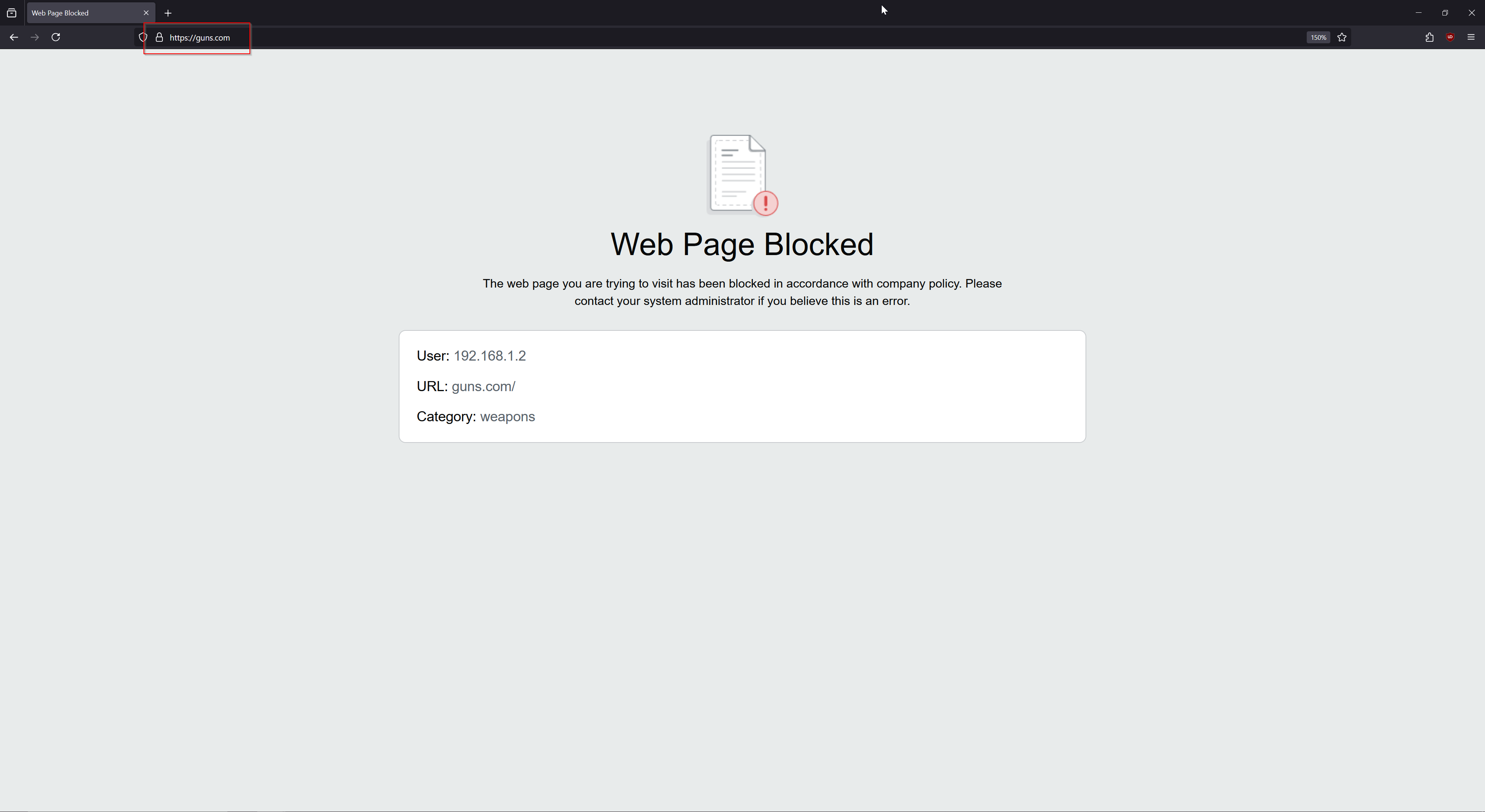
1. Install the root certificate. On Windows, this requires extra permissions. Press “Windows + r” and launch mmc, the Windows management console. Add Certificates to the menu.

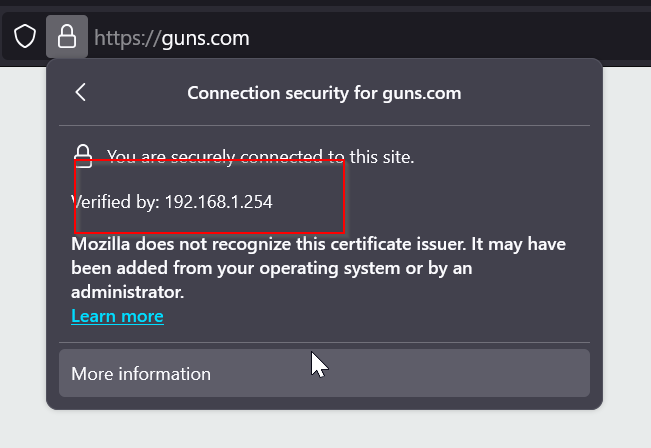
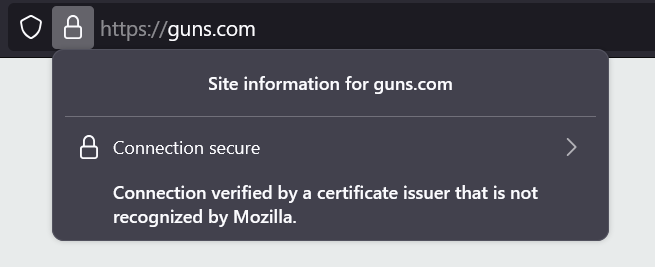


1. Navigate to the Trusted Root Certificate Authorities, and right click to add a new root certificate. Navigate to the download location of the certificate. Add the certificate.

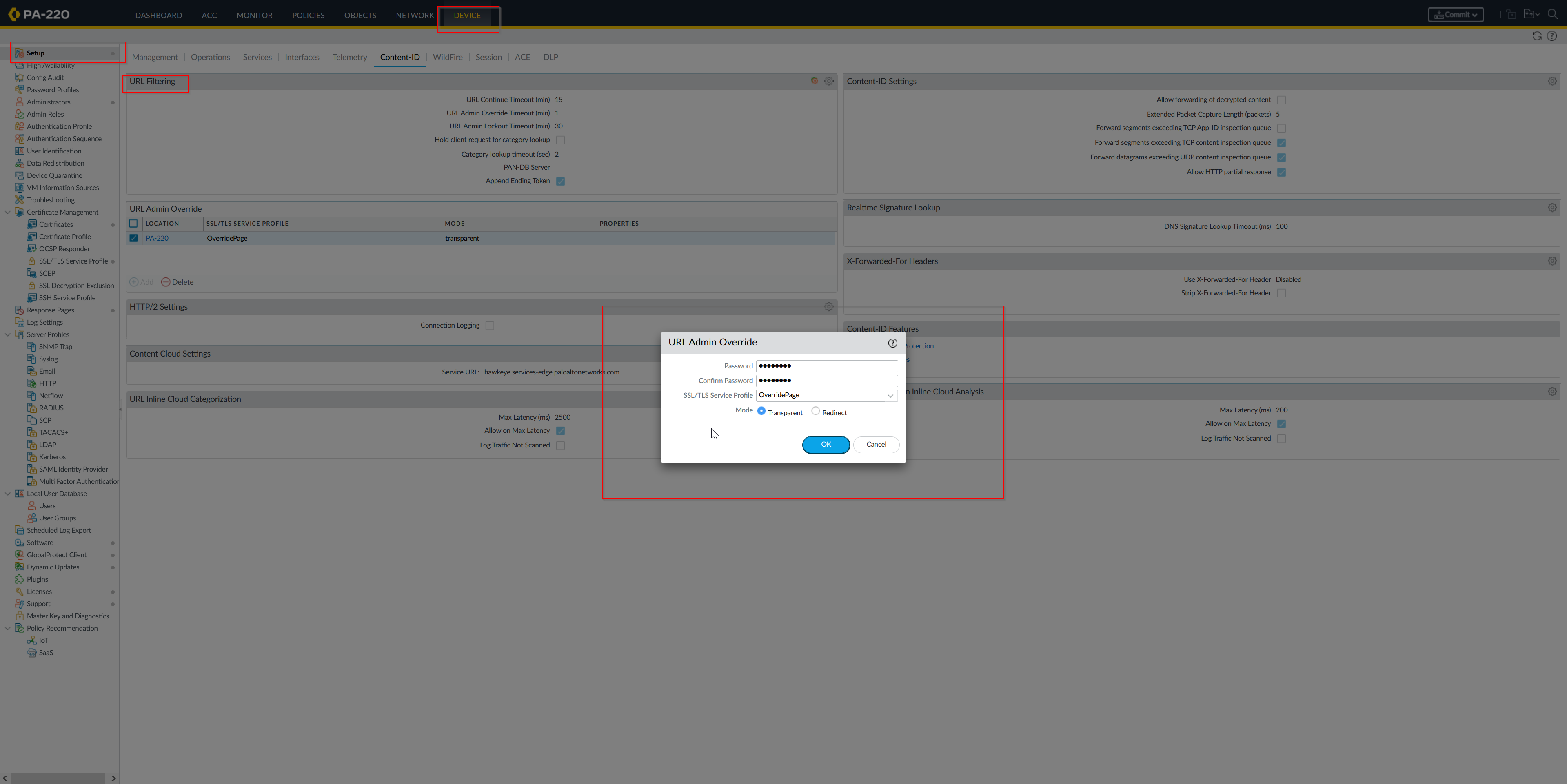


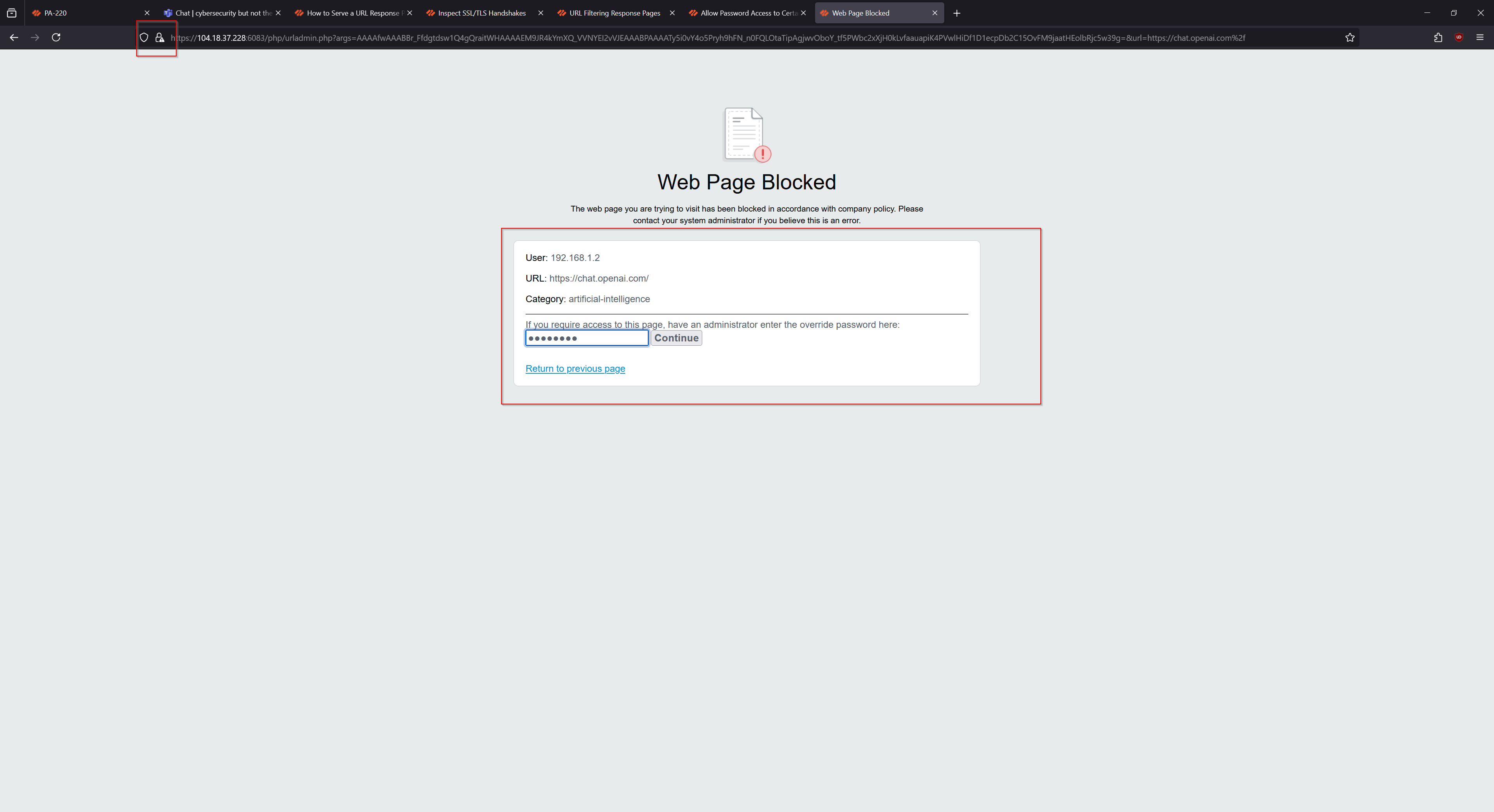
1. The root certificate should now be installed. Verify that HTTPS works with any blocked website.

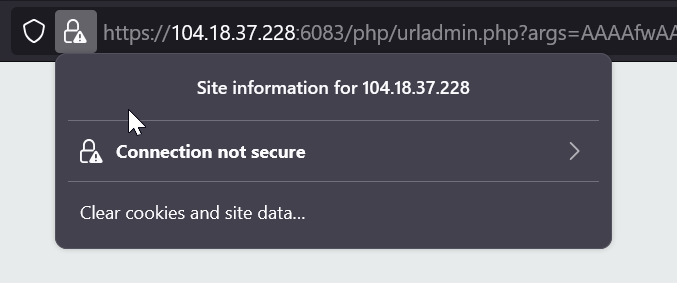


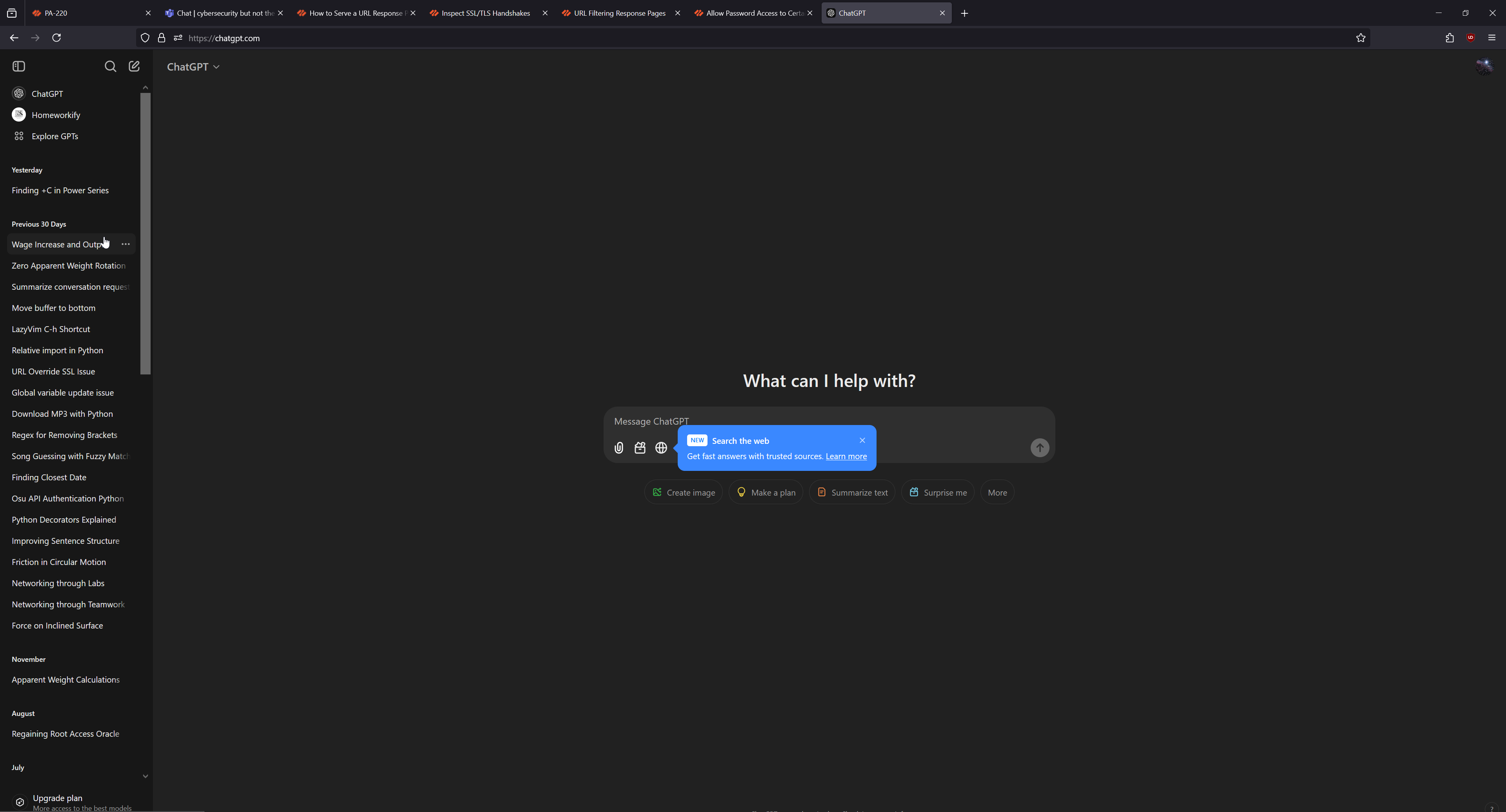


1. We can add the “override” settings now. Go to Device > Setup > URL Filtering and add a new URL Admin Override profile.



1. Verify that override works with a website that is configured to be override. In this case, artificial intelligence websites are used. 
2. Note that the website is shown as “Insecure”, because our certificate is self-signed, however our browser will still allow the connection to go through. URL Filtering is now done!





**Problems:**

1. **Invalid Palo Alto Device License/No URL Filtering License**

When we first started the lab, the main issue we had was with the licensing of our device. Since the PA-220 firewalls are handed down each year for the next class of students to configure, licenses may expire if not renewed. As a result, we needed to first obtain/renew our devices URL filtering license BEFORE configuring, as not doing so would result in error messages. Since categorization lists are handled by Palo Alto, licensing is required for our URL Filter to function properly.

1. **SSL Certificate Issues w/HTTPS**

We had a hard time understanding why HTTPS block pages were not working when we first started configuring the lab. Chrome would simply tell us a generic error, and it took us a while to realize that issuing the certificates was the issue. After self-signing and installing the root certificate on our device, the issues were fixed.

**Conclusion:**

This lab was a great introduction to how URL Filtering functions at a base level and what requirements are necessary for it to work. We were able to learn how HTTPS allows websites to securely transfer data, and certificates are required to enable HTTPS interception, and how block pages functions.

URL Filtering Signoff Sheet

Ryan Chen, P3-4 Cisco Cybersecurity, Mr. Mason

A white paper with a logo

Description automatically generated

