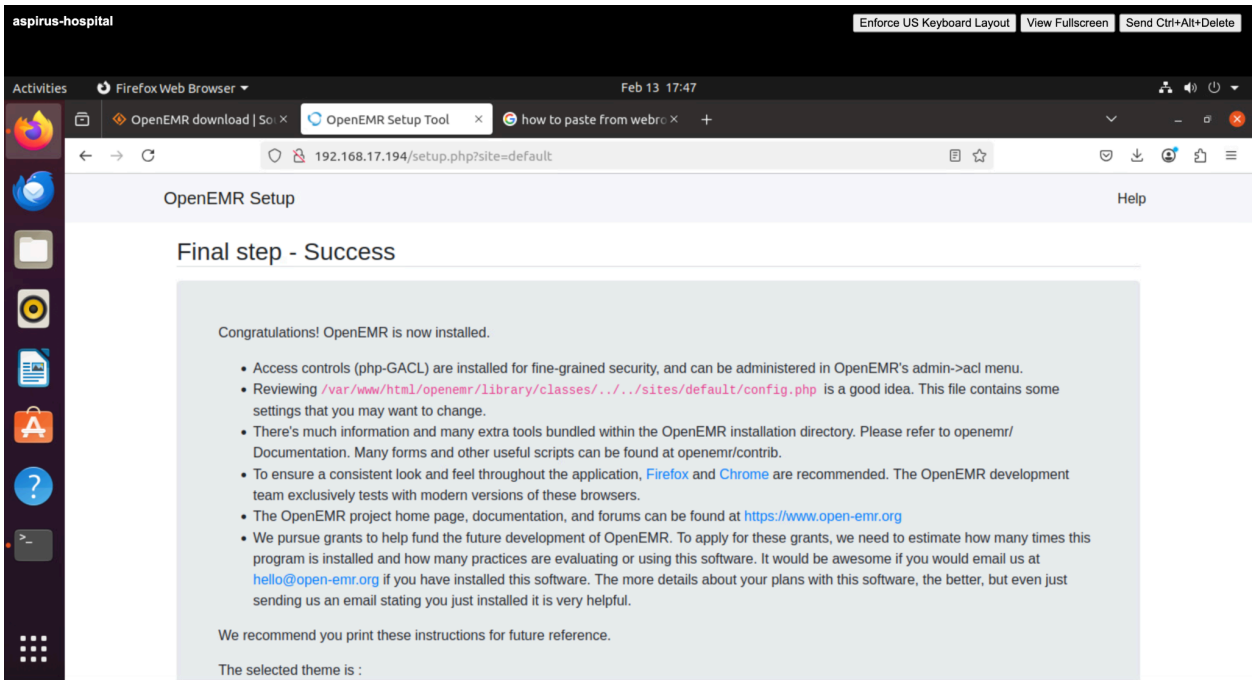
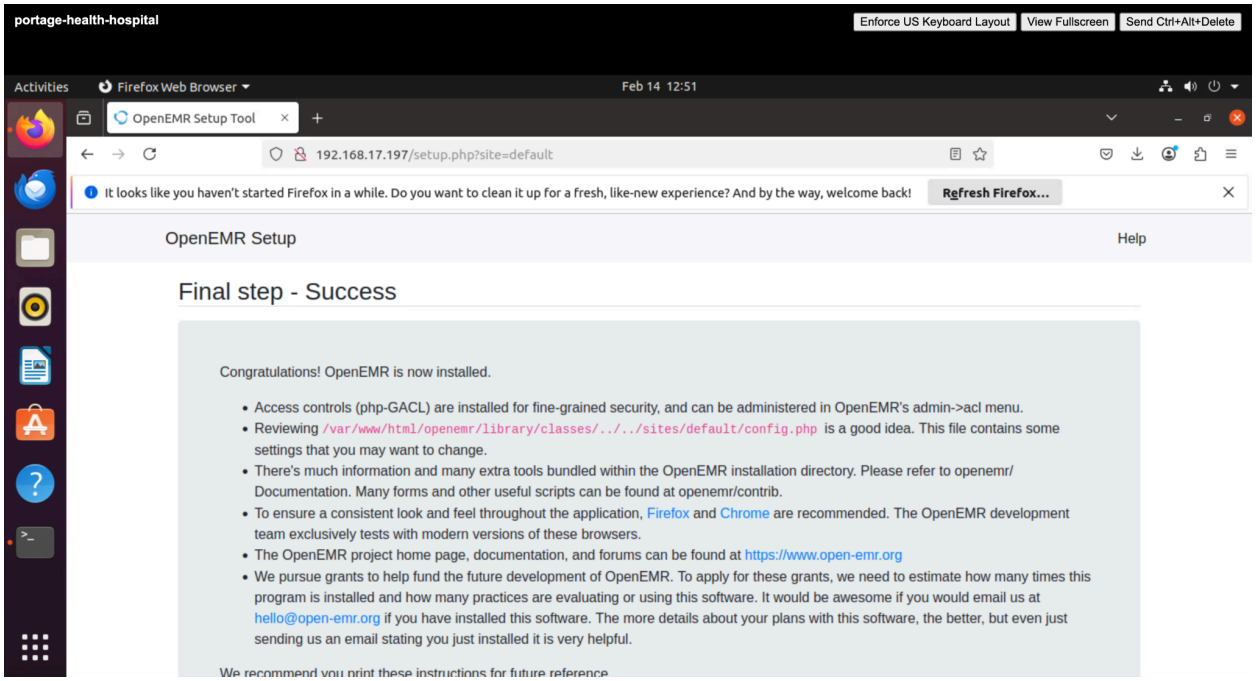
Architecture Development Part 2
SAT 5424 Population Health Informatics
Eli Pinnoo
Professor Changarnkothapeecherikkal

A. Web Page Screenshots of OpenEMR Installation:

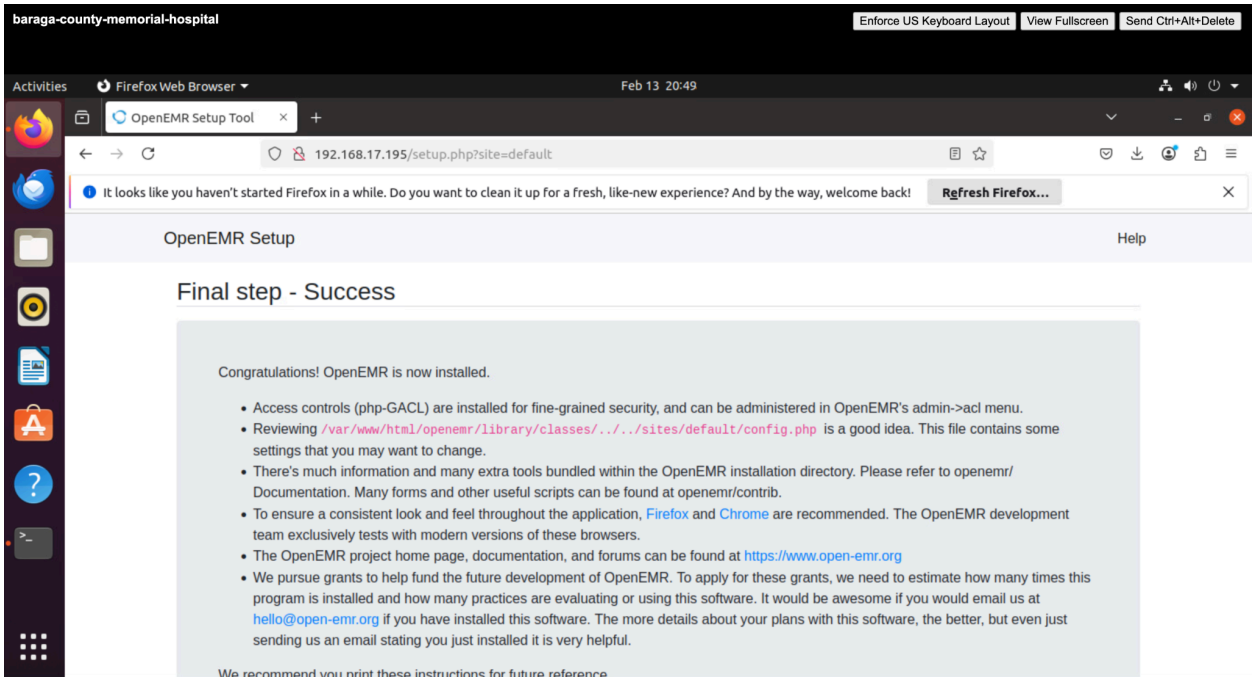
- Aspirus:



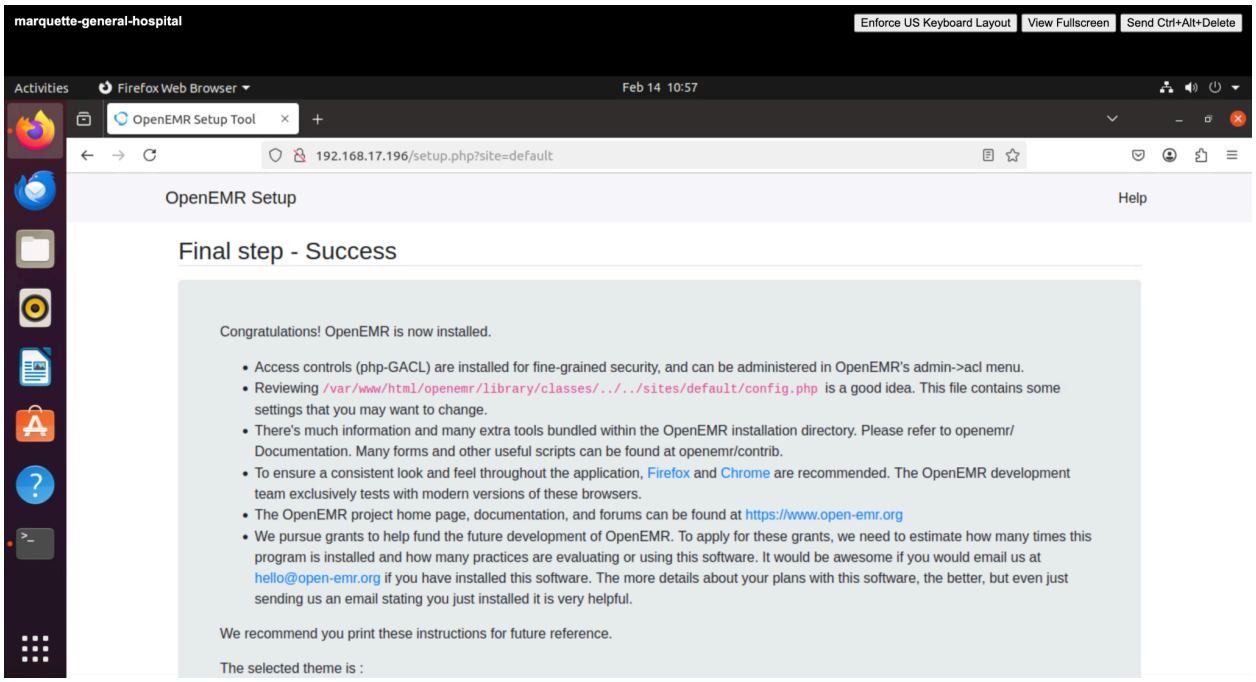
- Portage Health Hospital:



- Baraga County Memorial Hospital (BCMH):



- Marquette General Hospital (MGH):



B. Steps and their Commands to Secure OpenEMR:

1. Enabled Automatic Security Updates:
    a. First I installed the unattended-upgrades using the following command:
    sudo apt-get install unattended-upgrades
    b. I then enabled the automatic security updates with this command:
    sudo dpkg-reconfigure --priority=low unattended-upgrades

2. Configured a Firewall:
    a. First I installed the firewall using the following command:
    sudo apt-get install ufw
    b. I then allowed traffic for HTTP, HTTPS, and SSH with the following:
    sudo ufw allow http
    sudo ufw allow https
    sudo ufw allow ssh
    c. Finally, I enabled the firewall using this command:
    sudo ufw enable

3. Secured Apache:
    a. I opened the Apache security configuration file using this command:
    sudo nano /etc/apache2/conf-available/security.conf
    b. Inside of the Apache security config. file, I made the following changes:
    ServerTokens Prod
    ServerSignature Off
    TraceEnable Off
    Header set X-Content-Type-Options: "nosniff"
    Header set X-Frame-Options: "sameorigin"
    Header set X-XSS-Protection: "1; mode=block"
    Header set X-Robots-Tag: "none"
    Header set X-Download-Options: "noopen"
    Header set X-Permitted-Cross-Domain-Policies: "none"
    c. I then enabled the new security headers I added with:
    sudo a2enconf security
    d. Finally, I restarted apache2 to apply the security changes with:
    sudo systemctl restart apache2

4. Use Strong Passwords:
    a. I use a password for my account that is larger than 12 characters and meets OpenEMR's security standards.

C. Other Attacks still Susceptible to OpenEMR:

- Since OpenEMR uses MySQL, it is still a target for SQL injection attacks.
- With the database and the use of privileges, OpenEMR could still encounter privilege escalation. One of the best ways to protect against this would be to only allow users to access what they need to use.
- With the use of passwords, there is still potential for a few weak passwords to be cracked by brute force or other methodologies.
- There is the chance that careless users of OpenEMR will leak data or it could be of malicious intent.