

Nginx Reverse Proxy & AJP

[? Go to Questions](#)

When we come across an open AJP proxy port (8009 TCP), we can use Nginx with the `ajp_module` to access the "hidden" Tomcat Manager. This can be done by compiling the Nginx source code and adding the required module, as follows:

- Download the Nginx source code
- Download the required module
- Compile Nginx source code with the `ajp_module`.
- Create a configuration file pointing to the AJP Port

Download Nginx Source Code

Download Nginx Source Code

```
Govardhan Gujji22@htb[/htb]$ wget https://nginx.org/download/nginx-1.21.3.tar.gz
Govardhan Gujji22@htb[/htb]$ tar -xzvf nginx-1.21.3.tar.gz
```

Compile Nginx source code with the ajp module

Compile Nginx source code with the ajp module

```
Govardhan Gujji22@htb[/htb]$ git clone https://github.com/dvershinin/nginx_ajp_module.git
Govardhan Gujji22@htb[/htb]$ cd nginx-1.21.3
Govardhan Gujji22@htb[/htb]$ sudo apt install libpcre3-dev
Govardhan Gujji22@htb[/htb]$ ./configure --add-module=../nginx_ajp_module --prefix=/etc/nginx
Govardhan Gujji22@htb[/htb]$ make
Govardhan Gujji22@htb[/htb]$ sudo make install
Govardhan Gujji22@htb[/htb]$ nginx -V

nginx version: nginx/1.21.3
built by gcc 10.2.1 20210110 (Debian 10.2.1-6)
configure arguments: --add-module=../nginx_ajp_module --prefix=/etc/nginx --sbin-path=/usr/st...
```

Note: In the following configuration, we are using port 8009, which is Tomcat's default port for AJP, and this is how we would use it in a real environment. However, to complete the exercise at the end of this section you should specify the IP and port of the target you will spawn (they will both be visible right next to "Target"). The port you will see is essentially mapped to port 8009 of the underlying Docker container.

Comment out the entire `server` block and append the following lines inside the `http` block in `/etc/nginx/conf/nginx.conf`.

Pointing to the AJP Port

Pointing to the AJP Port

```
upstream tomcats {
    server <TARGET_SERVER>:8009;
    keepalive 10;
}
server {
    listen 80;
    location / {
        ajp_keep_conn on;
        ajp_pass tomcats;
    }
}
```

Note: If you are using Pwnbox, then port 80 will be in use already, so, in the above configuration change port 80 to 8080. Finally, in the next step, use port 8080 with cURL.

Start Nginx and check if everything is working correctly by issuing a cURL request to your local host.

Pointing to the AJP Port

```
Govardhan Gujji22@htb[/htb]$ sudo nginx
Govardhan Gujji22@htb[/htb]$ curl http://127.0.0.1:80

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8" />
        <title>Apache Tomcat/X.X.XX</title>
        <link href="favicon.ico" rel="icon" type="image/x-icon" />
        <link href="favicon.ico" rel="shortcut icon" type="image/x-icon" />
        <link href="tomcat.css" rel="stylesheet" type="text/css" />
    </head>
    <body>
        <div id="wrapper">
            <div id="navigation" class="curved container">
                <span id="nav-home"><a href="https://tomcat.apache.org/">Home</a></span>
                <span id="nav-hosts"><a href="/docs/">Documentation</a></span>
                <span id="nav-config"><a href="/docs/config/">Configuration</a></span>
                <span id="nav-examples"><a href="/examples/">Examples</a></span>
                <span id="nav-wiki"><a href="https://wiki.apache.org/tomcat/FrontPage">Wiki</a></span>
                <span id="nav-lists"><a href="https://tomcat.apache.org/lists.html">Mailing Lists</a></span>
                <span id="nav-help"><a href="https://tomcat.apache.org/findhelp.html">Find Help</a></span>
            </div>
            <div id="asf-box">
                <h1>Apache Tomcat/X.X.XX</h1>
            </div>
            <div id="upper" class="curved container">
                <div id="congrats" class="curved container">
                    <h2>If you're seeing this, you've successfully installed Tomcat. Congrat...
                </div>
            </div>
        </div>
    </body>

```

<SNIP>

Table of Contents

Introduction to Server-Side Attacks	✓
Abusing Intermediary Applications	
AJP Proxy	✓
Nginx Reverse Proxy & AJP	✓
Apache Reverse Proxy & AJP	✓

Server-Side Request Forgery (SSRF)

Server-Side Request Forgery (SSRF) Overview	✓
SSRF Exploitation Example	✓
Blind SSRF	✓
Blind SSRF Exploitation Example	✓
Time-Based SSRF	✓

Server-Side Includes (SSI) Injection

Server-Side Includes Overview	✓
SSI Injection Exploitation Example	✓

Edge-Side Includes (ESI) Injection

Edge-Side Includes (ESI)	✓
--------------------------	---

Server-Side Template Injections

Introduction to Template Engines	✓
SSTI Identification	✓
SSTI Exploitation Example 1	✓
SSTI Exploitation Example 2	✓
SSTI Exploitation Example 3	✓

Extensible Stylesheet Language Transformations Server-Side Injections

Attacking XSLT	✓
----------------	---

Skills Assessment

Server-Side Attacks - Skills Assessment	✓
---	---

My Workstation



OFFLINE

Start Instance

∞ / 1 spawns left

Start Instance

∞ / 1 spawns left

Waiting to start...

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target: [Click here to spawn the target system!](#)

- + 2 🌟 Replicate the steps shown in this section to connect to the above server's "hidden" Tomcat page through the AJP proxy, then write the Tomcat version as your answer. Remember that the port you will see next to "Target" will be the AJP proxy port. Answer format: X.X.XX

Submit your answer here...

Submit

◀ Previous

Next ▶