Week 7 Brief

10/26/21

Team Activities

- Jake and Justin:
  - Continued working on a script to create vulnerabilities
  - Started a new script to exploit the vulnerabilities

- Megan and Jacob:
  - Cleaned up Gitlab to make it more accessible (Removed VMs)
  - Began a website to emulate a Web Interface for IoT device on Gitlab

Basic Configuration Script

- Tested what we have thus far in VirtualBox
  - Was successful
- Will continue to brainstorm and expand
- May create a firewall script
  - To set basic firewall rules
  - Using iptables

```
#!/bin/bash

#create a user
echo "create user"
useradd sdmay14

#pipe password into passwd command
echo -e "password\npassword" | passwd sdmay14

#give initial user sudo privileges
#possibly wheel instead of sudo
usermod -aG sudo sdmay14

echo "install updates"
#yum update
apt-get update

echo "install ssh"
#yum -y install openssh-server
apt install openssh-server

echo "enable ssh"
systemctl enable ssh
systemctl start ssh

echo "install telnet"
#yum install telnet-server telnet
apt-get install telnetd

echo "enable telnet"
systemctl enable telnet.socket
systemctl start telnet.socket
```

Basic Firewall script idea

```
#!/bin/bash

#drop incoming packets by default
iptables -t filter -P INPUT DROP

#accept packets forwarded through router/firewall by default
iptables -t filter -P FORWARD ACCEPT

#accept outgoing packets by default
iptables -t filter -P OUTPUT ACCEPT

#allow ping
iptables -A FORWARD -p icmp -j ACCEPT

#allow SSH
iptables -A FORWARD -p tcp --dport 22 -j ACCEPT
```

Exploitation Script

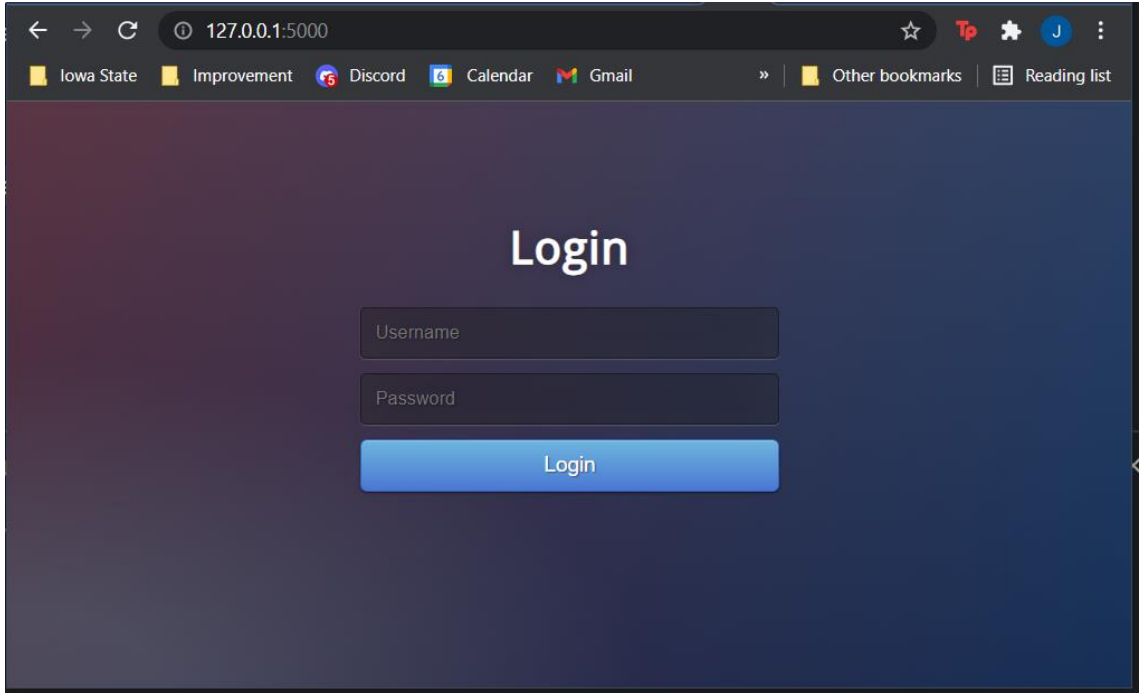
- Script will be ran(acting as Red Team)
  - Will be score based
- Exploitations:
  - Telnet and su using default creds
  - SSH
    - Use “expect” to automate
- Will continue to brainstorm and expand
  - DoS or DDoS attacks
  - Ping or SYN flood to take down services

```
#!/bin/expect -f

#keep track of failed vulnerabilities
#increment on successful exploitation
score = 0

#install Expect to automate SSH script
#apt-get install expect
yum install expect-devel

#telnet in and use echo to type commands using telnet
telnet <IoTIP>

#try logging in to default account with default password
echo "su sdmay14"
echo "password"

#ssh in
spawn ssh sdmay14@<IoTIP>

#antipate password prompt
expect "password: "

#send default password
send "password\r"

#if successful, expect user prompt
expect "$ "

#exit ssh session
send "exit\r"
```

Current IoT Web Interface

Many IoT devices have apps or web interfaces that enable users to interact and control the device remotely.

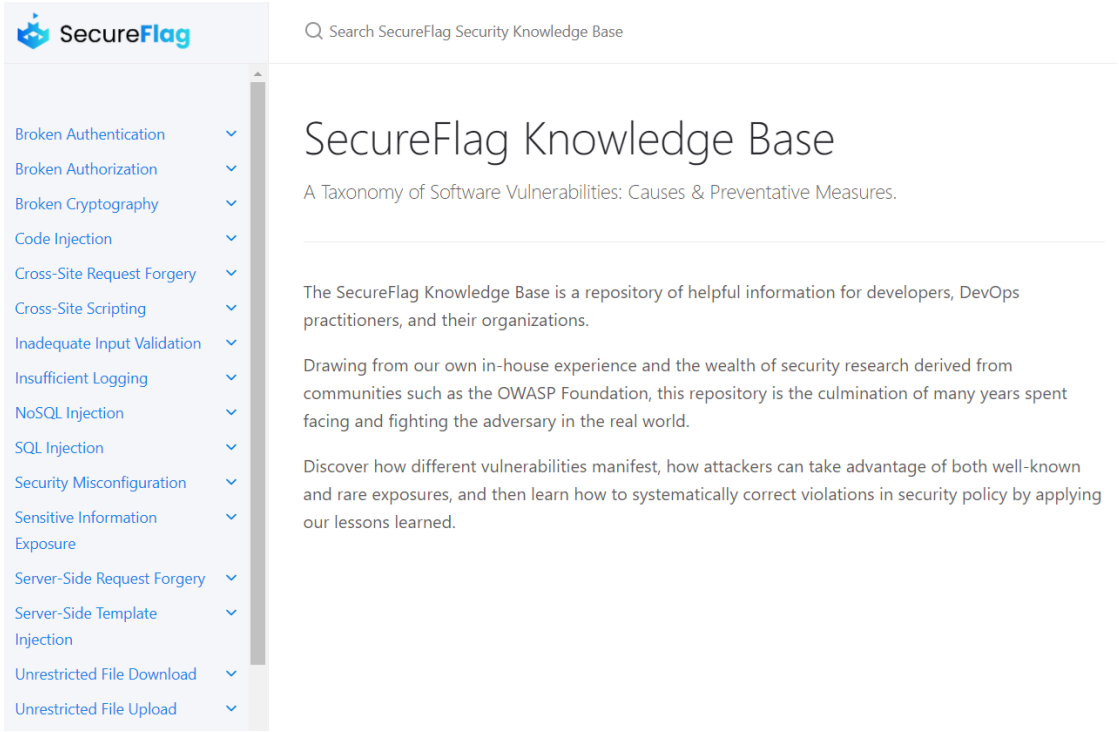
These applications and web interfaces are commonly insecure. Therefore, or team can develop a vulnerable website hosted on an IoT VM to emulate real world weaknesses.



Web Vulnerabilities

Our team has located a Website with code examples for web-based vulnerabilities such as XSS, Broken Authentication, SQL injection, etc.

We can use these code examples as reference and inspiration for our vulnerable Website design.



Web Vulnerabilities Examples:

- Broken Authentication

  Navigating to http://X.X.X.X/?login=true bypasses authentication

- Local File Inclusion

  Navigating to http://X.X.X.X/../../../../etc/shadow prints contents of file

- Broken Authorization

  Navigating to http://X.X.X.X/userid=**4** gives you session of user 4

This week “To Do”

- Continue expanding and researching scripting

- **Decide what Virtualized IoT device we want to build** so we can customize our scripts, website, and vulnerabilities to said device

  (Not a camera)