AJP Proxy

According to Apache, AJP (or JK) is a wire protocol. It is an optimized version of the HTTP protocol to allow a standalone web server such as Apache to talk to Tomcat. Historically, Apache has been much faster than Tomcat at serving static content. The idea is to let Apache serve the static content when possible but proxy the request to Tomcat for Tomcat-related content.

When we come across open AJP proxy ports (`8009 TCP`) during penetration tests, we may be able to use them to access the "hidden" Apache Tomcat Manager behind it. Although AJP-Proxy is a binary protocol, we can configure our own Nginx or Apache webserver with AJP modules to interact with it and access the underlying application. This way, we can discover administrative panels, applications, and websites that would be otherwise inaccessible.

To see how we can configure our own Nginx or Apache webserver with AJP modules to interact with an open AJP proxy and access the underlying application, jump to the next interactive section.

Note: If you want to replicate such a vulnerable environment on a local machine, you can start an Apache Tomcat Docker exposing only the AJP-Proxy as follows:

First, create a file called `tomcat-users.xml` including the below.

tomcat-users.xml

tomcat-users.xml

```
<tomcat-users>
  <role rolename="manager-gui"/>
  <role rolename="manager-script"/>
  <user username="tomcat" password="s3cret" roles="manager-gui,manager-script"/>
</tomcat-users>
```

After this file is created, install the docker package in your local machine and start the Apache Tomcat Server by issuing the commands below.

Docker Installation

Docker Installation

```
Govardhan Gujji22@htb[/htb]$ sudo apt install docker
Govardhan Gujji22@htb[/htb]$ sudo docker run -it --rm -p 8009:8009 -v `pwd`/tomcat-users.xml:
```

Previous Next Mark Complete & Next

Table of Contents

My Workstation

OFFLINE

Start Instance

∞ / 1 spawns left