Task 1: Putting It All Together

In this task, all what we learned in previous modules is summarized.

- **I've read this…**

No answer needed just hit **"Complete"**

Task 2: Other Components

In this task, we were briefed regarding other required components i.e. Load Balancers, CDN (Content Delivery Networks), Databases, and WAF (Web Application Firewall).

- **What can be used to host static files and speed up a clients visit to a website?**

Flag: CDN

Reason: As stated in the description of this task, "**It allows**

you to host static files from your website, such as JavaScript, CSS, Images, Videos …”.

- **What does a load balancer perform to make sure a host is still alive?**

Flag: Health Check

Reason: As stated in the description of this task, “**Load Balancers also perform periodic checks with each server to ensure they are running correctly; this is called a health check.”**

- **What can be used to help against the hacking of a website?**

Flag: WAF

Reason: As stated in the description of this task, “**A WAF sits between your web request and the web server; its primary**

purpose is to protect the webserver from hacking or denial of service attacks.”

<u>Task 3</u>: How Web Servers Work

In this task, we were briefed regarding Web Server, Virtual Hosts, Static vs. Dynamic Content, Scripting and Backend Languages.

- **What does web server software use to host multiple sites?**

Flag: Virtual Hosts

Reason: As stated in the description of this task, “**Web servers can host multiple websites with different domain names; to achieve this, they use virutal hosts**”.

- **What is the name for the type of content that can change?**

Flag: Dynamic

Reason: As stated in the description of this task, "**Dynamic content, on the other hand, is content that could change with different requests**".

- **Does the client see the backend code? Yay/Nay**

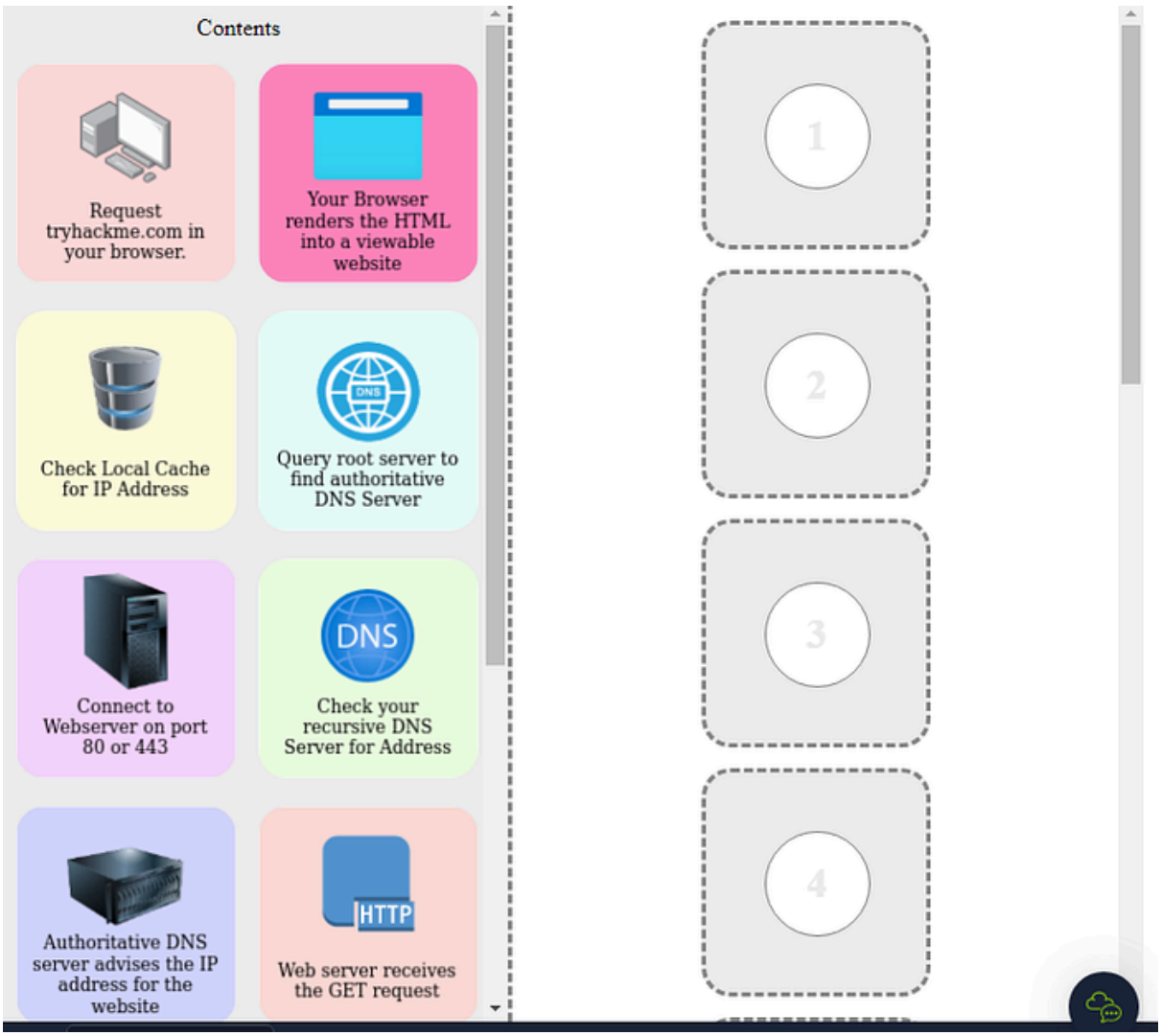
Flag: Nay

Reason: As stated in the description of this task, "**You can't view the websites' HTML source and see what's happening in the Backend…**".

Task 4: Quiz

In this task, we were to drag and drop the tiles into the correct order of how a request to a website works to reveal the flag. Deploy the lab by clicking on "View Site" and you will be welcomed with following screen.

Press enter or click to view image in full size



- **Flag**

Flag: THM{YOU_GOT_THE_ORDER}

Reason: Place the tiles in following order. And you will be flashed the flag via in-browser message, as can be witnessed from the screenshot attached below…

1. Request tryhackme.com in your browser.
2. Check Local Cache for IP Address
3. Check your recursive DNS Server for Address
4. Query root server to find authoritative DNS Server
5. Authoritative DNS server advises for the website
6. Request passes through a Web Application Firewall
7. Request passes through a Load Balancer
8. Connect to Webserver on port 80 or 443
9. Web server receives the GET request
10. Web Application talks to Database
11. Your Browser renders the HTML into a viewable website