**NICE Challenge Lab Report**

**Student Name:** Dustin McClure

**Lab Name:** Engineer's Audit Advice

**What software did you install?**

* **HTTP-6640(**<https://phoenixnap.com/kb/apache-mod-evasive>)**:** 
  + sudo apt install apache2-utils
  + sudo apt install libapache2-mod-evasive
  + **SEE CONFIGURATION MODIFICATION SECTION**
* **HTTP-6641(**[**https://ourcodeworld.com/articles/read/950/how-to-protect-your-apache-server-from-dos-attacks-denial-of-service-using-the-quality-of-service-qos-module-on-ubuntu-1604**](https://ourcodeworld.com/articles/read/950/how-to-protect-your-apache-server-from-dos-attacks-denial-of-service-using-the-quality-of-service-qos-module-on-ubuntu-1604)**):**
  + sudo apt install libapache2-mod-qos
  + **SEE CONFIGURATION MODIFICAION SECTION BEFORE MOVING TO NEXT COMMAND**
  + sudo service apache2 restart
* **AUTH-9262 and AUTH-9286 and Ashley Request (**[**https://www.cyberciti.biz/faq/securing-passwords-libpam-cracklib-on-debian-ubuntu-linux/**](https://www.cyberciti.biz/faq/securing-passwords-libpam-cracklib-on-debian-ubuntu-linux/)**):**
  + sudo apt install libpam-cracklib
  + sudo cp /etc/pam.d/common-password /root/
  + **SEE CONFIGURATION MODIFICATION SECTION**
* **HTTP-6643(**[**https://phoenixnap.com/kb/setup-configure-modsecurity-on-apache**](https://phoenixnap.com/kb/setup-configure-modsecurity-on-apache)**):**
  + sudo apt install libapache2-modsecurity
  + sudo systemctl restart apache2
  + **SEE CONFIGURATION MODIFICAION SECTION BEFORE MOVING TO NEXT COMMAND**
  + sudo systemctl restart apache2
  + Install git:
    - sudo apt install git
  + git clone <https://github.com/SpiderLabs/owasp-modsecurity-crs.git>
  + cd owasp-modsecurity-crs
  + sudo mv crs-setup.conf.example /etc/modsecurity/crs-setup.conf
  + sudo mv rules/ /etc/modsecurity
  + **SEE CONFIGURATION MODIFICATION SECTION BEFORE MOVING TO NEXT COMMAND**
  + sudo systemctl restart apache2
* **DEB-0880(**[**https://linuxize.com/post/install-configure-fail2ban-on-debian-10/**](https://linuxize.com/post/install-configure-fail2ban-on-debian-10/)**):**
  + sudo apt install fail2ban
  + sudo cp /etc/fail2ban/jail.{conf,local}
  + **SEE CONFIGURATION MODIFICATION SECTION**
* **ACCT-9622(**[**https://linuxhint.com/enable-process-accounting-ubuntu/**](https://linuxhint.com/enable-process-accounting-ubuntu/)**):**
  + sudo apt install acct
  + sudo /usr/sbin/accton on
  + To view process accounting log use the ***ac*** command

**What configuration files did you modify?**

* **HTTP-6640(**<https://phoenixnap.com/kb/apache-mod-evasive>)**:**
  + sudo nano /etc/apache2/mods-enabled/evasive.conf
    - Remove hash from following comments:
      * #DOSEmailNotify
      * #DOSHashTableSize 3097
      * #DOSPageCount 2
      * #DOSSiteCount 50
      * #DOSPageInterval 1
      * #DOSSiteInterval 1
      * #DOSBlockingPeriod 10
      * #DOSEmailNotify [mail@yourdomain.com](mailto:mail@yourdomain.com)
      * Text

        Description automatically generated#DOSLogDir "/var/log/apache2/"

*This is a screenshot of the modified evasive.conf file for mod\_evasive that sets parameters for mod\_evasive to reference, enables logging in /var/log/mod\_evasive, and sends email notifications of potential DOS attacks to me.*

* **HTTP-6641(**[**https://ourcodeworld.com/articles/read/950/how-to-protect-your-apache-server-from-dos-attacks-denial-of-service-using-the-quality-of-service-qos-module-on-ubuntu-1604**](https://ourcodeworld.com/articles/read/950/how-to-protect-your-apache-server-from-dos-attacks-denial-of-service-using-the-quality-of-service-qos-module-on-ubuntu-1604)**):**
  + sudo nano /etc/apache2/mods-available/qos.conf
    - Change between opening tag ***<IfModule qos\_module>*** and closing tag ***<IfModule>*** to the following:
      * QS\_ClientEntries 100000
      * QS\_SrvMaxConnPerIP 50
      * MaxClients 256
      * QS\_SrvMaxConnClose 180
      * QS\_SrvMinDataRate 150 1200
* **AUTH-9262 and AUTH-9286 and Ashley Request (**[**https://www.cyberciti.biz/faq/securing-passwords-libpam-cracklib-on-debian-ubuntu-linux/**](https://www.cyberciti.biz/faq/securing-passwords-libpam-cracklib-on-debian-ubuntu-linux/)**):**
  + sudo nano /etc/pam.d/common-password
    - Change line:
      * password requisite pam\_cracklib.so retry=3 minlen=8 difok=3
      * TO
      * password requisite pam\_cracklib.so retry=3 minlen=10 difok=3 ucredit=-1 lcredit=-2 dcredit=-2 ocredit=-2
  + sudo nano /etc/login.defs
    - Change line:
      * Text

        Description automatically generated***PASS\_MIN\_DAYS 0*** to ***PASS\_MIN\_DAYS 3***

*This is a screenshot of the modified pam common-password file that extends the minimum password length to 10 characters and requires lowercase letters, uppercase letters, digits, and at least one special character.*

* **HTTP-6643(**[**https://phoenixnap.com/kb/setup-configure-modsecurity-on-apache**](https://phoenixnap.com/kb/setup-configure-modsecurity-on-apache)**):**
  + sudo cp /etc/modsecurity/modsecurity.conf-recommended /etc/modsecurity/modsecurity.conf
  + sudo cd /etc/modsecurity
  + sudo nano modsecurity.conf
  + Change line: ***SecRuleEngine DetectionOnly*** to ***SecRuleEngine On***
  + **GO BACK TO SOFTWARE INSTALL SECTION BEFORE MOVING FORWARD**
  + sudo nano /etc/apache2/mods-enabled/security2.conf
  + Ensure these lines are included in security2.conf: ***IncludeOptional /etc/modsecurity/\*.conf*** and ***Include /etc/modsecurity/rules/\*.conf***
  + **GO BACK TO SOFTWARE INSTALL SECTION**
* **DEB-0880(**[**https://linuxize.com/post/install-configure-fail2ban-on-debian-10/**](https://linuxize.com/post/install-configure-fail2ban-on-debian-10/)**):**
  + sudo nano /etc/fail2ban/jail.local
  + Ensure line ***ignoreip*** includes ***127.0.0.1/8 ::1*** and applicable internal subnets
  + Find line starting with bantime and set it to ***bantime = 1d***
  + Find line starting with findtime and set it to ***findtime = 10m***
  + Find line starting with maxretry and set it to ***maxretry = 5***

**What I Learned**

I have used Linux for quite some time (about 12 years at this point). During this time, I have always used it as a personal operating system and have never really had to maintain a server with multiple users and multiple accounts. Being the only user and consistently setting complex passwords, I did not really need to dig into pam’s password modules. Doing this lab helped familiarize me with the different pam modules available for ensuring a secure password, where to find the configuration files necessary to set password policy using these modules, and what options are available for setting password policies.

In the past when I have helped administer servers, their purpose was of a limited scope, and they were hosted with service providers that provided some assistance with DoS/DDoS protection. With this being the case and because the company in question was low risk for a DoS attack, we trusted the service provider’s existing protections and I never looked into further protection.

In doing this lab I was introduced to two Apache modules used for the purpose of protecting against DoS/DDoS, mod\_evasive and mod\_qos. Though it is quite simple, I learned how to install both of these modules; more importantly I learned what their purpose is and how to configure them. Apache mod\_evasive as it’s name implies, provides a framework for servers to evade DoS/DDoS attacks, and mod\_qos helps to mitigate the slowloris flavor of DoS attack which attempts to max out the number of concurrent connections allowed on the server by keeping multiple connections open for extended periods of time.

Apache mod\_evasive does it’s job by denying ip addresses if they exceed a certain number of requests for a page, try to open too many connections to a site at the same time, or try to make requests for too many pages on a site in a short period of time; blocking any requests for a period of time from any IP that has recently been denied.

Apache mod\_qos helps to mitigate the slowloris attack by ensuring that traffic of certain types gets priority on a network. Apache “mod\_qos controls the number of concurrent requests to a name space (URL)….”, “… controls the number of TCP connections...” and drops suspicious requests. *(Mod qos, 2021)*

# Works Cited

*Mod qos*. (2021, September 25). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Mod\_qos