# Scenario 1:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Vulnerable Service Exploitation

**RED TEAM**

**Goals:**

* Attacker should perform enumeration/recon on target
* Attacker should see an service on port 6969 (vulnerable jenkins)
* Attacker should discover that there is no authentication required
* Attacker should then launch MSF and search for the jenkins\_console\_exec
* Attacker should fill out required options and attempt to run the exploit
* exploit should fail so manual exploitation required
* Attacker should then traverse to the “Manage Jenkins” → “Script Console”
* Attacker then should find a reverse shell written in groovy
* connect to reverse shell and running whoami should show NT AUTHORITY\SYSTEM

**Tools:**

* nc
* msf
* nmap

**Blue Team**

**Goal:**

* Detect any exploit attempts against services running
* Check for the suspicious process using tools such as ProcExp and Task Manager, terminate if needed
* Watch the splunk alerts console for incoming alerts.
* *Source=”WinEventLog::security” EventCode=4688 “PowerShell”*

# Scenario 2:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### SQL injection

**RED TEAM**

**Goals:**

* Attacker should run new nmap scan or refer to previous one
* Attacker should go to the DVWA application and navigate to SQLi page
* Attacker should either try manual or or automated exploit using sql
* If using sqlmap capture request using burpsuite
* Attack should be able to dump the user table as well as crack the passwords

**Tools:**

* NMAP
* SQLmap
* *Burpsuite*

**BLUE TEAM**

**Goal**

* Detect SQLi attempts
* Watch the splunk alerts console for incoming alerts.
* On receiving a “Potential SQLi Attempt” alert, review logs in splunk to find any potentially suspicious activity.
* Look for suspicious sql queries

# Scenario 3:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### File Upload

**RED TEAM**

**Goals:**

* Attacker should perform new or refer to previous nmap scan
* Attacker should navigate to to DVWA and the file upload
* Attacker should check that the backend is running php
* Attacker should attempt to upload a file and notice only image files are allowed
* Attacker should change the php reverse shell to php.jpeg
* Attacker should upload file, intercept request and change the file extension back to .php
* catch reverse shell and should be NT AUTHORITY/SYSTEM

**Tools:**

* NMAP
* nc
* *Burpsuite*

**BLUE TEAM**

**Goals**

* Detect malicious file uploads
* Watch the splunk alerts console for incoming alerts.
* On receiving a “Potentially Malicious Upload Access” alert, review logs in splunk to find any potentially suspicious activity.
* Flag any potential malicious files