Encrypt communications with SSI/TSL

Locate web.conf and change: sudo nano /opt/splunk/etc/system/local/server.conf

Splunk Defense Playbook for CCDC

1. Exploiting Known Vulnerabilities

Attack Scenario:

Exploiting known Splunk vulnerabilities for remote code execution or privilege escalation.

Detection Commands:

```
# Check for unusual processes spawned by Splunk
egrep -i "splunkd|bash|cmd.exe" /var/log/syslog
ps aux | grep splunk

# Identify version and known vulnerabilities
splunk version
curl -s https://splunk.com/security-updates
```

Mitigation Steps:

```
# Update Splunk to the latest version
wget -O splunk-latest.rpm https://download.splunk.com/path-to-latest.rpm
rpm -Uvh splunk-latest.rpm

# Apply security patches
splunk apply shcluster-bundle

# Restrict internet access for Splunk
iptables -A OUTPUT -p tcp --dport 80 -j DROP
```

---

2. Credential Theft and Privilege Escalation

Attack Scenario:

Using brute force or stolen credentials to access the Splunk web UI.

Detection Commands:

```
# Monitor failed login attempts
grep "failed login" /opt/splunk/var/log/splunk/splunkd.log

tail -f /opt/splunk/var/log/splunk/audit.log | grep "login attempt"
```

Mitigation Steps:

```
# Enforce strong password policies
splunk edit auth ldap -minPwdLength 12 -mustChangePassword true

# Enable MFA
splunk enable auth-mfa

# Restrict access by IP
iptables -A INPUT -p tcp --dport 8000 -s TRUSTED_IP -j ACCEPT
iptables -A INPUT -p tcp --dport 8000 -j DROP
```

---

3. Denial-of-Service (DoS) Attacks

Attack Scenario:

Flooding Splunk with excessive log ingestion.

Detection Commands:

```
# Check for high CPU/memory usage
top -u splunk

iostat -x 1

# Monitor Splunk's internal metrics
splunk search "index=_internal sourcetype=splunk_resource_usage"
```

Mitigation Steps:

```
# Implement rate limiting
splunk edit limits -rate_limit 500
```

```
# Enable firewall rules to block traffic
iptables -A INPUT -p tcp --dport 9997 -m limit --limit 50/s --limit-burst 100 -j ACCEPT
iptables -A INPUT -p tcp --dport 9997 -j DROP

# Increase Splunk's resource allocation
splunk edit server -maxThreads 200
```

---

4. Corrupting or Deleting Logs

Attack Scenario:

Using Splunk queries to delete or alter logs.

Detection Commands:

```
# Check for delete commands
grep "| delete" /opt/splunk/var/log/splunk/splunkd.log

tail -f /opt/splunk/var/log/splunk/audit.log
```

Mitigation Steps:

```
# Restrict log deletion to authorized users
splunk edit user admin -role readonly

# Enable file immutability
chattr +i /opt/splunk/var/log/splunk/*

# Regular backups
splunk backup data -location /backups/splunk
```

---

5. Disrupting Splunk Services

Attack Scenario:

Stopping Splunk services or deleting configurations.

Detection Commands:

```
# Monitor service status
systemctl status Splunkd
ps aux | grep splunk
```

Mitigation Steps:

```
# Auto-restart service
systemctl enable splunk

# Lock config files
chattr +i /opt/splunk/etc/system/local/*

# Create a cron job to restart Splunk
(crontab -l ; echo "* * * * * /opt/splunk/bin/splunk restart") | crontab -
```

---

6. Data Exfiltration via Misconfigured Forwarders

Attack Scenario:

Redirecting logs to an unauthorized Splunk instance.

Detection Commands:

```
# Review forwarder configurations
cat /opt/splunk/etc/system/local/outputs.conf

# Monitor network connections
netstat -an | grep 9997
```

Mitigation Steps:

```
# Encrypt log traffic
echo "sslPassword = securepass" >> /opt/splunk/etc/system/local/outputs.conf

# Allow only trusted forwarders
iptables -A INPUT -p tcp --dport 9997 -s TRUSTED_FORWARDER_IP -j ACCEPT
```

---

7. Rogue App Deployment

Attack Scenario:

Uploading malicious Splunk apps.

Detection Commands:

```
# List installed apps
splunk display app list

# Scan for new app files
find /opt/splunk/etc/apps -type f -mtime -1
```

Mitigation Steps:

```
# Restrict app installations
splunk edit user admin -role limited_access

# Review installed apps regularly
splunk cmd btool apps list
```

---

8. Persistence via Scheduled Searches

Attack Scenario:

Automating malicious actions using scheduled searches.

Detection Commands:

```
# List scheduled searches
splunk search "index=_internal sourcetype=scheduler"
```

Mitigation Steps:

```
# Disable unnecessary scheduled searches
splunk disable savedsearch -name malicious_search
```

---

9. Exploiting Open Ports

Attack Scenario:

Scanning and exploiting open services.

Detection Commands:

```
# Scan for open ports
nmap -p 8000,8089 splunk-server
```

Mitigation Steps:

```
# Close unused ports
iptables -A INPUT -p tcp --dport 8089 -j DROP
```

---

10. Exploiting Unencrypted Communications

Attack Scenario:

Intercepting traffic to manipulate data.

Detection Commands:

```
# Monitor Splunk traffic
sudo tcpdump -i eth0 port 9997
```

Mitigation Steps:

```
# Enable encryption
splunk edit server -sslEnable 1
```